ADVANCES IN
Sampling Theory and Techniques

ADVANCES IN Sampling Theory and Techniques

Leonid P. Yaroslavsky

SPIE PRESS
Bellingham, Washington USA

Library of Congress Cataloging-in-Publication Data

Names: Yaroslavsky, L. P. (Leonid Pinkhusovich), author.
Title: Advances in sampling theory and techniques / L. Yaroslavsky.
Description: Bellingham, Washington : SPIE, [2020] | Includes bibliographical references and index.
Identifiers: LCCN 2019042348 | ISBN 9781510633834 (paperback) | ISBN 9781510633841 (pdf) | ISBN 9781510633858 (epub) | ISBN 9781510633865 (kindle edition)
Subjects: LCSH: Signal processing–Digital techniques–Mathematics. | Image processing–Digital techniques–Mathematics. | Fourier transformations.
Classification: LCC TK5102.9 .I225 2020 | DDC 621.382/20151952–dc23
LC record available at https://lccn.loc.gov/2019042348

Published by

SPIE
P.O. Box 10
Bellingham, Washington 98227-0010 USA
Phone: +1 360.676.3290
Fax: +1 360.647.1445
Email: books@spie.org
Web: http://spie.org

Printed in the United States of America.
Last updated 5 March 2020
For updates to this book, visit http://spie.org and type “PM315” in the search field.
Cover image courtesy of udi Steinwell.

Contents

Preface

Signal sampling is the major method for converting analog signals into sets of numbers that form digital models of the signals. The key issues in the sampling theory and practice are

- What is the minimal amount of numbers, or what is the minimal sampling rate, sufficient to represent analog signals with a given accuracy?
- What kinds of signal distortions are caused by their sampling?
- What signal attributes determine the minimal sampling rate?
- How can one sample signals with sampling rates close to the theoretical minimum?
- Is it possible to resample sampled signals without introducing additional distortions due to the resampling?
- What are adequate discrete representations of signal transforms, such as convolution and Fourier transforms?

All of these issues are addresed in this book, supplemented by MATLAB® exercises, which you can download via the following link: http:/spie.org/Samples/Pressbook_Supplemental/PM315_sup.zip

Researchers, engineers, and students will benefit from the most updated formulations of the sampling theory, as well as practical algorithms of signal and image sampling with sampling rates close to the theoretical minimum and interpolation-error-free methods of signal/image resampling, geometrical transformations, differentiation, and integration.

Leonid Yaroslavsky
December 2019

Chapter 1
Introduction

1.1 A Historical Perspective of Sampling: From Ancient Mosaics to Computational Imaging

It is easier to grasp a subject by tracing its evolution. Sampling as a mechanism for image formation was first developed by nature in the form of compound and retinal eyes (Figs. 1.1 and 1.2).

Light-sensitive cells (photoreceptors) of the eyes convert the luminosity of small individual areas around different points of observed objects into signals that are sent to the visual cortex of the brain, thus creating an object image.

At the dawn of human culture, ancient people discovered that small pieces of glass, pottery, or small tiles placed together in an appropriate order create a picture (Fig. 1.3). This led to the art of mosaic, which appeared throughout the world. In Byzantium, from the 4th to 14th centuries, it became the leading form of pictorial art.

Ancient artists knew how to intuitively choose the size of the tiles to make good-quality mosaic pictures with the minimal number of tiles. They taught their apprentices that skill; however, no "sampling theory" governed their knowledge, and no need for such a theory existed.

Such a need arose with the creation of the first electrical communication devices. Shortly after the early commercial success of telegraphy in the 1840s, engineers attempted to send more than one signal over a single wire and over increasingly larger distances. In 1854, the first transatlantic telegraph cable project began.[1] The first official telegram to pass between two continents was a letter of congratulations from Queen Victoria of the United Kingdom to the President of the United States, James Buchanan, on August 16, 1858.

However, the signal quality declined rapidly, slowing transmission to an almost unusable speed. The Atlantic Telegraph Company's chief electrician, E. O. W. Whitehouse, decided that the reason for the low speed of transmission was insufficiently high voltage, and so he applied excessive voltage to the cable in the hope of achieving faster operation. The excessive voltage destroyed the cable.[1]

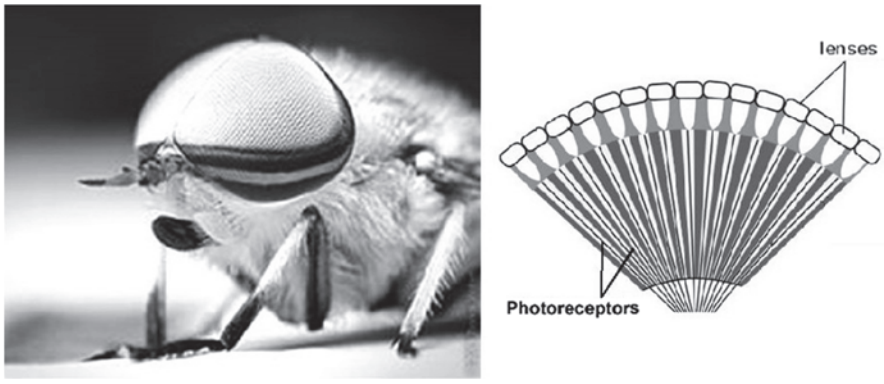


Figure 1.1 Anatomy and structure of compound eyes of insects.

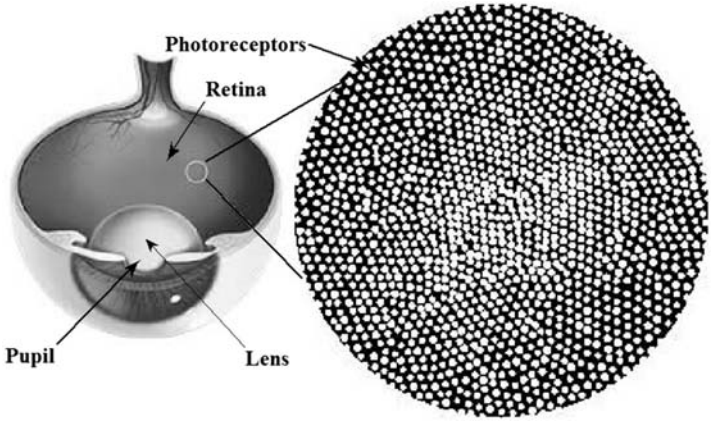


Figure 1.2 The retinal eye of vertebrates.

To explain and solve the problem, the company invited famous physicist Sir William Thomson, who sought to apply his theory of heat propagation. In particular, he used Fourier series to solve differential equations.

After accumulating sufficient experience in the transmission of telegraph and, later, sound and image signals, communication engineers eventually understood that the speed of information transmission is limited by the wave bandwidth of communication channels. This is what one of the founders of communication theory, Dennis Gabor, wrote in his seminal paper[2] in 1944:

> The principle that the transmission of a certain amount of information per unit time requires a certain minimum wave-band width dawned gradually upon communication engineers during the third decade of this (20-th) century. Similarly, as the principle of conservation of energy emerged from slowly hardening conviction of impossibility of perpetuum mobile, this fundamental principle of communication

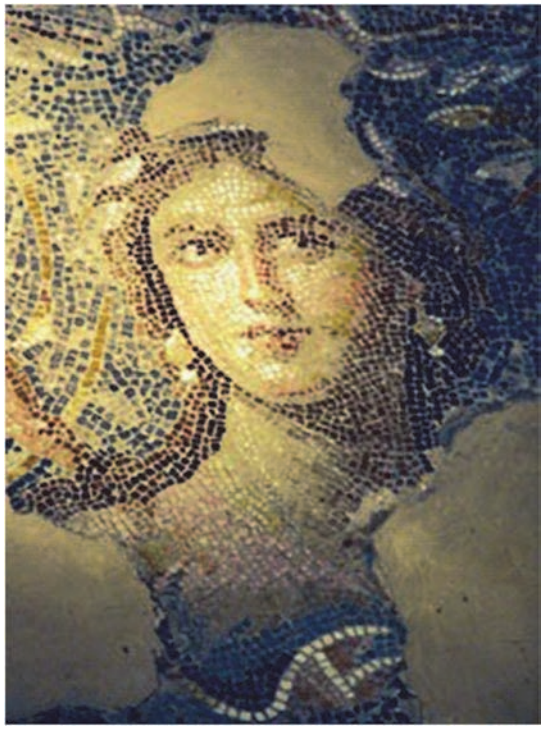


Figure 1.3 *Mona Lisa of the Galilee*, 4th-century Roman mosaic in Sepphoris, Israel. Image courtesy of udi Steinwell.

> engineering arose from the refutation of ingenious attempts to break the as yet unformulated law. When in 1922 John Carson[3] disproved the claim that frequency modulation could economize some of the bandwidth required by amplitude modulation methods, he added that such schemes "are believed to involve some fallacy." This conviction was cast into a more solid shape, when, in 1924, Nyquist[4] and Küpfmüller[5] independently discovered an important special form of the principle by proving that the number of telegraph signals, which can be transmitted over any line, is directly proportional to its waveband width. In 1924 Hartley[6] generalized this and other results, partly by inductive reasoning, and concluded that "the total amount of information, which may be transmitted … is proportional to frequency range which is transmitted and the time which is available for the transmission…"

From this understanding, only one step was left to formulate a mathematical theorem.

In 1933, the young Russian communications engineer Vladimir Aleksandrovich Kotelnikov published a paper[7] in which he formulated the sampling theorem for low-pass and band-pass signals, and he considered the

bandwidth requirements of discrete signal transmission for telegraphy and images. Although Kotelnikov's name later became known in the West as a result of his subsequent work, particularly that on optimal detection, his pioneering 1933 results received little attention at the time outside Russian-speaking areas.

In 1949, Claude Shannon published the paper "Communication in the Presence of Noise," which set the foundation of information theory.[8] In order to formulate his rate distortion theory, Shannon needed a general mechanism for converting an analog signal into a sequence of numbers. This led him to state the classic sampling theorem at the very beginning of his paper, which is one of the theoretical works that has had the greatest impact on modern communications and electrical engineering.

While Shannon deserves full credit for formulating the sampling theorem and for realizing its potential for communication theory and signal processing, he did not claim it as his own. In fact, he said the following about the theorem: "This is a fact which is common knowledge in the communication art." He was also well aware of equivalent forms of the theorem that had appeared in the mathematical literature, i.e., particularly the work of Whittaker.[9]

In subsequent years, sampling theory was further refined and extended in many respects, such as the analysis of errors associated with the sampling and reconstruction of real non-band-limited signals, sampling multi-dimensional signals, interpolation of sampled data, irregular sampling as opposed to regular sampling dictated by the sampling theorem, and signal super-resolution by means of combining several individually sampled, low-resolution copies of a signal.

Perhaps the major achievement of these years was the realization that sampling is a special case of signal discretization by means of their expansion over sets of basis functions of signal transforms and that much more compact discrete representation of analog signals can be achieved if signal transform coefficients, rather than their samples, are used for this purpose. This idea, which can be traced back to Gabor,[2] eventually led to the development of transform signal coding methods such as JPEG and MPEG, without which modern digital audio, video, and TV would not exist.

Transform coding, however, is a two-step process, in which signals are first redundantly sampled in the regular classic way, and then obtained samples are compressed. The compressibility of data acquired by standard sampling motivated attempts to find sampling methods that would not require subsequent compression. These attempts resulted in the concept of "compressed sensing"[10] (or "compressed sampling"[11]), and a number of nonlinear techniques were suggested that allow for the exact reconstruction of signals sampled with speeds lower than that prescribed by the classic theory, under a condition that the signals are sparse in a certain "sparsifying" transform, i.e., if they have only a limited number of non-zero transform coefficients. This concept gained

considerable popularity and initiated a steady flow of publications demonstrating its applicability in different applications.

Although it eventually turned out that the compressed sensing methods still require certain redundancy in signal sampling, the concept of signal sparsity as a generalization of the classic concept of signal-band limitations proved to be very fruitful. It stimulated reformulation of the sampling theory in terms of signal sparsity, the proof that signal sparsity defines the minimal signal sampling rate sufficient for signal reconstruction with the accuracy that corresponds to the level of its sparsity, and the demonstration that this minimal sampling rate can be achieved using the means of computational imaging.[12,13]

1.2 Book Overview

The book is divided into two parts, supplemented with MATLAB-based exercises. The first part (Chapters 2 to 7) is devoted to different aspects of signal sampling.

Chapter 2 begins with the classic formulation (Section 2.1) of the sampling theorem for 1D band-limited signals. It is then extended in Section 2.2 to sampling 1D band-limited band-pass signals and, in Section 2.3, to band-limited 2D signals, for which the concept of optimal 2D regular lattices is introduced. Section 2.4 deals with sampling distortions, which take place when sampling real, non-band-limited signals. In Section 2.5, a realistic reformulation of the sampling theorem is provided that does not assume the band-limitedness of signals and is based on mathematical models of real signal sampling and reconstruction devices, as well as the realization that no precise signal reconstruction from a sampled representation is possible. The last two sections of the chapter address, through two complementary approaches, the problem of evaluating the minimal sampling rate sufficient for signal reconstruction with a given accuracy. In Section 2.6, sampling of signal subband decompositions is considered as a model of sampling with the minimal sampling rate, and in Section 2.7, a formulation of the discrete sampling theorem and its extension to continuous signals are provided for use in Chapters 3 and 4.

Chapter 3 is devoted to demystifying the concept of compressed sensing. First, in Section 3.1, the ubiquitous compressibility of images sampled using the standard regular sampling methods is demonstrated and explained with examples of sampled images. In Section 3.2, the concept of compressed sensing is elucidated with a simple model that demonstrates how and under what conditions can one precisely reconstruct a signal sampled with aliasing, i.e., sampled by violating the sampling theorem. In Section 3.3, the sampling redundancy required by compressed sensing methods for precise signal reconstruction is estimated to evaluate how far these methods are from reaching the minimal signal sampling rates.

In Chapter 4, a method of arbitrary image sampling and bounded spectrum reconstruction is introduced, which allows for image sampling rates close to the minimal rate defined by the sampling theory. Section 4.1 presents a detailed description of the method. In Section 4.2, results of experimental verification of the method are provided. Section 4.3 discusses some important practical issues of implementing the method, such as noise robustness, the choice of standard spectrum-bounding shapes for image reconstruction, practical aspects of image anti-aliasing pre-filtering, and computational complexity of the spectrum-bounded spectrum image reconstruction. In Section 4.4, possible applications of the method are illustrated, such as achieving image super-resolution by means of fusing several low-resolution, stochastically sampled copies, image reconstruction from sparsely sampled or decimated projections, and image reconstruction from sparsely sampled Fourier spectra.

Chapter 5 offers a thorough treatment of the problem of signal and image resampling. The introductory Section 5.1 poses resampling as an interpolation problem and introduces convolutional interpolators. In Section 5.2, the concept of the perfect resampling filter is introduced, and it is shown that discrete sinc interpolation is the gold standard for resampling sampled data. Section 5.3 describes methods for the efficient algorithmic implementation of discrete sinc interpolation for image sub-sampling, fractional shift, and rotation. Section 5.4 presents experimental evidence of the superiority of the discrete sinc interpolation over other convolutional interpolation methods.

Chapter 6 exposes discrete sinc interpolation in other applications and implementations. In Section 6.1, perfect numerical differentiation and integration algorithms are introduced that are capable of precise differentiation and, correspondingly, integration of analog signals presented by their sampled data. In addition, the experimental results of their comparison with traditional numerical differentiation and integration algorithms are presented that demonstrate their paramount superiority. Section 6.2 describes local ("elastic") algorithms for image shift-variant resampling, i.e., resampling according to arbitrary maps of pixel displacement. Section 6.3 presents the application of discrete sinc interpolation for "filtered back-projection" and "direct Fourier method" algorithms of image reconstruction from parallel projections and for projection data "rebinning" for image reconstruction from fan beam projections.

Chapter 7 concludes Part 1. Section 7.1 introduces the discrete uncertainty principle, a discrete counterpart of the classic continuous uncertainty principle, and Section 7.2 demonstrates the existence of discrete signals with sharply bounded support both in signal and Fourier domains, and postulates the existence of discrete orthogonal systems of sharply band-limited basis functions with sharply limited support.

Sampling theory would not be complete without treating the discrete representation of analog signal transformations such as convolution and the integral Fourier transform. Three chapters—8, 9, and 10—address this issue in Part 2 of the book.

In Chapter 8, the basic principles of the discrete representation of signal transformations are formulated, which are used to derive discrete representations of the convolution integral transformation in Chapter 9 and of the integral Fourier transform in Chapter 10.

Chapter 9 is dedicated to the discrete representation of the convolution integral. In Section 9.1, the sampled representation of continuous signals is used to convert the convolution integral into a digital convolution. In Section 9.2, the concept of a continuous filter equivalent to a given digital filter is introduced, and such important characteristics of digital filters are derived as their point spread functions and frequency responses, as well as the point spread functions and frequency responses of the continuous filters that correspond to them. Section 9.3 discusses different methods of treating signal borders in digital convolution.

Chapter 10 is dedicated to the discrete representation of the integral Fourier transform. In Section 10.1, the sampled representation of continuous signals is used to convert the 1D Fourier integral to the canonical 1D discrete Fourier transform (DFT) and its more general modifications, shifted DFT and scaled shifted DFT. In Section 10.2, 2D DFTs are derived in their rotated, scaled, and shifted 2D DFT forms. In Section 10.3, it is shown that mirror signal reflection from its borders converts the DFT into the discrete cosine transform (DCT), a very useful signal transform with a good energy-compaction property. Section 10.4 demonstrates how the DCT can used to implement a boundary-effect-free fast digital convolution. Section 10.5 shows that the frequency responses of digital filters, which are their main characteristic, can be expressed through the DFT coefficients of their point spread functions.

Appendices A1 and A2 contain auxiliary reference materials concerning the properties of the integral Fourier transform and DFT, correspondingly.

Part I
Signal Sampling

Chapter 2
Sampling Theorems

2.1 Kotelnikov–Shannon Sampling Theorem: Sampling Band-Limited 1D Signals

Kotelnikov's and Shannon's classic formulation of the sampling theorem reads as follows.[7,8]

Theorem: If a function of time $a(t)$ contains no frequencies higher than $F/2$ cycles per second (cps), it is completely determined by giving its ordinates at a series of points spaced $1/F$ seconds apart:

$$a(t) = \sum_{k=-\infty}^{\infty} a(k/F) \frac{\sin[\pi F(t - k/F)]}{\pi F(t - k/F)}. \tag{2.1}$$

The proof of this theorem is based on the properties of the integral Fourier transform (Appendix A1). Let $\alpha(f)$ be the Fourier spectrum of $a(t)$:

$$\alpha(f) = \int_{-\infty}^{\infty} a(t) \exp(i2\pi f t) dt \tag{2.2}$$

and

$$\alpha(|f| > F/2) = 0,$$

i.e.,

$$a(t) = \int_{-\infty}^{\infty} \alpha(f) \exp(-i2\pi f t) df = \int_{-F/2}^{F/2} \alpha(f) \exp(-i2\pi f t) df. \tag{2.3}$$

In the interval $[-F/2, F/2]$, the function $\alpha(f)$ can be represented by its Fourier series

$$\alpha(f) = \mathrm{rect}\left(\frac{f}{F}\right) \sum_{k=-\infty}^{\infty} \alpha_k \exp\left(i \frac{2\pi k f}{F}\right), \tag{2.4}$$

where

$$\alpha_k = \frac{1}{F}\int_{-F/2}^{F/2} \alpha(f)\exp\left(-i2\pi\frac{kf}{F}\right) = \frac{1}{F}a\left(\frac{k}{F}\right) \tag{2.5}$$

are Fourier series coefficients of the function, and

$$\mathrm{rect}(x) = \begin{cases} 1, & -1/2 < x < 1/2 \\ 0, & \text{otherwise} \end{cases} \tag{2.6}$$

is a *rectangular window function*.

Therefore,

$$\alpha(f) = \frac{1}{F}\mathrm{rect}\left(\frac{f}{F}\right)\sum_{k=-\infty}^{\infty} a(k/F)\exp\left(i\frac{2\pi kf}{F}\right). \tag{2.7}$$

Signal $a(t)$ can be found through the inverse Fourier transform of its spectrum $\alpha(f)$, as given by Eq. (2.7):

$$\begin{aligned} a(t) &= \int_{-\infty}^{\infty} \alpha(f)\exp(-i2\pi ft)df \\ &= \frac{1}{F}\sum_{k=-\infty}^{\infty} a(k/F)\int_{-\infty}^{\infty} \mathrm{rect}\left(\frac{f}{F}\right)\exp[-i2\pi f(t-k/F)]df \\ &= \frac{1}{F}\sum_{k=-\infty}^{\infty} a(k/F)\int_{-F/2}^{F/2} \exp[-i2\pi f(t-k/F)]df \\ &= \sum_{k=-\infty}^{\infty} a(k/F)\frac{\sin[\pi F(t-k/F)]}{\pi F(t-k/F)}. \end{aligned} \tag{2.8}$$

This proves the theorem.

In order to give a physical interpretation of this result, consider a virtual signal $\tilde{a}(t)$ whose spectrum $\tilde{\alpha}(f)$ is the Fourier series of spectrum $\alpha(f)$ of signal $a(t)$:

$$\tilde{\alpha}(f) = \frac{1}{F}\sum_{k=-\infty}^{\infty} a\left(\frac{k}{F}\right)\exp\left(i\frac{2\pi kf}{F}\right). \tag{2.9}$$

By applying to spectrum $\tilde{\alpha}(f)$ the inverse Fourier transform and using the definition of the *delta function* given by Eq. (A1.40) in Appendix A1, the result is

$$\begin{aligned}\tilde{a}(t) &= \frac{1}{F}\int_{-\infty}^{\infty}\left[\sum_{k=-\infty}^{\infty} a\left(\frac{k}{F}\right)\exp\left(i\frac{2\pi kf}{F}\right)\right]\exp(-i2\pi ft)df \\ &= \frac{1}{F}\sum_{k=-\infty}^{\infty} a\left(\frac{k}{F}\right)\int_{-\infty}^{\infty}\exp\left[-i2\pi f\left(t-\frac{k}{F}\right)\right]df \\ &= \frac{1}{F}\sum_{k=-\infty}^{\infty} a\left(\frac{k}{F}\right)\delta\left(t-\frac{k}{F}\right),\end{aligned} \tag{2.10}$$

i.e., virtual signal $\tilde{a}(t)$ is a sequence of delta functions weighted by samples $\{a(k/F)\}$ of signal $a(t)$ and placed at sampling points $\{k/F\}$. Note that spectrum $\tilde{\alpha}(f)$, as a Fourier series of signal spectrum $\alpha(f)$, is a periodic replication of this spectrum:

$$\tilde{\alpha}(f) = \frac{1}{F}\sum_{k=-\infty}^{\infty} a\left(\frac{k}{F}\right)\exp\left(i\frac{2\pi kf}{F}\right) = \sum_{m=-\infty}^{\infty}\alpha(f+Fm). \tag{2.11}$$

Equations (2.10) and (2.11) imply that

- Signal sampling with sampling interval $\Delta_t = 1/F$ can be treated as generating a virtual signal $\tilde{a}(t)$ whose specrum is the periodic replication of the spectrum of signal $a(t)$ with a replication period $F = 1/\Delta_t$.
- Signal reconstruction from its samples defined by Eq. (2.8) can be treated as the extraction of the signal spectrum $\alpha(f)$ from the periodically replicated spectrum

$$\tilde{\alpha}(f) = \sum_{m=-\infty}^{\infty}\alpha(f+Fm) \tag{2.12}$$

of the virtual discrete signal

$$\tilde{a}(t) = \frac{1}{F}\sum_{k=-\infty}^{\infty} a\left(\frac{k}{F}\right)\delta\left(t-\frac{k}{F}\right) \tag{2.13}$$

by multiplying its periodic spectrum by a rectangular window function $\mathrm{rect}(f/F)$, i.e., by passing the virtual discrete signal $\tilde{a}(t)$ through the ideal low-pass filter with frequency response $\mathrm{rect}(f/F)$ and point-spread function

$$PSF_{ilpf}(t) = \frac{\sin(\pi Ft)}{\pi Ft} = \mathrm{sinc}(\pi Ft), \tag{2.14}$$

which interpolates signal samples using the *sinc function* [defined in Appendix A1.5, Eq. (A1.41)] as an interpolation kernel. This interpolation method is called *sinc interpolation* (illustrated in Fig. 2.1).

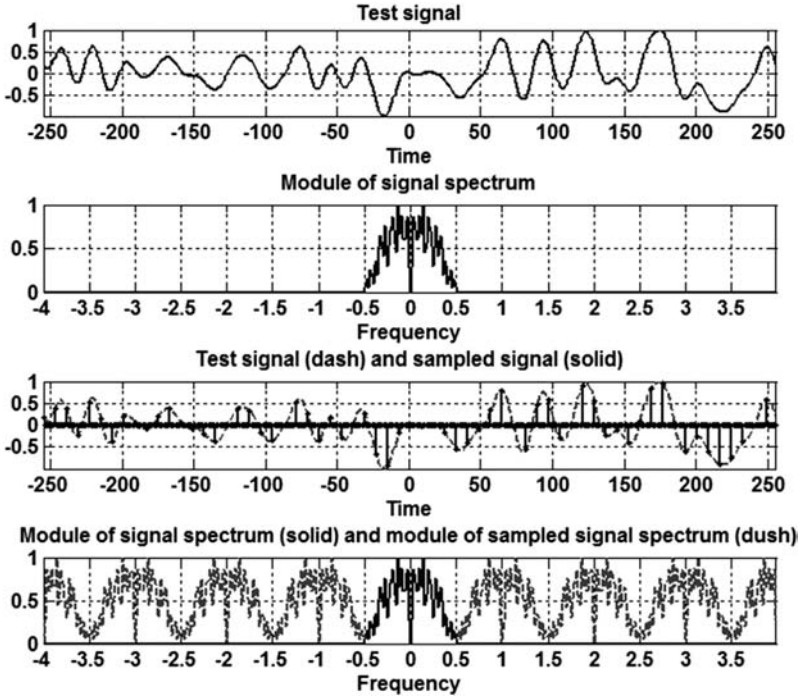


Figure 2.1 Interpretation of signal sampling to generate a virtual discrete signal with a Fourier spectrum formed by the periodically replicated Fourier spectrum of the signal. From top to bottom: a test signal, its Fourier spectrum, a virtual discrete signal comprising samples of the test signal, and its Fourier spectrum.

The sampling theorem implies that the number of samples of band-limited signals with bandwidth $[-F/2, F/2]$ per unit of signal length, i.e., the signal sampling rate, equals F. This rate is called the *Nyquist sampling rate*, a name that was coined by Shannon[8] in recognition of Nyquist's important contributions to communication theory.

The Nyquist sampling rate F is the minimal sampling rate sufficient to reconstruct band-limited signals from their samples. If the signal sampling rate is lower than F, the period of the periodic spectrum replication due to signal sampling will be lower than the spectrum width. Therefore, the signal spectrum will overlap with its periodic replicas and cannot be separated by the ideal low-pass filter without distortions.

2.2 Sampling 1D Band-Pass Signals

1D band-limited signals treated in the previous section are called *baseband signals*. Their Fourier spectrum is concentrated within a bounded interval $[-F/2, F/2]$ around zero frequency. This interval is called the *signal baseband*.

1D band-limited signals whose spectrum is concentrated within intervals $[f_0 - F/2, f_0 + F/2]$ and $[-f_0 - F/2, -f_0 + F/2]$ around a non-zero frequency f_0 are called *band-pass signals*. According to the properties of the Fourier transform (Appendix A1), spectral components or real-valued signals on these intervals are mutually complex conjugates, and a band-pass signal $a_{PB}(t)$ can be regarded as a baseband signal $a_{BB}(t)$ modulated by a sinusoidal signal of frequency f_0:

$$a_{PB}(t) = a_{BB}(t)\sin(2\pi f_0 t). \tag{2.15}$$

Therefore, the classic sampling theorem can be applied to band-pass signals if, before sampling, band-pass signals are converted into corresponding baseband signals by demodulating them with a sinusoidal signal of frequency f_0 and subsequent low-pass filtering within the *baseband* $[-F/2, F/2]$. This band-pass–to–baseband conversion is illustrated in Fig. 2.2.

Correspondingly, the reconstruction of band-pass signals for their sampled representation can be performed by reconstructing their baseband copies by sinc interpolation of their samples and the subsequent modulation-obtained baseband signal with a sinusoidal signal of frequency f_0. This implies

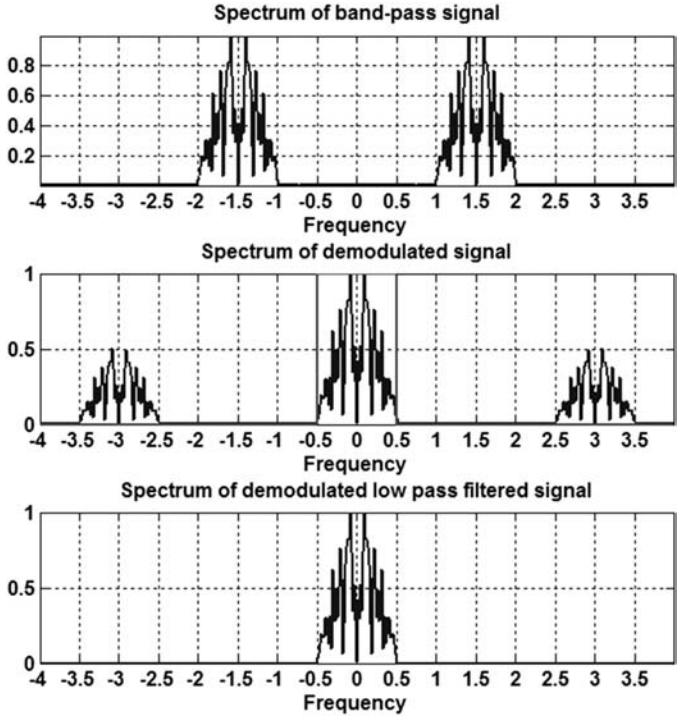


Figure 2.2 From top to bottom: spectra of a band-pass signal and its demodulated version, and a demodulated low-pass filtered version.

that the number of samples per unit of signal length is the same as that for their corresponding baseband signal, i.e., equal to the width F of the baseband signal spectrum.

2.3 Sampling Band-Limited 2D Signals; Optimal Regular Sampling Lattices

The classic 1D sampling theorem can be extended to 2D signals $a(x, y)$ given in Cartesian coordinates (x, y) if their Fourier spectra $\alpha(f_x, f_y)$ are in Cartesian coordinates (f_x, f_y) bounded by a rectangle $[-F_x/2 \le f_x \le F_x/2; -F_y/2 \le f_y \le F_y/2]$. Such signals can be precisely reconstructed from their samples $a(k/F_x, l/F_y)$ taken at nodes $(k/F_x, l/F_y)$ of a regular rectangular sampling lattice with sampling intervals $(1/F_x, 1/F_y)$ by means of their 2D separable sinc interpolation:

$$a(x, y) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} a\left(\frac{k}{F_x}, \frac{l}{F_y}\right) \operatorname{sinc}\left[\pi F_x\left(x - \frac{k}{F_x}\right)\right] \operatorname{sinc}\left[\pi F_y\left(y - \frac{l}{F_y}\right)\right], \tag{2.16}$$

that is, by means of the ideal 2D low-pass filtering a virtual discrete signal composed of signal samples:

$$\tilde{a}(x, y) = \frac{1}{F_x F_y} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} a\left(\frac{k}{F_x}, \frac{k}{F_y}\right) \delta\left(x - \frac{k}{2F_x}\right) \delta\left(y - \frac{l}{F_y}\right). \tag{2.17}$$

In such a sampling, the number of samples per unit of signal area equals the area $(F_x \times F_y)$ of the rectangle that bounds the signal spectrum. This is the minimum number of samples per unit of signal area that suffices for the exact reconstruction of a signal with a Fourier spectrum bounded by rectangle $[-F_x/2 \le f_x \le F_x/2; -F_y/2 \le f_y \le F_y/2]$.

Regular sampling at nodes of rectangular sampling lattices is the simplest implementation of sampling 2D signals. It has become the standard for the discrete representation of images and the assumed default in image processing software and image displays. It is also the base for building discrete representations of signal transformations (Chapters 8–10). However, it secures the minimal sampling rate only for images with a spectrum bounded by a rectangle. If the image spectrum is bounded by another figure, other sampling lattices will optimally minimize the image sampling rate. Note that whatever regular periodic sampling lattice is used always corresponds to a reciprocal periodic replication of the image spectra and that the sampling rate always equals the area of repeated elements of the periodic spectral pattern. Therefore, minimization of the sampling rate requires maximization of the density of the spectra packed in their periodic replication.

Two of the most immediate examples of spectral bounding figures other than a rectangle are a superellipse (Fig. 2.3) and a circle (Fig. 2.4).

The first case conforms to the properties of human vision, which is less sensitive to diagonal spatial frequencies then to horizontal and vertical ones. The area of the 45° rotated square (Fig. 2.3(b)), which circumscribes the superellipse and is the element of the periodic spectral pattern (Fig. 2.3(d)) that corresponds to the 45° rotated sampling lattice, is two times less than the area of the square (Fig. 2.3(a)) with vertical and horizontal sides, which is the element of the periodic spectral pattern (Fig. 2.3(c)) that corresponds to the canonic rectangular sampling lattice. Therefore, using a 45° rotated rectangular sampling lattice instead of the canonic vertical/horizontal version yields a twofold reduction of the image sampling rate. This method of image sampling and display has found an application in the print industry (Fig. 2.3(g)).

The second case conforms to natural images that have isotropic spectra. For sampling such images, the hexagonal sampling lattice shown in Fig. 2.4(f)) secures the densest packing of periodically replicated copies of image spectra (Fig. 2.4(d)) and yields a 13.4% reduction of the sampling rate with respect to the canonic sampling over the rectangular sampling lattice. This estimate is a relative difference between the area $4R^2$ of the square (Fig. 2.4(a)) that circumscribes a circle of radius R and the area $2\sqrt{3}R^2 \approx 3.46R^2$ of the hexagon (Fig. 2.4(b)) that circumscribes the same circle.

There are numerous examples of hexagonal arrangements of light-sensitive cells in the eyes of animals and humans (see, for instance, Figs. 1.1 and 1.2), because they place the maximum number of cells of a certain diameter in a given area. Hexagonal sampling lattices are also used in the print industry for color printing and in some image displays.

2.4 Sampling Real Signals; Signal Reconstruction Distortions due to Spectral Aliasing

The classic sampling theorem holds for band-limited signals. In reality, the Fourier spectra of signals and images are never band-limited; they only more or less rapidly decay with growing frequency. Therefore, precise signal reconstruction from samples is never possible because signal sampling inevitably introduces distortions in reconstructed signals.

These distortions are associated with the phenomenon of *spectral aliasing*: due to the signal spectrum periodic replication caused by sampling, replicas of two complex conjugated signal spectral components with frequencies $-F/2 - \Delta f$ and $F/2 + \Delta f$, which are outside the signal baseband $[-F/2, F/2]$, penetrate the baseband with frequencies $-F/2 - \Delta f + F = F/2 - \Delta f$ and $F/2 + \Delta f - F = -F/2 + \Delta f$. For 2D

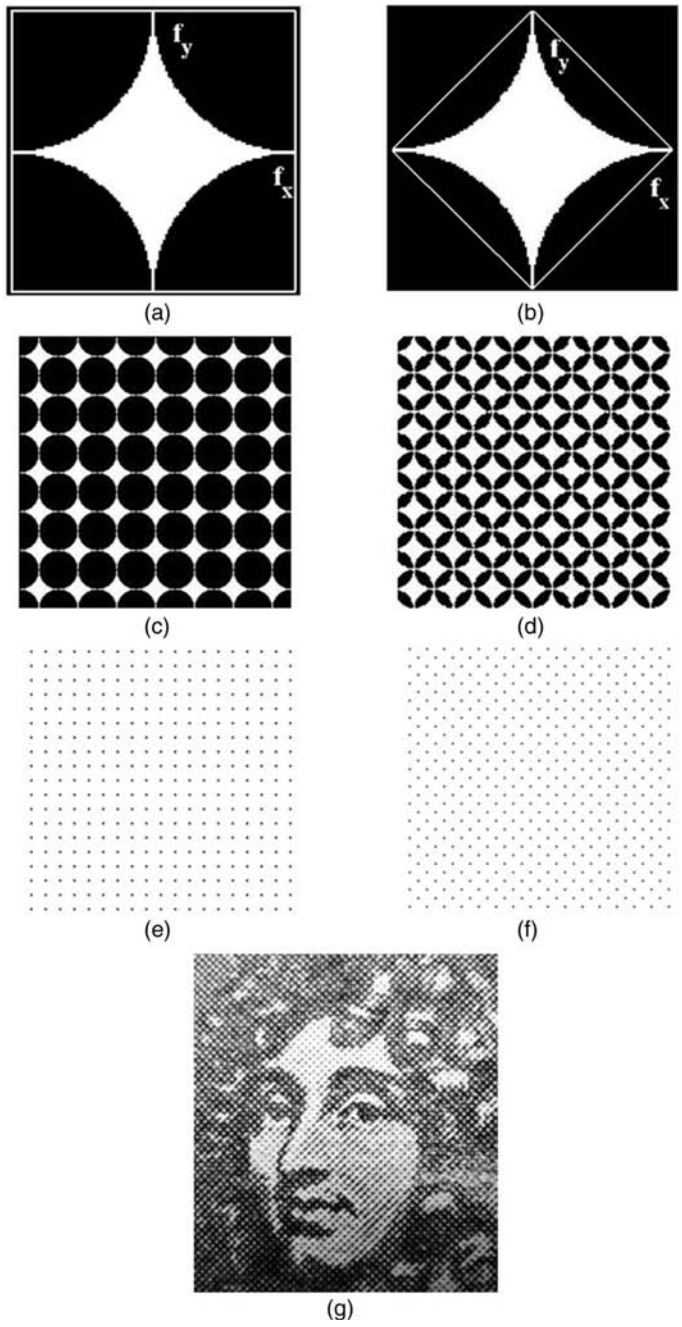


Figure 2.3 Spectrum bounding shape "superellipse" (a) inscribed in a square with horizontal and vertical sides and (b) inscribed in a 45° rotated square; (c) patterns of spectra periodically replicated in vertical and horizontal directions and (d) in 45° rotated directions; (e) rectangular sampling lattice and (f) 45° rotated rectangular sampling lattice; and (g) an example of a rastered image sampled in a 45° rotated rectangular sampling lattice.

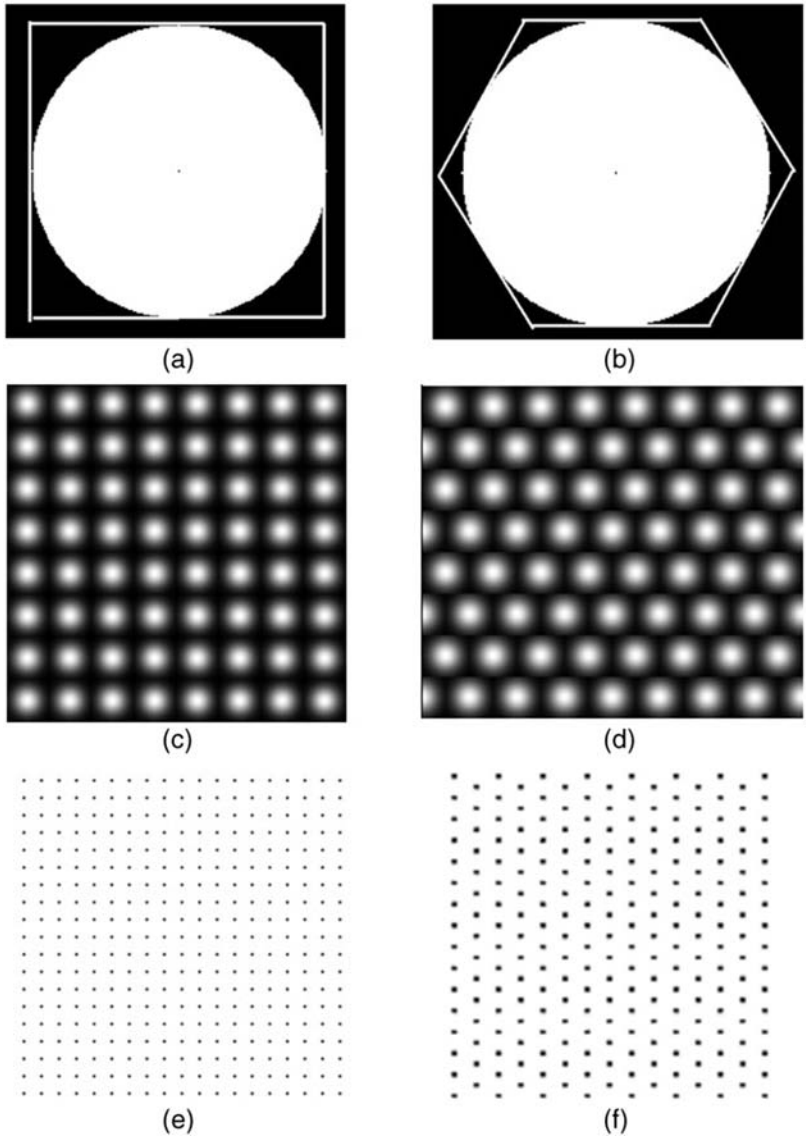


Figure 2.4 Circular spectrum bounding shape inscribed (a) in a square and (b) in a hexagon; circular spectra periodically replicated (c) in rectangular and (d) in hexagonal coordinates; and corresponding (e) rectangular and (f) hexagonal sampling lattices.

signals, this phenomenon is illustrated in Fig. 2.5, generated by the program **Aliasing_2D_SPIE.m** (see Exercises).

With respect to sinusoidal signals, the lowering of the signal frequency due to its sub-Nyquist sampling is called the *strobe effect*. Figures 2.5 and 2.6 illustrate spectral aliasing and the strobe effect for a

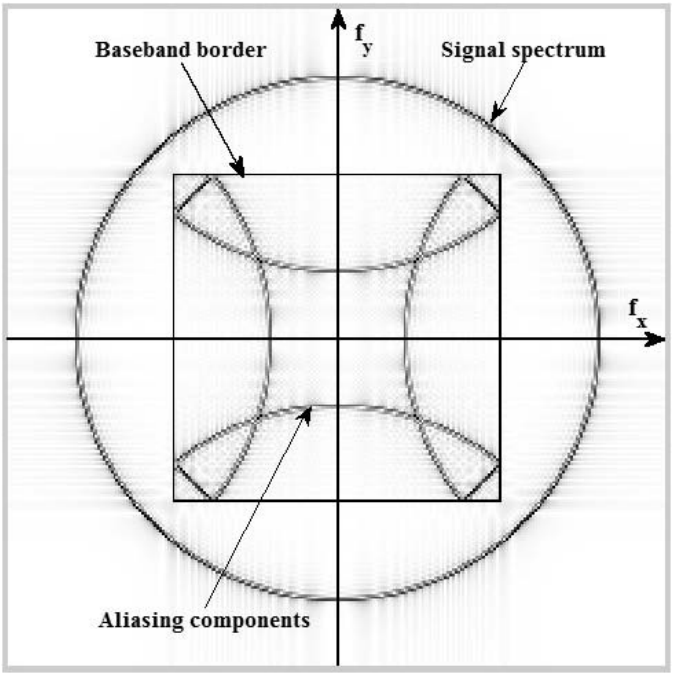


Figure 2.5 Spectrum of a signal sampled with aliasing due to sub-Nyquist sampling.

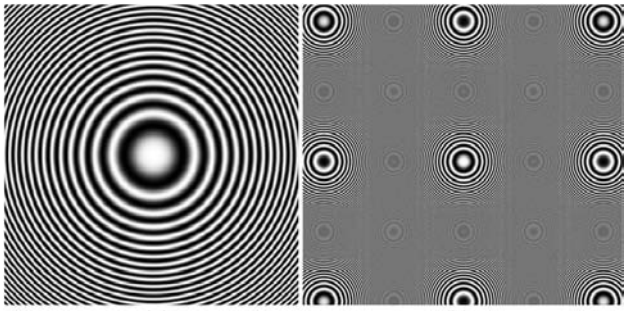


Figure 2.6 Strobe effect in sampling and reconstruction of two 2D chirp signals, (a) in which local spatial frequencies do not exceed the Nyquist sampling frequency, and (b) in which local spatial frequencies exceed the Nyquist sampling frequency and is reconstructed with the strobe effect.

2D sinusoidal signal $a(x, y) = \cos(2\pi f\sqrt{x^2 + y^2})$ and for a 2D chirp signal $a(x, y) = \cos[2\pi f(x^2 + y^2)]$, respectively.

To avoid these distortions, the signal should be subjected to the ideal low-pass pre-filtering before sampling, which will convert the signal to its band-limited approximation. By virtue of Parseval's relation (Appendix A1.3),

the *mean squared error* (MSE) σ^2 of this approximation equals the signal spectrum $\alpha(f)$ energy outside the sampling baseband:

$$\sigma^2 = 2 \int_{F/2}^{\infty} |\alpha(f)|^2 df. \tag{2.18}$$

If the ideal low-pass filter is used for signal reconstruction from samples, as dictated by the sampling theorem, this error is the minimal error caused by signal sampling. If, however, the signal reconstruction filter is not the ideal low-pass filter, i.e., if the signal reconstruction filter passes some signal components outside the baseband $[-F/2, F/2]$, then additional distortions occur. In the case of sinusoidal signals with frequencies $\pm(F/2 - \Delta f)$, these additional distortions appear to beat with a frequency Δf of these signals with aliasing components of frequencies $\pm(F/2 + \Delta f)$. This effect is called the *moiré effect* (Fig. 2.7).

Appropriate image pre-sampling filtering and interpolation of samples is crucial for the readability of images reconstructed from their samples, as illustrated in Fig. 2.8.

2.5 The Sampling Theorem in a Realistic Reformulation

The classic sampling theorem is based on idealistic assumptions of the existence of band-limited signals and ideal low-pass filters. However, there are no ideal band-limited signals, just as there are no ideal low-pass filters. Therefore, no precise signal reconstruction from a sampled representation is possible, and one can only try to minimize signal reconstruction errors. This section provides a realistic reformulation of the sampling theorem that directly addresses these issues. The approach is based on the following mathematical model of real sampling and reconstruction devices.

Signal sampling devices produce samples $\{a_k^{(s)}\}$ of signal $a(x)$ as samples

$$a_k^{(s)} = \int_X a(\xi) PSF^{(s)}(x - \xi) d\xi \Big|_{x=k\Delta_x} \tag{2.19}$$

of the result of convolution of the signal with the point spread function (PSF) $PSF^{(s)}(x)$ of the signal sampling device, i.e., of the result of filtering the signal by signal sampling device. The samples are taken with equidistant intervals Δ_x, called the signal *sampling interval*, at points $\{k\Delta_x\}$, where $\{k\}$ are integer indices of the samples. The superscript (s) in $\{a_k^{(s)}\}$ indicates that the signal samples are not instantaneous values of the signal itself but rather instantaneous samples of this signal filtered by the sampling device.

Signal reconstruction from samples is carried out in signal reconstruction devices by means of the interpolation

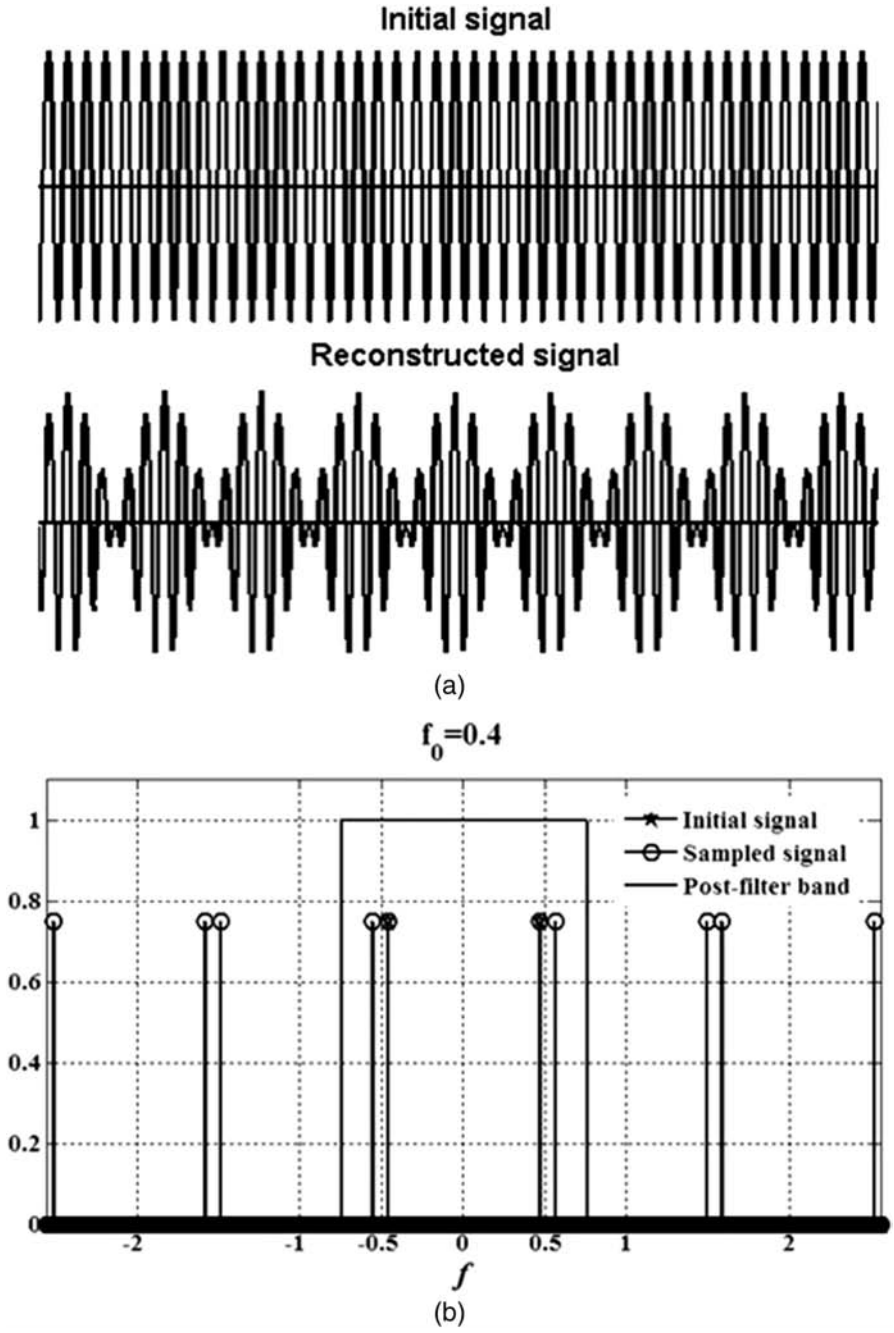


Figure 2.7 Moiré effect in sampling and reconstruction of a sinusoidal signal: (a) test signal of frequency $f_0 = 0.4$ (fraction of the sampling baseband width) and this signal sampled and reconstructed; (b) spectra of the test and sampled test signals for the case where the bandwidth of the reconstruction filter [−0.6, 0.6] exceeds the baseband [−0.5, 0.5] defined by the signal sampling rate.

Figure 2.8 Results of the computer-simulated reconstruction of an image from its samples for different pre-sampling filtering and reconstruction interpolations of samples: (top left) test image oversampled to imitate a non-sampled image; (top right) image reconstructed by means of nearest-neighbor interpolation of its samples obtained by sampling with steps 2×2 with no pre-sampling filtering; (bottom left) image reconstructed by means of nearest-neighbor interpolation of its samples obtained by sampling an image with steps 2×2 after pre-sampling filtering by means of computing the local means over 2×2 samples; (bottom right) image reconstructed by means of sinc interpolation of its samples obtained by sampling a test image with steps 2×2 after low-pass pre-sampling filtering.

$$\bar{a}(x) = \sum_k a_k^{(s)} PSF^{(r)}(x - k\Delta_x) \tag{2.20}$$

of signal samples $\{a_k\}$ by the point spread function $PSF^{(r)}(x)$ of the signal reconstruction device as an interpolation kernel.

Note that this formulation mades a deliberate, simplifying assumption that the coordinate of the signal sample with index $k = 0$ is $x = 0$ in both

signal sampling and reconstruction devices, and that the sampling intervals of image sampling and reconstruction devices are identical. In reality, there might be an arbitrary shift of the signal sampling lattice with respect to the sampling- and reconstruction-device coordinate systems and arbitrary scaling of the sampled and reconstructed images. However, in the analysis of signal sampling, possible arbitrary shifts and scaling play no essential role and can be disregarded. Appendix A2 will show that the possible shifts of sampling positions in coordinate systems of signal sampling and reconstruction devices, as well as image scaling, must be taken into account in discrete representations of the integral Fourier transform.

Generally, integration over ξ in Eq. (2.19), as well as summation over k in Eq. (2.20), are performed within signal boundaries. Without violating the generality, one can, in order to simplify further mathematical analysis, rewrite integration in Eq. (2.19) in infinite limits:

$$a_k^{(s)} = \int_{-\infty}^{\infty} a(\xi) PSF^{(s)}(x-\xi)d\xi \Big|_{x=k\Delta_x} \tag{2.21}$$

assuming that signal $a(\xi)$ equals zero outside the real intervals of support. Accordingly, this section assumes that an unlimited number of signal samples are available for signal reconstruction and rewrites Eq. (2.20) as

$$a^{(r)}(x) = \sum_{k=-\infty}^{\infty} a_k^{(s)} PSF^{(r)}(x-k\Delta_x). \tag{2.22}$$

Now modify Eqs. (2.19) and (2.21) in the following way using the delta function $\delta(.)$ and its properties defined and described in Appendix A1:

$$\begin{aligned} a_k^{(s)} &= \int_{-\infty}^{\infty} a(\xi) PSF^{(s)}(x-\xi)d\xi \Big|_{x=k\Delta_x} \\ &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} a(\xi) PSF^{(s)}(x-\xi)d\xi \right] \delta(k\Delta x - x)dx \\ &= \int_{-\infty}^{\infty} a^{(s)}(x)\delta(k\Delta x - x)dx, \end{aligned} \tag{2.23}$$

$$\begin{aligned} a^{(r)}(x) &= \sum_{k=-\infty}^{\infty} a_k PSF^{(r)}(x-k\Delta_x) = \sum_{k=-\infty}^{\infty} a_k \int_{-\infty}^{\infty} PSF^{(r)}(x-\xi)\delta(\xi-k\Delta_x)d\xi \\ &= \int_{-\infty}^{\infty} \left[\sum_{k=-\infty}^{\infty} a_k^{(s)} \delta(\xi-k\Delta_x) \right] PSF^{(r)}(x-\xi)d\xi = \int_{-\infty}^{\infty} \tilde{a}^{(s)}(\xi) PSF^{(r)}(x-\xi)d\xi. \end{aligned} \tag{2.24}$$

These representations introduce "virtual" signals

$$a^{(s)}(x) = \int_{-\infty}^{\infty} a(\xi) PSF^{(s)}(x - \xi) d\xi \tag{2.25}$$

and

$$\tilde{a}^{(s)}(x) = \sum_{k=-\infty}^{\infty} a_k^{(s)} \delta(x - k\Delta_x). \tag{2.26}$$

Signal $a^{(s)}(x)$ is the initial signal $a(x)$ filtered by an image sampling device before sampling it. Signal $\tilde{a}^{(s)}(x)$ is a virtual continuous signal comprising samples $\{a_k^{(s)}\}$ of signal $a^{(s)}(x)$.

Further analysis of signal sampling is simplified in the Fourier transform domain. Compute the Fourier spectra of virtual signals $a^{(s)}(x)$ and $\tilde{a}^{(s)}(x)$. According to the convolution theorem (Appendix A1), the spectrum $\alpha^{(s)}(f)$ of signal $a^{(s)}(x)$ (Eq. (2.25)), whose instantaneous values $\{a_k^{(s)}\}$ form the sampled representation of signal $a(x)$, is thus a product

$$\alpha^{(s)}(f) = \alpha(f) FR^{(s)}(f) \tag{2.27}$$

of the spectrum $\alpha(f)$ of the signal $a(x)$ and frequency response $FR^{(s)}(f)$ of the sampling device, and the Fourier transform of its point spread function $PSF^{(s)}(x)$:

$$FR^{(s)}(f) = \int_{-\infty}^{\infty} PSF^{(s)}(x) \exp(i2\pi f x) dx. \tag{2.28}$$

The Fourier spectrum of the virtual signal $\tilde{a}^{(s)}(x)$ (Eq. (2.26)) involved in the reconstruction of signal $a(x)$ from its sampled representation $\{a_k^{(s)}\}$ can be, according to the definition of the delta function, found as

$$\begin{aligned}
\tilde{\alpha}^{(s)}(f) &= \int_{-\infty}^{\infty} \tilde{a}^{(s)}(x) \exp(i2\pi f x) dx \\
&= \int_{-\infty}^{\infty} \left[\sum_{k=-\infty}^{\infty} a_k^{(s)} \delta(x - k\Delta_x) \right] \exp(i2\pi f x) dx = \sum_{k=-\infty}^{\infty} a_k^{(s)} \exp(i2\pi f k \Delta_x) \\
&= \sum_{k=-\infty}^{\infty} \exp(i2\pi f k \Delta_x) \int_{-\infty}^{\infty} \alpha^{(s)}(p) \exp(-i2\pi p k \Delta_x) dp \\
&= \int_{-\infty}^{\infty} \alpha^{(s)}(p) \sum_{k=-\infty}^{\infty} \exp[i2\pi(f - p) k \Delta_x] dp.
\end{aligned} \tag{2.29}$$

By virtue of the Poisson summation formula (Eq. (A1.46)),

$$\sum_{k=-\infty}^{\infty} \exp[i2\pi(f-p)k\Delta_x] = \frac{1}{\Delta_x}\sum_{m=-\infty}^{\infty} \delta\left(f-p-\frac{m}{\Delta_x}\right). \tag{2.30}$$

The following is then obtained:

$$\tilde{\alpha}^{(s)}(f) = \frac{1}{\Delta_x}\int_{-\infty}^{\infty}\sum_{m=-\infty}^{\infty} \alpha^{(s)}(p)\delta\left(f-p-\frac{m}{\Delta_x}\right)dp = \frac{1}{\Delta_x}\sum_{m=-\infty}^{\infty} \alpha^{(s)}\left(f-\frac{m}{\Delta_x}\right), \tag{2.31}$$

i.e.,

$$\tilde{\alpha}^{(s)}(f) = \frac{1}{\Delta_x}\sum_{m=-\infty}^{\infty} \alpha^{(s)}\left(f-\frac{m}{\Delta_x}\right) = \frac{1}{\Delta_x}\sum_{m=-\infty}^{\infty} \alpha\left(f-\frac{m}{\Delta_x}\right)FR^{(s)}\left(f-\frac{m}{\Delta_x}\right). \tag{2.32}$$

Equation (2.32) means that the Fourier spectrum $\tilde{\alpha}^{(s)}(f)$ of the virtual signal $\tilde{a}^{(s)}(x)$ consists of periodic replicas of the spectrum $\alpha^{(s)}(f)$ of signal $\alpha^{(s)}(f)$.

This analysis reveals that

- Signal sampling devices can be treated as a combination of a linear filter characterized by its point spread function $PSF^{(s)}(x)$ and a sampling unit that takes instantaneous values $\{a_k^{(s)}\}$ of a "pre-filtered" signal $a^{(s)}(x)$ at the filter output.
- Signal sampling can be interpreted as converting a pre-filtered input signal $a^{(s)}(x)$ into a virtual signal $\tilde{a}^{(s)}(x)$ (Eq. (2.26)), whose Fourier spectrum consists of periodically replicated replicas (with replication period $1/\Delta_x$) of the initial signal spectrum modified by the frequency response of a signal sampling device (Eq. (2.32)). Signal sampling interval Δ_x defines the signal baseband $BB = [-1/2\Delta_x, 1/2\Delta_x]$.
- Signal reconstruction from its sampled representation can be treated as a "post-filtering" of the virtual discrete signal $\tilde{a}^{(s)}(x)$ (Eq. (2.26)) in the reconstruction device characterized by its point spread function $PSF^{(r)}(\cdot)$:

$$a^{(r)}(x) = \int_{-\infty}^{\infty} \tilde{a}^{(s)}(\xi)PSF^{(r)}(x-\xi)d\xi. \tag{2.33}$$

This "post-filtering" can be described in the Fourier transform domain as the multiplication

$$\alpha^{(r)}(f) = \tilde{\alpha}^{(s)}(f) FR^{(r)}(f) \tag{2.34}$$

of the spectrum $\tilde{\alpha}^{(s)}(f)$ of the virtual signal $\tilde{a}^{s}(x)$ by the frequency response $FR^{(r)}(f)$ of the signal reconstruction device:

$$FR^{(r)}(f) = \int_{-\infty}^{\infty} PSF^{(r)}(x) \exp(i2\pi f x) dx. \tag{2.35}$$

The described interpretation of signal sampling and signal reconstruction from its samples is illustrated by flow diagrams in Fig. 2.9. Fourier domain representation of signal sampling and reconstruction is illustrated in Fig. 2.10.

Distortions of the reconstructed signal compared to the initial non-sampled signal are due to the following reasons:

- Modifications of the signal spectrum by its pre-filtering in the sampling device (Eq. (2.27)).
- Penetrating tails of periodic replicas of the pre-filtered signal spectrum into the signal baseband, which causes spectral aliasing.
- Fragments of the signal spectrum periodic replicas outside the signal baseband not perfectly filtered out by the "post-filtering" in the reconstruction device.

Therefore, signal distortions due to sampling can be minimized in terms of the signal-reconstruction mean square error (MSE) if signal pre-filtering and post-filtering are performed by the ideal low-pass filters with frequency responses $FR^{(s)}(f) = \mathrm{rect}(f\Delta_x)$ and $FR^{(r)}(f) = \Delta_x \mathrm{rect}(f\Delta_x)$, respectively. In this case, the MSE of signal reconstruction is

$$MSE_{\mathrm{Rec}} = 2 \int_{1/2\Delta_x}^{\infty} |\alpha(f)|^2 df. \tag{2.36}$$

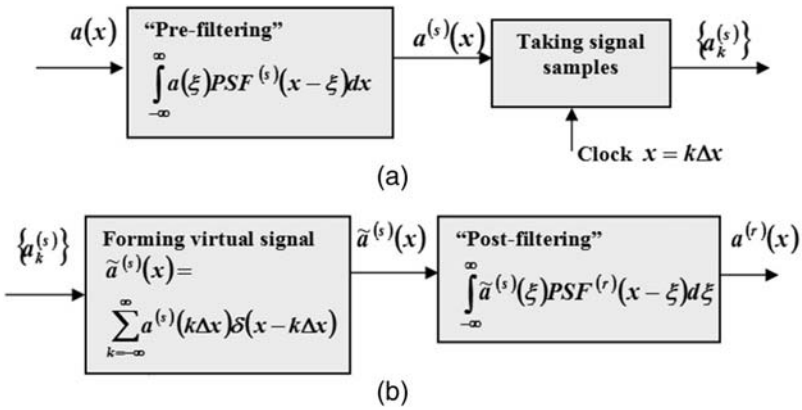


Figure 2.9 Flow diagram of (a) signal sampling and (b) reconstruction.

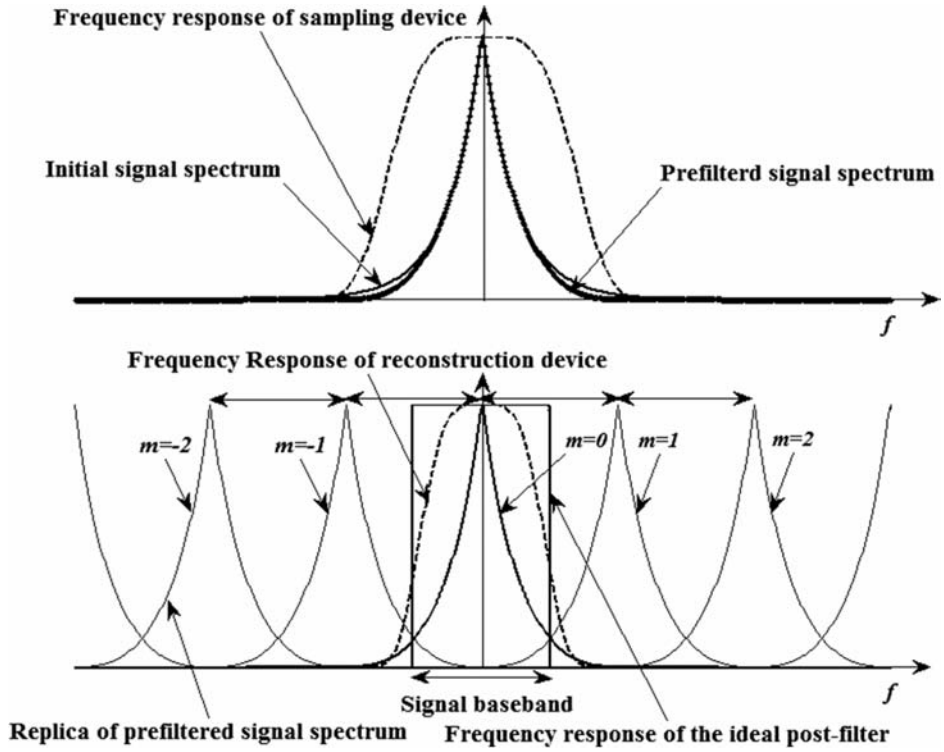


Figure 2.10 Fourier domain representation of signal sampling and reconstruction. The upper plot shows the initial signal spectrum, its spectrum after "pre-filtering" in the sampling device, and the frequency response of the sampling device. The lower plot shows periodic replicas of a sampled signal spectrum (Eq. (2.32)) and frequency responses of the signal reconstruction device and of the ideal low-pass reconstruction filter.

In the ideal case of band-limited signals with no spectral components outside the baseband, the signal is reconstructed precisely ($MSE_{\mathrm{Rec}} = 0$).

The above represents a realistic reformulation of the classic sampling theorem. Its extension to 2D signals is straightforward in its assumption of signal sampling over a rectangular sampling lattice.

Let (Δ_x, Δ_y) be 2D signal sampling intervals in signal Cartesian coordinates (x, y), and $PSF^{(s)}(x, y)$ and $PSF^{(r)}(x, y)$ be point spread functions of the signal sampling and reconstruction devices, respectively. Then

$$a_{k,l}^{(s)} = \int_{-\infty}^{\infty} a(x, y) PSF^{(s)}(x - k\Delta_x, y - l\Delta_y) dxdy \tag{2.37}$$

and

$$a^{(r)}(x, y) = \sum_{k=-\infty}^{\infty} a_{k,l}^{(s)} PSF^{(r)}(x - k\Delta_x, y - l\Delta_y) \tag{2.38}$$

are signal samples, and the reconstructed signal from samples

$$a^{(s)}(x, y) = \int_{-\infty}^{\infty} a(\xi, \eta) PSF^{(s)}(x - \xi, y - \eta) d\xi d\eta, \tag{2.39}$$

$$\tilde{a}^{(s)}(x, y) = \sum_{k=-\infty}^{\infty} a_{k,l}^{(s)} \delta(x - k\Delta_x) \delta(x - k\Delta_x), \tag{2.40}$$

$$\begin{aligned} \tilde{\alpha}^{(s)}(f_x, f_y) &= \frac{1}{\Delta_x \Delta_y} \sum_{m=-\infty}^{\infty} \alpha^{(s)}\left(f_x - \frac{m}{\Delta_x}, f_y - \frac{n}{\Delta_x}\right) \\ &= \frac{1}{\Delta_x \Delta_y} \sum_{m=-\infty}^{\infty} \alpha\left(f_x - \frac{m}{\Delta_x}, f_y - \frac{n}{\Delta_x}\right) FR^{(s)}\left(f_x - \frac{m}{\Delta_x}, f_y - \frac{n}{\Delta_x}\right) \end{aligned} \tag{2.41}$$

are the virtual pre-sampling signal and the virtual input signal of the signal reconstruction device and its Fourier spectrum, respectively. Then,

$$MSE_{\text{Rec}} = \int_{-\infty}^{\infty} \int_{-\infty}^{-\infty} |\alpha(f_x, f_y)|^2 df_x df_y - \int_{-1/2\Delta_x}^{1/2\Delta_x} \int_{-1/2\Delta_y}^{1/2\Delta_y} |\alpha(f_x, f_y)|^2 df_x df_y \tag{2.42}$$

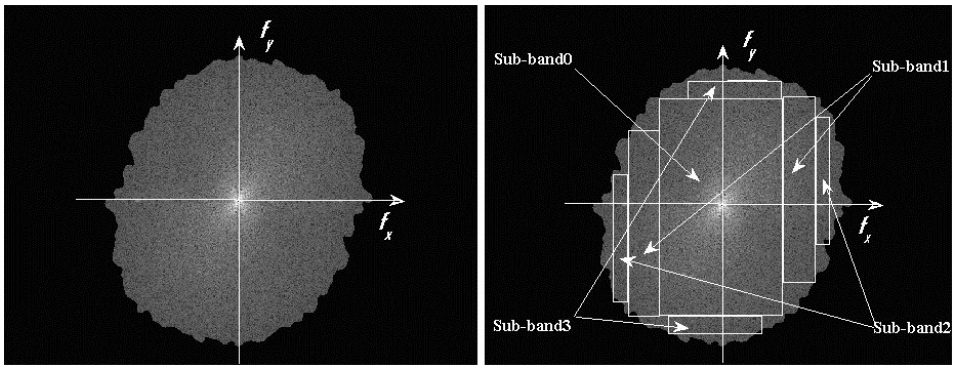
is the minimal MSE of signal reconstruction from its samples that assumes ideal low-pass signal pre-filtering before sampling and post-filtering during reconstruction from its samples.

2.6 Image Sampling with a Minimal Sampling Rate by Means of Image Sub-band Decomposition

The fundamental question of the sampling theory is this: what is the minimal rate of image sampling sufficient for image reconstruction from sampled data with a given accuracy?

Assume that the accuracy of signal reconstruction from its sample representation is evaluated in terms of the MSE of reconstruction. Define a zone in the image Fourier spectrum that contains the largest spectral components, which reconstructs the image with a given MSE; this zone will be called the energy compaction zone (EC zone). Borders of the image-spectrum EC zone bound the image spectrum at the level of the given MSE. In this sense, given their reconstruction MSE, images can be considered with a bounded spectrum.

Consider the image shown in Fig. 2.11(a). Split the figure of this spectrum into a set of M rectangles until they cover the largest possible portion of the



(a) (b)

Figure 2.11 (a) Bounded spectrum of an image and (b) its sub-bands.

area of the spectrum (Fig. 2.11(b)). These rectangles designate fractions of the image spectrum called spectrum sub-bands, and each one corresponds to an image, a component of the initial image. Decomposition of an image into a set of images that correspond to the image sub-bands is called *sub-band decomposition*.

Of all the sub-bands shown in Fig. 2.11(b), one is a baseband image (sub-band-0), and the rest are band-pass images. Sub-band-0 image can be sampled with minimal sampling rate equal to the area, which sub-band-0 occupies in entire image spectrum. Other sub-band images should be "demodulated" to bring their spectra to the baseband, as described in Section 2.2, and then they can be sampled with a minimal sampling rate that equals their corresponding areas in the image spectrum. Thus, the minimal total number of samples per unit of image area sufficient for reconstruction of the entire image with an MSE defined by the covered portion of the image spectrum equals to the sum of the minimal numbers of samples per unit of image area for sub-bands. In the limit, when $M \to \infty$, the sub-band decomposition covers the entire image bounded spectrum, i.e., its spectrum EC zone. Therefore, the lower bound of the image sampling rate sufficient for image reconstruction with a given MSE equals the area S_{Secz} of its spectrum energy compaction zone, defined by the reconstruction MSE. This statement is the essence of the sampling theorem in its realistic formulation.

As opposed to the case where images have a spectrum bounded by a rectangle, for which the classic sampling theorem holds, sampling sub-band-decomposed images results in a set of samples of its sub-band components rather than in a set of samples of the image itself.

Image sub-band decomposition has been applied to image compression in the form of multi-scale sampling using wavelets (see, for instance,[14] Chapter 7).

2.7 The Discrete Sampling Theorem and Its Generalization to Continuous Signals

2.7.1 Theorem formulation

This section supports the above formulated statement of the sampling theorem that the minimal sampling rate of signals per unit of its size sufficient for signal reconstruction with a given MSE σ^2 equals the size S_{Secz} of the signal spectral EC zone that contains a $(E-\sigma^2)/E$ fraction of the signal energy by an alternative approach based on the the discrete sampling theorem and its generalization to continuous signals. This approach assumes direct signal sampling and opens a way to practical algorithms that allow one to reach the minimal rate.

Consider the following discrete model. Let $\mathbf{A}_N$ be a vector of N samples $\{a_k\}_{k=0,\ldots,N-1}$ of a discrete signal, Φ_N be an $N \times N$ orthonormal transform matrix, composed of orthonormal basis functions $\{\varphi_r(k)\}$

$$\Phi_N = \{\varphi_r(k)\}_{r=0,1,\ldots,N-1}, \tag{2.43}$$

and Γ_N be a vector of signal transform coefficients $\{\gamma_r\}_{r=0,\ldots,N-1}$ such that

$$\mathbf{A}_N = \Phi_N \Gamma_N = \left\{ \sum_{r=0}^{N-1} \gamma_r \varphi_r(k) \right\}_{k=0,1,\ldots N-1}. \tag{2.44}$$

Assume that only $K < N$ signal samples $\{a_{\tilde{k}}\}_{\tilde{k} \in \tilde{\mathbf{K}}}$ are available, where $\tilde{\mathbf{K}}$ is a K-size subset $\{\tilde{k}\}$ of indices $\{0,1, .., N-1\}$. These available K signal samples define a system of K equations:

$$\left\{ a_{\tilde{k}} = \sum_{r=0}^{N-1} \gamma_r \varphi_r(\tilde{k}) \right\}_{\tilde{k} \in \tilde{\mathbf{K}}} \tag{2.45}$$

for K signal transform coefficients $\{\gamma_r\}$ of certain K indices r.

Select a subset $\tilde{\mathbf{R}}$ of K transform coefficients indices $\{\tilde{r} \in \tilde{\mathbf{R}}\}$ and define a "KofN"-bounded spectrum approximation $\hat{A}_N^{(BS)}$ to the signal $\mathbf{A}_N$ as

$$\hat{\mathbf{A}}_N^{(BS)} = \left\{ \hat{a}_k = \sum_{\tilde{r} \in \tilde{R}} \gamma_{\tilde{r}} \varphi_{\tilde{r}}(k) \right\}. \tag{2.46}$$

Rewrite this equation in a more general form that involves all transform coefficients:

$$\hat{\mathbf{A}}_N^{BS} = \left\{ \hat{a}_k = \sum_{r=0}^{N-1} \tilde{\gamma}_r \varphi_r(k) \right\}, \tag{2.47}$$

assuming that all transform coefficients with indices $r \notin \tilde{\mathbf{R}}$ are set to zero:

$$\tilde{\gamma}_r = \begin{cases} \gamma_r, & r \in \tilde{R} \\ 0, & \text{otherwise.} \end{cases} \tag{2.48}$$

Then the vector $\tilde{\mathbf{A}}_K$ of available signal samples $\{a_{\tilde{k}}\}$ can be expressed in terms of the basis functions $\{\varphi_r(k)\}$ of transform Φ_N as

$$\tilde{\mathbf{A}}_K = \mathrm{KofN}_\Phi \cdot \tilde{\Gamma}_K = \left\{ a_{\tilde{k}} = \sum_{\tilde{r} \in \tilde{R}} \gamma_{\tilde{r}} \varphi_{\tilde{r}}(\tilde{k}) \right\}, \tag{2.49}$$

where a $K \times K$ sub-transform matrix KofN_Φ consists of samples $\varphi_{\tilde{r}}(\tilde{k})$ of the basis functions with indices $\{\tilde{r} \in \tilde{\mathbf{R}}\}$ for signal sample indices $\tilde{k} \in \tilde{\mathrm{K}}$, and $\tilde{\Gamma}_K$ is a vector composed of the corresponding sub-set $\{\gamma_{\tilde{r}}\}$ of signal non-zero transform coefficients. This subset of the coefficients can be found by means of an inverting matrix KofN_Φ as

$$\tilde{\Gamma}_K = \{\gamma_{\tilde{r}}\} = \mathrm{KofN}_\Phi^{-1} \cdot \tilde{\mathbf{A}}_K \tag{2.50}$$

provided that the matrix KofN_Φ^{-1} inverse to the matrix KofN_Φ exists, which, in general, is conditioned for a specific transform by positions $\tilde{k} \in \tilde{\mathrm{K}}$ of available signal samples and by the selection of the subset $\{\tilde{R}\}$ of the transform basis functions that correspond to non-zero transform coefficients.

By virtue of the Parseval's relationship for orthonormal transforms, the bounded spectrum signal $\hat{\mathrm{A}}_N^{BS}$ approximates the complete signal A_N with the MSE:

$$MSE = \left\| A_N - \hat{A}_N \right\|^2 = \sum_{k=0}^{N-1} |a_k - \hat{a}_k|^2 = \sum_{r \notin R} |\gamma_{\tilde{r}}|^2. \tag{2.51}$$

This error can be minimized by an appropriate selection of K basis functions of the sub-transform KofN_Φ. In order to do so, one must know the energy compaction ordering of the basis functions of the transform Φ_N. If, in addition, one knows for a class of signals a transform that features the best energy compaction in the smallest number of transform coefficients, one can, by choosing this transform, secure the best minimal MSE bounded spectrum approximation of the signal $\{a_k\}$ for the given subset $\{\tilde{a}_k\}$ of its samples.

In this way, we arrive at the following discrete sampling theorem, which can be formulated in two statements:

1. For any discrete signal of N samples defined by its $K \leq N$ samples, its bounded spectrum approximation, in terms of a certain transform Φ_N, defined by Eq. (2.46), can be obtained with the MSE defined by

Eq. (2.51) provided positions of the samples secure the existence of the matrix $\mathrm{KofN}_{\Phi}^{-1}$ inverse to the sub-transform matrix KofN_{Φ} that corresponds to the spectrum bounding. The approximation error can be minimized by using a transform with the best capability of compacting the signal energy in a small number of signal transform coefficients (i.e., *energy compaction capability*).

2. Any signal of N samples that is known to have only $K \leq N$ non-zero transform coefficients for a certain transform Φ_N (a Φ_N-transform "bounded spectrum" signal) can be precisely recovered from exactly K samples provided that positions of the samples secure the existence of the matrix $\mathrm{KofN}_{\Phi}^{-1}$ inverse to the sub-transform matrix KofN_{Φ} that corresponds to the spectrum bounding.

In this formulation, the discrete sampling theorem is applicable to signals of any dimensionality. It requires no assumptions regarding the compactness of the signal spectral non-zero coefficients in the transform domain. The signal dimensionality affects only the formulation of the signal spectrum bounding.

2.7.2 Discrete sampling theorem formulations for specific transforms

The applicability of particular transforms for a bounded spectrum signal approximation from its given K samples depends on whether the KofN matrix for this transform is invertible, i.e., whether placement of the available signal samples is compatible with the type of signal spectrum bounding chosen for this transform. This subsection addresses the invertibility conditions most widely used in applications such as the discrete Fourier transform (DFT), discrete cosine transform (DCT), and Walsh and wavelet transforms.

Discrete Fourier transform and discrete cosine transforms

For discrete Fourier and discrete cosine transforms, the following statements hold:[15,16]

1. For any discrete signal of N samples, its bounded DFT (DCT) spectrum approximation with $K \leq N$ non-zero spectral coefficients defined by Eq. (2.46) can be obtained with a mean squared approximation error defined by Eq. (2.51) from exactly K of their samples taken in arbitrary positions.
2. DFT and DCT bounded spectrum signals of N samples with only K non-zero transform coefficients can be precisely recovered from exactly K of their samples taken in arbitrary positions.

Statements 1–4 constitute the *discrete sampling theorem* for discrete Fourier and discrete cosine transforms. Although the validity of these

statements was mathematically proved only for 1D signals and for 2D signals with separable spectrum bounding,[15] it is supported by considerable experimental evidence.

Presented here are the results of the verification experiments, which were conducted with computer-generated test signals and images with bounded DFT and DCT spectra using a direct matrix inversion and an iterative Gerschberg–Papoulis-type algorithm for signal reconstruction from samples. This algorithm, described in detail in Sections 4.1 and 4.2, computes at each iteration a signal DFT or corresponding DCT spectrum that zeros all transform coefficients but the selected ones, reconstructs the signal by inverse DFT (DCT) of the modified spectrum, and then restores the available signal samples at their known positions.

Plots in Fig. 2.12(a–b) illustrate the exact reconstruction of a DFT "bounded spectrum" signal (solid line) by matrix inversion for two cases: (a) all available signal samples are randomly placed within signal support, and (b) the available signal samples form a compact group.

Figures 2.13 and 2.14 illustrate precise reconstruction from sparse samples of computer-generated images band-limited in the DCT domain by a square (separable band-limitation) and by a 90° pie sector (inseparable band-limitation).

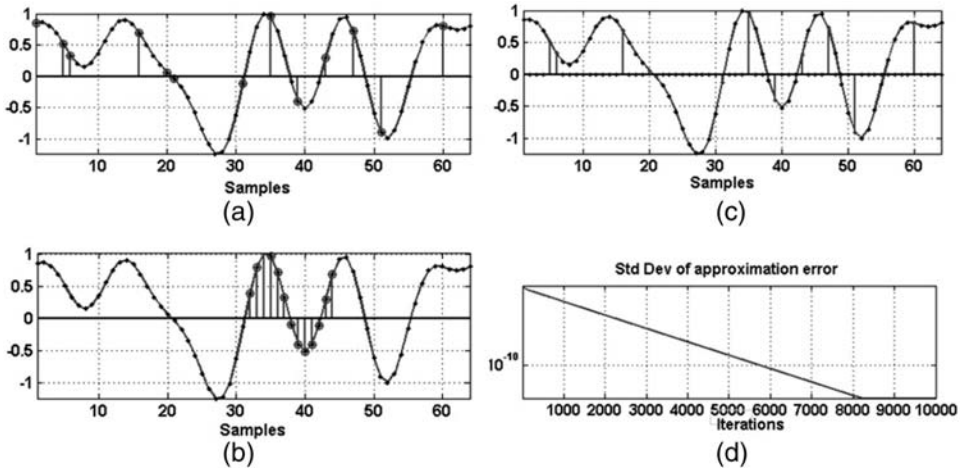


Figure 2.12 Reconstruction of a DFT bounded-spectrum signal by means of matrix inversion for the cases of (a) random and (b) compactly placed signal samples, as well as (c) the case using the iterative Gerschberg–Papoulis-type algorithm for the case of randomly placing signal samples. Plot (d) shows the standard deviation of signal reconstruction error as a function of the number of iterations in reconstruction by the iterative algorithm. The experiment was conducted for a test signal of 64 samples and a bandwidth of 13 frequency samples (~1/5 of the signal baseband). In all plots, the original signal is represented in the solid line obtained, for display purposes, by linear interpolation of its samples; available samples are represented by stems, and reconstructed samples are represented by dots.

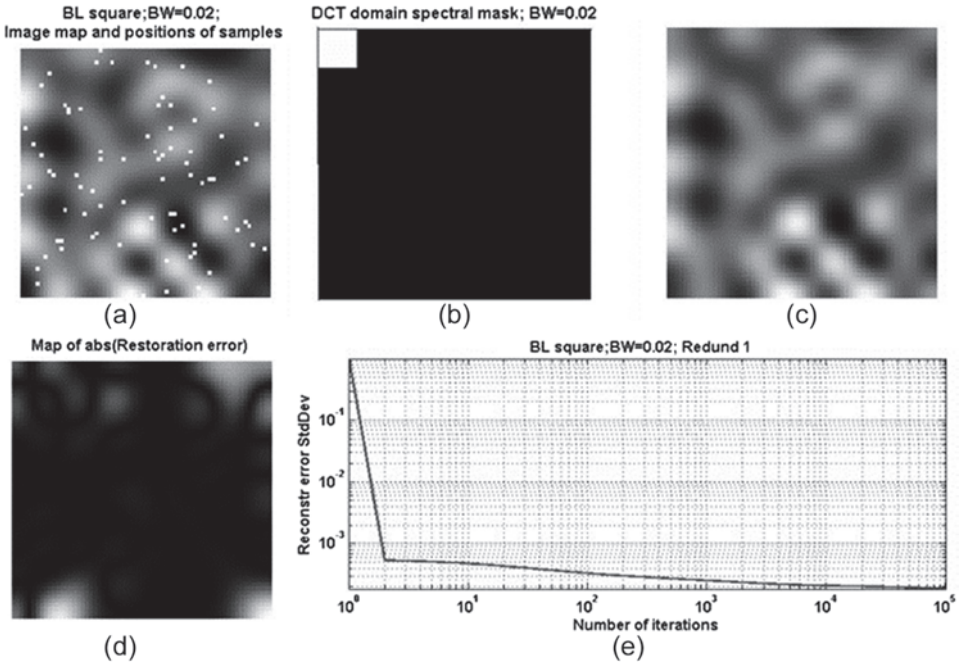


Figure 2.13 Reconstruction of an image with spectrum bounded by a square in the DCT domain: (a) initial image with 82 "randomly" placed samples in positions shown by white dots; (b) DCT spectrum bounding shape; (c) image reconstructed by the iterative algorithm after 100,000 iterations; (d) iterative reconstruction error (white = large errors, dark = small errors); and (e) RMS of iterative reconstruction error vs. the number of iterations.

The image in Fig. 2.13(a) is a 64- × 64-pixel test image low-pass band-limited in the DCT domain by a square of 9×9 samples (Fig. 2.13(b)). It has only 81 non-zero DCT spectral components out of the $64 \times 64 = 4096$ ones. This image was sampled at 82 "random" positions obtained from the standard MATLAB pseudo-random number generator. The figure shows that the iterative algorithm provides accurate reconstruction of the initial image after only a few iterations, though more precise reconstruction may require many more.

An important peculiarity of the iterative reconstruction process is that the convergence of iteration is not uniform within the image area. Usually, the image reconstruction error quickly reduces throughout the image, and only in some parts with a low sample density do reconstruction errors remain substantial and decay slowly.

Image band limitation by a square is separable over image coordinates and is not isotropic. Experiments show that in the case of the isotropic band limitation in the DCT domain by a pie sector, the speed of convergence of the iterative algorithm drops significantly, though again reconstruction error remains substantial only in limited areas of the image. The speed of

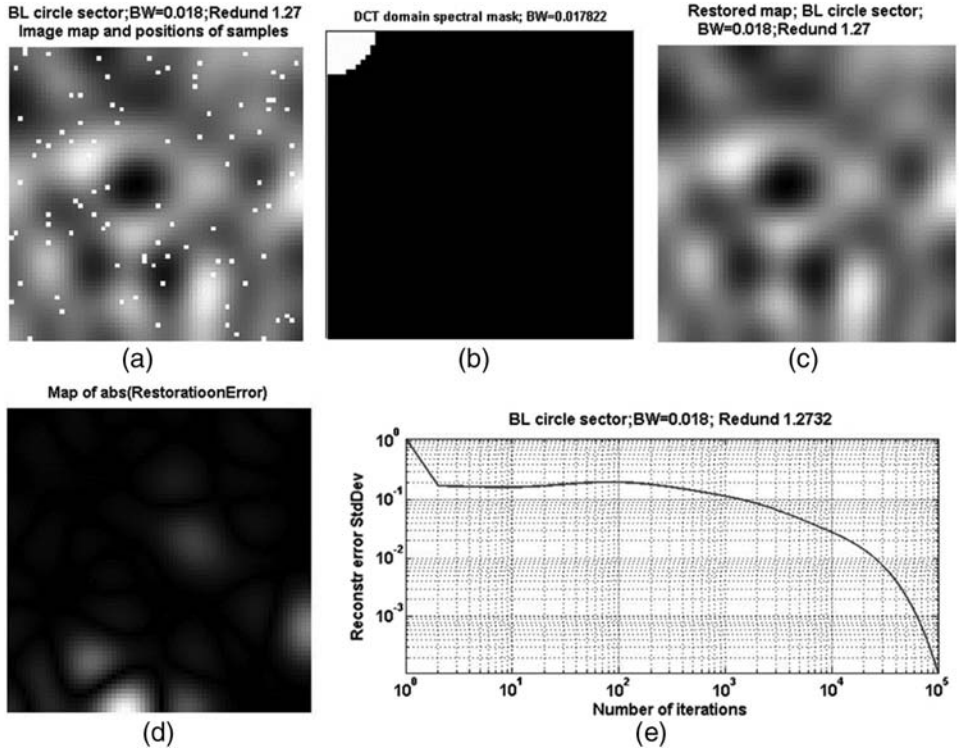


Figure 2.14 Reconstruction of an image with DCT spectrum bounded by a pie sector: (a) test image with 93 "randomly" placed samples in positions shown by white dots; (b) the spectrun bounding shape in the DCT domain, where the upper-left corner corresponds to the lowest spatial frequencies and the bottom-right corner corresponds to the highest spatial frequencies; (c) image reconstructed by the iterative algorithm after 100,000 iterations; (d) reconstruction error found as a difference between the initial test and reconstructed images displayed as an image with light corresponding to large errors and dark corresponding to small errors; and (e) RMS of reconstruction errors vs. the number of iterations.

convergence can be increased if the number of samples slightly exceeds the minimum defined by the sampling theorem (see Fig. 2.14).

The image in Fig. 2.14(a) is a 64- $\times$ 64-pixel test computer-generated image low-pass band-limited in the DCT domain by a pie sector. It has 73 non-zero DCT spectral components out of $64 \times 64 = 4096$, all located within a pie sector shown as white in Fig. 2.14(b). Compared to Fig. 2.13(a), this image was sampled at 93 "random" positions. The redundancy $93/73 = 1.27$ in the number of samples with respect to the number of non-zero spectral coefficients approximately equals the ratio of the area of a square to the area of the circle sector inscribed into this square. As shown in Fig. 2.14(e), with such a redundancy, iterative reconstruction converges quickly, and the overall reconstruction error after 100,000 iterations is comparable with

that for the separable band limitation by a square illustrated in Fig. 2.13. Once again, the convergence of the iterative algorithm is not uniform over the image, and relatively large reconstruction errors occur only in a small area of the image where the density of available samples happens to be low.

Wavelets and other bases

The main peculiarity of wavelet bases is that their basis functions are most naturally ordered in terms of two parameters: scale, and position within the scale. The scale index is analogous to the frequency index for the DFT and DCT. The position index only describes the shift of the same basis function within the signal extent on each scale. Therefore, band limitation for the DFT translates to scale limitation for wavelets. Limitation in terms of position is trivial: it simply means that some parts of the signal are not relevant.

Discrete wavelets are commonly designed for signals whose length is an integer power of 2 ($N = 2^n$). For such signals, there are $s \leq n$ scales and possible "band limitations."

The simplest special case of wavelet bases is the Haar basis. Signals with $N = 2^n$ samples and with only a K lower index non-zero Haar transform (only transform coefficients with indices $\{0, 1, \ldots, K-1\}$ are non-zero) are $\tilde{s} = (\lfloor \log_2(K-1) \rfloor + 1)$, i.e., "band limited," where $\lfloor x \rfloor$ is an integer part of x. Such signals are piecewise constant. The shortest interval of the signal constancy contains $2^{n-\tilde{s}}$ samples.

The first eight basis functions of the Haar transform are presented in the right column of Fig. 2.15. For any two samples located within the same interval, all Haar basis functions on this and lower scales have the same value. Therefore, more than one sample per constant interval will not change the rank of the matrix KofN_Φ. The condition for perfect reconstruction is, therefore, to have at least one sample on each of those intervals.

For other wavelets, as well as for other bases, a general necessary, sufficient, and easily verified condition for the invertibility of a KofN_Φ-trimmed transform sub-matrix has yet to be found. Standard linear algebra procedures to determine matrix rank can be used to test the invertibility of the matrix in each case.

Compared to other bases, the Walsh function basis is one of the most attractive thanks to its relation to the sinusoidal functions of the Fourier transform. For Walsh basis functions, the function index corresponds to the number of zero crossings of the basis function, called the *sequency*, which carries a certain analogy with the signal frequency. For many natural signals, the ordering of basis functions according to their sequency (which is characteristic for the Walsh transform) preserves the more or less regular

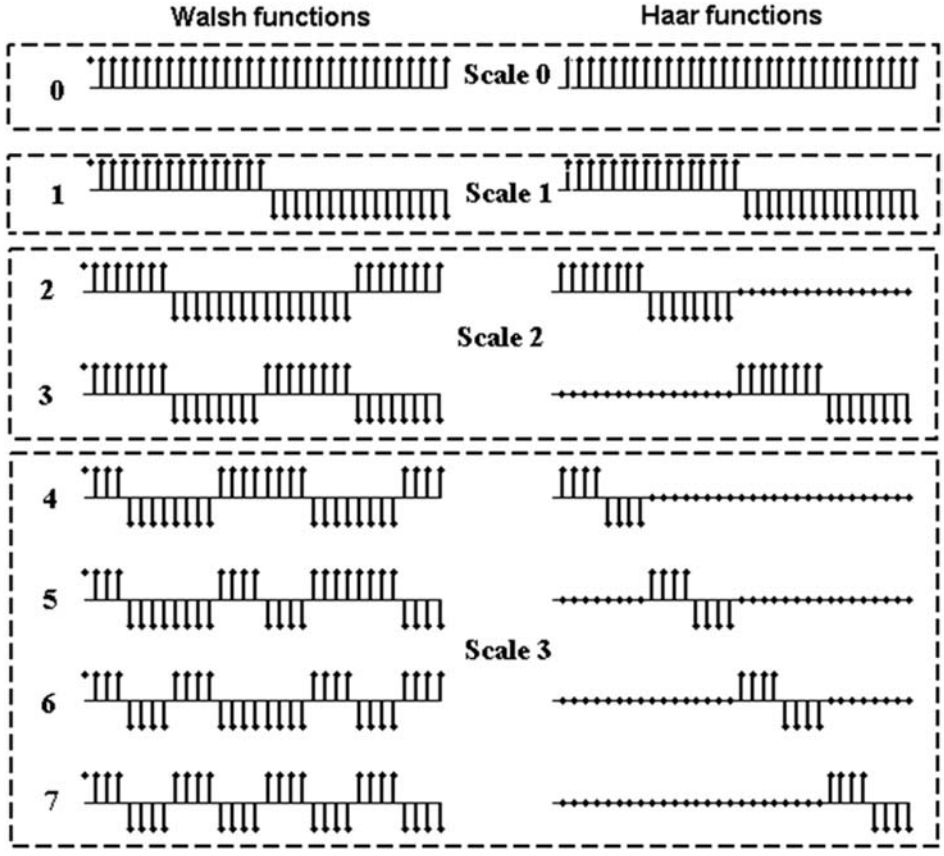


Figure 2.15 First eight Walsh and Haar basis functions grouped according to the scale parameter.

decay of the transform coefficients' energy with their index.[15] Therefore, for the Walsh transform, a low-pass band-limited signal approximation similar to that described for the DFT can be used.

Walsh basis functions (the left column in Fig. 2.15), similar to the Haar basis function, can be characterized by the scale index, which specifies the shortest interval of signal constancy. Signals with $N = 2^n$ samples and the band-limitation of K Walsh transform coefficients have the shortest intervals of signal constancy of $2^{n-\tilde{s}}$ samples, $\tilde{s} = (\lfloor \log_2(K-1) \rfloor + 1)$. Perfect reconstruction requires K signal samples taken on different intervals. Unlike the Haar transform case, not all of the intervals need to be sampled, only K intervals of the total. For a special case of K equal to a power of 2, there are K intervals, each of which has to be sampled to secure perfect reconstruction. This is the case when the reconstruction condition for the Walsh transform is identical to that for the Haar transform.

2.7.3 The general sampling theorem

In the limit when $N \to \infty$, the considered discrete model converts to a continuous model. In particular, if the discrete Fourier transform is chosen as the image transform, it converts to the integral Fourier transform, and the discrete sampling theorem for the DFT converts to the general sampling theorem, which refers to the signal Fourier spectra and states

1. The minimal number of signal samples per unit of signal area (sampling rate) sufficient for signal reconstruction from their arbitrarily placed samples with the MSE of reconstruction σ^2 equals the area B_Ω of the spectrum EC zone Ω in the signal Fourier domain that contains the $(E - \sigma^2)/E$-th fraction of the image signal energy E.
2. Signals known to have a Fourier spectrum bounded by a figure Ω can be precisely reconstructed from their arbitrarily placed samples taken with the density per unit of signal area equal to the area B_Ω of the spectrum bounding figure.

Chapter 4 discusses sampling with rates close to the theoretical miminum.

2.8 Exercises

The following MATLAB programs are provided as examples:

Aliasing_1D_SPIE.m
A demonstration, in the form of a movie, of sampling aliasing effects: strobe and moiré effects. Displayed are test sinusoidal signals of frequencies that linearly grow from frame to frame, corresponding Fourier spectra at a fixed sampling rate, corresponding signals reconstructed from their samples and their Fourier spectra, and the frequency response of the reconstruction filter.

Aliasing_2D_SPIE.m
A demonstration, in the form of a movie, of aliasing effects in sampling 2D signals. As test signals, 2D sinusoidal signals with different spatial frequencies that linearly grow from movie frame to frame are used. Displayed are test images, corresponding images reconstructed from their samples, and their Fourier spectra.

Fringe_aliasing_demo_SPIE.m
A demonstration, in the form of a movie, of the strobe effect in the sampling and reconstruction of 2D chirp signals with different growth speeds of spatial frequencies.

StrobEffect3D_SPIE.m
A demonstration, in the form of a movie, of the 3D strobe effect in image sampling and reconstruction with an example row-wise readout of a rotating image.

IdealVsNonidealSampling_SPIE.m
A demonstration of the visual effects of image sampling and reconstruction using pre-sampling and reconstruction filters with rect functions and sinc functions as filter point spread functions.

Five pre-prepared test images are provided; using any other image from image database as test images is also possible. Three downsampling rates can be chosen: 1/2, 1/3, and 1/4 of sampling rate of test images. Displayed are

- the chosen test image;
- the image reconstructed by nearest-neighbor interpolation (rect function as the reconstruction filter PSF) from the result of down-sampling the test image with the chosen sampling rate;
- the image reconstructed by nearest-neighbor interpolation (rect function as the reconstruction filter PSF) from the result of test image downsampling with the chosen sampling rate after pre-filtering by a filter with rect-function as the filter PSF of a width equal to the inverse of the sub-sampling rate;
- the image reconstructed by sinc interpolation (sinc function as the the reconstruction filter PSF) from the result of test image downsampling with the chosen sampling rate after its pre-filtering by the ideal low-pass filter of the corresponding bandwidth.

The option of displaying the Fourier spectra of tested and reconstructed images is also provided.

Chapter 3
Compressed Sensing Demystified

3.1 Redundancy of Regular Image Sampling and Image Spectra Sparsity

As mentioned previously, contemporary digital display devices and image processing software imply by default that sampling over regular square sampling lattices is used for image sampling. For such a sampling, the image sampling interval $(\Delta_x, \Delta_y = \Delta_{xy})$ is supposed to be chosen so that the square sampling baseband $[-1/2\Delta_{xy} \le f_x \le 1/2\Delta_{xy}, -1/2\Delta_{xy} \le f_y \le 1/2\Delta_{xy}]$ in the image Fourier domain (f_x, f_y) fully embraces the image spectrum EC zone for the given accuracy of image reconsruction. This means that the area $1/\Delta_{xy}^2$ of this square zone, by necessity, exceeds the area of image spectrum EC zone, which, by virtue of the general sampling theorem, defines the minimal sampling rate sufficient for image reconstruction with the required MSE. Threrefore, conventional regular image sampling is generally redundant. It would be instructive to numerically evaluate this sampling redundancy of sampled natural images.

Figure 3.1 presents a set of ten test images along with estimations of their Fourier-spectrum EC zones. The Fourier spectra of the images were estimated using the DFT as a discrete representation of the integral Fourier transform (see Appendix A2). In order to avoid spectrum estimation errors due to boundary effects as much as possible, images were multiplied before spectral analysis by a circular apodization mask to smoothly bring them down to zero at the edges of the sampled region. Highlighted in the figures of image spectra are spectral EC zones that contain the largest image spectral components sufficient for image reconstruction with the same MSE as that of the image JPEG compression implemented by MATLAB.

These figures show that the spectral EC zones of all images occupy only a fraction of the area of the sampling baseband. This fraction, i.e., the ratio of the area of the image spectrum EC zone to the area of the sampling baseband,

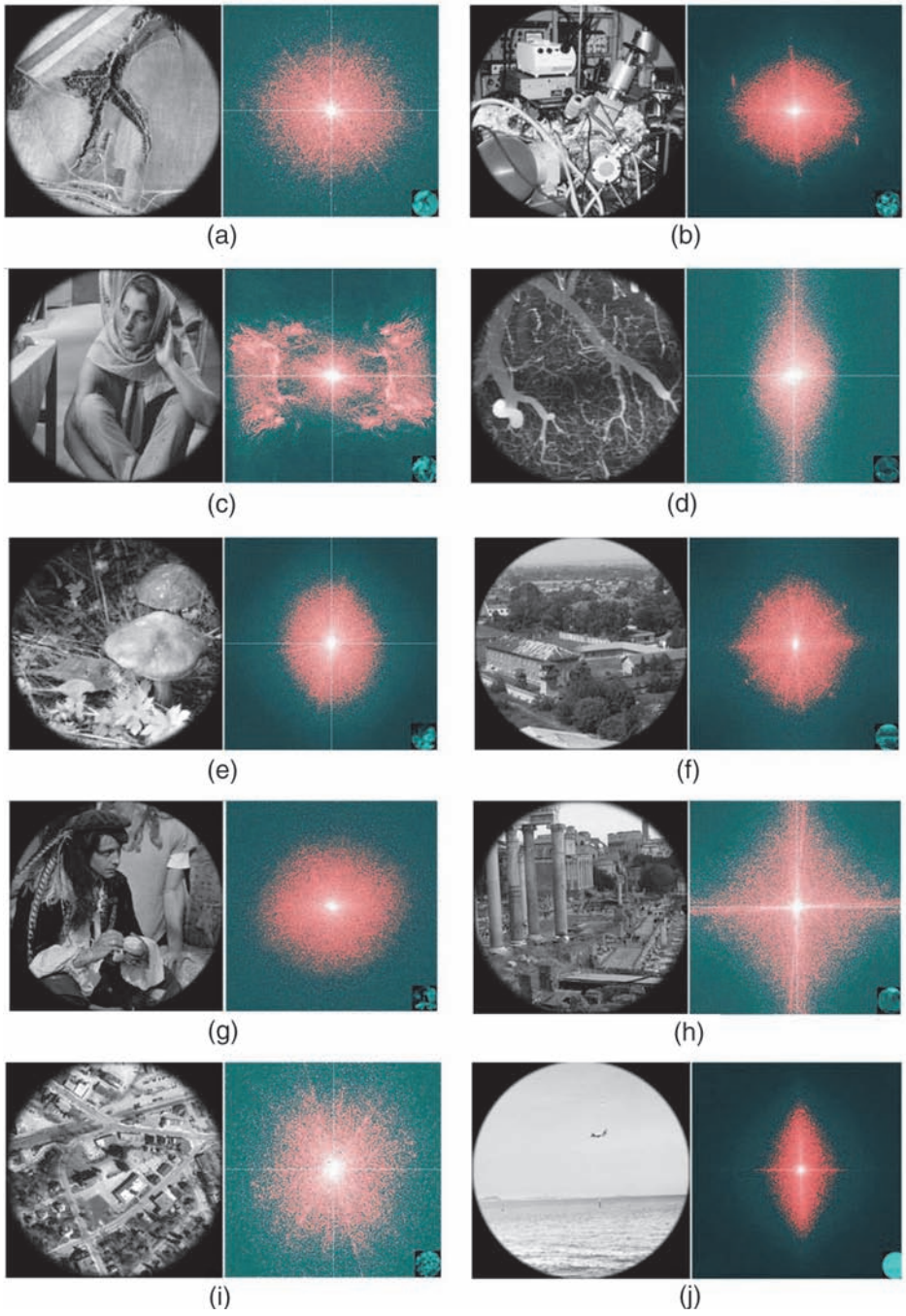


Figure 3.1 A set of sampled test images (first and third columns from the left) and their corresponding Fourier spectra centered at their DC component (second and fourth columns). From top to bottom, from left to right: "AerialPhoto512," "AFM512," "Barbara512," "BloodVessels512," "Mushrooms512," "Nish1024," "Pirate1024," "Rome512," "WestConcord364," and "Test4CS1024." Highlighted are the largest spectral components sufficient for image reconstruction with the same MSE as that of their JPEG compression.

Table 3.1 Estimates of image spectra sparsities for test images presented in Fig. 3.1.

Test Images	Image Spectrum Sparsity
AerialPhoto512	0.29
AFM512	0.19
Barbara512	0.31
BloodVessels512	0.20
Mushrooms512	0.19
Nish1024	0.21
Pirate1024	0.29
Rome512	0.39
WestConcord364	0.37
Test4CS1024	0.10

is called the image *spectrum sparsity* for the given MSE of signal sparse approximation. According to the discrete sampling theorem (Section 2.7), signal spectrum sparsity defines the minimal sampling rate sufficient for signal reconstruction with the MSE, for which the sparsity is found.

Estimates of image spectra sparsities for these images are given in Table 3.1. They show that these images are 2.5 to 10 times oversampled. These data illustrate the ubiquitous compressibility of images sampled in common ways and the degree of sampling redundancy that is usually removed by image compression after sampling.

3.2 Compressed Sensing: Why and How It Is Possible to Precisely Reconstruct Signals Sampled with Aliasing

The phenomenon of ubiquitous compressibility of images acquired by conventional methods raises a question: is it possible to directly measure the minimal amount of data that will not be discarded? This question was apparently first posed by the inventors of the *compressed sensing approach* as a solution to this problem.[10,11] Since its introduction, this approach (known also known as *compressed sampling*) to digital image acquisition has gained considerable popularity.

Conventional image sampling is based on image band-limited approximation. The compressed sensing approach is based on a similar but more general idea of signal sparse approximation: an approximation by signals whose spectrum in a certain chosen "sparsifying" transform is sparse, i.e., contains fewer non-zero transform coefficients than the total number of transform coefficients. The theory of compressed sensing states that if an image of N samples is known to have, in the domain of a certain transform, only K non-zero transform coefficients out of N, then the image can be precisely reconstructed from $M > K$ measurements by minimizing the number of image non-zero spectral coefficients, i.e., by minimizing the L0 norm in the image

transform domain or, the more practical option, by minimizing the sum of the modulus of the image spectral coefficients (minimizing the L1 norm (total variation) of the signal transform coefficients). This, in particular, means that the signal sampling rate, contrary to the conventional sampling, does not directly depend on the signal's highest frequency, which implies that the signal sampling rate can be lower than twice the signal's highest-frequency components, i.e., a kind of "sub-Nyquist" sampling or sampling with aliasing, is possible, which makes the compressed sensing approach especially appealing.

The following simple example will demonstrate the mechanism of image acquisition and reconstruction with compressed sensing. Let a signal of N samples composed of a known number $K < N$ of sinusoidal components be sub-sampled arbitrarily in $M < N$ points; it is required to precisely reconstruct the signal from these M samples, i.e., determine the amplitudes and frequencies of the signal sinusoidal components. Figure 3.2, obtained using the program **Random_Sampling_Sinusoids_SPIE.m** provided in the exercises, illustrates how and when this can be done. Plots in the figure are numbered from top to bottom.

In the figure, the test signal (the first plot) consists of three sinusoidal components ($K = 3,\ N = 256$) seen as three Kronecker deltas in the signal DCT spectrum (the second plot in Fig. 3.2). When this signal is downsampled as shown in the third plot in Fig. 3.2, aliasing spectral components appear in the spectrum of the sub-sampled signal, shown in the fourth plot in Fig. 3.2. In this example, the signal is downsampled in random positions with a 0.15 sampling rate ($M = 38$) of the baseband.

The intensity of the aliasing components grows as the downsampling rate decreases. However, until the downsampling rate drops significantly, the true spectral components of the signal will exceed the highest peaks of the aliasing spectral components, and these true components can be detected by finding the positions of the given number K (in this case, $K = 3$) of the largest spectral components (the fourth plot in Fig. 3.2). Once these positions are determined, the following steps of an iterative Gerchberg–Papoulis-type signal reconstruction algorithm can be run:

- Compute the DCT of the current estimate of the reconstructed signal.
- Detect the given number K of largest spectral components.
- Set to zero all spectral components except the detected ones.
- Compute the inverse DCT of the signal spectrum thus modified to get a new estimate of the reconstructed signal.
- Replace the samples of the estimated reconstructed signal in the positions of available signal samples with those samples, and then repeat the loop.

The plot of the reconstruction root mean squared error (RMSE) versus the number of reconstruction iterations (the fifth plot in Fig. 3.2), and the plot

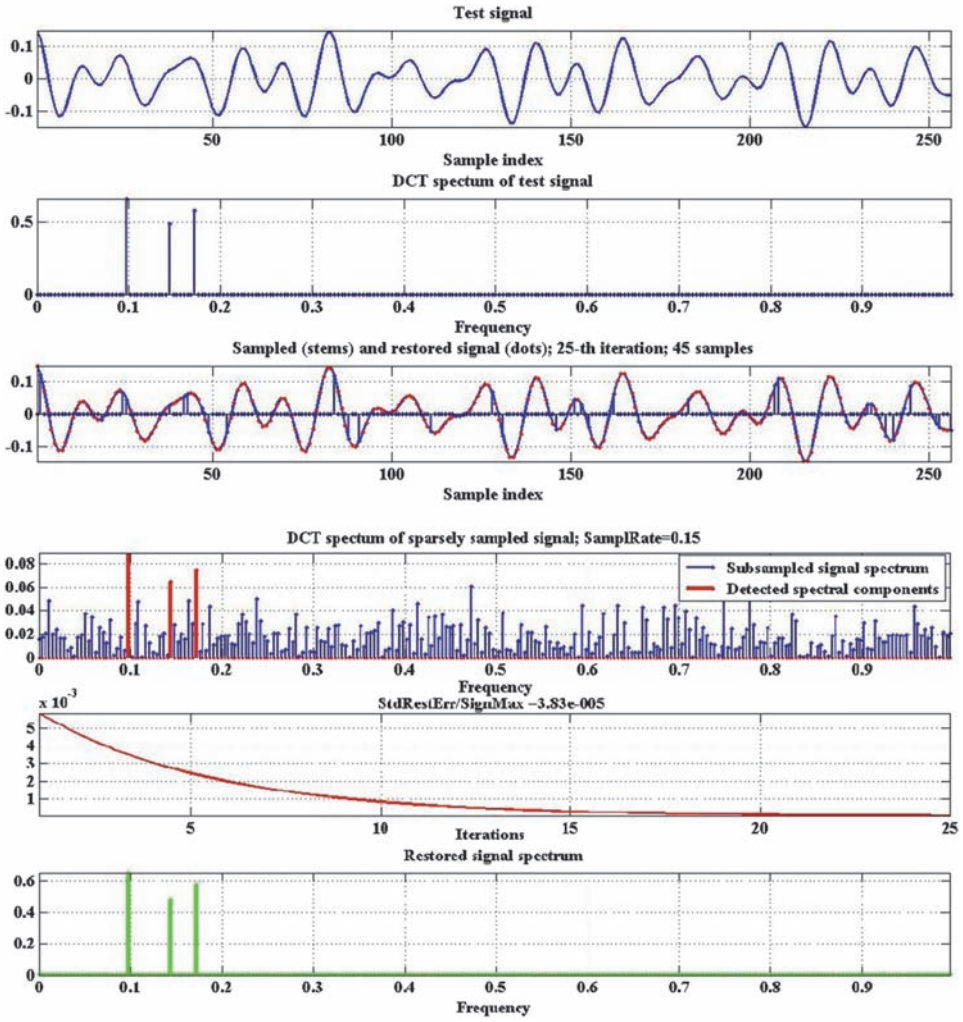


Figure 3.2 From top to bottom: test signal composed of three sinusoidal components; its DCT spectrum; this signal randomly sub-sampled (stems) and reconstructed (solid line); DCT spectrum of the sub-sampled signal; plot of the reconstruction RMSE vs. the number of reconstruction iterations; and the DCT spectrum of the reconstructed signal. The frequency is given in fractions of the width of the sampling baseband.

of the reconstructed signal DCT spectrum (the sixth plot) show that virtually precise reconstruction of the signal is achieved after a couple of tens of iterations: after 25 iterations, the reconstruction RMSE is 3.8×10^{-5}. Note that in this example the signal spectrum sparsity is $Ss = 3/256 \approx 1.2 \times 10^{-2}$, and therefore the sampling redundancy (ratio of the sampling rate to signal spectrum sparsity) is $R = M/K = 38/3 \cong 12.7$.

When the signal sub-sampling rate is too low and aliasing is severe, reliable detection of the signal spectral components in the spectrum of the

sampled signal, and thus signal reconstruction, becomes impossible. The presence of noise in the sampled data hampers reliable detection of the signal spectral components and requires additional sampling redundancy to produce reliable signal reconstruction.

When sampling is performed in random positions, one can evaluate the performance of the described method of signal reconstruction in terms of the probability of error in detecting signal spectral components. This probability depends on the downsampling rate and on the signal sparsity K/N. Figure 3.3(a) presents the results of the experimental evaluation of the probability of frequency identification error as a function of the downsampling rate for a sinusoidal signal ($K = 1$) of five frequencies (0.9, 0.7, 0.5, 0.3, and 0.1 of the signal baseband width), five signal lengths N (128, 256, 512, 1024, 2048) and, correspondingly, of five signal sparsities K/N. The results for each value of signal sparsity are averages over the results obtained for different frequencies (Fig. 3.3(b)). Figure 3.3 shows the sampling redundancies required for signal reconstruction by the above described algorithm with a probability of signal frequency identification error less than 10^{-4}, 10^{-3}, and 10^{-2}. The results were obtained by a Monte Carlo simulation of the algorithm with 5×10^4 realizations of random sampling for each individual experiment with a given sampling rate, signal frequency, and signal length.

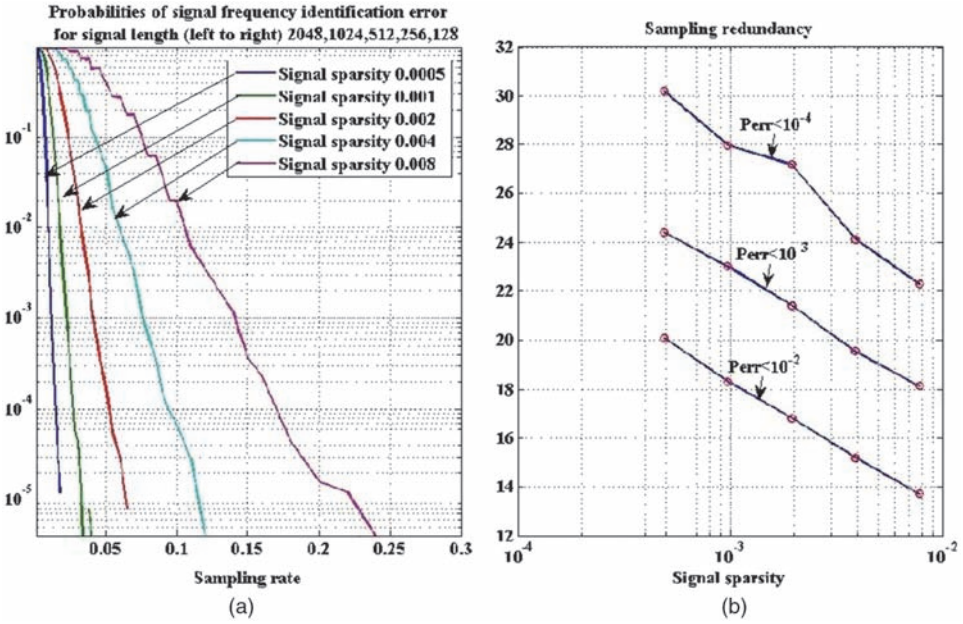


Figure 3.3 Plots of (a) the probabilities of error in identification of signal frequency and (b) estimates of sampling redundancy as functions of signal sparsity K/N for $K = 1$ and $N = 128,\ 256,\ 512,\ 1024,\ 2048$.

These results illustrate the principles of the compressed sensing approach to reconstructing signals sampled with aliasing. They demonstrate also that considerable sampling redundancy is required to secure reliable reconstruction of signals sampled with aliasing, using *a priori* assumption of their sparsity.

3.3 Compressed Sensing and the Problem of Minimizing the Signal Sampling Rate

The potential benefits of compressed sensing to image sampling and reconstruction are widely advertised in the literature, but much less is known about its limitations, especially how many measurements are required for signal and image acquisition compared to the theoretical minimum defined by the sampling theory.

According to the theory of compressed sensing, the precise reconstruction of a signal of N samples that has $K < N$ non-zero transform coefficients requires a number of measurements M that satisfies the following inequality:[17]

$$M/K > -2\log[(M/K)(K/N)]. \tag{3.1}$$

According to the discrete sampling theorem, the minimum number of signal samples for signals with K non-zero spectral components equals K. Therefore, the signal sparsity $Ss = K/N$ is the theoretical minimum of the sampling rate required for signal reconstruction, and the ratio $R = M/K$ of the number of required measurements M to the number K of signal non-zero transform coefficients represents the sampling redundancy with respect to the theoretical minimum. Equation (3.1) can be rewritten as a relationship between the signal sparsity $Ss = K/N$ and sampling redundancy $R = M/K$ as

$$R > -2\log(R \times Ss). \tag{3.2}$$

Numerical evaluation of Eq. (3.2) between the sampling redundancy and signal sparsity is presented in Fig. 3.4 by the dash-dotted line. The solid line in this figure is plotted on the base of experimental data collected in the literature.[18]

Figure 3.4 shows that in the range of sparsities from 0.10 to 0.39 of the test images in Fig. 3.1, the sampling redundancy of compressed sensing should theoretically be larger than 2–3, and according to the experimental curve, it should be larger than 2.5–5.0. Thus, the compressed-sensing sampling redundancy for this set of test images is not much lower than the sampling redundancy of the regular sampling (2.5–10.0), which meams that the former is quite far from reaching the theoretical minimum of signal sampling rates.

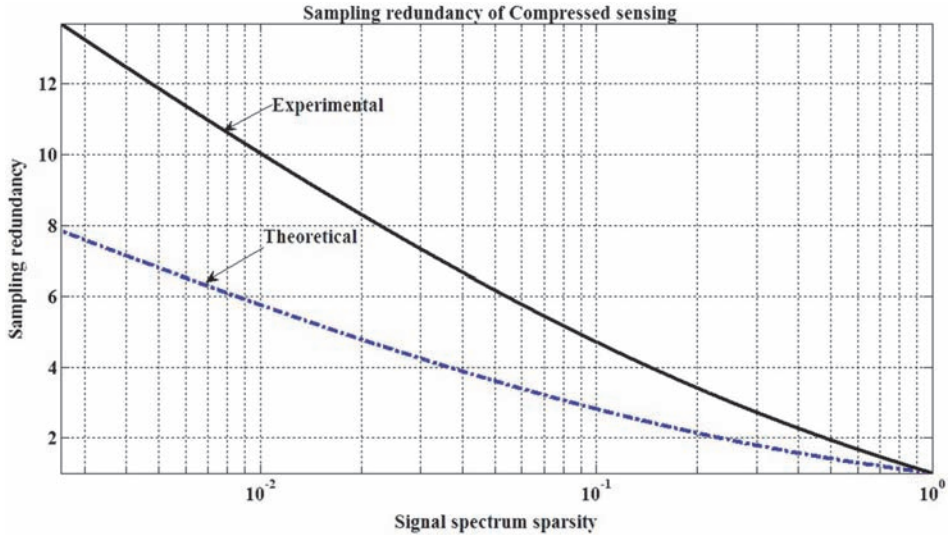


Figure 3.4 Theoretical and experimental relationships between signal sparsity and sampling redundancy required by compressed sensing. Signal reconstruction is only possible for sampling redundancies above the curves.

The sampling redundancy required by compressed sensing is not its only drawback. The approach's applicability is also impeded by its vulnerability to noise in sensed data and by its inability to predict and secure the resolving power of reconstructed images. The resolving power of images is determined by the size and shape of the EC zones of their spectra. The spectral EC zones of images reconstructed by compressed sensing are formed during reconstruction, not specified in advance from the requirements for image resolving power.

Compressed sensing methods are, to a certain degree, capable of reconstructing sparse approximations of signals and images sampled with aliasing. No knowledge regarding the EC zones of image spectra is required; one needs only choose an image sparsifying transform and define the desired spectrum sparsity. However, ignorance of the EC zones has a price: compressed sensing requires significant redundancy in the number of measurements sufficient for signal and image reconstruction compared to the theoretical minimum.

In many practical tasks of digital image acquisition, the assumption of complete uncertainty regarding image-spectra EC zones has no justification. In fact, if one is ready to accept a sparse approximation to an image and has chosen an image sparsifying transform, as supposed by the compressed sensing approach, one tacitly implies a certain knowledge of the energy compaction capability of the chosen transform. Chapter 4 will show how the use of this *a priori* knowledge allows image sampling with rates close to the theoretical minimum.

3.4 Exercise

RandomSamplingSinusoids_SPIE.m

Demonstration on a discrete simulating model of the effects of sparse random sampling of signals composed of a given number of sinusoidal components; reconstruction by detection in the spectra of the sampled signals of the given number of the largest sinusoidal components; and subsequent determination of their intensities using an iterative Gerschberg-Papoulis type algorithm.

The frequencies of the sinusoidal components of the test signals are chosen randomly by a random number generator. The requested parameters are

- The number of samples of the signal discrete model,
- The number of signal sinusoidal spectral components, and
- The signal sampling rate (in fraction of the width of the signal baseband).

Chapter 4
Image Sampling and Reconstruction with Sampling Rates Close to the Theoretical Minimum

4.1 The ASBSR Method of Image Sampling and Reconstruction

This chapter describes a method of image sampling and reconstruction with sampling rates close to the minimal rate defined by the sampling theory. We begin with a formulation of a discrete model, for which the discrete sampling theorem holds.

The theorem implies the following image sampling and reconstruction protocol:

- Choose the number N of image samples required for image display and processing.
- Choose an image sparsifying transform.
- Specify a desired spectrum energy compaction zone of the image spectrum, i.e., a set of $M \leq N$ transform coefficients to be used for image reconstruction.
- Measure M image samples.
- Use M image samples to determine M transform coefficients of the chosen EC zone.
- Set the $N - M$ transform coefficients to zero and use the obtained spectrum to reconstruct the required N image samples by applying the inverse transform to the formed signal spectrum.

Consider possible ways to implement this principle:

Choosing a transform. The choice of transform is governed by the transform's energy compaction capability, i.e., the ability to compact most of the image signal energy into a small number of transform coefficients. An additional requirement is the availability of a fast transform algorithm. From

this point of view, the DCT, DFT, and wavelet transforms are the primary candidates.[16]

Specification of image-spectrum EC zones. The specification of the subset of image transform coefficients to be used for image reconstruction, i.e., those of the image-spectrum EC zone, can be made based on the known energy compaction capability of the chosen transform. The following assumes the DCT is used as the image sparsifying transform. The DCT efficiently compacts the image transform's largest coefficients into more or less compact groups in the area of the coefficient's lower indices around the DC component.

Considerable practical experience, including that obtained in the course of developing zonal quantization tables for the JPEG image-compression standard, shows that although these groups of DCT coefficients do not have sharp borders, they are well concentrated and can be, with a reasonably good accuracy in terms of preservation of the group total energy, circumscribed by some standard shape that encompasses the area of the image's low spatial frequencies and is specified by few numerical parameters, such as area, aspect ratio, angular orientation, etc.

Figure 4.1 presents a set of five standard shapes suited for the DCT as the sparsifying transform: rectangle, triangle, pie-sector, ellipse, and superellipse. In principle, each standard shape can be associated with a certain class of images, such as micrographs, aerial photographs, space photos, indoor and outdoor scenes, etc.

Experimental experience shows that no fine tuning of the shape parameters is required to specify the chosen shapes as approximations of image-spectrum EC zones. This property of sparse DCT spectra is illustrated in Fig. 4.2 with an example of a sparse DCT spectrum of the test image "BloodVessels512." The image features the prevalence of horizontal edges, which causes anisotropy of the image sparse spectrum, as seen in Fig. 4.2. Images in Fig. 4.2(b–e) indicate the positions of the largest DCT coefficients that reconstruct this image with a root mean square error (RMSE) of 3.85 gray levels of 255 levels—the same as the reconstruction RMSE of this image after standard JPEG compression with MATLAB. These coefficients occupy 0.164 of the image baseband area and form the image-spectrum EC zone.

Figure 4.1 Standard shapes of image-spectrum EC zones for image DCT spectra. From left to right: rectangle, triangle, pie-sector, oval, and superellipse. Spectrum DC components are in the upper-left corner of the shapes.

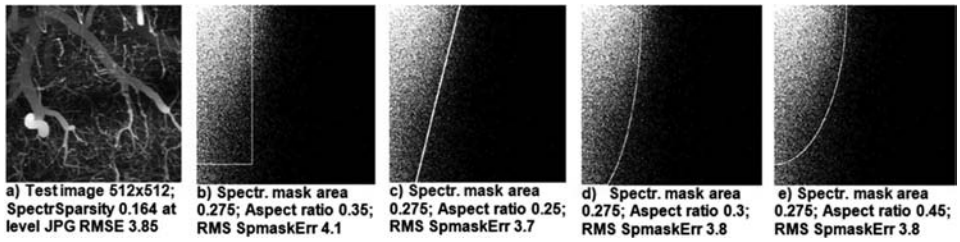


Figure 4.2 (a) Test image "BloodVessels512" with a 0.164 spectrum sparsity at the level of 3.85-RMSE JPEG encoding of image gray levels, and (b)–(e) the positions of the image's largest spectral components that reconstruct the image with a 3.85 RMSE (white dots) and borders (white lines) of rectangular, triangular, and two oval shapes, which approximate the image-spectrum EC zone.

Additionally, boxes (b)–(e) show the borders of rectangular, triangular, and oval shapes that have the same area (0.275 of the baseband area) and different aspect ratios (0.35, 0.25, 0.3, and 0.45, respectively). When used as spectrum-bounding shapes that approximate the image-spectrum EC zone, they all reconstruct the test image with practically the same reconstruction RMSEs (4.1, 3.7, 3.8, and 3.8 of the image gray levels, respectively) as that of the JPEG compression (3.85).

Images with spectra bounded by one of these shapes are visually indistinguishable from each other even though the patterns of the reconstruction errors look, naturally, a bit different. Two examples of such images obtained for spectrum bounding by a rectangle (as in Fig. 4.2(b)) and by an oval (as in Fig. 4.2(d)) are shown in Figs. 4.3(a) and (b) along with the corresponding patterns of reconstruction errors (Figs. 4.3(c) and (d)). For display purposes, reconstruction errors are eight times contrasted.

In Fig. 4.2, the shapes chosen to approximate the image-spectrum EC zone do not include all of the spectral coefficients; in fact, they include some spectral components that do not belong to the given zone. These components have, by definition, lower intensity than the components of spectrum EC zone that happen to be outside the shape. Therefore, given the energy of all spectral components encompassed by the chosen spectral shape, the number of these internal "no-EC-zone components" must exceed the number of spectrum EC zone components not encompassed by the shape. Thus, the area of the shape that defines the number of samples to be taken will always exceed the number of spectrum EC zone coefficients, which, theoretically, is the minimal number of samples required. For instance, the redundancies of approximating the spectrum EC zone shapes in Fig. 4.2 are $0.275/0.164 = 1.67$. The experimental results reported below show that the redundancies of standard spectrum EC zones for natural images are, as a rule, of the same order of magnitude. This redundancy is the tradeoff for not knowing the exact indices of the transform coefficients that form the image-spectrum EC zone.

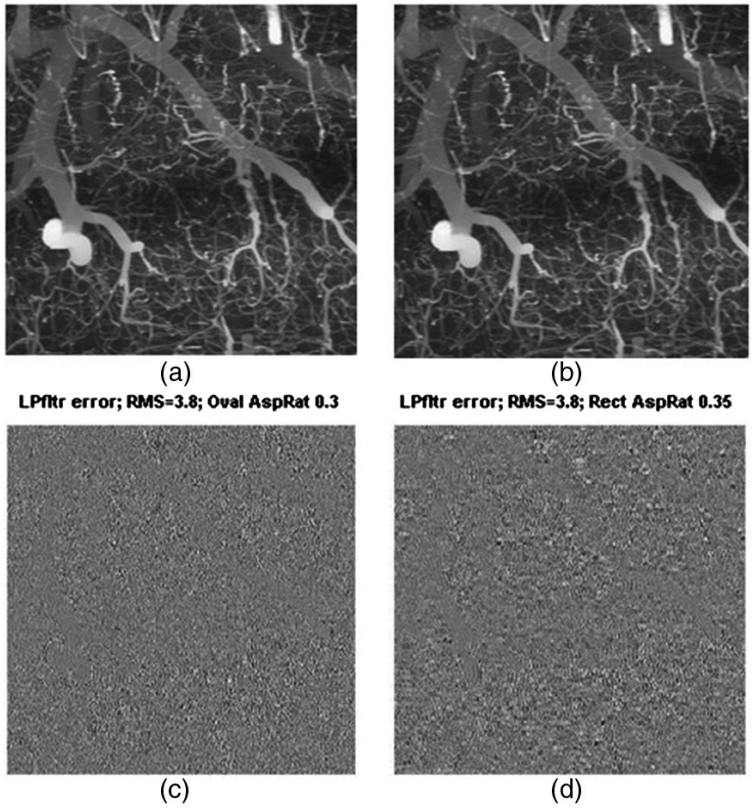


Figure 4.3 (a), (b) Images reconstructed after bounding the DCT spectrum of the test image of Fig. 4.2(a) by an oval (Fig. 4.2(d)) and a rectangular (Fig. 4.2) spectral mask; and (c), (d) patterns of the corresponding reconstruction errors, displayed eight times contrasted).

Specifying the positioning of image samples. Positioning of image samples should permit computation from them samples of the group of transform coefficients chosen for image reconstruction. As mentioned in Section 2.7.2, some image transforms, such as wavelets, impose in this respect a certain limitation on the positions of image samples. For the DFT and DCT, the positions of image samples can be arbitrary. An additional advantage of using the DFT and DCT as image sparsifying transforms is that, as shown in Sections 10.1 and 10.3, they are discrete representations of the integral Fourier transform and as such ideally match the characterization of imaging systems in terms of their frequency responses.

Numerical algorithms for image reconstruction. The following two options for image reconstruction are available:

- Direct inversion of the $M \times N$ transform sub-matrix that links M available samples and M transform coefficients specified by the chosen spectrum EC zone with the other $N - M$ coefficients set to zero. The

inverse transform is then applied to the found spectrum to reconstruct all required N samples. Generally, matrix inversion is a difficult computational task, and no fast matrix inversion algorithms are known. In the present case, a pruned fast transform matrix should be inverted. Pruned versions of fast transforms exist to compute subsets of transform coefficients of signals with all samples except several of those that equal zero,[20] which is inverse what is required in the given case. The question whether these pruned fast algorithms can be inverted to compute a subset of transform non-zero coefficients from a subset of signal samples is open.

- An iterative Papoulis–Gerchberg-type algorithm, where each iteration consists of two steps:
  - The iterated reconstructed image is subjected to the direct transform, and then spectral coefficients outside the chosen spectrum bounding EC zone are zeroed to obtain an iterated image spectrum.
  - The iterated image spectrum is inversely transformed, and then samples of the obtained image at their originally obtained positions are replaced by the corresponding available samples, which produces the next iterated reconstructed image. As a zeroth-order approximation from which the reconstruction iterations start, an image interpolated from the available samples can be taken. The next section describes an interpolation algorithm used in the verification experiments.

When the number of image samples N in the discrete model tends to infinity, the discrete sampling theorem converts to the general sampling theorem for continuous signals, and the above sampling and reconstruction protocol for the discrete model translates to the following protocol of image sampling and numerical reconstruction assuming the DCT as the image sparsifying transform.

Image sampling

- Choose a required image spatial resolution SpR (in dots per inch), as with conventional image sampling.
- Evaluate the image, and on this basis choose one of the standard spectral bounding shapes to bound the EC zone of the DCT spectrum and its shape parameters, such as the aspect ratio for a rectangle and triangle, the aspect ratio and orientation angle for an ellipse, etc.
- Evaluate the X and Y dimensions $ShSzX$ and $ShSzY$ of the chosen shape using SpR as the shape's largest diameter.
- Specify the required number of pixels N_x and N_y per inch in the X and Y dimensions of the reconstructed image $N_x \geq ShSzX$ and $N_y \geq ShSzY$.
- For the chosen shape, evaluate the fraction Fr of the area that the shape occupies in the rectangle $N_x \times N_y$. This fraction times $SpR \times SpR$

determines the spatial density $SpD = Fr \times SpR^2$ of samples to be taken (in dots per square inch). The number of samples M to be taken can be found as a product of SpD and the image area $ImgSzX \times ImgSzY$: $M = SpD \times ImgSzX \times ImgSzY$.

- Choose whatever sampling lattice appropriate for the available image sensor and sample the image in M positions using the sampling device with the point spread function, which approximates the point spread function that corresponds to the chosen image-spectrum EC zone.

Image reconstruction. Apply to the sampled image one of the aforementioned reconstruction options using as the image-spectrum EC zone the chosen spectrum bounding shape. In this way, an image with a spectrum in the chosen transform bounded by the chosen EC zone, or a bounded spectrum (BS) image, will be obtained, which has the prescribed spatial resolution SpR.

The described sampling protocol does not essentially differ from the standard 2D sampling protocol. The only difference is that in the suggested method arbitrary sampling lattices can be used, and evaluation of the image's expected spectrum shape to approximate the image-spectrum EC zone is required in addition to specification of the desired image resolution, which would be required by the standard sampling protocol anyway.

Image reconstruction from sampled data it not much different from the standard one either. In the suggested method, low-pass filtering during reconstruction is carried out numerically by bounding the image spectrum in the chosen transform by the chosen spectral shape. The method thus reaches the minimal sampling rate defined by the area of the chosen spectral shape. As mentioned previously, the latter is somewhat larger than the area occupied by the image sparse spectrum, which the chosen spectrum EC zone shape approximates and which defines the absolute minimum of the image sampling rate.

The described image sampling and reconstruction method can be called the arbitrary sampling and bounded spectrum reconstruction (ASBSR) method.

4.2 Experimental Verification of the Method

The ASBRS method of image sampling and reconstruction described in the previous section was experimentally verified on a considerable database of test images, including the ten images presented in Fig. 3.1. The experiments used the iterative Gerchberg–Papoulis-type algorithm of image BS reconstruction and tested three types of sampling lattices:

- "Quasi-uniform" sampling lattice, in which M image samples are distributed uniformly with appropriate rounding of their positions to the nearest nodes of the dense square sampling lattice of N samples;

- Uniform sampling lattice with random jitter, in which the horizontal and vertical positions of each of M samples are randomly chosen, independently in each of two image coordinates, within the primary uniform sampling intervals; and
- Totally pseudo-random sampling lattice, in which the positions of samples are uniformly distributed in a pseudo-random order over nodes of the dense sampling lattice of N samples.

The DCT was used as an image transform to compact the image spectrum. As an admissible RMSE of the approximation of test images with images of sparse DCT spectra, the RMSE of image compression by the standard JPEG compression in MATLAB is taken. These RMSEs were used to choose EC zone appriximating shapes for each image. The chosen shapes were used to bound the image DCT spectra both for image pre-filtering before sampling and during image reconstruction. The former is important not only to avoid aliasing errors but also to secure convergence of the iterative reconstruction algorithm to an image with a spectrum bounded by the chosen EC shape.

As a zeroth-order approximation of reconstructed images, from which the iterative reconstruction starts, each unavailable image sample was interpolated from the three nearest available samples taken with weights inversely proportional to their distances from the interpolated sample.

Figures 4.4–4.6 illustrate the results of experiments with ten images of the tested set. Shown in Fig. 4.4 are

1. a test image;
2. a reconstructed image;
3. a sampled test image;
4. the border of the chosen shape of the image EC zone (solid line) and the positions of the image DCT spectrum largest coefficients (white dots), which reconstruct the image with a RMSE that equals that of the image JPEG compression; and
5. plots of the RMSE of all reconstruction errors (the difference between the initial test image pre-filtered to bound its spectrum before sampling and the reconstructed image) and of the RMSE of the 90% smallest reconstruction errors versus the iteration number.

The RMS of the 90% smallest reconstruction errors is counted separately because the iterative reconstruction converges non-uniformly over the image area: most of the errors decay with iterations more rapidly than the few isolated large errors. Reconstruction RMSEs are given in units of image gray levels from 0–255.

Figures 4.5 and 4.6 show only (i) reconstructed images (left column), (ii) positions of the largest DCT coefficients of sparse approximation of the corresponding test images and borders of their chosen-spectrum EC zones

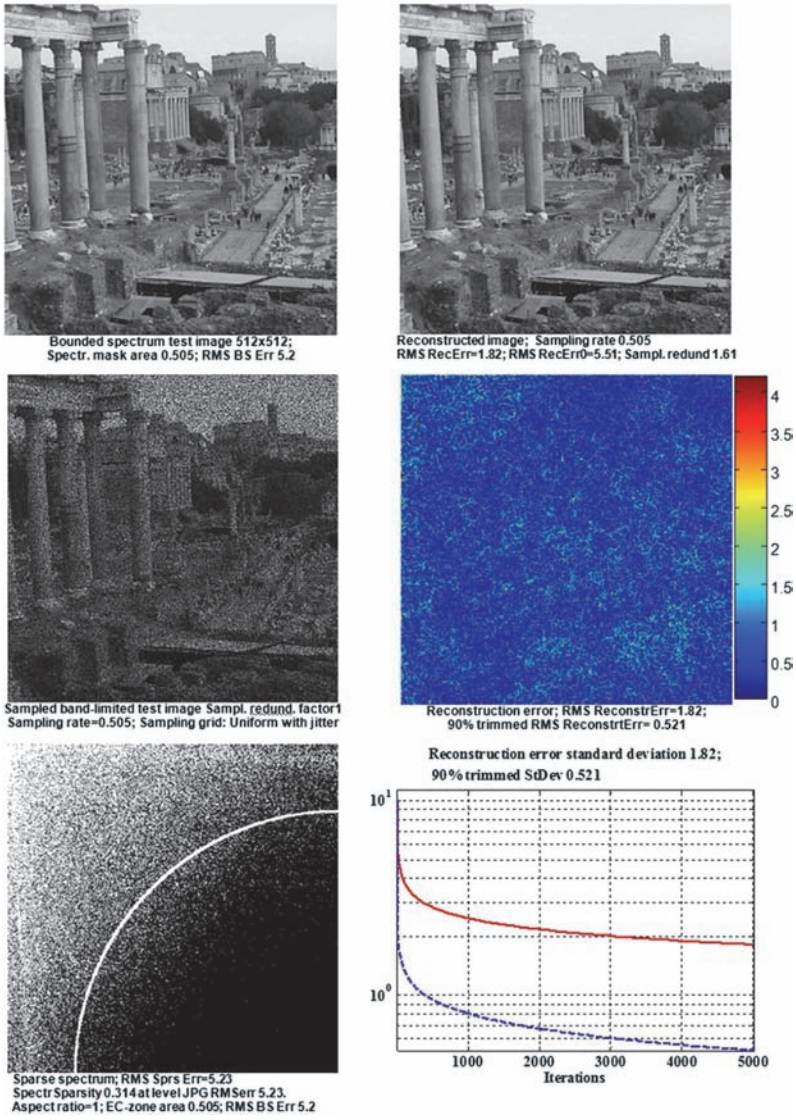


Figure 4.4 Results of experiments on the sampling and BS reconstruction of the test image "Rome512." From top to bottom, left to right: test image; reconstructed BS image; sampled test image; EC zone of the test image spectrum (white dots) and the border of the chosen-spectrum EC zone shape (white solid line); color-coded (MATLAB color map "jet") absolute value of reconstruction error (difference between initial and reconstructed images); plots of the RMSE of all (solid line) and of the smallest 90% (dashed line) reconstruction errors vs. the number of iterations.

(middle column), and (iii) plots of the reconstruction RMSE versus the number of iterations (right column).

For all shown images, sampling over uniform sampling lattices with random jitter was used, for which reconstruction errors decayed with

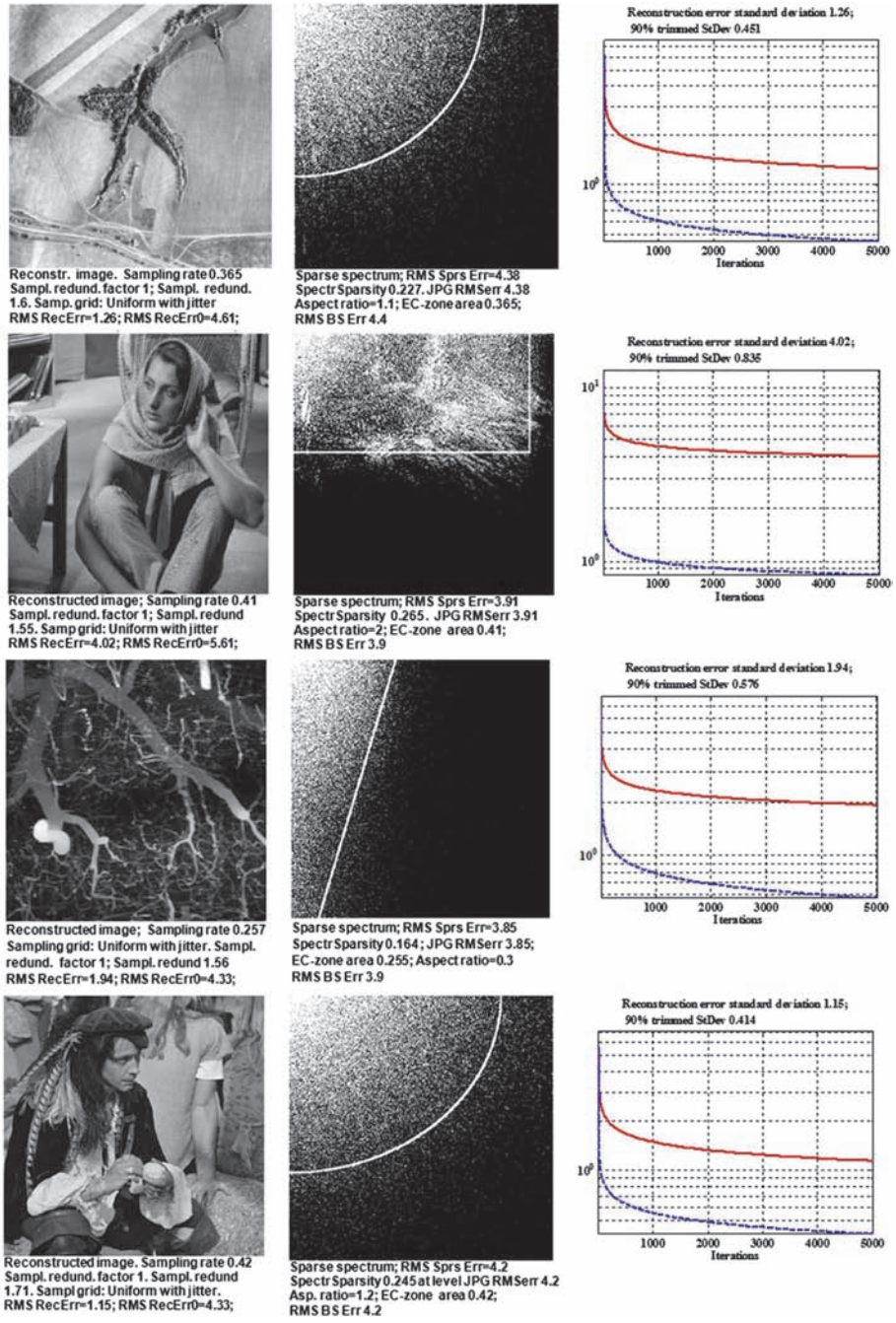


Figure 4.5 Results of experiments on the sampling and reconstruction of test images. From top to bottom: "AerialPhoto512," "Barbara512," "BloodVessels512," and "Pirate1024." From left to right: reconstructed images; EC zones of image spectra (white dots) and borders of the corresponding chosen-spectrum EC zone approximating shapes (white solid line); plots of the RMSE of all (solid line) and of the smallest 90% (dashed line) reconstruction errors vs. the number of iterations.

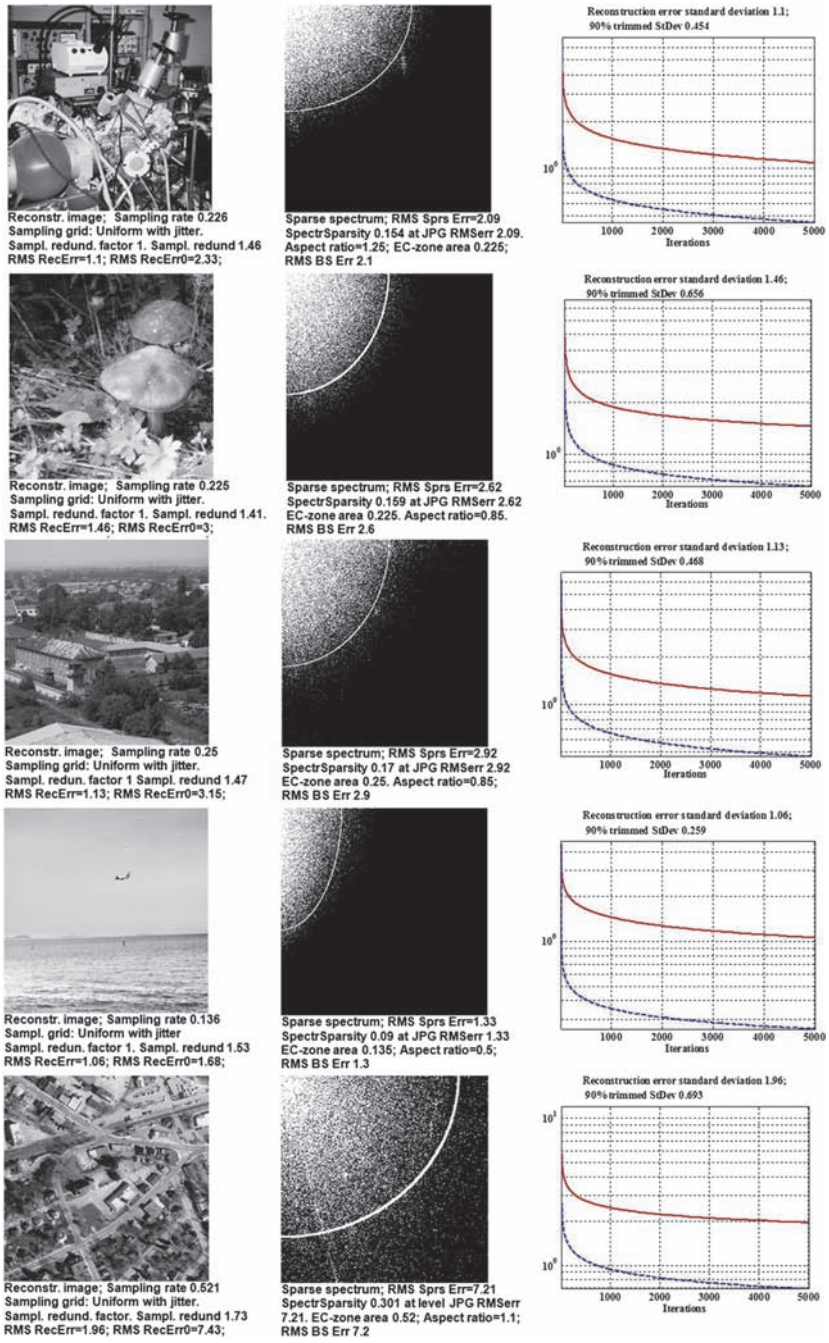


Figure 4.6 Results of experiments on sampling and reconstruction of test images. From top to bottom: "AFM1024," "Mushrooms512," "Nish1024," "Test4CS1024," and "WestConcord364." From left to right: reconstructed images; EC zones of image spectra (white dots) and borders of the corresponding chosen-spectrum EC zone approximating shapes (white solid line); plots of the RMS of all (solid line) and of the smallest 90% (dashed line) reconstruction errors vs. the number of iterations.

iterations most rapidly. For the same number of iterations, the reconstruction RMSE for a totally random sampling lattice is about 1.5–2.0 times, and for a "quasi-uniform" sampling lattice 2.0–2.5 times, larger than those for the "uniform with jitter" sampling lattice. For "quasi-uniform" sampling lattices, stagnation of the iteration process occurred, which can be attributed to the presence of regular patterns of thickening and rarefication of sampling positions due to the rounding of their coordinates to positions of nodes on the regular uniform sampling lattice.

Table 4.1 summarizes the experimental results obtained for all ten test images shown in Figs. 4.4–4.6 for reconstruction accuracy, spectrum sparsity, sampling rate, redundancy of the chosen-spectrum EC zone approximating shapes (the ratio of fractions of area they occupy in the sampling baseband to the spectrum sparsity), sampling redundancy (the ratio of the sampling rate to the relative area of the chosen-spectrum EC zone approximating shapes, which is the theoretical minimum of sampling rate for the given shape), and the overall sampling redundancy (the ratio of the sampling rate to the spectrum sparsity).

Plots of the reconstruction RMSE versus the number of iterations in Figs. 4.4–4.6 show that the RMS of reconstruction errors decay at first couple of hundreds iterations quite rapidly but after they reach the value of about 2–3 quantization intervals, the error decay slows down. It was found in the experiments that one can substantially accelerate the error decay, if the number of samples is taken with a certain small redundancy, i.e., 10–20% larger than the minimal number equal to the area of the chosen-spectrum EC-zone-approximating shape (see the results for test image "Barbara512" in Table 4.1).

Table 4.1 Summary of experimental results.

				Sampling Redundancy		
Test Image	**Reconstruction RMSE (Peak Signal to RMSE Ratio)**	**Spectrum Sparsity**	**Sampling Rate**	**Chosen EC Zone Redundancy Approx. Shapes**	**Sampling Redundancy for the Chosen Shape**	**Overall Sampling Redundancy**
AerialPhoto 512	1.26 (46.2 dB)	0.227	0.365	1.6	1	1.6
AFM1024	1.1 (47.3 dB)	0.154	0.226	1.46	1	1.46
Barbara512	4.02 (36.1 dB)	0.265	0.412	1.55	1	1.55
	0.69 (51.4 dB)	0.265	0.474	1.55	1.15	1.78
BloodVessels 512	1.94 (42.4 dB)	0.164	0.257	1.56	1	1.56
Mushrooms 512	1.46 (44.9 dB)	0.159	0.226	1.41	1	1.41
Nish1024	1.13 (47.1 dB)	0.17	0.25	1.47	1	1.47
Pirat1024	1.15 (46.9 dB)	0.245	0.42	1.61	1	1.61
Rome512	1.92 (42.5 dB)	0.314	0.5	1.59	1	1.59
Test4CS1024	1.06 (47.7 dB)	0.09	0.136	1.53	1	1.53
WestConcord	1.96 (42.3 dB)	0.314	0.521	1.73	1	1.73

To summarize, the experiments confirm that images sampled with sampling rates equal to the minimal rate for the chosen-spectrum EC-zone-approximating shape of images can be reconstructed with sufficiently good accuracy. The redundancy in the number of required samples associated with the redundancy of the standard approximating shapes of image-spectra EC zones is of the order 1.5–1.6, never exceeding 2.0 in experiments with other images. These figures are estimates of the sampling redundancy of the described ASBSR method.

4.3 Some Practical Issues

This section addresses four practical issues when applying the suggested ASBSR method of image sampling and reconstruction: (i) robustness of the method to the presence of noise in sensor data; (ii) practical considerations regarding choosing shapes for bounding image spectra in image sampling and reconstruction; (iii) image anti-aliasing pre-filtering; and (iv) computational complexity of the method.

Noise robustness of the ASBSR method. Based on the description in Section 4.1, the ASBSR method, just like the conventional sampling and reconstruction, is linear, i.e., it satisfies the superposition principle. No parameter of sampling and reconstruction algorithms depends on signal values and, in particular, on the presence of noise in the signal. Therefore, the sampling and reconstruction of an image that contains additive noise will result in a reconstructed image that also contains additive noise with a power spectrum bounded by the shape of the spectrum EC zone used for reconstruction. In particular, if the sensor noise with variance σ^2 has a uniform power spectrum within the sampling baseband, noise in the reconstructed image will have variance $\kappa\sigma^2$, where $\kappa < 1$ is the relative area of the reconstruction-spectrum EC zone approximating shape, and its power spectrum will be uniform within the spectrum EC zone and zero outside it.

This conclusion is illustrated in Fig. 4.7 by the results of image sampling and reconstruction with and without additive noise. The presence of noise in sampled data has no influence on the work of the reconstruction algorithm.

Choosing the approximating shape of a spectrum EC zone to bound image spectra for image sampling and BS reconstruction. As mentioned in Section 4.1, no fine-tuning is practically required to specify the image-spectrum EC-zone-shape parameters. Therefore, it is suggested that several standard shapes, such as those shown in Fig. 4.1, should be found for different classes of images such as landscape, portrait, micrographs, aerial and space photographs, etc. This can be based, for instance, on a machine-learning algorithm trained on various image databases. To sample images in a particular application, the user should only specify an image class (this is the standard option for setting the parameters of modern digital photocameras).

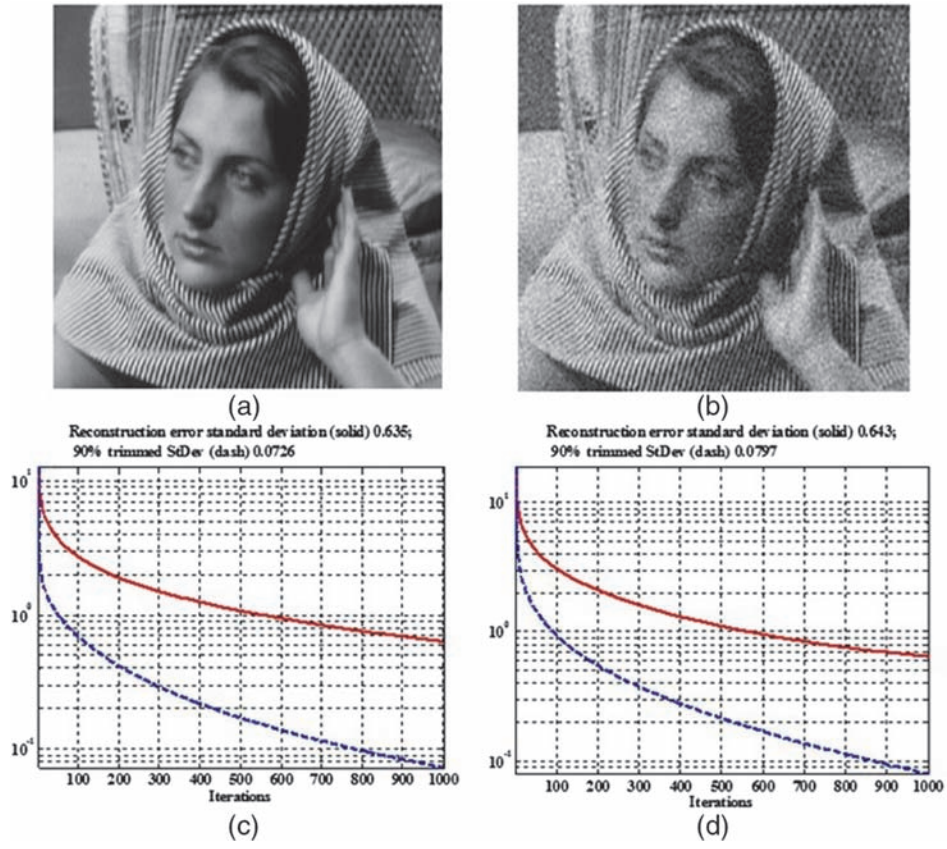


Figure 4.7 Images reconstructed (a) from a sampled noiseless test image and (b) from the same sampled image contaminated with additive uncorrelated Gaussian noise with a standard deviation of 20 gray levels; and (c), (d) the corresponding graphs of the RMSE of all (solid lines) and of the smallest 90% (dashed lines) reconstruction errors vs. the number of iterations.

Image anti-aliasing pre-filtering. As indicated in the previous section, image pre-filtering before sampling is necessary to avoid aliasing effects and to secure convergence of the iterative algorithm to an image with a bounded spectrum. In ordinary imaging devices, anti-aliasing pre-filtering is carried out by imaging optics together with the aperture of the image photosensor. The ASBSR method generally includes choosing a spectrum EC zone shape for each image. Ordinary photosensors cannot implement this choice. To solve this problem, multiple aperture sensors can be used, such as those suggested in Ref. 21. The required effective aperture of these sensors defined by the inverse Fourier transform of the chosen-spectrum EC zone shape is synthesized by combining, with appropriate weights, the signals from individual apertures. The multiple aperture sensors are well suited for so-called *single-pixel cameras*, where sampling is carried out using *digital micro-mirror devices*.

Computational complexity. The computational complexity of the image BS-reconstruction iterative algorithm, per iteration, is determined by the complexity $O(2N\log N)$ of floating point operations for direct and inverse fast transforms plus $O(N)$ replacement operations of sample-wise modifications of data in the image domain (M operations) and in its transform domain ($N - M$ operations). The order of magnitude of time required for one iteration can be estimated from these data: for example, the elapsed time for a MATLAB direct or inverse DCT of an array of $N = 512 \times 512$ numbers implemented on a PC (Lenovo ThinkPad X201 with an Intel i7 processor and Windows 7 operating system) is 52 ms.

4.4 Other Possible Applications of the ASBSR Method of Image Sampling and Reconstruction

The above discussed task of reconstruction of images of N samples from $M < N$ sampled data can be considered as a special case of underdetermined inverse imaging problems. The solution to this problem, i.e., BS image reconstruction, can be used to solve other underdetermined inverse imaging problems, as well: (i) image super-resolution from multiple chaotically sampled images, (ii) image reconstruction from sparsely sampled or decimated projections, and (iii) image reconstruction from sparsely sampled Fourier spectra.

4.4.1 Image super-resolution from multiple chaotically sampled video frames

The first potential application is image super-resolution from multiple video frames with chaotic pixel displacements due to atmospheric turbulence, camera instability, or similar random factors.[23] By means of elastic registration of the sequence of frames of the same scene, one can determine for each image frame and with sub-pixel accuracy (using the methods of image correlation decribed in Section 5.3.3) the pixel displacements caused by the random acquisition factors. Using these data, a synthetic fused image can be generated by placing pixels from all available video frames in their proper positions on the correspondingly denser sampling grid according to their displacements. During this process, some pixel positions on the denser sampling grid will remain unoccupied, especially when a limited number of image frames are fused. These missing pixels can then be reconstructed using the iterative BS-reconstruction algorithm described earlier. Figure 4.8 shows an example of super-resolution from multiple chaotically sampled video frames. The figure includes one of the low-resolution frames, an image fused from 50 frames with pixel chaotic displacements, and the result of image reconstruction from the fused image.[23]

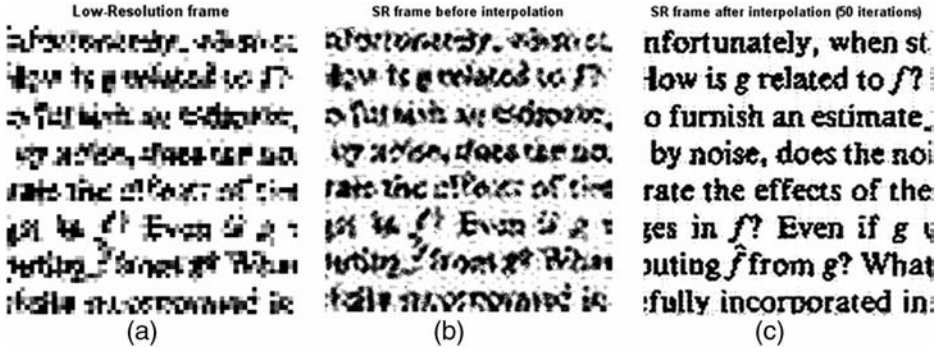


Figure 4.8 Image reconstruction with super-resolution from multiple chaotically sampled frames: (a) one of the low-resolution frames; (b) the image fused by elastic image registration from 50 frames; and (c) the result of iterative reconstruction with enhanced resolution, obtained from image (b) after 50 iterations.

4.4.2 Image reconstruction from their sparsely sampled or decimated projections

In computed tomography, a body slice often occupies only a fraction of the entire image frame. This means that slice projections are Radon-transform "bounded spectrum" functions. Therefore, regardless of how many projections or their samples are available, additional projections or samples commensurate with the size of the slice's empty zone, according to the discrete sampling theorem, can be obtained, and a corresponding resolution increase in the reconstructed images can be achieved using image BS reconstruction. This option is illustrated in Figs. 4.9 and 4.10.

Figure 4.9 demonstrates the recovery of missing samples of image slice projections sampled in random positions. In the experiment, simple segmentation of the test image (Fig. 4.9(a)) found that the outer 55% of the image area is empty. The same percentage of projection samples, selected randomly using the MATLAB random number generator, were then zeroed (Fig. 4.9(c)). The rest of the samples were used to recover missing samples and, correspondingly, for image reconstruction with an iterative reconstruction algorithm identical to the image BS-reconstruction algorithm described earlier except that the direct and inverse DCTs were replaced by direct and inverse discrete Radon transforms.[15]

At each iteration of the algorithm, the current set of projections is subjected to inverse Radon transform for obtaining a current estimate of the reconstructed image. The outer empty zone of the reconstructed image is then zeroed, and this modified image is subjected to a direct Radon transform to obtain the next estimate of slice projections. In the obtained projections, their available samples are restored, and the process repeats. The plot of the RMSE of the slice-projection reconstruction error versus the iteration number shown

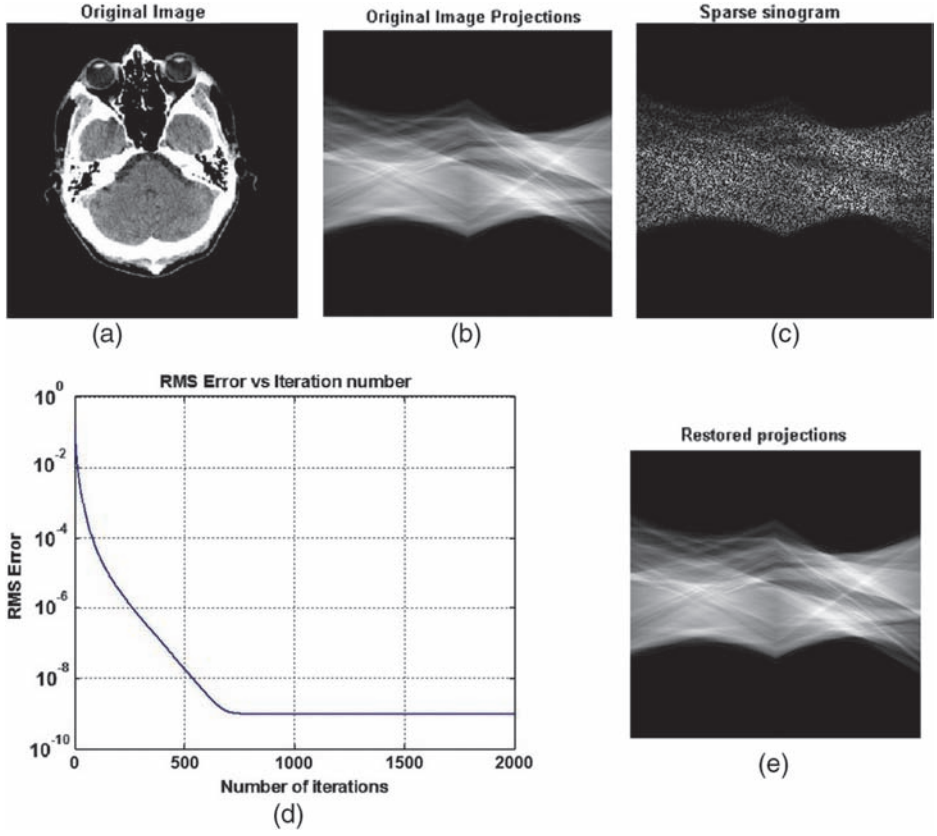


Figure 4.9 Recovery of randomly sampled slice projections: test image (a); its slice projections (b); slice projections randomly sub-sampled with rate 0.45 (c); plot of slice projections reconstruction RMSE vs. the iteration number (d); the recovered slice projections (e).

in Fig. 4.9(d) and the result of recovering missing samples (Fig. 4.9(e)) show that virtually perfect recovery of the missing 55% samples of slice projections is possible with the iterative reconstruction algorithm after a few hundred iterations.

Figure 4.10 illustrates that recovery of completely missing projections is also possible. In the experiment, every second projection of the test image shown in Fig. 4.9(a) was removed (Fig. 4.10(b)), and then all initial projections were recovered (Fig. 4.10(c) by the above described iterative algorithm that functions because the outer 55% of the image area is known to be empty. This result suggests that for such cases, when half or more of the image area is known to be empty, one can use image BS reconstruction to achieve image reconstruction with super-resolution that corresponds to a larger number of image projections than are available.

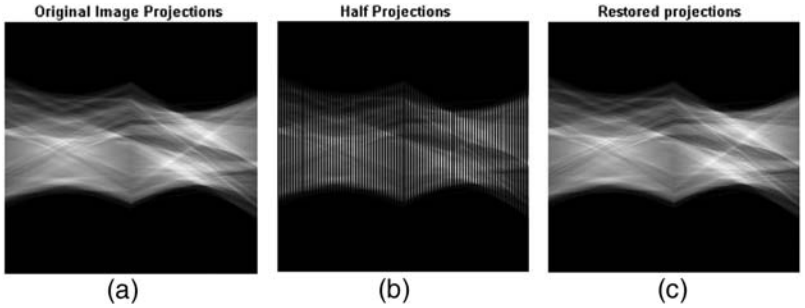


Figure 4.10 (a) Recovery of missing image projections: original projections of the test image of Fig. 4.9(a); (b) decimated projections with every second projection removed; and (c) slice projections recovered from (b) using the iterative reconstruction algorithm.

4.4.3 Image reconstruction from sparsely sampled Fourier spectra

Some imaging devices (e.g., some healthcare scanners) sample in the transform domain. The proposed ASBSR method can be used in such devices in common cases, when it is known that the object image is surrounded by empty space. Figure 4.11 demonstrates this option with an example of image reconstruction from its sparsely sampled Fourier spectrum.

In this example, the Fourier spectrum of the test image bounded by a circular binary image mask was randomly sampled with a sampling rate equal to the ratio of the image bounding circular area to the area of the entire image frame. Additionally, the spectrum was bounded by a circular binary spectral mask with a radius equal to the highest horizontal and vertical spatial frequency of the baseband. This gives an additional $1 - \pi/4 = 21.5\%$ savings in the number of spectrum samples.

For image reconstruction, the iterative algorithm was used. At each iteration, the iterated spectrum is inverse Fourier transformed to obtain an iterated reconstructed image, and then the latter is multiplied by the bounding circular image mask and Fourier transformed. Samples of the obtained spectrum in the positions of available ones are replaced with those, and the spectrum is bounded by the circular binary spectral mask to form an iterated spectrum for the next iteration. The initial sparsely sampled and bounded spectrum was used as a spectrum zeroth-order approximation, from which the iterative reconstruction starts.

4.5 Exercises

ArbitrSamplingAndBSReconstr_SPIE.m

Simulation on a discrete model that uses the ASBSR method of image sampling and reconstruction, featuring a DCT bounded spectrum iterative reconstruction of images sampled with a sampling rate (SR) equal to (in fractions of the sampling baseband in DCT domain) the area of a user-defined

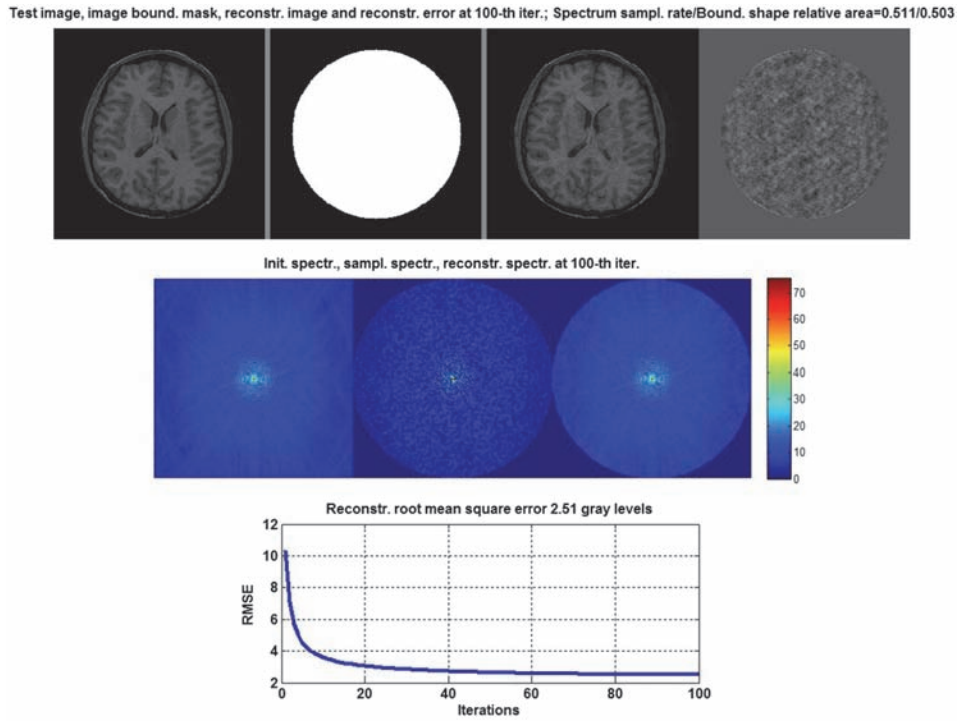


Figure 4.11 Image reconstruction from a sparsely sampled spectrum. Upper row, from left to right: test image bounded by a binary circular mask with radius equal to 0.35 of the image size; image bounding mask with bounding area 0.5; reconstructed image at the hundredth iteration with the RMS of the reconstruction error equal to 2.5 gray levels (PSNR = 40 dB); and pattern of reconstruction errors (the difference between test and reconstructed images). Middle row, from left to right: absolute values of the Fourier spectrum of the test image, of its spectrum randomly sampled with a sampling rate of 0.51, and of the reconstructed spectrum. For display purposes, the absolute values of spectral samples are displayed raised to a power of 0.3 and are shown with MATLAB color map "jet." Bottom row: plot of the reconstruction RMSE vs. the number of iterations.

spectrum bounding shape times a user-defined sampling redundancy factor Rednd (slightly larger than one). The latter is recommended to accelerate iteration convergence. The number of iterations is a user-defined parameter.

Four types of the spectrum-bounding shapes can be chosen:

- Rectangle;
- Triangle;
- Oval; and
- Superellipse

Three types of sampling lattices can be used:

- **Uniform**: The positions of image samples are rounded off to positions of the regular square sampling lattice that corresponds to the chosen

numbers of samples NofSamplX, NofSamplY (square root of the product of image display size SzXxSzY and sampling rate);
- **Uniform with random jitter**: The X and Y coordinates of image samples are randomly distributed between nodes of the square sampling lattice of NofSamplXxNofSamplY pixels; and
- **Totally random**: The positions of image samples are randomly distributed over nodes of the image display sampling lattice.

Zeroth-order approximation of the reconstructed image, from which iterative reconstruction starts, is obtained by weighted summation of available samples over Nint available pixels nearest to the sample being interpolated within a window of pre-defined size Lxy with interpolation weights inversely proportional to the distance of each nearest available sample from the position of the pixel to be interpolated (subprogram Interpolation_Random-Grid_spie.m).

Displayed are

- a test image,
- a sampled test image,
- a reconstructed image,
- a map of test image sparse spectrum and border of the chosen image spectrum bounding shape,
- a plot of the RMS reconstruction error versus the number of iterations, and
- the pattern of reconstruction error (difference between test and reconstructed images).

The reconstructed image, reconstruction errors, and plots of the RMS reconstruction error are displayed at each of the first 100 iterations and then at each hundredth iteration. Non-square images are cropped to a square shape.

SparseSampl_Recon_DFTspectrum_SPIE.m

Image reconstruction from its sparsely sampled DFT spectrum. The image is supposed to be bounded by a circle with a user-defined radius. The image DFT spectrum is sampled in random positions with a sampling rate equal to the relative area of the image bounding circle times a user-defined sampling redundancy coefficient. Slight redundancy (5–10%) is useful to accelerate the algorithm convergence.

Chapter 5
Signal and Image Resampling, and Building Their Continuous Models

5.1 Signal/Image Resampling as an Interpolation Problem; Convolutional Interpolators

Many image processing applications must resample available digital images in positions other than the original ones. Such applications include fusing image data from different image modalities, building image mosaics from many partly overlapped images, reconstructing images from projections, producing image super-resolution from video sequences, stabilizing video images distorted by atmosphere turbulence, locating targets, and tracking with sub-pixel accuracy, to name a few.

Image resampling assumes that approximations of the original non-sampled images are built by interpolating available image samples and then resampling the obtained models in the required new positions. The most feasible and amenable to optimization is signal interpolation by means of digital convolution:

$$\tilde{a}_k = \sum_{n=0}^{N-1} h_n^{(\mathrm{intp})} a_{k-n}, \tag{5.1}$$

where $\{\tilde{a}_k\}$ are samples of a signal obtained as a result of resampling the initial signal samples $\{a_k\}$, $\left\{h_n^{(intp)}\right\}$ are samples of the interpolation filter point spread function (PSF), and N is the number of available signal samples.

The problem of interpolating numerical data is one of the classic mathematical problems that can be traced back to Babylonian times.[22] Mathematical geniuses such as Newton, Euler, and Gauss contributed to its solution. Presently, numerous interpolation methods are known. The most popular in signal and image processing are convolutional methods, from the

simplest nearest-neighbor and linear (bilinear, for the 2D case) interpolations to more accurate cubic (bicubic, for the 2D case) and higher-order spline methods. None of them, however, is perfect. According to the sampling theorem (Section 2.5), sinc interpolation approximates continuous signals from their samples with minimal RMS error. However, sinc interpolation reconstruction requires an infinite number of samples, whereas digital signals always have a finite number of samples. In the next section we show that *discrete sinc interpolation*, i.e. interpolation with *discrete sinc function* as an interpolation kernel, is a perfect discrete representation of continuous sinc-interpolation and is a gold standard for numerical interpolation of samples signals.

5.2 Discrete Sinc Interpolation: A Gold Standard for Signal Resampling

Consider the design of a *perfect resampling filter*, i.e., determination of the PSF $\left\{h_n^{(intp)}\right\}$ of an interpolation filter that is a discrete representation of the continuous sinc function. To solve this problem, one can regard signal coordinate shift as a general resampling operation, because samples of the resampled signal for any required positions can be obtained one by one through the corresponding shifts of the original signal to the given sample positions.

In order to derive the perfect shifting filter, we will need the following characteristics of digital filters and their properties, formulated in Sections 9.2 and 10.5:

- Discrete frequency response $\{\eta_r\}$ of a digital filter with PSF $\{h_n\}$ applied to the sampled signals of N samples. It is the DFT of its PSF:

$$\eta_r = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} h_n \exp\left(i2\pi \frac{nr}{N}\right). \tag{5.2}$$

- Continuous frequency response $DFCFR(f)$ of a digital filter with the discrete frequency response $\{\eta_r\}$. According to Eq. (10.68), it is a function

$$DFCFR(f) = N \sum_{r=0}^{N-1} \eta_r^{\left(-\frac{N-1}{2},0\right)} \mathbf{sincd}[N,\ \pi(f/\Delta_f - r)], \tag{5.3}$$

interpolated from $\{\eta_r\}$ with the discrete sinc function

$$\mathbf{sincd}(N,\ x) = \frac{\sin(x)}{N\sin(x/N)} \tag{5.4}$$

as an interpolation kernel, i.e., coefficients $\{\eta_r\}$ of the discrete frequency response of a digital filter are samples of its continuous frequency response $DFCFR(f)$ taken with sampling interval Δ_f at sampling points $\{r\Delta_f\}$:

$$\{\eta_r = DFCFR(f = r\Delta_f)\}. \tag{5.5}$$

- Digital filter overall frequency response $DFOFR(f)$, i.e., frequency response of a digital filter with respect to continuous signals that correspond to samples of the filter input signals. For the ideal signal sampling and reconstruction anti-aliasing low-pass filters, the overall frequency response $DFOFR(f)$ of a digital filter coincides with the digital filter continuous frequency response $DFCFR(f)$ within the signal baseband $[-1/2\Delta_x, 1/2\Delta_x]$ defined by the signal sampling interval Δ_x.

In these formulas, f is the frequency parameter of the integral Fourier transform, and Δ_f is the spectrum sampling interval for signal sampling interval Δ_x and cardinal sampling $\Delta_f = 1/N\Delta_x$ (Eq. (10.10)).

According to these definitions, the overall continuous frequency response $DFOFR_{\delta x}^{(Shift)}(f)$ of the perfect shifting filter for the x-coordinate shift δ_x must equal (within the signal baseband $[-1/2\Delta_x, 1/2\Delta_x]$) the frequency response $H_{\delta\tilde{x}}^{(Shift)}(f)$ of the continuous δ_x-shifting filter. The latter, by virtue of the Fourier transform shift theorem (Appendix A1.4), equals $\exp(i2\pi f\delta_x)$. Therefore,

$$DFOFR_{\delta\tilde{x}}^{(Shift)}(f) = H_{\delta\tilde{x}}^{(Shift)}(f) = \exp(i2\pi f\delta_x);\ (-1/2\Delta_x < f < 1/2\Delta_x). \tag{5.6}$$

Based on the formulated properties of the overall, continuous, and discrete frequency responses of digital filters, it follows that the discrete frequency response coefficients $\left\{\eta_r^{(\delta_x)}\right\}$ of the perfect δ_x-shifting filter, for indices from $r = 0$ to $r = (N-1)/2$ for odd N and to $r = N/2$ for even N (which correspond to the lowest and highest frequencies in the signal baseband), must be samples in the sampling points $\{r\Delta_f = r/N\Delta_x\}$ of its continuous frequency response. The rest of the coefficients should be set according to the symmetry property $\eta_r^{(\delta_x)} = \eta_{N-r}^{*(\delta_x)}$ of the DFT for real valued data (Eq. (A2.36)). Thus, for an odd number of signal samples N, coefficients $\left\{\eta_r^{(\delta x)}\right\}$ must be set to

$$\eta_r^{(\bar{\delta}_x)} = \begin{cases} \frac{1}{\sqrt{N}}\exp\left(i2\pi\frac{r\bar{\delta}_x}{N}\right), & r = 0,\ 1,\ \ldots,\ (N-1)/2 \\ \left(\eta_{N-r}^{(\bar{\delta}_x)}\right)^*, & r = (N+1)/2,\ \ldots,\ N-1, \end{cases} \tag{5.7}$$

where shift δ_x is expressed in fractions of the signal sampling interval:

$$\bar{\delta}_x = \delta_x / \Delta_x, \tag{5.8}$$

the asterisk $*$ symbolizes the complex conjugacy, and the multiplier $1/\sqrt{N}$ is introduced for normalization purposes.

For an even number of signal samples N, from the same requirement $\eta_r^{(\delta_x)} = \eta_{N-r}^{*(\delta_x)}$ it follows that coefficient $\eta_{N/2}^{(\delta_x)}$, which corresponds to the signal's higher frequency in its baseband, must be a real number. Thus, for even N this coefficient cannot be taken just as a sample of $\exp(i2\pi r\bar{\delta}_x/N\Delta_x)$ for $r = N/2$ and requires special treatment. The most natural setting is

$$\eta_{r,opt}^{(\delta_x)} = \begin{cases} \frac{1}{\sqrt{N}} \exp(i2\pi r\bar{\delta}_x/N), & r = 0,\ 1,\ \ldots,\ N/2-1 \\ \frac{C}{\sqrt{N}} \cos(\pi\bar{\delta}_x), & r = N/2 \\ \eta_{r,opt}^{(\delta_x)} = (\eta_{N-r,opt}^{(\delta_x)})^*, & r = N/2+1,\ \ldots,\ N-1, \end{cases} \tag{5.9}$$

where C is a weight coefficient that defines the signal spectrum shape at its highest-frequency component $r = N/2$.

For even N, the following three options for C are considered:

- Case 0 : $C = 0$,
- Case 1 : $C = 1$, and
- Case 2 : $C = 2$. (5.10)

Find the point spread function of the perfect shifting filter for an odd number of samples N with a discrete frequency response defined by Eq. (5.7):

$$\begin{aligned} h_n^{(\delta_x)} &= \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \eta_r^{(\bar{\delta}_x)} \exp\left(-i2\pi\frac{nr}{N}\right) = \\ &\frac{1}{\sqrt{N}} \left\{ \sum_{r=0}^{(N-1)/2} \eta_r^{(\bar{\delta}_x)} \exp\left(-i2\pi\frac{nr}{N}\right) + \sum_{r=(N+1)/2}^{N-1} \eta_r^{(\bar{\delta}_x)} \exp\left(-i2\pi\frac{nr}{N}\right) \right\} = \\ &\frac{1}{\sqrt{N}} \left\{ \sum_{r=0}^{(N-1)/2} \eta_r^{(\bar{\delta}_x)} \exp\left(-i2\pi\frac{nr}{N}\right) + \sum_{r=1}^{(N-1)/2} \eta_{N-r}^{(\bar{\delta}_x)} \exp\left[-i2\pi\frac{n}{N}(N-r)\right] \right\} = \\ &\frac{1}{N} \left\{ \sum_{r=0}^{(N-1)/2} \exp\left(i2\pi\frac{\bar{\delta}_x}{N}r\right) \exp\left(-i2\pi\frac{nr}{N}\right) + \sum_{r=1}^{(N-1)/2} \exp\left(-i2\pi\frac{\bar{\delta}_x}{N\Delta x}\right) \exp\left(i2\pi\frac{nr}{N}\right) \right\} = \\ &\frac{1}{N} \left\{ \sum_{r=0}^{(N-1)/2} \exp\left(-i2\pi\frac{n-\bar{\delta}_x}{N}r\right) + \sum_{r=1}^{(N-1)/2} \exp\left(i2\pi\frac{n-\bar{\delta}_x}{N}r\right) \right\}. \end{aligned} \tag{5.11}$$

Now, Eq. (A2.8) can be used to obtain

$$h_n^{(\delta_x)} = \mathrm{sincd}[N,\ \pi(n - \bar{\delta}_x)], \tag{5.12}$$

where

$$\mathrm{sincd}(N;\ x) = \frac{\sin x}{N \sin(x/N)} \tag{5.13}$$

is the discrete sinc function, or *sincd function*.

In a similar way, one can obtain that for even N (Cases 0 and 2); the perfect shifting filter PSFs are

$$h_n^{(\bar{\delta}_x,\ 0)} = \mathrm{sincdd}[N - 1;\ N;\ \pi(n - \bar{\delta}_x)] \tag{5.14}$$

and

$$h_n^{(\delta_x,2)}() = \mathrm{sincdd}[N + 1;\ N;\ \pi(n - \delta_x/\Delta x)], \tag{5.15}$$

respectively, where $\mathrm{sincdd}(M;\ N;\ x)$ is the *sincdd function* defined by Eq. (A2.49):

$$\mathrm{sincdd}(M;\ N;\ x) = \frac{\sin(Mx/N)}{\sin(x/N)}. \tag{5.16}$$

Case 1 is an additive combination of Cases 0 and 2:

$$\begin{aligned} h_n^{(\bar{\delta}_x,\ 1)} &= [h_n^{(\delta_x,\ 0)}(\bar{\delta}_x) + h_n^{(\delta_x,\ 2)}(\bar{\delta}_x)]/2 \\ &= \frac{\sin\left[\pi(N-1)\frac{(n-\bar{\delta}_x)}{N}\right] + \sin\left[\pi(N+1)\frac{(n-\bar{\delta}_x)}{N}\right]}{N \sin\left(\pi\frac{n-\bar{\delta}_x}{N}\right)} \\ &= \frac{\sin[\pi(n - \bar{\delta}_x)]}{N \sin[\pi(n - \bar{\delta}_x)/N]} \cos[\pi(n - \bar{\delta}_x)/N], \end{aligned} \tag{5.17}$$

that is,

$$h_n^{(\delta_x,1)}(\bar{\delta}_x) = \cos\left(\pi\frac{n - \bar{\delta}_x}{N}\right) \mathrm{sincd}[N;\ \pi(n - \bar{\delta}_x)]. \tag{5.18}$$

These three versions of sincd functions for Cases 0, 1, and 2 are illustrated for comparison in Fig. 5.1.

The figure shows that the sincd function for Case 1 (Eq. (5.18)), apodized by the cosine window, converges to zero substantially faster than for Cases 0 and 2 by halving the highest-frequency spectral coefficient $\eta_{N/2}^{(\mathrm{int}\,1)}$. Thus, signal

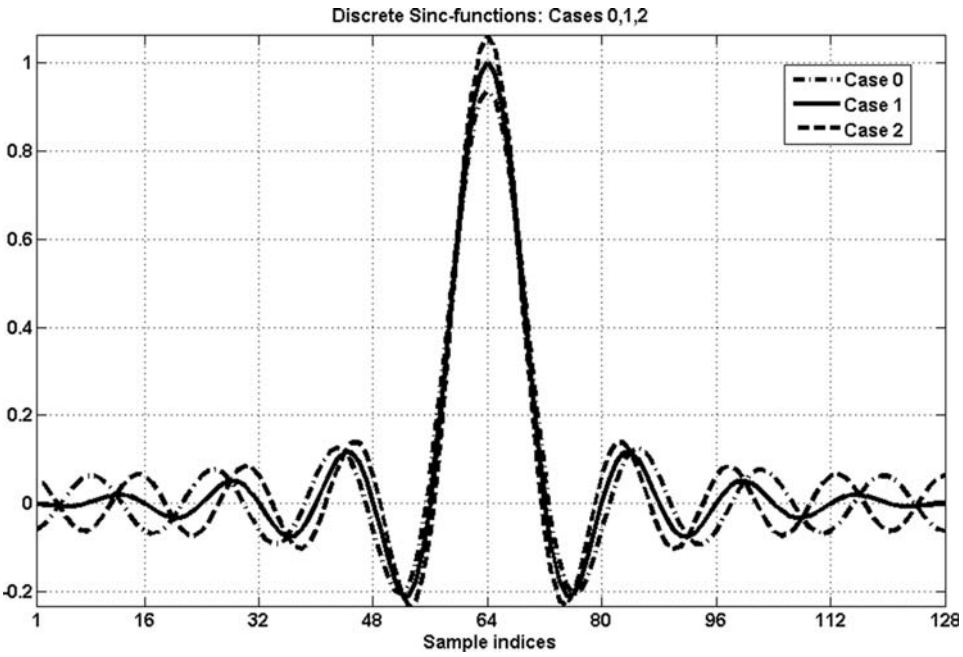


Figure 5.1 Comparison of three versions of sincd functions.

interpolation using this function as the interpolation kernel produces fewer ringing oscillations in the interpolated signal at the signal edges, which makes it preferable in practical applications.

Numerical interpolation using the discrete sincd functions given by Eqs. (5.12), (5.14), (5.15), and (5.18) as the interpolation kernel is called *discrete sinc interpolation*.

Figure 5.2 shows the continuous frequency response of the discrete sinc interpolator (solid line) along with samples (stems) of the discrete frequency response of the discrete sincd interpolator.

By definition (Eqs. (5.7) and (5.9)), the discrete frequency response of the discrete sincd interpolator is by modulus a flat function within the signal baseband except for halving the highest signal $N/2$th frequency component for Case 1 of the discrete sincd function for an even number N of signal samples. This implies that discrete sinc interpolation does not distort the intensity of the signal discrete spectrum within the baseband, and therefore it secures perfect (for a given number of signal samples) resampling of discrete signals that preserves the corresponding continuous signal power spectra in their sampling points. All resampling filters with PSFs other than the discrete sincd function will distort samples of the signal spectrum in the signal baseband and, therefore, introduce additional interpolation error to the signal distortions due to sampling. It is in this sense that discrete sinc interpolation can be regarded as the "gold standard" of discrete signal interpolation.

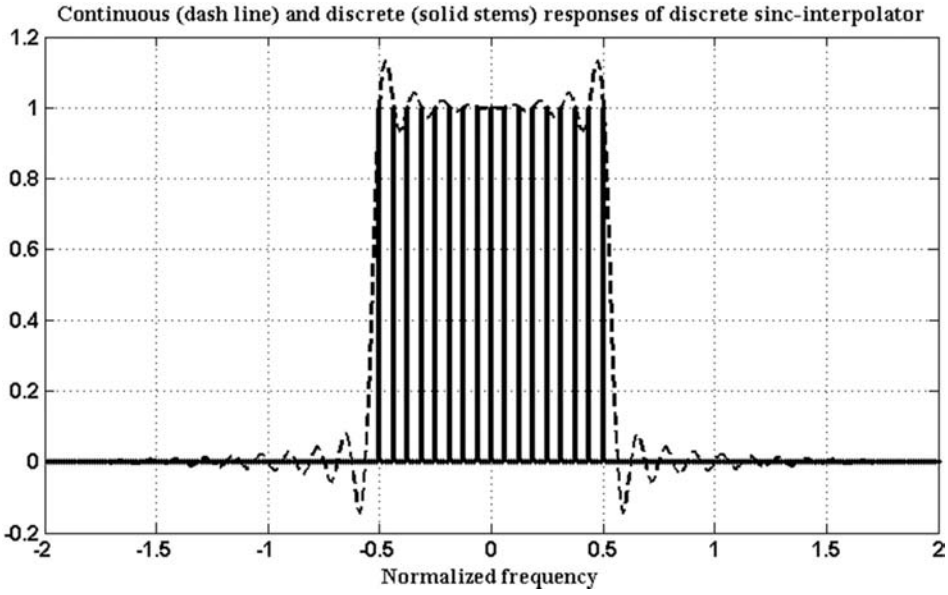


Figure 5.2 Continuous (solid line) and discrete (solid stems) frequency responses of the discrete sinc interpolator. Frequency indices are normalized by the width of the signal sampling baseband.

Section 5.4 will compare discrete sinc interpolation with other methods and provide experimental evidence of its perfect performance and superiority.

5.3 Fast Algorithms of Discrete Sinc Interpolation and Their Applications

5.3.1 Signal sub-sampling with DFT or DCT spectral zero-padding

One of the basic image resampling tasks is image sub-sampling (zooming in), i.e., computing a set of intermediate samples from the given set of samples. From properties of signal DFT spectra of sparse signals, discussed in Appendix A2.9 (Eqs. (A2.47), (A2.48), (A2.50), and (A2.61)) it follows that discrete sinc interpolated signal sub-sampling can be achieved by zero-padding its DFT spectrum. Given the desired nomber N of the zoomed-in signal samples and the number N_0 of samples of the original signal, this algorithm is described by the following equation:

$$\tilde{a}_{\tilde{k}} = \mathrm{IFFT}_N\{\mathrm{DFT\_ZP}_{N/N_0}[\mathrm{FFT}_{N_0}(a_k)]\}, \tag{5.19}$$

where $\{a_k\}$, $k = 0, 1, \ldots, N_0 - 1$ are the initial signal samples, $\{\tilde{a}_{\tilde{k}}\}$ are zoomed-in signal samples, $\tilde{k} = 0, 1, \ldots N - 1$, $\mathrm{FFT}_{N_0}(\cdot)$ and $\mathrm{IFFT}_N(\cdot)$ are N_0-point direct and N-point inverse fast Fourier transform operators, respectively, and $\mathrm{DFT\_ZP}_{N/N_0}[\,\cdot\,]$ is a zero-padding operator. The zero-padding

operator forms from an N_0-point sequence of samples of the DFT spectrum of signal $\{a_k\}$ an N-point sequence by padding the former with a corresponding number of zeros. Due to the symmetry properties of the signal DFT spectra, the way in which the signal spectrum shoud be padded with zeros depends on whether the number N_0 of signal spectrum is odd or even (Fig. 5.3).

When N_0 is an odd number, $N - N_0$ zeros are placed between the $(N-1)/2$th and $(N+1)/2$th samples of the N_0-point sequence of spectral coefficients. When N_0 is an even number, there are three options:

i. $N - N_0 + 1$ zeros are placed between the $N_0/2 - 1$th and $N_0/2 + 1$th samples of the N_0-point sequence of spectral coefficients, and the $N_0/2$th sample is zeroed;
ii. $N - N_0 - 1$ zeros are placed after the $N/2$th sample, and then the sequence of spectral coefficients is repeated beginning from its $N_0/2$th sample; and
iii. $N_0/2$th sample of the sequence is halved, $N - N_0 - 1$ zeros are placed after it, and then the $N_0/2$th through $(N_0 - 1)$th samples of the sequence are placed at the end (the $N_0/2$th sample is also halved).

Cases i–iii implement the previously described Cases 0, 2, and 1 of discrete sinc interpolation, respectively.

Sub-sampling of the 2D signals and image is implemented as separable in two consecutive steps over each of two coordinates.

Although discrete sinc interpolation can perfectly preserve the signal power spectrum, it has one major drawback. With the use of the FFT in the resampling algorithm (Eq. (5.19)), images behave like they are periodic in both coordinates. Therefore, samples at their left and right (and, correspondingly, top and bottom) borders become immediate neighbors in the interpolation process. Any discontinuity between opposite border samples causes heavy oscillations due to the tails of the discrete sincd function that propagate far from the borders.

A simple and very practical solution to this problem is zero-padding in the domain of the DCT. For 1D signals, the DCT zero-padding algorithm for the sub-sampling signal $\{a_k\}$ of N_0 samples ($k = 0, 1, \ldots, N_0 - 1$), used to obtain $N > N_0$ samples of the sub-sampled signal $\{\tilde{a}_{\tilde{k}}\}$, $\tilde{k} = 0, 1, \ldots, N - 1$, is defined by the equation

$$\tilde{a}_{\tilde{k}} = \mathrm{IDCT}_N\{\mathrm{DCT\_ZP}_{N/N_0}[\mathrm{DCT}_{N_0}\{a_k\}]\}, \tag{5.20}$$

where $\mathrm{DCT}_{N_0}(\cdot)$ and $\mathrm{IDCT}_N(\cdot)$ are N_0-point direct and N-point inverse DCTs, and $\mathrm{DCT\_ZP}_L[\,\cdot\,]$ is a DCT spectrum zero-padding operator that places $N - N_0$ zeros after the last $(N_0 - 1)$th DCT spectrum sample. For faster decay of the interpolation kernel, it is also advisable to halve this last

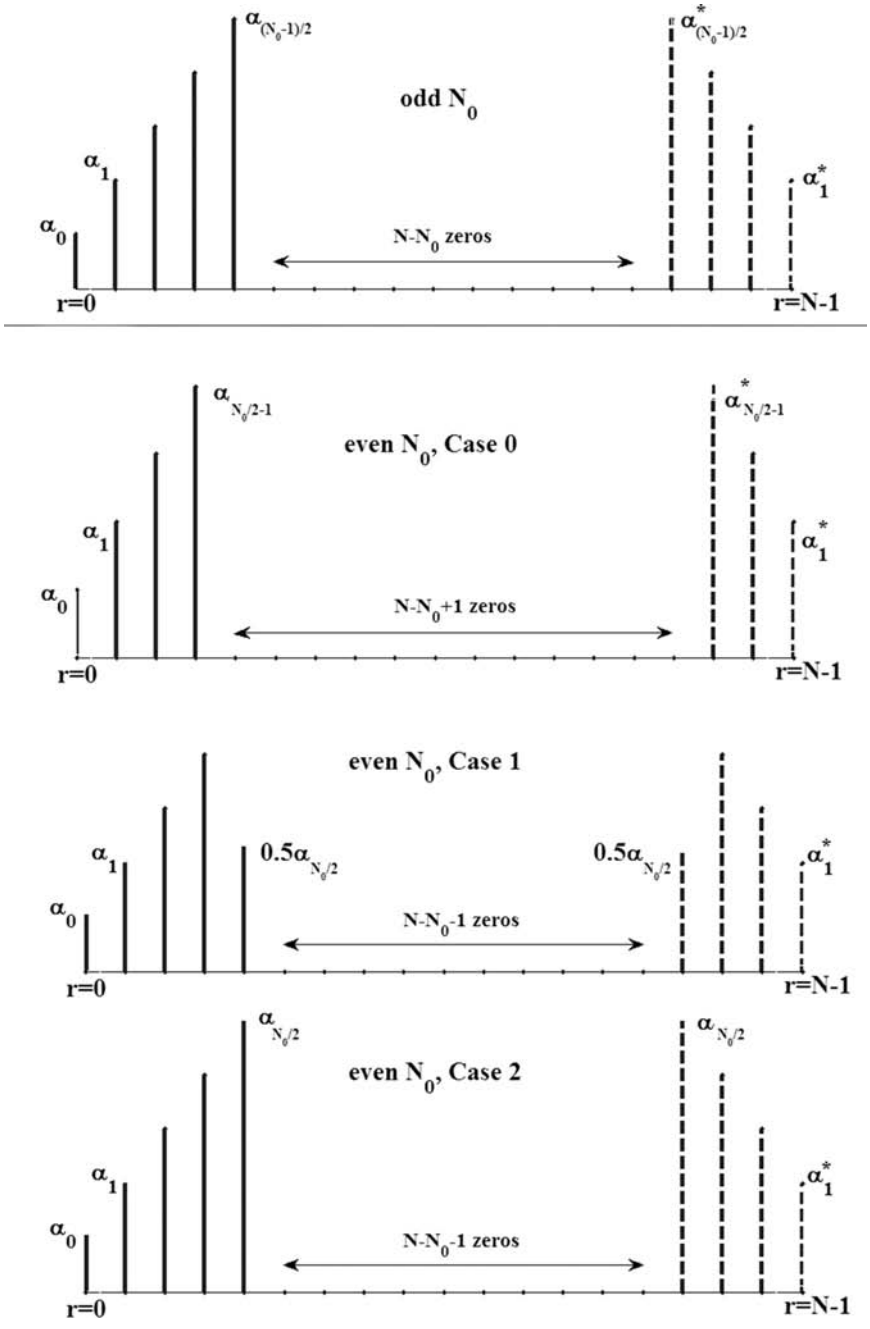


Figure 5.3 Signal DFT spectrum zero-padding for an odd and even number of samples N_0.

sample, which for the DCT represents the signal highest frequency component. For 2D signals and images, this algorithm can be applied separately to both coordinate indices.

The point spread function and frequency response of the DCT spectrum zero-padding with and without halving its highest frequency component do not exactly coincide with those of DFT, but they are very close to them. Figure 5.4 compares the point spread functions and frequency responses of DFT and DCT zero-padding of signal $5\times$ sub-sampling.

The figure shows that halving the signal's highest frequency component fastens decaying oscillations of interpolation point spread functions very substantially.

Contrary to the PSF of DFT zero-padding, which is a cyclic discrete sincd function with a period that equals the number N of zoomed signal samples, the PSF of DCT zero-padding is cyclic with a double period. Thus, signal resampling by means of DCT spectrum zero-padding does not involve signal samples from opposite borders, whereas DFT zero-padding does. Figure 5.5 shows the PSFs of DFT and DCT zero-padding at the left and right borders of the signal interval. This property of DCT zero-padding prevents appearance of oscillations on image boards, which are characteristic for image subsampling my means of DFT spectrum zero-padding. Figure 5.6 demonstrates an improvement, in terms of boundary effects, in image zooming by means of a DCT spectrum zero-padding algorithm compared to the DFT spectrum zero-padding.

Computation-wise, zooming in on a signal N/N_0 times by zero-padding its DFT or DCT spectra requires (with the use of FFT or fast DCT

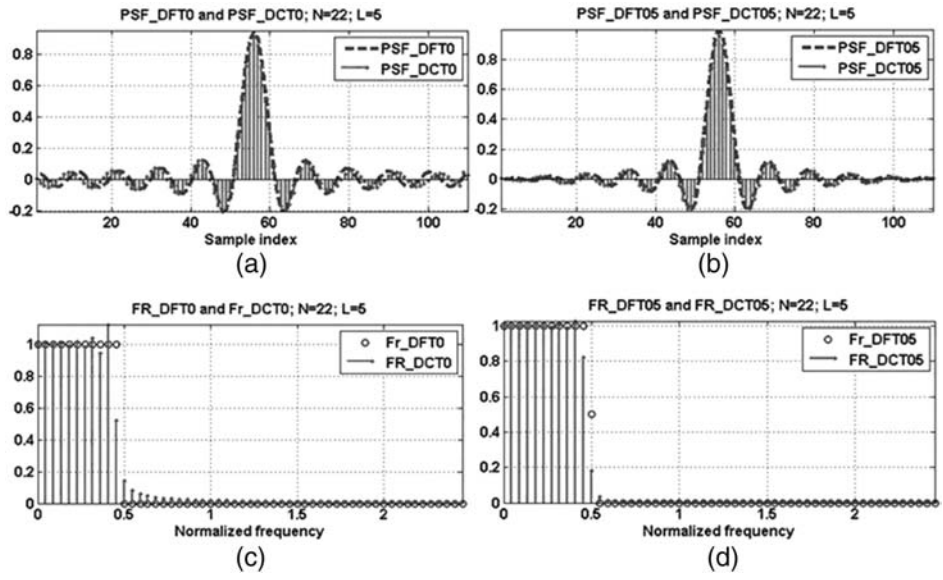


Figure 5.4 Discrete point spread functions (a, b) and frequency responses (c, d) of DFT and DCT zero-padding a signal with $5\times$ sub-sampling ($N = 22$, $L = 5$) after zeroing (a, c) and halving (b, d) the component with the highest signal frequency.

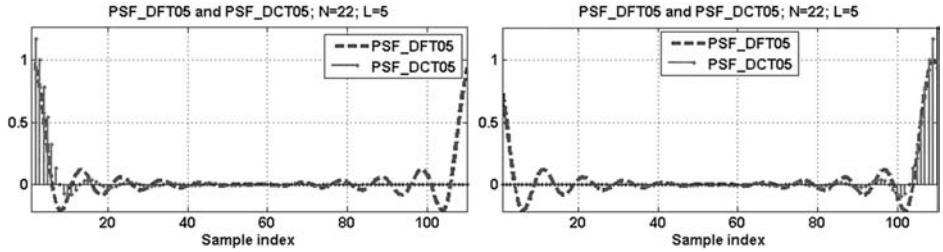


Figure 5.5 Point spread function of 5× signal sub-sampling by means of DFT and DCT zero-padding on the (a) left and (b) right signal borders.

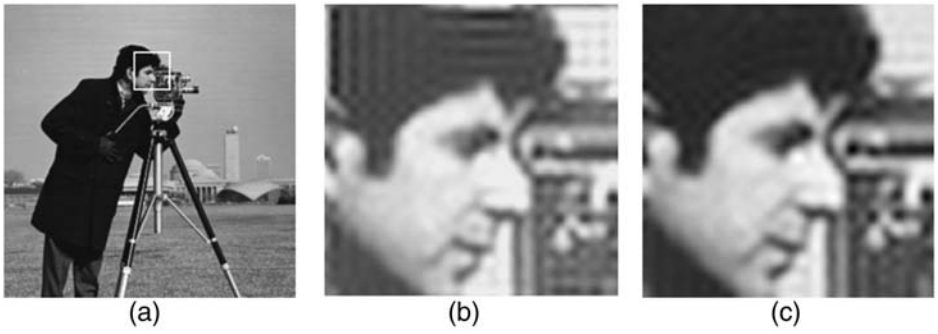


Figure 5.6 (a) Zooming in on an image fragment outlined by a white box (b) by zero-padding its DFT spectrum and (c) by zero-padding its DCT spectrum.

algorithms) $O(N_0 \log N_0)$ operations for the direct transform of the signal and $O(N \log N)$ operations for the inverse transform of the zero-padded spectra. The next section presents DFT- and DCT-based signal subsampling algorithms, which are, for integer zoom factors, less computationally costly.

The spectrum zero-padding method is a good practical solution for image zooming with rational zoom factors. Image zooming with DCT spectral zero-padding can be naturally used when images are represented in a compressed form, e.g., JPEG compression. In this case, zooming can be performed without the need to decompress images.

5.3.2 Signal sub-sampling (zooming-in) by means of DFT- and DCT-based perfect fractional shift algorithms

This section introduces signal sub-sampling algorithms that are alternatives to the above described spectrum zero-padding algorithms and are based on the perfect δ_x-shifting filter introduced in Section 5.2. This filter can be used to perform signal/image zooming (sub-sampling) with an arbitrary integer sub-sampling factor L by repeatedly generating $(L-1)$ signal/image copies $\left\{\tilde{a}_k^{(l/L)}\right\}$ shifted by a corresponding multiple of $1/L$ shifts:

$$\tilde{a}_k^{(l/L)} = \mathrm{IFFT}_N\left\{\left[\eta_r^{(1/L)}\right] \bullet \left[\mathrm{FFT}_N\left(\tilde{a}_k^{((l-1)/N)}\right)\right]\right\},\ l = 1,\ \ldots,\ L-1, \quad (5.21)$$

where $\mathrm{FFT}_N(\cdot)$ and $\mathrm{IFFT}_N(\cdot)$ are the direct and inverse N-point FFTs, respectively; $\bullet$ symbolizes the component-wise (Hadamard) matrix product of elements of two arrays; and $\{\eta_r^{(l/L)}\}$ is a set of coefficients defined for odd and even N by Eqs. (5.7) and (5.9), respectively, in which $\bar{\delta}_f$ are replaced by fractional shifts l/L.

The entire sub-sampled signal is obtained as a combination of the shifted copies in the corresponding order:

$$\tilde{\tilde{a}}_{kL+l} = \tilde{a}_k^{(l/L)};\ k = 0,\ \ldots,\ N-1,\ l = 0,\ \ldots,\ L-1. \quad (5.22)$$

The work of this algorithm is illustrated by a plot in Fig. 5.7 obtained using the program **Sincd_interpol_2D_SPIE.m** provided in the Exercises. For 2D signal/image sub-sampling (zooming in), this algorithm should be applied in two passes, say, row-wise and then column-wise, as illustrated in Fig. 5.8.

Mathematically, such a signal-zooming method is equivalent to the DFT spectrum zero-padding described earlier, but it is less computationally costly. Its computational complexity for a zooming factor L is $O(\log N)$ per output sample rather than $O(\log LN)$ for the zero-padding method.

Zoomed-in signals/images obtained in this way represent the most perfect "continuous" models that can be generated from the signal samples for a given sub-sampling rate. Such models can be used to perform image resampling over an arbitrary sampling grid. In this process, the required image samples (whose positions do not coincide with one of sampling nodes of the denser sampling lattice of the zoomed image) can be approximated by the nearest available sample of the zoomed image (Fig. 5.9(a)).

Provided appropriately chosen zoom-factor, nearest-neighbor interpolation in combination with discrete sinc interpolated zooming with a sufficiently large zoom factor will not compromise interpolation accuracy substantially.

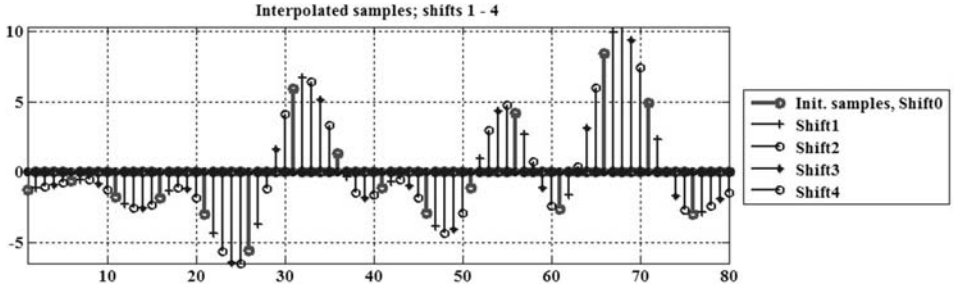


Figure 5.7 Initial signal samples (bold) and four sets of its sub-samples shifted by 1/5 (shift 1), 2/5 (shift 2), 3/5 (shift 3), and 4/5 (shift 4) of the sampling interval by the perfect shifting filter.

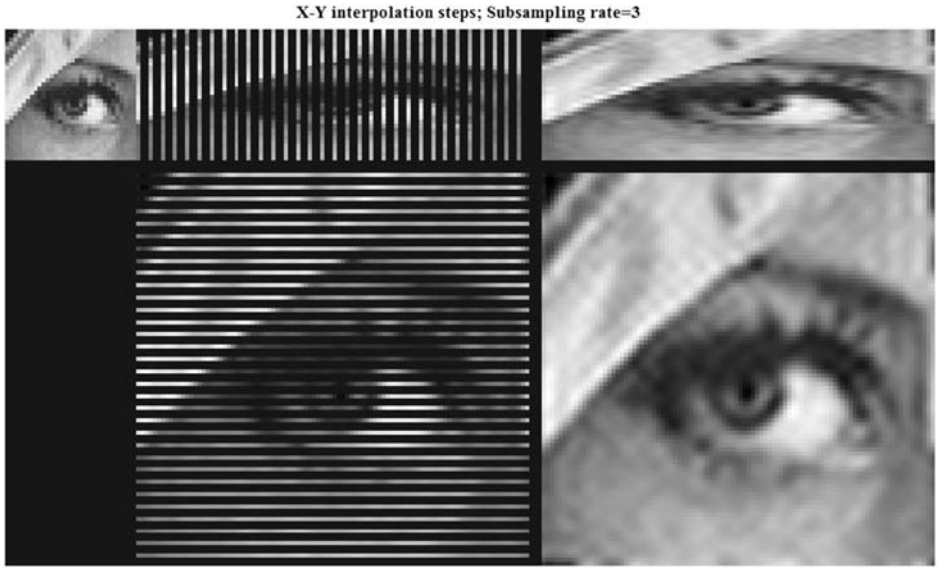


Figure 5.8 Row-wise (upper row) and column-wise (bottom row) image sub-sampling through fractional shifts.

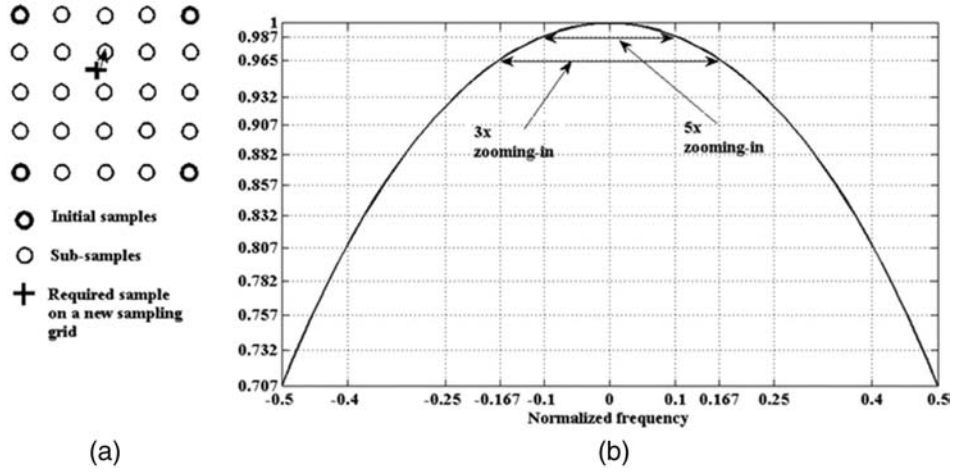


Figure 5.9 (a) Nearest-neighbor interpolation for resampling zoomed images and (b) its frequency response within the zoomed image baseband ($[-0.5 \div 0.5]$). Double arrows show the basebands of initial images before their $5\times$ and $3\times$ zoom.

For instance, as one can see from the plot of the frequency response of the nearest neighbor interpolator shown in Fig. 5.9(b) for a zoom-factor-5 decay of the frequency response on the highest spatial frequency of the initial image is only 1.3% and for a zoom factor of 3 it is 3.5%.

One of the important applications in which "continuous" image models are required is the fast location and tracking of moving targets in video

sequences. In this case, template images of the target with arbitrary orientation and scale can be very rapidly computed by means of corresponding resampling of the zoomed-in template image of the target obtained with a sufficiently large zoom factor.

Other application examples are image reconstruction from projections using the direct Fourier reconstruction algorithm and image reconstruction from fan-beam projections, which will be discussed in Section 6.2.

As a cyclic convolution, the "DFT-based" signal fractional shift algorithm suffers from the same boundary effects as the DFT zero-padding algorithm described earlier. An efficient practical solution computes the convolution in the DCT domain instead of in the DFT domain, as will be described in Section 10.4.

The DCT-based signal $\bar{\delta}_x$-shifting algorithm is defined by the equation

$$\tilde{a}_k^{(\bar{\delta}_x)} = \frac{1}{\sqrt{2N}} \left\{ \alpha_0^{(\mathrm{DCT})} \tilde{\eta}_0^{(\bar{\delta}_x)} + 2 \sum_{r=1}^{N-1} \alpha_r^{(\mathrm{DCT})} \mathrm{Re}\left(\tilde{\eta}_r^{(\bar{\delta}_x)}\right) \cos\left(\pi \frac{k+1/2}{N} r\right) - 2(-1)^k \sum_{r=1}^{N-1} \alpha_{N-r}^{(\mathrm{DCT})} \mathrm{Im}\left(\tilde{\eta}_{N-r}^{(\bar{\delta}_x)}\right) \cos\left(\pi \frac{k+1/2}{N} r\right) \right\}, \tag{5.23}$$

which follows from Eq. (10.63). $\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$ in Eq. (5.23) are, respectively, real and imaginary parts of the operand. The filter coefficients $\left\{\tilde{\eta}_r^{(\bar{\delta}_x)}\right\}$ in this equation are defined by Eqs. (10.55) and (10.58), in which one of the discrete sincd functions defined by Eqs. (5.14), (5.15), and (5.17) should be used as the convolution kernel $\{h_n\}$. Thus, the DCT-based algorithm is equivalent, in terms of the interpolation accuracy, to the above DFT-based versions of a perfect fractional shift algorithm.

When applied to L-times signal zoom, the algorithm must be applied $L-1$ times with shift $\bar{\delta}_x = l/L$, $l = 1, 2, \ldots, L-1$, similar to the DFT-based algorithm. With respect to boundary effects, the DCT-based fractional shift algorithm is as efficient as the DCT spectrum zero-padding algorithm. Figure 5.10 illustrates this with an example of sub-sampling a saw-tooth signal.

The figure shows that the heavy oscillations that propagate from signal borders in DFT-based discrete sinc interpolation almost completely disappear when the DCT-based version of the algorithm is used.

5.3.3 Quasi-continuous signal spectral and correlational analysis using the perfect fractional shift algorithm

One of the primary tasks of digital signal processing is signal spectral analysis, i.e., the estimation of signal Fourier spectra, detection of signals' periodic

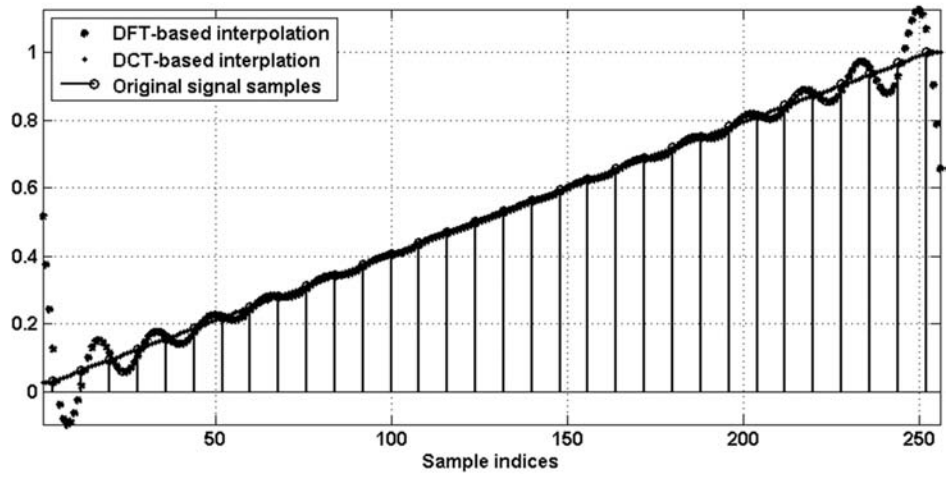


Figure 5.10 DFT-based vs. DCT-based 8× sub-sampling of a signal shown by dotted stems.

components, and measurement of their frequencies. The presence of periodic components appears in the peaks of the Fourier spectra of signals, the positions of which determine the frequencies of the components. As will be shown in Chapter 10, to numerically evaluate a signal's Fourier spectra from their samples, DFTs can be used as a discrete representation of the integral Fourier transform. Given a signal sampling interval Δ_x and a number of signal samples N, the frequencies of the signal DFT spectra are quantized with a spectrum sampling interval $\Delta_f = 1/N\Delta_x$ (Section 10.1, Eq. (10.10)). It is the basic resolving power of signal spectral analysis using DFTs. However, in applied signal spectral analysis, it is frequently required to measure frequencies of periodic signal components with sub-sampling interval accuracy, i.e., to perform numerical "quasi-continuous" Fourier spectral analysis. Above introduced perfect fractional shift filter applied to signal DFT spectra enables such a "quasi-continuous" Fourier spectral analysis.

Analogous to the signal fractional shift algorithm (Eq. (5.21)) for the "quasi-continuous" spectrum analysis of a signal $\{a_k\}$ with sub-sampling rate L, before computing the signal's fractionally shifted spectra $\left\{\alpha_r^{(l/L)}\right\}$, $l = 0, \ldots, L-1$, it should be modulated by the corresponding phase shifting coefficients $\lambda_k^{(l/N)}$:

$$\tilde{\alpha}_r^{(l/L)} = \mathrm{FFT}_N\left(a_k \lambda_k^{(l/L)}\right) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1`} a_k \lambda_k^{(l/L)} \exp\left(i2\pi \frac{kr}{N}\right). \tag{5.24}$$

For an odd number N of signal samples, the modulation coefficients $\left\{\lambda_k^{(l/N)}\right\}$ are defined as

$$\lambda_k^{(l/N)} = \begin{cases} \frac{1}{\sqrt{N}} \exp\left(i2\pi \frac{lk}{NL}\right), & k = 0, 1, \ldots, \frac{N-1}{2} \\ \left(\lambda_{N-k}^{(l/L)},\right)^* & k = \frac{N+1}{2}, \ldots, N-1, \end{cases} \tag{5.25}$$

where the asterisk $*$ denotes complex conjugacy.

For an even number N of signal samples, the modulation coefficients $\left\{\lambda_k^{(l/N)}\right\}$ are defined as

$$\begin{gathered} \lambda_k^{(l/N)} = \begin{cases} \exp(i2\pi l/LN), & k = 0, 1, \ldots, N/2 - 1 \\ C\cos(\pi l/N), & k = N/2 \end{cases} \\ \lambda_k^{(l/N)\,(\delta_f)}{}_k = \left(\lambda_k^{(l/N)}\right)^*, \quad k = N/2 + 1, \ldots, N-1, \end{gathered} \tag{5.26}$$

which can use the same three options as those for signal fractional shift (Eq. (5.10)):

- Case 0: $C = 0$;
- Case 1: $C = 1$; and
- Case 2, $C = 2$.

As shown in Appendix A2.12, the spectrum $\left\{\tilde{\alpha}_r^{(l/L)}\right\}$ of such a modulated signal for odd N is l/L-shifted over the frequency coordinate and discrete sinc interpolated spectrum of the non-modulated signal

$$\tilde{\alpha}_r^{(l/L)} = \sum_{s=0}^{N-1} \alpha_s \text{sincd}[N;\, \pi(r - s + l/L)]. \tag{5.27}$$

For even N, similar to the shifted discrete sinc interpolated signal (Eqs. (5.14), (5.15), (5.16), and (5.18)), the l/L-shifted discrete sinc interpolated spectrum for Case 0 is

$$\tilde{\alpha}_r^{(l/L)} = \sum_{s=0}^{N-1} \alpha_s \text{sincdd}[(N-1);\, N;\, \pi(r - s + l/L)]; \tag{5.28}$$

for Case 1,

$$\tilde{\alpha}_r^{(l/L)} = \sum_{s=0}^{N-1} \alpha_s \cos[\pi(n - \bar{\delta}_x)/N] \text{sincd}[\pi(n - s + l/L)]; \tag{5.29}$$

and for Case 2,

$$\tilde{\alpha}_r^{(l/L)} = \sum_{s=0}^{N-1} \alpha_s \text{sincdd}[(N+1);\, N;\, \pi(r-s+l/L)]; \tag{5.30}$$

where sincdd[M; N; x] is the sincdd function (Eq. (5.16)).

The L mutually shifted interpolated copies of the signal spectrum should be combined with the corresponding shifts to form a sub-sampled LN-point signal spectrum:

$$\tilde{\tilde{\alpha}}_{rL+l}^{(l/L)} = \tilde{\alpha}_r^{(l/L)};\ r = 0,\ \ldots,\ N-1,\ l = 0,\ \ldots,\ L-1. \tag{5.31}$$

This method of signal spectral analysis with sub-sample resolution is illustrated in Fig. 5.11, obtained by using the program **ContinuousSpectral Analysis_SPIE.m** provided in the Exercises.

Yet another important task in signal and image processing is computing signal/image correlations. For instance, for target location in images, cross-correlation between the image of the target object to be located and the image, where the target is searched for, is computed and the position of the highest cross-correlation peak is determined in the cross-correlation image and is taken as coordinates of the target.[20]

In signal correlational analysis, direct localization of correlation peaks in sampled signal is possible with the accuracy of the signal sampling interval Δ_x, although some applications require more accurate localization. Such a "quasi-continuous" correlation analysis can be implemented using the perfect fractional shift algorithm.

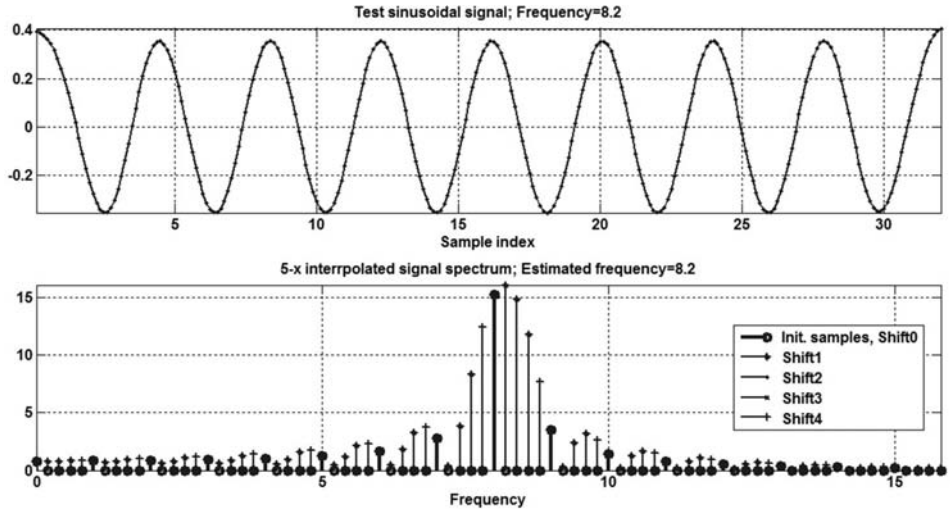


Figure 5.11 Spectral analysis with sub-sample resolution.

In order to enable locatization with sub-pixel accuracy cross-correlation, the cross-correlation image should be computed as sub-sampled with respect to correlated images. According to the convolution theorem (Appendix A2.5), the DFT spectrum of the cross-correlation between two images is a product of the DFT spectrum of the first image and the complex conjugate to the DFT spectrum of the second image. Therefore, the sub-sampled cross-correlation $\{c_k\}$ between two images $\{a_k\}$ and $\{b_k\}$, $k = 0, \ldots, N-1$, can, by direct analogy with signal sub-sampling (Eqs. (5.21) and (5.22)), be computed as

$$\tilde{\tilde{c}}_{kL+l} = \tilde{c}_k^{(l/L)};\ k = 0, \ldots, N-1,\ l = 0, \ldots, L-1, \tag{5.32}$$

where

$$\tilde{c}_k^{(l/N)} = \mathrm{IFFT}_N\left\{\left[\eta_r^{(l/N)}\right] \bullet [\mathrm{FFT}_N(a_k)] \bullet [\mathrm{FFT}_N(b_k)]^*\right\},\ l = 1, \ldots, L-1, \tag{5.33}$$

$\mathrm{FFT}_N(\cdot)$ and $\mathrm{IFFT}_N(\cdot)$ are N-point direct and inverse FFT operators, the dot $\bullet$ symbolizes an element-wise product of arrays, the asterisk $*$ symbolizes complex conjugacy, and the phase modulation coefficients $\left\{\eta_r^{(l/N)}\right\}$ are derermined for odd and even N by the corresponding Eqs. (5.7), (5.9), and (5.10).

5.3.4 Fast image rotation using the fractional shift algorithm

Rotation of a 2D coordinate system by an angle θ can be described in terms of the geometrical transformation of signal coordinates as the multiplication of signal coordinate vector (x, y) by a rotation matrix $\mathbf{ROT}_\theta$:

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = \mathbf{ROT}_\theta \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{5.34}$$

Given sampling intervals (Δ_x, Δ_y), the physical coordinates (x, y) of sampled signals are represented by the integer indices of pixels $\{k, l\}$:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} k\Delta_x \\ l\Delta_y \end{bmatrix}, \tag{5.35}$$

and the rotation matrix is applied to the vector of indices:

$$\begin{bmatrix} \tilde{k} \\ \tilde{l} \end{bmatrix} = \mathbf{ROT}_\theta \begin{bmatrix} k \\ l \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} k \\ l \end{bmatrix}. \tag{5.36}$$

Equation (5.36) describes the resampling rule that should be applied to the input image pixel indices $\{k, l\}$ to generate the indices $\{\tilde{k}, \tilde{l}\}$ of its rotated

copy. In order to reduce the computational complexity of this transformation, it is advisable to factorize the rotation matrix into a product of three matrices, each of which modifies only one coordinate:

$$\mathbf{ROT}_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} = \begin{bmatrix} 1 & -\tan(\theta/2) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \sin\theta & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan(\theta/2) \\ 0 & 1 \end{bmatrix}, \tag{5.37}$$

which implies that performing rotation in three passes will shift along one of the coordinates: along rows for the first pass, along columns for the second pass, and again along rows for the third pass. Specifically, when rotating an image of $N_x \times N_y$ pixels ($k = 0, 1, \ldots, N_x - 1$, $l = 0, 1, \ldots, N_y - 1$) around point ($0 \le k_0 \le N_x - 1$; $0 \le l_0 \le N_y - 1$), during the first pass the kth row is shifted by $\delta_x^{(k)}/\Delta_x = -\tan(\theta/2)(k - k_0)$, during the second pass the lth column is shifted by $\delta_y^{(l)}/\Delta_y = \sin\theta(l - l_0)$, and during the third pass the kth row is shifted by $\delta_x^{(k)}/\Delta_x = -\tan(\theta/2)(k - k_0)$. This implementation is known as the *three-pass rotation algorithm*, illustrated in Fig. 5.12.

The introduced DFT- or DCT-based signal fractional shift algorithms are ideally suited to perform these shifts. As far as these algorithms implement a cyclic convolution, image rotation by this method entails characteristic aliasing effects at image borders. They are illustrated in Figs. 5.12(b) and (d). They can be avoided by inscribing the image into an array of correspondingly larger size, as shown in Fig. 5.12(a), or by using only the aliasing-free image part inside the circle of the diameter, equal to the image linear size (see Fig. 5.12(d)).

5.3.5 Signal and image resampling using scaled and rotated DFTs

DFT- and DCT-based fractional shift algorithms enable perfect signal/image discrete sinc interpolated sub-sampling with rates defined by integer numbers. The scaled DFT introduced in Chapter 10 (Eqs. (10.16) and (10.17)) enables, in principle, signal/image discrete sinc interpolated resampling with arbitrary rational scale factors. This capability follows from the fact that, for signals $\{a_k\}$ with N samples, the DFT scaled with a scale factor σ is by definition (Eq. (10.17))

$$\alpha_r^{(\sigma)} = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{k=0}^{\lceil \sigma N \rceil - 1} a_k \exp\left(i2\pi \frac{kr}{\lceil \sigma N \rceil} \right), \tag{5.38}$$

equivalent to a $\lceil \sigma N \rceil$-point canonic DFT of this signal padded with $(\lceil \sigma N \rceil - N)$ zeros:

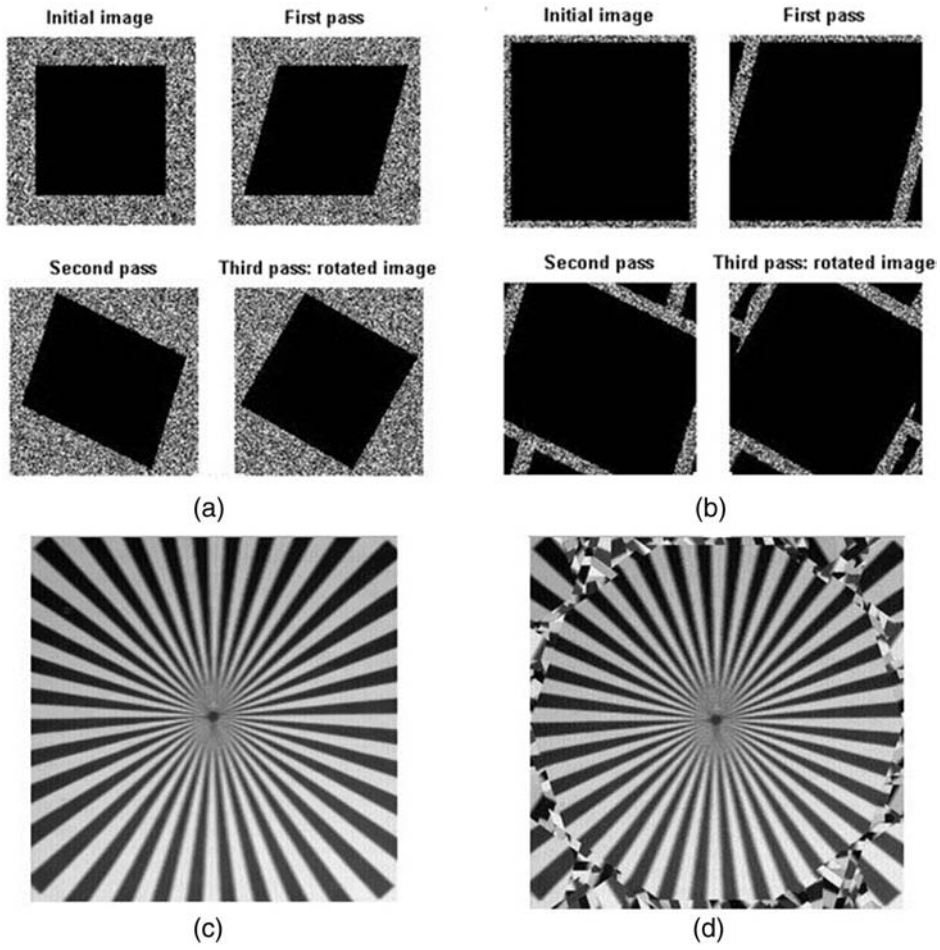


Figure 5.12 The principle of the three-pass image rotation algorithm and aliasing effects associated with its implementation through cyclic convolution: (a) rotation without aliasing; (b) rotation with aliasing effects owing to the cyclicity of the convolution; (c) initial image; and (d) image rotated 36° ten times, which shows aliasing effects outside the circle of the diameter equal to the image's linear size.

$$\alpha_r^{(\sigma)} = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{k=0}^{\sigma N-1} \tilde{a}_k \exp\left(i2\pi \frac{kr}{\lceil \sigma N \rceil} \right);$$
$$\tilde{a}_k = \begin{cases} a_k, & k = 0, 1, \ldots, N-1 \\ 0, & k = N, N+1, \ldots, \lceil \sigma N \rceil - 1. \end{cases} \tag{5.39}$$

$\lceil \sigma N \rceil$ in these equations denotes the integer closest to $\lceil \sigma N \rceil$.

This, in particular, means that the direct scaled DFT (Eq. (5.38)) perfoms spectral analysis with a sub-sampling factor $\lceil \sigma N \rceil / N$, and the signal scaling

that uses the scaled DFT (ScDFT) is nothing but signal sub-sampling by means of DFT spectrum zero-padding.

A non-trivial application involves the scaled DFT combined with image rotation using rotated scaled DFT for simultaneous image zooming and rotation. The algorithm of image resizing and rotation uses the following steps:

- For image $\{a_{m,n}\}$ of $N \times N$ pixels, compute its shifted DFT spectrum $\{\alpha_{r,s}\}$ with shift parameters $\{u_m;\ u_n;\ v_r = -(N-1)/2;\ v_s = -(N-1)/2\}$;
- Pad the obtained spectrum $\{\alpha_{r,s}\}$ with zeros in both coordinates to size $\lceil \sigma N \rceil \times \lceil \sigma N \rceil$, where σ is the desired zoom factor; and
- Apply the inverse rotated and scaled shifted DFT (ShDFT) (Eq. (10.32)) to the zero-padded spectrum with scale and rotation angle parameters σ and θ, and with shift parameters in the signal domain $\{u_k;\ u_l\}$ and in the spectral domain $\{v_r;\ v_s\}$ identical to those used to compute the signal-shifted DFT.

Appendix A2.13 (Eq. (A2.83)) will show that the resulting image $\tilde{a}_{k,l}$ is a $\lceil \sigma N \rceil / N$-times zoomed and discrete sinc interpolated copy of the initial image $\{a_{m,n}\}$ rotated by angle θ:

$$\begin{aligned}\tilde{a}_{k,l} = \frac{N^2}{\lceil \sigma N \rceil} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{m,n}\ \mathrm{sincd}\{N;\ \pi[\tilde{m} - N(\tilde{k}\cos\theta + \tilde{l}\sin\theta)/\lceil \sigma N \rceil]\} \\ \times\ \mathrm{sincd}\{\pi[\tilde{n} + N(\tilde{k}\sin\theta - \tilde{l}\cos\theta)/\lceil \sigma N \rceil]\}, \\ \tilde{m} = m + u_m,\ \tilde{n} = n + u_n;\ \tilde{k} = k + u_k,\ \tilde{l} = l + u_l.\end{aligned} \tag{5.40}$$

Although the inverse rotated and scaled ShDFT cannot be directly computed using FFT algorithms, it can be represented as a digital convolution (as will be shown in Appendix A2.14), and thus it can be efficiently computed using the FFT. To avoid the boundary effects associated with the cyclic convolution of the FFT, convolution in the DCT domain is recommended.

To conclude the discussion of image resampling algorithms through manipulations with the image DFT and DCT spectra: these algorithms are ideally suited to adaptive image restoration and enhancement by nonlinear modification of image spectra, such as soft and hard thresholding.[16] Figure 5.13 illustrates this option. It shows the result of simultaneous image rotation and scaling, as well as the result of image rotation, scaling, de-noising, and enhancement by hard thresholding the low-energy DCT spectral coefficients combined with the rising absolute values of the remaining (not zeroed by thresholding) image spectral coefficients to a power $P < 1$ (in this case, $P = 0.5$).

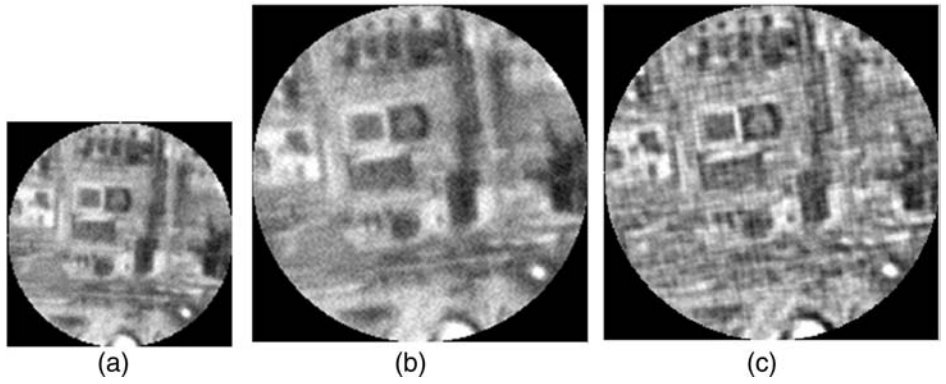


Figure 5.13 Simultaneous image resizing, rotation, denoising, sharpening and enhancement using the rotated scaled DFT (RotScDFT): (a) initial image; (b) 10°-rotated and 1.7×-magnified image; and (c) 10°-rotated, 1.7×-magnified, denoised, sharpened, and enhanced image.

5.4 Discrete Sinc Interpolation versus Other Interpolation Methods: Performance Comparison

This section provides experimental evidence of the superiority of the discrete sinc interpolation over other, more traditional numerical interpolation methods in terms of the interpolation accuracy and signal preservation. Compared with discrete sinc interpolation are the methods offered by the MATLAB image-processing toolbox: nearest-neighbor interpolation, linear (bilinear) interpolation, and cubic (bicubic) spline interpolation. Point spread functions and frequency responses of the compared interpolation methods are shown in Fig. 5.14.

Plots of the frequency responses of the methods (Fig. 5.14(c)) reveal a major drawback of the traditional interpolation methods: they tend to substantially attenuate high-frequency components within the signal baseband (frequency interval $[-0.5 \div 0.5]$) and pass substantial aliasing frequency components outside this interval. In imaging, this tendency results in image blurring that can worsen the visual image quality and its applicability to object recognition, target location, and such.

Figures 5.15–5.17 present the test images and results of comparison experiments on image rotation by multiple of 360° performed using the program **RotateComparis_demo.m** provided in the Exercises. In the experiments, test images, a piece of text (Fig. 5.15(a)), and a realization of pseudo-random image with a uniform Fourier spectrum within a square $[-0.35 < f_x < 0.35; -0.35 < f_y < 0.35]$ of the baseband (Fig. 5.15(b)) were rotated 1080° in 60 rotations of 18° using nearest-neighbor, bilinear, bicubic, and discrete sinc interpolation. For nearest-neighbor, bilinear, and spline interpolation methods, rotations were performed using the standard

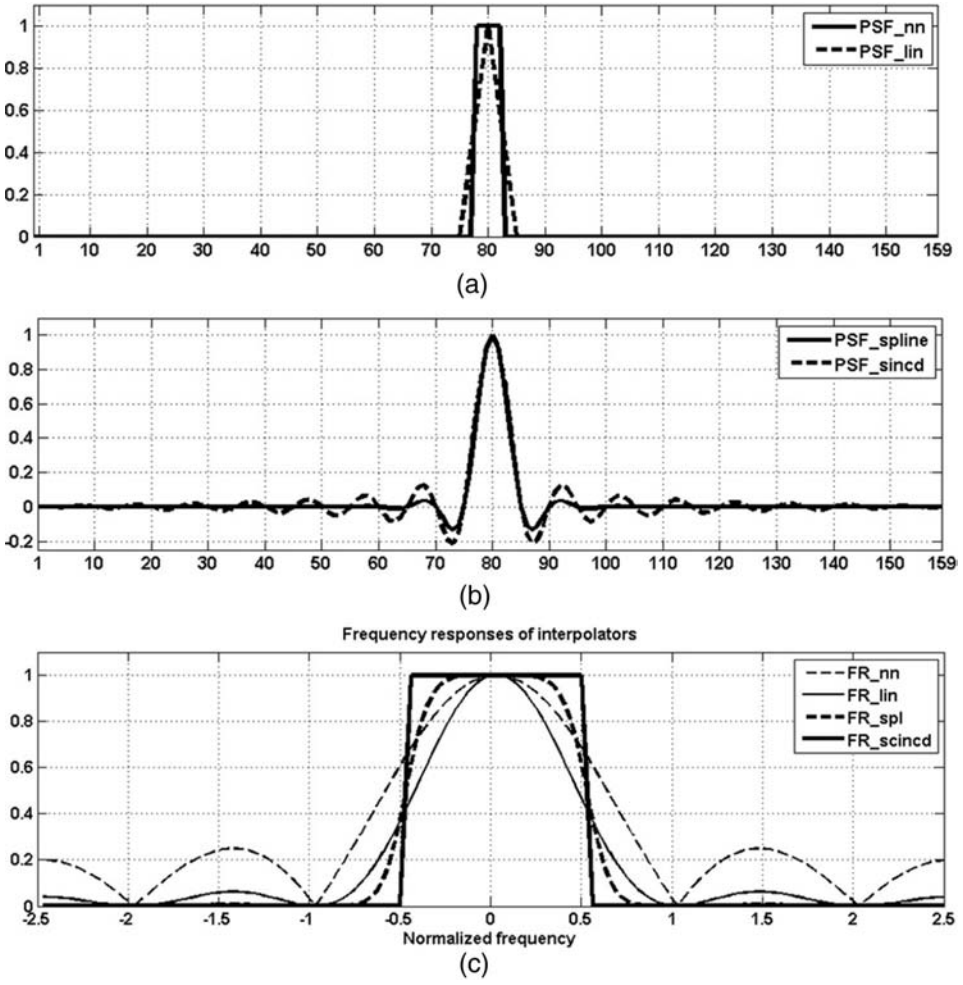
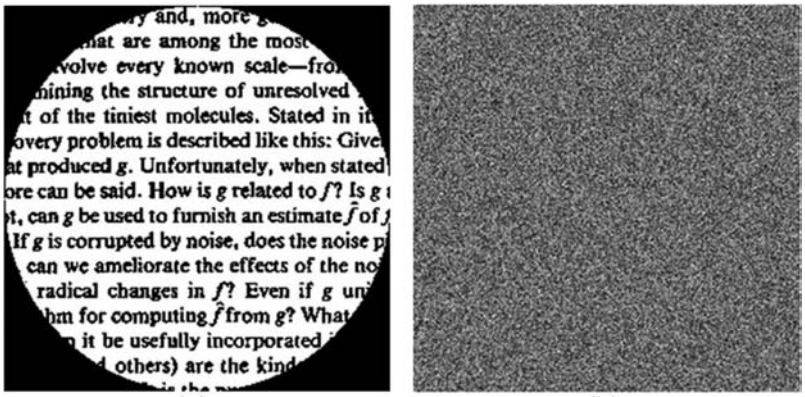


Figure 5.14 (a, b) Point spread functions and (c) frequency responses of nearest-neighbor (nn), linear (lin), bicubic spline (spl), and discrete sinc interpolators.

MATLAB program **imrotate.m** from the image-processing toolbox. For discrete sinc interpolated rotation, a program was used that implements the three-step rotation algorithm through the DFT-based fractional shift algorithm described in Section 5.3.4.

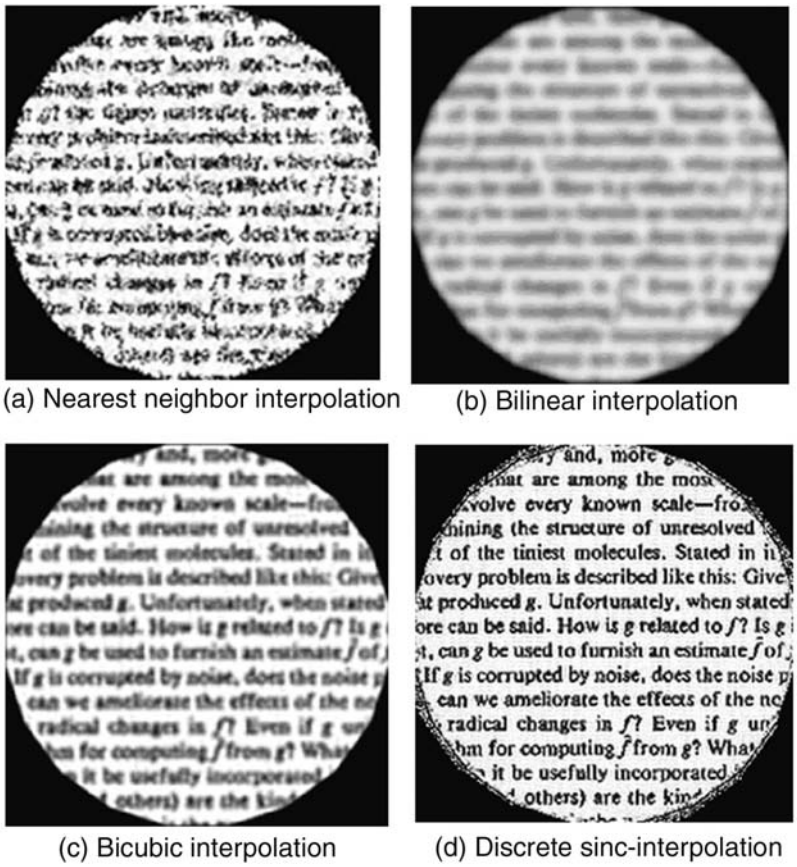
The rotation results shown in Fig. 5.16 for the "Text" image demonstrate (1) that the readability of the test image is completely destroyed after 60 steps of rotation due to blur when the standard nearest-neighbor, bilinear, and bicubic interpolation were used, and (2) that the test image is perfectly preserved after the same number of rotations when using discrete sinc interpolation.

In the analysis of interpolation errors, it is instructive to compare their power spectra to see which spectral components suffer more. For this purpose,



(a) (b)

Figure 5.15 Test images for comparison of interpolation methods: (a) "Text" image and (b) realization of a pseudo-random image with a uniform spectrum within 0.7 of the baseband ("Prus" image).



(a) Nearest neighbor interpolation (b) Bilinear interpolation

(c) Bicubic interpolation (d) Discrete sinc-interpolation

Figure 5.16 Discrete sinc interpolation vs. conventional numerical interpolation methods used for $60 \times 18°$ rotations of the test image "Text": (a) nearest-neighbor, (b) bilinear, (c) bicubic, and (d) discrete sinc interpolation.

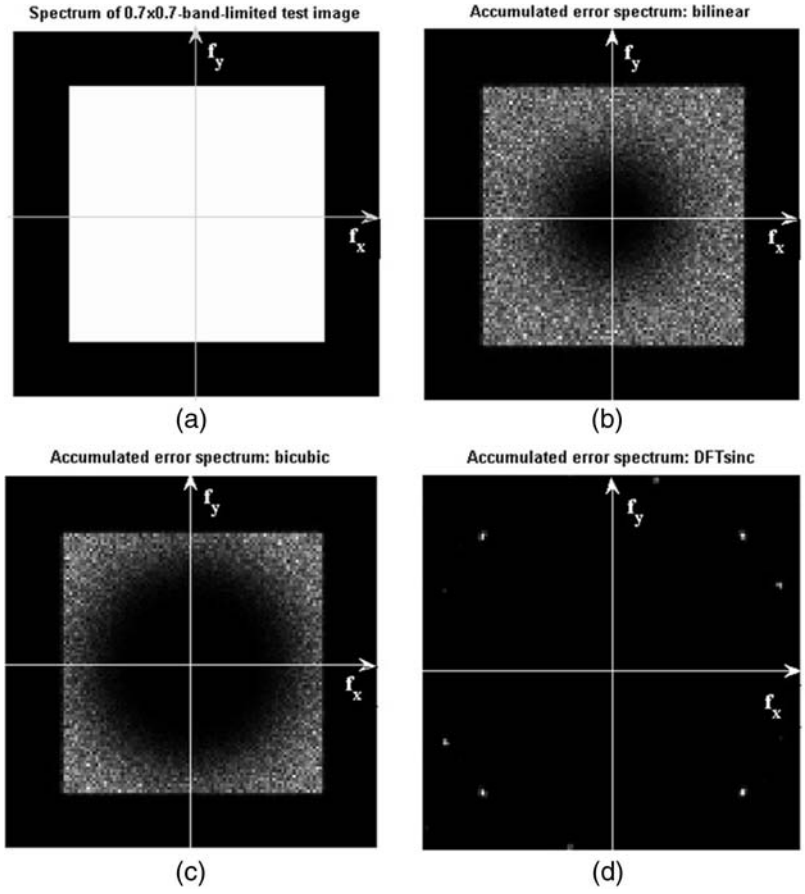


Figure 5.17 (a) Spectrum of test image "Prus" and spectra of rotation error after $60 \times 18°$ rotations of the image using (b) bilinear, (c) bicubic, and (d) discrete sinc interpolation. All spectra are shown in frequency coordinates (white arrows) centered at the spectrum zero frequency (dc component), with the image luminosity proportional to the error spectra intensity; the bright points in (d) are spectral aliasing components intentionally left at the borders of the test image baseband to secure, for display purposes, the same image dynamic range as in (b) and (c).

test image "Prus" is appropriate. Figure 5.17 presents the spectrum of the test image (Fig. 5.17(a)) and the spectra of rotation errors computed as the spectra of differences between the initial test image "Prus" and the results of its rotation by 1080° performed in 60 steps using bilinear, bicubic, and discrete sinc interpolations. The displayed error spectra were accumulated over 100 realizations of the test image "Prus." The figure shows that in the case of discrete sinc interpolation, the error spectrum is practically zero within the baseband circle, whereas for bicubic and bilinear interpolation, the error spectra intensity is low only for low spatial frequencies and grows to high frequencies.

To conclude this discussion of discrete sinc interpolation methods and their applications, note that the fast Fourier transform algorithm was invented about 200 years ago by Karl Friedrich Gauss to facilitate the numerical interpolation of sampled astronomical observation data by means of spectral zero-padding.[25]

5.5 Exercises

Sincd_interpol_2D_SPIE.m

Demonstration of 1D and 2D signal discrete sinc interpolated fast sub-sampling with signal fractional shifts. User-defined parameters are the type of test signals and the signal sub-sampling rate.

ContinuousSpectralAnalysis_SPIE.m

Demonstration of continuous spectrum analysis. Sinusoidal signals with non-integer frequencies specified by the user are used as test signals. Displayed are the test signal, interpolated signal DFT spectrum, and found estimate of the signal frequency.

RotateComparis_demo_SPIE.m

Comparison of three interpolation methods of MATLAB's routine imrotate.m (nearest-neighbor, bilinear, and bicubic) and the three-step image rotation algorithm with discrete sinc interpolation implemented in the DFT and DCT domains. Three options are offered:

- Single rotation of test image "Text,"
- Multiple rotations of test image "Text," and
- Multiple rotations of a pseudo-random test image by $10 \times 36°$ with the rotation error spectra accumulating over a number of test image realizations specified by the user.

Displayed are (a) images in the process of rotation, (b) rotation errors found as the difference between the input and multiple 360°-rotated images, and (c) the rotation-error DFT power spectra. In the titles of rotated images, the elapsed computation time T is also indicated.

Chapter 6
Discrete Sinc Interpolation in Other Applications and Implementations

6.1 Precise Numerical Differentiation and Integration of Sampled Signals

6.1.1 Perfect digital differentiator and integrator

Signal numerical differentiation and integration are operations that require measuring infinitesimal increments of signals and their arguments. Therefore, the numerical computation of signal derivatives and integrals assumes the building of "continuous" models of signals specified by their samples through explicit or implicit interpolation between available signal samples.

Because differentiation and integration are shift-invariant linear operations, methods of computing signal derivatives and integrals from their samples can be conveniently designed and compared in the Fourier transform domain. Let the Fourier transform spectrum of a continuous signal $a(x)$ be $\alpha(f)$:

$$a(x) = \int_{-\infty}^{\infty} \alpha(f) \exp(-i2\pi f x) df. \tag{6.1}$$

Then the Fourier spectrum of its derivative

$$\frac{d}{dx} a(x) = \int_{-\infty}^{\infty} [(-i2\pi f)\alpha(f)] \exp(-i2\pi f x) df \tag{6.2}$$

will be $(-i2\pi f)\alpha(f)$, and the Fourier spectrum of its integral

$$\bar{a}(x) = \int a(x) dx = \int_{-\infty}^{\infty} \left[\left(-\frac{1}{i2\pi f} \right) \alpha(f) \right] \exp(-i2\pi f x) df \tag{6.3}$$

will be $\alpha(f)/(-i2\pi f)$. Therefore, signal differentiation and integration can be regarded as signal linear filtering with filter frequency responses, respectively:

$$H^{(diff)}(f) = -i2\pi f \tag{6.4}$$

and

$$H^{(intg)}(f) = i/2\pi f. \tag{6.5}$$

Let signal $a(x)$ now be represented by its samples $\{a_k\},\ k = 0,\ 1, \ldots,\ N-1$, and let $\{\alpha_r\}$ be a set of DFT coefficients of the discrete signal $\{a_k\}$:

$$a_k = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \alpha_r \exp\left(-i2\pi \frac{kr}{N}\right). \tag{6.6}$$

Following the argumentation of Section 5.2 for the optimal resampling filter and using the relationship of Eq. (A2.39) in Appendix A2.8 that establishes mutual correspondence between the continuous signal frequency f and frequency index r of its DFT, one can conclude that samples $\left\{\eta_{r,opt}^{(diff)}\right\}$ and $\left\{\eta_{r,opt}^{(intg)}\right\}$ of the continuous frequency responses of perfect numerical differentiation and integration filters are defined for even N as

$$\eta_r^{(diff)} = \begin{cases} -i2\pi r/N, & r = 0,\ 1, \ldots,\ N/2-1 \\ -\pi/2, & r = N/2 \\ i2\pi(N-r)/N, & r = N/2+1, \ldots,\ N-1 \end{cases} \tag{6.7}$$

and

$$\eta_{r,opt}^{(intg)} = \begin{cases} 0, & r = 0 \\ iN/2\pi r, & r = 1, \ldots,\ N/2-1 \\ -\pi/2, & r = N/2 \\ iN/2\pi(N-r), & r = N/2+1, \ldots,\ N-1, \end{cases} \tag{6.8}$$

and for odd N as

$$\eta_r^{diff} = \begin{cases} -i2\pi r/N, & r = 0,\ 1, \ldots,\ (N-1)/2-1 \\ i2\pi(N-r)/N, & r = (N+1)/2, \ldots,\ N-1 \end{cases} \tag{6.9}$$

and

$$\eta_r^{(intg)} = \begin{cases} iN/2\pi r, & r = 0,\ 1, \ldots,\ (N-1)/2-1 \\ iN/2\pi(N-r), & r = (N+1)/2, \ldots,\ N-1. \end{cases} \tag{6.10}$$

Note that the coefficients $\eta_{N/2}^{(diff)}$ and $\eta_{N/2}^{(intg)}$ in Eqs. (6.7) and (6.8) are halved, which corresponds to the aforementioned Case 1 of discrete sinc interpolation (Eq. (5.9)).

Equations (6.7)–(6.10) suggest the following algorithmic implementation to compute the derivatives and integrals of signals specified by their samples:

$$\{\dot{a}_k\} = \mathrm{IFFT}\left(\left\{\eta_r^{(diff)}\right\} \bullet \mathrm{FFT}(\{a_k\})\right), \tag{6.11}$$

$$\{\bar{a}_k\} = \mathrm{IFFT}\left(\left\{\eta_r^{(intg)}\right\} \bullet \mathrm{FFT}(\{a_k\})\right), \tag{6.12}$$

where FFT(·)and IFFT(·) are direct and inverse FFTs, and • symbolizes the element-wise multiplication of arrays. Thanks to the FFT, the computational complexity of the algorithms is $O(\log N)$ operations per signal sample. The digital filter described by Eq. (6.11) is called the *discrete ramp filter*.

Like all DFT-based discrete sinc interpolation algorithms, DFT-based differentiation and integration algorithms (the most accurate in terms of preserving signal spectral components within the baseband) suffer from boundary effects. Especially vulnerable in this respect is DFT-based differentiation because of the potential discontinuities at signal borders due to their periodic replication when processing in the DFT domain. This drawback can be sufficiently alleviated by extending the signals to double their length with mirror reflection at their borders before applying the DFT-based algorithms described earlier. For such extended signals, DFT-based differentiation and integration are reduced to using the fast DCT instead of the FFT:

$$\{\dot{a}_k\} = -\frac{2\pi}{N\sqrt{2N}}(-1)^k \sum_{r=1}^{N-1}(N-r)\alpha_{N-r}^{(DCT)} \cos\left(\pi \frac{k+1/2}{N} r\right) \tag{6.13}$$

and

$$\{\bar{a}_k\} = \frac{\sqrt{N}}{2\pi\sqrt{2}}(-1)^k \sum_{r=1}^{N-1} \frac{\alpha_{N-r}^{(DCT)}}{N-r} \cos\left(\pi \frac{k+1/2}{N} r\right), \tag{6.14}$$

where $\left\{\alpha_r^{(DCT)}\right\}$ are the DCT transform coefficients of the signal. Naturally, the computational complexity of these algorithms is the same $O(\log N)$ operations per signal sample. These formulas can be obtained as special cases of the fast digital convolution algorithms described in Section 10.4 if one substitutes into Eq. (10.63) the frequency responses of differentiation and integration filters given by Eqs. (6.7)–(6.10). We will refer to the filters defined

by Eqs. (6.13) and (6.14) as the *DCT-based differentiation ramp filter* and *DCT-based integration filter*, respectively.

6.1.2 Conventional numerical differentiation and integration algorithms versus perfect DFT/DCT versions: performance comparison

In numerical mathematics, signal numerical differentiation and integration are commonly implemented through signal discrete convolution in the signal domain:

$$\dot{a}_k = \sum_{n=0}^{N_h-1} h_n^{(diff)} a_{k-n} \tag{6.15}$$

and

$$\bar{a}_k = \sum_{n=0}^{N_h-1} h_n^{int} a_{k-n}, \tag{6.16}$$

and the following differentiating kernels of two and five samples are recommended in manuals on numerical methods:[26]

$$h_n^{diff(1)} = [-0.5, \quad 0, \quad 0.5] \tag{6.17}$$

and

$$h_n^{diff(2)} = [-1/12,\ 8/12,\ 0\,,\ -8/12,\ 1/12]. \tag{6.18}$$

Both are based on an assumption that, given inter-sample distances, signals can be approximated by their Taylor series. We will refer to them as the *D1 and D2 differentiation methods*.

The best-known numerical integration methods are the *Newton–Cotes quadrature rules*.[26] The first three rules are the trapezoidal, Simpson, and 3/8 Simpson. In all of these methods, the value of the integral in the first point is not defined because it affects only the result's constant bias and can be chosen arbitrarily. When it equals zero, the trapezoidal, Simpson, and 3/8-Simpson numerical integration methods are defined respectively (with k as a running sample index) by the following equations:

$$\bar{a}_1^{(T)} = 0,\ \bar{a}_k^{(T)} = \bar{a}_{k-1}^{(T)} + \frac{1}{2}(a_{k-1} + a_k), \tag{6.19}$$

$$\bar{a}_1^{(S)} = 0, \quad \bar{a}_k^{(S)} = \bar{a}_{k-2}^{(S)} + \frac{1}{3}(a_{k-2} + 4a_{k-1} + a_k), \tag{6.20}$$

and

$$\bar{a}_0^{(3/8S)} = 0, \;\; \bar{a}_k^{(3/8S)} = \bar{a}_{k-3}^{(3/8S)} + \frac{3}{8}(a_{k-3} + 3a_{k-2} + 3a_{k-1} + a_k). \tag{6.21}$$

As will be discussed in Section 9.2, the continuous and overall frequency responses of digital filters are determined, given signal sampling and reconstruction devices, by their discrete frequency responses (DFT of their point spread functions). Applying an N-point DFT to Eqs. (6.17), (6.18), (6.19), (6.20), and (6.21) will produce the respective discrete frequency responses $\eta_r^{diff(1)}$, $\eta_r^{diff(2)}$, $\eta_r^{\text{int},T}$, $\eta_r^{\text{int }S}$, and $\eta_r^{(int,3/8S)}$ of the numerical differentiation and integration methods:

$$\eta_r^{diff(1)} \propto \sin(2\pi r/N);\; r = 0,\, 1, \ldots,\, N_\perp - 1; \tag{6.22}$$

$$\eta_r^{diff(2)} \propto \frac{8\sin(2\pi r/N) - \sin(4\pi r/N)}{12},\; r = 0,\, 1, \ldots,\, N_\perp - 1; \tag{6.23}$$

$$\eta_r^{(int,T)} = \frac{\bar{\alpha}_r^{(Tr)}}{\alpha_r} = \begin{cases} 0, & r = 0, \\ -\dfrac{\cos(\pi r/N)}{2i\sin(\pi r/N)}, & r = 1, \ldots,\, N_\perp - 1; \end{cases} \tag{6.24}$$

$$\eta_r^{(int,S)} = \frac{\bar{\alpha}_r^{(S)}}{\alpha_r} = \begin{cases} 0, & r = 0 \\ -\dfrac{\cos(2\pi r/N) + 2}{3i\sin(2\pi r/N)}, & r = 1, \ldots,\, N_\perp - 1; \end{cases} \tag{6.25}$$

and

$$\eta_r^{(int,3/8S)} = \frac{\bar{\alpha}_r^{(3S)}}{\alpha_r} = \begin{cases} 0, & r = 0 \\ -\dfrac{\cos(3\pi r/N) + 3\cos(\pi r/N)}{i\sin(3\pi r/N)}, & r = 1, \ldots,\, N_\perp - 1; \end{cases} \tag{6.26}$$

where $N_\perp$ is an index that corresponds to the highest signal frequency:

$$N_\perp = \begin{cases} (N-1)/2 \text{ for odd } N \\ N/2 \text{ for even } N. \end{cases} \tag{6.27}$$

These frequency responses and the frequency responses of DFT-based differentiation and integration filters are presented in Figs. 6.1 and 6.2, respectively.

One can see from these figures that standard numerical differentiation and integration methods entail certain and sometimes very substantial distortions of signal spectral contents at high frequencies. All of them attenuate high signal frequencies, and the Simpson and 3/8-Simpson integration methods—being slightly more accurate than the trapezoidal method in the middle of the

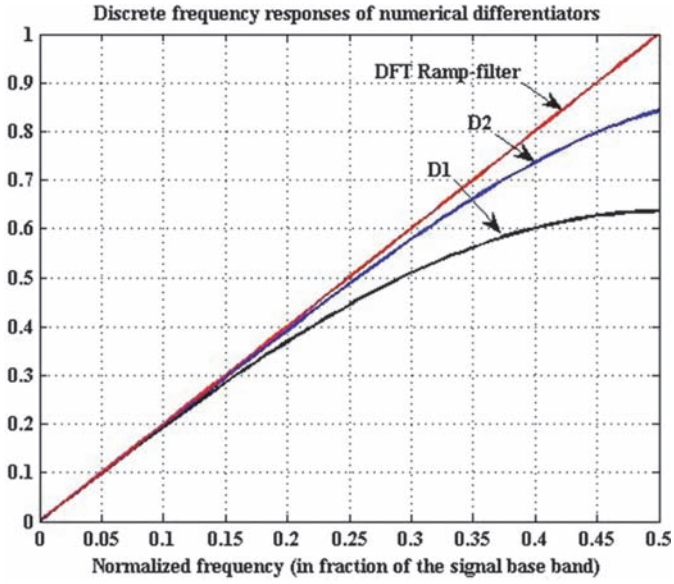


Figure 6.1 Absolute values of the frequency responses of differentiation filters described by Eq. (6.22) (curve D1), Eq. (6.23) (curve D2), and Eqs. (6.7) and (6.9) ("ramp" filter). The frequency coordinate is normalized by the highest signal frequency in the sampling baseband.

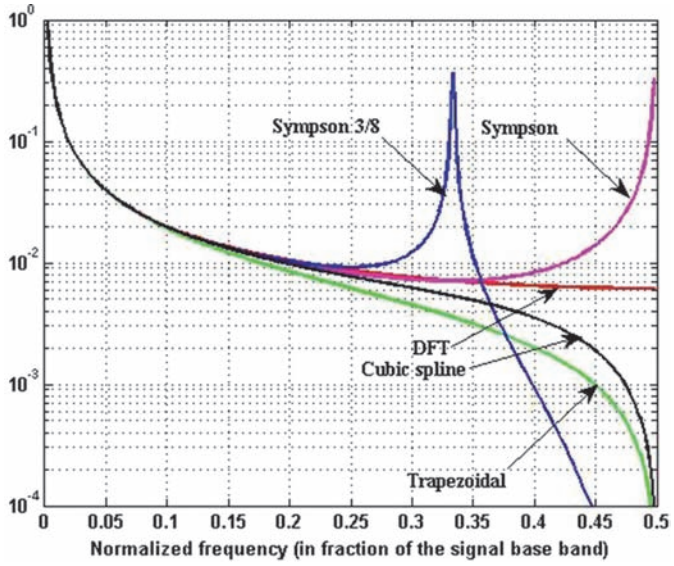


Figure 6.2 Absolute values of the frequency responses of numerical integration filters described by Eqs. (6.24), (6.25), and (6.26), as well as the DFT-based method (Eqs. (6.8) and (6.10)). The frequency coordinate is normalized by the highest signal frequency in the sampling baseband.

signal baseband—tend to generate substantial integration errors if the signals contain higher frequencies. The frequency response of the 3/8-Simpson rule tends to infinity for 2/3 of the maximum frequency, and the frequency response of the Simpson rule has almost the same tendency for the maximal frequency in the baseband. This means, in particular, that noise that might be present in input data and round-off computation errors will be overamplified by Simpson and 3/8 Simpson at these frequencies.

Figures 6.3 and 6.4 present the results of the experimental performance evaluation of the considered differentiation methods performed with the simulation program **differentiator_comparison_SPIE.m** provided in the Exercises. The program implements a statistical simulation of differentiation by using pseudo-random signals with uniform spectrum in bands in the range of 1/6 to 16/16 of the baseband.

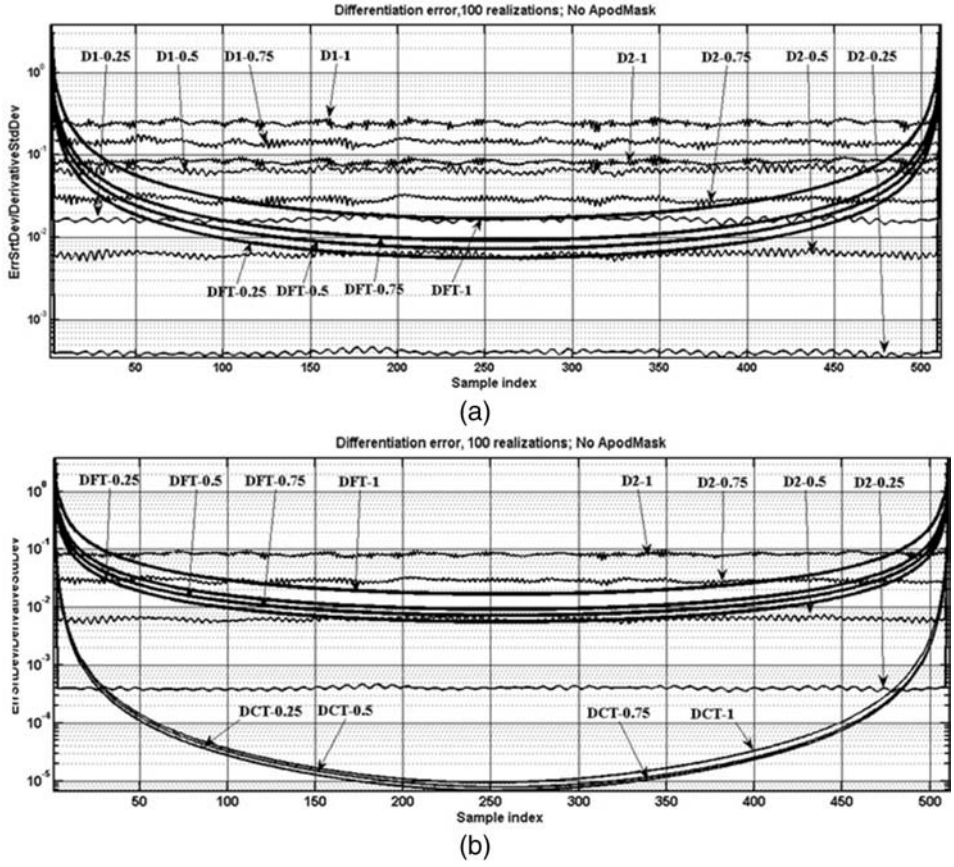


Figure 6.3 Experimental data about a signal's sample-wise normalized standard deviation of the differentiation error (a) for D1, D2, and DFT-based differentiation methods and (b) for D2-, DFT- and DCT-based methods. The numbers at the curves indicate the fraction (from one-quarter to one) of the test signal bandwidth with respect to the width of the signal baseband.

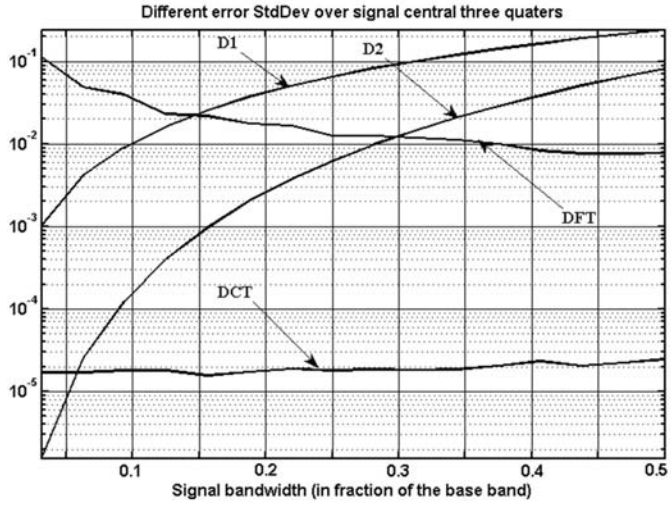


Figure 6.4 Normalized standard deviation of the differentiation error averaged over 100 samples in the middle of the test signal (samples from 200th to 300th) for D1, D2, DFT and DCT differentiation methods as a function of test signal bandwidth (in fractions of the width of the signal sampling baseband)

In the simulation, 16 series of statistical experiments with 100 experiments in each run were carried out. In the runs, realizations of pseudo-random signals of 32704 samples with a uniform Fourier spectrum were generated to imitate, by means of 32-fold oversampling, continuous signals. In each run, the generated pseudo-random signals were low-pass filtered to 1/32 of its baseband using the ideal low-pass filter implemented in the DFT domain. The filtered signal was then used as a model of a continuous signal, and its derivative was computed using DFT domain ramp-filter and used as an estimate of the signal ideal derivative. Then the central half of this signal that encompasses 16352 samples taken 8196 samples apart from the signal borders was sub-sampled at a rate of 32 to generate 511 signal samples that were used in the differentiation by the D1 method (Eq. (6.17)), D2 method (Eq. (6.18)), and the DFT-based (Eq. (6.11)) and DCT-based methods (Eq. (6.13)). The corresponding central part of the ideal derivative signal was also sub-sampled at a rate of 32 and was used as a reference to evaluate the differentiation error for the tested methods. The differentiation error was computed as the difference between the "ideal" derivative and the results of applying tested differentiation methods. It was divided by the standard deviation of the "ideal" derivative over all samples, thus producing estimates of the error mean square value normalized to the energy of the signal derivative. Finally, the standard deviation of the normalized error over 100 realizations was found for each signal sample. The obtained results are plotted sample-wise in Fig. 6.3(a) for the D1, D2, and DFT methods, and in Fig. 6.3(b) for D2, DFT, and DCT.

Figure 6.3(a) shows that the simplest method, D1, performs very poorly, whereas the D2 method outperforms the DFT method for signals with a bandwidth less than 0.5 of the baseband because of the boundary effects for the latter. The accuracy of the DFT differentiation method substantially improves with distance from signal borders. However, even for samples that are far from signal borders, boundary effects badly deteriorate the differentiation accuracy. The data presented in Fig. 6.3(b) show that the DCT-based differentiation method successfully overcomes the boundary effect problem and substantially outperforms both the D1 and D2 methods, even for narrowband signals.

Figure 6.4 presents plots of normalized error standard deviation computed on average over only the central 100 samples (from 200th to 300th) of test signals. These samples are sufficiently far from the signal borders, so the presented data are not practically influenced by boundary effects. The plots convincingly show that the D2 method secures better differentiation accuracy only for signals with a bandwidth less than 0.05 of the baseband, which corresponds to 20 times oversampling, and even for such signals normalized error standard deviation for DCT method is anyway less than 10^{-5}. For signals with broader bandwidth, the accuracy of the DCT differentiation method is better than of other methods by at least two orders of magnitude. One can also see that masking the signal with a window ("apodization") function (which gradually brings signal samples in the vicinity of its border to zero) substantially improves the differentiation accuracy of both DFT and DCT methods even further.

The above results imply that conventional numerical differentiation methods maintain a good differentiation accuracy if the signals are substantially oversampled, which undermines their only advantage of low computational complexity.

One more way to evaluate the accuracy of numerical differentiation and integration is the iterative application of successive differentiation and integration in tandem to a test signal and comparing the reconstructed signals with the initial signal. The plots in Fig. 6.5 obtained using the program **differentiat_integrat_error_SPIE.m** (provided in the Exercises) illustrate the results of a comparison of DCT-based differentiation and integration with the conventional D2 method of differentiation and trapezoidal method of integration. Plots in the left column show the original test rectangular impulse (solid line) and a result of the iterative application of differentiation and integration by DCT-based menthods (upper left plot, dots) and a D2 differention and trapezoidal integration method (bottom left plot, dots). Plots in the right column show the RMS of signal reconstruction errors for DCT-based differentiation and integration methods (upper right plot) and for D2 differentiation and trapezoidal integration methods (bottom fight plot). These results are clear evidence of the superiority of DCT-based methods over conventional methods of numerical differentiation and integration.

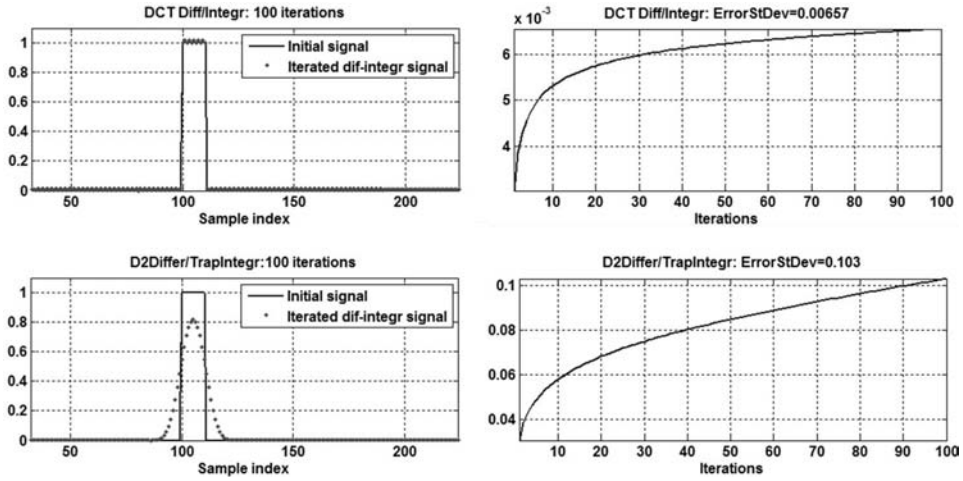


Figure 6.5 Comparison of signal reconstruction after iterative successive 100 differentiations and integrations applied in tandem to a rectangular test signal for DCT-based differentiation and integration methods (upper plots) and for a D2 differentiator and trapezoidal rule integrator (bottom plots).

6.2 Local ("Elastic") Image Resampling: Sliding-Window Discrete Sinc Interpolation Algorithms

Perfect DFT- and DCT-based fractional shift algorithms are computationally efficient at performing regular shifts of whole signals and images. For image resampling in arbitrary sampling lattices, they can be used to generate sufficiently highly oversampled "quasi-continuous" image models, as described in Section 5.3.2. However, this requires additional large memory buffers.

An alternative solution for image resampling in arbitrary sampling lattices is the implementation of discrete sinc interpolation in sliding-window processing. In sliding-window signal interpolation, the perfect shifting filter is applied only to pixels within the window, and only interpolated signal samples that correspond to the window central sample have to be computed at each window position from signal samples within the window. The interpolation function in this case is a discrete sincd-function, whose extent equals the window size rather than the whole image size required for the perfect discrete sinc interpolation. Therefore, sliding-window discrete sinc interpolation cannot provide as prefect interpolation, as the "global" (full size) discrete sinc interpolation. Figure 6.6 illustrates how well sliding-window discrete sinc interpolation approximates the global one. It shows 1D frequency responses of sliding-window discrete sinc interpolation for a window size of 15 pixels and that of global discrete sinc interpolations for 3× image zoom.

Sliding-window implementation of the discrete sinc interpolation can be regarded as a special case of signal-domain convolution interpolation methods. As follows from the above theory, it has the highest interpolation

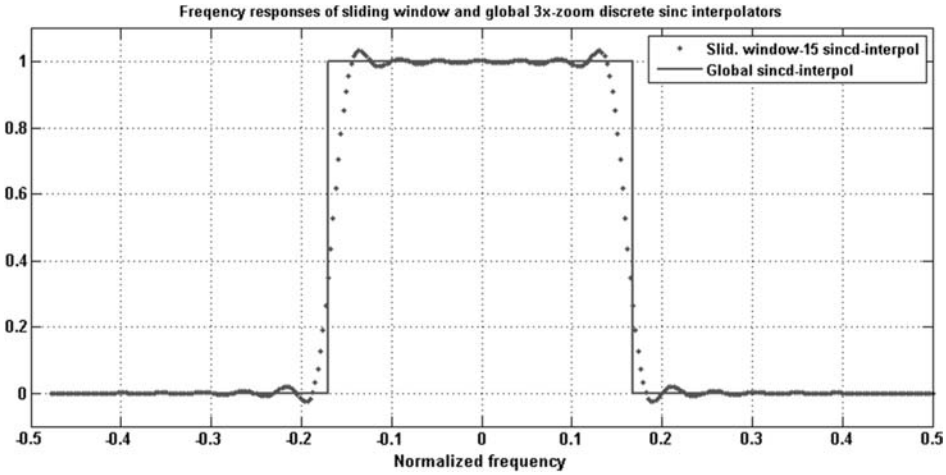


Figure 6.6 Frequency responses for a sliding window of 15 samples (dotted line) and of the perfect (global) discrete sinc interpolation (solid line) for 3× signal zoom.

accuracy, in principle, among all convolution interpolation methods with the same filter window size. Additionally, implementation in the DFT or DCT domain offers the option of image resampling with simultaneous restoration and enhancement by local adaptive filtering.[16,20]

In local adaptive filtering carried out in a sliding window, at each position of the transform coefficients of the window, samples are computed and then nonlinearly modified by means of hard or soft thresholding to obtain transform coefficients of the output signal samples in the window with a reduced level of noise. These coefficients are then used to generate an estimate of the window central pixel by inverse transform computed for the window central pixel. Figure 6.7 illustrates the application of combined filtering and interpolation for irregular-to-regular image resampling with denoising.

Figure 6.7 Image resampling and denoising in a sliding window: (a) noisy and irregularly sampled image, and (b) elasic resampled (rectified) and denoised image.

In this example, the left image is distorted by known displacements of pixels with respect to regular equidistant positions and by additive noise. In the right image, these displacements are compensated and noise is substantially reduced with a sliding-window resampling and denoising algorithm.

6.3 Image Data Resampling for Image Reconstruction from Projections

6.3.1 Discrete Radon transform and filtered back-projection method for image reconstruction

Precise data resampling is a crucial issue in image reconstruction from projections, which is based on the properties of the integral Radon transform (RT). The main problem in the discrete representation of the integral RT is the definition (for computing image projections) of the line integral under an arbitrary angle over an image-sampling lattice.

Any definition of the discrete line integral should assume some method of image interpolation to find image values along the projection line in points that do not coincide with available image samples. For regular rectangular sampling grids, only column-wise, row-wise, and 45°-diagonal-wise integrations do not require any interpolation. One possible solution to this problem is line integration over a "continuous" image model, obtained by means of the above described method of image sub-sampling with discrete sinc interpolation. An alternative and more computationally efficient solution is the following algorithmic implemention of the discrete Radon transform (DRT):

$$Pr(\theta_r, s) = \mathrm{SUM}_l[\mathrm{ROT}_{\theta_r}(\{a_{k,l}\})], \tag{6.28}$$

where $Pr(\theta_r, s)$ are samples of the rth projection, taken under angle θ_r, of the image defined by its samples $\{a_{k,l}\}$ over a square sampling lattice $\{k, l\}$; $\mathrm{ROT}_{\theta_r}(\bullet)$ is an operator of image rotation by the angle θ_r around the center of the sampling lattice; and $\mathrm{SUM}_l[\bullet]$ is the summation operator of the samples of the rotated image over index l.

According to this algorithm, the required image interpolation is carried out in the process of image rotation. In order to secure the smallest possible interpolation error, image rotation should be performed with discrete sinc interpolation. We will assume that for the implementation of the rotation operator, the above described fast three-step rotation algorithm is used, which preserves the number of image samples (although the previously described image rotation algorithm with scaled coordinates can, in principle, be used as well).

The most frequently used algorithm of image reconstruction from projections is the *filtered back-projection algorithm*.[16] This algorithm assumes the accumulation of derivatives of projections projected backward (repeated)

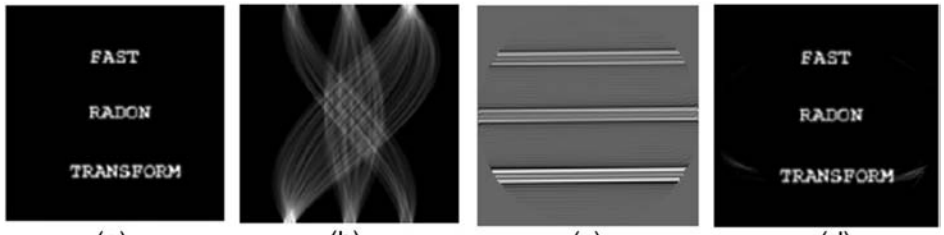


Figure 6.8 (a) Test image, (b) set of its projections with projection angle as the vertical coordinate, (c) an example of a filtered projection projected backward along the horizontal projected line, and (d) reconstructed image.

in the directions that they were obtained. According to the above definition of the DRT, this image reconstruction algorithm can be implemented as

$$a_{k,l} = \mathrm{SUM}_r\{\mathrm{ROT}_{-\theta_r}\{\mathrm{BckP}_l\{\mathrm{RAMPF}\{Pr(\theta_r, k)\}\}\}\}, \tag{6.29}$$

where $\{Pr(\theta_r, k)\}$ are samples of the image's rth projection, taken under angle θ_r; RAMPF$\{\bullet\}$ is an operator of ramp filtering for differentiation described in Section 6.1; BckP$_l\{\bullet\}$ is a back-projection operator to replicate the operand over index r; and SUM$_r\{\bullet\}$ is a summation operator that sums up the replicated (projected backward) ramp-filtered projections over the entire set of projection angles θ_r.

Figure 6.8 is generated using the program **radon_invradon_demo_SPIE.m** provided in the Exercises to illustrate the DRT and image reconstruction using the filtered back-projection algorithm.

6.3.2 Direct Fourier method of image reconstruction

According to the projection theorem for the RT, the Fourier spectra of image projections are cross-sections of the image 2D spectrum under corresponding angles. Therefore, if one computes the spectra of projections of an image and appropriately arranges them in a polar coordinate system in the Fourier domain to form a 2D image spectrum, one can reconstruct the image by the inverse Fourier transform of this spectrum. This is called the *direct Fourier method of image reconstruction from projections*. The problem is in the implementation of the 2D inverse Fourier transform in polar coordinates.

The polar coordinate system of spectral samples in a Cartesian coordinate system is shown in Fig. 6.9. Spectral samples are non-uniformly spaced in Cartesian coordinates and are very sparse, especially high-frequency ones. Available 2D inverse FFT algorithms assume a uniform sampling lattice in Cartesian coordinates. Therefore, for spectral samples in polar coordinates, one has to either resample the spectrum from polar to Cartesian coordinates or, to reconstruct spectral samples and images on a uniform dense lattice, apply the iterative algorithms for image recovery from sparse non-uniform

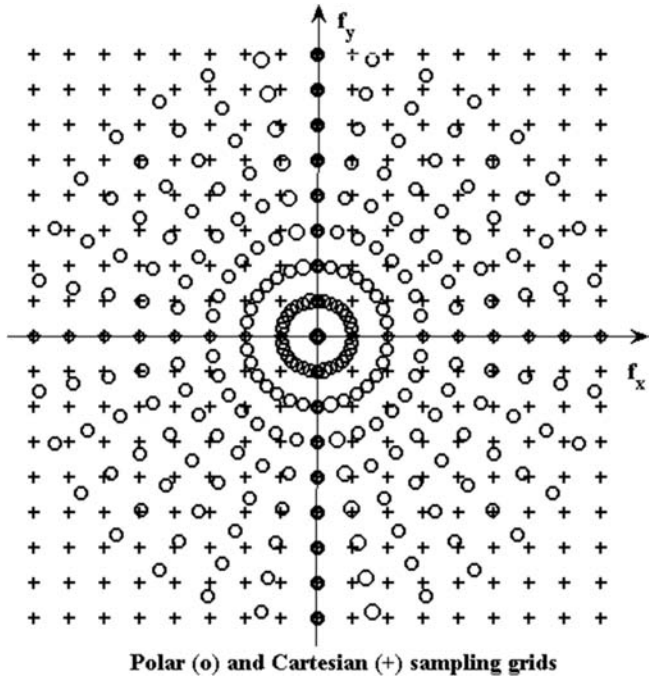


Figure 6.9 Spectral samples (small circles) in a Cartesian sampling lattice (crosses).

samples considered in Chapter 4. However, the latter option does not seem feasible. High-frequency spectral samples are sparse, whereas at low frequencies the spectra are very oversampled, and this is not compensated by the image natural redundancy associated with empty areas in the image domain and image compressibility.

A feasible practical option for solving the problem image reconstruction from its spectrum in polar coordinate system is polar-to-Cartesian coordinate conversion of spectra of projections to 2D image spectrum by means of resampling, in Cartesian coordinates, of a "continuous" model of the projection spectra in a polar coordinate system formed through a separable (radial-frequency- and angle-index-wise) sub-sampling projection spectra using discrete sinc interpolation. Figure 6.10 presents an illustrative example of image reconstruction from projections achieved through the inverse DFT of the image 2D spectrum obtained by resampling the image spectrum in the polar coordinate system.

6.3.3 Image reconstruction from fan-beam projections

The methods of image reconstruction from projections described earlier assume image projection in parallel x-ray beams. This is the original classic image tomographic projection method, for which well-developed reconstruction

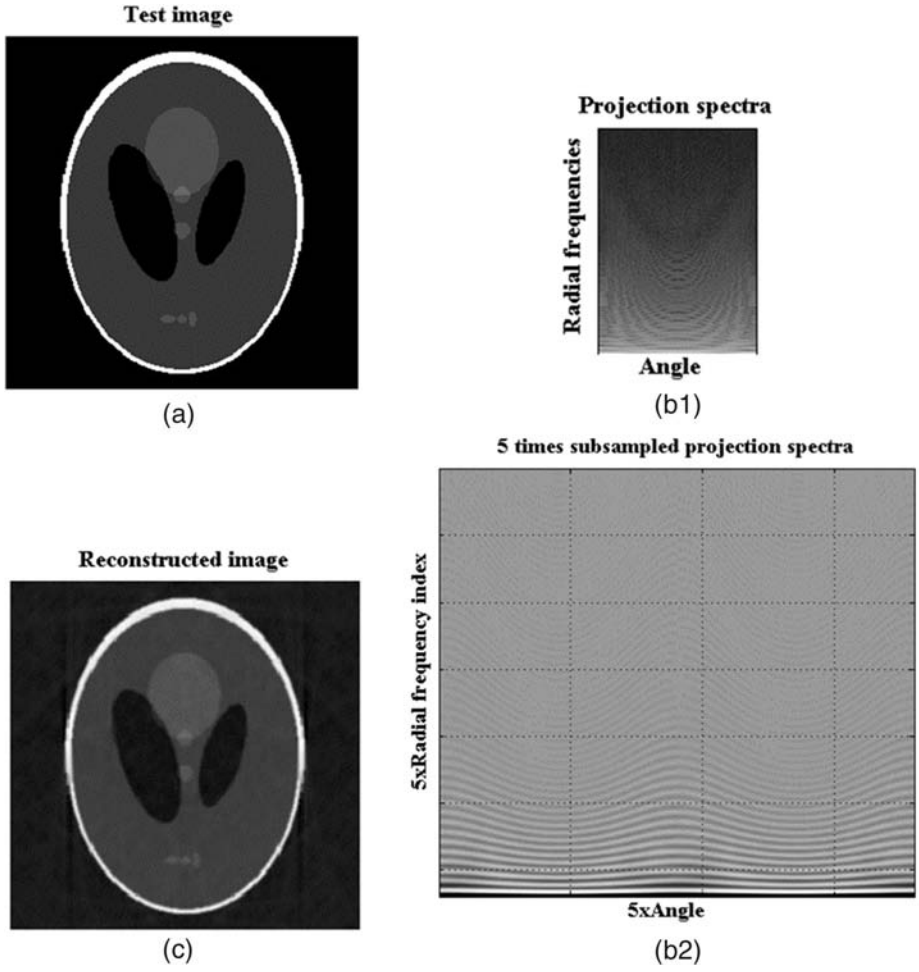


Figure 6.10 An illustrative example of image reconstruction from projections by the DFT method through polar-to Cartesian resampling of zoomed-in spectra of projection: (a) test image; (b1) spectra of projection; (b2) projection spectra 5 times sub-sampled (zoomed-in) in both coordinates for polar-to-Cartesian coordinate conversion (only half of the spectral coefficients that correspond to frequencies from zero to the highest one in the baseband are displayed; the others that are complex conjugate to them are not shown); and (c) image reconstructed by means of the inverse DFT applied to the 2D spectrum obtained by resampling the zoomed-in projection spectra.

algorithms are available. In practice, commercial CT scanners use fan-beam projections rather than parallel-beam because fan-beam projections can be obtained with a point source of x rays, which is much easier for fabrication than collimated parallel-beam sources.

The geometry of fan-beam projection is sketched in Fig. 6.11. The point source of radiation makes a full 360° revolution around the object, and the absorption integrals of radiation by the object over projection lines form

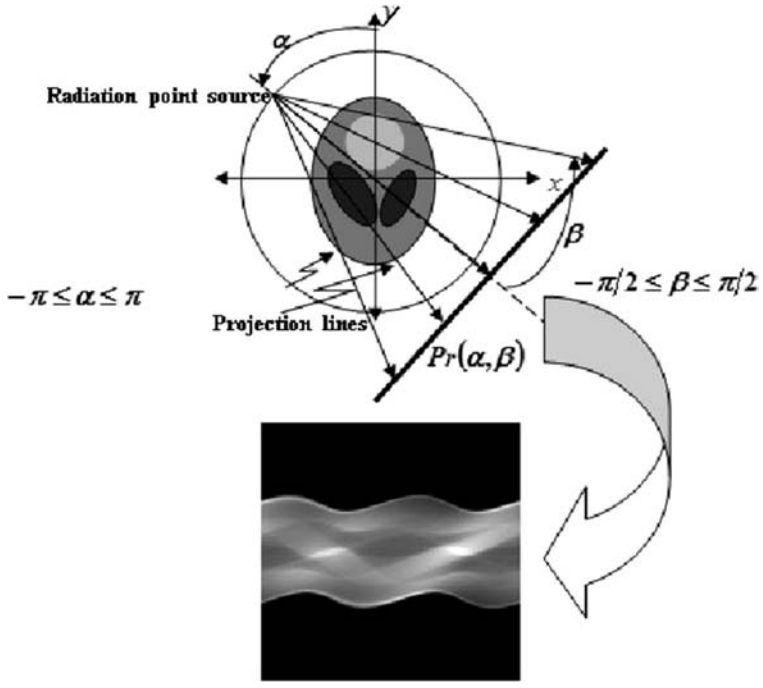


Figure 6.11 Geometry of image fan projections.

(for each position angle α of the point source) projections $Pr(\alpha, \beta)$ as a function of the ray angles β. A set of projections for $-\pi \le \alpha \le \pi$ is then used for image reconstruction.

In principle, inverting the RT in fan-beam projection geometry requires reconstruction algorithms that work in fan-beam projection geometry. There is however an alternative and attractive option of converting, by an appropriate resampling, the set of fan projections into a set of parallel projections and to enable in this way image reconstruction with algorithms for image reconstruction from parallel projections. For the resampling, one can use the above described algorithms for resampling by image sub-sampling using global or local discrete sinc interpolation. The latter can, if required, be combined with denoising, as discussed in Section 6.2. This process of converting one type of projection into another type is called *data rebinning*. Figure 6.12 illustrates this method of image reconstruction.

6.4 Exercises

differentiator_comparison_SPIE.m

Comparison of signal differentiation accuracy of two conventional differentiators with point spread functions

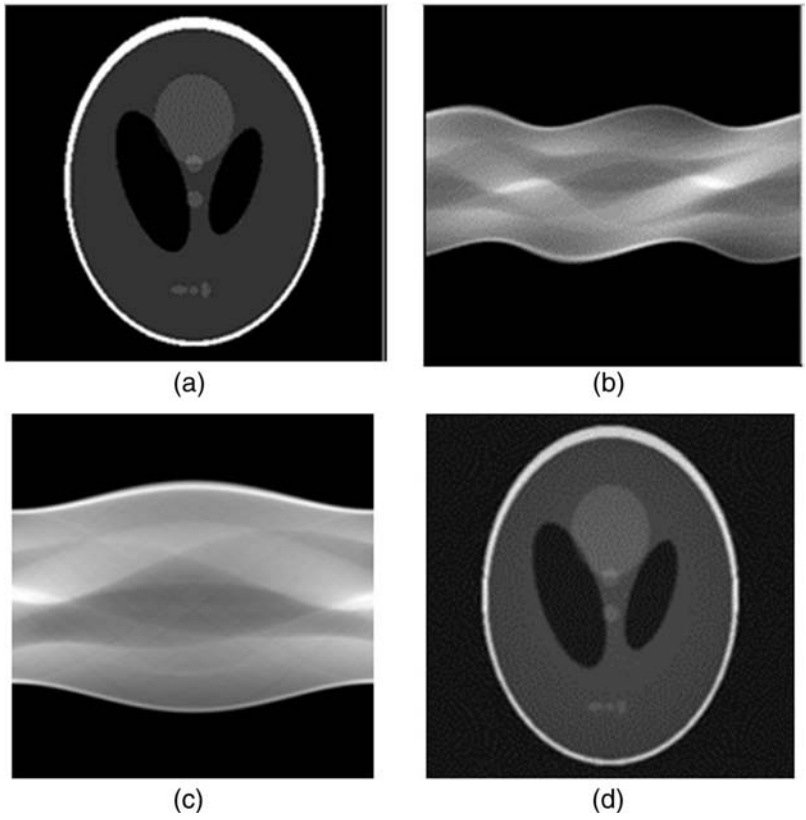


Figure 6.12 Image reconstruction from fan-beam projections: (a) initial test image; (b) its fan-beam projections; (c) parallel projections converted from the fan-beam projections by means of appropriate resampling them; and (d) image reconstructed from converted parallel projections.

- PSFD1 = [0.5 0.0 –0.5],
- PSFD2 = [–1/12 2/3 0 –2/3 1/12],

and DFT- and DCT-based differentiators using, as test signals, realizations of pseudo-random signals with 16 different bandwidths from 1/16 to 16/16 of the signal baseband width. User-defined parameters:

- The number N of samples of the signal and its derivative,
- Oversampling rate M,
- The number N_{test} of test realizations of pseudo-random signals.

diffrentiat_integrat_error_SPIE.m

Comparison of DCT-based differentiation and integration and D2&Simpson differentiation and integration by means of repeated differentiation and

integration operations in tandem of a test rectangular impulse. User-defined parameter:

- The number of iterations N_{it}.

radon_invradon_demo_SPIE.m

Demo of the direct Radon transform implemented through projecting the rotated image and image reconstruction using an inverse Radon ramp-filtered back-projection algorithm.

Chapter 7
The Discrete Uncertainty Principle, Sinc-lets, and Other Peculiar Properties of Sampled Signals

7.1 The Discrete Uncertainty Principle

It is well known that continuous signals cannot both be sharply band-limited, in terms of their Fourier spectra, and have sharply bounded support. In reality, continuous signals are neither sharply band-limited nor have sharply bounded support. They can only be more or less densely concentrated in the signal and spectral domains. This property is mathematically formulated in the form of the famous "*uncertainty principle*":

$$X \times F > O(1), \tag{7.1}$$

where X is the interval in the signal domain, within which a signal is concentrated; F is the interval in the signal Fourier spectral domain, within which the signal spectrum is concentrated; and $O(1)$ is a constant on the order of one, which depends on the methods of defining the degree of signal and its spectrum concentration within those intervals. This principle is a fundamental law of nature discovered by W. Heisenberg in quantum mechanics[27] and extended to signal theory by D. Gabor.[2]

How does this property translate to sampled continuous signals and their DFT and DCT spectra, which originate from the integral Fourier transform? Experiments show that discrete sampled signals specified by a finite number N of samples that represent continuous signals can, in contrast with continuous signals, be sharply bounded in both the signal and spectral domain, i.e., they have a non-zero certain number $N_{sign} \leq N$ of their samples and certain number $N_{spectr} \leq N$ of samples of their DFT or DCT spectra. Such space-frequency sharply bounded signals can be generated using a simple iterative

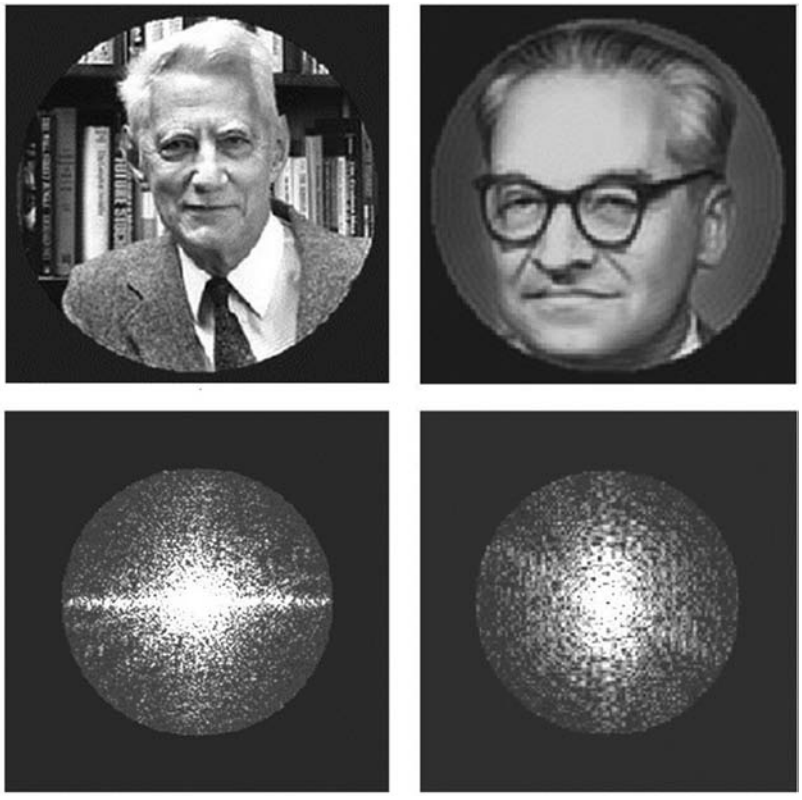


Figure 7.1 Portraits of C. Shannon (left column) and V. Kotelnikov (right column), and their corresponding DFT spectra bounded by circular shapes (shown centered at their DC component).

algorithm, which at each iteration applies the requested bounds alternatively in the signal and spectral domain. Two examples of such space-frequency sharply bounded images are shown in Fig. 7.1. They were generated using the program **SpaceLimSpectrlim_SPIE.m**, provided in the Exercises.

The relationship between bounds in signal and DFT domains is defined by the *discrete uncertainty principle*, which can be derived from the continuous version. Let Δ_x and Δ_f be the signal and its Fourier spectrum sampling intervals, correspondingly. Then the size of the interval occupied by the continuous signal that corresponds to the given sampled signal with N_{sign} non-zero samples can be estimated as $X \approx N_{sign}\Delta_x$, and the size of the interval in the signal Fourier domain occupied by the signal spectrum N_{spectr} non-zero samples can be estimated as $F \approx N_{spectr}\Delta_f$. From the uncertainty principle for continuous signals (Eq. (7.1)), it follows that

$$N_{sign}\Delta_x N_{spectr}\Delta_f > O(1), \tag{7.2}$$

or

$$N_{sign}N_{spectr} > \frac{O(1)}{\Delta_x \Delta_f}. \tag{7.3}$$

According to the cardinal sampling relationship, for which the DFT represents the integral Fourier transform (Eq. (10.13)),

$$\frac{1}{\Delta x \Delta f} = N, \tag{7.4}$$

so the numbers N_{sign} and N_{spectr} of the signals and their spectral non-zero samples are linked by the relationship

$$N_{sign}N_{spectr}/N \geq O(1). \tag{7.5}$$

This relationship can also be expressed in terms of densities $D_{sign} = N_{sign}/N$ and $D_{spectr} = N_{spectr}/N$ of the signal and their spectral non-zero samples:

$$D_{sign}D_{spectr}N \geq O(1). \tag{7.6}$$

Inequalities (7.5) and (7.6) formulate the discrete uncertainty principle.

Figure 7.2 illustrates the results of an experimental evaluation of the constant $O(1)$ in these inequalities. In the experiment carried out using the program **UncertaintyPrinciple_SPIE.m**, provided in the Exercises, the above-mentioned iterative algorithm was used to generate (from a delta function as a seed signal) signals of a given fixed width and DFT spectra of different widths. The experiment evaluated the maximal amplitude of signals for different values of the product $D_{sign}D_{spectr}N$ varied from 0.5 to 10. Figure 7.2 shows that space-limited and band-limited signals for which $D_{sign}D_{spectr}N < 2$ do not exist.

7.2 Sinc-lets: Sharply-Band-Limited Basis Functions with Sharply Limited Support

The existence of sharply-space-frequency-bounded signals allows one to hypothesize the existence of correspondingly sharply-space-frequency-bounded basis functions that can be used to represent such signals. One can generate them by means of the above-mentioned iterative procedure for generating sharply-space-frequency-bounded signals using delta functions with different locations within the chosen signal support interval in the signal domain as seed signals:

$$\begin{aligned}
&[\text{sinclet}^{(0)}(S_{\text{lim}};\, B_{\text{lim}};\, k_0;\, k)] = \delta(k - k_0),\\
&[\text{sinclet}^{(t)}(S_{\text{lim}};\, B_{\text{lim}};\, k_0;\, k)] =\\
&S_{\text{lim}}(IDFT\{B_{\text{lim}}(DFT[\text{sinclet}^{(t-1)}(S_{\text{lim}};\, B_{\text{lim}};\, k_0;\, k)])\}),\ t = 1,\, 2,\, \ldots,
\end{aligned} \tag{7.7}$$

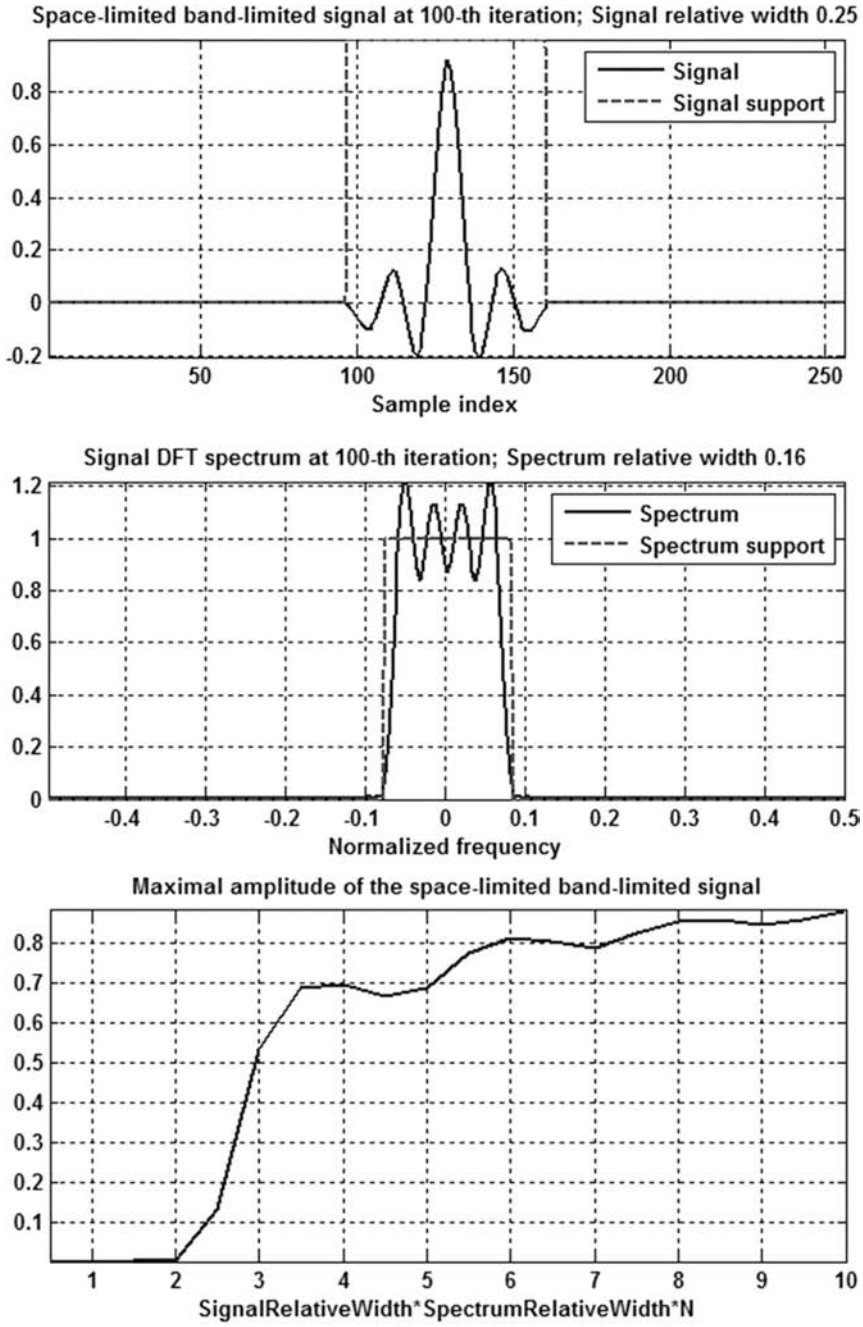


Figure 7.2 Experimental evaluation of the uncertainty relationship between the length of the signal and the width of its DFT spectrum for sharply-space-limited and sharply-band-limited signals. From top to bottom: a realization of the space-limited and band-limited signal with $D_{sign}D_{spectr}N = 10$, its DFT spectrum, and a plot of maximal amplitudes of signals vs. the value of the product $D_{sign}D_{spectr}N$.

where $DFT\{\bullet\}$ and $IDFT\{\bullet\}$ are operators of the direct and inverse DFT; $B_{\text{lim}}(\bullet)$ and $S_{\text{lim}}(\bullet)$ are, correspondingly, operators of the signal band limitation and space limitations; k_0 is the index of position of the delta function; and t is the iteration index. Functions

$$\text{sinclet}(S_{\text{lim}}; B_{\text{lim}}; k_0; k) = S_{\text{lim}}(IDFT\{B_{\text{lim}}(DFT\{\text{sinclet}(S_{\text{lim}}; B_{\text{lim}}; k_0; k)\})\}) \tag{7.8}$$

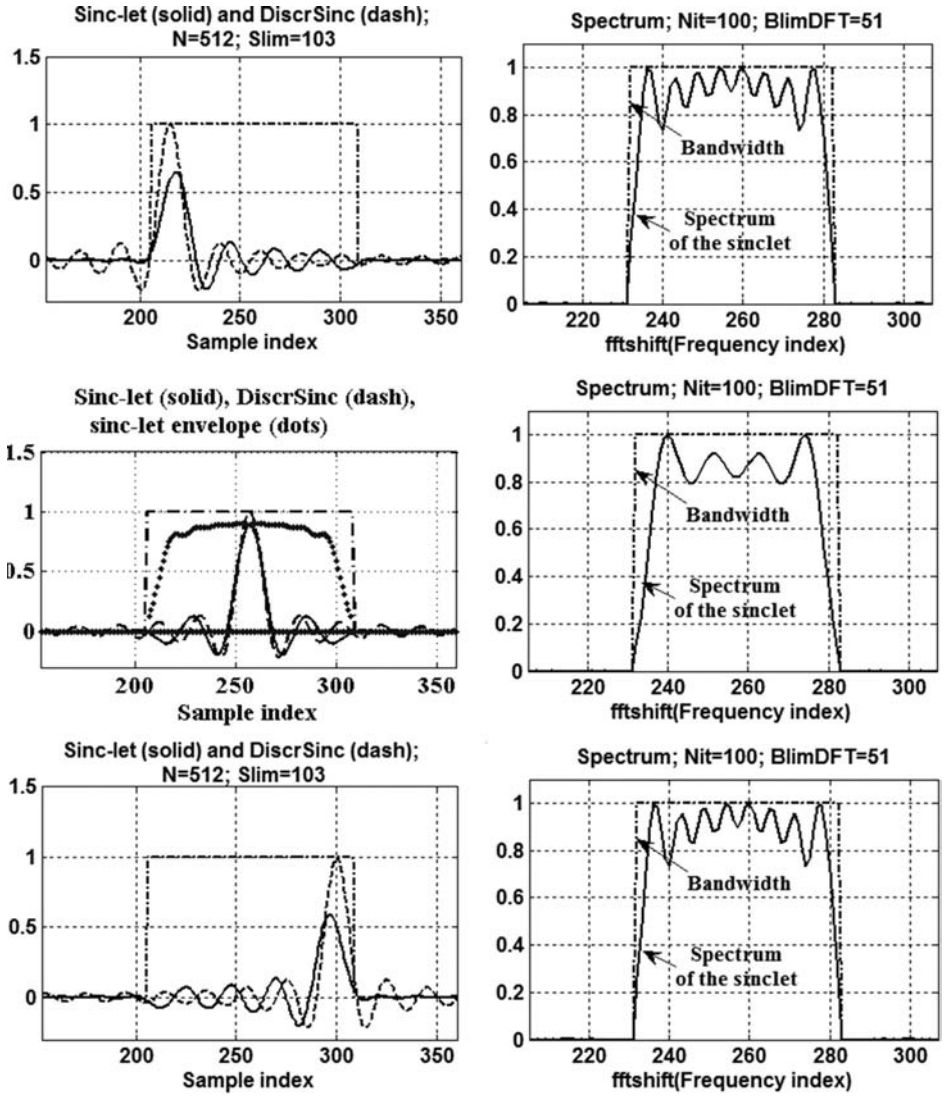


Figure 7.3 Examples of LP sinc-lets in different positions within an interval of 103 samples for a signal of 512 samples (solid lines, left column), and their corresponding DFT spectra (right column). LP sinc-lets are shown along with the corresponding discrete sincd functions of the same bandwidth (dashed lines).

that are a fixed point of this algorithm are called "*sinc-lets*" because they resemble discrete sinc functions, as can be seen from the illustrative examples generated by the program **Sinc_Lets_1D_DFT_SPIE.m** provided in the Exercises and presented in Fig. 7.3 for the case of a space interval of 103 samples and a spectral interval of 51 out of 512.

The speed of convergence of the intertive algorithm to generate sinc-lets is illustrated in Fig. 7.4 for two cases: (i) a space interval of 103 samples and spectral interval of 51 samples, and (ii) a space interval of 103 samples and spectral interval of 103 samples out of 512. On the vertical axes on these plots, the fraction of signal energy outside the chosen bounding interval is indicated, in this case, an interval of 103 samples out of 512.

Sinc-lets with spectra concentrated around signal dc components, such as those shown in Fig. 7.3, are low-pass band-limited sinc-lets (LP sinc-lets). Figure 7.3 shows that LP sinc-lets are shift-variant functions: their shapes and heights depend on the position.

Figures 7.5(a) and (b) show matrices of mutual correlations of LP sinc-lets in different positions and their central cross-sections [(c) and (d)] obtained for a space interval of 103 samples and spectral intervals of 51 and 103 samples. These matrices allow one to hypothesize that sinc-lets shifted by an interval $\Delta_N = N/B_{\text{lim}}$ inversely proportional to their bandwidth interval B_{lim} form a family of orthogonal functions. Band limitation in the DCT domain generates similar sinc-lets, as illustrated in Fig. 7.6.

The shape of 2D LP sinc-lets depends on the shape of their space and spectrum limitation. Obviously, for separable space and spectrum limitations,

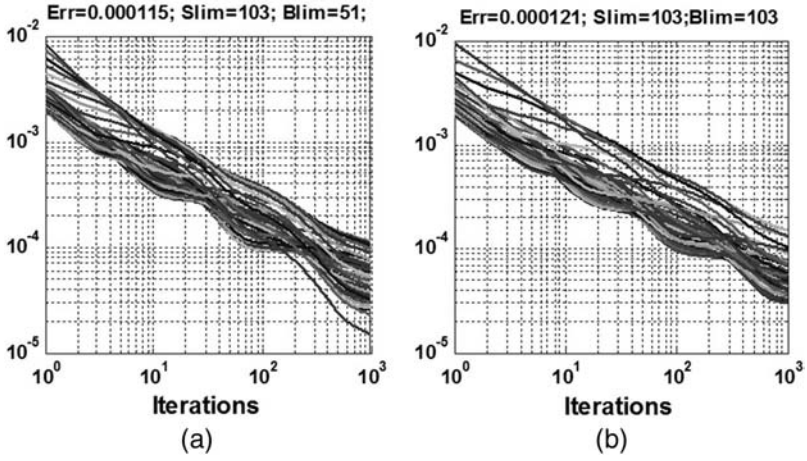


Figure 7.4 Illustrative examples of the iteration convergence that generates (a) LP sinc-lets shown in Fig. 7.3, left column, and (b) sinc-lets obtained on a sampling lattice of 512 samples for 103 non-zero signal samples and 103 non-zero samples of their DFT spectrum. The curves on these plots correspond to different positions of sinc-lets. The vertical axes on the plots represent the fraction of signal energy outside the chosen interval of 103 samples.

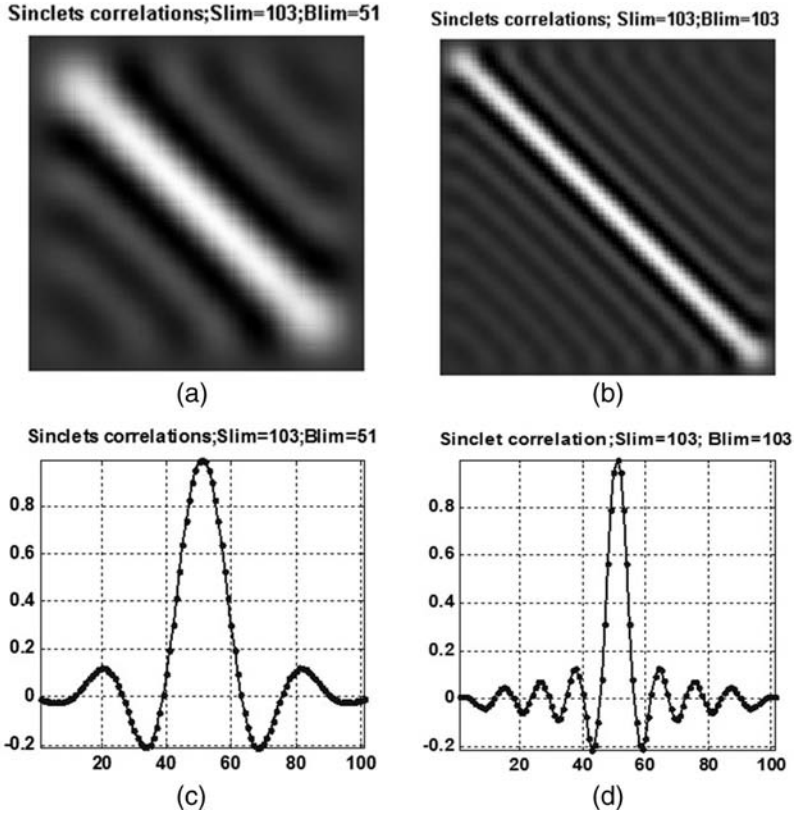


Figure 7.5 Matrices of mutual correlations of LP sinc-lets in different positions for two families of sinc-lets obtained for a space limitation of 103 samples and a band-limitation of 51 and 103 samples (displayed as images, upper row), and their corresponding central cross-sections (bottom row).

2D sinc-lets are products of corresponding 1D sinc-lets. Examples of 2D sinc-lets limited in space by circles and squares and circularly limited in the DFT and DCT domains are shown in Fig. 7.7.

7.3 Exercises

SpaceLimSpectrlim_SPIE.m

Generates space-limited and DFT- or DCT-spectrum band-limited images.
Three options for choosing seed images:

- Delta function,
- "White noise" (an array of uncorrelated pseudo-random numbers),
- Any image from an image database.

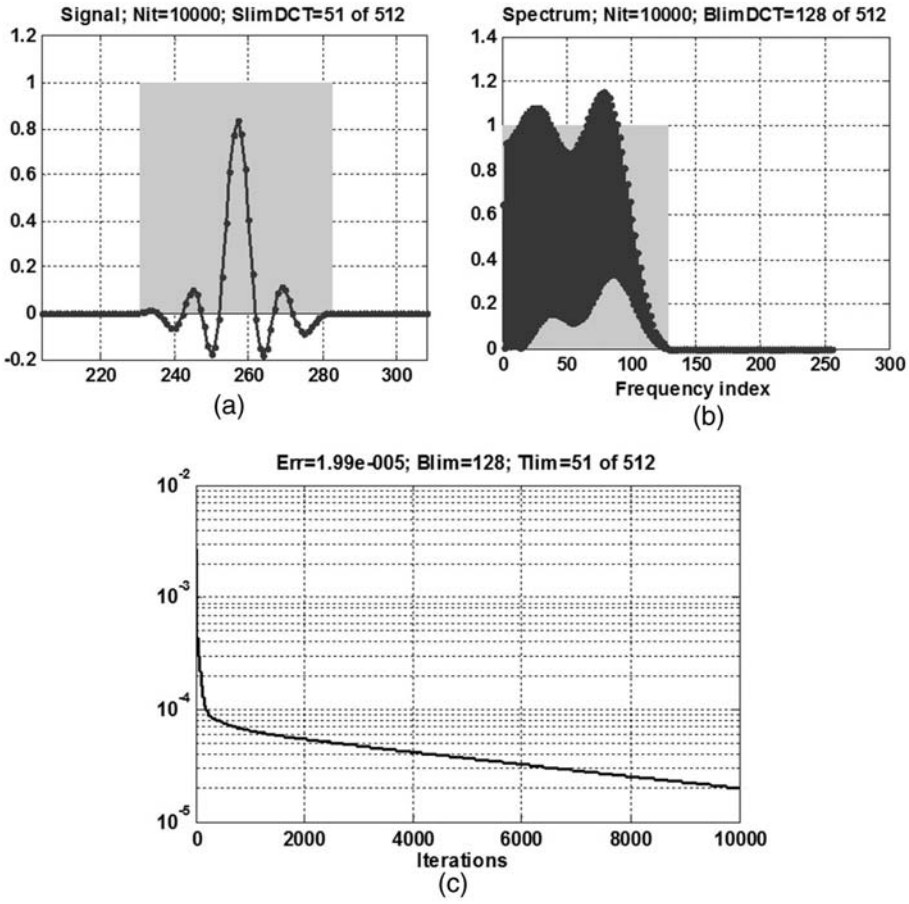


Figure 7.6 (a) Sinc-let and (b) its DCT spectrum generated by band limitation in the DCT domain, and (c) plot of the signal residual energy outside the selected interval vs. the number of iterations. Gray rectangles indicate intervals in the space and frequency domain within which the signal and its DCT spectrum are non-zero.

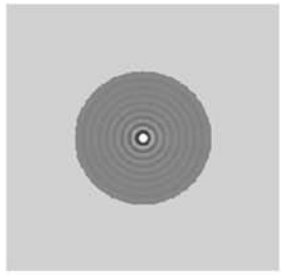
Two options for space-limitation shapes:

- Square,
- Circle.

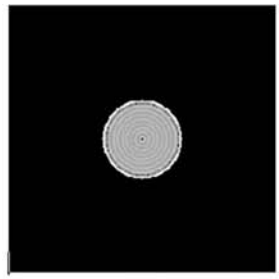
Two options for spectrum-band limitation:

- Square centered at the spectrum dc component,
- Circle for the DFT or pie sector for the DCT centered at the spectrum dc component.

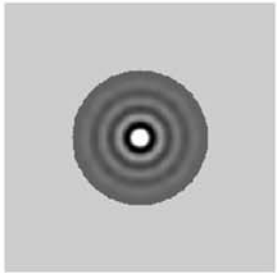
The area of the space- and spectrum-bounding shapes are specified in a fraction of the baseband.



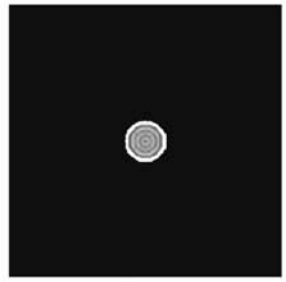
(a) LP DFT-Sinc-let:
N=256x256; SzLim=0.5; Blim=0.3



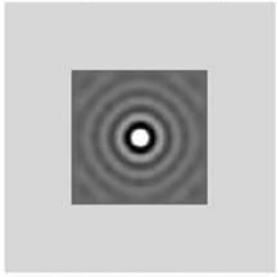
(e) LP DFT-Sinc-let spectrum:
N=256x256; SzLim=0.5; Blim=0.3



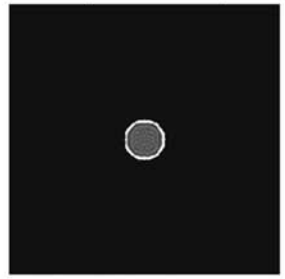
(b) LP DFT-Sinc-let:
N=256x256; SzLim=0.5; Blim=0.15



(f) LP DFT-Sinclet spectrum:
N=256x256; SzLim=0.5; Blim=0.15



(c) LP DFT-Sinc-let:
N=256x256; SzLim=0.5; Blim=0.15



(g) LP DFT-Sinc- let spectrum;
N=256x256; SzLim=0.5; Blim=0.15

(d) LP DCT-sinclet:
N=256x256; SzLim=0.5; Blim=0.15

(h) LP DCT-Sinc-let spectrum:
N=256x256; SzLim=0.5; Blim=0.15

Figure 7.7 Examples of (a)–(d) 2D circularly limited LP sinc-lets and (e)–(h) their circularly limited DFT and DCT spectra. Size limitation (Slim) and band limitation (Blim) are given in fractions of the size of the corresponding domain. All images are contrast enhanced for display purposes.

UncertaintyPrinciple_SPIE.m

Experimental verification of the uncertainty relationship between signal width and width of its DFT spectrum using an iterative algorithm for generating space-limited and band-limited signals. The initial test signal is the delta impulse. The range of the product is tested:

SignalWidth $*$ SpectrumWidth $*$ NumberofSamples: 0.5–10.0.

User-defined parameters:

- The total number of signal samples N,
- Test signal width (in fraction of N),
- The number of iterations.

Sinc_Lets_1D_DFT_SPIE.m

Iteratively generates 1D sinc-lets (space-limited and DFT band-limited signals). The seed signals are delta functions in different positions within the space-limitation interval. User-defined parameters:

- The number of samples N,
- The number of iterations N_{it},
- Fraction of the signal domain where the signal is a non-zero Slim,
- Fraction of the baseband where the spectrum is a non-zero Blim.

Part II
Discrete Representation of Signal Transformations

Chapter 8
Basic Principles of Discrete Representation of Signal Transformations

Digital signal processing always deals with processing analog (continuous) signals. For processing in digital processors, analog signals are first converted into a discrete form, which most frequently is implemented by means of signal sampling. The obtained discrete signals are then processed by computers or specialized digital processors. There are generally two goals of processing:

- generating numerical control data for executive devices, such as guiding moving vehicles;
- generating processed signals of the same nature as the input signals, such as imaging or audio processing.

In the latter case, signal processing is carried out according to the general diagram shown in Fig. 8.1.

The goal of obtaining output analog signals that, defined by the processing task, correspond to given input signals dictates the necessity of treating the entire processing system, including a signal sampling device, digital processing unit, and analog signal reconstruction device, as an equivalent analog system that should be characterized in terms of analog signal transformations. This implies that the following two principles of the digital representation of analog signal transformations must be put in the base discrete representation of analog signal transformations:

- *Consistency principle* with digital representation of signals, and
- *Mutual correspondence principle* between analog and discrete signal transformations.

The consistency principle requires that the digital representation of signal transformations should parallel that of signals, i.e., should be consistent. The mutual correspondence principle between continuous and digital transformations

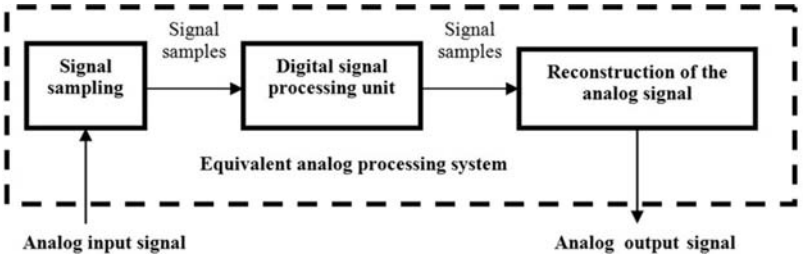


Figure 8.1 General flow-diagram of digital signal processing.

requires that both should act to transform identical input signals into identical output signals.

Following the consistency principle, we will assume (a) that each continuous signal $a(x)$ is represented in the discrete form by its samples $\{a_k\}$:

$$a_k = \int_{-\infty}^{\infty} a(x) PSF^{(s)}(k\Delta_x - x)dx \tag{8.1}$$

taken at nodes of a uniform sampling lattice with sampling interval Δ_x by a sampling device with point spread function $PSF^{(s)}(x)$, as defined by Eq. (2.19); and (b) that each continuous signal is reconstructed from its samples by their interpolation

$$a(x) = \sum_{k=-\infty}^{\infty} a_k PSF^{(r)}(x - k\Delta_x) \tag{8.2}$$

in a signal reconstruction device with point spread function $PSF^{(s)}(x)$, as defined by Eq. (2.22).

Chapters 9 and 10 will use these principles to derive discrete representations of the most important factors in analog signal transformations for signal processing: the convolution integral and the integral Fourier transform.

As a discrete representation of the convolution integral, a digital filter equivalent to a given analog filter is introduced, and its characterization in terms of point spread function and frequency response of the corresponding analog filter is developed. This enables the filter design and analysis.

As a discrete representation of the integral Fourier transform, various versions of the DFT (which play a fundamental role in digital signal processing) are derived, and their basic proprties are discussed.

Chapter 9
Discrete Representation of the Convolution Integral

9.1 Discrete Convolution

In signal theory, the convolution integral

$$b(x) = \int_{-\infty}^{\infty} a(\xi)h(x-\xi)d\xi \tag{9.1}$$

is a mathematical model of shift-invariant filtering signal $a(x)$ by a linear filter with point spread function (PSF) $h(x)$. A discrete representation of the convolution intergral can be obtained by finding a relationship between samples

$$b_k = \int_{-\infty}^{\infty} b(x)PSF^{(s)}(k\Delta_x - x)dx \tag{9.2}$$

of the convolution result $b(x)$ with samples $\{a_n\}$ of the convolved signal $a(x)$. Insert Eq. (9.1) in Eq. (9.2) and replace the former signal $a(x)$ with its expression (Eq. (8.2)) through its samples $\{a_n\}$:

$$\begin{aligned} b_k &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} a(\xi)h(x-\xi)d\xi\right]PSF^{(s)}(k\Delta_x - x)dx \\ &= \int_{-\infty}^{\infty} PSF^{(s)}(k\Delta_x - x)dx\left\{\int_{-\infty}^{\infty}\left[\sum_n a_n PSF^{(r)}(\xi - n\Delta_x)\right]h(x-\xi)d\xi\right\} \\ &= \sum_{n=-\infty}^{\infty} a_n \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\xi - x + (k-n)\Delta_x)PSF^{(r)}(\xi)PSF^{(s)}(x)d\xi dx \\ &= \sum_{n=-\infty}^{\infty} a_n h_{k-n}, \end{aligned} \tag{9.3}$$

where

$$h_m = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\xi - x - m\Delta_x)PSF^{(r)}(\xi)PSF^{(s)}(x)d\xi dx \tag{9.4}$$

are samples of the convolution kernel $h(x)$ taken using PSFs of the signal sampling and reconstruction devices.

The obtained equation

$$b_k = \sum_{n=-\infty}^{\infty} a_n h_{k-n} \tag{9.5}$$

represents the convolution integral (Eq. (9.1)) in a discrete form, in which samples $\{h_m\}$ act as a discrete convolution kernel or a filter PSF. Equation (9.4) shows how can one find samples $\{h_m\}$ given the analog convolution kernel $h(x)$ and point spread functions $PSF^{(s)}(x)$ and $PSF^{(r)}(x)$ of the analog signal sampling and reconstruction devices.

In practical applications, one should take into account that the number of signal samples N, as well as the number N_h of filter PSF samples, is finite and that as a rule $N_h << N$. Equation (9.5) should thus be modified in the following way:

$$b_k = \sum_{n=0}^{N_h-1} h_n a_{k-n}. \tag{9.6}$$

This equation is considered as the canonical discrete representation of analog shift-invariant filters and is referred to as a *digital filter*.

9.2 Point Spread Functions and Frequency Responses of Digital Filters

Similar to analog filters, digital filters are characterized by their PSFs and frequency responses. The set of coefficients $\{h_n\}$ of the digital filter defined by Eq. (9.6) is called a digital filter *discrete point spread function* (DPSF).

The correspondence principle between analog and discrete signal transformations dictates that a digital filter should also be characterized by the PSF of an equivalent analog filter. In order to derive this characteristic, the express analog signal $b(x)$ is reconstructed from a finite number N of output samples $\{b_k\}$ of a digital filter as

$$b(x) = \sum_{k=0}^{N-1} b_k PSF^{(r)}(x - k\Delta_x) = \sum_{k=0}^{N-1}\left(\sum_{n=0}^{N_h-1} h_n a_{k-n}\right)PSF^{(r)}(x - k\Delta_x). \tag{9.7}$$

Insert into Eq. (9.7) the expression Eq. (8.1) for samples $\{a_{k-n}\}$ of the input analog signal $a(x)$ to obtain

$$b(x) = \sum_{k=0}^{N-1}\left(\sum_{n=0}^{N_h-1} h_n \int_{-\infty}^{\infty} a(\xi)PSF^{(s)}[(k-n)\Delta_x - \xi]d\xi\right)PSF^{(r)}(x-k\Delta_x)$$
$$= \int_{-\infty}^{\infty} a(x)dx \sum_{k=0}^{N-1}\sum_{n=0}^{N_h-1} h_n PSF^{(s)}((k-n)\Delta_x - \xi)PSF^{(r)}(x-k\Delta_x). \tag{9.8}$$

The double sum in Eq. (9.8)

$$h_{eq}(x,\,\xi) = \sum_{k=0}^{N-1}\sum_{n=0}^{N_h-1} h_n PSF^{(s)}[(k-n)\Delta_x - \xi]PSF^{(r)}(x-k\Delta_x) \tag{9.9}$$

can be considered as the PSF of an analog filter that corresponds to a given digital filter. This equivalent analog filter

$$b(x) = \int_{-\infty}^{\infty} a(\xi)h_{eq}(x,\,\xi)d\xi \tag{9.10}$$

is a general, not shift-invariant, filter, though its corresponding digital filter is designed to represent the shift-invariant filter defined by Eq. (9.1).

Characterization of digital filters in terms of their point spread functions is complemented by their characterization in terms of filter frequency responses. In order to derive them, consider the Fourier spectrum of the filter output signal $b(x)$ defined by Eq. (9.10):

$$\beta(f) = \int_{-\infty}^{\infty} b(x)\exp(i2\pi f x)dx = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} a(\xi)h_{eq}(x,\,\xi)d\xi\right]\exp(i2\pi f x)dx$$
$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty}[\alpha(p)\exp(-i2\pi p\xi)dp]h_{eq}(x,\,\xi)d\xi\right]\exp(i2\pi f x)dx$$
$$\int_{-\infty}^{\infty}\alpha(p)dp\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_{eq}(x,\,\xi)d\xi\exp[i2\pi(fx-p\xi)]dx$$
$$= \int_{-\infty}^{\infty}\alpha(p)H_{eq}(f,\,p)dp, \tag{9.11}$$

where the function

$$H_{eq}(f,\,p) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_{eq}(x,\,\xi)\exp[i2\pi(fx-p\xi)]dxd\xi \tag{9.12}$$

can be regarded as a frequency response of the general filter defined by Eq. (9.10).

Now we find the frequency response of an analog filter equivalent to a given the digital filter defined by Eq. (9.6). Insert the expression Eq. (9.9) for its PSF $h_{eq}(x, \xi)$ into Eq. (9.12) to obtain

$$\begin{aligned}
H(f,p) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left\{\sum_{k=0}^{N-1}\sum_{n=0}^{N_h-1} h_n PSF^{(s)}[(k-n)\Delta_x-\xi]PSF^{(r)}(x-k\Delta_x)\right\} \\
&\quad\times\exp[i2\pi(fx-p\xi)]dxd\xi \\
&= \sum_{k=0}^{N-1}\sum_{n=0}^{N_h-1} h_n \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} PSF^{(s)}(\xi)PSF^{(r)}(x)\exp\{i2\pi[f(x+k\Delta_x)+p\xi \\
&\quad-(k-n)p\Delta_x]\}dxd\xi \\
&= \sum_{k=0}^{N}\sum_{n=0}^{N_h-1} h_n \int_{-\infty}^{\infty} PSF^{(s)}(\xi)\exp(i2\pi p\xi)d\xi \int_{-\infty}^{\infty} PSF^{(r)}(x)\exp(i2\pi fx)dx \\
&\quad\times\exp\{i2\pi[(f-p)k\Delta_x+pn\Delta_x]\} \\
&= \left[\int_{-\infty}^{\infty} PSF^{(r)}(x)\exp(i2\pi fx)dx\right]\times\left[\int_{-\infty}^{\infty} PSF^{(s)}(\xi)\exp(i2\pi p\xi)d\xi\right] \\
&\quad\times\left[\sum_{n=0}^{N_h-1} h_n\exp(i2\pi pn\Delta_x)\right]\times\left[\sum_{k=0}^{N-1}\exp[i2\pi(f-p)k\Delta_x]\right].
\end{aligned} \tag{9.13}$$

The first two multiplicands in the right part of Eq. (9.13) are frequency responses of the signal reconstruction and sampling devices:

$$FR^{(r)}(f) = \int_{-\infty}^{\infty} PSF^{(r)}(x)\exp(i2\pi fx)dx, \tag{9.14}$$

$$FR^{(s)}(p) = \int_{-\infty}^{\infty} PSF^{(r)}(\xi)\exp(i2\pi p\xi)d\xi. \tag{9.15}$$

The third multiplicand

$$DFCFR(p) = \sum_{n=0}^{N_h-1} h_n\exp(i2\pi pn\Delta_x) \tag{9.16}$$

is a Fourier series with coefficients $\{h_n\}$ of the digital-filter PSF. It can be treated as a *digital-filter continuous frequency response* (DFCFR).

The fourth multiplicand

$$DFSVM_N(f-p) = \sum_{k=0}^{N-1} \exp[i2\pi(f-p)k\Delta_x] \tag{9.17}$$

is defined solely by the number of samples of the digital filter output involved in the reconstruction of its analog output (Eq. (9.7)). It can be regarded as a *digital filter space-variance measure* (DFSVM) as it reflects the fact that the digital filters correspond to space-variant analog filters.

Function $DFSVM_N(f-p)$ can be computed in a closed form as

$$\begin{aligned} DFSVM_N(f-p) &= \sum_{k=0}^{N} \exp[i2\pi(f-p)k\Delta_x] = \frac{\exp[i2\pi(f-p)N\Delta_x]-1}{\exp[i2\pi(f-p)\Delta_x]-1} \\ &= \frac{\exp[i\pi(f-p)N\Delta_x]-\exp[-i\pi(f-p)N\Delta_x]}{\exp[i\pi(f-p)\Delta_x]-\exp[-i\pi(f-p)\Delta_x]}\exp[i\pi(f-p)(N-1)\Delta_x] \\ &= \frac{\sin[\pi(f-p)N\Delta_x]}{\sin[\pi(f-p)\Delta_x]}\exp[i\pi(f-p)(N-1)\Delta_x] \\ &= N\,\mathrm{sincd}[N;\,\pi(f-p)N\Delta_x]\exp[i\pi(f-p)(N-1)\Delta_x], \end{aligned} \tag{9.18}$$

where

$$\mathrm{sincd}(N;\,x) = \frac{\sin(x)}{N\sin(x/N)} \tag{9.19}$$

is the discrete sincd function.

The second multiplicand $\exp[i\pi(f-p)(N-1)\Delta_x]$ in the right part of Eq. (9.18) is only a phase-shift function that is defined solely by the order in which the signal samples $\{b_k\}$ are counted. It carries no other important information, so we will disregard it and define the DFSVM as

$$DFSVM(f-p) = N\,\mathrm{sincd}[\pi(f-p)N\Delta_x]. \tag{9.20}$$

Finally, the frequency response of the digital filter defined by its discrete point spread function $\{h_n\}$ is

$$DFOFR_N(f,\,p) = FR^{(r)}(f)FR^{(s)}(p)DFCFR(p)N\,\mathrm{sincd}[N;\,\pi(f-p)N\Delta_x] \tag{9.21}$$

This function links the Fourier spectra of filter input and output analog signals and can therefore be called the *digital-filter overall frequency response*.

One can easily see, when N tends to infinity, function $\mathrm{sincd}[N;\,\pi(f-p)N\Delta_x]$ converts to sinc function $\mathrm{sinc}[\pi(f-p)N\Delta_x]$, which, in its turn, tends to the delta function (see Appendix A1, Eq. (A1.40):

$$\lim_{N\to\infty} N\,\mathrm{sincd}[N;\ \pi(f-p)N\Delta_x] = \delta(f-p). \tag{9.22}$$

Therefore, in the limit, when the number of signal samples involved in the analog reconstruction tends to infinity, the digital-filter overall frequency response converts to

$$\lim_{N\to\infty}\{OFRDF_N(f,p)\} = FR^{(r)}(f)FR^{(s)}(p)CFRDF^{(h)}(p)\delta(f-p), \tag{9.23}$$

and Eq. (9.11) converts to

$$\begin{aligned}\beta(f) &= \int_{-\infty}^{\infty}\alpha(p)OFRDF(f,p)dp \\ &= \int_{-\infty}^{\infty}\alpha(p)FR^{(r)}(f)FR^{(s)}(p)CFRDF^{(h)}(p)\delta(f-p)dp \\ &= FR^{(r)}(f)FR^{(s)}(f)DFCFR(f)\alpha(f),\end{aligned} \tag{9.24}$$

which is the expression that, according to the convolution theorem for the Fourier transform (see Section A2.5), links the spectra $\alpha(f)$ and $\beta(f)$ of the input and output of a shift-invariant analog filter. Therefore, for a sufficiently large number N of signal samples, the analog filter equivalent to a given digital filter can be regarded as shift invariant.

The shift-invariant approximation of the digital filter overall frequency response

$$DFOFR_{SpInv}(f) = FR^{(r)}(f)FR^{(s)}(f)DFCFR(f) \tag{9.25}$$

has a clear physical interpretation: it equals the product of frequency responses of all stages of digital filtering analog signals. In particular, it shows that through an appropriate design of the digital filter (term $DFCFR(f)$), one can compensate signal distortions in the sampling baseband introduced by sampling and reconstruction devices (terms $FR^{(r)}(f)$ and $FR^{(s)}(f)$).

Note also that the continuous frequency response of a digital filter $DFCFR(f)$ is, according to Eq. (9.16), a periodic function with period $1/\Delta_x$. Periodic replicas of this function outside the signal sampling baseband affect periodic replicas of the signal spectrum, which, ideally, are supposed to be cut out by frequency responses of signal sampling and reconstruction devices. Otherwise, they contribute to signal sampling errors.

All of the above relationships are derived for 1D signals. Their extension to 2D signals is straightforward. In particular, a 2D digital filter is defined as

$$b_{k,l} = \sum_{m=0}^{N_h-1}\sum_{n=0}^{M_h-1} h_{n,m}a_{k-m,l-n}, \tag{9.26}$$

where $\{h_{n,m}\}$ is the filter's discrete PSF. The overall frequency response of a 2D digital filter is

$$\begin{aligned} &DFOFR_{N_x,\,N_y}(f_x, p_x; f_y, p_y) \\ &\quad = FR^{(r)}(f_x, f_y)FR^{(s)}(-p_x,\ -p_y)DFCFR(p_x, p_y) \\ &\quad \times N_x N_y \mathrm{sincd}[\pi(f_x - p_y)N_x\Delta_x]\mathrm{sincd}[\pi(f_y - p_y)N_y\Delta_y], \end{aligned} \tag{9.27}$$

where $FR^{(s)}(-p_x,\ -p_y)$ and $FR^{(r)}(f_x, f_y)$ are the frequency responses of signal sampling and reconstruction devices, (f_x, f_y) and (p_x, p_y) are spatial frequencies, and

$$DFCFR(p) = \sum_{m=0}^{N_h-1}\sum_{n=0}^{M_h-1} h_{m,n}\exp[i2\pi(p_x m\Delta_x + p_y y\Delta_y)] \tag{9.28}$$

is a continuous frequency response of the 2D digital filter with discrete PSF $\{h_{m,n}\}$.

9.3 Treatment of Signal Borders in Digital Convolution

Equation (9.6) defines digital convolution output signal samples $\{b_k\}$ only for $N_h - 1 \le k \le N - 1$, where N is the number of input signal samples, and N_h is the number of samples of the convolution kernel $\{h_n\}$. Signal samples for $k < N_h - 1$ and $k > N - 1$ are not defined because the input signal samples $\{a_n\}$ for $n < 0$ and $n > N - 1$ are not available. Therefore, digital filtering reduces the number of output signal samples unless the unavailable input signal samples are defined by some method of signal extrapolation. If so, then the $N^{(h)}/2$ output signal samples at the signal borders are influenced by the extrapolated values of the input signal; hence, the extrapolation method must be selected to minimize the undesirable signal filtering distortions at the signal borders.

Appropriate border processing is especially important in image processing because of the "curse of dimensionality." Compare, for instance, cases of an audio signal with 10^6 samples, which corresponds to approximately one minute of phonation, and an image frame of $10^3 \times 10^3 = 10^6$ pixels. Let the 1D digital filter PSF has $N^{(h)} = 100$ samples. Correspondingly, the 2D filter PSF of the same extent has $N^{(h)} = 100 \times 100$ samples. Then, for the audio signal, border effects influence only 10^{-4}th of the samples (6 ms of phonation), whereas for the image signal 10^{-2}th of the pixels are influenced, i.e., one-tenth of each of image frame dimension.

In principle, image extrapolation outside the available frame of image samples is a special case of signal recovery from an incomplete set of data. The solution to this problem requires the formulation of *a priori* knowledge of the

images or image ensembles and sophisticated and computationally expensive methods of signal recovery. In practice, the following simpler methods are commonly adopted:

– Padding signals outside the signal frame with zeros. This is, for instance, implemented in MATLAB signal and image processing. Justification for zero-padding may be found in the assumption that signals can be extended by their mean value, which by default is

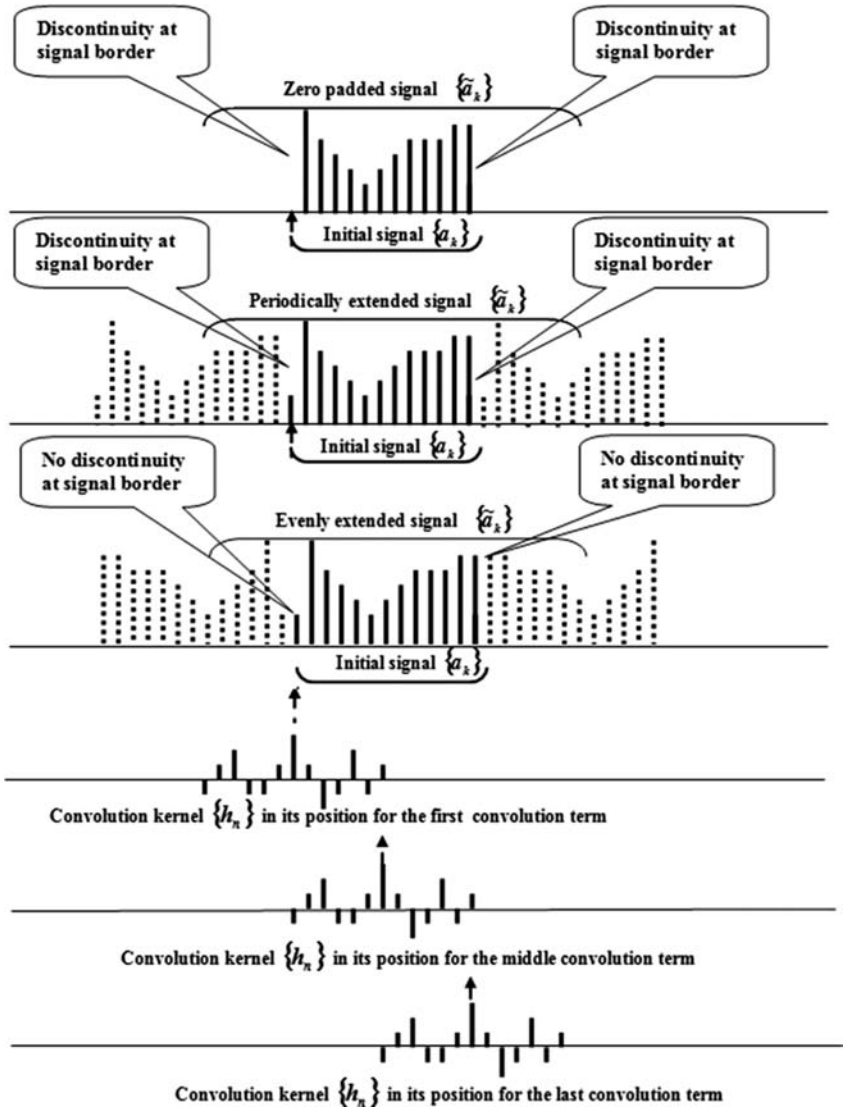


Figure 9.1 Signal border processing: zero-padding, periodic extension, and even extension by mirror reflection at the signal borders.

assumed to be zero. For many 1D signals, e.g., audio signals, the zero mean assumption appears quite natural. For images, this assumption (as a rule) is not relevant, although in some applications the images of objects have an empty (zero) background, which can be extended beyond the image frame. Astronomical images of extraterrestrial objects and some tomographic images of body slices exemplify these cases.

– In numerical and applied mathematics, another assumption, i.e., signal periodicity, is quite popular. It originates from the Fourier series expansion of functions and is well suited for tasks that assume the DFT, which, as shown in Appendix 2, implies by the definition that signals are periodic with a period that equals the number of signal samples. However, in most image processing tasks, the periodic image extension beyond its borders, which makes image samples at the left and right, or top and bottom, borders immediate neighbors, is not appropriate and frequently results in heavy border effects.
– The major source of border effects associated with the use of the above two signal extrapolation methods is that they tend to introduce discontinuities in the extrapolated signal at the borders of the initial signal, which causes severe distortions when signals are processed using the FFT, for instance, for fast signal convolution, correlation, resampling, or spectral analysis. A method that allows one to avoid these potential discontinuities is signal extension by using mirror reflection from the borders.

The described signal extension methods and the way they work in signal convolution are illustrated in Fig. 9.1.

Section 10.3 will show that even signal extension by mirror reflection from its borders translates the discrete Fourier transform into the discrete cosine transform (DCT), an orthogonal transform with very good energy compaction that can replace the DFT for all applications (including Fourier analysis, fast convolution, signal resampling, signal differentiation, etc.) that are sensitive to boundary effects.

Chapter 10
Discrete Representation of the Fourier Integral Transform

10.1 1D Discrete Fourier Transforms

Let $a(x)$ be a signal defined in its coordinate system (x) (upper plot in Fig. 10.1) and $\alpha(f)$ its integral Fourier transform (Fourier spectrum)

$$\alpha(f) = \int_{-\infty}^{\infty} a(x) \exp(i2\pi f x) dx \tag{10.1}$$

defined in the coordinate system of frequencies (f) in the Fourier transform domain (lower plot in Fig. 10.1).

According to the consistency principle for deriving the discrete representation of signal transformations, the signal $a(x)$ is assumed to belong to the family of signals represented by its N samples $\{a_k\}$ placed with sampling interval Δ_x at nodes of a uniform sampling lattice shifted with respect to the signal coordinate system by the $u^{(r)}$th fraction of sampling interval Δ_x (upper plot in Fig. 10.1), from which it can be reconstructed using a signal reconstruction device with the point spread function $PSF^{(r)}(x)$:

$$a(x) = \sum_{k=0}^{N-1} a_k PSF^{(r)}[x - (k + u^{(r)})\Delta_x]. \tag{10.2}$$

Assume also that the discrete representation of the signal spectrum $\alpha(f)$ is its samples $\{\alpha_r\}$

$$\alpha_r = \int_{-\infty}^{\infty} \alpha(f) PSF^{(s)}\left[(r + v^{(s)})\Delta_f - f\right] \tag{10.3}$$

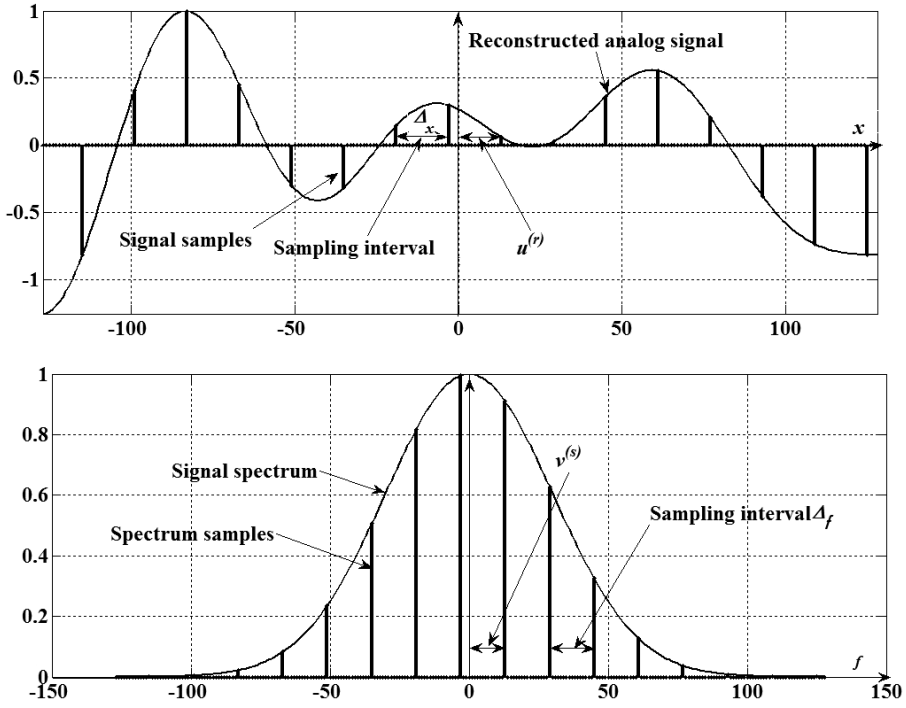


Figure 10.1 Sampling lattices for the signal (upper plot) and its Fourier spectrum (bottom plot).

taken by a sampling device with the point spread function $PSF^{(s)}(f)$ at the nodes of a uniform sampling lattice shifted with respect to the signal spectrum coordinate system by the $v^{(s)}$th fraction of the sampling interval Δ_f (bottom plot in Fig. 10.1). Accounting for possible shifts of sampling positions in the signal and transform domains is essential to derive the discrete representation of the integral Fourier transform because the Fourier transform is a shift-variant transform—as opposed to the convolution integral, where possible shifts of sampling positions with respect to the signal coordinate system do not matter—and they were for this reason ignored in the derivation of the discrete representation of the convolution integral in Chapter 9.

To make the formulas more compact, the shifted indices are denoted as

$$\tilde{r} = r + v^{(s)} \quad \text{and} \quad \tilde{k} = k + u^{(r)}. \tag{10.4}$$

Insert into Eq. (10.3) the expression Eq. (10.1) for signal spectrum and obtain

$$
\begin{aligned}
\alpha_r &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} a(x) \exp(i2\pi f x) dx \right] PSF^{(s)}(\tilde{r}\Delta_f - f) df \\
&= \int_{-\infty}^{\infty} a(x) dx \int_{-\infty}^{\infty} \exp(i2\pi f x) PSF^{(s)}(\tilde{r}\Delta_f - f) df \\
&= \int_{-\infty}^{\infty} a(x) dx \int_{-\infty}^{\infty} \exp[i2\pi(\tilde{r}\Delta_f - f)x] PSF^{(s)}(f) df \\
&= \int_{-\infty}^{\infty} a(x) \exp(i2\pi\tilde{r}\Delta_f x) dx \int_{-\infty}^{\infty} PSF^{(s)}(f) \exp(-i2\pi f x) df \\
&= \int_{-\infty}^{\infty} a(x) \exp(i2\pi\tilde{r}\Delta_f x) FR^{(s)}(x) dx,
\end{aligned}
\tag{10.5}
$$

where

$$
FR^{(s)}(x) = \int_{-\infty}^{\infty} PSF^{(s)}(f) \exp(-i2\pi f x) df \tag{10.6}
$$

is the frequency response of the spectrum sampling device. Replace $a(x)$ in Eq. (10.5) with its expression through its samples and obtain

$$
\begin{aligned}
\alpha_r &= \int_{-\infty}^{\infty} a(x) \exp(i2\pi\tilde{r}\Delta_f x) FR^{(s)}(x) dx \\
&= \int_{-\infty}^{\infty} \left[\sum_{k=0}^{N-1} a_k PSF^{(r)}(x - \tilde{k}\Delta_x) \right] \exp(i2\pi\tilde{r}\Delta_f x) FR^{(s)}(x) dx \\
&= \int_{-\infty}^{\infty} \left[\sum_{k=0}^{N-1} a_k PSF^{(r)}(x) \right] \exp[i2\pi\tilde{r}\Delta_f(x + \tilde{k}\Delta_x)] FR^{(s)}(x + \tilde{k}\Delta_x) dx \\
&= \sum_{k=0}^{N-1} a_k \exp(i2\pi\tilde{k}\,\tilde{r}\,\Delta_x\Delta_f) \int_{-\infty}^{\infty} PSF^{(r)}(x) FR^{(s)}(x + \tilde{k}\Delta_x) \exp(i2\pi\tilde{r}\Delta_f x) dx \\
&= \sum_{k=0}^{N-1} a_k \exp(i2\pi\tilde{k}\,\tilde{r}\,\Delta_x\Delta_f) \cdot \mathbf{PSFFR}(k, r),
\end{aligned}
\tag{10.7}
$$

where

$$\mathbf{PSFFR}(k, r) = \int_{-\infty}^{\infty} PSF^{(r)}(x) FR^{(s)}(x + \tilde{k}\Delta_x) \exp(i2\pi\tilde{r}\Delta_f x) dx. \tag{10.8}$$

This term is a constant when the PSFs of a spectrum sampling device $PSF^{(s)}(f)$ and a signal reconstruction device $PSF^{(r)}(x)$ are delta functions, and it is believed to be approximately constant when the PSFs are well-concentrated functions of their arguments. For this reason, the term $\mathbf{PSFFR}(k, r)$ is disregarded, and the definition of the discrete representation of the Fourier integral is built on the basis of the shortened relationship:

$$\alpha_r = \sum_{k=0}^{N-1} a_k \exp[i2\pi(k + u^{(r)})(r + v^{(s)})\Delta_x\Delta_f]. \tag{10.9}$$

Parameters $\left(u^{(r)}, v^{(s)}, \Delta_x, \Delta_f\right)$ are analog parameters of the signal reconstruction and spectrum sampling devices. Shift parameters $\left\{u^{(r)}, v^{(s)}\right\}$ can be chosen arbitrarily. Sampling intervals Δ_x, Δ_f are interrelated: according to the uncertainty principle for functions and their Fourier spectra (Section 7.1), the signal extent $X = N\Delta_x$ and sampling interval Δ_f in the signal spectrum domain are inversely proportional:

$$\Delta_f = 1/N\Delta_x. \tag{10.10}$$

This relationship between the number of signal samples N and sampling intervals Δ_x and Δ_f in the signal and spectral domains is called the *cardinal sampling relationship*. By applying it in Eq. (10.9) and assuming zero shifts of the signal and its spectrum sampling lattices ($u^{(r)} = 0$, $v^{(s)} = 0$), we obtain the following discrete representation of the Fourier integral transform:

$$\alpha_r = \sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{kr}{N}\right). \tag{10.11}$$

It is an orthogonal transform. In order to make it orthonormal, introduce to it a normalizing multiplier $1/\sqrt{N}$. In this way, we arrive at the following definition of the DFT:

$$\alpha_r = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{kr}{N}\right). \tag{10.12}$$

Depending on the sampling parameters, other versions of the DFT should be considered.

First, it is the case for non-zero shift parameters ($u^{(r)} \neq 0$, $v^{(s)} \neq 0$):

$$\alpha_r^{(u,v)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left[i2\pi \frac{(k+u^{(r)})(r+v^{(s)})}{N} \right]. \tag{10.13}$$

We will call this version of the DFT the *shifted discrete Fourier transform* (ShDFT).

Second, one should take into account that sampling in the signal or spectral domain can be performed in scaled coordinates with respect to those assumed in the definition of the integral transform given by Eq. (10.1), such that

$$\Delta_f = 1/\sigma N \Delta_x, \tag{10.14}$$

where $\sigma \neq 1$ is a scale parameter. In this case, obtain from Eq. (10.9) the following:

$$\alpha_r^{(u,v)} = \frac{1}{\sqrt{\sigma N}} \sum_{k=0}^{N-1} a_k \exp\left[i2\pi \frac{(k+u^{(r)})(r+v^{(s)})}{\sigma N} \right]. \tag{10.15}$$

The discrete transform in this form is irreversible unless σN is an integer number. Therefore, we must replace in Eq. (10.15) σN with its closest integer number $\lceil \sigma N \rceil$ and define the *scaled shifted discrete Fourier transform* (ScShDFT) as

$$\alpha_r^{(\sigma;u,v)} = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{k=0}^{N-1} a_k \exp\left[i2\pi \frac{(k+u^{(r)})(r+v^{(s)})}{\lceil \sigma N \rceil} \right]. \tag{10.16}$$

Its special case of zero shift parameters ($u^{(r)} = 0$, $v^{(s)} = 0$) is called the *scaled discrete Fourier transform* (ScDFT):

$$\alpha_r^{(\sigma)} = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{kr}{\lceil \sigma N \rceil} \right). \tag{10.17}$$

Note that direct scaled DFTs are just $\lceil \sigma N \rceil$-point DFTs of a signal padded with $(\lceil \sigma N \rceil - N)$ zeros.

The primary version of DFTs given by Eq. (10.12) is sometimes called a *canonic DFT*, depending on context, to distinguish it from other versions of the DFT.

As it is shown in Appendix A2.1, all of these transforms are invertible:

$$\text{Inverse canonic DFT: } a_k = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \alpha_r \exp\left(-i2\pi \frac{kr}{N} \right); \tag{10.18}$$

$$\text{Inverse shifted DFT: } a_k = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \alpha_r^{(u,v)} \exp\left[-i2\pi \frac{(k+u)(r+v)}{N}\right]; \quad (10.19)$$

$$\text{Inverse scaled DFT: } a_k = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{k=0}^{\lceil \sigma N \rceil - 1} \alpha_r^{(\sigma)} \exp\left(-i2\pi \frac{kr}{\lceil \sigma N \rceil}\right); \quad (10.20)$$

$$\text{Inverse scaled shifted DFT: } a_k = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{k=0}^{\lceil \sigma N \rceil - 1} \alpha_r^{(u,v,\sigma)} \exp\left[-i2\pi \frac{(k+u)(r+v)}{\lceil \sigma N \rceil}\right]. \quad (10.21)$$

Inverse scaled DFTs are equivalent to $\lceil \sigma N \rceil$-point inverse canonic DFTs of the spectrum padded with $\lceil \sigma N \rceil - N$ zeros to the length $\lceil \sigma N \rceil$.

Thanks to the availability of fast Fourier transform algorithms, the canonic DFT plays a fundamental role in signal processing. All other versions of the DFT are computable through the canonic DFT.

10.2 2D Discrete Fourier Transforms

Assuming that 2D signals are defined on a 2D rectangular sampling lattice of $N_1 \times N_2$ samples, 1D shifted scaled (direct and inverse) DFTs can be straightforwardly generalized to 2D shifted scaled DFTs as

$$\alpha_{r,s}^{(u_1,v_1;u_2,v_2)} = \frac{1}{\sqrt{\lceil \sigma_1 N_1 \rceil \lceil \sigma_2 N_2 \rceil}} \sum_{k=0}^{N_1-1} \sum_{l=0}^{N_2-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\tilde{r}}{\lceil \sigma_1 N_1 \rceil} + \frac{\tilde{l}\tilde{s}}{\lceil \sigma_2 N_2 \rceil}\right)\right]; \quad (10.22)$$

$$a_{k,l} = \frac{1}{\sqrt{\lceil \sigma_1 N_1 \rceil \lceil \sigma_2 N_2 \rceil}} \sum_{r=0}^{\lceil \sigma_1 N_1 \rceil - 1} \sum_{s=0}^{\lceil \sigma_2 N_2 \rceil - 1} \alpha_{r,s}^{(u_1,v_1;u_2,v_2)} \exp\left[-i2\pi \left(\frac{\tilde{k}\tilde{r}}{\lceil \sigma_1 N_1 \rceil} + \frac{\tilde{l}\tilde{s}}{\lceil \sigma_2 N_2 \rceil}\right)\right], \quad (10.23)$$

where $\{\tilde{k}, \tilde{l}\}$ and $\{\tilde{r}, \tilde{s}\}$ are biased indices of the signal and its spectrum samples:

$$\begin{aligned} &\tilde{k} = k + u_1;\ k = 0, \ldots, N_1 - 1;\ \tilde{l} = l + u_2;\ l = 0, \ldots, N_2 - 1; \\ &\tilde{r} = r + v_{1_1};\ r = 0, \ldots, N_1 - 1,\ \tilde{s} = s + v_{2_2},\ s = 0, \ldots, N_2 - 1; \end{aligned} \quad (10.24)$$

$\{u_1, u_2\}$ and $\{v_{1_1}, v_{2_2}\}$ are corresponding shifts of sampling lattices in the signal and Fourier transform domains; and $\{\sigma_1, \sigma_2\}$ are scale parameters that define the relationships between the signal, its spectrum sampling intervals

$\{\Delta x_1, \Delta x_2\}$, $\{\Delta f_1, \Delta f_2\}$, and the numbers $\{N_1, N_2\}$ of the signal and spectrum samples $\Delta x_1 \Delta f_{x_1} = 1/\sigma_1 N_2$, $\Delta x_2 \Delta f_{x_2} = 1/\sigma_2 N_2$. $\lceil x \rceil$ denotes the closest integer to x.

By definition, 2D DFTs are separable over their indices and can be computed in two separate passes, row-wise and column-wise:

$$\begin{aligned}\alpha_{r,s}^{(u,v)} &= \frac{1}{\sqrt{\lceil \sigma_1 N_1 \rceil \lceil \sigma_2 N_2 \rceil}} \sum_{k=0}^{N_1-1} \sum_{l=0}^{N_2-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\tilde{r}}{\lceil \sigma_1 N_1 \rceil} + \frac{\tilde{l}\tilde{s}}{\lceil \sigma_2 N_2 \rceil} \right) \right] \\ &= \frac{1}{\sqrt{N_1 N_2}} \sum_{k=0}^{N_1-1} \exp\left(i2\pi \frac{\tilde{k}\tilde{r}}{\lceil \sigma_1 N_1 \rceil} \right) \sum_{l=0}^{N_2-1} a_{k,l} \exp\left(i2\pi \frac{\tilde{l}\tilde{s}}{\lceil \sigma_2 N_2 \rceil} \right).\end{aligned} \tag{10.25}$$

Correspondingly, direct and inverse canonic 2D DFTs are defined as

$$\alpha_{r,s} = \frac{1}{\sqrt{N_1 N_2}} \sum_{k=0}^{N_1-1} \sum_{l=0}^{N_2-1} a_{k,l} \exp\left[i2\pi \left(\frac{kr}{N_1} + \frac{ls}{N_2} \right) \right], \tag{10.26}$$

$$a_{k,l} = \frac{1}{\sqrt{N_1 N_2}} \sum_{r=0}^{N_1-1} \sum_{s=0}^{N_2-1} \alpha_{r,s} \exp\left[-i2\pi \left(\frac{kr}{N_1} + \frac{ls}{N_2} \right) \right]. \tag{10.27}$$

A natural generalization of 2D shifted and scaled DFTs is the 2D *affine discrete Fourier transform* (AffDFT). The AffDFT is obtained with the assumption that either the signal or its spectrum sampling or reconstruction is carried out in coordinate systems $(\tilde{x}_1, \tilde{x}_2)$, which are affine transformed, with respect to signal/spectrum coordinate systems (x_1, x_2):

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix}, \tag{10.28}$$

where $\begin{bmatrix} A & B \\ C & D \end{bmatrix}$ is a matrix of the coordinate affine transform.

With $\sigma_A = 1/N_1 A \Delta_{\tilde{x}_1} \Delta_{f_1}$, $\sigma_B = 1/N_2 B \Delta_{\tilde{x}_2} \Delta_{f_2}$, $\sigma_C = 1/N_1 C \Delta_{\tilde{x}_2} \Delta_{f_{x_2}}$, and $\sigma_D = 1/N_2 D \Delta_{\tilde{x}_2} \Delta_{f_2}$—where $\Delta_{\tilde{x}_1}$, $\Delta_{\tilde{x}_2}$, Δ_{f_1}, and Δ_{f_2} are the signal and its Fourier transform sampling intervals in the image $(\tilde{x}_1, \tilde{x}_2)$ and Fourier (f_1, f_2) planes—the AffDTF is defined as

$$\alpha_{r,s} = \sum_{k=0}^{N_1-1} \sum_{l=0}^{N_2-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{r}\tilde{k}}{\lceil \sigma_A N_1 \rceil} + \frac{\tilde{s}\tilde{k}}{\lceil \sigma_C N_1 \rceil} + \frac{\tilde{r}\tilde{l}}{\lceil \sigma_B N_2 \rceil} + \frac{\tilde{s}\tilde{l}}{\lceil \sigma_D N_2 \rceil} \right) \right]. \tag{10.29}$$

A special case of affine transforms is rotation. For rotation angle θ, the signal coordinate transformation is defined as

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix}. \tag{10.30}$$

With $N_1 = N_2 = N$, $\Delta_{\tilde{x}_1} = \Delta_{\tilde{x}_2} = \Delta_x$, $\Delta_{f_1} = \Delta_{f_2} = \Delta_f$, and $\Delta_x \Delta_f = 1/N$, the 2D *rotated DFT* (RotDFT) is obtained as

$$\alpha^{\theta}_{r,s} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\cos\theta + \tilde{l}\sin\theta}{N} \tilde{r} - \frac{\tilde{k}\sin\theta - \tilde{l}\cos\theta}{N} \tilde{s} \right) \right]. \tag{10.31}$$

An obvious generalization of the RotDFT is the *rotated and scaled DFT* (RotScDFT)

$$\alpha^{\theta}_{r,s} = \frac{1}{\lceil \sigma N \rceil} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\cos\theta + \tilde{l}\sin\theta}{\lceil \sigma N \rceil} \tilde{r} - \frac{\tilde{k}\sin\theta - \tilde{l}\cos\theta}{\lceil \sigma N \rceil} \tilde{s} \right) \right], \tag{10.32}$$

which assumes 2D signal sampling in a θ-rotated and σ-scaled coordinate system.

The most important properties of the DFTs are reviewed in Appendix 2.

10.3 Discrete Cosine Transform

There are special cases of shifted DFTs worth special consideration, all of which are associated with signals and/or spectra that exhibit certain symmetry. The most important case is that of *discrete cosine transform* (DCT).

Consider a signal $\{a_k\}$, $k = 0, 1, \ldots, N-1$. Form from this signal an auxiliary symmetrized signal

$$\tilde{a}_k = \begin{cases} a_k, & k = 0, 1, \ldots, N-1 \\ a_{2N-k-1}, & k = N, \ldots, 2N-1 \end{cases} \tag{10.33}$$

and compute the shifted DFT of signal $\{\tilde{a}_k\}$ with shift parameters $(1/2, 0)$:

$$\begin{aligned}
\alpha_r &= \frac{1}{\sqrt{2N}} \sum_{k=0}^{2N-1} \tilde{a}_k \exp\left(i2\pi \frac{k+1/2}{2N} r\right) \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{k=0}^{N-1} \tilde{a}_k \exp\left(i2\pi \frac{k+1/2}{2N} r\right) + \sum_{k=N}^{2N-1} \tilde{a}_k \exp\left(i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{k+1/2}{2N} r\right) + \sum_{k=N}^{2N-1} a_{2N-k-1} \exp\left(i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{k+1/2}{2N} r\right) + \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{2N-k-1/2}{2N} r\right) \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{k+1/2}{2N} r\right) + \sum_{k=0}^{N-1} a_k \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k \cos\left(\pi \frac{k+1/2}{N} r\right).
\end{aligned} \tag{10.34}$$

In this way we produce the DCT defined for signal $\{a_k\}$ with N samples as

$$\alpha_r^{DCT} = \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k \cos\left(\pi \frac{k+1/2}{N} r\right). \tag{10.35}$$

Based on the definition of the DCT (Eq. (10.34)), the DCT signal spectrum is an odd (anti-symmetric) sequence if regarded outside its base interval $[0, N-1]$

$$\alpha_r^{DCT} = -\alpha_{2N-r}^{DCT};\ \alpha_N = 0, \tag{10.36}$$

whereas the signal, by definition (Eq. (10.33)), is perfectly evenly symmetric:

$$a_k = a_{2N-k-1}. \tag{10.37}$$

The inverse DCT can be found as the inverse ShDFT $(1/2, 0)$ of the spectrum $\{\alpha_r^{DCT}\}$ with its odd symmetry (Eq. (10.36)):

$$
\begin{aligned}
a_k &= \frac{1}{\sqrt{2N}} \sum_{r=0}^{2N-1} \alpha_r^{DCT} \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \\
&= \frac{1}{\sqrt{2N}} \left\{ \alpha_0^{DCT} + \sum_{r=1}^{N-1} \alpha_r^{DCT} \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=N+1}^{2N-1} \alpha_r^{DCT} \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \alpha_0^{DCT} + \sum_{r=1}^{N-1} \alpha_r^{DCT} \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=1}^{N-1} \alpha_r^{DCT} \exp\left[-i2\pi \frac{(k+1/2)(2N-r)}{2N}\right] \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \alpha_0^{DCT} + \sum_{r=1}^{N-1} \alpha_r^{DCT} \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=1}^{N-1} \alpha_r^{DCT} \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \alpha_0^{DCT} + 2 \sum_{r=1}^{N-1} \alpha_r^{DCT} \cos\left(\pi \frac{k+1/2}{N} r\right).
\end{aligned}
\tag{10.38}
$$

Thus, the definition of the inverse DCT is

$$
a_k = \frac{1}{\sqrt{2N}} \left[\alpha_0^{DCT} + 2 \sum_{r=1}^{N-1} \alpha_r^{DCT} \cos\left(\pi \frac{k+1/2}{N} r\right) \right]. \tag{10.39}
$$

As a shifted DFT of symmentrized signals, the DCT satisfies the Parseval's relationship:

$$
\sum_{k=0}^{N-1} a_k^2 = \sum_{r=0}^{N-1} \left|\alpha_r^{DCT}\right|^2. \tag{10.40}
$$

The coefficient α_0^{DCT} is, similar to the case of the DFT, proportional to the signal dc component:

$$
\alpha_0^{DCT} = \sqrt{2N} \left(\frac{1}{N} \sum_{k=0}^{N-1} a_k \right). \tag{10.41}
$$

The $(N-1)$th DCT coefficient

$$\alpha_{N-1}^{DCT} = \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k (-1)^k \sin\left(\pi \frac{k+1/2}{N}\right) \tag{10.42}$$

represents the signal's highest frequency. This equation is similar to that for the DFT (Eq. (A2.49)) except that the signal is multiplied by an "apodization" function $\left\{\sin\left(\pi \frac{k+1/2}{N}\right)\right\}$, which gradually puts the sinal values toward its borders down to zero.

The DCT was first introduced by Ahmed, Natarajan, and Rao[28] as an approximation of the Karunen–Loew transform, and it has proved to have some of the best energy compaction among orthogonal transforms.[16] This feature can be attributed to the fact that the DCT is the ShDFT (1/2, 0) of signals that are evenly extended outside its borders (Eq. (10.33)), which eliminates any signal discontinuities at the signal borders that are characteristic for DFT by virtue of its cyclicity and, therefore, eliminates the need to reproduce them in high-frequency spectral components.

Being a derivative of DFTs, the DCT can, in principle, be computed using the FFT with the same computational complexity of $\log N$ operations per each of N signal samples. There also exist dedicated fast transform algorithms of the FFT type to compute the DCT.

The absence of discontinuities at signal borders and the computational efficiency make the DCT attractive for many signal and image processing applications, such as audio and image compression, image perfection and enhancement, as well as signal and image resampling (which are discussed in Chapters 5 and 6). The DCT proved also to be a perfect substitute for the DFT to implement a boundary-effect-free, fast signal digital convolution with fast transforms. This application will be discussed in the next section.

The DCT has a complementary sine transform, the *discrete cosine-sine transform* (DcST):

$$\alpha_r^{DcST} = \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k \sin\left(\pi \frac{k+1/2}{N} r\right). \tag{10.43}$$

The DcST is an imaginary part of the ShDFT with shift parameters (1/2,0)

$$\alpha_r^{DcST} = \frac{1}{i\sqrt{2N}} \sum_{k=0}^{2N-1} \tilde{a}_k \exp\left(\pi \frac{k+1/2}{N} r\right) = \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k \sin\left(\pi \frac{k+1/2}{N} r\right) \tag{10.44}$$

of a signal

$$\tilde{a}_k = \begin{cases} a_k, & k = 0, 1, \ldots, N-1 \\ -a_{2N-k-1}, & k = N, N+1, \ldots, 2N-1 \end{cases} \tag{10.45}$$

extended to the interval $[0, 2N-1]$ in an odd (anti-symmetric) way. From this definition, it follows that the DcST spectrum exhibits even symmetry when regarded in an extended interval $[0, 2N-1]$:

$$\alpha_r^{DcST} = \alpha_{2N-r}^{DcST}, \tag{10.46}$$

and that it assumes periodic replication, with a period $2N$, of the signal complemented with its anti-symmetrical copy:

$$\tilde{a}_k = \tilde{a}_{(k) \bmod 2N}. \tag{10.47}$$

Distinct from the DFT and the DCT, the DcST does not contain a signal dc component:

$$\alpha_0^{DcST} = 0, \tag{10.48}$$

which is an obvious consequence of the signal's odd symmetry (Eq. (10.45)), and the inverse DcST

$$a_k = \frac{1}{\sqrt{2N}} \left[\alpha_N^{DcST} + 2 \sum_{r=1}^{N-1} \alpha_r^{DcST} \sin\left(\pi \frac{k+1/2}{N} r \right) \right] \tag{10.49}$$

involves spectral coefficients with indices $\{1, 2, \ldots, N\}$ rather than $\{0, 1, \ldots, N-1\}$ for the DCT.

Direct and inverse DcSTs can be computed through the DCT if one reverses the order of the transform coefficient:

$$\begin{aligned} \alpha_{N-r}^{DcST} &= \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k \sin\left[\pi \frac{(k+1/2)(N-r)}{N} \right] \\ &= \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} a_k \left\{ \sin[\pi(k+1/2)] \cos\left(\pi \frac{k+1/2}{N} r \right) \right. \\ &\quad \left. - \cos[\pi(k+1/2)] \sin\left(\pi \frac{k+1/2}{N} r \right) \right\} \\ &= \frac{2}{\sqrt{2N}} \sum_{k=0}^{N-1} (-1)^k a_k \cos\left(\pi \frac{k+1/2}{N} r \right), \end{aligned} \tag{10.50}$$

$$
\begin{aligned}
a_k &= \frac{1}{\sqrt{2N}} \left\{ \alpha_N^{DcST} + 2 \sum_{r=1}^{N-1} \alpha_{N-r}^{DcST} \sin\left[\pi \frac{(k+1/2)(N-r)}{N} \right] \right\} \\
&= \frac{1}{\sqrt{2N}} \left(\alpha_N^{DcST} + 2(-1)^k \sum_{r=1}^{N-1} \alpha_{N-r}^{DcST} \cos\left(\pi \frac{k+1/2}{N} r \right) \right).
\end{aligned}
\tag{10.51}
$$

2D and multi-dimensional DCTs and DcSTs are defined as a separable combination of 1D transforms on each of the dimensions. For instance, the 2D DCT of a signal $\{a_{k,l}\}$, $k = 0, 1, \ldots, N_1 - 1$, $l = 0, 1, \ldots, N_2 - 1$ is defined as

$$
\alpha_{r,s}^{DCT} = \frac{2}{\sqrt{N_1 N_2}} \sum_{k=0}^{N_1-1} \sum_{l=0}^{N_2-1} a_{k,l} \cos\left(\pi \frac{k+1/2}{N_1} r \right) \cos\left(\pi \frac{l+1/2}{N_2} s \right), \tag{10.52}
$$

which corresponds to the four-fold image symmetry

$$
a_{k,l} = a_{2N-k-1,l} = a_{k,2N-l-1} = a_{2N-k-,2N-l-1} \tag{10.53}
$$

illustrated in Fig. 10.2.

10.4 Boundary-Effect-Free Signal Convolution in the DCT Domain

One of the most important applications of DFT in signal processing is fast computing signal convolution. It is based on the convolution theorem (Appendix A2, Section A2.5) for the DFT and the availability of FFT algorithms. However, convolution using the FFT is a cyclic convolution that implies that signals and convolution kernels are periodic sequences. Because of this, signal samples at the left and the right signal borders act like immediate neighbors in computing convolution in the vicinity of signal borders. This can substantially deteriorate the signal structure and cause heavy boundary effects, which will be discussed in Section 9.3. It was shown that even signal extension to double length by means of mirror reflection from their borders is a reasonable practical solution for treating border effects in signal convolution. This section will show that the application of the DFT convolution theorem to signals evenly extended to double length by mirror reflection from their borders translates to a convolution in the DCT domain.

Let signal $\{\tilde{a}_k\}$ be obtained from the signal $\{a_k\}$ of N samples by mirror reflection extension and periodic replication of the result with a period of $2N$ samples:

Figure 10.2 Four-fold image symmetry.

$$\tilde{a}_{(k)\bmod 2N} = \begin{cases} a_k, & k = 0, 1, \ldots, N-1 \\ a_{2N-k-1}, & k = N, N+1, \ldots, 2N-1, \end{cases} \tag{10.54}$$

and let $\{\tilde{h}_n\}$ be a desired convolution kernel $\{h_n\}$ of N samples ($n = 0, 1, \ldots, N-1$) zero-padded to the length $2N$:

$$\tilde{h}_n = \begin{cases} 0 \quad , & n = 0, \ldots, \lfloor N/2 \rfloor - 1 \\ h_{n-\lfloor N/2 \rfloor}, & n = \lfloor N/2 \rfloor, \ldots, \lfloor N/2 \rfloor + N - 1 \\ 0 \quad , & \lfloor N/2 \rfloor + N, \ldots, 2N-1, \end{cases} \tag{10.55}$$

where $\lfloor N/2 \rfloor$ is an integer part of $N/2$. The first N samples of the cyclic convolution

$$c_{(k)\bmod 2N} = \sum_{n=0}^{N-1} \tilde{h}_n \tilde{a}_{(k-n+\lfloor N/2 \rfloor)\bmod 2N} \tag{10.56}$$

are the desired convolution result, in which the involvement of distant signal samples from the signal's opposite sites associated with the cyclicity of

convolution is removed due to the signal mirrored extension shown in Fig. 9.1.

Consider the computing convolution of such signals by the inverse DFT of the product of the signal and convolution kernel DFT spectra. The DFT spectrum of the extended signal $\{\tilde{a}_k\}$ is

$$\begin{aligned}\tilde{\alpha}_r &= \frac{1}{\sqrt{2N}}\sum_{k=0}^{2N-1}\tilde{a}_k \exp\left(i2\pi\frac{kr}{2N}\right)\\ &= \left\{\frac{2}{\sqrt{2N}}\sum_{k=0}^{N-1}a_k\cos\left(\pi\frac{k+1/2}{N}r\right)\right\}\exp\left(-i\pi\frac{r}{2N}\right)\\ &= \alpha_r^{(DCT)}\exp\left(-i\pi\frac{r}{2N}\right),\end{aligned} \tag{10.57}$$

where $\{\alpha_r^{(DCT)}\}$ are the DCT coefficients of the initial signal $\{a_k\}$.

For computing convolution, the signal spectrum defined by Eq. (10.57) should be multiplied by the DFT coefficients of the zero-padded convolution kernel:

$$\tilde{\eta}_r = \frac{1}{\sqrt{2N}}\sum_{n=0}^{2N-1}\tilde{h}_n\exp\left(i2\pi\frac{nr}{2N}\right), \tag{10.58}$$

and then the inverse DFT of the product $\tilde{\alpha}_r\tilde{\eta}_r$ should be computed for the first N samples:

$$\begin{aligned}b_k &= \frac{1}{\sqrt{2N}}\sum_{r=0}^{2N-1}\alpha_r^{(DCT)}\exp\left(-i\pi\frac{r}{2N}\right)\tilde{\eta}_r\exp\left(-i2\pi\frac{kr}{2N}\right)\\ &= \frac{1}{\sqrt{2N}}\sum_{r=0}^{2N-1}\alpha_r^{(DCT)}\tilde{\eta}_r\exp\left(-i2\pi\frac{k+1/2}{2N}r\right).\end{aligned} \tag{10.59}$$

Split the sum in Eq. (10.59) into two terms and change the index r of summation in the second term to $2N - r$. Then obtain

$$
\begin{aligned}
b_k &= \frac{1}{\sqrt{2N}} \sum_{r=0}^{2N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{r=0}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=N}^{2N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{r=0}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=1}^{N} \alpha_{2N-r}^{(DCT)} \tilde{\eta}_{2N-r} \exp\left[-i2\pi \frac{(k+1/2)(2N-r)}{2N}\right] \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{r=0}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=1}^{N} \alpha_{2N-r}^{(DCT)} \tilde{\eta}_{2N-r} \exp[-i2\pi(k+1/2)] \exp\left(i2\pi \frac{k+1/2}{2N} r\right) \right\} \\
&= \frac{1}{\sqrt{2N}} \left\{ \sum_{r=0}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. - \sum_{r=1}^{N} \alpha_{2N-r}^{(DCT)} \tilde{\eta}_{2N-r} \exp\left(i2\pi \frac{k+1/2}{2N} r\right) \right\}.
\end{aligned}
\tag{10.60}
$$

By virtue of the properties of the DFT and DCT spectra (Eq. (10.36)), it follows that $\alpha_N^{DCT} = 0$, $\alpha_r^{DCT} = -\alpha_{2N-r}^{DCT}$, and $\{\tilde{\eta}_r = \tilde{\eta}_{2N-r}^*\}$. The relationships in Eq. (10.60) produce

$$
\begin{aligned}
b_k &= \frac{1}{\sqrt{2N}} \left\{ \sum_{r=0}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r \exp\left(-i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&\quad \left. + \sum_{r=1}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r^* \exp\left(i2\pi \frac{k+1/2}{2N} r\right) \right. \\
&= \frac{1}{\sqrt{2N}} \left\{ \alpha_0^{(DCT)} \tilde{\eta}_0 + 2 \sum_{r=1}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r^{re} \cos\left(\pi \frac{k+1/2}{N} r\right) \right. \\
&\quad \left. - 2 \sum_{r=1}^{N-1} \alpha_r^{(DCT)} \tilde{\eta}_r^{im} \sin\left(\pi \frac{k+1/2}{N} r\right) \right\},
\end{aligned}
\tag{10.61}
$$

where $\tilde{\eta}_r^{re}$ and $\tilde{\eta}_r^{im}$ are real and imaginary parts of $\tilde{\eta}_r$. The first two terms of this expression represent the inverse DCT of the product $\{\alpha_r^{(DCT)}\tilde{\eta}_r^{re}\}$, while the third term is the DcST of the product $\{\alpha_r^{(DCT)}\tilde{\eta}_r^{im}\}$. As shown in the previous section, the inverse DcST can be converted to the DCT by changing the summation index r in the DcST by $N - r$ (Eq. (10.50)):

$$\sum_{r=1}^{N-1} \alpha_{N-r}^{(DCT)}\tilde{\eta}_{N-r}^{im} \sin\left[\pi \frac{(k+1/2)(N-r)}{N}\right] = (-1)^k \sum_{r=1}^{N-1} \alpha_{N-r}^{(DCT)}\tilde{\eta}_{N-r}^{im} \cos\left(\pi \frac{k+1/2}{N} r\right). \tag{10.62}$$

Substitute this expression into Eq. (10.61) and obtain the final formula for computing (through the DCT) the digital convolution with virtually no boundary effects:

$$b_k = \frac{1}{\sqrt{2N}} \left\{ \alpha_0^{(DCT)}\tilde{\eta}_0 + 2\sum_{r=1}^{N-1} \alpha_r^{(DCT)}\tilde{\eta}_r^{re} \cos\left(\pi \frac{k+1/2}{N} r\right) - 2(-1)^k \sum_{r=1}^{N-1} \alpha_{N-r}^{(DCT)}\tilde{\eta}_{N-r}^{im} \cos\left(\pi \frac{k+1/2}{N} r\right) \right\}. \tag{10.63}$$

10.5 DFT and Discrete Frequency Responses of Digital Filters

This section will show that the frequency responses of digital filters, which are their main characteristics, can be expressed through the DFT coefficients of their PSFs. Consider the continuous frequency response (Eq. (9.16)) of a digital filter defined by its discrete point spread function (DPSF) $\{h_n\}$

$$DFCFR(f) = \sum_{n=0}^{N-1} h_n \exp\left[i2\pi f \Delta_x \left(n + u^{(s)}\right)\right] \tag{10.64}$$

or

$$DFCFR(f) = \sum_{n=0}^{N-1} h_n \exp\left(i2\pi \frac{f}{\Delta_f} \frac{n+u^{(s)}}{N}\right) \tag{10.65}$$

because according to Eq. (10.10)), $\Delta_x \Delta_f = 1/N$. This expression accounts for the possible shift $u^{(s)}$ of the sampling lattice with respect to the signal coordinate system because it affects the phase of the filter frequency response.

Let $\left\{\eta_r^{(u,0)}\right\}$ be a set of shifted DFT coefficients of the filter DPSF $\{h_n\}$ with shift parameters $(u, 0)$:

$$h_n = \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},\,0)} \exp\left(-i2\pi \frac{n+u}{N} r\right). \tag{10.66}$$

The set of coefficients $\{\eta_r^{(u^{(s)},0)}\}$ can be called the *digital filter discrete frequency response* (DFDFR). Then obtain

$$\begin{aligned}
DFCFR(f) &= \sum_{n=0}^{N-1} \left\{ \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},0)} \exp\left(-i2\pi \frac{n+u}{N} r\right) \right\} \exp\left(i2\pi \frac{f}{\Delta_f} \frac{n+u^{(s)}}{N}\right) \\
&= \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},0)} \exp\left[i2\pi\left(\frac{f}{\Delta_f}\frac{u^{(s)}}{N} - \frac{ur}{N}\right)\right] \left\{ \sum_{n=0}^{N-1} \exp\left[i2\pi\left(\frac{f}{\Delta_f} - r\right)\frac{n}{N}\right] \right\} \\
&= \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},0)} \exp\left[i2\pi\left(\frac{f}{\Delta_f}\frac{u^{(s)}}{N} - \frac{ur}{N}\right)\right] \frac{\exp[i2\pi(f/\Delta_f - r)] - 1}{\exp\left(i2\pi\frac{f/\Delta_f - r}{N}\right) - 1} \\
&= \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},0)} \exp\left[i2\pi\left(\frac{f}{\Delta_f}\frac{u^{(s)}}{N} - \frac{ur}{N}\right)\right] \\
&\quad \times \frac{\exp[i\pi(f/\Delta_f - r)] - \exp[-i\pi(f/\Delta_f - r)]}{\exp\left(i\pi\frac{f/\Delta_f - r}{N}\right) - \exp\left(-i\pi\frac{f/\Delta_f - r}{N}\right)} \\
&= \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},0)} \exp\left[i2\pi\left(\frac{f}{\Delta_f}\frac{u^{(s)}}{N} - \frac{ur}{N}\right)\right] \frac{\sin[\pi(f/\Delta_r - r)]}{\sin\left(\pi\frac{f/\Delta_f - r}{N}\right)} \\
&\quad \times \exp\left[i\pi\left(\frac{f}{\Delta_f} - r\right)\frac{N-1}{N}\right] \\
&= \sum_{r=0}^{N-1} \eta_r^{(u^{(s)},0)} \exp\left[i\pi\left(\frac{f}{\Delta_f}\frac{2u^{(s)}+N-1}{N} - \frac{2u+N-1}{N} r\right)\right] N \frac{\sin[\pi(f/\Delta_f - r)]}{N\sin\left(\pi\frac{f/\Delta_f - r}{N}\right)}.
\end{aligned} \tag{10.67}$$

Finally, with the natural choice of shift parameters $u = u^{(s)} = -(N-1)/2$, obtain the continuous frequency response of a digital filter with a discrete frequency response $\left\{\eta_r^{\left(-\frac{N-1}{2},0\right)}\right\}$:

$$DFCFR(f) = N \sum_{r=0}^{N-1} \eta_r^{\left(-\frac{N-1}{2},0\right)} \mathrm{sincd}[N,\, \pi(f/\Delta_f - r)], \tag{10.68}$$

where $\mathrm{sincd}(N;\, x)$ is the discrete sincd function.

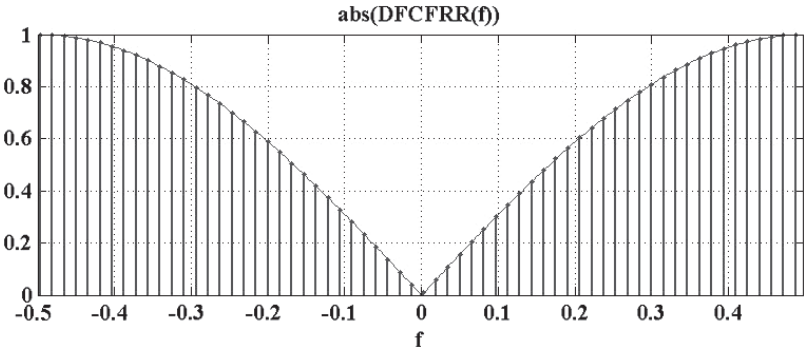


Figure 10.3 Continuous (solid line) and discrete frequency response (stems) of a digital filter that is frequently used for numerical evaluation of signal first derivatives.

Equation (10.68) implies that, at points $f = r\Delta f$, $r = 0, \ldots, N-1$ within one period of the periodicity of $CFRDF(f)$, its values are proportional to the coefficients $\{\eta_r\}$ of the filter's discrete frequency response. Between these sampling points, $CFRDF(f)$ is discrete sincd interpolated from samples $\{\eta_r\}$ of its discrete frequency response.

Figure 10.3 illustrates the discrete and continuous frequency responses of a digital filter with a PSF $[-1, 1]$, which is frequently used for numerical differentiation. The solid line in the figure shows the absolute value of the frequency response in the signal baseband $[-1/2\Delta x \div 1/2\Delta x]$; stems on the curve indicate samples $\{\eta_r\}$ of the continuous frequency response that represent the filter's discrete frequency response.

Chapter 5 showed how the notion of digital-filter continuous and discrete frequency responses is used to design discrete filters and, specifically, perfect discrete interpolation, differentiation, and integration filters.

10.6 Exercises

dft_demo_SPIE.m

Demonstration of properties and applications of the DFT. Test options are

- Symmetry properties of the DFT;
- Signal spectra
  - DFT spectra of sinusoidal signals,
  - DFT spectra of sampled images,
  - DFT spectra of interactively chosen image 32- × 32-pixel fragments;
- Shift theorem: shift invariance of DFT power spectrum;
- Convolution theorem (cyclic convolution);
- Ideal low-pass filtering in the DFT domain.

convolution_demo_SPIE.m

Demonstration of digital convolution implemented with MATLAB function conv2m, which assumes that the image is supposed to be zero outside its border. Test options include

- the principle of convolution;
- computing the image's local means in windows with 3×3, 5×5, and 15×15 pixels;
- computing the image's Laplacian.

conv_evenextension_demo_SPIE.m

Demo of convolution in the DCT domain (convolution of signals extended outside their borders by mirror reflection from the borders). Three options for the convolution kernel are

- Rectangular impulse (computing signal local mean),
- "Mexical hat,"
- Simple differentiator (convolution kernel [–1,1]).

Pseudo-random, band-limited, computer-generated sequences are used as test signals. Displayed are the

- Test signal with current position of the convolution kernel in the course of convolution;
- Test signal extended to its double length;
- Test signal extended to its double length with a convolution kernel; overlaid in its current position in the course of convolution;
- Filtered signal in the process of filtering with a convolution kernel overlaid in its current position.

dct_vs_dft_conv_demo_SPIE.m

Illustration of differences in the treatment of signal borders between DCT-based and DFT-based convolutions. Shown is a rectangular convolution kernel moving sample by sample along the test signal (Kronecker delta) from the signal center toward its right border, then from its left border back to the center.

Appendix 1
Fourier Series, Integral Fourier Transform, and Delta Function

A1.1 1D Fourier Series

Function $a(x)$ given on an interval $[-X/2 < x < X/2]$ can be expanded in a Fourier series

$$a(x) = \text{rect}\left(\frac{x}{X}\right) \sum_{k=-\infty}^{\infty} C_k \exp\left(i2\pi \frac{kx}{X}\right), \tag{A1.1}$$

where

$$C_k = \frac{1}{X} \int_{-X/2}^{X/2} a(x) \exp\left(i2\pi \frac{kx}{X}\right) dx \tag{A1.2}$$

and

$$\text{rect}(x) = \begin{cases} 1, & -1/2 < x < 1/2 \\ 0, & \text{otherwise} \end{cases} \tag{A1.3}$$

is a rectangular window function.

Function

$$\tilde{a}(x) = \sum_{k=-\infty}^{\infty} C_k \exp\left(-i2\pi \frac{kx}{X}\right) \tag{A1.4}$$

is a periodic function with a period X: for any integer m,

$$\begin{aligned}\tilde{a}(x+Xm) &= \sum_{k=-\infty}^{\infty} C_k \exp\left[-i2\pi\frac{k(x+Xm)}{X}\right] \\ &= \sum_{k=-\infty}^{\infty} C_k \exp(-i2\pi km)\exp\left(-i2\pi\frac{kx}{X}\right) \\ &= \sum_{k=-\infty}^{\infty} C_k \exp\left(-i2\pi\frac{kx}{X}\right) = \tilde{a}(x).\end{aligned} \tag{A1.5}$$

A1.2 2D Fourier Series

The 2D function $a(x, y)$ given on an interval $[-X/2 < x < X/2; [-Y/2 < y < Y/2;]]$ can be expanded in a 2D Fourier series

$$a(x, y) = \mathrm{rect}\left(\frac{x}{X}\right)\mathrm{rect}\left(\frac{y}{Y}\right)\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} C_{k,l}\exp\left[-i2\pi\left(\frac{kx}{X}+\frac{ly}{Y}\right)\right], \tag{A1.6}$$

where

$$C_k = \frac{1}{XY}\int_{-X/2}^{X/2}\int_{-Y/2}^{Y/2} a(x, y)\exp\left[i2\pi\left(\frac{kx}{X}+\frac{ly}{Y}\right)\right]dxdy. \tag{A1.7}$$

Function

$$\tilde{a}(x) = \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} C_{k,l}\exp\left[-i2\pi\left(\frac{kx}{X}+\frac{ly}{Y}\right)\right] \tag{A1.8}$$

is a periodic function in Cartesian coordinates with a 2D period (X, Y): for any integer m and n,

$$\begin{aligned}\tilde{a}(x+Xm, y+Yn) &= \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} C_{k,l}\exp\left[-i2\pi\left(\frac{k(x+Xm)}{X}+\frac{l(y+Yn)}{Y}\right)\right] \\ &= \sum_{k=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} C_{k,l}\exp(-i2\pi km)\exp\left(-i2\pi\frac{kx}{X}\right)\exp(-i2\pi ln)\exp\left(-i2\pi\frac{ly}{Y}\right) \\ &= \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} C_{k,l}\exp\left[-i2\pi\left(\frac{kx}{X}+\frac{ly}{Y}\right)\right] = \tilde{a}(x, y).\end{aligned} \tag{A1.9}$$

A1.3 1D Integral Fourier Transform

The 1D integral Fourier transform of a function $a(x)$ is defined as

$$\alpha(f) = \int_{-\infty}^{\infty} a(x) \exp(i2\pi f x) dx. \tag{A1.10}$$

Function $\alpha(f)$ is called the Fourier spectrum of function $a(x)$. The argument f of the spectrum is called its frequency.

The integral Fourier transform is invertible:

$$a(x) = \lim_{F \to \infty} \int_{-F/2}^{F/2} \alpha(f) \exp(-i2\pi f x) df. \tag{A1.11}$$

In this sense, the inverse 1D integral Fourier transform is defined as

$$a(x) = \int_{-\infty}^{\infty} \alpha(f) \exp(-i2\pi f x) df. \tag{A1.12}$$

The most important properties of the integral Fourier transform are

1. Spectrum symmetry

If function $a(x)$ is real, i.e., $a(x) = a^*(x)$, where the asterisk symbolizes a complex conjugate, the following relationship holds for its Fourier spectrum $\alpha(f)$:

$$\alpha(f) = \alpha^*(-f). \tag{A1.13}$$

2. Convolution theorem

Let $\alpha(f)$ be the Fourier spectrum of function $a(x)$, and let $\beta(f)$ be the Fourier spectrum of function $b(x)$. Then, function $c(x)$ with a Fourier spectrum equal to the product of spectra $\alpha(f)$ and $\beta(f)$ is a convolution of functions $a(x)$ and $b(x)$:

$$c(x) = \int_{-\infty}^{\infty} a(\xi) b(x - \xi) d\xi. \tag{A1.14}$$

Proof:

$$c(x) = \int_{-\infty}^{\infty} \alpha(f)\beta(f)\exp(-i2\pi f x)df$$
$$= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} a(\xi)\exp(i2\pi f\xi)d\xi \right) \beta(f)\exp(-i2\pi f x)df$$
$$= \int_{-\infty}^{\infty} a(\xi)d\xi \left(\int_{-\infty}^{\infty} \beta(f)\exp\left[-i2\pi f(x-\xi)\right]df \right) = \int_{-\infty}^{\infty} a(\xi)b(x-\xi)d\xi. \tag{A1.15}$$

In signal theory, convolution describes signal shift-invariant filtering:

$$a(x) = \int_{-\infty}^{\infty} PSF(\xi)a(x-\xi)d\xi. \tag{A1.16}$$

The convolution kernel $PSF(\xi)$ is called the filter PSF. Its Fourier transform

$$FR(f) = \int_{-\infty}^{\infty} PSF(x)\exp(i2\pi f x)dx \tag{A1.17}$$

is called the *filter frequency response*. According to the convolution theorem, signal shift-invariant filtering is described in the frequency domain of the signal Fourier transform as multiplication of the signal spectrum by the filter frequency response.

3. Shift theorem

Let $\alpha(f)$ be the Fourier spectrum of function $a(x)$. The spectrum of function $a(x+\delta_x)$ shifted by an δ_x interval with respect to function $a(x)$ is $\alpha(f)\exp(-i2\pi f\delta_x)$. This follows directly from the definition of the Fourier transform (Eq. (A1. 10)).

4. Modulation theorem

Let $\alpha(f)$ be the Fourier spectrum of function $a(x)$. The spectrum $\tilde{\alpha}_{\cos}(f)$ of signal $\tilde{a}_{\cos}(x) = a(x)\cos(2\pi f_0 x)$ modulated by a cosinusoidal signal of frequency f_0 is

$$\tilde{\alpha}_{\cos}(f) = \frac{\alpha(f+f_0) + \alpha(f-f_0)}{2}, \tag{A1.18}$$

and the spectrum $\tilde{\alpha}_{\sin}(f)$ of signal $\tilde{a}_{\sin}(x) = a(x)\sin(2\pi f_0 x)$ modulated by a sinusoidal signal of frequency f_0 is

$$\tilde{\alpha}(f) = \frac{\alpha(f+f_0) - \alpha(f-f_0)}{2i}. \tag{A1.19}$$

These relationships follow directly from Eq. (A1.10) and from identities

$$\cos\theta = [\exp(i\theta) + \exp -(i\theta)]/2;\ \sin\theta = [\exp(i\theta) - \exp -(i\theta)]/2i. \tag{A1.20}$$

5. Parseval's relation

The signal energy, defined as

$$E_a = \int_{-\infty}^{\infty} |a(x)|^2 dx, \tag{A1.21}$$

is equal to the energy of its Fourier spectrum:

$$\int_{-\infty}^{\infty} |a(x)|^2 dx = \int_{-\infty}^{\infty} |\alpha(f)|^2 df. \tag{A1.22}$$

Proof:

$$\begin{aligned}\int_{-\infty}^{\infty} |a(x)|^2 dx &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \alpha(f)\exp(-i2\pi f x)df\right] a^*(x)dx \\ &= \int_{-\infty}^{\infty} \alpha(f)\left[\int_{-\infty}^{\infty} a^*(x)\exp(-i2\pi f x)dx\right] df \\ &= \int_{-\infty}^{\infty} \alpha(f)\alpha^*(f)df = \int_{-\infty}^{\infty} |\alpha(f)|^2 df.\end{aligned} \tag{A1.23}$$

A1.4 2D Integral Fourier Transform

The 2D direct and inverse integral Fourier transform are defined in a rectangular coordinate system as separable over two coordinates (x, y) as

$$\alpha(f_x, f_y) = \int_{-\infty}^{\infty} a(x, y)\exp[i2\pi(f_x x + f_y y)]dxdy, \tag{A1.24}$$

$$a(x, y) = \int_{-\infty}^{\infty} \alpha(f_x, f_y)\exp[-i2\pi(f_x x + f_y y)]df_x df_y. \tag{A1.25}$$

Five of the most important properties of the 1D integral Fourier transform listed above translate to the 2D integral Fourier transfom as follows.

1. Spectrum symmetry

If function $a(x, y)$ is real, i.e., $a(x, y) = a^*(x, y)$, where the asterisk symbolizes a complex conjugate, then

$$\alpha(f_x, f_y) = \alpha^*(-f_x, -f_y). \tag{A1.26}$$

2. Convolution theorem

Let $\alpha(f_x, f_y)$ be the Fourier spectrum of function $a(x, y)$ and let $\beta(f_x, f_y)$ be the Fourier spectrum of function $b(x, y)$. Function $c(x, y)$ with a Fourier spectrum equal to the product of spectra $\alpha(f_x, f_y)$ and $\beta(f_x, f_y)$ can be found as a convolution of functions $a(x, y)$ and $b(x, y)$:

$$c(x) = \int_{-\infty}^{\infty} a(\xi, \eta) b(x - \xi, y - \eta) d\xi d\eta. \tag{A1.27}$$

3. Shift theorem

Let $\alpha(f_x, f_y)$ be the 2D Fourier spectrum of function $a(x, y)$. The spectrum $\tilde{\alpha}(f_x, f_y)$ of function $a(x + \delta_x, y + \delta_y)$ shifted by a 2D interval (δ_x, δ_y) with respect to function $a(x, y)$ is

$$\tilde{\alpha}(f_x, f_y) = \alpha(f_x, f_y) \exp[-i2\pi(f_x\delta_x, f_y\delta_y)]. \tag{A1.28}$$

4. Modulation theorem

Let $\alpha(f_x, f_y)$ be the Fourier spectrum of function $a(x, y)$. The spectrum $\tilde{\alpha}_{\cos}(f_x, f_y)$ of signal $\tilde{a}_{\cos}(x, y) = a(x, y)\cos[2\pi(f_{x_0}x + f_{y_0}y)]$ modulated by a 2D cosinusoidal signal of frequency (f_{x_0}, f_{y_0}) is

$$\tilde{\alpha}_{\cos}(f) = \frac{\alpha(f_x + f_{x_0}, f_y + f_{y_0}) + \alpha(f_x - f_{x_0}, f_y - f_{y_0})}{2}, \tag{A1.29}$$

and the spectrum $\tilde{\alpha}_{\sin}(f_x, f_y)$ of signal $\tilde{a}_{\sin}(x, y) = a(x)\sin[2\pi(f_{x_0}x + f_{y_0}y)]$ modulated by a sinusoidal signal of frequency (f_{x_0}, f_{y_0}) is

$$\tilde{\alpha}_{\sin}(f) = \frac{\alpha(f_x + f_{x_0}, f_y + f_{y_0}) - \alpha(f_x - f_{x_0}, f_y - f_{y_0})}{2i}. \tag{A1.30}$$

5. Parseval's relation

Parseval's relation (Eq. (A1.22)) holds for a signal and its Fourier spectra of any dimensionality.

A1.5 Delta Function, Sinc Function, and the Ideal Low-Pass Filter

Let $\varphi(x, \xi)$ be a kernel of an invertible integral transform Φ:

$$\alpha(\xi) = \Phi[a(x)] = \int_{-\infty}^{\infty} a(x)\varphi_{Dir}(x, \xi)dx, \tag{A1.31}$$

and $\varphi_{inv}(x, \xi)$ is a reciprocal kernel of the transform $\bar{\Phi}$ inverse to Φ:

$$a(x) = \bar{\Phi}[\alpha(\xi)] = \int_{-\infty}^{\infty} \alpha(\xi)\varphi_{Inv}(\xi, x)d\xi. \tag{A1.32}$$

Insert Eq. (A1.31) into Eq. (A1.32),

$$\begin{aligned} a(x) &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} a(\bar{x})\varphi_{Dir}(\bar{x}, \xi)d\bar{x} \right) \varphi_{Inv}(\xi, x)d\xi \\ &= \int_{-\infty}^{\infty} a(\bar{x}) \left(\int_{-\infty}^{\infty} \varphi_{Dir}(\bar{x}, \xi)\varphi_{Inv}(\xi, x)d\xi \right) d\bar{x} \end{aligned} \tag{A1.33}$$

and introduce a function

$$\delta(x, \bar{x}) = \int_{-\infty}^{\infty} \varphi_{Dir}(\bar{x}, \xi)\varphi_{Inv}(\xi, x)d\xi \tag{A1.34}$$

such that

$$a(x) = \int_{-\infty}^{\infty} a(\bar{x})\delta(x, \bar{x})d\bar{x}. \tag{A1.35}$$

The function $\delta(x, \bar{x})$, according to Eq. (A1.35), can be regarded as a kernel of the identical integral transform and symbolizes the invertibility of the transform Φ; it is called the *Dirac delta function*. Equation (A1.35) is the general definition of the delta function. The most important properties are

– The delta function $\delta(x, \bar{x})$ of the integral Fourier transform is shift invariant:

$$\delta(x, \bar{x}) = \delta(x - \bar{x}); \tag{A1.36}$$

– The delta function is an even function:

$$\delta(x) = \delta(-x); \tag{A1.37}$$

– The integral of the delta function over any finite interval equals one:

$$\int_{-X/2}^{X/2} \delta(x)dx = \int_{-\infty}^{\infty} rect\left(\frac{x}{X}\right)\delta(x)dx = rect\left(\frac{0}{X}\right) = 1. \tag{A1.38}$$

The delta function of the integral Fourier transform can be found by inserting Eq. (A1.10) into Eq. (A1.11):

$$\begin{aligned} a(x) &= \lim_{F\to\infty} \int_{-F/2}^{F/2} \left(\int_{-\infty}^{\infty} a(\xi)\exp(i2\pi f\xi)d\xi \right) \exp(-i2\pi f x)df \\ &= \int_{-\infty}^{\infty} a(\xi)d\xi \lim_{F\to\infty} \int_{-F/2}^{F/2} \exp\left[-i2\pi f(x-\xi)\right]df. \end{aligned} \tag{A1.39}$$

Therefore, the delta function of the integral Fourier transform is defined as

$$\begin{aligned} \delta(x,\xi) = \delta(x-\xi) &= \lim_{F\to\infty} \int_{-F/2}^{F/2} \exp\left[-i2\pi f(x-\xi)\right]df \\ &= \lim_{F\to\infty} \left(\frac{\exp\left[-i\pi F(x-\xi)\right] - \left[i\pi F(x-\xi)\right]}{-i2\pi(x-\xi)} \right) \\ &= \lim_{F\to\infty} \left(F\frac{\sin\left[\pi F(x-\xi)\right]}{\pi F(x-\xi)} \right) \\ &= \lim_{F\to\infty} \left(F\,\mathrm{sinc}[\pi F(x-\xi)]\right) \\ &= \int_{-\infty}^{\infty} \exp\left[-i2\pi f(x-\xi)\right]df. \end{aligned} \tag{A1.40}$$

Function

$$\mathrm{sinc}(x) = \frac{\sin x}{x} \tag{A1.41}$$

is called the *sinc function*, which plays a fundamental role in sampling theory. The most important properties are

– The sinc function is an even function:

$$\text{sinc}(x) = \text{sinc}(-x); \tag{A1.42}$$

– The sinc function $\text{sinc}(\pi F x)$ has zeros at equidistant points $\{x = k/F\}$, $k = \pm 1,\ \pm 2,$ ...., except point $x = 0$, where it equals one:

$$\text{sinc}(\pi F x) = \begin{cases} 1, & x = 0 \\ 0, & x = k/F,\ k = \pm 1,\ \pm 2,\ .... \end{cases} \tag{A1.43}$$

A plot of function $\text{sinc}(\pi F x)$ is shown in Fig. A1.1.

– The sinc function $\text{sinc}(\pi F x)$ is the Fourier transform of the rectangular window function $\text{rect}(f/F)/F$:

$$\begin{aligned} \frac{1}{F}\int_{-\infty}^{\infty} \text{rect}\left(\frac{f}{F}\right)\exp(-i2\pi f x)df &= \frac{1}{F}\int_{-F/2}^{F/2}\exp(-i2\pi f x)df \\ &= \frac{\sin(\pi F x)}{\pi F x} = \text{sinc}(\pi F x). \end{aligned} \tag{A1.44}$$

Therefore, the sinc function is a PSF of the *ideal low-pass filter* that passes signal frequency components within frequency interval $[-F/2, F/2]$ and rejects all frequencies outside this interval. According to the sampling theorem, ideal low-pass filters are those used for image reconstruction from samples with mininal mean square reconstruction error.

A1.6 Poisson Summation Formula

Consider function $\sum_{k=-\infty}^{\infty}\delta\left(f - \frac{k}{\Delta_x}\right)$. It is a periodic function with period $1/\Delta_x$. Represent it as a Fourier series:

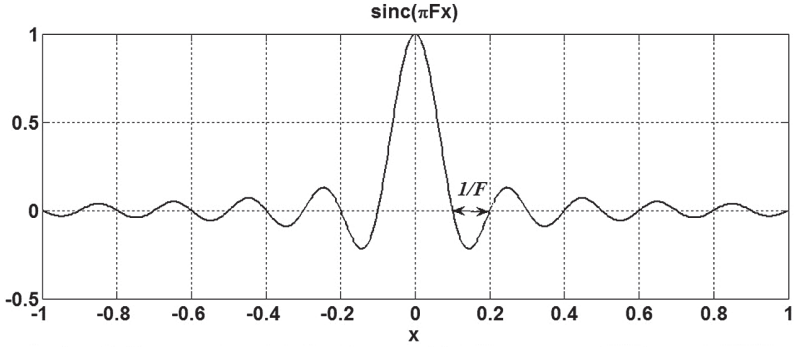


Figure A1.1 Plot of the sinc function $\text{sinc}(40\pi x)$.

$$\sum_{k=-\infty}^{\infty} \delta\left(f - \frac{k}{\Delta_x}\right) = \sum_{m=-\infty}^{\infty} D_m \exp\left(i2\pi f m\Delta_x\right). \tag{A1.45}$$

Coefficients D_m can be found as

$$D_m = \Delta_x \int_{-1/2\Delta_x}^{1/2\Delta_x} \delta(f) \exp(-i2\pi f m\Delta_x) df = \Delta_x. \tag{A1.46}$$

Therefore,

$$\sum_{k=-\infty}^{\infty} \delta\left(f - \frac{k}{\Delta_x}\right) = \Delta_x \sum_{m=-\infty}^{\infty} \exp\left(i2\pi f m\Delta_x\right). \tag{A1.47}$$

This identity is called the *Poisson summation formula*.

Appendix 2
Discrete Fourier Transforms and Their Properties

A2.1 Invertibility of Discrete Fourier Transforms and the Discrete Sinc Function

Consider the general scaled shifted DFT of a signal with samples $\{a_k\}$

$$\alpha_r = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{n=0}^{N-1} a_n \exp\left(i2\pi \frac{\tilde{n}\tilde{r}}{\lceil \sigma N \rceil} \right);\ \tilde{n} = n + u;\ \tilde{r} = r + v \tag{A2.1}$$

and show that its inverse is the transform

$$a_k = \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{r=0}^{\lceil \sigma N \rceil - 1} \alpha_r \exp\left(-i2\pi \frac{\tilde{k}\tilde{r}}{\lceil \sigma N \rceil} \right);\ \tilde{k} = k + u;\ \tilde{r} = r + v. \tag{A2.2}$$

Insert the expression Eq. (A2.1) for $\{\alpha_r\}$ into Eq. (A2.2) to obtain

$$\begin{aligned}
a_k &= \frac{1}{\sqrt{\lceil \sigma N \rceil}} \sum_{r=0}^{\lceil \sigma N \rceil - 1} \alpha_r \exp\left(-i2\pi \frac{\tilde{k}\tilde{r}}{\lceil \sigma N \rceil} \right) \\
&= \frac{1}{\lceil \sigma N \rceil} \sum_{r=0}^{\lceil \sigma N \rceil - 1} \left[\sum_{n=0}^{N-1} a_n \exp\left(i2\pi \frac{\tilde{n}\tilde{r}}{\lceil \sigma N \rceil} \right) \right] \exp\left(-i2\pi \frac{\tilde{k}\tilde{r}}{\lceil \sigma N \rceil} \right) \\
&= \frac{1}{\lceil \sigma N \rceil} \sum_{n=0}^{N-1} a_n \sum_{r=0}^{\lceil \sigma N \rceil - 1} \exp\left[i2\pi \frac{(n-k)}{\lceil \sigma N \rceil} (r + v) \right] \\
&= \frac{\exp\left[i2\pi \frac{(n-k)}{\lceil \sigma N \rceil} v \right]}{\lceil \sigma N \rceil} \sum_{n=0}^{N-1} a_n \frac{\exp[i2\pi(n-k)] - 1}{\exp\left(i2\pi \frac{n-k}{\lceil \sigma N \rceil} \right) - 1}
\end{aligned}$$

$$= \frac{1}{\lceil \sigma N \rceil} \sum_{n=0}^{N-1} a_n \frac{\exp[i\pi(n-k)] - \exp[i\pi(n-k)]}{\exp\left(i\pi \frac{n-k}{\lceil \sigma N \rceil}\right) - \exp\left(i\pi \frac{n-k}{\lceil \sigma N \rceil}\right)} \exp[i\pi(n-k) \frac{2v+N-1}{\lceil \sigma N \rceil}]$$

$$= \sum_{n=0}^{N-1} a_n \frac{\sin[\pi(n-k)]}{N \sin\left(\pi \frac{n-k}{\lceil \sigma N \rceil}\right)} \exp\left[i\pi(n-k) \frac{2v+N-1}{\lceil \sigma N \rceil}\right]$$

$$= \sum_{n=0}^{N-1} a_n \operatorname{sincd}[\lceil \sigma N \rceil;\, \pi(k-n)] \exp\left[i\pi(n-k) \frac{N-1}{N}\right], \tag{A2.3}$$

where

$$\operatorname{sincd}(N;\, x) = \frac{\sin x}{N \sin(x/N)} \tag{A2.4}$$

is the *discrete sincd function*.

For integers n and k,

$$\operatorname{sincd}[\pi(n-k),\, \sigma N] = \bar{\delta}(n-k) = \begin{cases} 1, & k = N \\ 0, & k \neq N. \end{cases} \tag{A2.5}$$

Therefore,

$$\sum_{n=0}^{N-1} a_n \operatorname{sincd}[N;\, \pi(n-k)] \exp\left[i\pi(n-k) \frac{N-1}{N}\right] = a_k. \tag{A2.6}$$

The binary function $\bar{\delta}(\,\cdot\,)$ in Eq. (A2.5) is called the *Kronecker delta*. It is an analog of the Dirac delta function, defined in Appendix A1 (Eq. (A1.35)).

The discrete sincd function is a discrete analog of the continuous sinc function defined by Eq. (A1.38). Both functions are plotted for comparison in Fig. A2.1. The figure shows that the continuous sinc function and discrete sincd function are almost identical within the basic interval of N samples for the discrete sinc function and its corresponding interval $N\Delta x$ for the continuous sinc function. Within this interval, their relative difference does not exceed 10^{-2}; outside this interval, the difference is dramatic. While the sinc function continues to decay, the sincd function $\operatorname{sincd}(N;\, \pi x)$ is a periodic function with period N. The type of periodicity depends on whether N is an odd or even number:

$$\operatorname{sincd}(N;\, \pi(k+gN)) = (-1)^{g(N-1)} \operatorname{sincd}(N;\, \pi k) \tag{A2.7}$$

for any integer number g.

This book uses the following expression as a reference for the discrete sinc function:

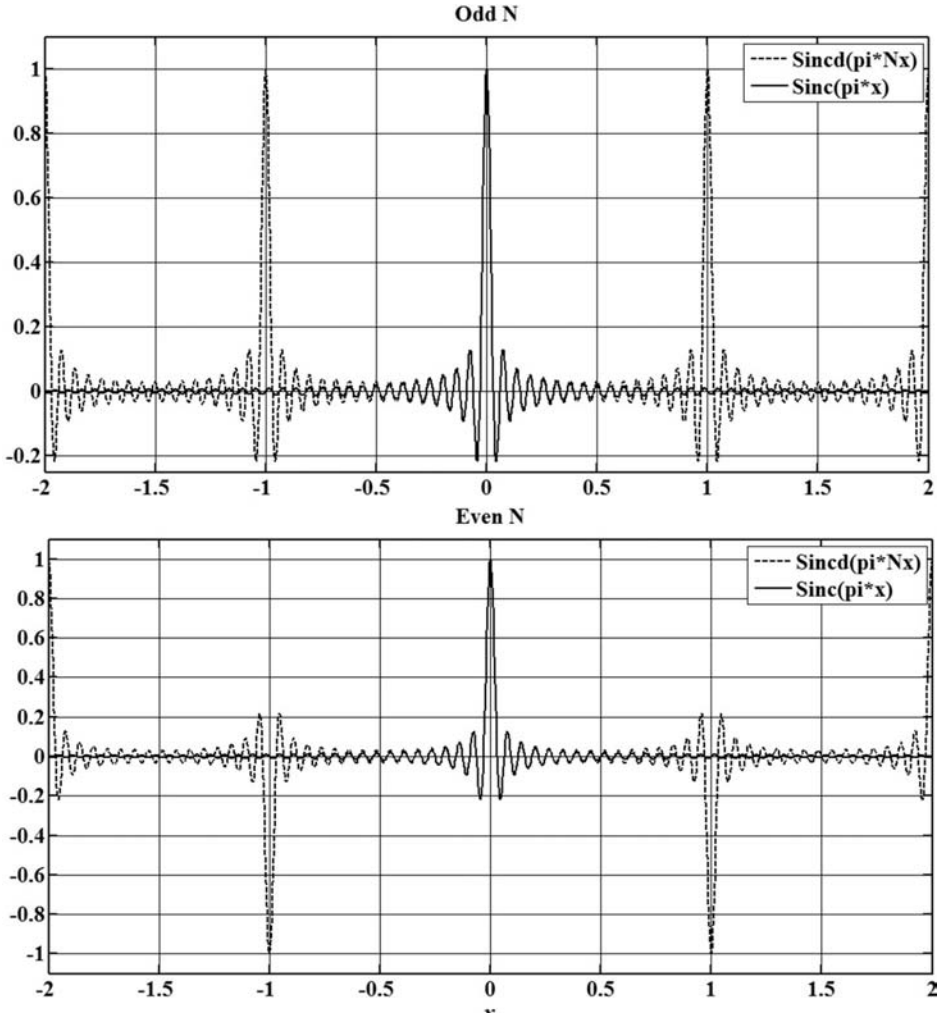


Figure A2.1 Discrete sinc (dashed line) and continuous sinc (solid line) functions for an odd (upper plot) and even (bottom plot) number of samples N within the basic sampling interval $N\Delta_x$.

$$\begin{aligned}
&\frac{1}{N}\sum_{r=0}^{N-1}\exp\left(i2\pi\frac{kr}{N}\right)\\
&\quad=\frac{\exp(i2\pi k)-1}{\exp\left(i2\pi\frac{kr}{N}\right)-1}=\frac{1}{N}\frac{\exp(i\pi k)-\exp(-i\pi k)}{\exp\left(i\pi\frac{k}{N}\right)-\exp\left(-i\pi\frac{k}{N}\right)}\exp\left(-i\pi\frac{N-1}{N}k\right)\\
&\quad=\frac{\sin(\pi k)}{N\sin(\pi k/N)}\exp\left(-i\pi\frac{N-1}{N}k\right)=\operatorname{sincd}(N;\,\pi k)\exp\left(-i\pi\frac{N-1}{N}k\right).
\end{aligned}\tag{A2.8}$$

A2.2 Parseval's Relation for the DFT

The invertibility of the DFT implies that it is an orthonormal transform, and as such it satisfies the Parseval's relation:

$$\sum_{k=0}^{N-1} |a_k|^2 = \sum_{r=0}^{N-1} |\alpha_r|^2. \tag{A2.9}$$

Proof:

$$\begin{aligned}\sum_{k=0}^{N-1} |a_k|^2 &= \sum_{k=0}^{N-1} a_k a_k^* = \sum_{k=0}^{N-1} \left[\frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \alpha_r \exp\left(-i2\pi \frac{kr}{N}\right)\right] a_k^* \\ &= \sum_{r=0}^{N-1} \alpha_r \left[\frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k^* \exp\left(-i2\pi \frac{kr}{N}\right)\right] = \sum_{r=0}^{N-1} \alpha_r (\alpha_r)^* = \sum_{r=0}^{N-1} |\alpha_r|^2,\end{aligned} \tag{A2.10}$$

where the asterisk denotes complex conjugacy.

A2.3 Cyclicity of the DFT

Based on the definitions of the direct and inverse DFT, it follows that for any integer g

$$\alpha_{r+gN} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi k \frac{r+gN}{N}\right) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{r}{N}\right) = \alpha_{(r) \bmod N} \tag{A2.11}$$

and

$$\begin{aligned}a_{k+gN} &= \frac{1}{\sqrt{N}} \sum_{r=0}^{\sigma N-1} \alpha_r \exp\left(-i2\pi \frac{k+gN}{N} r\right) = \frac{1}{\sqrt{N}} \sum_{r=0}^{\sigma N-1} \alpha_r \exp\left(-i2\pi \frac{kr}{N}\right) \\ &= a_{(k) \bmod N},\end{aligned} \tag{A2.12}$$

where $(\cdot) \bmod N$ denotes residual from division of the variable by N.

Owing to the separability of the DFTs, the same cyclicity feature holds for 2D DFTs. The cyclicity is the main distinction of the DTF from the integral Fourier transform it represents.

A2.4 Shift Theorem

The shift theorem for the DFT is analogous to that for the integral Fourier transform: the absolute value of the signal ShDFT spectrum is invariant to the

signal shift; signal shift causes only linear modulation of the phase of the spectrum:

$$\begin{aligned} a_{k+k_0} &= \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \alpha_r \exp\left(-i2\pi \frac{k+k_0}{N} r\right) \\ &= \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \left\{ \alpha_r \exp\left(-i2\pi \frac{k_0 r}{N}\right) \right\} \exp\left(-i2\pi \frac{kr}{N}\right). \end{aligned} \tag{A2.13}$$

Similarly, the shifted signal spectrum $\{\alpha_{r+r_0}\}$ corresponds to the phase-modulated signal

$$\begin{aligned} \alpha_{r+r_0} &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi k \frac{r+r_0}{N}\right) \\ &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \left\{ a_k \exp\left(i2\pi \frac{kr_0}{N}\right) \right\} \exp\left(i2\pi \frac{kr}{N}\right). \end{aligned} \tag{A2.14}$$

A2.5 Convolution Theorem

The convolution theorem for the DFT is an analog of the convolution theorem for the integral Fourier transform (see Appendix A1): a product of the DFT spectra of two signals is the DFT spectrum of the result of signal convolution. An important difference is that the convolution computed through the DFT is cyclic.

Let $\{a_k\}$ and $\{b_k\}$ be two signals of N samples, and $\{\alpha_r\}$ and $\{\beta_r\}$ are their corresponding DFT spectra:

$$\alpha_r = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_n \exp\left(i2\pi \frac{nr}{N}\right);\ \beta_r = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} b_n \exp\left(i2\pi \frac{nr}{N}\right). \tag{A2.15}$$

Find signal $\{c_k\}$, whose spectrum is a product $\{\alpha_r \beta_r\}$ of spectra $\{\alpha_r\}$ and $\{\beta_r\}$:

$$\begin{aligned} c_k &= \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \alpha_r \beta_r \exp\left(-i2\pi \frac{kr}{N}\right) \\ &= \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \left[\frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} a_n \exp\left(i2\pi \frac{nr}{N}\right) \right] \beta_r \exp\left(-i2\pi \frac{kr}{N}\right) \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} a_n \left[\frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \beta_r \exp\left(i2\pi \frac{n-k}{N} r\right) \right] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} a_n b_{n-k}, \end{aligned} \tag{A2.16}$$

which, by virtue of the cyclicity of signals, is a cyclic convolution

$$c_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} a_{(n)\mathrm{mod}} b_{(n-k)\mathrm{mod}N}. \tag{A2.17}$$

This convolution theorem can be reformulated for image correlation: the DFT spectrum of the cross-correlation between two images

$$c_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} a_{(n)\mathrm{mod}} b_{(k-n)\mathrm{mod}N} \tag{A2.18}$$

equals the product $\{\alpha_r \beta_r^*\}$ of the DFT spectrum of the first image and the complex conjugate to the DFT spectrum of the second image.

A2.6 Symmetry Properties

The definition of the direct and inverse canonic DFTs verifies that for signal $\{a_k\}$ and its DFT spectrum $\{\alpha_r\}$ the following relationships hold:

$$\{a_{N-k}\} \overset{DFT}{\rightarrow} \{\alpha_{N-r}\};\ \{a_k^*\} \overset{DFT}{\rightarrow} \{\alpha_{N-r}^*\}, \tag{A2.19}$$

$$\{a_k = a_{N-k}\} \overset{DFT}{\rightarrow} \{\alpha_r = \alpha_{N-r}\}. \tag{A2.20}$$

A reversal of the signal and its DFT spectrum indices corresponds to a reversal of the sign of the signal coordinate and frequency for the integral Fourier transform. However, this analogy is not complete: the signal and its DFT spectrum samples a_0 and α_0 for even N ($a_{N/2}$ and $\alpha_{N/2}$) are not inverted, and index $N/2$ (which for odd N is virtual) plays a role in the symmetry center, as do points $x = 0$ and $f = 0$ for the integral Fourier transform.

For real-valued signals:

$$\{a_k = a_k^*\} \overset{DFT}{\rightarrow} \{\alpha_r = \alpha_{N-r}^*\};\ \alpha_0 = \alpha_0^*. \tag{A2.21}$$

For real-valued signals with an even number N of signal samples:

$$\alpha_{N/2} = \alpha_{N/2}^*, \tag{A2.22}$$

$$\{a_k = a_k^* = \pm a_{N-k}\} \overset{DFT}{\rightarrow} \{\alpha_r = \pm\alpha_{N-r}^* = \pm\alpha_{N-r}\}. \tag{A2.23}$$

The shifted direct and inverse DFTs with semi-integer shift parameters ($u = 1/2$, $v = 1/2$),

$$\alpha_r^{(1/2,1/2)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left[i2\pi \frac{(k+1/2)(r+1/2)}{N} \right] \tag{A2.24}$$

and

$$a_k^{(1/2,1/2)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \alpha_r^{(1/2,1/2)} \exp\left[-i2\pi \frac{(k+1/2)(r+1/2)}{N} \right], \tag{A2.25}$$

exhibit more perfect symmetry with no exceptions:

$$\{a_{N-1-k}\} \xrightarrow{ShDFT(u=1/2,v=1/2)} \left\{ \alpha_{N-1-r}^{(1/2,1/2)} \right\}, \tag{A2.26}$$

$$\{a_k = \pm a_{N-1-k}\} \xrightarrow{ShDFT(u=1/2,v=1/2)} \left\{ \alpha_r^{(1/2,1/2)} = \pm \alpha_{N-1-r}^{(1/2,1/2)} \right\}, \tag{A2.27}$$

$$\{a_k = \pm a_k^* = \pm a_{N-1-k}\} \xrightarrow{ShDFT(u=1/2,v=1/2)} \left\{ \alpha_r^{(1/2,1/2)} = \mp \alpha_{N-1-r}^{*(1/2,1/2)} = \pm \alpha_{N-1-r}^{(1/2,1/2)} \right\}. \tag{A2.28}$$

For the shifted DFT with shift parametrs $(1/2, 0)$, the following properties are useful:

$$\{a_k^*\} \xrightarrow{SDFT(u,0)} \left\{ -\alpha_{N-r}^{*(1/2,0)} \right\}, \tag{A2.29}$$

$$\{a_k = a_k^*\} \xrightarrow{SDFT(u,0)} \left\{ \alpha_r^{(1/2,0)} = -\alpha_{N-r}^{(1/2,0)} \right\}. \tag{A2.30}$$

A2.7 SDFT Spectra of Sinusoidal Signals

Find the shifted DFT (u, v) spectrum

$$\alpha_r^{(u,v)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left[i2\pi \frac{(k+u)(r+v)}{N} \right] \tag{A2.31}$$

of a sinusoidal signal

$$\left\{ a_k = \cos\left(2\pi \frac{\omega k}{N} + \varphi \right) \right\}. \tag{A2.32}$$

With $\tilde{r} = r + v$, obtain

$$\begin{aligned}\alpha_r^{(u,v)} &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\cos\left(2\pi\frac{\omega k}{N}+\varphi\right)\exp\left[i2\pi\frac{(k+u)(r+v)}{N}\right]\\ &= \frac{1}{2\sqrt{N}}\sum_{k=0}^{N-1}\left[\exp\left(i2\pi\frac{\omega k}{N}+i\varphi\right)+\exp\left(-i2\pi\frac{\omega k}{N}-i\varphi\right)\right]\exp\left[i2\pi\frac{(k+u)\tilde{r}}{N}\right]\\ &= \frac{\exp\left(i2\pi\frac{u\tilde{r}}{N}+i\varphi\right)}{2\sqrt{N}}\sum_{k=0}^{N-1}\exp\left[i2\pi\frac{k(\tilde{r}+\omega)}{N}\right]\\ &\quad+\frac{\exp\left(i2\pi\frac{u\tilde{r}}{N}-i\varphi\right)}{2\sqrt{N}}\sum_{k=0}^{N-1}\exp\left[i2\pi\frac{k(\tilde{r}-\omega)}{N}\right]=(\Sigma)_1+(\Sigma)_2.\end{aligned} \tag{A2.33}$$

First, compute the term $(\Sigma)_1$:

$$\begin{aligned}(\Sigma)_1 &= \frac{\exp\left(i2\pi\frac{u\tilde{r}}{N}+i\varphi\right)}{2\sqrt{N}}\sum_{k=0}^{N-1}\exp\left[i2\pi\frac{k(\tilde{r}+\omega)}{N}\right]\\ &= \frac{\exp\left(i2\pi\frac{u\tilde{r}}{N}+i\varphi\right)}{2\sqrt{N}}\frac{\exp[i2\pi(\tilde{r}+\omega)]-1}{\exp\left[i2\pi\frac{(\tilde{r}+\omega)}{N}\right]-1}\\ &= \frac{\exp\left(i2\pi\frac{u\tilde{r}}{N}+i\varphi\right)}{2\sqrt{N}}\frac{\exp[i\pi(\tilde{r}+\omega)]-\exp[-i\pi(\tilde{r}+\omega)]}{\exp\left[i\pi\frac{(\tilde{r}+\omega)}{N}\right]-\exp\left[i\pi\frac{(\tilde{r}+\omega)}{N}\right]}\exp\left[i\pi\frac{(N-1)(\tilde{r}+\omega)}{N}\right]\\ &= \frac{\sqrt{N}}{2}\exp\left[i\pi\left(\frac{2u+N-1}{N}\tilde{r}+\frac{N-1}{N}\omega\right)+i\varphi\right]\frac{\sin[\pi(\tilde{r}+\omega)]}{N\sin\left[\pi\frac{(\tilde{r}+\omega)}{N}\right]}.\end{aligned} \tag{A2.34}$$

Correspondingly,

$$(\Sigma)_2 = \frac{\sqrt{N}}{2}\exp\left[i\pi\left(\frac{2u+N-1}{N}\tilde{r}-\frac{N-1}{N}\omega\right)-i\varphi\right]\mathrm{sincd}[N;\ \pi(r+v-\omega)], \tag{A2.35}$$

and therefore,

$$\alpha_r^{(u,v)} = \frac{\sqrt{N}}{2}\exp\left[i\pi\left(\frac{2u+N-1}{N}\tilde{r}+\frac{N-1}{N}\omega\right)+i\varphi\right]\mathrm{sincd}[N;\,\pi(r+v+\omega)] + \frac{\sqrt{N}}{2}\exp\left[i\pi\left(\frac{2u+N-1}{N}\tilde{r}-\frac{N-1}{N}\omega\right)-i\varphi\right]\mathrm{sincd}[N;\,\pi(r+v-\omega)]. \tag{A2.36}$$

If the shift parameter $u = -(N-1)/2$ and $v = 0$, then

$$\alpha_r^{(0,v)} = \frac{\sqrt{N}}{2}\exp\left[i\pi\left(\frac{N-1}{N}\omega+\frac{\varphi}{\pi}\right)\right]\mathrm{sincd}[\pi(r+\omega)] + \frac{\sqrt{N}}{2}\exp\left[-i\pi\left(\frac{N-1}{N}\omega+\frac{\varphi}{\pi}\right)\right]\mathrm{sincd}[\pi(r-\omega)]. \tag{A2.37}$$

Equations (A2.36) and (A2.37) imply that the ShDFT specrum of the sinusoidal signal has sharp maxima in the vicinity of samples with indices $r = \pm\omega$. This opens the possibility of determining the frequency of sinusoidal signals with sub-sample accuracy (see Section A2.12).

For integer ω and $v = 0$, the discrete sincd functions in Eq. (A2.37) are reduced to two Kronecker deltas:

$$\alpha_r^{(0,0)} = \frac{\sqrt{N}}{2}\exp\left[i\pi\left(\frac{(N-1)}{N}\omega+\frac{\varphi}{\pi}\right)\right]\bar{\delta}(r+\omega) + \frac{\sqrt{N}}{2}\exp\left[i\pi\left(-\frac{(N-1)}{N}\omega+\frac{\varphi}{\pi}\right)\right]\delta(r-\omega), \tag{A2.38}$$

which is in complete agreement with the continuous Fourier spectrum.

Examples of the DFT spectra of sinusoidal signals with an integer and non-integer frequency parameter ω are presented in Fig. A2.2. Mutual correspondence between the frequency indices $\{r\}$ and frequencies of sinusoidal signals is discussed in Section A2.8.

A2.8 Mutual Correspondence between the Indices of ShDFT Spectral Coefficients and Signal Frequencies

From the derivation of the ShDFTs in Section 10.1, it follows that the index $\tilde{r} = r + v$ of the SDFT spectral coefficients is linked with the frequency f of the signal Fourier spectra by the following relationship:

$$f = \tilde{r}\Delta_f = \tilde{r}/N\Delta_x, \tag{A2.39}$$

where Δ_x and Δ_f are the sampling intervals of the signal and its spectra, respectively.

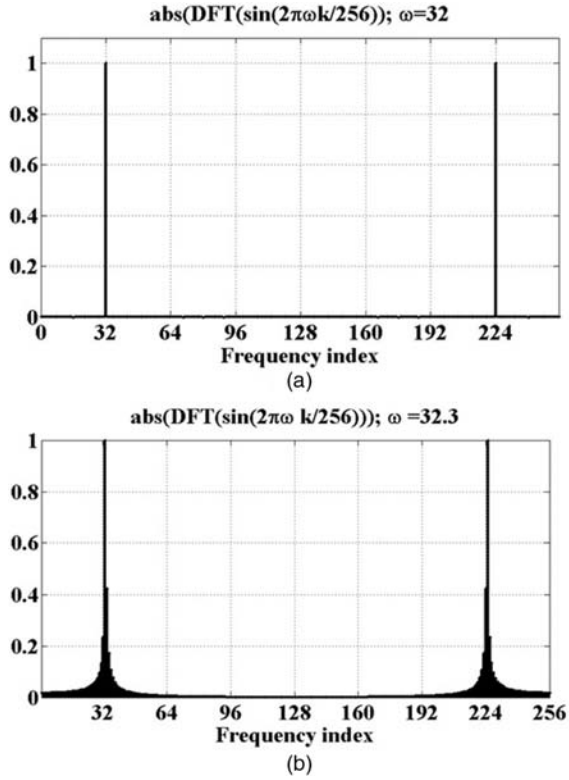


Figure A2.2 DFT spectra of sinusoidal signals with an (a) integer and (b) non-integer frequency parameter ω.

The canonic DFT spectral coefficient $\alpha_0^{(0,0)}$ corresponds to the zero-frequency component of the integral Fourier transform. It is proportional to the signal mean value, or its *dc component*:

$$\alpha_0 = \sqrt{N}\left(\frac{1}{N}\sum_{k=0}^{N-1} a_k\right). \tag{A2.40}$$

For even N, the coefficient DFT $\alpha_{N/2}^{(0,0)}$

$$\alpha_{N/2} = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}(-1)^k a_k \tag{A2.41}$$

represents the highest-frequency signal component $f_{\max} = \frac{N}{2}\Delta f = 1/2\Delta x$.

For odd N, coefficients $\alpha_{(N-1)/2} = \alpha^*_{(N+1)/2}$ (for signals with real values) represent the highest signal frequency $f = \frac{N-1}{2}\Delta f = (N-1)/2N\Delta x$.

In order to maintain mutual correspondence between the DFT spectrum $\{\alpha_r\}$ of a discrete signal $\{a_k\}$ and Fourier spectrum $\alpha(f), f \in [-\infty, \infty]$ of the corresponding continuous signal $a(x)$, it is convenient to cyclically shift the sequence $\{\alpha_r\}$ to place the dc-component coefficient $\{\alpha_0\}$ in the middle of the sequence. For even and odd N, the shifts are $N/2$ and $(N-1)/2$, respectively, and the spectral coefficients should be taken in the following order: for even N, $[\alpha_{N/2}, \alpha_{N/2+1}, \ldots, \alpha_{N-1}, \alpha_0, \alpha_1, \ldots, \alpha_{N/2-1}]$, and for odd N, $[\alpha_{(N+1)/2}, \alpha_{(N+1)/2+1}, \ldots, \alpha_{N-1}, \alpha_0, \alpha_1, \ldots, \alpha_{(N-1)/2}]$.

For 2D signals of $N_1 \times N_2$ samples, these cyclical shifts should be taken along corresponding two indices. In MATLAB, a special function **fftshift** performs this centering cyclic shift. Figures A2.3 and A2.4 illustrate spectral coefficient reordering for the DFT spectra of 1D and 2D sinusoidal signals.

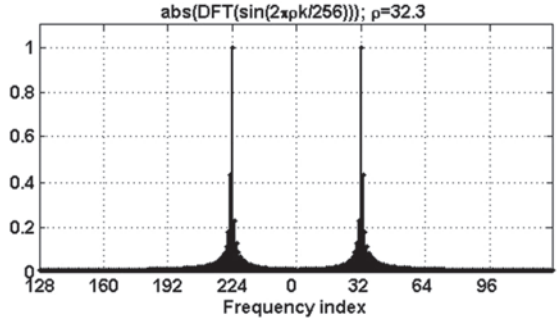


Figure A2.3 Spectrum of a sinusoidal signal shown in Fig. A2.2(a), reordered by the MATLAB command fftshift for spectrum centering.

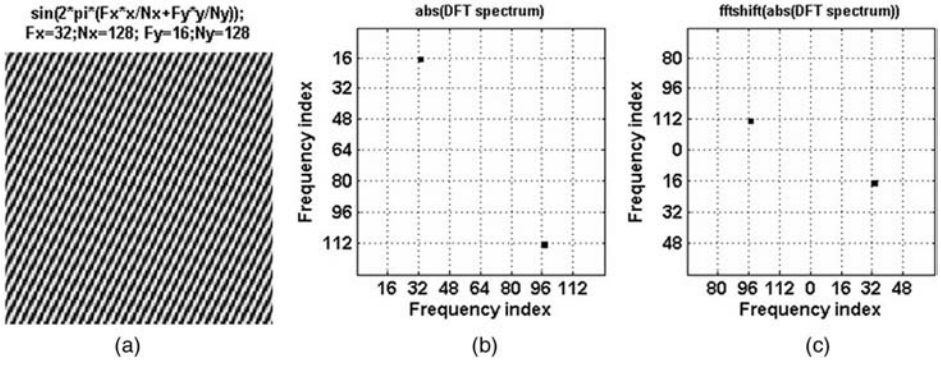


Figure A2.4 (a) A 2D sinusoidal signal, and its DFT spectrum represented as an image (b) in the original index order and (c) in the fftshifted index order, which correspond to the analog Fourier spectrum coordinate system centered at point $(f_x = 0; f_y = 0)$. Two small dark boxes in images (b) and (c) indicate the position of signal spectral components.

A2.9 DFT Spectra of Sparse Signals and Spectral Zero-Padding

Let a discrete signal $\{\ddot{a}_n\}$ of LN samples ($\tilde{n} = 0, 1, \ldots, LN-1$) be obtained from signal $\{a_k\}$ of N samples ($k = 0, 1, \ldots N-1$) by placing between its samples $(L-1)$ zero samples. Represent index n as a two-component index:

$$\tilde{n} = kL + l,\ \kappa = 0, 1, \ldots, N-1,\ l = 0, 1, \ldots, L-1. \tag{A2.42}$$

The signal $\{\ddot{a}_k\}$ with sparse samples can then be represented as

$$\ddot{a}_{\tilde{n}} = a_k \bar{\delta}(l), \tag{A2.43}$$

where $\bar{\delta}(\cdot)$ is the Kronecker delta. Compute the DFT of this signal

$$\begin{aligned}\ddot{\alpha}_{\tilde{r}} &= \frac{1}{\sqrt{LN}} \sum_{\tilde{n}=0}^{LN-1} \ddot{a}_{\tilde{n}} \exp\left(i2\pi \frac{\tilde{n}\tilde{r}}{LN}\right) = \frac{1}{\sqrt{LN}} \sum_{l=0}^{L-1} \sum_{k=0}^{N-1} a_k \delta(l) \exp\left(i2\pi \frac{kL+l}{LN}\tilde{r}\right) \\ &= \frac{1}{\sqrt{LN}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{k\tilde{r}}{N}\right) = \frac{1}{\sqrt{L}} \alpha_{(r)\mathrm{mod}N},\ \tilde{r} = 0, 1, \ldots, LN-1,\end{aligned} \tag{A2.44}$$

where $\{\alpha_{(r)\mathrm{mod}N}\}$ is the DFT of signal $\{a_k\}$. Equation (A2.44) shows that placing zeros between signal samples results in periodic replication of the DFT spectrum with the number of replicas equal to the number of zeros plus one. This property of DFT spectra is an analog of virtual spectrum replication in sampling continuous signals discussed with the sampling theorem in Chapter 2 (illustrated in Fig. A2.5).

Now zero all periods in the spectrum $\{\ddot{\alpha}_{\tilde{r}}\}$ but the initial one, as shown in Fig. A2.6(a). This operation is called *spectral zero-padding*. Spectral symmetry $\{\alpha_{\tilde{r}} = \alpha^*_{LN-\tilde{r}}\}$ should be maintained for real-valued signals (Eq. (A2.21)). Thus, the method used for the zero-padding depends on whether N is odd or even.

For odd N, spectral coefficients with indices from $\tilde{r} = (N+1)/2$ to $\tilde{r} = LN - (N+1)/2$ should be set to zero. In this case, the inverse DFT of the zero-padded spectrum of LN samples produces a signal

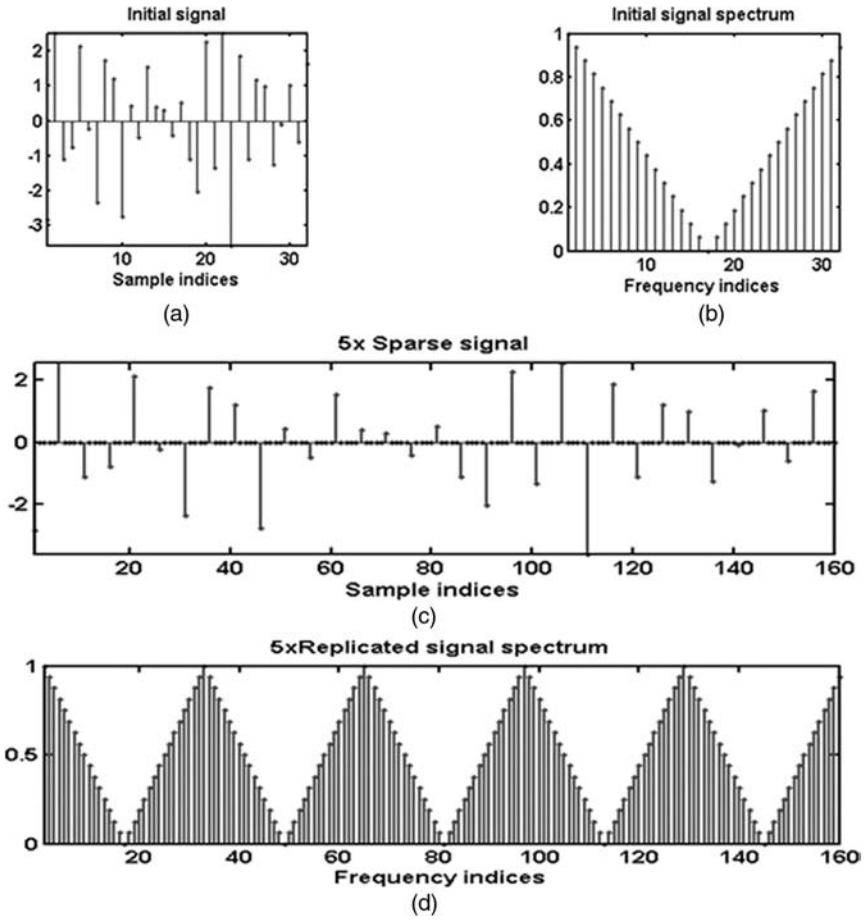


Figure A2.5 (a) A test signal; (b) its DFT spectrum; (c) a sparse signal obtained from signal (a) by placing four zeros between its samples; and (d) the DFT spectrum of the obtained sparse signal.

$$\begin{aligned}\widetilde{a}_{\tilde{k}} &= \frac{1}{\sqrt{LN}}\left[\sum_{\tilde{r}=0}^{(N-1)/2}\alpha_{(\tilde{r})\bmod N}\exp\left(-i2\pi\frac{\tilde{k}\,\tilde{r}}{LN}\right)\right.\\ &\quad\left.+\sum_{\tilde{r}=LN-(N-1)/2}^{LN-1}\alpha_{(\tilde{r})\bmod N}\exp\left(-i2\pi\frac{\tilde{k}\,\tilde{r}}{LN}\right)\right]\\ &= \frac{1}{\sqrt{LN}}\left\{\sum_{\tilde{r}=0}^{(N-1)/2}\left[\frac{1}{\sqrt{N}}\sum_{n=0}^{N-1}a_n\exp\left(i2\pi\frac{n\tilde{r}}{N}\right)\right]\exp\left(-i2\pi\frac{\tilde{k}\,\tilde{r}}{LN}\right)\right.\\ &\quad\left.+\sum_{\tilde{r}=LN-(N-1)/2}^{LN-1}\left[\frac{1}{\sqrt{N}}\sum_{n=0}^{N-1}a_n\exp\left(i2\pi\frac{n\tilde{r}}{N}\right)\right]\exp\left(-i2\pi\frac{\tilde{k}\,\tilde{r}}{LN}\right)\right\}\end{aligned}$$

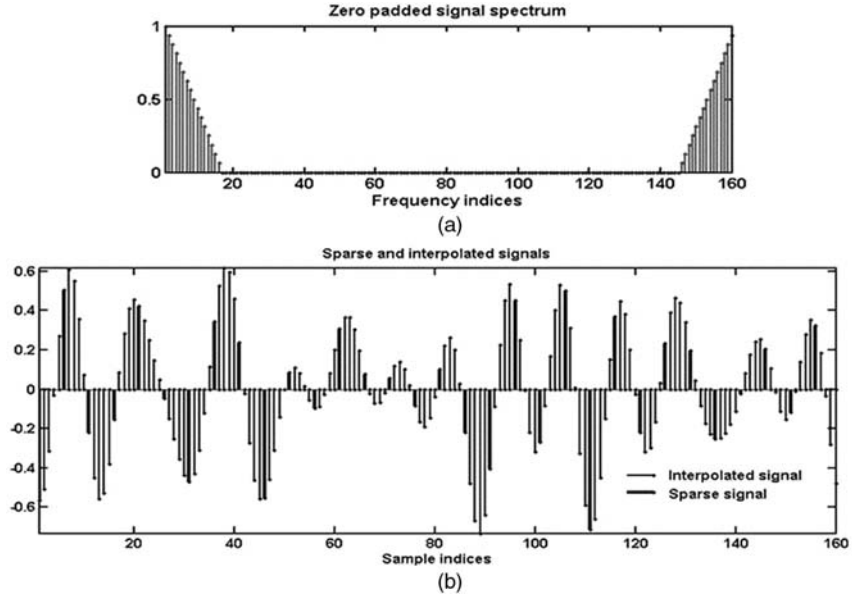


Figure A2.6 Zero-padded spectrum of the sparse signal after removing all its periodic components but one (a) and sincd-interpolated signal reconstructed from the zero-padded spectrum (b).

$$
\begin{aligned}
&= \frac{1}{N\sqrt{L}} \left\{ \sum_{n=0}^{N-1} a_n \left[\sum_{\tilde{r}=0}^{(N-1)/2} \exp\left(-i2\pi \frac{\tilde{k} - Ln}{LN} \tilde{r} \right) \right.\right. \\
&\quad \left.\left. + \sum_{\tilde{r}=LN-(N-1)/2}^{LN-1} \exp\left(-i2\pi \frac{\tilde{k} - Ln}{LN} \tilde{r} \right) \right] \right\} \\
&= \frac{1}{N\sqrt{L}} \left(\sum_{n=0}^{N-1} a_n \left\{ \frac{\exp\left[-i2\pi \frac{(\tilde{k}-Ln)(N+1)}{2LN} \right] - 1}{\exp\left(-i2\pi \frac{\tilde{k}-Ln}{LN} \right) - 1} \right.\right. \\
&\quad \left.\left. + \frac{\exp\left[-i2\pi \frac{LN(\tilde{k}-Ln)}{LN} \right] - \exp\left[-i2\pi \frac{(\tilde{k}-Ln)(LN-(N-1)/2)}{LN} \right]}{\exp\left(-i2\pi \frac{\tilde{k}-Ln}{LN} \right) - 1} \right\} \right) \qquad \text{(A2.45)} \\
&= \frac{1}{N\sqrt{L}} \sum_{n=0}^{N-1} a_n \frac{\exp\left[-i2\pi \frac{(\tilde{k}-Ln)(N+1)}{2LN} \right] - \exp\left[-i2\pi \frac{(\tilde{k}-Ln)(N-1)}{2LN} \right]}{\exp\left(-i2\pi \frac{\tilde{k}-Ln}{LN} \right) - 1} \\
&= \frac{1}{\sqrt{L}} \sum_{n=0}^{N-1} a_n \frac{\sin\left(\pi N \frac{\tilde{k}-Ln}{LN} \right)}{N \sin\left(\pi \frac{\tilde{k}-Ln}{LN} \right)} = \frac{1}{\sqrt{L}} \sum_{n=0}^{N-1} a_n \operatorname{sincd}[N;\, \pi(\tilde{k} - nL)/L].
\end{aligned}
$$

Thus, for an odd number N of signal samples, spectrum padding with $(L-1)N$ zeros results in a signal

$$\widetilde{a}_{Lk+l} = \frac{1}{\sqrt{L}} \sum_{n=0}^{N-1} a_n \text{sincd}\left[N;\, \pi\left(k - n + \frac{l}{L}\right)\right], \qquad k = 0,\, 1, \ldots,\, N-1,\; l = 0,\, 1, \ldots,\, L-1 \tag{A2.46}$$

in which each Lnth sample equals $1/\sqrt{L}$ of the nth sample of the initial signal, and the rest of the samples are discrete sinc interpolated from those initial samples.

For even N, the $N/2$th sample has no symmetrical counterpart in the spectrum's complex conjugated symmetry $\{\alpha_{\tilde{r}} = \alpha^*_{LN-\tilde{r}}\}$ in spectral zero-padding, and there are two options to maintain this symmertry:

– Case 0: zero components from $r = N/2$ to $r = LN - N/2$, which means zeroing the highest-frequency component;
– Case 2: zero components from $r = N/2 + 1$ to $r = LN - 1 - N/2$, which means duplicating the highest-frequency component.

Derivations similar to that of Eq. (A2.45) can be used to obtain the first case of a spectral-zero-padded signal reconstructed by the inverse DFT of the zero-padded spectrum of signal $\{a_k\}$, defined by the equation

$$\widetilde{a}^{(0)}_{kL+l} = \frac{1}{\sqrt{L}} \sum_{n=0}^{N-1} a_n \text{sincdd}\left[N-1;\, N;\, \pi\left(k - n + \frac{l}{L}\right)\right], \tag{A2.47}$$

and in the second case by

$$\widetilde{a}^{(2)}_{kL+l} = \frac{1}{\sqrt{L}} \sum_{n_1=0}^{N-1} a_n \text{sincdd}\left[N+1;\, N;\, \pi\left(k - n + \frac{l}{L}\right)\right], \tag{A2.48}$$

where $\text{sincdd}(M;\, N;\, x)$ is the *sincdd function* defined as

$$\text{sincdd}[M;\, N;\, x] = \frac{\sin(Mx/N)}{\sin(x/N)}. \tag{A2.49}$$

Equations (A2.46), (A2.47), and (A2.48) are discrete analogs of the reconstruction of continuous signals from their samples, discussed in Chapter 2.

Of practical importance is an intermediate option: halving and duplicating the halved highest-frequency $N/2$th component. We will call this option Case 1. In this case, the reconstructed signal is

$$\widetilde{a}^{(1)}_{kL+l} = \frac{\widetilde{a}^{(0)}_{kL+l} + \widetilde{a}^{(2)}_{kL+l}}{2} = \frac{1}{\sqrt{L}} \sum_{n=0}^{N-1} a_n \mathrm{sincdd}\left(\pm 1;\, N;\, \pi\left(k - n + \frac{l}{L}\right)\right), \tag{A2.50}$$

where

$$\begin{aligned} \mathrm{sincdd}[\pm 1;\, N;\, x] &= \{\mathrm{sincdd}[N-1;\, N;\, x] + \mathrm{sincdd}[N+1;\, N;\, x]\}/2 \\ &= \frac{\sin\left(\frac{N-1}{N}x\right) + \sin\left(\frac{N+1}{N}x\right)}{2N\,\sin\left(\frac{x}{N}\right)} = \frac{\sin(x)}{N\sin(x/N)}\cos(x/N) \\ &= \cos(x/N)\mathrm{sincd}[N;\, x] \end{aligned} \tag{A2.51}$$

Note that, generally, the length of the zero-padded spectrum and, correspondingly, the size of its corresponding interpolated signal, should not necessarily be multiple (LN_0) the size of the spectrum baseband N_0. In fact, it can be arbitrary, and, therefore, the subsampling factor N/N_0 can be an arbitrary rational number.

Consider, for instance, this arbitrary zero-padding of a signal with an odd number of samples. Let $\{a_n\}$ be a signal of N_0 samples, N_0 be an odd number, and

$$\alpha_r = \frac{1}{\sqrt{N_0}} \sum_{n=0}^{N_0-1} a_n \exp\left(i2\pi \frac{nr}{N_0}\right) \tag{A2.52}$$

be the signal DFF spectrum. Pad this spectrum to length N with $N - N_0$ zeros in the following way:

$$\widetilde{\alpha}_r = \begin{cases} \alpha_r, & r = 0 : (N_0 - 1)/2 \\ 0, & r = (N_0 + 1)/2 : N - (N_0 + 1)/2) \\ \alpha_{r-N+N_0}, & r = N - (N_0 - 1)/2 : N - 1) \end{cases} \tag{A2.53}$$

and compute the inverse Fourier transform of this zero-padded spectrum:

$$\begin{aligned}
\widetilde{a}_k &= \frac{1}{\sqrt{N}}\sum_{r=0}^{N-1}\widetilde{\alpha}_r \exp\left(-i2\pi\frac{kr}{N}\right) \\
&= \frac{1}{\sqrt{N}}\sum_{r=0}^{(N_0-1)/2}\widetilde{\alpha}_r \exp\left(-i2\pi\frac{kr}{N}\right) + \frac{1}{\sqrt{N}}\sum_{r=N-(N_0-1)/2}^{N-1}\widetilde{\alpha}_r \exp\left(-i2\pi\frac{kr}{N}\right) \\
&= \frac{1}{\sqrt{N}}\sum_{r=0}^{(N_0-1)/2}\widetilde{\alpha}_r \exp\left(-i2\pi\frac{kr}{N}\right) + \frac{1}{\sqrt{N}}\sum_{r=N-(N_0-1)/2}^{N-1}\widetilde{\alpha}_{r-N+N_0} \exp\left(-i2\pi\frac{kr}{N}\right) \\
&= \frac{1}{\sqrt{N}}\sum_{r=0}^{(N_0-1)/2}\alpha_r \exp\left(-i2\pi\frac{kr}{N}\right) + \frac{1}{\sqrt{N}}\sum_{r=.(N_0+1)/2}^{N_0-1}\alpha_r \exp\left(-i2\pi\frac{k(r+N-N_0)}{N}\right).
\end{aligned} \tag{A2.54}$$

Now replace the spectral coefficients $\{\alpha_r\}$ in Eq. (A2.54) with their expression through signal samples $\{a_n\}$ (Eq. (A2.52)) and change the order of summation:

$$\begin{aligned}
\widetilde{a}_k &= \frac{1}{\sqrt{NN_0}}\sum_{r=0}^{(N_0-1)/2}\sum_{n=0}^{N_0-1} a_n \exp\left(i2\pi\frac{nr}{N_0}\right)\exp\left(-i2\pi\frac{kr}{N}\right) \\
&\quad + \frac{1}{\sqrt{NN_0}}\sum_{r=.(N_0+1)/2}^{N_0-1}\sum_{n=0}^{N_0-1} a_n \exp\left(i2\pi\frac{nr}{N_0}\right)\exp\left(-i2\pi\frac{k(r-N_0)}{N}\right) \\
&= \frac{1}{\sqrt{NN_0}}\sum_{n=0}^{N_0-1} a_n \sum_{r=0}^{(N_0-1)/2}\exp\left(i2\pi\frac{nr}{N_0}\right)\exp\left(-i2\pi\frac{kr}{N}\right) \\
&\quad + \frac{1}{\sqrt{NN_0}}\sum_{n=0}^{N_0-1} a_n \sum_{r=.(N_0+1)/2}^{N_0-1}\exp\left(i2\pi\frac{nr}{N_0}\right)\exp\left(-i2\pi\frac{k(r-N_0)}{N}\right) \\
&= \frac{1}{\sqrt{NN_0}}\sum_{n=0}^{N_0-1} a_n \left\{\sum_{r=0}^{(N_0-1)/2}\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)r\right]\right. \\
&\quad \left. + \exp\left(i2\pi\frac{kN_0}{N}\right)\sum_{r=.(N_0+1)/2}^{N_0-1}\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)r\right]\right\} \\
&= \frac{1}{\sqrt{NN_0}}\left\{(\Sigma)_1 + \exp\left(i2\pi\frac{kN_0}{N}\right)(\Sigma)_2\right\}.
\end{aligned} \tag{A2.55}$$

Compute the sums in figure brackets:

$$(\Sigma)_1 = \sum_{r=0}^{(N_0-1)/2} \exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)r\right] = \frac{\exp\left[i\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0+1)\right]-1}{\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]-1}; \tag{A2.56}$$

$$\begin{aligned}\exp\left(i2\pi\frac{kN_0}{N}\right)(\Sigma)_2 &= \exp\left(i2\pi\frac{kN_0}{N}\right)\sum_{r=.(N_0+1)/2}^{N_0-1}\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)r\right]\Bigg\} \\ &= \exp\left(i2\pi\frac{kN_0}{N}\right)\frac{\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)N_0\right]-\exp\left[i\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0+1)\right]}{\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]-1} \\ &= \frac{1-\exp\left(i2\pi\frac{kN_0}{N}\right)\exp\left[i\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0+1)\right]}{\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]-1}.\end{aligned} \tag{A2.57}$$

After some identical transformations, obtain the product of two exponential terms in Eq. (A2.57):

$$\begin{aligned}&\exp\left(i2\pi\frac{kN_0}{N}\right)\exp\left[i\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0+1)\right] \\ &\quad= \exp\left[i\pi\left(\frac{kN_0}{N}+n+\left(\frac{n}{N_0}-\frac{k}{N}\right)\right)\right] \\ &\quad= \exp\left[i\pi\left(\frac{kN_0}{N}+n-2n+\left(\frac{n}{N_0}-\frac{k}{N}\right)\right)\right] \\ &\quad= \exp\left[-i\pi\left(\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0-1)\right)\right].\end{aligned} \tag{A2.58}$$

Therefore,

$$\exp\left(i2\pi\frac{kN_0}{N}\right)(\Sigma)_2 = \frac{1-\exp\left(-i\pi\left[\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0-1)\right]\right)}{\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]-1}, \tag{A2.59}$$

and

$$\begin{aligned}\widetilde{a}_k &= \frac{1}{\sqrt{NN_0}}\left\{(\Sigma)_1 + \exp\left(i2\pi\frac{kN_0}{N}\right)(\Sigma)_2\right\} \\ &= \frac{1}{\sqrt{NN_0}}\frac{\exp\left[i\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0+1)\right]-1+1-\exp\left(-i\pi\left[\left(\frac{n}{N_0}-\frac{k}{N}\right)(N_0+1)\right]\right)}{\exp\left[i2\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]-1} \\ &= \frac{1}{\sqrt{NN_0}}\frac{\exp\left[i\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)N_0\right]-\exp\left(-i\pi\left[\left(\frac{n}{N_0}-\frac{k}{N}\right)N_0\right]\right)}{\exp\left[\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]-\exp\left[-\pi\left(\frac{n}{N_0}-\frac{k}{N}\right)\right]} \\ &= \frac{1}{\sqrt{NN_0}}\frac{\sin[\pi(n-kN_0/N)]}{\sin[\pi(n-kN_0/N)/N_0]}.\end{aligned} \tag{A2.60}$$

Hence,

$$\widetilde{a}_k = \sqrt{\frac{N_0}{N}}\sum_{n=0}^{N_0-1} a_n \mathrm{sincd}\{N_0;\,[\pi(n-kN_0/N)]\}. \tag{A2.61}$$

For signals with an even number of samples, the same aforementioned options of spectral zero-padding are possible: zeroing the highest-frequency component (Case 0), halving the highest-frequency component and duplicating it (Case 1), and duplicating the highest-frequency component (Case 2).

A2.10 Invertibility of the Shifted DFT and Signal Resampling

Let us assume that the direct ShDFT of a signal with samples $\{a_k\}$ is computed with a signal and its spectral sampling shift parameters (u, v):

$$\alpha_r^{(u,v)} = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} a_k \exp(i2\pi\frac{\tilde{k}_u\tilde{r}_v}{N});\ \tilde{k}_u = k+u;\ \tilde{r}_v = r+v; \tag{A2.62}$$

and the inverse ShDFT is computed with other shift parameters (p, q):

$$a_n^{(u/p;v/q)} = \frac{1}{\sqrt{N}}\sum_{r=0}^{N-1}\alpha_r^{(u,v)}\exp\left(-i2\pi\frac{\tilde{n}_p\tilde{r}_q}{N}\right);\ \tilde{n}_p = n+p;\ \tilde{r}_q = r+q. \tag{A2.63}$$

Replace the spectral coefficients $\alpha_r^{(u,v)}$ in Eq. (A2.63) with their expression through signal samples a_k (Eq. (A2.64)):

$$
\begin{aligned}
a_n^{(u/p;v/q)} &= \frac{1}{\sqrt{N}}\sum_{r=0}^{n-1}\left[\frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{\tilde{k}_u\tilde{r}_v}{N}\right)\right]\exp\left(-i2\pi\frac{\tilde{n}_p\tilde{r}_q}{N}\right)\\
&= \frac{1}{N}\sum_{k=0}^{N-1} a_k \sum_{r=0}^{N-1}\exp\left[i2\pi\frac{\tilde{k}_u(r+v)-\tilde{n}_p(r+q)}{N}\right]\\
&= \frac{\exp(-i2\pi\tilde{n}_p q/N)}{N}\sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{\tilde{k}_u v}{N}\right)\sum_{r=0}^{N-1}\exp\left(i2\pi\frac{\tilde{k}_u-\tilde{n}_p}{N}r\right)\\
&= \frac{\exp(-i2\pi\tilde{n}_p q/N)}{N}\sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{\tilde{k}_u v}{N}\right)\frac{\exp\left[i2\pi\left(\tilde{k}_u-\tilde{n}_p\right)\right]-1}{\exp\left(i2\pi\frac{\tilde{k}_u-\tilde{n}_p}{N}\right)-1}\\
&= \frac{\exp(-i2\pi\tilde{n}_p q/N)}{N}\sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{\tilde{k}_u v}{N}\right)\\
&\quad\times\frac{\exp\left[i\pi\left(\tilde{k}_u-\tilde{n}_p\right)\right]-\exp\left[-i\pi\left(\tilde{k}_u-\tilde{n}_p\right)\right]}{\exp\left(i\pi\frac{\tilde{k}_u-\tilde{n}_p}{N}\right)-\exp\left(-i\pi\frac{\tilde{k}_u-\tilde{n}_p}{N}\right)}\exp\left[i\pi\frac{\tilde{k}_u-\tilde{n}_p}{N}(N-1)\right]\\
&= \exp\left(-i\pi\frac{2q+N-1}{N}\tilde{n}_p\right)\sum_{k=0}^{N-1} a_k \exp\left(i\pi\frac{2v+N-1}{N}\tilde{k}_u\right)\frac{\sin\left[\pi\left(\tilde{k}_u-\tilde{n}_p\right)\right]}{N\sin\left(\pi\frac{\tilde{k}_u-\tilde{n}_p}{N}\right)}.
\end{aligned}
\tag{A2.64}
$$

Finally, obtain

$$
\begin{aligned}
a_n^{(u/p;v/q)} &= \exp\left(-i\pi\frac{2q+N-1}{N}\tilde{n}_p\right)\\
&\quad\times\sum_{k=0}^{N-1} a_k \exp\left(i\pi\frac{2v+N-1}{N}\tilde{k}_u\right)\mathrm{sincd}[N;\,\pi(k-n+u-p)].
\end{aligned}
\tag{A2.65}
$$

When shift parameters in the frequency domain v and q are chosen as

$$
v = q = -(N-1)/2, \tag{A2.66}
$$

Eq. (A2.65) converts to

$$
a_n^{(u/p;v/q)} = \sum_{k=0}^{N-1} a_k \mathrm{sincd}[N;\,\pi(n-k+p-u)], \tag{A2.67}
$$

which means that the inverse ShDFT with shift parameters $(p,\, q = -(N-1)/2)$ applied to the result of the direct ShDFT with shift

parameters $(u, v = -(N-1)/2)$ of signal with samples $\{a_k\}$ reconstructs a signal with samples $\{a_n^{(u/p;v/q)}\}$ discrete sinc interpolated from samples $\{a_k\}$ and shifted with respect to them by interval $(u-p)$. Chapter 5 showed that discrete sinc interpolation is the gold standard for the numerical interpolation of sampled data.

A2.11 DFT as a Spectrum Analyzer

One of the immediate applications of DFTs is signal Fourier spectrum analysis. Spectral coefficients

$$\alpha_r = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp\left(i2\pi \frac{\tilde{k}\,\tilde{r}}{N}\right);\ \tilde{k} = k+u;\ \tilde{r} = r+v \tag{A2.68}$$

obtained by applying the direct shifted DFT with shift parameters (u, v) to samples $\{a_k\}$ of a signal $a(x)$ can be regarded as samples

$$\alpha_r = \int_{-\infty}^{\infty} \alpha(f) PSF^{(SA)}(r\Delta_f - f)df \tag{A2.69}$$

of the Fourier spectrum

$$\alpha(f) = \int_{-\infty}^{\infty} a(x) \exp(i2\pi f x)dx \tag{A2.70}$$

of this signal.

Find the $PSF^{(SA)}(\cdot)$ of the DFT as a spectrum analyzer. Replace the signal samples $\{a_k\}$ of signal $a(x)$ in Eq. (A2.68) with their expression

$$a_k = \int_{-\infty}^{\infty} a(x) PSF^{(s)}(k\Delta_x - x)dx \tag{A2.71}$$

through the signal and $PSF^{(s)}(\cdot)$ of the signal samplimg device to obtain

$$\alpha_r = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} a_k \exp\left(i2\pi\frac{\tilde{k}\tilde{r}}{N}\right) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\left[\int_{-\infty}^{\infty} a(x)PSF^{(s)}(k\Delta_x - x)dx\right]\exp\left(i2\pi\frac{\tilde{k}\tilde{r}}{N}\right)$$

$$= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\left[\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty}\alpha(f)\exp(-i2\pi f x)df\right]PSF^{(s)}(k\Delta_x - x)dx\right]\exp\left(i2\pi\frac{\tilde{k}\tilde{r}}{N}\right)$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)df\sum_{k=0}^{N-1}\left[\int_{-\infty}^{\infty}\exp(-i2\pi f x)PSF^{(s)}(\tilde{k}\Delta_x - x)dx\right]\exp\left(i2\pi\frac{\tilde{k}\tilde{r}}{N}\right)$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)df\sum_{k=0}^{N-1}\left[\int_{-\infty}^{\infty}\exp\left[-i2\pi f(\tilde{k}\Delta_x - x)\right]PSF^{(s)}(x)dx\right]\exp\left(i2\pi\frac{\tilde{k}\tilde{r}}{N}\right)$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)df\sum_{k=0}^{N-1}\left[\int_{-\infty}^{\infty}\exp(i2\pi f x)PSF^{(s)}(x)dx\right]\exp\left(i2\pi\left(\frac{\tilde{k}\tilde{r}}{N} - \tilde{k}f\Delta_x\right)\right)$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)FR^{(s)}(f)df\sum_{k=0}^{N-1}\exp\left(i2\pi\left(\frac{k+u}{N}\tilde{r} - (k+u)f\Delta_x\right)\right)$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)FR^{(x)}(f)df\exp\left(i2\pi\frac{\tilde{r} - fN\Delta_x}{N}u\right)\sum_{k=0}^{N-1}\exp\left(i2\pi\frac{k}{N}(\tilde{r} - fN\Delta_x)\right)$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)FR^{(x)}(f)df\exp\left(i2\pi\frac{\tilde{r} - fN\Delta_x}{N}u\right)\frac{\exp(i2\pi(\tilde{r} - fN\Delta_x)) - 1}{\exp\left(i2\pi\frac{\tilde{r} - fN\Delta_x}{N}\right) - 1}$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)FR^{(s)}(f)df\exp\left(i2\pi\frac{\tilde{r} - fN\Delta_x}{N}u\right)$$

$$\times\frac{\exp(i\pi(\tilde{r} - kN\Delta_x)) - \exp(-i\pi(\tilde{r} - fN\Delta_x))}{\exp\left(i\pi\frac{\tilde{r} - kN\Delta_x}{N}\right) - \exp\left(-i\pi\frac{\tilde{r} - fN\Delta_x}{N}\right)}\exp\left[i2\pi\frac{N-1}{N}(\tilde{r} - fN\Delta_x)\right]$$

$$= \frac{1}{\sqrt{N}}\int_{-\infty}^{\infty}\alpha(f)FR^{(s)}(f)df\exp\left[i\pi\frac{2u + (N-1)}{N}(\tilde{r} - fN\Delta_x)\right]\frac{\sin[\pi(\tilde{r} - fN\Delta_x)]}{\sin\left(\pi\frac{\tilde{r} - fN\Delta_x}{N}\right)}. \tag{A2.72}$$

Set $u = -(N-1)/2$ to obtain

$$\alpha_r = \sqrt{N} \int_{-\infty}^{\infty} \alpha(f) FR^{(s)}(f) \text{sincd}[N;\, \pi(\tilde{r}\Delta_f - f)/\Delta_f] df, \tag{A2.73}$$

where $\Delta_f = 1/N\Delta_x$ is the spectral sampling interval. Equation (A2.73) implies that, to the accuracy of a constant multiplier, the PSF of the DFT as a spectrum analyzer

$$PSF^{(SA)}(r\Delta_f - f) = \text{sincd}[N;\, \pi(\tilde{r}\Delta_f - f)/\Delta_f] \tag{A2.74}$$

is the discrete sinc function. As much as the discrete sinc function approximates, for sufficiently large N, a continuous sinc function DFT can be considered as a perfect spectrum sampler.

A2.12 Quasi-continuous Spectral Analysis

Let $\{a_k\}$ be samples of a signal, $\{\alpha_r\}$ be samples of its DFT spectrum

$$a_k = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1`} \alpha_r \exp\left(-i2\pi \frac{kr}{N}\right), \tag{A2.75}$$

N be an odd number, and

$$\lambda_k^{(\delta_f)} = \begin{cases} \exp\left(i2\pi \dfrac{\omega_f k}{N}\right), & k = 0,\, 1, \ldots,\, \frac{N-1}{2} \\ \left(\lambda_{N-k}^{(\delta_f)}\right)^*, & k = \frac{N+1}{2}, \ldots,\, N-1 \end{cases} \tag{A2.76}$$

be a signal modulating function of arbitrary (not necessarily an integer) frequency ω_f, with an asterisk denoting complex conjugacy. Compute the DFT of signal $\{a_k\lambda_k\}$ and establish its link with the signal DFT spectrum $\{\alpha_r\}$:

$$\begin{aligned} \widetilde{\alpha}_s &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1`} a_k \lambda_k^{(\delta_f)} \exp(i2\pi \frac{ks}{N}) \\ &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \lambda_k^{(\delta_f)} \left[\frac{1}{\sqrt{N}} \sum_{r=0}^{N-1`} \alpha_r \exp\left(-i2\pi \frac{kr}{N}\right)\right] \exp\left(i2\pi \frac{ks}{N}\right) \\ &= \frac{1}{N} \sum_{r=0}^{N-1} \alpha_r \sum_{k=0}^{N-1`} \lambda_k^{(\delta_f)} \exp\left(i2\pi \frac{s-r}{N} k\right). \end{aligned} \tag{A2.77}$$

The internal sum over $\{\alpha_r\}$ in Eq. (A2.77) equals

$$\sum_{k=0}^{N-1/2`} \lambda_k^{(\delta_f)} \exp\left(i2\pi\frac{s-r}{N}k\right) + \sum_{k=N+1/2}^{N-1`} \lambda_k^{(\delta_f)} \exp\left(i2\pi\frac{s-r}{N}k\right)$$
$$= \sum_{k=0}^{(N-1)/2`} \lambda_k^{(\delta_f)} \exp\left(i2\pi\frac{s-r}{N}k\right) + \sum_{k=1}^{(N-1)/2`} \lambda_{N-k}^{(\delta_f)} \exp\left[i2\pi\frac{(s-r)(N-k)}{N}\right]$$
$$= \sum_{k=0}^{(N-1)/2`} \exp\left(i2\pi\frac{s-r+\delta_f}{N}k\right) + \sum_{k=1}^{(N-1)/2`} \exp\left(-i2\pi\frac{s-r+\delta_f}{N}k\right)$$
$$= \frac{\exp\left[i\pi\frac{(s-r+\omega_f)(N+1)}{N}\right]-1}{\exp\left(i2\pi\frac{s-r+\omega_f}{N}\right)-1} + \frac{\exp\left[-i\pi\frac{(s-r+\delta_f)(N+1)}{N}\right]-\exp\left(-i2\pi\frac{s-r+\delta_f}{N}\right)}{\exp\left(-i2\pi\frac{s-r+\delta_f}{N}\right)-1}$$
$$= \frac{\exp\left[i\pi(s-r+\omega_f)\frac{(N+1)}{N}\right]-1}{\exp\left(i2\pi\frac{s-r+\omega_f}{N}\right)-1} - \frac{\exp\left[-i\pi(s-r+\omega_f)\frac{(N-1)}{N}\right]-1}{\exp\left(i2\pi\frac{s-r+\omega_f}{N}\right)-1}$$
$$= \frac{\exp\left[i\pi(s-r+\omega_f)\frac{(N+1)}{N}\right]-\exp\left[-i\pi(s-r+\omega_f)\frac{(N-1)}{N}\right]}{\exp\left(i2\pi\frac{s-r+\omega_f}{N}\right)-1}$$
$$= \frac{\exp[i\pi(s-r+\omega_f)]-\exp[-i\pi(s-r+\omega_f)]}{\exp\left(i\pi\frac{s-r+\omega_f}{N}\right)-\exp\left(-i\pi\frac{s-r+\omega_f}{N}\right)} = \frac{\sin[\pi(s-r+\overline{\omega}_f)]}{\sin\left(\pi\frac{s-r+\omega_f}{N}\right)}. \tag{A2.78}$$

Insert this expression in Eq. (A2.77) and obtain the DFT spectrum of signal $\{a_k\lambda_k\}$ modulated by function $\{\lambda_k\}$ with frequency ω_f is the ω_f-shifted sincd-interpolated spectrum of the initial signal $\{a_k\}$

$$\widetilde{\alpha}_r = \sum_{r=0}^{N-1} \alpha_r \mathrm{sincd}[N;\ \pi(s-r+\xi_f)]. \tag{A2.79}$$

This result shows how one can perform quasi-continuous signal spectral analysis with sub-sample resolution.

A2.13 Signal Resizing and Rotation Capability of the Rotated Scaled DFT

Let $\{\alpha_{r,s}\}$ be the shifted DFT spectrum of signal $\{a_{m,n}\}$:

$$\alpha_{r,s} = \frac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} a_{m,n} \exp\left[i2\pi\left(\frac{\tilde{m}\,\tilde{r}}{N}+\frac{\tilde{n}\,\tilde{s}}{N}\right)\right]; \tag{A2.80}$$
$$\tilde{m} = m + u_m,\ \tilde{n} = \tilde{n} + u_n,\ \tilde{r} = r + v,\ \tilde{s} = s + v,$$

where u_m, u_n, and v are shift parameters in the signal domain and Fourier domain, respectively. Apply to this spectrum the inverse rotated scaled shifted DFT with the scale and rotation angle parameters σ and θ and with shift parameters $\{u_k; u_l; v; v\}$ (Eq. (10.32)), assuming that the spectrum, which is of size $N \times N$, is padded with zeros in both coordinates to size $\lceil \sigma N \rceil \times \lceil \sigma N \rceil$:

$$\begin{aligned}
\tilde{a}_{k,l} &= \frac{1}{\lceil \sigma N \rceil} \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \alpha_{r,s} \exp\left[-i2\pi \left(\frac{\tilde{k} \cos\theta + \tilde{l} \sin\theta}{\lceil \sigma N \rceil} \tilde{r} - \frac{\tilde{k} \sin\theta - \tilde{l} \cos\theta}{\lceil \sigma N \rceil} \tilde{s} \right) \right] \\
&= \frac{1}{\lceil \sigma N \rceil} \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \left\{ \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{m,n} \exp\left[i2\pi \left(\frac{\tilde{m}\tilde{r}}{N} + \frac{\tilde{n}\tilde{s}}{N} \right) \right] \right\} \\
&\quad \times \exp\left[-i2\pi \left(\frac{\tilde{k} \cos\theta + \tilde{l} \sin\theta}{\lceil \sigma N \rceil} \tilde{r} - \frac{\tilde{k} \sin\theta - \tilde{l} \cos\theta}{\lceil \sigma N \rceil} \tilde{s} \right) \right] = \frac{1}{\lceil \sigma N \rceil} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{m,n} \\
&\quad \times \left\{ \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \exp\left[i2\pi \left(\frac{\tilde{m} - N(\tilde{k} \cos\theta + \tilde{l} \sin\theta)/\lceil \sigma N \rceil}{N} \tilde{r} \right. \right. \right. \\
&\qquad \left. \left. \left. + \frac{\tilde{n} + N(\tilde{k} \sin\theta - \tilde{l} \cos\theta)/\lceil \sigma N \rceil}{N} \tilde{s} \right) \right] \right\},
\end{aligned} \tag{A2.81}$$

where $\tilde{k} = k + u_k$, $\tilde{l} = l + u_l$, $\tilde{r} = r + v$, $\tilde{s} = s + v$, and $\lceil \sigma N \rceil$ is the integer closest to σN. Compute the 2D sum over r and s, using for the sake of brevity the denotations $C_{mkl} = \tilde{m} - N(\tilde{k} \cos\theta + \tilde{l} \sin\theta)/\lceil \sigma N \rceil$ and $C_{nkl} = \tilde{n} + N(\tilde{k} \sin\theta - \tilde{l} \cos\theta)/\lceil \sigma N \rceil$:

$$\begin{aligned}
&\sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \exp\left[i2\pi \left(\frac{\tilde{m} - N(\tilde{k} \cos\theta + \tilde{l} \sin\theta)/\lceil \sigma N \rceil}{N} \tilde{r} + \frac{\tilde{n} + N(\tilde{k} \sin\theta - \tilde{l} \cos\theta)/\lceil \sigma N \rceil}{N} \tilde{s} \right) \right] \\
&= \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \exp\left[i2\pi \left(\frac{r+v}{N} C_{mkl} + \frac{s+v}{N} C_{nkl} \right) \right] \\
&= \exp\left[i2\pi \left(\frac{C_{mkl} + C_{nkl}}{N} v \right) \right] \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \exp\left[i2\pi \left(\frac{C_{mkl} r}{N} + \frac{C_{nkl} s}{N} \right) \right] \\
&= \exp\left[i2\pi \left(\frac{C_{mkl} + C_{nkl}}{N} v \right) \right] \frac{\exp(i2\pi C_{mkl}) - 1}{\exp\left(i2\pi \frac{C_{mkl}}{N} \right) - 1} \frac{\exp(i2\pi C_{nkl}) - 1}{\exp\left(i2\pi \frac{C_{nkl}}{N} \right) - 1} \\
&= \exp\left[i2\pi \left(\frac{C_{mkl} + C_{nkl}}{N} v \right) \right] \frac{\exp(i\pi C_{mkl}) - \exp(i\pi C_{mkl})}{\exp\left(i\pi \frac{C_{mkl}}{N} \right) - \exp\left(-i\pi \frac{C_{mkl}}{N} \right)} \exp\left(i\pi \frac{N-1}{N} C_{mkl} \right)
\end{aligned}$$

$$\times \frac{\exp(i\pi C_{nkl}) - \exp(i\pi C_{nkl})}{\exp\left(i\pi \frac{C_{nkl}}{N}\right) - \exp\left(-i\pi \frac{C_{nkl}}{N}\right)} \exp\left(i\pi \frac{N-1}{N} C_{nkl}\right)$$
$$= \exp\left\{ i\pi \left[\frac{2v+N-1}{N} (C_{mkl} + C_{nkl}) \right] \right\} \frac{\sin(\pi C_{mkl})}{\sin(\pi C_{mkl}/N)} \frac{\sin(\pi C_{nkl})}{\sin(\pi C_{nkl}/N)}. \tag{A2.82}$$

Choose a shift parameter in the spectral domain $v = -(N-1)/2$ and insert the obtained expression (Eq. (A2.82)) in Eq. (A2.81) and obtain

$$\begin{aligned} \widetilde{a}_{k,l} &= \frac{N^2}{\lceil \sigma N \rceil} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{m,n} \frac{\sin(\pi C_{mkl})}{\sin(\pi C_{mkl}/N)} \frac{\sin(\pi C_{nkl})}{\sin(\pi C_{nkl}/N)} \\ &= \frac{N^2}{\lceil \sigma N \rceil} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{m,n} \mathrm{sincd}\Big\{ N;\ \pi\tilde{m} - N(\tilde{k}\cos\theta + \tilde{l}\sin\theta)/\lceil \sigma N \rceil \Big\} \\ &\quad \times \mathrm{sincd}\Big\{ \pi[\tilde{n} + N(\tilde{k}\sin\theta - \tilde{l}\cos\theta)/\lceil \sigma N \rceil] \Big\}, \end{aligned} \tag{A2.83}$$

which means that the resulting image $\{\widetilde{a}_{k,l}\}$ is a discrete sinc interpolated $\lceil \sigma N \rceil / N$-scaled copy of image $\{a_{m,n}\}$ taken in θ-angle rotated coordinates.

A2.14 Rotated and Scaled DFT as Digital Convolution

Consider the rotated scaled DFT defined by Eq. (10.33):

$$\begin{aligned} \alpha_{r,s}^{(\sigma,\theta)} &= \frac{1}{\lceil \sigma N \rceil} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\cos\theta + \tilde{l}\sin\theta}{\lceil \sigma N \rceil} \tilde{r} - \frac{\tilde{k}\sin\theta - \tilde{l}\cos\theta}{\lceil \sigma N \rceil} \tilde{s} \right) \right] \\ &= \frac{1}{\lceil \sigma N \rceil} \sum_{k=0}^{N-1-1} \sum_{l=0}^{N-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\,\tilde{r} + \tilde{l}\,\tilde{s}}{\lceil \sigma N \rceil} \cos\theta + \frac{\tilde{l}\,\tilde{r} - \tilde{k}\,\tilde{s}}{\lceil \sigma N \rceil} \sin\theta \right) \right], \end{aligned} \tag{A2.84}$$

where $\tilde{k} = k + u$; $\tilde{l} = l + u$; $\tilde{r} = r + v$; $\tilde{s} = s + v$; u and v are the signal domain and frequency domain shift parameters, respectively; σ is a scale parameter; $\lceil \sigma N \rceil$ is the integer number closest to σN; and θ is the rotation angle. The following identities are used

$$\begin{aligned} 2(\tilde{k}\,\tilde{r} + \tilde{l}\,\tilde{s}) &= \tilde{r}^2 + \tilde{k}^2 + \tilde{s}^2 + \tilde{l}^2 - (\tilde{r} - \tilde{k})^2 - (\tilde{s} - \tilde{l})^2, \\ 2(\tilde{l}\,\tilde{r} - \tilde{k}\,\tilde{s}) &= \tilde{r}^2 - \tilde{k}^2 - \tilde{s}^2 + \tilde{l}^2 - (\tilde{r} - \tilde{l})^2 + (\tilde{s} - \tilde{k})^2 \end{aligned} \tag{A2.85}$$

to obtain

$$
\begin{aligned}
\alpha_{r,s}^{(\sigma,\theta)} &= \frac{1}{\lceil \sigma N \rceil} \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} a_{k,l} \exp\left[i2\pi \left(\frac{\tilde{k}\,\tilde{r} + \tilde{l}\,\tilde{s}}{\lceil \sigma N \rceil} \cos\theta + \frac{\tilde{l}\,\tilde{r} - \tilde{k}\,\tilde{s}}{\lceil \sigma N \rceil} \sin\theta \right) \right] \\
&= \frac{1}{\lceil \sigma N \rceil} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} a_{k,l} \left\{ \exp\left[i\pi \frac{\tilde{r}^2 + \tilde{k}^2 + \tilde{s}^2 + \tilde{l}^2 - (\tilde{r} - \tilde{k})^2 - (\tilde{s} - \tilde{l})^2}{\lceil \sigma N \rceil} \cos\theta \right] \right. \\
&\quad \left. \times \exp\left[i\pi \frac{\tilde{r}^2 - \tilde{k}^2 - \tilde{s}^2 + \tilde{l}^2 - (\tilde{r} - \tilde{l})^2 + (\tilde{s} - \tilde{k})^2}{\lceil \sigma N \rceil} \sin\theta \right] \right\} \\
&= \frac{\exp\left[i\pi \left(\frac{\tilde{r}^2 + \tilde{s}^2}{\lceil \sigma N \rceil} \cos\theta + \frac{\tilde{r}^2 - \tilde{s}^2}{\lceil \sigma N \rceil} \sin\theta \right) \right]}{\lceil \sigma N \rceil} \\
&\quad \times \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \left\{ a_{k,l} \exp\left[i\pi \left(\frac{\tilde{k}^2 + \tilde{l}^2}{\lceil \sigma N \rceil} \cos\theta - \frac{\tilde{k}^2 - \tilde{l}^2}{\lceil \sigma N \rceil} \sin\theta \right) \right] \right\} \\
&\quad \times \exp\left\{ -i\pi \left[\frac{(\tilde{r} - \tilde{k})^2 - (\tilde{s} - \tilde{l})^2}{\lceil \sigma N \rceil} \cos\theta + \frac{(\tilde{r} - \tilde{l})^2 - (\tilde{s} - \tilde{k})^2}{\lceil \sigma N \rceil} \sin\theta \right] \right\}.
\end{aligned}
\tag{A2.86}
$$

The sum in this expression is the 2D digital convolution of a signal $\{a_{k,l}\}$ modulated by a chirp function $\{\exp[i\pi(\frac{\tilde{k}^2+\tilde{l}^2}{\lceil \sigma N \rceil}\cos\theta - \frac{\tilde{k}^2-\tilde{l}^2}{\lceil \sigma N \rceil}\sin\theta)]\}$ with a convolution kernel, which is also a chirp function:

$$
\exp\left\{ -i\pi \left[\frac{(\tilde{r} - \tilde{k})^2 - (\tilde{s} - \tilde{l})^2}{\lceil \sigma N \rceil} \cos\theta + \frac{(\tilde{r} - \tilde{l})^2 - (\tilde{s} - \tilde{k})^2}{\lceil \sigma N \rceil} \sin\theta \right] \right\}.
$$

References

1. de Cogan, D., "Dr E.O.W. Whitehouse and the 1858 trans-Atlantic Cable", *History of Technology*, Vol. **10**, 1985, pp. 1–15.
2. Gabor, D., "Theory of Communication". *Journal of the Institution of Electrical Engineers*, **93**, 429–441, (1946).
3. Carson, J. R., "Notes on the Theory of Modulation", *Proceedings of the Institute of Radio Engineers*, 1922, **10**, p. 57.
4. Nyquist, H., "Certain Factors Affecting Telegraph Speed", *Bell System Technical Journal*, 1924, **3**, p. 324.
5. Küpfmüller, K., "Transient Phenomena in Wave Filters", *Electrische Nachrichten Technik*, 1924, **1**, p. 121.
6. Hartley, R. V. L., "Transmission of Information", *Bell System Technical Journal*, 1928, **7**, p. 535.
7. Kotel'nikov, V. A., "On the transmission capacity of "ether" and wire in electrocommunications," *Izd. Red. Upr. Svyazzi RKKA (Moscow)*, 1933. (Reprint: On the transmission capacity of "ether" and wire in electrocommunications," in Modern Sampling Theory: Mathematics and Applications, J. J. Benedetto and P. J. S. G. Ferreira, Eds. Boston, MA: Birkhauser, 2000).
8. Shannon, C. E., "Communication in the presence of noise," in *Proc.IRE*, vol. **37**, 1949, pp. 10–21.
9. Whittaker, J. M., "The Fourier theory of the cardinal functions," in *Proc. Math. Soc. Edinburgh*, vol. **1**, 1929, pp. 169–176.
10. Donoho, D., "Compressed sensing," *IEEE Trans. Inform. Theory* **52**(4), 1289–1306 (2006).
11. Candès, E., "Compressed sampling," *Proc. Int. Congress of Math.*, Madrid, Spain (2006).
12. Yaroslavsky, L. P., "How can one sample images with sampling rates close to the theoretical minimum?" *Journal of Optics*. v. **19**, Number 5, May 2017.
13. Yaroslavsky, L. P., "Compressed Sensing, the ASBSR-Method of Image Sampling and Reconstruction, and the Problem of Digital Image Acquisition with the Lowest Possible Sampling Rate", In: *Compressed Sensing: Methods, Theory and Applications*, Chapt.1., Ed. M. Jonathon Sheppard, Nova Publishers, 2018, ISBN: 978-1-53613-082-9.

14. Gonzalez, R. C. and R. E. Woods, Digital Image Processing, Prentice Hall, 2002.
15. Yaroslavsky, L. P., G. Shabat, B. G. Salomon, I. A. Ideses, and B. Fishbain "Nonuniform sampling, image recovery from sparse data and the discrete sampling theorem", *J. Opt. Soc. Am*. A/Vol. **26**, No. 3/March 2009.
16. Yaroslavsky, L. P., Theoretical Foundations of Digital Imaging using Matlab, CRC Press, 2013.
17. Donoho D. L. and Tanner J., "Exponential bounds implying construction of compressed sensing matrices, error-correcting codes, and neighborly polytopes by random sampling," *IEEE Trans. Inf. Theory* **56**, 2002 (2010).
18. Yaroslavsky, L. P., "Can compressed sensing beat the Nyquist sampling rate?" *Opt. Eng.*, **54**(7), 079701-1–079701-4 (2015).
19. Yaroslavsky, L. P. and B. Fishbain, Shabat G. and Ideses I., (2007), "Super-resolution in turbulent videos: making profit from damage", *Optics Letters*, Vol. **32**(20), pp. 3038–3040.
20. Yaroslavsky, L., Digital *Holography and Digital Image Processing*, Kluwer Academic Publishers, Boston, 2004.
21. Fiete, R. D., Multiple aperture imaging system, US patent US 6,943,946 B2, Sep. 13, 2005 (https://www.google.com/patents/US6943946).
22. Meijering, E. From Ancient Astronomy to Modern Signal and Image Processing. A Chronology of Interpolation, Proceedings of the IEEE, vol. **90**, no. 3, March 2002, pp. 319–342.
23. Yaroslavsky, L., B. Fishbain, G. Shabat, and I. Ideses, Super-resolution in turbulent videos: making profit from damage, *Optics Letters*, Nov. 1, 2007, Vol. **32**, No. 21.
24. Bilevich, L. and L. Yaroslavsky, New boundary effect free algorithm for fast and accurate image arbitrary scaling and rotation, Proc. SPIE 9139, Real-Time Image and Video Processing 2014, 913905 (May 15, 2014).
25. Gauss, C. F., "Nachclass: Theoria interpolationis methodo nova tractata", Werke, Band 3, 265–327, Königlishe Gesellshaft der Wissenshaften, Göttingen, 1866 (cited after M. T. Heideman, D. H. Johnson and C. S. Burrus, "Gauss and the history of the fast Fourier transform", IEEE ASSP Magazine, **1** (4), 14–81, 1984).
26. Press, W. H., B. P. Flannery, S. A Teukolsky, and W. T. Vetterling, Numerical recipes.The art of scientific computing. Cambridge University Press, Cambridge, 1987.
27. Heisenberg, W., *The Physical Principles of the Quantum Theory* (C. Eckart and F. C. Hoyt, trans.), University of Chicago Press, (1930).
28. Ahmed, N., T. Natarajan, and K.R. Rao, Discrete Cosine Transform, IEEE Trans. Computers, v. C-23 (1974), pp. 90–93.

Index

Leonid P. Yaroslavsky, Fellow of the Optical Society of America, M.S. (1961, Faculty of Radio Engineering, Kharkov Polytechnic Institute, Kharkov, Ukraine), Ph.D. (1968, Institute for Information Transmission Problems, Moscow, Russia), Dr. Sc. Habilitatus in Physics Mathematics (1982, State Optical Institute, St. Petersburg, Russia). From 1963–1983, he headed a the Digital Image Processing and Digital Holography group at the Institute for Information Transmission Problems, Russian Academy of Sciences, which digitally processed the images transmitted from the spaceships Mars-4, Mars-5, Venus-9, and Venus-10, and obtained the first color images of the surface of Mars and the first panoramas from the surface of Venus.

From 1983–1995, he headed the Digital Optics Laboratory at the Institute. From 1995–2008, he was a Professor with the Faculty of Engineering at Tel Aviv University, where, at present, he is a Professor Emeritus. He was also a Visiting Professor at the University of Erlangen, Germany; the National Institute of Health, Bethesda, MD, USA; Institute of Optics, Orsay, France; Institute Henri Poincare, Paris, France; International Center For Signal Processing, Tampere University of Technology, Tampere, Finland; Agilent Laboratories, Palo Alto, Ca, USA; Gunma University, Kiryu, Japan; and the Autonomous University of Barcelona. He has supervised more than 20 Ph.D. candidates, and he has authored or edited several books and more than 100 papers on digital image processing and digital holography.