Clifford A. Pickover

Computers, Pattern, Chaos, and Beauty

Graphics from an Unseen World

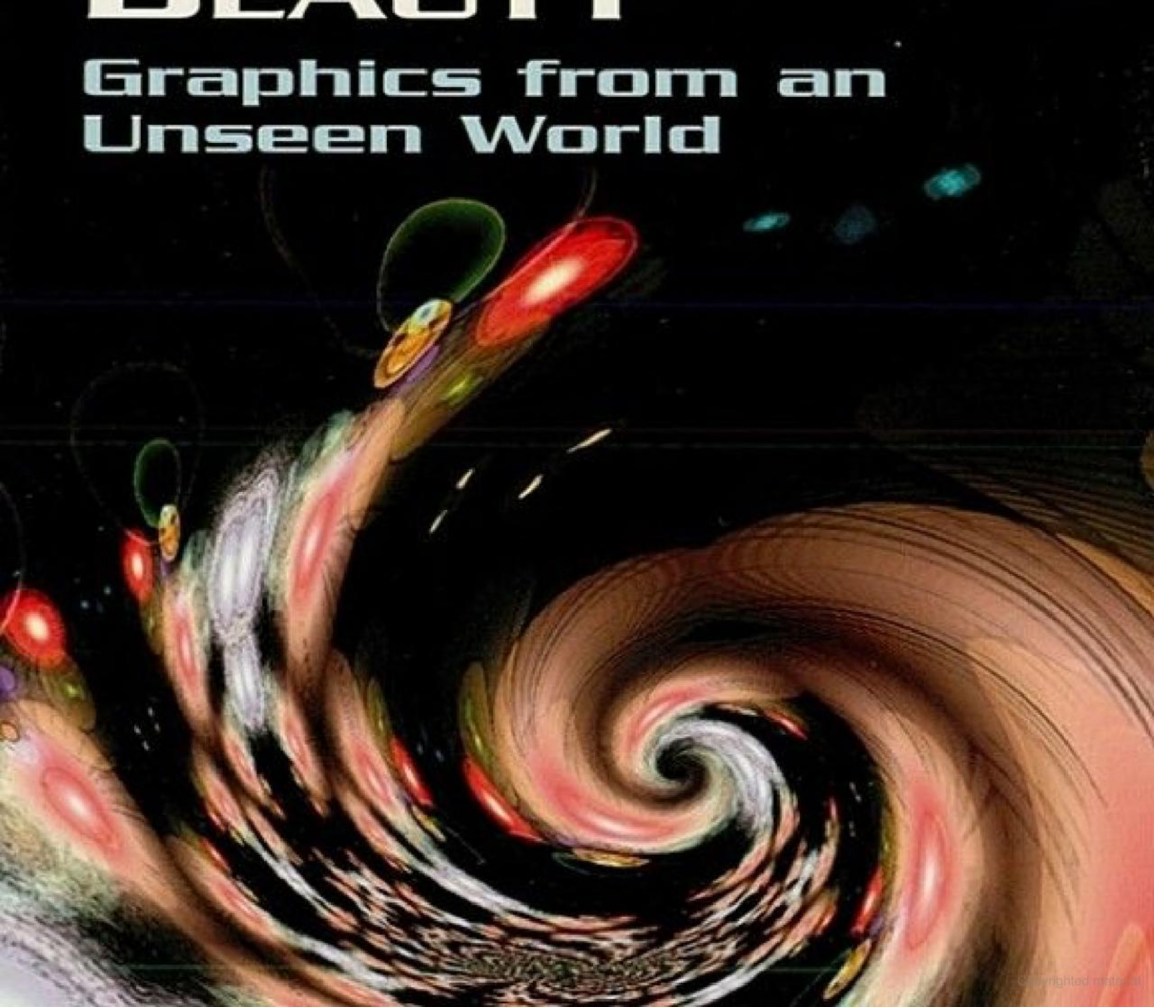
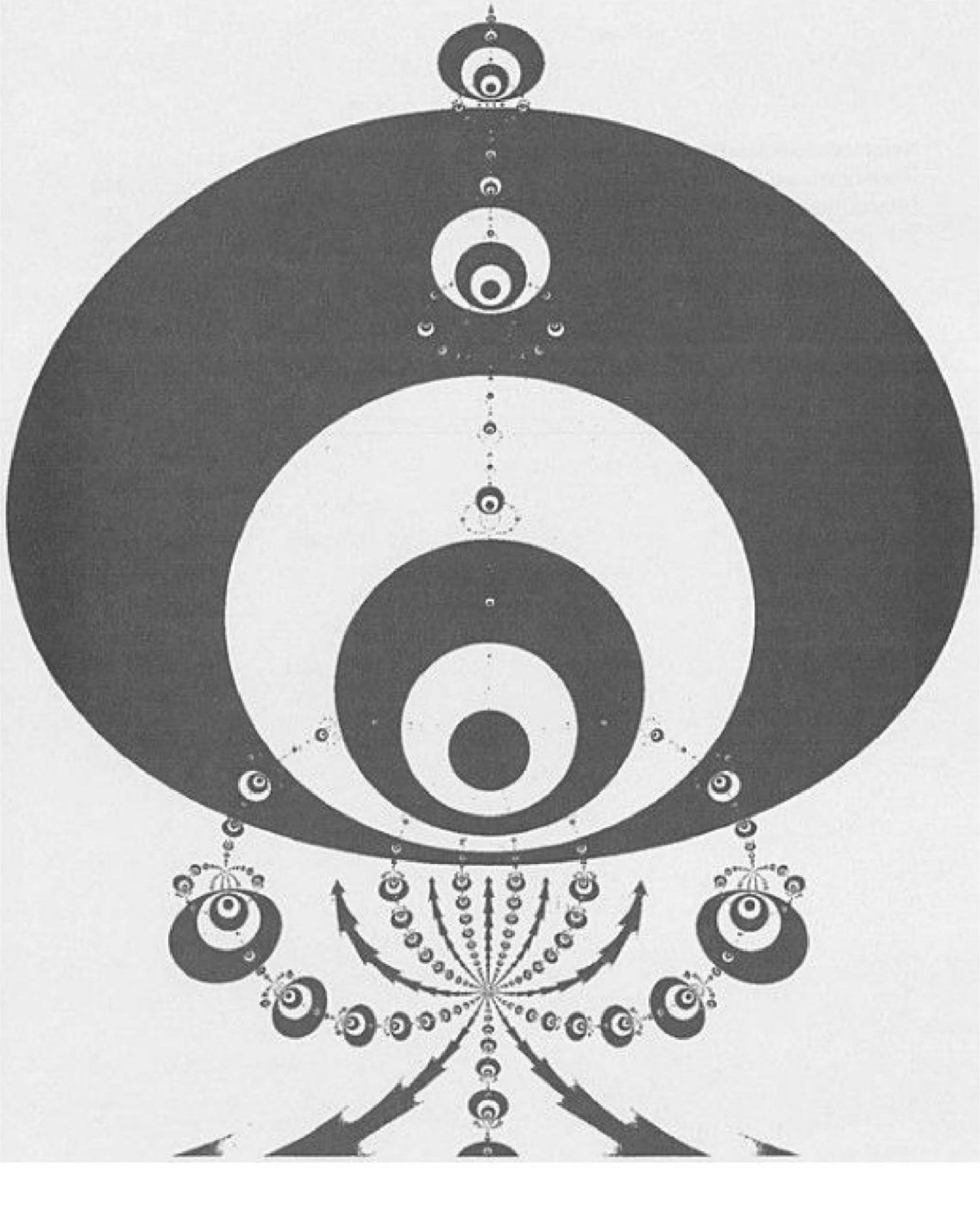
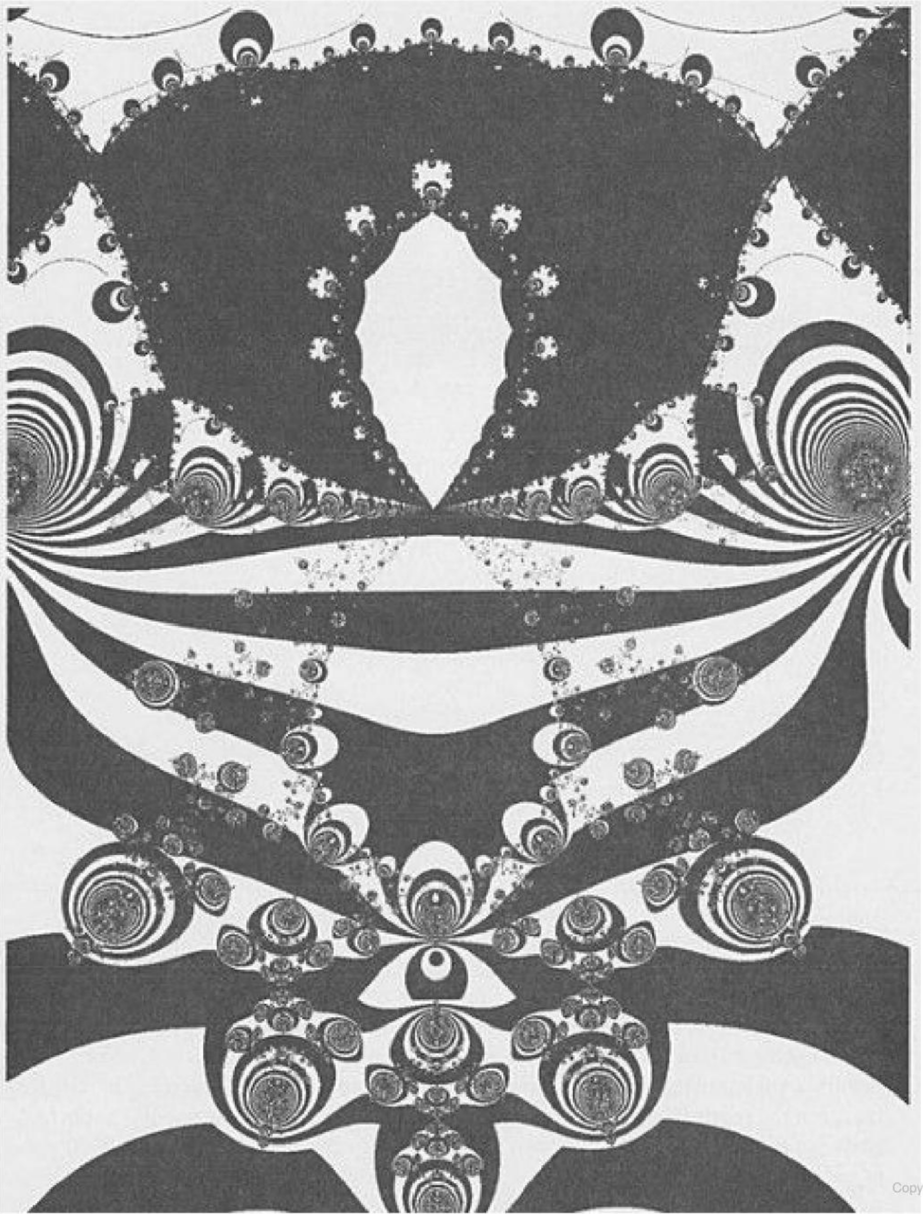


Table of Contents



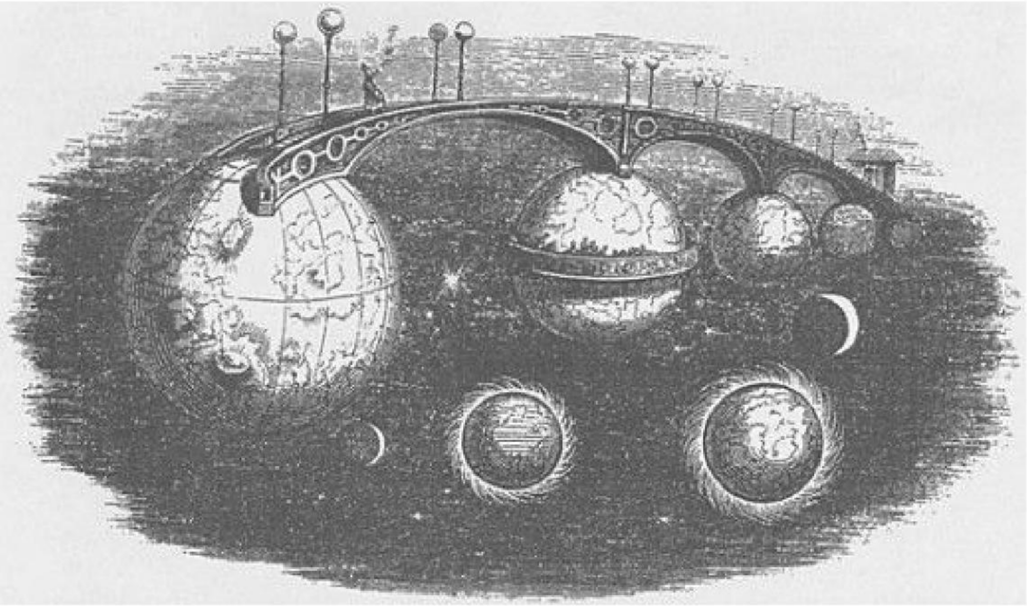
Part I

INTRODUCTION



Chapter 1

Computers and Creativity



"The heavens call to you, and circle about you, displaying to you their eternal splendors, and your eye gazes only to earth."

Dante

Imagine a world with no shadows, no sun.

Imagine computing, with no mystery, no creativity, no human dreamer. The beauty and importance of computers lie mainly in their usefulness as a tool for reasoning, creating and discovering. Computers are one of our most important tools for reasoning beyond our own intuition. In order to show the eclectic nature of computer "territory," this book contains a collage of topics which have in common their highly visual nature, and each can be effectively explored using a computer.

Imagery is the heart of much of the work described in this book. To help understand what is around us, we need eyes to see it. Computers with graphics can be used to produce visual representations with a myriad of perspectives. These perspectives are demonstrated by the subjects presented in this book. The applications are varied and include fields as diverse as speech synthesis, molecular biology, mathematics, and art. Yet it is hoped that they all combine to illustrate the wonder in "lateral thinking" with computers (defined in Sect. 1.2 "Lateral Use of Computer Software Tools" on page 4).

1.1 Objectives

Where possible, the material is organized by subject area. The purpose of this book is:

1. to present several novel graphical ways of representing complicated data,
2. to show the role of the aesthetics in mathematics and to suggest how computer graphics gives an appreciation of the complexity and beauty underlying apparently simple processes,
3. to show, in general, the beauty, adventure and potential importance of creative thinking using computers,
4. to show how the computer can be used as an instrument for simulation and discovery.

1.2 Lateral Use of Computer Software Tools

> *"He calmly rode on, leaving it to his horse's discretion to go which way it pleased, firmly believing that in this consisted the very essence of adventures. "*
>
> Cervantes, *Don Quixote*

"Lateral thinking" is a term discussed by writer/philosopher, Robert Pirsig (author of *Zen and the Art of Motorcycle Maintenance*). As he explains it, lateral thinking is reasoning in a direction not naturally pointed to by a scientific discipline. It is reasoning in a direction unexpected from the actual goal one is working toward (see also de Bono, 1975). In this book, the term "lateral thinking" is used in an extended way to indicate not only action motivated by unexpected results, but also the deliberate drift of thinking in new directions to discover what can be learned. It is also used to indicate the application of a single computer software tool to several unrelated fields.

Let’s list a few examples of the lateral use of computer software tools. These examples will be discussed in greater detail later in the book. To give some personal history and examples: while creating analysis tools for speech synthesis research (Chapter 3), the author drifted laterally and examined their application to the study of the breathing motions of proteins. This naturally led to other biological molecules such as genes. In this application, the sequence of bases in a human bladder cancer gene is treated as if it were a speech waveform in order to gain a new perspective. These studies presented traditional graphics and analysis in new applications in an effort to visualize complex data.

This idea of novel ways for making complicated data understandable led to the application of Chernoff faces (cartoon faces whose facial coordinates depend on the input data). These faces can be applied to a range of sounds, mathematical equations, and genetic sequences. The faces rely on the feature-integration abilities of the human brain to condense a vast amount of data.

Does there exist an optimal representation for visual characterization and detection of significant information in data? This question, along with the face research, further stimulated my interest in the human visual system. Part of Chapter 4 discusses the use of a perceptual illusion, achieved with patterns of dots, to the characterization of subunit relationships in proteins. These patterns, called “Moire interference patterns,” resemble galaxies and whirlpools. The interference patterns led to another question concerning vision and data characterization: Can symmetry operators, like the mirrors in a child’s kaleidoscope, help us to understand data? To answer this question, another dot-based tool was developed; this representation is comprised of snowflake-like patterns of colored dots and is used to characterize sounds.

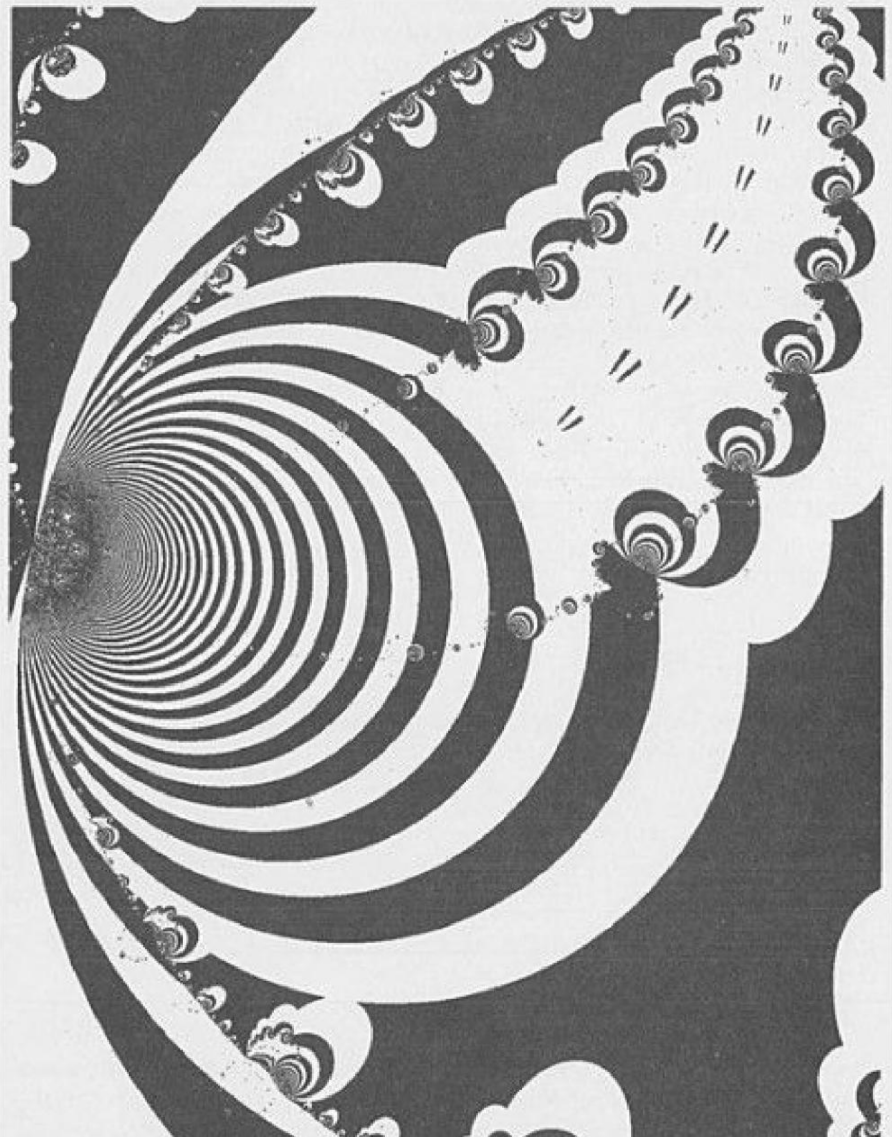
Intriguing even as an art form, these dot patterns may be a way of visually fingerprinting natural and synthetic sounds and of allowing researchers to detect patterns not easily captured with traditional analyses.

A short quote from Robert Pirsig can apply to the joy computer programmers, artists, and scientists often experience when experimenting on a computer:

> *“It’s the sides of the mountain which sustain life, not the top. Here’s where things grow.”*

1.3 Reading List for Chapter 1

Two interesting books on the topics of creativity and lateral thinking are De Bono (1970) and Pirsig (1975).



Chapter 2

Hidden Worlds

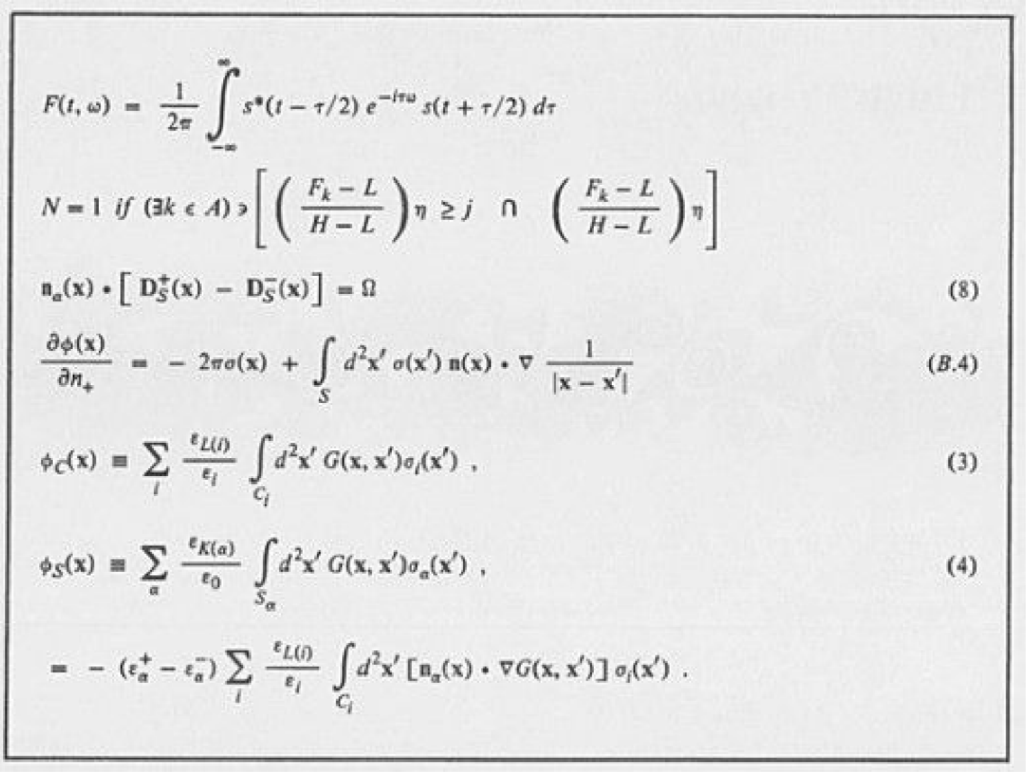
> *"If we wish to understand the nature of the Universe we have an inner hidden advantage: we are ourselves little portions of the universe and so carry the answer within us."*
>
> Jacques Boivin, *The Heart Single Field Theory*

2.1 Digits, Symbols, Pictures

We live in a civilization where numbers play a role in virtually all facets of human endeavor. Even in our daily lives we encounter multidigit zip-codes, social security numbers, credit card numbers, and phone numbers. In many ways the requirements for ordinary living are a great deal more complicated than ever before. Digits...digits...digits.... It all seems so dry sometimes. And yet, when one gazes at a page in a scientific journal and sees a set of complicated-looking equations, such as those chosen from pages of scientific texts (Figure 2.1), a sense of satisfaction is generated: *the human mind, when aided by numbers and symbols, is capable of expressing and understanding concepts of great complexity*. Ever since "visionary" mathematical and physical relations trickled like rain onto the rooftop of 20th century man, we have begun to realize that some descriptions of nature lie beyond our traditional, unaided ways of thinking.

The *expression* of complicated relations and equations is one magnificent step — *insight* gained from these relations is another. Today, computers with graphics can be used to produce representations of data from a number of perspectives and to characterize natural phenomena with increasing clarity and usefulness. ***"Mathematicians couldn't solve it until they could see it!"*** a caption in a popular scientific magazine recently exclaimed when describing work done on curved mathematical surfaces (*Science Digest*, January, 1986, p. 49). In addition, cellular automata and fractals — classes of simple mathematical systems with exotic behavior — are beginning to show promise as models for a variety of physical processes (see "Genesis Equations" on page

104 and “Tesselation Automata Derived from a Single Defect” on page 295). Though the rules governing the creation of these systems are simple, the patterns they produce are complicated and some-times seem almost random, like a turbulent fluid flow or the output of a cryptographic system.



$$F(t, \omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} s^*(t - \tau/2)\, e^{-i\tau\omega}\, s(t + \tau/2)\, d\tau$$

$$N = 1 \ \textit{if} \ (\exists k \in A) \ni \left[\left(\frac{F_k - L}{H - L} \right) \eta \geq j \ \cap \ \left(\frac{F_k - L}{H - L} \right) \eta \right]$$

$$\mathbf{n}_\alpha(\mathbf{x}) \cdot \left[\mathbf{D}_S^+(\mathbf{x}) - \mathbf{D}_S^-(\mathbf{x}) \right] = \Omega \tag{8}$$

$$\frac{\partial \phi(\mathbf{x})}{\partial n_+} = -2\pi\sigma(\mathbf{x}) + \int_S d^2\mathbf{x}'\, \sigma(\mathbf{x}')\, \mathbf{n}(\mathbf{x}) \cdot \nabla \frac{1}{|\mathbf{x} - \mathbf{x}'|} \tag{B.4}$$

$$\phi_C(\mathbf{x}) \equiv \sum_i \frac{\varepsilon_{L(i)}}{\varepsilon_i} \int_{C_i} d^2\mathbf{x}'\, G(\mathbf{x}, \mathbf{x}')\sigma_i(\mathbf{x}') \ , \tag{3}$$

$$\phi_S(\mathbf{x}) \equiv \sum_\alpha \frac{\varepsilon_{K(\alpha)}}{\varepsilon_0} \int_{S_\alpha} d^2\mathbf{x}'\, G(\mathbf{x}, \mathbf{x}')\sigma_\alpha(\mathbf{x}') \ , \tag{4}$$

$$= -(\varepsilon_\alpha^+ - \varepsilon_\alpha^-) \sum_i \frac{\varepsilon_{L(i)}}{\varepsilon_i} \int_{C_i} d^2\mathbf{x}' \left[\mathbf{n}_\alpha(\mathbf{x}) \cdot \nabla G(\mathbf{x}, \mathbf{x}') \right] \sigma_i(\mathbf{x}') \ .$$

***Figure 2.1**. The symbols of mathematics.*

Today, in almost all branches of the scientific world, computer graphics is helping to provide incito and to reveal hidden relationships in complicated systems. Figure 2.2 is just one example of the use of graphics to represent the behavior of mathematical functions. Notice the complexity of the behavior exhibited by the function used to create Figure 2.2 — behavior mathematicians could not fully appreciate before computers could display it.

Like computer models of a host of natural phenomena such as vortices, fluid flow, and other chaotic (irregular) systems, pictures such as these reveal an unpredictable, exciting and visually attractive universe.

2.2 Computers and Art

> *"Salvador Dali once exploded a bomb filled with nails against a copper plate, producing a striking but random pattern. Many other artists have also utilized explosives in their work, but the results have generally been unpredictable."*
>
> Febr. 1989, *Scientific American*

Not only can computers and graphics be used in counting and measuring, but they also are of enormous help in producing visual art (Figure 2.3). (See the

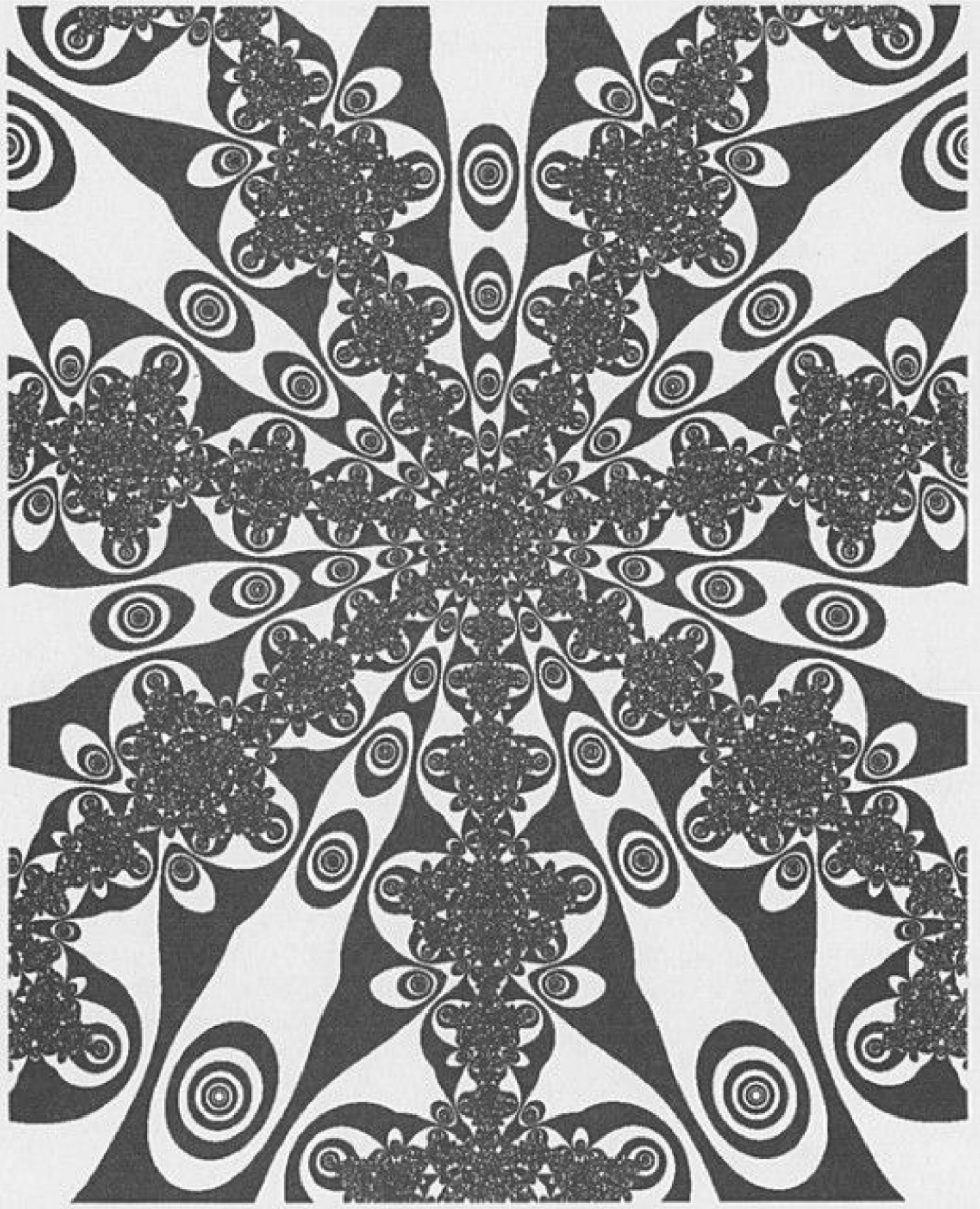


Figure 2.2. *Mathematics and beauty. Appealing as an art form, this intricate diagram is called a Halley map, and it can be used to represent properties of*

numerical methods. The generating function is z z^7 1 (see “Numerical Approximation Methods” on page 275).

Reading List at the end of the chapter for more information on computer art.) The break between artistic and scientific pursuits is often apparent today. Whereas the earlier thinkers pursued science and art in the light of guiding principles such as harmony and proportion, today some hold the view that science stifles the artistic spirit. Nevertheless, the computer is capable of creating images of captivating beauty and power. Techniques such as animation, color and shading all help to create fantastic effects (Figure 2.4).

In much of the work in this book, beauty, science and art are intertwined, and — judging from the response from readers — this contributes to the fascination of these approaches for both scientists and laypeople. From an artistic standpoint, mathematical equations provide a vast and deep reservoir from which artists can draw. New algorithms (“recipes”), such as those outlined in this book, interact with such traditional elements as form, shading and color to produce futuristic images. The mathematical recipes function as the artist’s assistant, quickly taking care of much of the repetitive and sometimes tedious detail. By becoming familiar with advanced computer graphics, the computer artist may change our perception of art.

2.3 Computer Graphics: Past and Present

“Computers are useless. They can only give you answers. ”

Pablo Picasso

In the beginning of the modern computer age, computer graphics consisted of the multitude of Abe Lincolns, Mona Lisas, and Charlie Brown cartoons spewed forth from crude character line-printers in campuses and laboratories. Better hardware led to better images. In the 1970s we saw an increasing amount of computer animation, computer generated-commercials and films — and Pacman. Today, in science, computer graphics is used to reveal a variety of subtle patterns in nature and mathematics. The field of computer graphics is very important in: 1) revealing hidden correlations and unexpected relationships (and as an adjunct to numerical analysis), 2) simulating nature, and 3) providing a source of general scientific intuition. Naturally, these three uses overlap. Pseudo-color, animation, three-dimensional figures, and a variety of shading schemes are among the techniques used to reveal relations not easily visible in more traditional data representations.

2.4 Computers: Past and Present

Taking a step back: how long ago did computing really begin? Probably, the first calculating machine to help expand the mind of man was the abacus. The abacus is a manually operated storage device which aids a *human* calculator. It consists of beads and rods and originated in the Orient more than 5,000 years ago. Archeologists have since found geared calculators, dated back to 80 BC, in the sea off northwestern Crete. Since then, other primitive calculating machines have evolved, with a variety of esoteric sounding names, including: Napier's bones (consisting of sticks of bones or ivory), Pascal's arithmetic machine (utilizing a mechanical gear system), Leibniz' Stepped Reckoner, and Babbage's analytical engine (which used punched cards) (see Gardner, 1986, for more detail).

Continuing with more history: the Atanasoff-Berry computer, made in 1939, (Mackintosh, 1988), and the 1500 vacuum tube Colossus, were the first program-mable electronic machines. The Colossus first ran in 1943 in order to break a German coding machine named Enigma. The first computer able to store programs was the Manchester University Mark I. It ran its first program in 1948. Later, the transistor and the integrated circuit enabled micro-miniaturization and led to the modern computer.

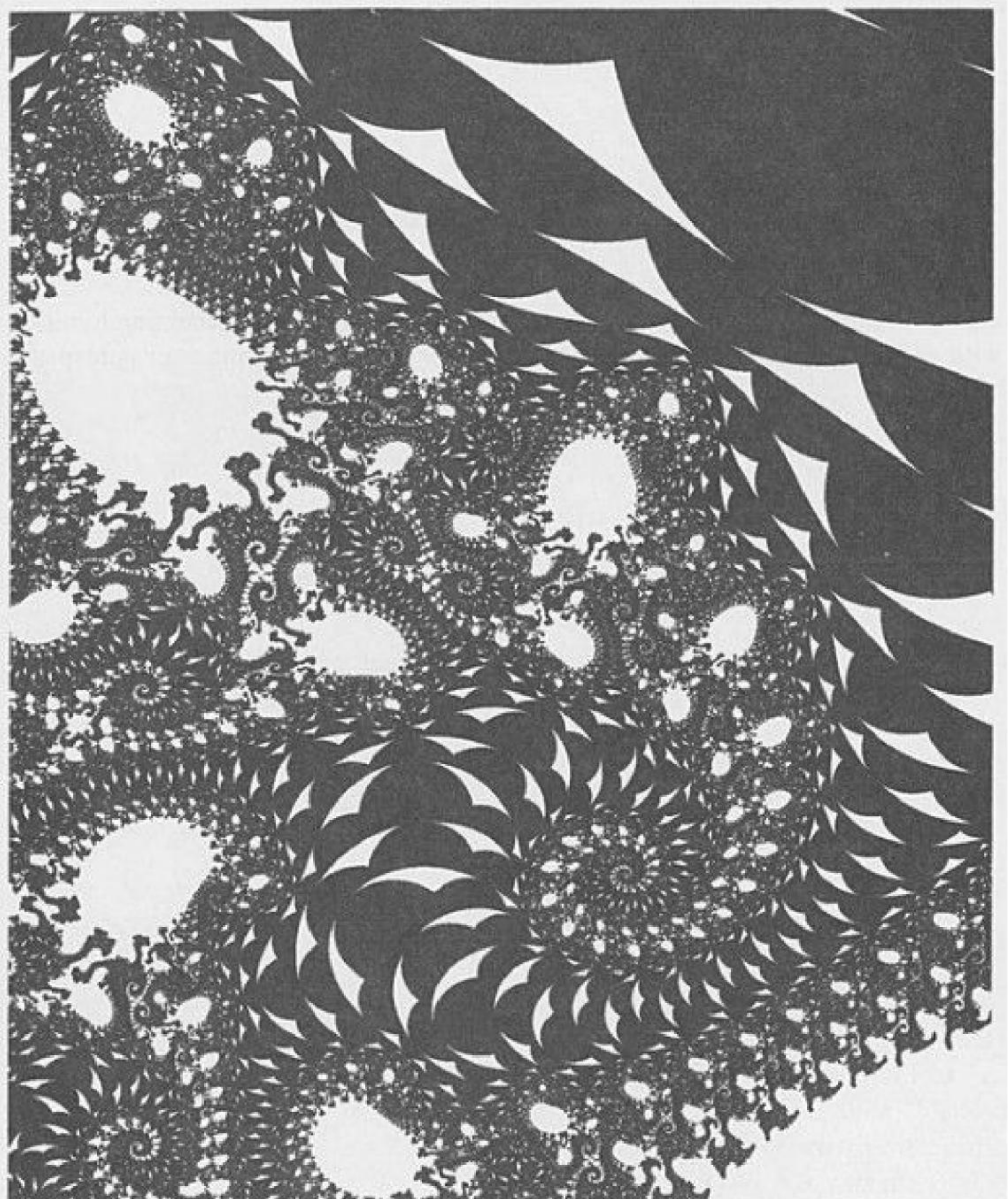


***Figure 2.3*. Complexity and simplicity. This iteration map represents the complicated behavior of a simple function, $z \quad z^2 + \quad$. (See "More Beauty from Complex Variables" on page 113 for more information on this plot).**

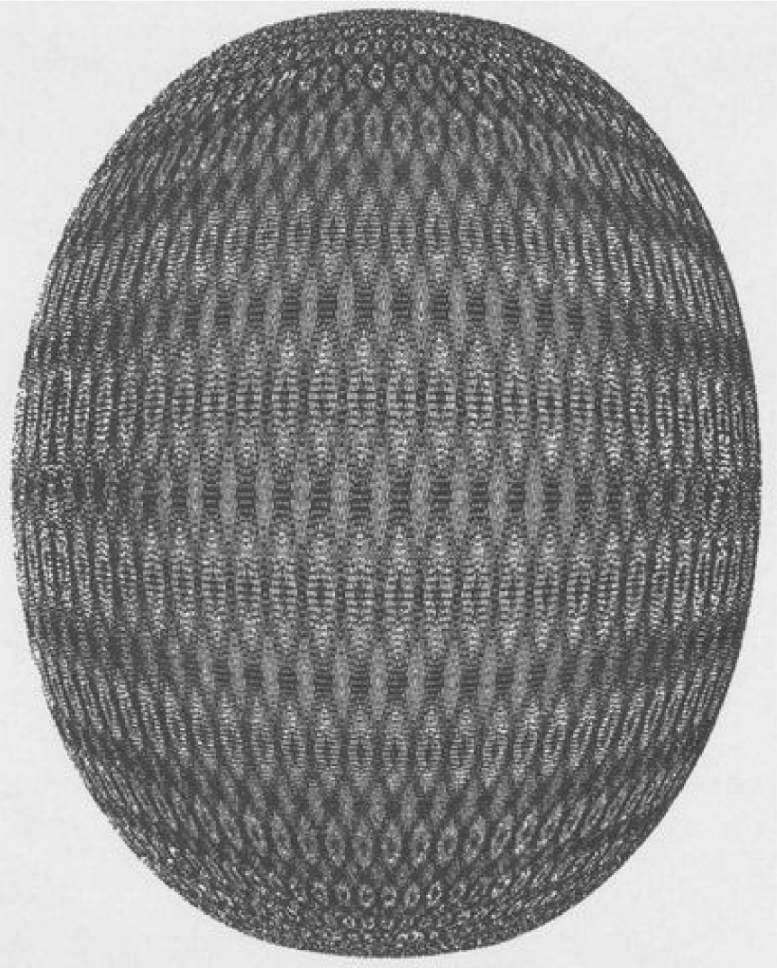


***Figure 2.4*. Computers and art. Many of the ornaments of modern man and his ancient cultures consist of symmetrical and repeating designs. Here, this tiled egg-shape is produced from a simple generating formula, $z_{xy} = \quad (\sin x + \sin y)$ (see "Synthesizing Ornamental Textures" on page 227).**

In 1988, one of the world's most powerful and fastest computers is the liquid-cooled Cray-2 produced by Cray Research. It performs 250 million floating point arithmetic operations per second — much more expensive than the abacus or Napier's bones, but also much faster!

2.5 The Human Brain vs. the Computer Brain

2.5.1 The Human Brain

While it's clear that the computer "brain" is vastly superior to man's brain in certain tasks, for perspective it is useful to mention some of the lesser known capacities of the human mind-machine.

The human brain weighs about three pounds and is made of roughly 10 billion neurons, each neuron receiving connections from perhaps 100 other neurons and connecting to still 100 more (Figure 2.5). The web of interconnections is so complex that the whole cortex can be thought of as one entity of integrated activity. Many neurobiologists believe that memory, learning, emotions, creativity, imagination — all the unique elements of human character — will ultimately be shown to reside in the precise patterns of synaptic interconnections in the human brain. The importance of the brain's system of pathways has led some scientists to hypothesize an *equation for consciousness* itself: $C = f_1(n)f_2(s)$ (Rose, 1976). Consciousness C is represented on the cellular level by a function of neural cell number, n, and connectivity s. It has been shown that small systems of neurons (i.e., under 10,000 neurons), such as those in simple invertebrates, are capable of learning and memory. In 1987, computer models of neural networks helped researchers begin to untangle the complexities of biological processes such as vision.[1]

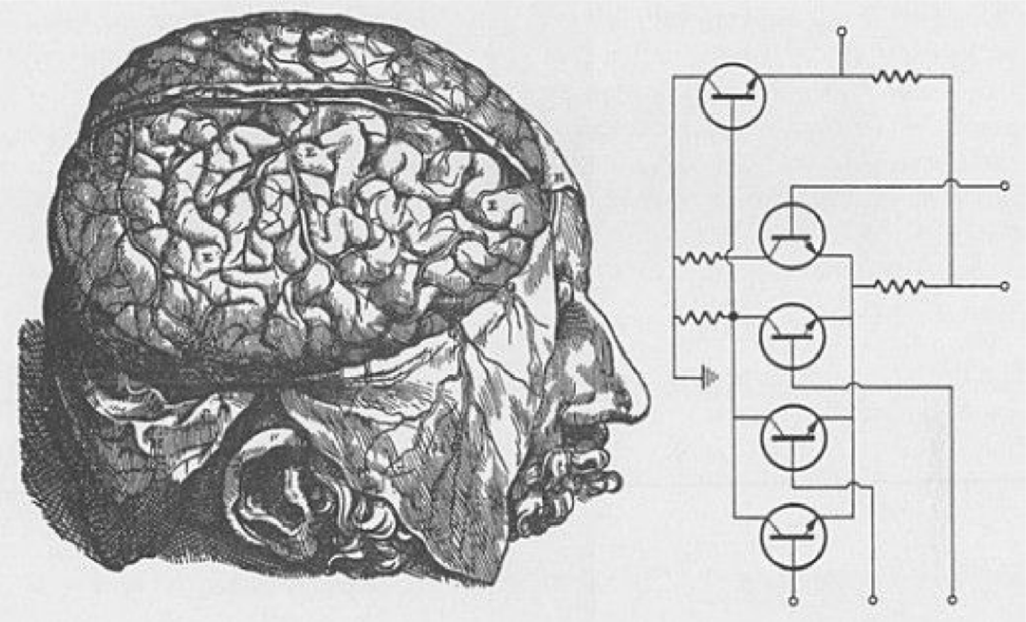


Figure 2.5. The human brain and an electronic circuit.

2.5.2 Human Computers

We know that the human brain is capable of profound and important functions such as creativity and imagination, but often little is said of its computing and storage capabilities. In some instances, the human memory can be great. For example, in 1974, one individual recited 16,000 pages of Buddhist texts without error. Later, a 23-year old Indian man recited   from memory to 31,811 places in about 3 hours. (Note that in 1987 a NEC SX-2 supercomputer *calculated*   to more than 134 million digits. In 1989, the Chudnovsky brothers, two Columbia University mathematicians, computed over one billion digits of   using a Cray 2 and an IBM 3090-VF computer.)

As an example of computational capabilities of the human brain, Willem Klein in 1981 was able to extract the 13th root of a 100-digit number in about one minute. In addition, there are the autistic savants — people who can perform mental feats at a level far beyond the capacity of a normal person but whose overall IQ is very low.

2.6 New Applications of Calculating: A Sampling and Digression

In many branches of science, progress is enhanced by finding new ways to calculate and simulate. Among these fields are plasma physics, astrophysics, molecular biology, and geology. In music and speech, computers are making an impact. With the increasing availability and improvement of voice synthesis technology, singing synthesizers may herald the next revolution in music, much as did the electronic music synthesizers of the 1970s. Not only will computers be singing songs, but they will be writing the songs they sing (see “Singing Computers” on page 23).

In medicine, carefully designed computer programs and systems assist in making medical diagnoses. Psychiatrists at the Salt Lake City Veterans’ Administration Hospital have been testing such systems to diagnose mental illness. A patient is first greeted by a human receptionist who takes him to a private room containing a computer terminal. A computer program then presents a battery of tests, and twenty minutes later a staff psychiatrist reviews the questions to make a final judgement about the person’s condition. Diagnosis by computer programs has meant substantial savings in time and money for the patient, and, most importantly, computer diagnosis has proved to be remarkably accurate. The computer never tires, never is biased — and the impersonal relationship appears actually to help. Another medical use is in the hospital operating room which has become increasingly colonized by “electronic nurses”: computers that allow doctors to regulate operating-room lighting, adjust instruments, or even call “scalpel” with just the punch of a keyboard.

Advances in bioengineering allow, in some cases, the restoration of sight to the blind and hearing to the deaf. Computer-driven prostheses have been designed to take over the function of body organs such as bladders, blood vessels, testicles, and fallopian tubes. Perhaps one of the greatest advances in computers and medicine is the display and reconstruction of the interior of portions of the body (tomography). Magnetoencephalography and magnetocardiography now describe the subtle magnetic fields emanating from the brain and heart.

In short, computer calculations are now beginning to radically *change how scientists pursue and conceptualize problems,* and computer models open up entire new areas of exploration. In fact, of all the changes in scientific methodology, probably none is more important than the use of computers. The sheer amount of data generated by experiments is so large that comparisons and conclusions could not be made without computers. For example, massive DNA sequences have been uncovered — and only with the computer can hidden correlations be found within these bases in the genetic materials of organisms. Not unlike the search for extraterrestrial signals from space, scientists try to reconstruct “messages” and patterns in DNA strands, mathematical progressions, and a range of natural phenomena.

Educational areas vastly benefit from computerization. With appropriate software, university networks of personal workstations are facilitating remedial learning. Students are also using the computer as a tool for learning computational physics. The computer-assisted videodisc combines computer and video technique in an instructional tool which is changing the way chemistry, physics, and engineering is taught. Even Chaucer has become computerized; for example, a student studying Chaucer's *Canterburry Tales* can now display the original text side-by-side with notes on the meanings of individual words and a modern English translation.

The remarkable panoply of computer applications seems to be growing: computers play a role in the design of other computers, in video analysis systems, in protein structure determination and design, cryptographic systems, robotics, and molecular evolution studies. Computer-drawn 3-D structures of viruses, such as polio viruses, may lead us to new cures. Thunderstorm modelling today involves the simulation of the growth of a single cloud into storms that can produce tornadoes. Extensive earthquake-detecting systems consist of the interplay of computers, instrumentation, telemetry and data reduction. The search for extraterrestrial intelligence employs the automatic detection of interstellar signals and requires sophisticated computers. Flight-deck automation is changing the role of the human pilot. Computer information services offer biographies and high-resolution graphics of the FBI's most wanted fugitives (*Science Digest*, 1986, January, p. 15). Now some computers not only learn how to draw artistically in the style of famous artists, but actually improve with practice (Kluger, 1987). (Readers interested in computer programs which mimic the artwork of Miro should see Kirsh and Kirsh, 1988.)

Computer-aided design (CAD) is rendering the blueprint obsolete. CAD programs can allow on-screen tours inside buildings, and future systems will be able to simulate the flow of sunlight at various times of the day. Finally, just a week before this chapter was written, an article appeared which described the use of high-speed computers to create the perfect bowling ball! After seven years of research, a precisely weighted polyester ball has been created that is almost entirely free of minute wiggles and bounces that bedevil the average bowler.

2.7 One Final Word

> *"Au fond de l'Inconnu pour trouver du nouveau"* (Into the depths of the Unknown in quest of something new)
>
> Charles Baudelaire, in *Le Voyage*

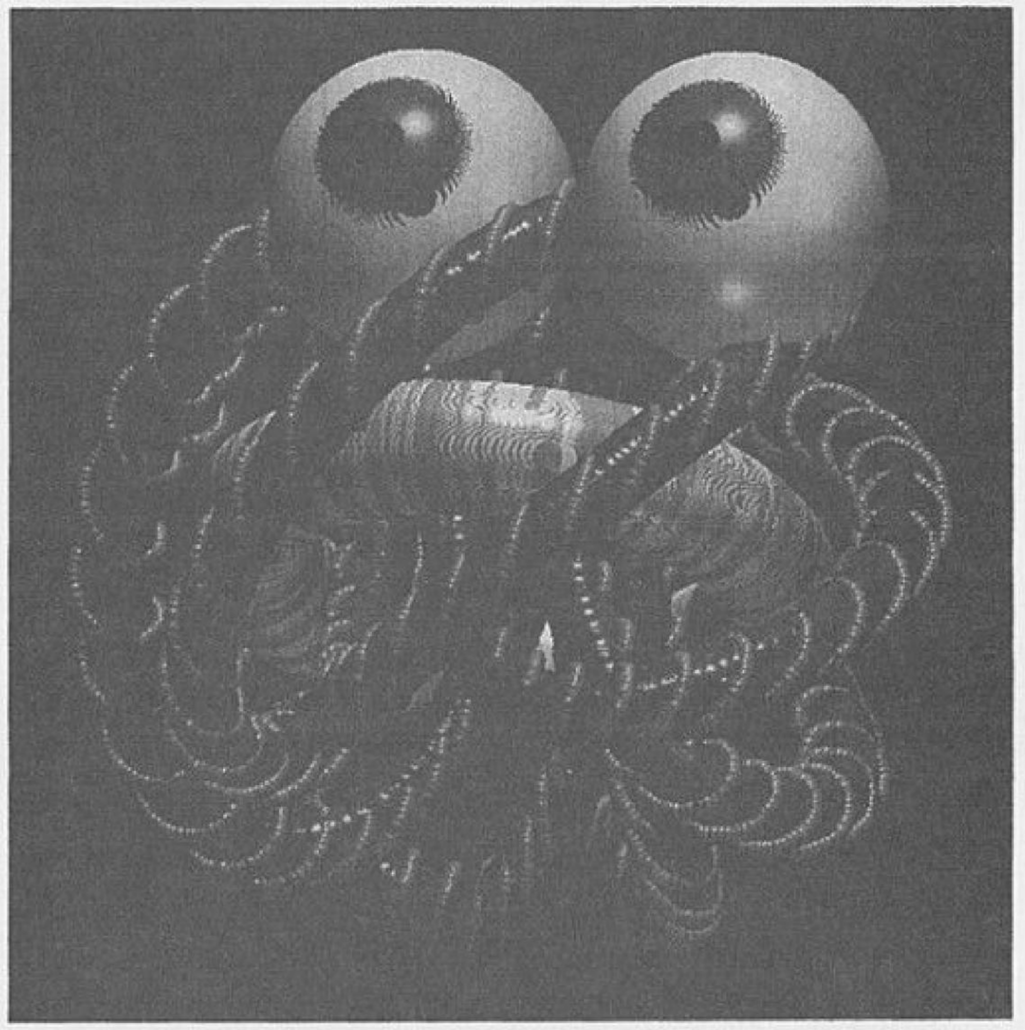
If the properties we assign to the natural world are partly expressions of the way we think and our capacity for understanding, then the introduction of new tools such as the computer will *change* those properties. The computer, like a microscope, expands the range of our senses. The world made visible by the computer seems limitless.

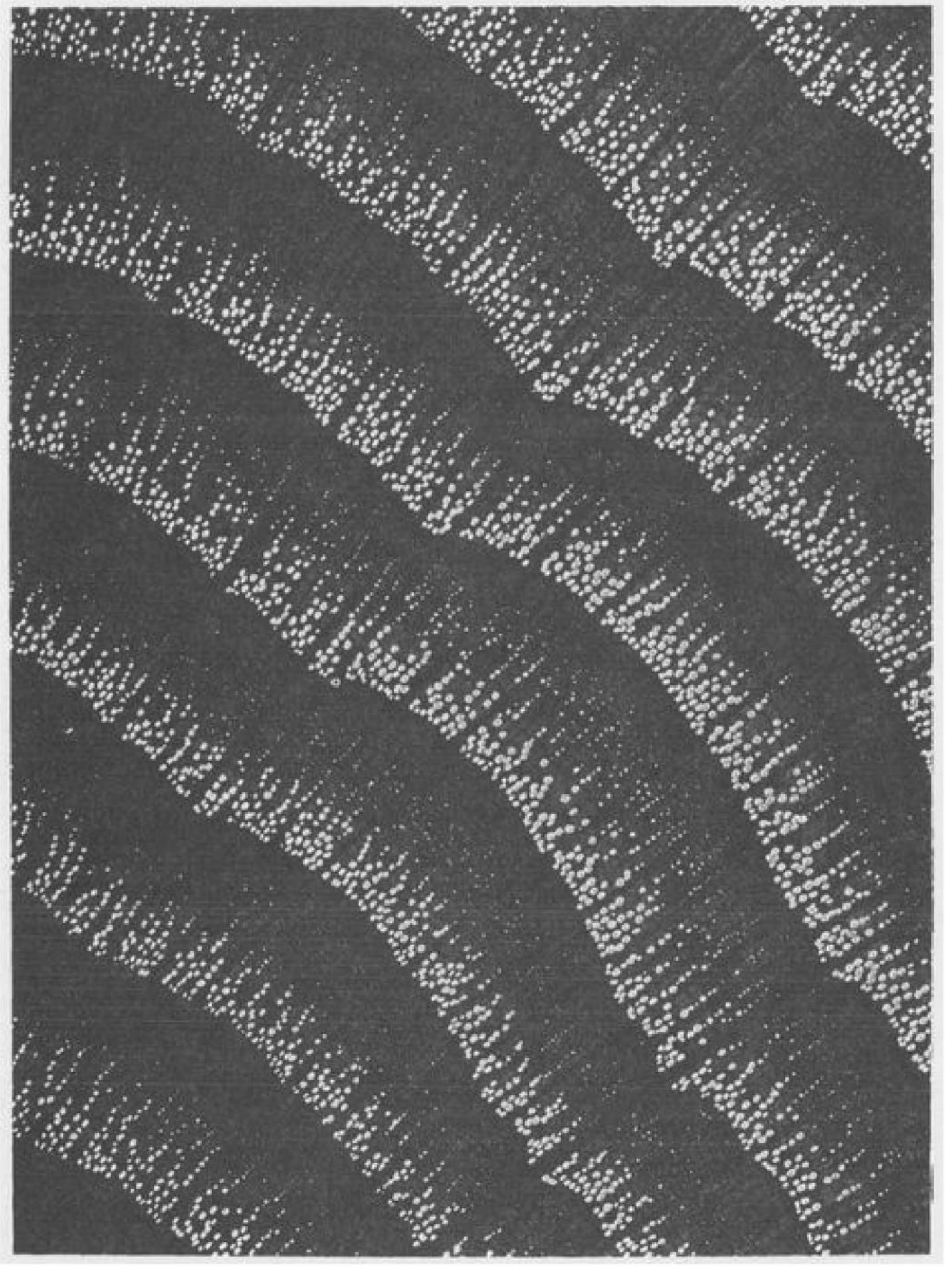
2.8 Reading List for Chapter 2

Gardner (1986) gives additional information on one of the early computing machines mentioned in this chapter (Napier's bones). Mackintosh (1988) describes the early 1939 Atanasoff-Berry computer. For interesting speculative information on the brain, computers, and consciousness, see Rose (1976).

There has been a well-developed history of computer art since the 1960s. Some references to this work are given in "Reading List for Chapter 13" on page 238. Of particular interest is J. Reichardt's book *Cybernetic Serendipity: The Computer and the Arts* which contains a very stimulating collection of papers on the alliance of art and technology. J. Kluger's 1987 article in *Discover Magazine* presents some beautiful artwork from many contemporary computer artists such as Yoichiro Kawaguchi, Jennifer Bartlett, Larry Rivers, Harold Cohen, and Melvin Prueitt, just to name a few. The reader is also directed to the wonderful journal *Leonardo* (2030 Addison St, Suite 400 Berkeley, CA 94704) which is devoted to the interaction of the arts, sciences, and technology. The journal focuses on the visual arts and also addresses music, video, and performance.

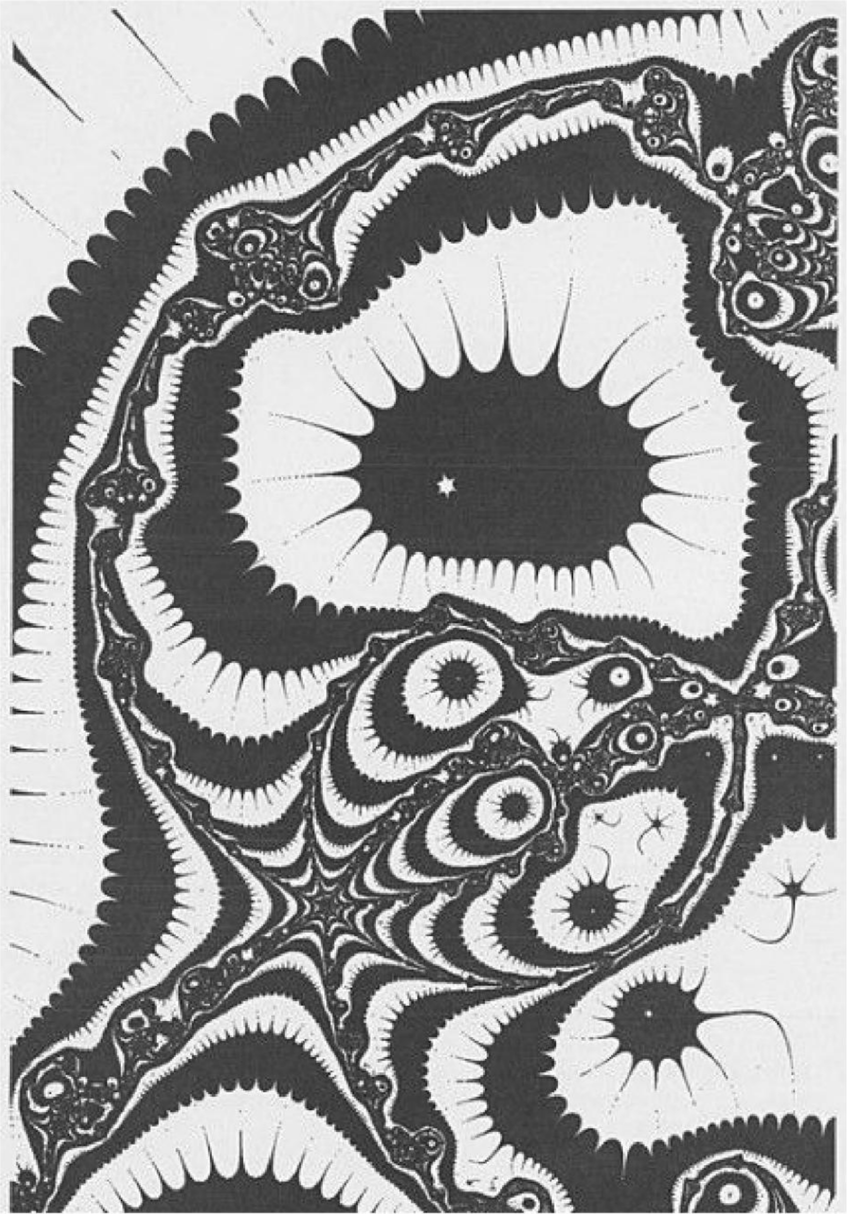
The endpiece figure for this chapter is entitled "A Twilight Friend," and the computer graphic illustration is from a collection of the author's mathematically derived sculptures entitled *I Have Dreams at Night*. Only three trigonometric curves are used to shape the creature's body.





Part II

REPRESENTING NATURE



Chapter 3

Fourier Transforms (The Prisms of Science)

> *"It is indeed a surprising and fortunate fact that nature can be expressed by relatively low-order mathematical functions."*
>
> Rudolf Carnap

> *"Art, literature, and music create order. Science searches for order that already exists."*
>
> Anonymous

We live in a world filled with a maze of aural and visual patterns. Later chapters deal with the characterization of patterns residing in infinitely complicated mathematical worlds. In this chapter we are interested in representing the many facets of nature, that is, in finding patterns in phenomena as diverse as animal vocalizations, music and the genetic sequences of cancer genes. To help characterize this cacophony of complicated data, we first review a famous mathematical technique called the Fourier transformation. Like a prism which separates white light passing through it into its rainbow-colored components, the Fourier transform give us an idea of the hidden components in complicated input. The varied topics in the following sections all have in common the fact that the Fourier transform was used as a tool for finding patterns.

3.1 Fourier Analysis: A Digression and Review

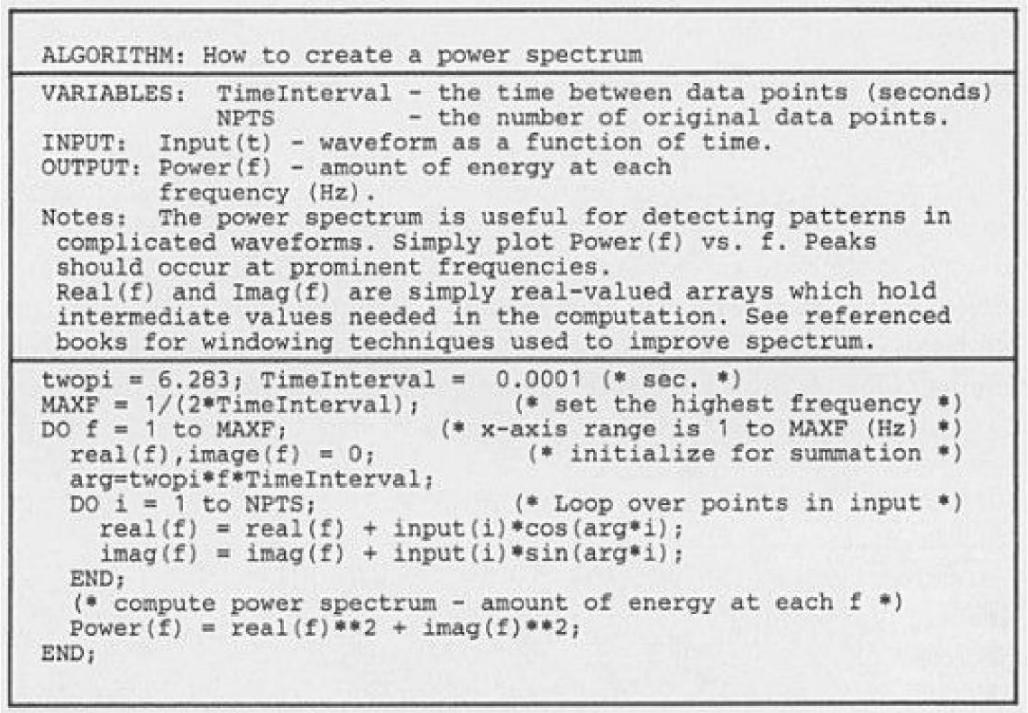
The advent of the personal computer simplifies explorations of the patterns in nature. Let us begin gradually, with the introduction of a simple sine wave. A sine wave is a mathematical function that has unique and important properties. Usually it is graphed as a wiggly curve that periodically rises and falls.

Mathematically, the sine wave has two attributes: amplitude and frequency.[2] A sine wave can be represented by the equation $y(t) = A \sin(2\pi ft)$, where A is the amplitude or maximum displacement of a varying quantity from its average value. *f* is the frequency, which tells us about the number of peaks occurring per second, and *t* represents time.[3] The higher the frequency, the more times the sine wave goes up and down each second. *f* is often expressed in cycles per second or *Hertz*. The sound heard from a loudspeaker producing air pressure varying sinusoidally in time is a "pure tone."

Generally speaking, a waveform can be represented as the sum of a group of sinusoids. To those who are knowledgeable in mathematics, this statement is equivalent to saying that one can represent a wave by a Fourier series. For example, a complicated wave can actually be constructed by summing three sine wavπes: $y(t) = 3 \sin(2 \quad \pi 100t) + 5 \sin(2 \quad 500t) + 10 \sin(2 \quad 1500t)$. This equation represents a Fourier series, named after the French mathematician Joseph Fourier (1768-1830). What Fourier showed is that the behavior of the most complicated wave-shape can be described in terms of sines and cosines. If we had some way to solve for the amplitudes of each frequency component, we could decompose a function into its sinusoidal components and find the amplitude of each component. In fact, solving for the amplitude(s) is relatively simple. The process is called a Fourier transformation, and simple pseudocode is included here for its computation.[4] Usually what is plotted is the energy, or *"power"*, at each frequency; power is calculated from the square of the amplitudes. In the previous equation, all *A*'s are zero, except for three frequencies. A *power spectrum*, computed from the Fourier transform, is visually shouting at the viewer, "hey, your input has repeating features which occur at three different, prominent frequencies." The three peaks indicate three important periodicities: one with frequency 100, one with frequency 500, and another with frequency 1500. For more mathematical information on the power spectrum, see Koopmans (1974). Pseudocode 3.1 shows the necessary steps for power spectrum computation.

[The operation in the inner DO loop is essentially an averaging of the product of the analyzed signal and a sine and cosine of each examined frequency. The method in the pseudocode is more rigorously known as a *discrete Fourier transform* and it allows one to compute the energy of each sinusoidal component with frequency, *f* (Morgan, 1984).]

Of course, most phenomena of nature are not nearly as simple as in the previous waveform example (see speech waveform in next section). Figure 3.1 contrasts an example of a three-dimensional power spectrum for the waveform representing sounds of a speech synthesizer and a human saying the word "seventeen." The 3-D power spectrum is an assemblage of 2-D spectra (such as in the simple sine wave example power spectra) stacked through time to help us see how the occurrence of different frequencies changes through time.[5]



```
ALGORITHM: How to create a power spectrum
VARIABLES:  TimeInterval - the time between data points (seconds)
            NPTS         - the number of original data points.
INPUT:  Input(t) - waveform as a function of time.
OUTPUT: Power(f) - amount of energy at each
        frequency (Hz).
Notes:  The power spectrum is useful for detecting patterns in
 complicated waveforms. Simply plot Power(f) vs. f. Peaks
 should occur at prominent frequencies.
 Real(f) and Imag(f) are simply real-valued arrays which hold
 intermediate values needed in the computation. See referenced
 books for windowing techniques used to improve spectrum.
twopi = 6.283; TimeInterval =  0.0001 (* sec. *)
MAXF = 1/(2*TimeInterval);        (* set the highest frequency *)
DO f = 1 to MAXF;            (* x-axis range is 1 to MAXF (Hz) *)
  real(f),image(f) = 0;            (* initialize for summation *)
  arg=twopi*f*TimeInterval;
  DO i = 1 to NPTS;               (* Loop over points in input *)
    real(f) = real(f) + input(i)*cos(arg*i);
    imag(f) = imag(f) + input(i)*sin(arg*i);
  END;
  (* compute power spectrum - amount of energy at each f *)
  Power(f) = real(f)**2 + imag(f)**2;
END;
```

Pseudocode 3.1. *How to create a power spectrum.*

In the next sections, a variety of applications will be presented in which the 3-D Fourier transform has been used to analyze and detect patterns in data. Though the applications are varied, the idea that binds them is the use of this standard analysis tool in new ways. After one has developed a tool for analysis in a specific area, it can sometimes be applied to totally unrelated fields. For example, in the following section, the synthesis of a singing chorus by computer is discussed. Here, a 3-D power spectrum was used to examine the frequency composition of the various vocal parts.

3.2 Singing Computers

Speech synthesizers are common; we hear talking toasters, microwave ovens, scales, toys, and cars. These may seem like unimportant applications, but as devices become more complex some will become unworkable without speech. As equipment manufacturers begin to add sensors to determine different conditions, hazard monitoring by visual means alone will become difficult and dangerous. New speech products are also aiding the blind (for an excellent review of speech synthesis, see Morgan (1984) and Witten (1982)). Most of these methods have one limitation — they have “canned” or recorded speech stored in their memory chips. Of greater interest are techniques for synthesizing a potentially unlimited variety of continuous speech utterances. With the method of synthesis discussed here, 11 parameters (Figure 3.2 shows these parameters in bubbles) are changed 100 times a second to control the output sound. (Using fewer than 11 parameters produced a less intelligible vocal output.) These parameters control such factors as how loudly vowels are spoken relative to consonants, and also the frequencies of *filters* which simulate various vocal tract resonances known as *formants*. As a useful analogy, think of a coffee filter which lets the coffee liquid go through but retains the coffee grounds. Similarly, an audio filter permits some frequencies to “go through,” while stopping others. Filters consist of mathematical formulae which act somewhat like the tone knob on a hi-fi system — turning the knob, or altering the filter equation, reshapes the frequency characteristics of the resulting sound. The reader may be interested in viewing pseudocode for producing a simple singing sound (Pseudocode 3.2). The code is presented here without a detailed explanation of all the variables. For more information on filter theory, see Witten (1982) and Morgan (1984). [For readers interested in more detail, the filter in the pseudocode produces a waveform with constant center frequency and bandwidth. Bandwidth controls the spread of frequencies. In synthetic singing, both bandwidth and band center vary slowly and smoothly, since they change as a result of motion of articulators (such as the tongue). The filter in the pseudocode is known as a second-order resonance filter.]

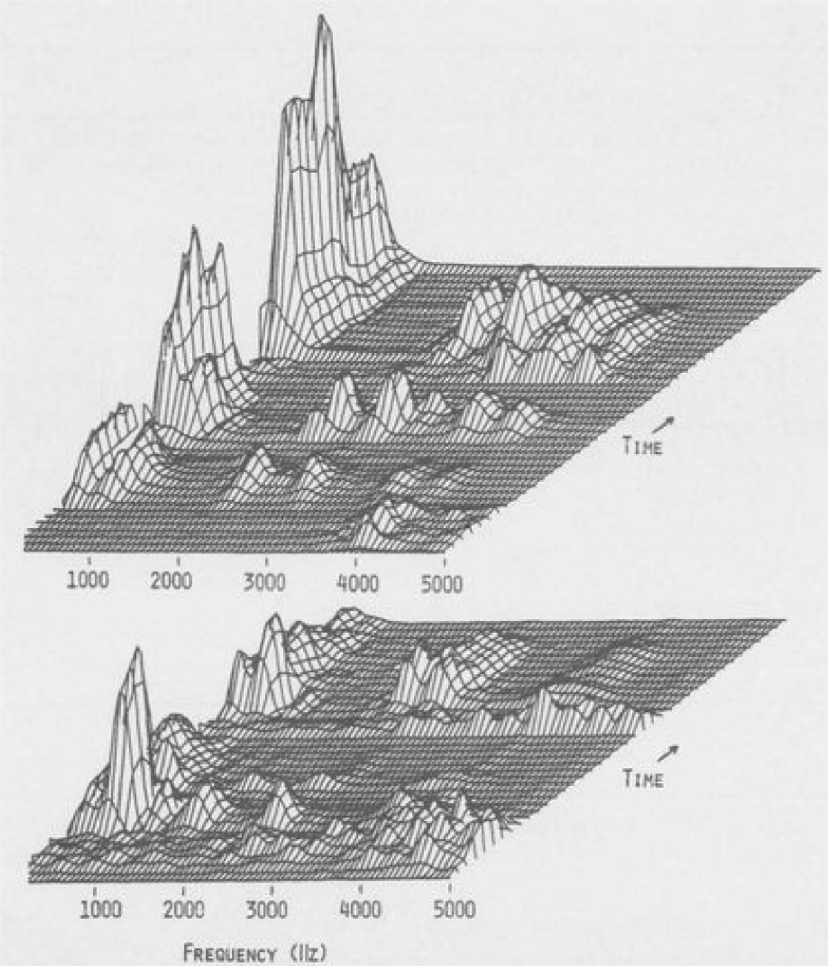


***Figure 3.1**. Three-dimensional power spectrum for speech. Graphs are computed for a synthesizer (top) and human (bottom) saying the word "seventeen." Time goes into the back of the page. The heights of the various maxima give an indication of how loud a particular frequency is. Frequency goes from 0 Hz (left) to 5000 Hz (right). By comparing the various mountain peaks, researchers are able to understand why a synthesizer does not always sound human.*

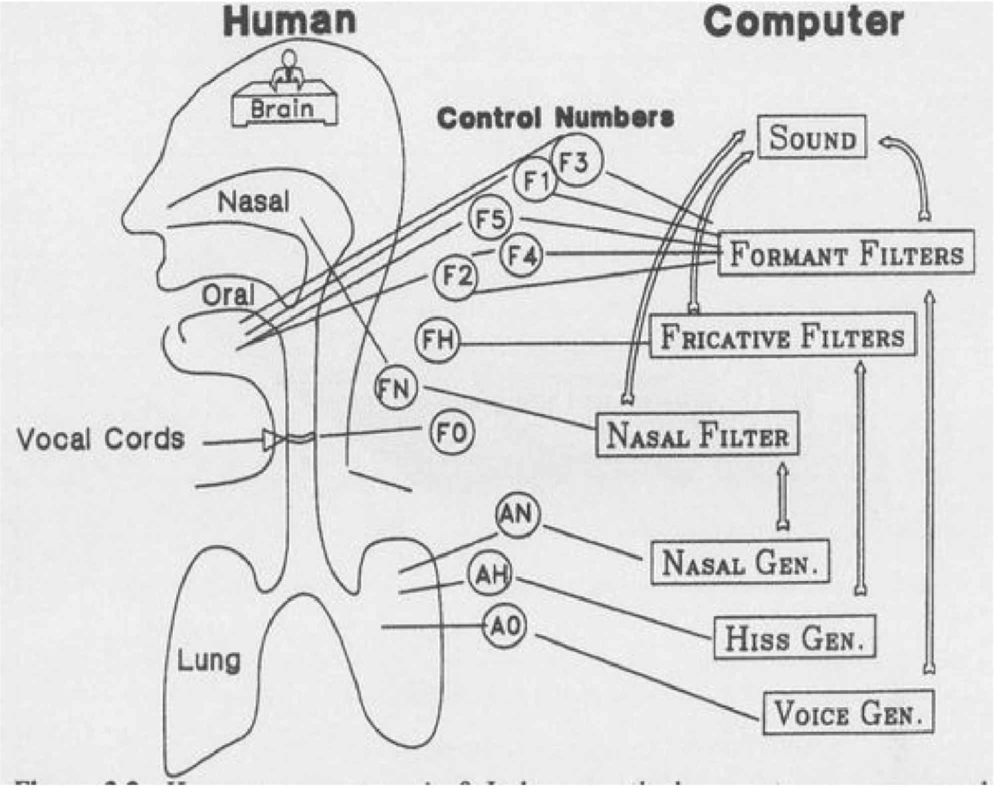
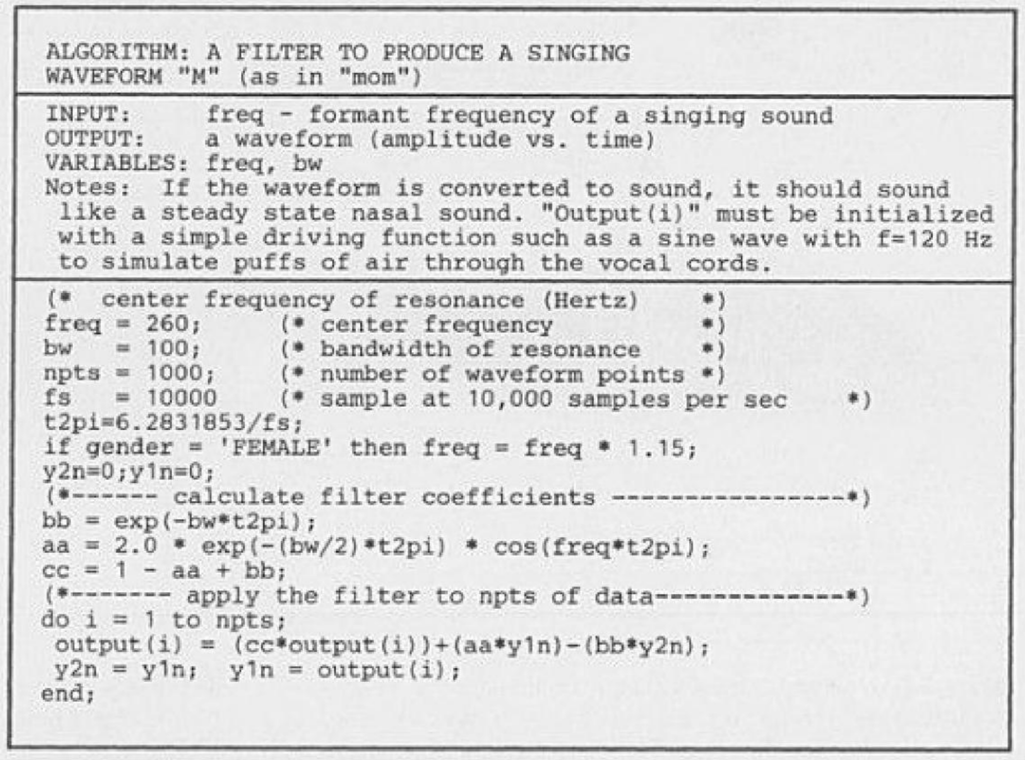


Figure 3.2. How can computers sing? In humans, the lungs act as a power supply forcing air through the vibrating vocal cords. In certain speech synthesizers, simple waveform generators substitute for the lungs. These are marked vowel, nasal, and fricative generators in the diagram. Filters shape the waveform further (taking the place of the changing vocal tract shape).

Speech production uses generating functions which simulate the air pressure waves produced by the lungs. These simple pressure waveforms are then shaped by filter functions in order to produce a more complicated speech-like signal. In other words, the generator functions simulate the pulses of air through the lungs and mouth. [For those already familiar with this field’s terminology, the 11 control bytes which guide the synthesizer are: AN (nasal amplitude), FN (frequency of nasal resonance), F1, F2, F3, F4, F5 (frequencies of formants 1,2,3,4 and 5), A0 (voice amplitude), AH (hiss amplitude), FH (primary fricative frequency), F0 (fundamental frequency), and C0 (binary data, controlling: aspiration/frication, formant bandwidths (f1, f2, f3, f4, f5), and hiss modulation).

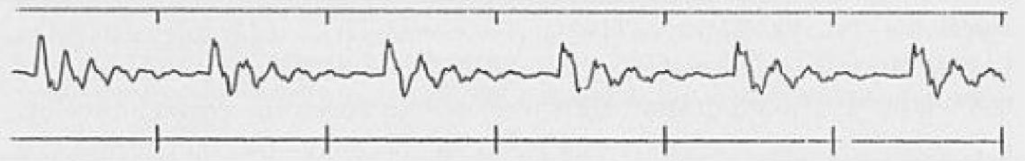


```
ALGORITHM: A FILTER TO PRODUCE A SINGING
WAVEFORM "M" (as in "mom")

INPUT:      freq - formant frequency of a singing sound
OUTPUT:     a waveform (amplitude vs. time)
VARIABLES: freq, bw
Notes:  If the waveform is converted to sound, it should sound
 like a steady state nasal sound. "Output(i)" must be initialized
 with a simple driving function such as a sine wave with f=120 Hz
 to simulate puffs of air through the vocal cords.

(*  center frequency of resonance (Hertz)       *)
freq = 260;      (* center frequency            *)
bw   = 100;      (* bandwidth of resonance      *)
npts = 1000;     (* number of waveform points *)
fs   = 10000     (* sample at 10,000 samples per sec      *)
t2pi=6.2831853/fs;
if gender = 'FEMALE' then freq = freq * 1.15;
y2n=0;y1n=0;
(*------ calculate filter coefficients ----------------*)
bb = exp(-bw*t2pi);
aa = 2.0 * exp(-(bw/2)*t2pi) * cos(freq*t2pi);
cc = 1 - aa + bb;
(*------- apply the filter to npts of data-------------*)
do i = 1 to npts;
 output(i) = (cc*output(i))+(aa*y1n)-(bb*y2n);
 y2n = y1n;  y1n = output(i);
end;
```

Pseudocode 3.2. *A filter to produce a singing sound.*

This sequence of control data, after having been passed through the synthesizer and a digital-to-analog converter, results in speech output.]

A typical speech waveform (amplitude vs. time) looks like:



In order to establish the feasibility of using a synthetic speech system for producing a high-quality singing ensemble, the vocal parts of an excerpt of Handel's Hallelujah chorus were generated and mixed. The 3-D power spectrum (energy vs. frequency vs. time, discussed in the last section) was useful in analyzing and shaping the sounds (Figure 3.3). As suggested in the last section, one can look at the various spectral prominences and see the overall frequency composition and the relative loudness of vowels and consonants. One can also make comparisons with human speech. The spectrum from one of the synthetic voices is included so that the reader can visualize the complexity of the various mountain peaks representing the frequencies in the singing voice.

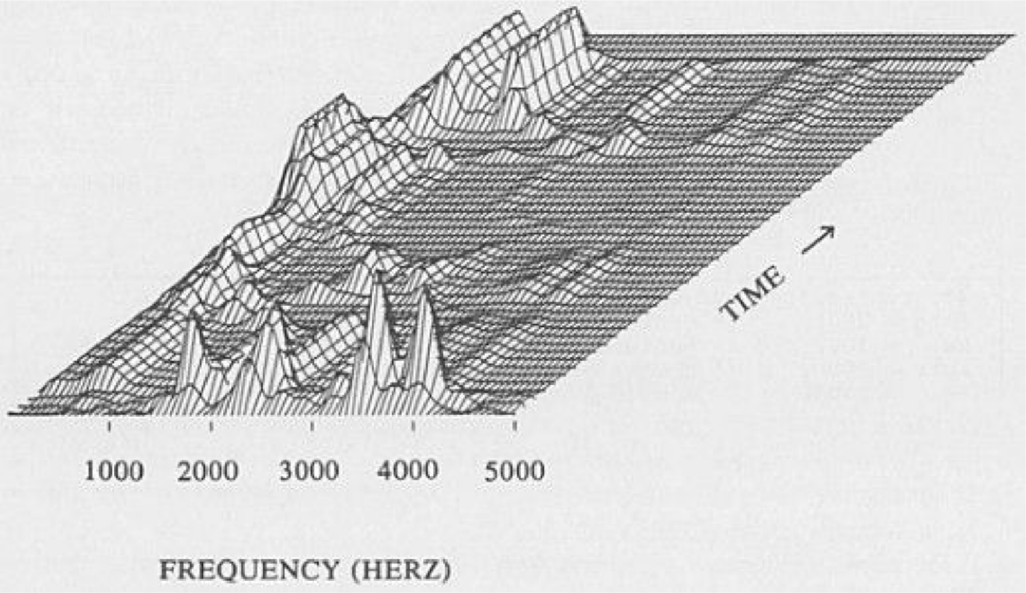


Figure 3.3**.** *Computer chorus. This three-dimensional map was used to analyze the synthetic singing of the Hallelujah chorus. In particular, this was computed for the synthetic bass singing voice: "Hal-le-lu-jah!" Again, loudness of the different frequencies is represented by height of the various hills.*

The Hallelujah chorus is a piece from *The Messiah*, Handel's most successful and best-known oratorio, composed in the year 1741. The soprano, alto, tenor, and bass voices were each created separately and subsequently combined with a multi-track recorder. Short pieces introducing the work have appeared in popular magazines such as *Omni* (Rivlin, 1986) and business journals such as *Voice News* (Creitz, 1984). The most detailed report appears in

Computer Tech. Rev. (see references). For more information on precisely how to generate the chorus, see these articles. Through this interesting exercise, it is apparent that with the increasing availability and improvement of voice synthesis technology, singing synthesizers may herald the next *revolution* in music, much as did the electronic music synthesizers of the 1970s.

Mormon Tabernacle Choir, watch out! Even the most proficient soprano or "basso profundo" is limited in the range, duration, and timbre of notes able to be generated due to the physical constraints of the vocal apparatus. The machine discussed here (Figure 3.2) has no such limits, and future composers will no doubt create songs which only synthesizers can sing. As stated earlier, not only will computers be singing songs, but they will be writing the songs they sing. In a preliminary exploration of this concept, programs were created which produce lyrics from lists containing 10 categories of English parts of speech (nouns, verbs, adjectives, etc.). Simple grammatical rules were used. Subsequently these lyrics were spoken by a commercially available voice synthesizer (Speech Plus PR2020 speech synthesizer) and mixed with several musical tracks. As strange as these songs sounded, they're just the first step. Using graphics, non-human, vowel-like sounds were constructed (by changing the vocal tract resonances), and it is clear that such novel near-human sounds can provide the musician with an entirely unexplored milieu within which to work. Perhaps similar speech systems will be developed for use by individuals dedicated to applying the resources of modern technology to the needs and problems of contemporary musical expression. (Pick88b, *Computer Tech. Rev*.)

3.3 Bach, Beethoven, The Beatles

In the last section, graphical representations of the 3-D Fourier transform were presented for synthetic and human speech sounds. Another useful application of the 3-D Fourier Transform is in the area of musical score representation. In this application the output spectrum is plotted as intensity vs. frequency vs. position in the melodic sequence of notes (explained below).

Various techniques of music visualization, music transcription, melody storage, and melody matching have been proposed (see Mitroo, 1979, and Dillon and Hunter, 1982). However, none of these methods has had as their primary focus the mathematical characterization of melody patterns using an interactive graphics system with a wide variety of controlling parameters. Spectra have been used to characterize instruments, voices, and large mixed musical ensembles (Cogan, 1984). In the work described in this section, a research system was developed which accepts as input a coded version of the musical score and subsequently computes digital spectrograms and topographic spectral distribution functions (another name for "power spectra") of melodic sequences.

While the power spectral analysis of instrument sounds, such as violins, has significantly advanced our understanding of psychoacoustics, it should be stressed that the methods presented in this section are sensitive to the periodicities in the melodic pitch sequence, and they do not have as an input a traditional acoustic sound source. The computer is not analyzing time waveforms as in the last section. Therefore, such parameters as loudness, attack, and timbre are not characterized. What is analyzed is something akin to the frequencies of the progression of hills and valleys of the musical score (which themselves represent fundamental frequencies of notes on a keyboard). The sounds are not analyzed.

Both the spectrogram and 3-D power spectrum present melodic sequence data in a way which allows patterns to be visually detected. As input, the ups and downs of a score were used (thereby producing an input resembling a picket fence). As output, 3-D maps were computed. A range of classical and contemporary pieces were tested. Duration-weighted mean and standard deviation of the input sequence of notes can be reported; for example, of the tested pieces, standard deviations ranged from a low of 49 Hertz for “Let the Sun Shine In” (from the rock opera “Hair”) to a high of 371 Hertz for “Flight of the Bumblebee” (by Rimsky-Korsakoff). The resultant displays contain a rich variety of spectral features. The various mountain peaks indicate prominent periodicities and patterns in the musical score. The classical pieces tested appear to have a greater number of spectral prominences than do the more contemporary ones, and of the seven tested musical sequences, the Bach piece shows the greatest spectral amplitudes — followed closely by Chopin (Figure 3.4). Interestingly, the music of Frederic Chopin has been previously represented by a simple bar graph of the notes on the musical staff (by D. Hofstadter in *Scientific American*). His graph also suggests visually distinctive, regular patterns of the melodic sequence. In general, both Bach and Chopin (who revered Bach’s music) were especially interested in the forms and patterns of music.

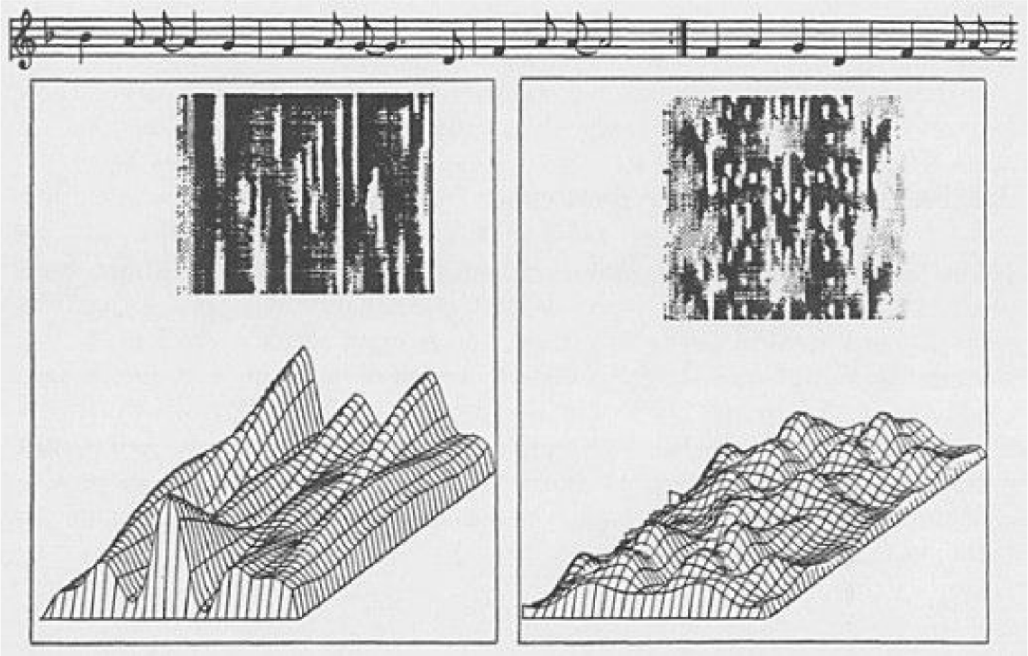


Figure 3.4. *Musical scores. The hills and valleys of musical scores (top) can be analyzed with the same tool used in the previous figures. Top: spectrogram. Bottom: 3-D power spectrum. Here, the three-dimensional plots' axes are: intensity, frequency, and position in the melodic sequence of notes. Songs shown are Cantata No. 96 Aria. Ach, ziehe die Seele mit Seilen der Liebe, by J. S. Bach (left), and The Entertainer (a Ragtime Two-Step), by Scott Joplin (right). High frequency patterns are shown towards the right of each of these "fingerprints."*

Many music researchers have yet to incorporate the power of the computer in their theorizing, and by far the most prevalent methods of computer analysis have involved the simple tallying of such features as "the number of C-sharps" in a composition (for a recent review of the research on music and artificial intelligence, see Roads (1985)). While there are theories to describe musical patterns and progressions, no current methods exist which truly distinguish compositions which will touch man's spirit from those which will not. However, it may be possible to use spectra, like those described here, as digital "fingerprints" for either certain historical musical eras or for certain composers, much as similar analyses are used in forensic voice identification and authentication.

Theories of musical quality are still primarily descriptive and do not allow one to create a new piece of extreme aesthetic interest. Those knowledgeable in noise theory will appreciate R. Voss's demonstration that melodies generated using $1/f$ noise generators produce progressions closest to "real" music when = 1 (compared with = 0 or = 2).[6] However, none of the results would be judged as a sophisticated composition of a specific type of music. With the analysis routines presented here, we would certainly predict that a power spectrum with just one spectral peak corresponds to an input melody which is boring. Would it be possible to state that a certain number of peaks per unit time is most appealing for most people? This question remains unanswered. However, it may be possible to start with the 3-D map, isolate prominent spectral peaks, shift minor ones, and then inverse transform the spectra to create a new piece with characteristics of a particular author or musical era. (For more information on these topics, see Pick86b, *Computer Music J.*)

3.4 Breathing Proteins

This book's Introduction discusses the lateral use of computer software tools. The topics in this section and the previous section are good examples. The 3-D surface produced by a Fourier transform is used commonly in speech analysis. In this section, the same tools are used to study the "breathing motions" of proteins where the protein size-changes are represented as a waveform input to a Fourier transform.

As background, proteins are the structural building blocks of life and the catalysts for life's chemical reactions. Like all matter at room temperature, the protein molecules are continually vibrating due to thermal energy (M. Karplus has conducted extensive work in this area; see, for example, Karplus and McCammon (1979)). When large groups of atoms in the protein move in unison, this is sometimes referred to as "breathing" of the protein molecule. A complete description of a globular protein requires not only a static three-dimensional x-ray structure, but also an understanding of its flexibility and the role that structural fluctuations play in the protein's function. Two useful ways of describing the frequency composition of the breathing motions of globular proteins are the spectrogram and three-dimensional power spectrum, representations similar to those frequently used in the field of speech analysis (see sections "Fourier Analysis: A Digression and Review" on page 21 and "Singing Computers" on page 23). In this section, we are most interested in low frequency vibrations of globular proteins which correspond to the collective oscillations of atoms from many different amino acids. Fluctuations in radii of gyration (R_g) (defined below) provide a sensitive way to characterize such concerted motions of proteins. One protein of interest is bovine pancreatic trypsin inhibior (BPTI), a small globular protein of molecular weight 6,500 daltons, consisting of one polypeptide chain with 58 amino acids and three disulfide bonds (Figure 3.5).

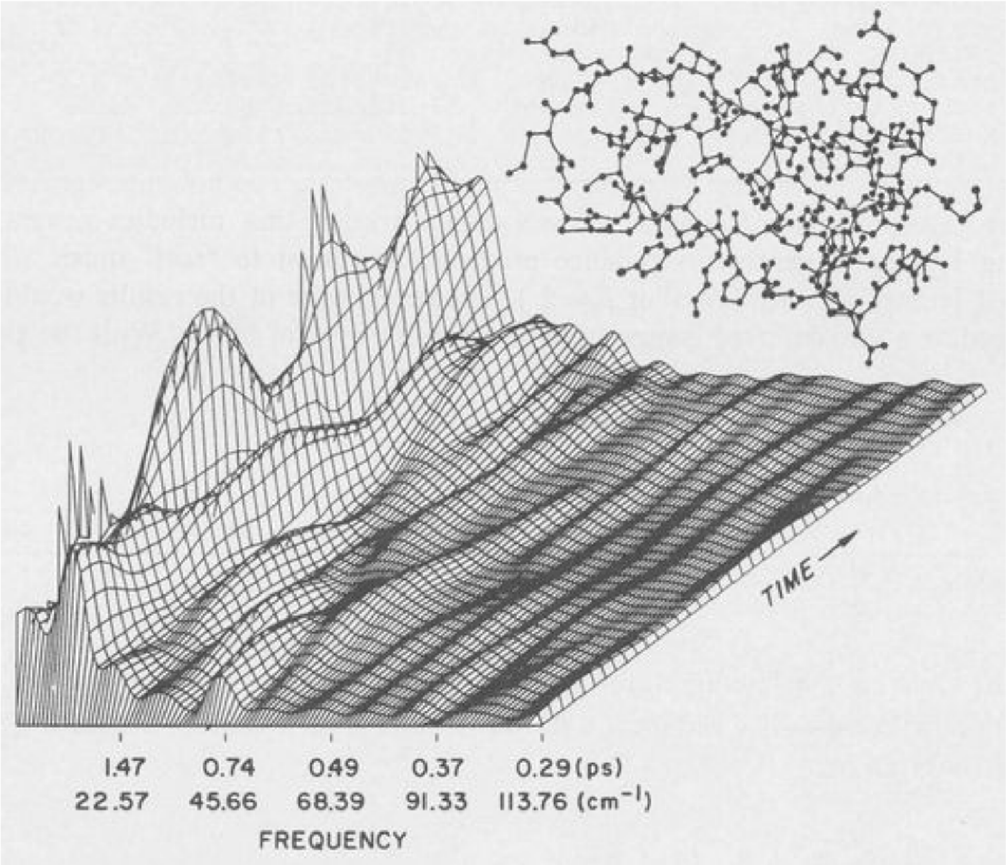


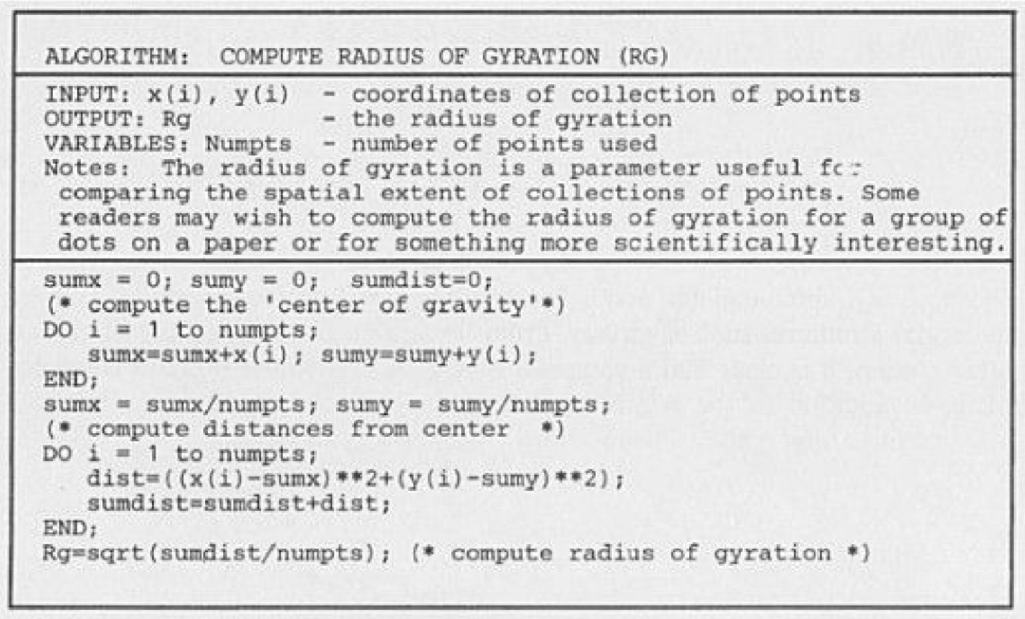
Figure 3.5. *Representing the motions of a breathing protein. This three-dimensional map (bottom) computed from a Fourier transform gives an indication of the fluctuations in the shape of a small protein, bovine pancreatic trypsin inhibitor (top). 96 picoseconds (1 ps = 1×10^{12} s) of data are represented in this plot.*

The radius of gyration of an object is an interesting physical measurement. It gives an idea of an object's spatial extent and shape (see Pseudocode 3.3).[7] R_g is in the same units as the input, for example, inches or angstroms. For proteins, R_g changes with time. The radius of gyration of BPTI, derived from atomic coordinates, is given by:

$$R_g^2 = \left[\frac{\sum Z_i a_i^2}{\sum Z_i} \right]$$

(3.1)

where Z_i is the atomic number for atom *i*, increased by the number of attached hydrogen atoms. The sums are over all atoms. a_i is the distance of atom i from the electronic center of gravity. Equation (3.1) is described in detail in *Science* magazine. [8] The reader may be interested in computing R_g for a collection of dots. Figure 3.5 shows a stick-figure diagram of the protein where dots represent atoms.



```
ALGORITHM:  COMPUTE RADIUS OF GYRATION (RG)

INPUT: x(i), y(i)   - coordinates of collection of points
OUTPUT: Rg          - the radius of gyration
VARIABLES: Numpts   - number of points used
Notes:  The radius of gyration is a parameter useful fc:
 comparing the spatial extent of collections of points. Some
 readers may wish to compute the radius of gyration for a group of
 dots on a paper or for something more scientifically interesting.

sumx = 0; sumy = 0;  sumdist=0;
(* compute the 'center of gravity'*)
DO i = 1 to numpts;
   sumx=sumx+x(i); sumy=sumy+y(i);
END;
sumx = sumx/numpts; sumy = sumy/numpts;
(* compute distances from center  *)
DO i = 1 to numpts;
   dist=((x(i)-sumx)**2+(y(i)-sumy)**2);
   sumdist=sumdist+dist;
END;
Rg=sqrt(sumdist/numpts); (* compute radius of gyration *)
```

Pseudocode 3.3. *Compute radius of gyration.*

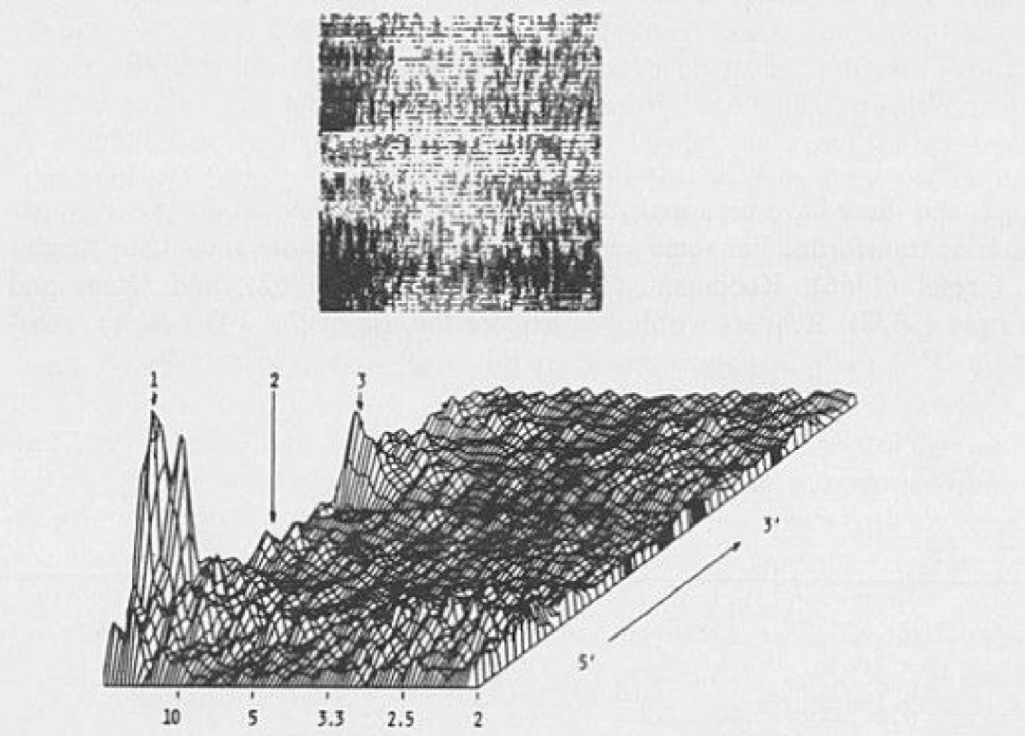
A 3-D power spectrum computed for the R_g fluctuations (Figure 3.5) indicates that most of the power is below 1 picosecond, with a particularly prominent breathing mode centered at 3 picoseconds. Higher frequencies are evident to a lesser degree. The high ridge close to zero frequency may correspond to a slower radial oscillation or an infrequent process which is not observed long enough for adequate characterization in this conventional molecular dynamics simulation. Longer simulations would be required to determine the significance of such a slower oscillation. There is experimental evidence for the existence of low frequency breathing vibrations in other proteins.

Since both the spectrogram and 3-D power spectrum present breathing motion data in a way which can easily be understood by the biophysicist, characterization of the dynamical richness of proteins is greatly facilitated. Why do we care about breathing motions? It has been hypothesized that the motions of globular proteins play an essential role in their function and may affect a number of important processes such as: binding of ligands, enzyme catalysis, hemoglobin cooperativity, immunoglobin action, electron transfer, and the assembly of supra-molecular structures such as viruses. From the work described in this section and other studies, it is clear that a compact, rigid view of globular proteins is incomplete. In addition to the relatively fast processes including collisions between neighboring atoms and localized group vibrations, proteins may undergo somewhat regular low frequency breathing motions of varying complexity. These motions involve the collective motion of a large number of different atoms. The functional importance of such breathing modes, and protein motion in general, has begun to attract the interest of an increasing number of physical chemists as evidenced by the growing number of spectroscopic, kinetic, and theoretical studies of protein dynamics. (Much of the material in section 3.4 appeared in: Pick84a, Science. Related material on protein dynamics and conformation appears in: Pickover et al., 1979, *J. Biol. Chem.;* Pickover and Engelman, 1982, *Biopolymers;* McKay, Pickover, and Steitz, 1982, *J. Mol. Biol*.; Pickover and Engelman, 1982, *Biophys. J.;* Levinson, Pickover, and Richards, 1983, *J. Biol. Chem*.)

3.5 Cancer Genes (DNA Waveforms)

The 3-D Fourier transform representations may be applied to genetic sequences. In this work, the sequence of bases in a human bladder cancer gene is treated as if it were a speech waveform. As background, DNA contains the basic genetic information for all life on earth and is expressed in a four letter code: A, C, G and T (adenine, cytosine, guanine, and thymine). By associating each letter to a number we can treat the DNA sequence as a waveform, thereby opening up the whole array of speech analysis tools for molecular genetics.

The search for patterns in the sequence of bases in long DNA sequences is an active topic in molecular graphics. Periodicities and various patterns affect physical, chemical and biological properties of the DNA. An example of the output of a graphics system for an actual DNA sequence is presented in Figure 3.6. The calculation was performed for a 4000 base human bladder oncogene sequence. Oncogenes have been detected in tumors representative of each of the major forms of human cancer, and some have been shown to be able to induce malignant transformations in certain cell lines. This bladder carcinoma oncogene is derived from a sequence of similar structure present in the normal human genome (Reddy, 1983). Several prominent features can be seen on the map, and, interestingly, these features correspond to biologically important areas of the DNA sequence. The largest peak (1) occurring roughly between bases 590 and 900 corresponds with the sequence which, when deleted, drastically reduces the transforming activity of the oncogene, indicating the crucial role played by this non-coding sequence.



```
GGATCCCAGCCTTTCCCCAGCCCGTAGCCCCGGACCTCCGCGGTGGGCG    50
GCGCCGCGCTGCCGGCGCAGGGAGGGCCTCTGGTGCACCGGCACCGCTGA   100
.
.
.
etc.
```

Figure 3.6. *A cancer gene. Another three-dimensional map (amplitude vs. frequency vs. position in sequence) computed for the DNA sequence of a 4000 base human bladder cancer gene (bottom). The three peaks (1,2,3) are areas of biological interest. The spectrogram (inset), portrays position in sequence (ordinate) vs. frequency (abscissa). Amplitude is indicated by darkness on the plot.*

It may be possible to discover interesting periodicities in the DNA sequence by having the program produce many DNA maps by automatically iterating through a large number of different base-to-number assignments for input parameters (e.g. G=1, C=1, A=0, T=0; G=1, C=2, A=3, T=4 etc.). These assignments can be based on relative molecular weight, electrostatic potential, or other physical parameters. In this way, the program may suggest to the human analyst important features and parameters which would not even be considered otherwise. (More information can be found in Pick89, *Speculations in Science and Tech.;* Pick84b, *J. Mol. Graphics*.)

3.6 Reading List for Chapter 3

The history and theory of Fourier analysis has been extensively covered in the literature, and there have been many excellent books published on the practical use of Fourier transforms. For some general overviews, the reader should see Bendat and Piersol (1966), Koopmans (1974), MacDonald (1962), and Otnes and Enochson (1978). Readers wishing to explore the use of the 2-D Fourier transform for DNA pattern analysis should see Silverman and Linsker (1986).

There is also an extensive literature on speech science and computer synthesis of speech. For further information on human speech pathology see Borden and Harris (1983), and Ladefoged (1982). For information specific to the computer synthesis of speech, see Dixon and Maxey (1968), and Morgan (1984). For additional information on the visual representation of musical signals, see Cogan, R. (1984). A personal favorite on the subject of music and acoustics is: Pierce, J. (1983) *The Science of Musical Sound.* Scien. Amer. Library: New York. The reader may also consult *The Computer Music Journal* published by MIT Press.

The bibliography at the end of this book has additional references.

Chapter 4

Unusual Graphic Representations

> *"Who knows what secrets of nature lay buried in the terabytes of data being generated each day by physicists studying the results of numerical simulations or the image of a distant galaxy. Given the volume and complexity of scientific data, visualization in the physical sciences has become a necessity in the modern scientific world."*
>
> Robert Wolff

In the last chapter, we saw some brief examples of how a Fourier transform could be used to capture patterns in data. The various mountain peaks indicated the frequency composition of data ranging from speech to breathing proteins. Unfortunately, the traditional techniques don't always distinguish potentially interesting features in input data. As an example, let's first consider speech. With the traditional Fourier representation, many perceptibly different sounds may give rise to only very subtle differences in the spectra. There has been past research which points out the limitations of the Fourier method in displaying acoustic features which are of importance in auditory perception. Such limitations naturally motivate the development of novel display techniques to help capture subtleties which may be difficult to see in the traditional displays.

This chapter includes several novel ways of displaying data which are applied to a range of fields including acoustics and genetics. Note that this chapter does not attempt to give a detailed historical background to the visual display of quantitative information. For background, the reader should consult the various books in the reference section, and in particular, the works of Tufte (*Graphics Press*, 1983), Wainer and Thissen (*Ann. Rev. Psychol.*, 1981), and Wolff (*Comput. in Sci.*, 1988) (all cited in "Recommended Reading" on page 349). Also note that some of the display methods devised by the author are new and speculative; however it is hoped that many of the methods presented here will stimulate other researchers to extend and further test these techniques in related fields in order to assess their usefulness.

4.1 Acoustics

From the dull, stentorian roar of a lion to the clanging of a cathedral bell, the remarkable range of audible sounds makes analysis exceedingly difficult. It is difficult to rigorously compare and characterize sounds by ear alone since the listening process is subject to the limitations and artifacts of both memory and perception. Also, there are individual variations in listeners' ability to localize and describe acoustic features. This problem is the primary motivation for graphic displays of speech. Some novel ways of graphically representing speech waveforms in order to capture information missing in the spectrogram will be discussed in the following sections; these include *phase-vectorgrams* displaying phase, frequency and amplitude in a cylindrical plot resembling a pipe cleaner, and *autocorrelation-faces* displaying speech in visually memorable ways for children.

The speech waveform is a complex entity which is difficult to manage, manipulate and characterize (see example waveform, "Singing Computers" on page 23). Simply recording the pressure variation over time in the acoustic signal generated by human speech produces a complicated waveform. The signal itself alternates between quasi-periodic vowel-like sounds, which often look something like smooth rippling ocean waves when plotted, and certain consonants "looking" much like plots of random noise. Scattered through the signal are rapidly occurring high-energy pops known as plosives interspersed by perceptually important silences. While traditional graphic analyses, such as the spectrogram (intensity vs. frequency vs. time), have been invaluable in showing the general frequency content of an input signal, sometimes it is difficult for users to see on the spectrogram differences which are perceptible to the ear. These difficulties motivate representations which can make subtle differences in input signals obvious to the human analyst. First, a review of data display methods in general.

4.2 New Ways of Displaying Data

The use of visual displays to present quantitative material has a long history. There are many examples in the chronicle of science where important phenomena have been detected using visual displays and have heralded the emergence of entire new fields of scientific endeavor.[9] The usefulness of a particular display is determined by the embodiment of desirable characteristics such as descriptive capacity, potential for comparison, aid in focussing attention, and versatility. New representations called iconic graphs are now being explored. In contrast to the most common graphs which are restricted to two or three dimensions, "icons" (or symbols) such as computer-generated faces are now sometimes used to represent multidimensional data. With icons, the data parameters are mapped into figures with *n* features, each feature varying in size or shape according to the point's coordinate in that dimension. Such figures capitalize on the feature-integration abilities of the human visual system. Icons will be described in more detail in following sections.

4.3 Snowflakes from Sound: Art and Science

Of the many displays of acoustical data developed, one of the most striking and colorful data-display techniques produces figures with the six-fold symmetry of a snowflake. The trick is to convert sound waves (or any data) into a collection of dots which are then reflected through mirror planes by a simple computer program. The resulting representation, a *symmetrized dot-pattern* (SDP), provides a stimulus in which local visual correlations are integrated to form a global percept. It can potentially be applied to the detection and characterization of significant features of any sampled data. The symmetry, color and redundancy of the dot-pattern is useful in the visual detection and memorization of patterns by the human analyst.[10] The 1986 *J. Acoust. Soc. Am.* paper describes a simple recipe for taking points on a speech amplitude-time waveform and computing the pattern.[11]

Figure 4.1 shows a symmetrized dot-pattern (SDP) for the "EE" sound of a human and a synthesizer producing the same vowel sound. Since SDPs can be considered, to a first approximation, merely a replotting of the time waveform, it could be suggested that one would do as well to "look" at the waveform to compare and contrast signals. However, as indicated by the superimposed input signals, waveform similarities can often obscure differences. Figure 4.2 shows some more SDP examples — three different occurrences of the sound "OO" as in "boot" spoken by three people. Despite sensitivity to speaker individualities, SDPs have a global similarity for all "OO's." SDPs may also help differentiate nasalized and nonnasalized sounds (Figure 4.3). In general, it is hoped that SDPs can supplement traditional analysis to make for faster detection and diagnosis of certain important features in data.

To implement a symmetrized dot-pattern on a personal computer, start with a digitized waveform. The waveform may represent sound where the jagged trace on a graph indicates how the sound's loudness changes through time. The data is mapped to a snowflake-like pattern by comparing the loudness of pairs of adjacent points and plotting the result on a polar coordinate graph (a graph that looks a little like a polar view of the earth, with the North Pole at the graph's center). The points are then reflected, as though looking at them through a kaleidoscope (see Pseudocode 4.1). The correlations (relationships) between adjacent pairs of points determine the structure of the SDP.

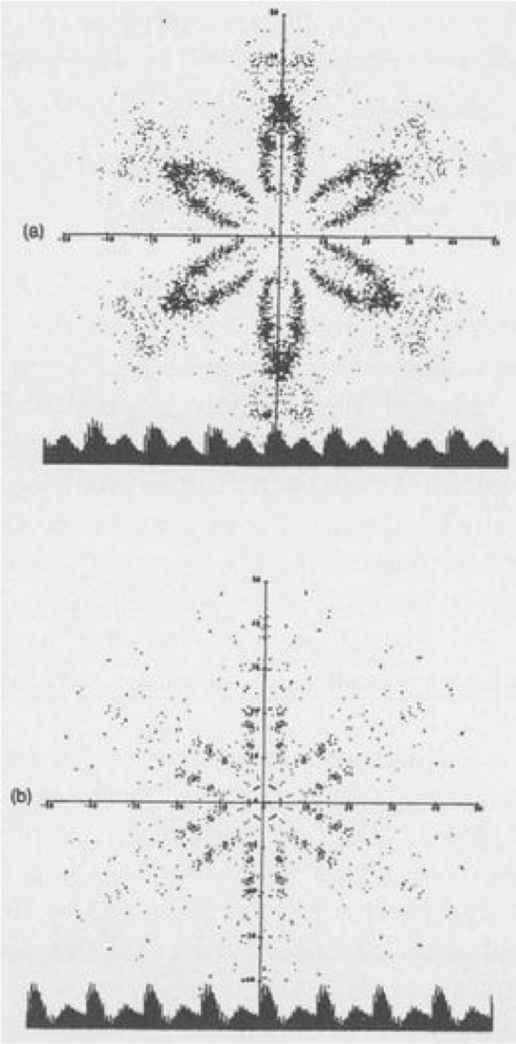


***Figure 4.1**. Symmetrized dot-patterns for “EE” vowel sound. Human-made “EE” sound (top) and a synthesized version of the same sound (bottom). Despite similar waveforms, symmetrized dot-patterns clearly show differences. (This figure and several others in this section appeared in J. Acoust. Soc. Am. (Pick86d).)*

The concentration in this section is focused on human speech sounds, but the SDP can also be applied to handwriting, and musical and animal sounds. To give the reader an indication of the variety of patterns sound can produce, Figure 4.4 shows symmetrized dot patterns computed from animal vocalizations. Of the animal vocalizations cataloged, dolphin-SDP were most similar to human-SDPs. Since dolphins have a developed and complex acoustic repertoire for communication and echolocation, SDPs may be of use in the visual characterization of their vocalizations. See (Pick86d, *J. Acoust. Soc. Am*.) for the use of the SDP to assess frequency content and waveform variability, and for the various pros and cons of the SDP approach for representing waveforms. See Pseudocode 4.1 for SDP color options, which make SDPs more useful in detecting data features (and more interesting from an artistic standpoint).

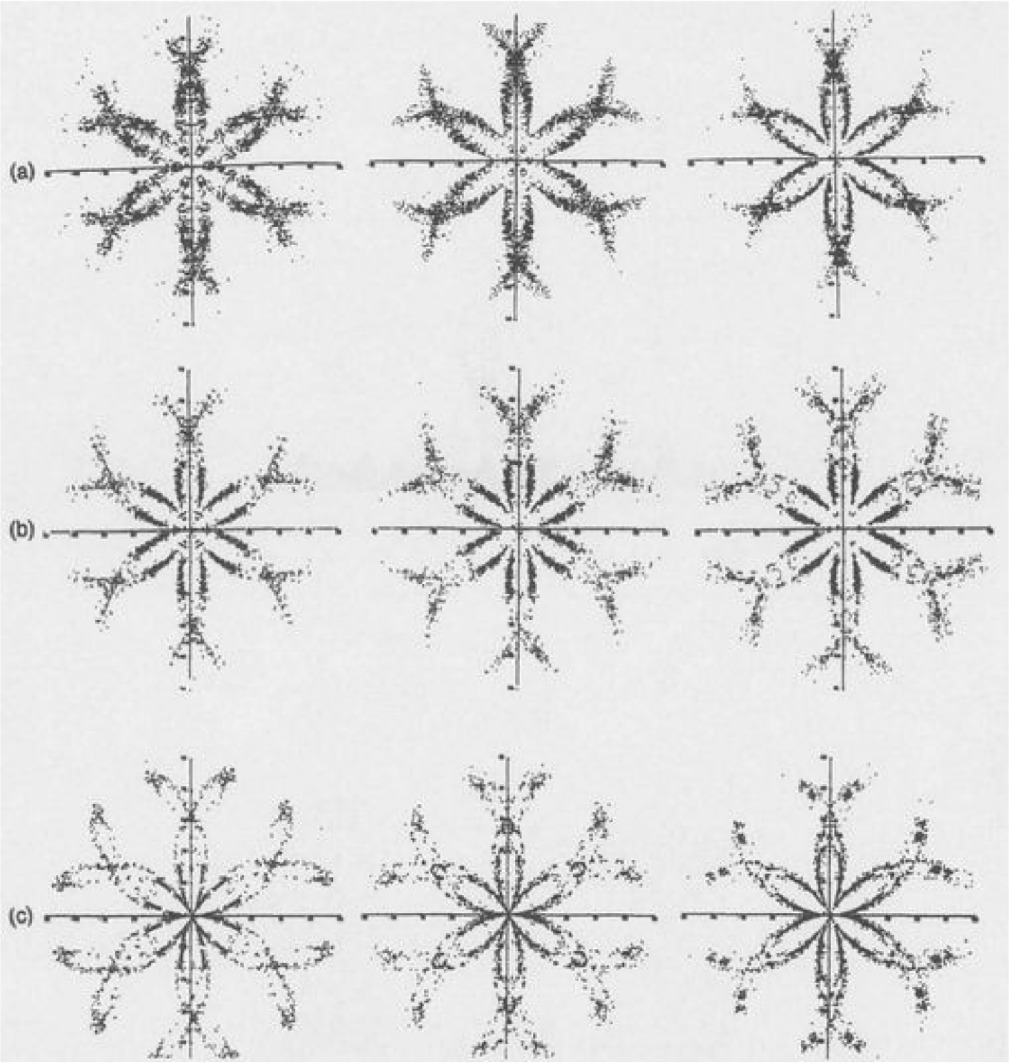


Figure 4.2. Family similarities for different speakers. These SDPs were computed from three individuals ((a), (b), and (c)) for three different occurrences of the sound "OO" as in "boot." Despite sensitivity to speaker individualities, SDPs have a global similarity for all "OO's."

The use of this data display in representing cardiac sounds is discussed in the next section.

4.4 Medicine: Cardiology and SDPs

Symmetrized dot-patterns may have applications in representing heart sounds. Nearly a million Americans die each year of cardiovascular disease, according to the American Heart Association. The traditional diagnostic methods for cardiac disease — listening with a stethoscope (auscultation), examination of graphic records of the audible sounds (phonocardiography), or electrocardiogram (ECG) analysis — have been used for years by physicians and other medical personnel to detect abnormalities of the heart. The symmetrized dot-pattern display mentioned in the previous section can be used to represent normal and pathological heart sounds (mild mitral stenosis, and mitral regurgitation). Figure 4.5 shows example SDPs for cardiac sounds.

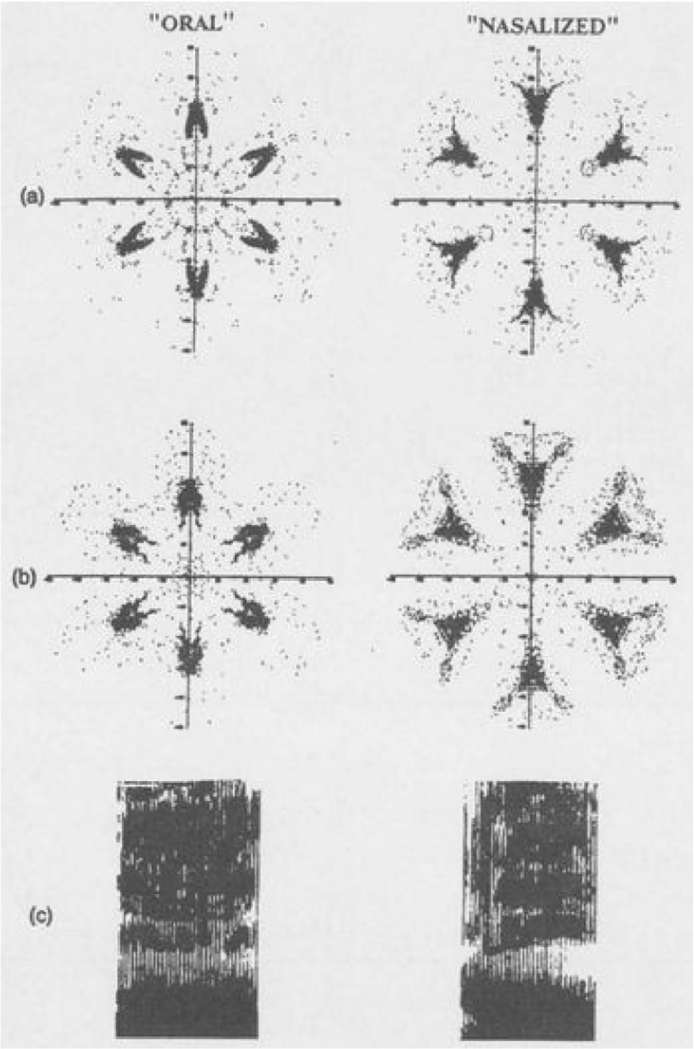
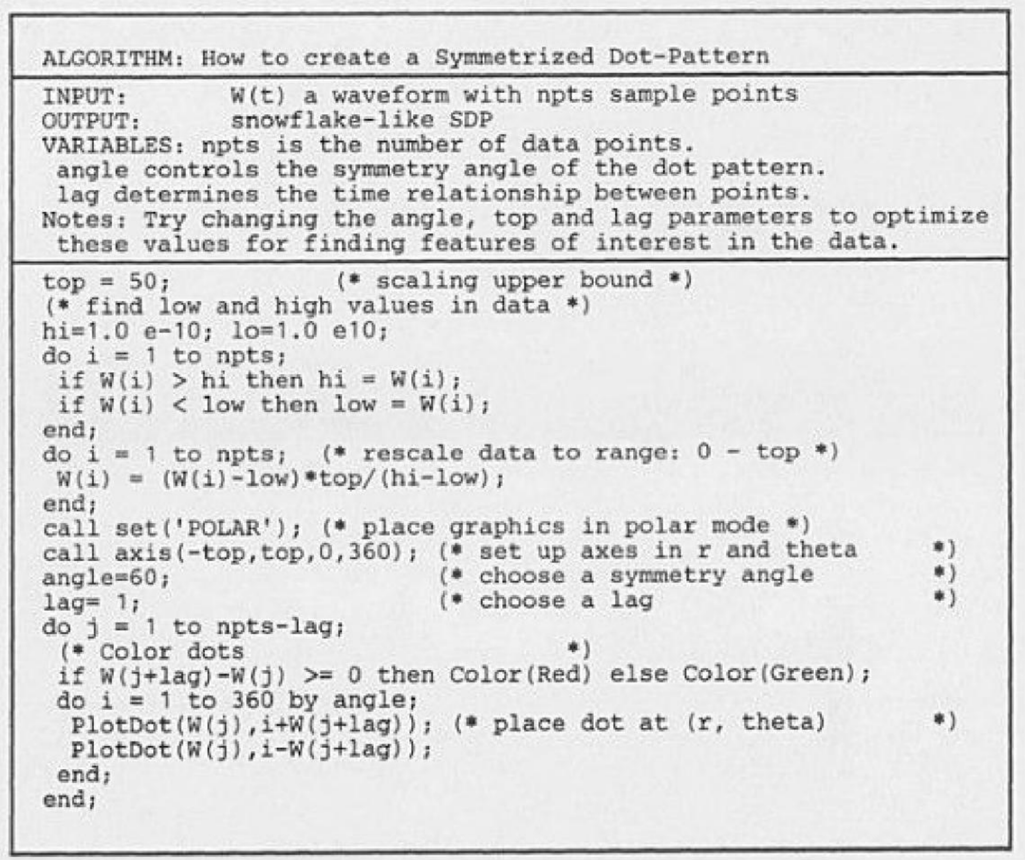


Figure 4.3. SDPs computed for oral and nasalized sounds. (a) "AH" and nasalized "AH" produced by a male speaker (as in the first vowel sound in "father" and "mom," respectively). (b) "AH" and nasalized "AH" produced by a female speaker. (c) Spectrograms for "AH" and nasalized "AH." SDPs can make the differences obvious, even for inexperienced users of SDPs. For comparisons of these patterns before and after symmetrization, see (Pick86d, J. Acoust. Soc. Am.).

Unlike the ECG which measures electrical activity of the heart, the SDP described here uses acoustic input. The symmetrized dot-pattern (SDP) characterizes waveforms using patterns of dots and requires very limited computational time as prerequisite. Previous studies in texture discrimination and pattern recognition have shown that symmetry elements can make features more obvious to the human observer, and for this reason the SDPs have a high degree of induced symmetry (and redundancy) in order to aid the human analyst in recognizing and remembering patterns. In Figure 4.5, the SDP marked “normal” was computed from a normal heart beat. Another SDP represents the sounds from a patient with mild mitral stenosis. Mitral stenosis is an abnormal narrowing of the mitral valve usually resulting from a disease such as rheumatic fever, and obstructing the free flow of blood from left atrium to left ventricle. Figure 4.5 also shows an SDP computed from a patient with mitral regurgitation, the abnormal back-flow of blood into the left atrium. Prior work in speech (last section) has suggested that the SDP functions somewhat like an autocorrelator and is also sensitive to general frequency content and waveform variability. The higher frequency noise characterizing the back-flow segment in mitral regurgitation gives rise to the characteristic “fuzzy” pattern in the SDP in Figure 4.5. An intermediate amplitude region in mitral stenosis gives rise to the dark “flying v” formations. These SDPs were computed for samples of about one-half second duration; however, other time-lengths were tried, including the capturing of several heart beats per frame, with essentially identical results. Also, when studying different segments in time, essentially the same SDP was generated.



```
ALGORITHM: How to create a Symmetrized Dot-Pattern

INPUT:       W(t) a waveform with npts sample points
OUTPUT:      snowflake-like SDP
VARIABLES: npts is the number of data points.
 angle controls the symmetry angle of the dot pattern.
 lag determines the time relationship between points.
Notes: Try changing the angle, top and lag parameters to optimize
 these values for finding features of interest in the data.

top = 50;              (* scaling upper bound *)
(* find low and high values in data *)
hi=1.0 e-10; lo=1.0 e10;
do i = 1 to npts;
 if W(i) > hi then hi = W(i);
 if W(i) < low then low = W(i);
end;
do i = 1 to npts;  (* rescale data to range: 0 - top *)
 W(i) = (W(i)-low)*top/(hi-low);
end;
call set('POLAR'); (* place graphics in polar mode *)
call axis(-top,top,0,360); (* set up axes in r and theta     *)
angle=60;                  (* choose a symmetry angle        *)
lag= 1;                    (* choose a lag                   *)
do j = 1 to npts-lag;
 (* Color dots                         *)
 if W(j+lag)-W(j) >= 0 then Color(Red) else Color(Green);
 do i = 1 to 360 by angle;
  PlotDot(W(j),i+W(j+lag)); (* place dot at (r, theta)      *)
  PlotDot(W(j),i-W(j+lag));
 end;
end;
```

Pseudocode 4.1. *How to create a symmetrized dot-pattern ("speech flake").*

The several demonstrations included in this section indicate that SDPs can make differences obvious even to inexperienced persons. Unlike SDPs, the traditional cardiac displays are not the same as pictures, since pictures have numerous visual features that can be readily identified, labeled, remembered, and integrated into a coherent whole. The ease with which patterns can be recognized may have value in instances where feedback to the patient or other medical personnel is useful, particularly when a physician is unavailable. This recognition ease and SDP-sensitivity may also be useful for researchers and physicians when comparing and studying heart sounds. Obviously, more cardiac sounds and many subjects need to be studied to fully assess the extent of SDPs' usefulness. The specific visual correlates of acoustic-cardiac features which people use to distinguish one member of a cardiac class from another would provide an interesting avenue of future research. (Pick89, *Leonardo*.)

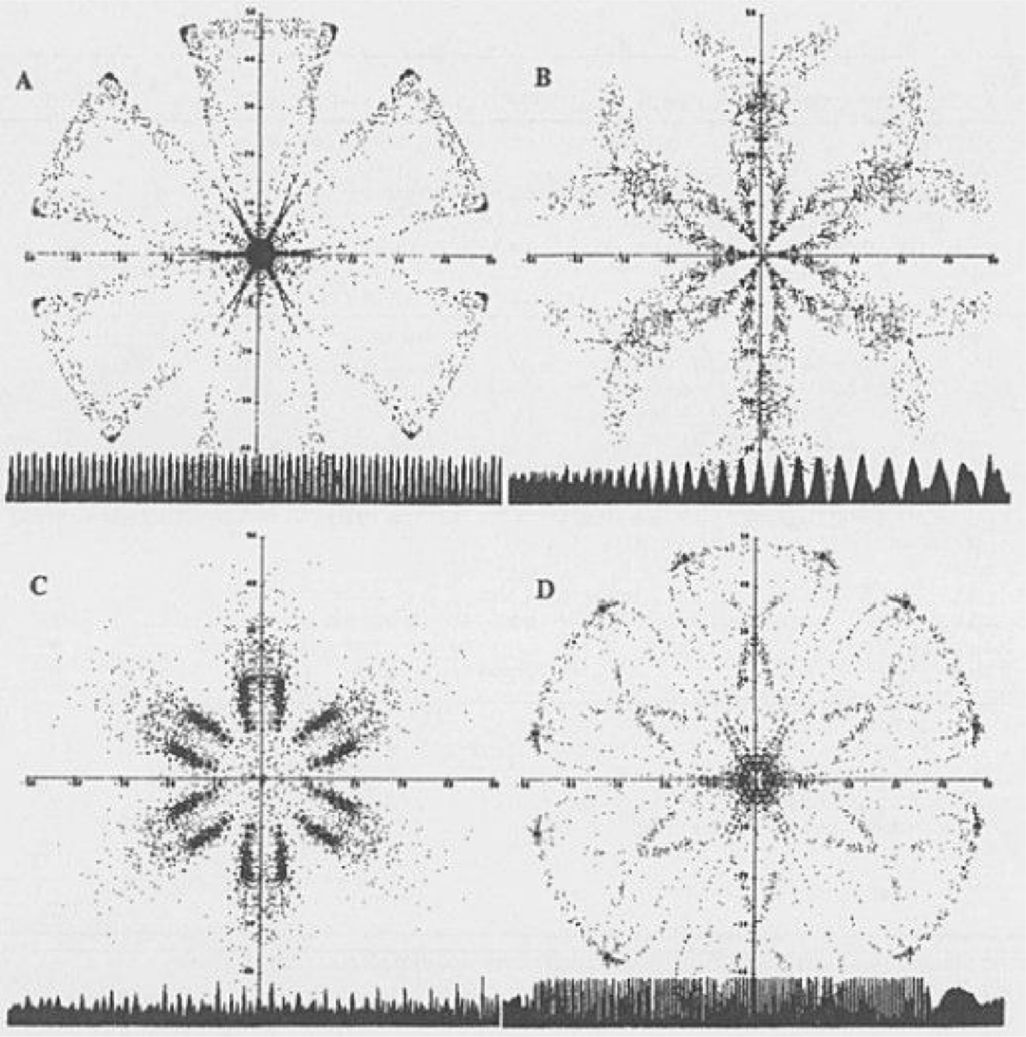


Figure 4.4*. Animal vocalizations. SDP are sensitive to frequency variations. This set shows the differences evident in the sounds of (clockwise, starting at upper left) a rooster, a dolphin, a frog and a cat.*

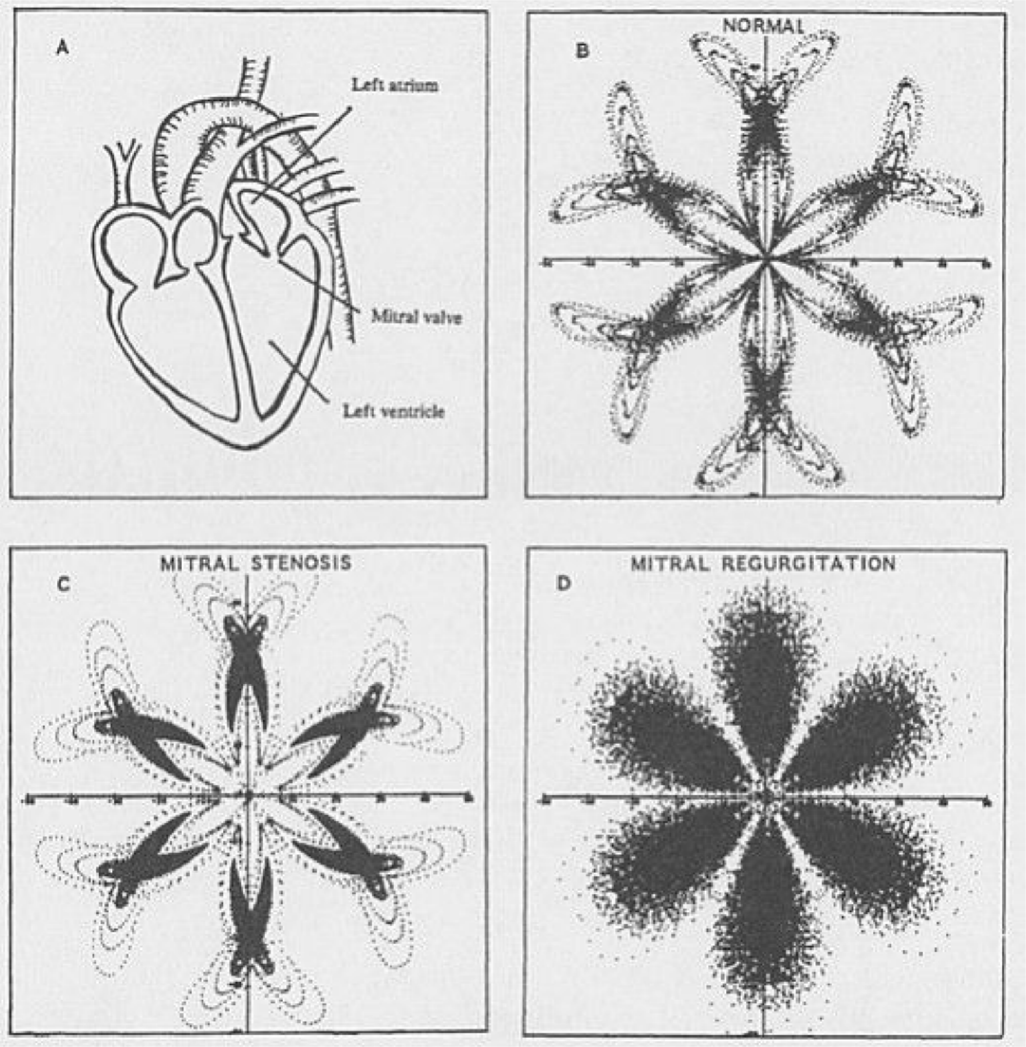


Figure 4.5. Heart sounds. A normal heart sound can be contrasted with cardiac sounds associated with various pathological conditions.

4.5 Another Dot-Display Used in Molecular Biophysics

Sections 4.3 and 4.4 showed how symmetrized dot-patters can represent data. Another example of a display used for data characterization at a more fundamental level of perception than the SDP is the random dot-display. This type of pattern was first researched in detail by Leon Glass in 1969 while studying visual perception. These patterns, also called random-dot moire patterns, are potentially useful in the global characterization of conformational changes occurring in biomolecules. As background, if a pattern of random dots is superimposed on itself and rotated by a small angle, concentric circles are perceived about the point of rotation. If the angle of rotation is increased, the perceived circles gradually disappear until a totally unstructured dot display is seen. This effect demonstrates the ability of the human visual system to detect local autocorrelations and may suggest a physiological basis of form perception in higher animals (Glass, 1969). Figure 4.6 shows an example of a dot interference pattern. The pattern is comprised of a set of ten thousand random dots which was superimposed on itself and subsequently rotated and uniformly expanded. Though the pattern was calculated by computer, similar patterns can easily be generated using sprinkled ink and transparencies.

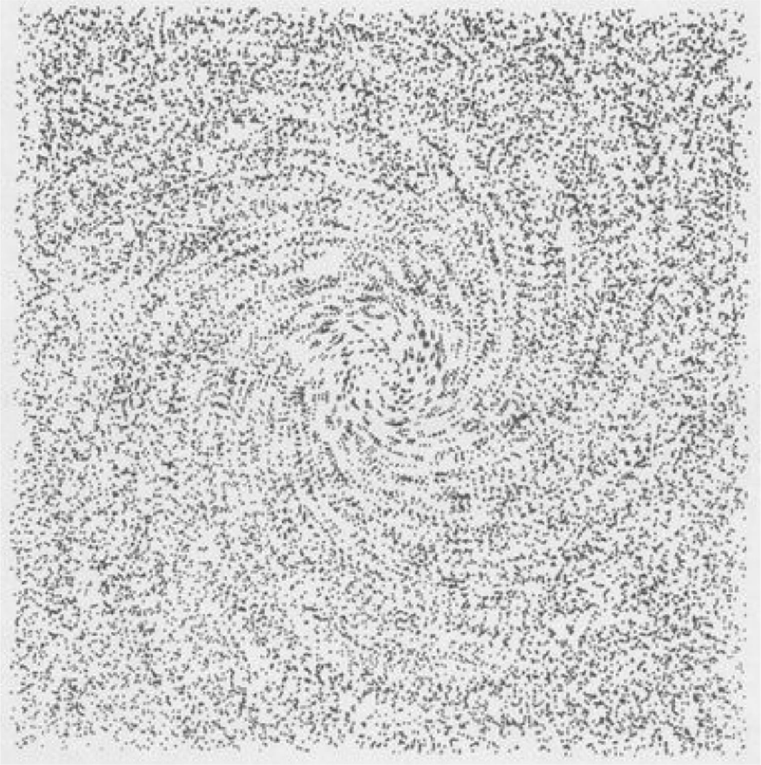
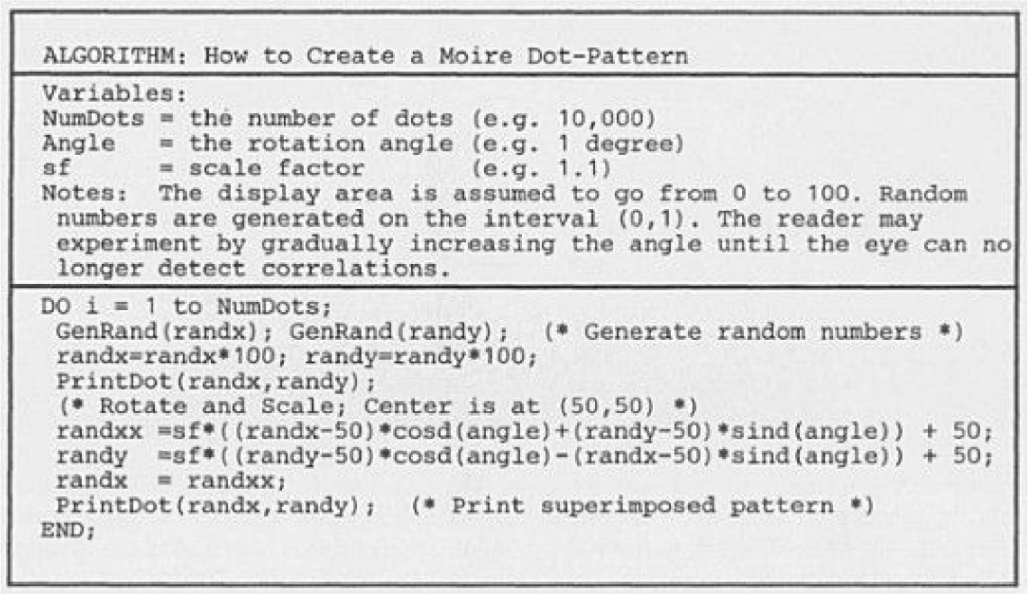


Figure 4.6. *Random dot-patterns. Random dot-display produced by superimposing a figure containing 10,000 random dots upon itself and subsequent rotation by three degrees and uniform expansion by a factor of 1.1.*

Note: if the rotation is much larger, the eye looses the ability to perceive the spiral patterns.

As an example of lateral use of computer software tools ("Lateral Use of Computer Software Tools" on page 4), the "Glass patterns" can be applied to a problem in biophysics. By placing random dots on a graphics representation of a protein molecule before and after rotation, the center of rotation can easily be found (just as we can easily perceive the center in Figure 4.6). There are sometimes crystal structures of two forms of a biomolecule related by a conformational change. Often it is desirable to ascertain an equivalent axis of rotation relating two structures in conformational space regardless of the fact that the actual transformation between the starting and ending form of the molecule may have involved many small intermediate rotational and translational components. (To learn more about how to use these techniques to visually capture motions in proteins and advantages over brute-force numerical methods, see Pick84, *J. Mol. Graph*.)



```
ALGORITHM: How to Create a Moire Dot-Pattern

Variables:
NumDots = the number of dots (e.g. 10,000)
Angle   = the rotation angle (e.g. 1 degree)
sf      = scale factor       (e.g. 1.1)
Notes:  The display area is assumed to go from 0 to 100. Random
 numbers are generated on the interval (0,1). The reader may
 experiment by gradually increasing the angle until the eye can no
 longer detect correlations.

DO i = 1 to NumDots;
 GenRand(randx); GenRand(randy);   (* Generate random numbers *)
 randx=randx*100; randy=randy*100;
 PrintDot(randx,randy);
 (* Rotate and Scale; Center is at (50,50) *)
 randxx =sf*((randx-50)*cosd(angle)+(randy-50)*sind(angle)) + 50;
 randy  =sf*((randy-50)*cosd(angle)-(randx-50)*sind(angle)) + 50;
 randx  = randxx;
 PrintDot(randx,randy);  (* Print superimposed pattern *)
END;
```

Pseudocode 4.2. *How to create a Moire dot-pattern.*

4.6 Autocorrelation Cartoon-Faces for Speech

> *"The most exotic journey would not be to see a thousand different places, but to see a single place through a thousand person's eyes."*

Presented in this section is a rather unorthodox computer graphics characterization of sound and DNA sequences using computer generated cartoon faces. As background, computer graphics has become increasingly useful in the representation and interpretation of multidimensional data with complex relationships. Pseudo-color, animation, three-dimensional figures, and a variety of shading schemes are among the techniques used to reveal relationships not easily visible from simple correlations based on two-dimensional linear theories.

Showing correlations between two or three variables is easy: simply plot a two-dimensional or three-dimensional graph. But what if one is trying to present four or five or even ten different variables at once? The face method of representing multivariate data was first presented in 1973 by Chernoff, a Harvard statistician. Using gradations of various facial features, such as the degree of eyebrow slant or pupil size, a single face can convey the value of many different variables at the same time (Figure 4.7). Such faces have been shown to be more reliable and more memorable than other tested icons (or symbols), and allow the human analyst to grasp many of the essential regularities and irregularities in the data. In general, *n* data parameters are mapped into a figure with *n* features, each feature varying in size or shape according to the point's coordinate in that dimension. The data sample variables are mapped to facial characteristics; thus, each multivariate observation is visualized as a computer-drawn face. This aspect of the graphical point displays capitalizes on the feature integration abilities of the human visual system and is particularly useful for higher levels of cognitive processing. Figure 4.8 shows the range of faces produced when random numbers ("white noise") are mapped to facial coordinates.

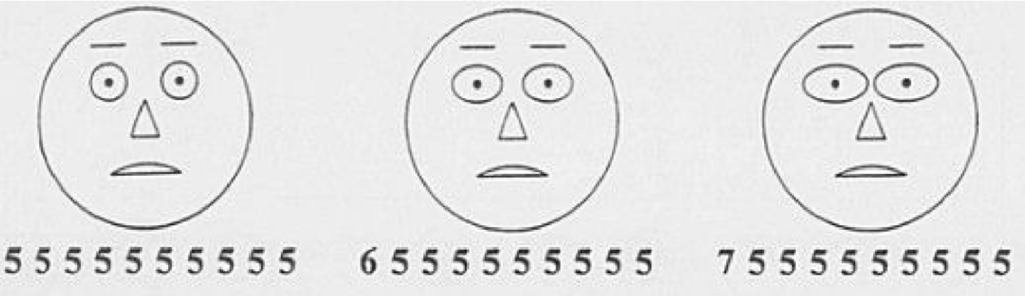


Figure 4.7. *Cartoon faces and data analysis. These cartoon faces can be used to represent the values of as many as 10 variables, each variable corresponding to a facial feature. Here only one facial feature, horizontal eye length, is changed. Other facial coordinates are set to a constant middle-position.*

For speech applications, an autocorrelation analysis coupled to computer generated cartoon faces can be used to represent speech sounds. The autocorrelation of a signal $x(n)$ with lag k is defined as

$$\phi(k) = \sum_{n=-\infty}^{\infty} x(n)\, x(n+k). \tag{4.1}$$

The autocorrelation function for data describes the general dependance of the values of the data at one time on its values at another time. On a computer an autocorrelation function for a finite window in time can be implemented as shown in Pseudocode 4.3. For more on autocorrelation theory, see Bendat and Piersol (1968) and Witten (1982). In 1985, I devised the “autocorrelation-face” where 10 facial parameters are computed from the first 10 points of the autocorrelation function of a 50 ms sample of the speech sound. This process is described in more detail in the *J. Educ. Tech. Syst.*

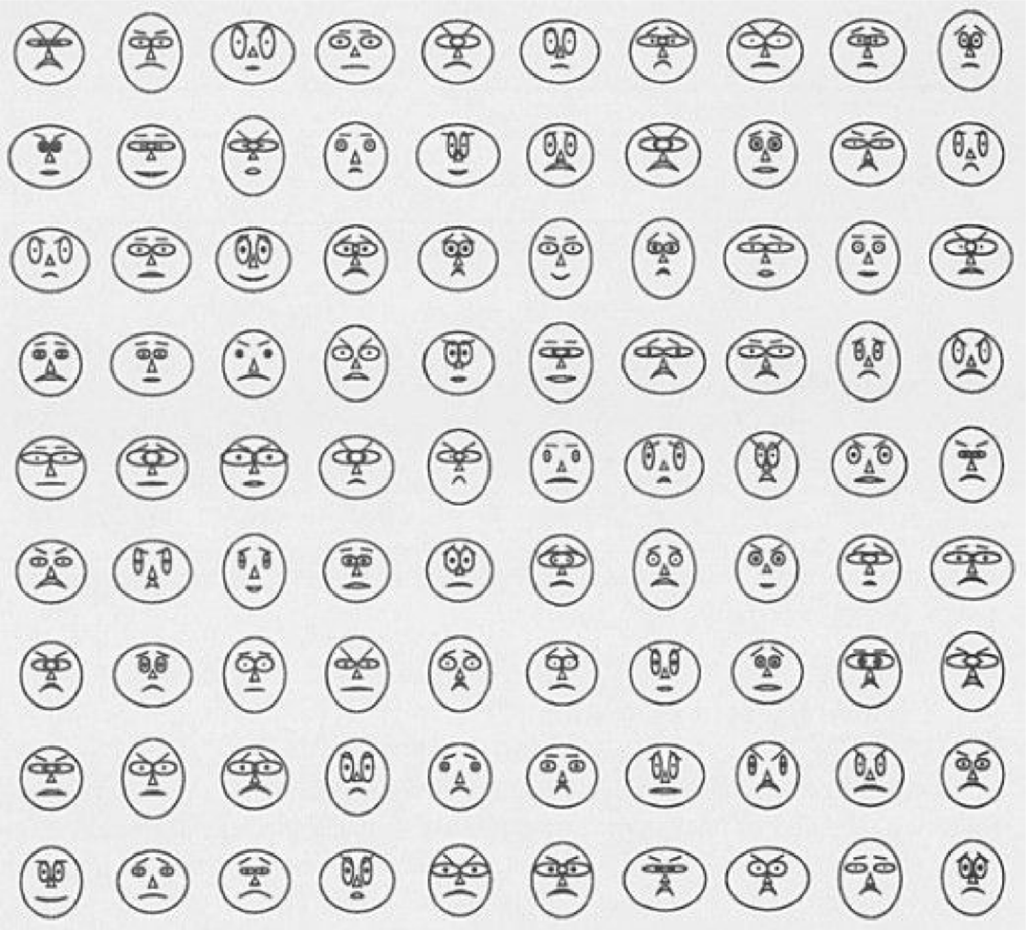
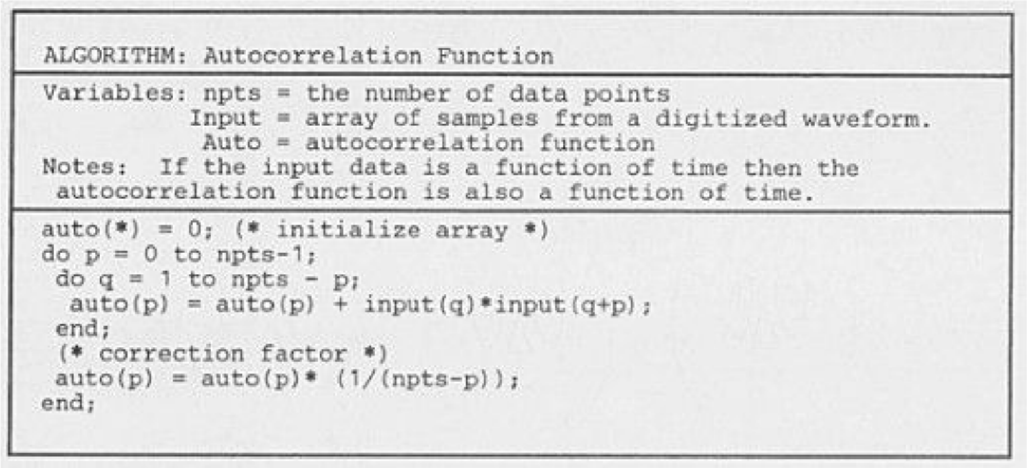


Figure 4.8. The diversity of computer-generated faces. The settings for each of the ten facial parameters were computed using a random number generator.

In this and the following applications, ten facial parameters, *F*(1, 2, 3, 4, 5, 6, 7, 8, 9, 10) are used, and each facial characteristic has ten settings, *S*(1, 2, 3, 4, 5, 6, 7, 8, 9, 10), providing for 10 billion possible different faces. The controlled features are: head eccentricity, eye eccentricity, pupil size, eyebrow slant, nose size, mouth shape, eye spacing, eye size, mouth length, and degree of mouth opening. Head eccentricity, for example, controls how elongated the head is in either the horizontal or vertical direction. The mouth is constructed using parabolic interpolation routines, and the other features are derived from circles, lines, and ellipses. Pseudocode B.3 (in Appendix B) gives details on the computer generation of faces. Figure 4.9 shows some examples computed from human speech.[12] The resultant speech-faces could provide useful biofeedback targets for helping deaf and severely hearing-impaired individuals to modify their vocalizations in selective ways — especially since they may provide simple and memorable features to which children could relate. The traditional speech spectrogram displays are not the same as pictures, since pictures have numerous visual features that can be readily identified, labeled, and integrated into a coherent whole. To compare SDPs (previous section) and faces: note that unlike faces, SDPs do not elicit an emotional reaction. Emotion does confer a mnemonic advantage for the faces, but can sometimes obscure the association, e.g. a smiling face representing cancer statistics. (Pick85a, *J. Educ. Tech. Syst.*)



```
ALGORITHM: Autocorrelation Function

Variables: npts = the number of data points
           Input = array of samples from a digitized waveform.
            Auto = autocorrelation function
Notes:  If the input data is a function of time then the
 autocorrelation function is also a function of time.

auto(*) = 0; (* initialize array *)
do p = 0 to npts-1;
 do q = 1 to npts - p;
  auto(p) = auto(p) + input(q)*input(q+p);
 end;
 (* correction factor *)
 auto(p) = auto(p)* (1/(npts-p));
end;
```

Pseudocode 4.3. *Autocorrelation function.*

4.7 Cartoon Faces in Education

A number of recreational and educational uses for the faces are suggested in the following sections. As background, research has demonstrated the potential value that visualization and iconic systems play in learning and instruction. Popular educational software for home computers is becoming available (e.g., the “FaceMaker” by Spinnaker (see references)) which allows children to create faces from sets of eyes, ears, and noses. Programs such as these help children become comfortable with computer fundamentals such as menus and cursors. The computer-drawn faces presented in the current section have particular value in that they are created under parametric control and can provide immediate visual feedback to the user. In addition, any face can easily be regenerated at a later time from its control-data.

4.7.1 Cognitive Association of Coordinates with Facial Features

There have been several studies in the literature which have explored the child’s ability to organize and represent body location information. Here, a simple face-drawing system was developed where children can type numbers at the terminal keyboard and immediately view the results on an adjacent graphics screen. For example, faces were constructed from the control-data entered by Lisa, a 6-year-old girl with no prior experience with computers. One face in particular was her favorite, because she found the shape of the mouth amusing. She worked on the figure for several minutes, developing the mouth to her specifications, and subsequently she recorded the final control-data on a piece of paper. This indicated that she understood the concept of number-to-face parameter mapping.

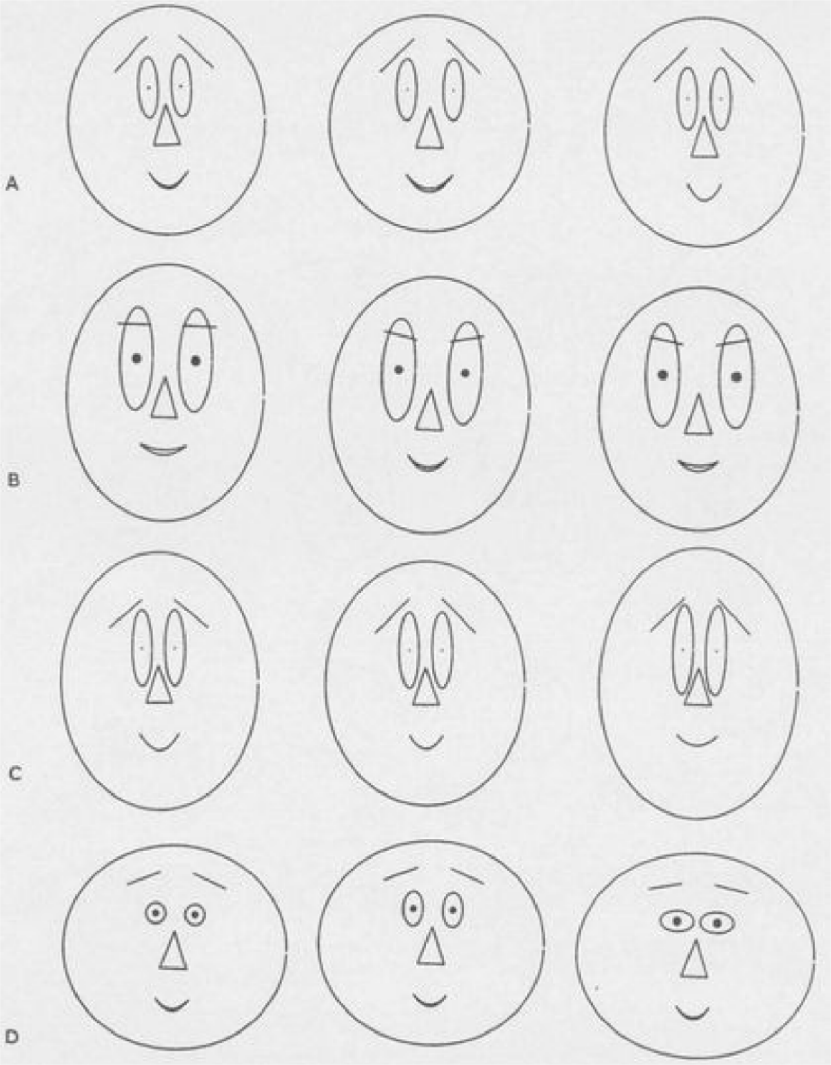


Figure 4.9. *Faces from sound. Cartoon faces can be used to characterize speech sounds. The top row represents the fricative sound "s"; the second, "sh"; the third, "z"; and the fourth, "v." Sounds were repeated three times. For vowel and nasal sounds, see J. Educ. Tech. Syst. See* ***Appendix B*** *for pseudocode for face generation.*

4.7.2 Target-Pictures for Children

The faces may also serve as target-pictures for children to draw. [For background information on the differential cue utilization by children in pattern copying, the development of drawing rules by children, and “tadpole drawing” (body representations where the legs and arms are attached to the head), see Taylor and Bacharach (1981).] Since the computer faces are created from control-data, the resultant faces can easily be regenerated at a later time, or altered slightly, in order to test hand-eye coordination and development. Figure 4.10 includes four computer-drawn faces and children’s attempts to reproduce them. The drawing task can be made much more difficult if the child is asked to view the face first and then required to draw it as well as possible from memory. Computer software, and hardware such as digitization tablets, make an analytic comparison between computer- and child-generated faces easy. Simple parameters such as center of gravity, and radius of gyration (“Breathing Proteins” on page 30) can be computed to characterize the drawings in an objective way.

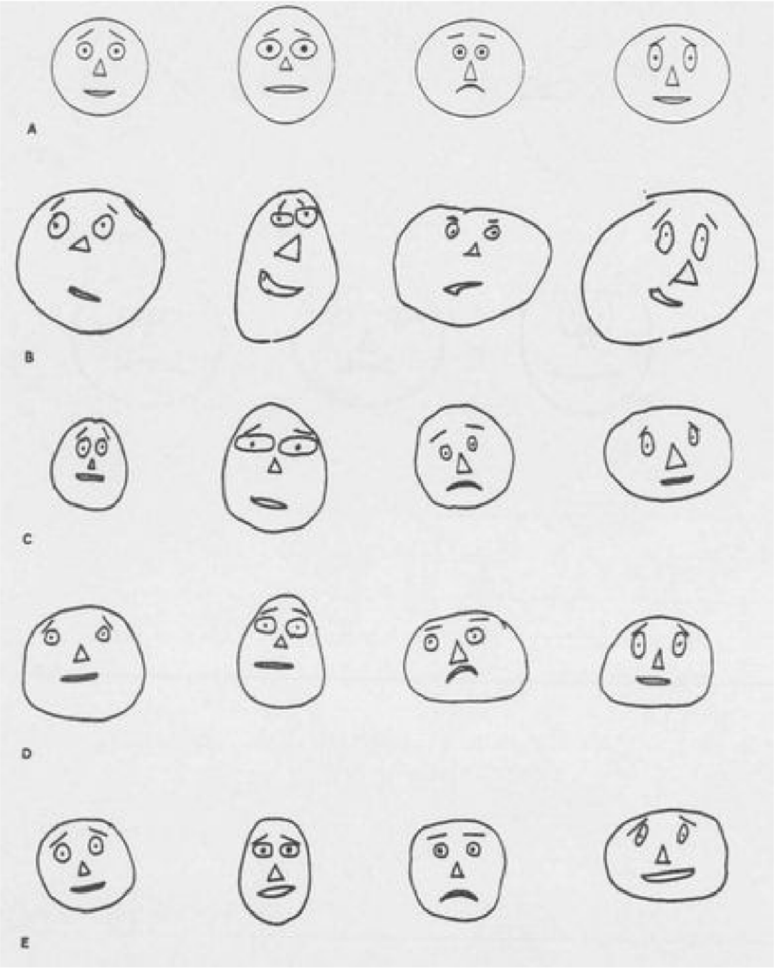


Figure 4.10. Target faces. Drawings made by children in an attempt to reproduce the four computer-drawn targets at top. From top to bottom, the ages of the children were 6 , 6, 8, and 10.

For years psychologists have tried to determine when infants first realize how the features of the human face are naturally arranged and when an infant's ability to perceive facial expressions begins. Computer-generated faces might be ideal for the study of infant's perception of natural and distorted arrangements of a schematic face. In the study of Maurer and Barrera (1981), it was shown that 2-month-old infants show a preference for a natural arrangement of facial features on a cartoon face, as opposed to scrambled features. Though their cartoon faces were not computer-generated, computers could be used in the placement (random or otherwise) of the facial features on the head, giving the researcher rigorous control of the resultant expressions.

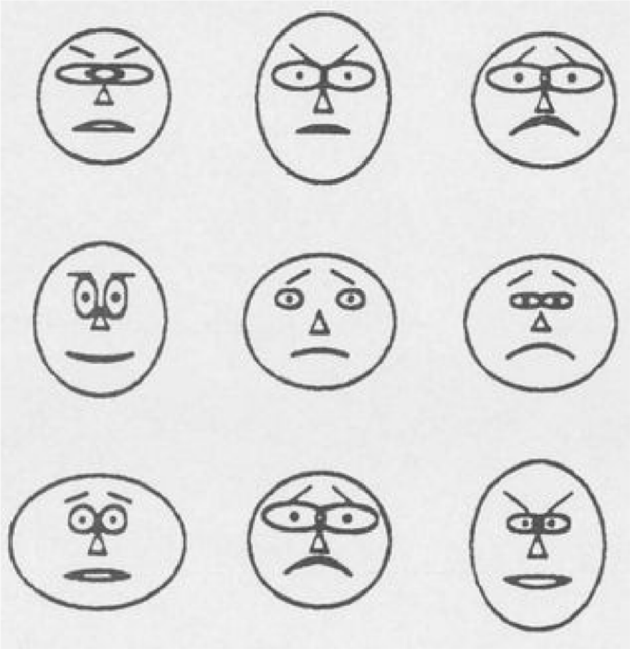


Figure 4.11**.** *Which two are the same? The faces can be used to illustrate the concept of similarity, sameness, and difference. Since the facial parameters are accurately controlled, the degree of difficulty of the task can be specified.*

4.7.3 Learning by Means of Analogy

The faces can be used to illustrate the concept of similarity, sameness, and difference. Since the facial parameters are accurately controlled, the degree of difficulty of the task can be specified (Figure 4.11). Possible tasks include: Which two are the same? and Which one is different? The faces can be used to explore memory abilities: initially, one face is shown, then erased, and the user can subsequently be asked to choose the face from a small group, somewhat like picking from a police line-up.

4.8 Educational Aid for the Presentation of Statistical Concepts

The faces may be suitable as visual supplements in the presentation of statistical concepts, particularly distribution theory, to individuals inexperienced in mathematics and with no prior knowledge of the methods of statistical evaluation. In this work, faces were used to illustrate the concept of white noise (totally random distribution) such as that shown in Figure 4.8, in contrast to Gaussian noise (normal error distribution) (Figure 4.12), theories usually not introduced to indi-viduals prior to the high-school level due to the mathematical complexity of the subject matter. For the case of white noise, one hundred faces were generated, each facial characteristic having a setting determined from a random number generator. In the case of the Gaussian noise, the facial settings S_i are given by:

$$S_i = \sum_{j=1}^{N} \frac{\delta_j}{N}$$

(4.2)

where $_j$ are random numbers, and *N* is 5 for weakly Gaussian noise or 20 for strongly Gaussian noise. *i* signifies the facial parameter used (from 1 to 10). Gaussian random noise was mapped to facial parameters (notice the faces have a more “middle of the road” look). For very young students, the faces could be used, in addition to standard techniques, for visualizing simpler concepts such as the mean, median, mode, and other measures of central tendency. For additional information, see Glossary entry *Gaussian white noise.* (Pick84e, *Computers and Graph*.)

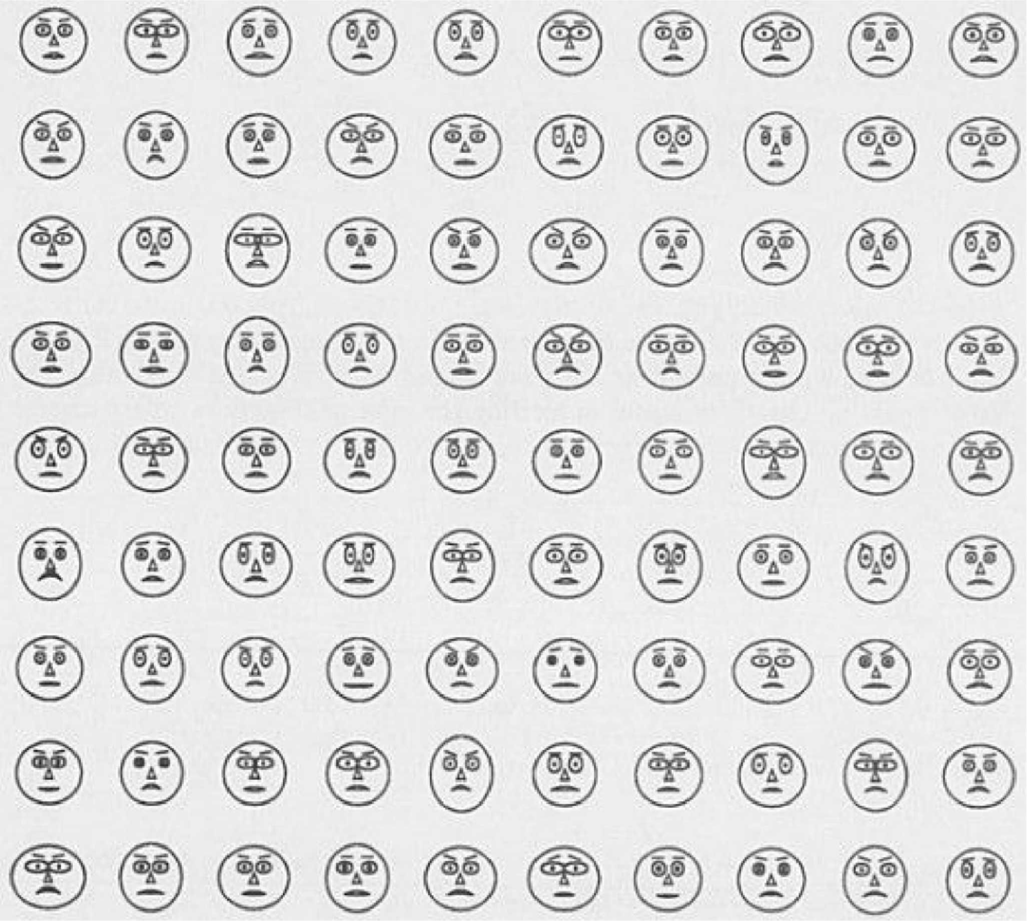


Figure 4.12. Faces produced by Gaussian noise.

4.9 Commercial and Military Air Traffic Control

One may speculate about potentially useful applications of the computer-drawn faces in the cockpit of airplanes. The growing complexity of aircraft controls and readouts are making aircraft almost too complex to fly. The faces can accommodate analog or digital input from a multitude of readouts, each facial parameter receiving input from one or more gauges. Deviations in the controls from their expected values would give rise to excursions of the facial parameters from their middle settings. While it is true that the more standard concept of having a gauge blink or beep when a parameter has gone beyond a critical value is valuable, the faces would be especially useful in alerting the pilot of conditions where several readings are not themselves at critical stages, but where the combined effect may be dangerous.

4.10 Faces and Cancer Genes

Face icons can be used to detect irregularities in DNA sequences. The deviation of the DNA statistical properties from their expected (random) value causes deviations of the facial parameters from their middle positions. The number of possible DNA characteristics that can be visualized by this method is large. (For further reading see: Pick89, *Speculations in Science and Tech.;* Pick84b, *J. Mol. Graph*.)

4.11 Back to Acoustics: Phase Vectorgrams

In this section, we return to a discussion of graphics for representing sounds. Here the "phase vectorgram" representation is presented, developed with colleagues in 1985.

Until recently it had been believed that the perceived sound of an audio signal could be completely characterized by its power spectrum (energy vs. frequency). As suggested in "Fourier Analysis: A Digression and Review" on page 21, a sine wave is characterized by amplitude (or extreme height), its period (the time between one peak and the next) and its phase (the position of the sine wave in time). Recent psychoacoustic experiments have revealed that phase relations between the sinusoidal components in signals can be perceived — yet they are not represented on spectrograms. This suggests that there is a more appropriate domain than that of the power spectrum in which to process signals. The phase information normally discarded via transformation to the domain of the power spectrum must be reconsidered. Unfortunately, the short-term phase of a dynamic signal is difficult to quantify and plot. [These difficulties are overcome in an analysis using an autocorrelation-based pitch detector, followed by discrete Fourier transform, and normalization of the plot to the phase of the fundamental frequency ($_1$) :

$$\phi_i = \phi_i - k\phi_1$$

(4.3)

where *k* is the harmonic number and $_1$ is the phase of the fundamental. (These terms may be unfamiliar to some readers; see Otnes and Enochson (1978) for more information.) With knowledge of the fundamental frequency we can adjust the Fourier analysis so the reference point is synchronized to the fundamental period of the waveform, and the frequency sample points can be taken at more meaningful points corresponding to the harmonics of the fundamental.]

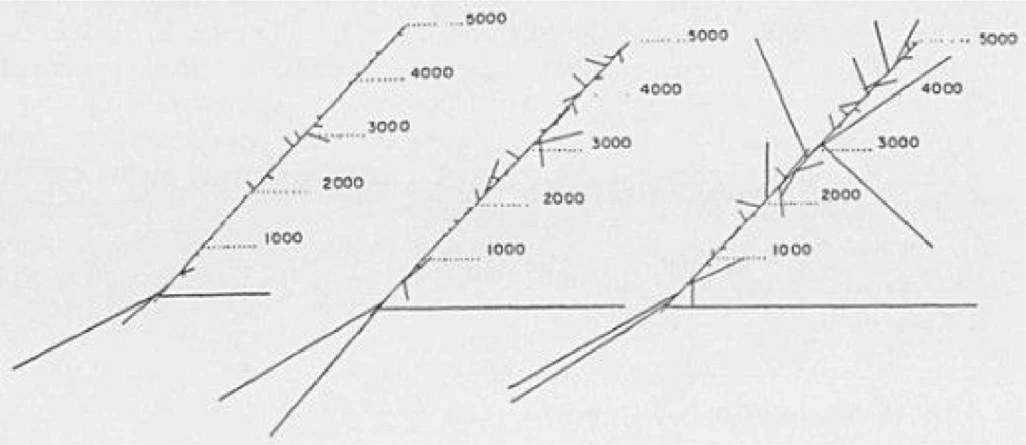


Figure 4.13*. *Phase vectorgram. Harmonics of a vowel sound are plotted as "bristles" emanating from a central axis, with an angle equal to the phase of each harmonic.

In the phase vectorgram (Figure 4.13) amplitude, frequency and phase are presented in a cylindrical plot resembling a pipe-cleaner or bottle-brush. Shown is a vectorgram for different regions in time for the human utterance "ee," as in "meet." Given the amplitudes for components k and the corrected phases (see Equation (4.3)), the "bristles" in the cylindrical plot may be drawn. The length of each bristle is determined by the gain at each frequency. Frequency extends from low frequency (0 Hz) to high frequency (5000 Hz) as the cylinder goes from front to back. Phase is represented by the angle the bristle makes with the central axis. Cylindrical plots are normalized so that all bristles may be accommodated in the same size graph. The cylindrical plot represents the phases more meaningfully than traditional two dimensional phase-vs.-frequency plots which typically have phase from -180 to +180 degrees on the ordinate and frequency on the abscissa — amplitude is not a factor in such plots. In our plots, the length of the bristle reflects the importance of that particular phase component since the bristle length is related to the amplitude. Low amplitude bristles due to noise or computational artifacts therefore do not obscure the plot. In addition, the cylindrical representation eliminates the need for "phase unwrapping" (Otnes and Enochson (1978)).

Generally, the cylindrical plot gives a clear indication of both phase and amplitude as a function of frequency. It may be of use in a variety of signal processing applications. Evidence of speaker-independent phase "signatures" for phonemes (the basic building blocks of speech) suggests the use of phase vectorgrams in speech recognition. (The phase vectorgram represents a collaboration between the author and M. Martin and M. Kubovy.)

4.12 Fractal Characterization of Speech Waveform Graphs

Presented in this section is an alternate way of characterizing speech. The methods use the concepts of fractal geometry set forth in B. Mandelbrot's book *The Fractal Geometry of Nature* (Mandelbrot (1983)). The beauty and complexity of fractal shapes in pure mathematics is discussed in Part III of this book.

4.12.1 Scale Invariance

Many objects and patterns in the natural world possess the quality of *"self-similarity"* — for a range of scales used to view the pattern, the magnified portion of the shape looks (qualitatively) like the original pattern. Such objects include mountains and coastlines, as well as several classes of patterns derived purely from mathematics. This scale invariance has been studied extensively by B. Mandelbrot, who coined the word "fractal" (in 1975) to describe such irregular shapes.

4.12.2 Fractal Characterization of Speech

A fractal object has a shape with increasingly detailed features revealed with increasing magnification, and examples include mountains and coastlines (this is explained in greater detail in the section on self-similarity). In contrast, the edge of a circle is not fractal since it is featureless upon increasing magnification (i.e., it becomes a straight line).

The fact that such complicated, and seemingly random, shapes of nature can be characterized by a single number, the "fractal dimension" D, motivates the test of fractal characterization in speech science. The speech waveform is a very irregularly shaped signal which can be treated as a coastline and studied using fractal mathematics. If speech waveforms can also be globally quantified and compared using a single number, a new way of understanding (and a new focus on) many problems in acoustics and phonetics might be provided.

In this work, the speech waveform in the time-range of ~ 2 seconds to ~ 10 milliseconds (ms) was studied, since this time scale represents the area in which important prosodic and phonetic events occur. Prosody includes a description of the "music" of speech (pitch, amplitude, timing and other "suprasegmental" features). Phonetics includes the study of the basic building blocks of speech such as phonemes, diphones, and coarticulation. The speech waveform is studied to determine whether its structure can be considered self-similar and whether a dimension (*D*) can be calculated. Synthetic and natural speech are also compared using this approach, and the effect of voice quality and nasality on *D* is determined. Graphic representations of speech and other natural phenomena are compared. The analyses and graphics are presented only for acoustic waveforms; however, these techniques can easily be applied to any data where one variable fluctuates through time or space. Past work by R. Voss in the global characterization of certain noise signals, and speech and music in a longer range of time (~ 10 seconds to several minutes), can be found in Clarke and Voss (1975). First, a classification of shapes and definition of terms used in this section are presented.

4.12.3 Classification of Shapes

4.12.3.1 Richardson's Coastlines and the Fractal Dimension

If one were to attempt to measure a coastline or the boundary of two nations, the value of the measurement would depend on the length of the measuring stick used. As the measuring stick decreased in length, the measurement would become sensitive to smaller and smaller bumps, and in fact the coastline's length would become infinite as the stick's length approached zero. Lewis Richardson was one of the first to quantify this phenomenon in an attempt to correlate the occurrence of wars with the shape of the boundary separating two or more nations (Richardson, (1960)). Mandelbrot built extensively upon Richardson's work and suggested that the relationship between the measuring-stick length (   ) and the apparent total length (*L*) of a coastline could be expressed by the parameter *D*, the fractal dimension. [An equivalent way to understand and calculate *D* is to study the relationship between the *number* of measuring sticks (*N*) and the length of the measuring stick (   ). For a smooth curve such as a circle, the relationship is:

$$N(\varepsilon) = \frac{c}{\varepsilon}$$

(4.4)

c is a constant. The number of sticks (*N*) needed to measure the circle's circumference increases as the length of the stick decreases. However for a fractal curve this relationship is altered slightly:

$$N(\varepsilon) = \frac{c}{\varepsilon^{D}}$$

(4.5)

If we multiply both sides of Equation (4.5) by the relation can be expressed in terms of the length of the measuring stick:

$$L(\varepsilon) = \frac{\varepsilon}{\varepsilon^{D}}$$

(4.6)

D corresponds somewhat to the traditional notion of dimension (a line is one-dimensional, a plane two-dimensional) except that *D* can be a fraction. Since a coastline is so convoluted (and has bumps upon bumps as it is magnified), it tends to fill space, and its dimension lies somewhere between a line and a plane. The fractal structure implies that repeated magnification of its graph reveals ever-finer levels of detail.] Fractals have many intriguing properties, such as being continuous everywhere but differentiable nowhere. Mandelbrot gives *D* = 1.26 for the coastline of Britain. For smooth lines, such as a circle, *D* = 1, and *L* quickly converges to the “true” circumference of a circle as decreases.

4.12.3.2 Self-similarity

If the features of an object (i.e., the general nature of its irregular bumpy structure) remains constant through successive magnifications, such as is the case for coastlines and mountains, the object is considered self-similar. This is the same as saying that the *D* calculated from the relationship between and *N* in Equation (4.5) remains constant for different . Researchers have *no problem with the idea that an object can be self-similar only in a certain range of length scales.* For practical purposes, *D* need only be constant for a suitably wide range of , a factor of 10 or more for example. Fractal objects need not be self-similar at all scales. Fractals need only show a certain amount of bumpiness with increasing magnification. However, researchers are most interested in self-similar fractals, and today the terms are often interchanged in the literature.

4.12.3.3 Types of Self-similarity: Definition of Terms

Figure 4.14 includes much of the self-similarity sub-classifications, as well as example patterns for some types. Self-similarity implies *scaling similarity* (i.e. the shapes are invariant under magnification). The edges of circles and lines are self-similar since they look the same at different magnification; however, they are smooth (and hence not fractal). They possess what is known as *standard scaling symmetry*. Objects with "bumpiness" but with scaling symmetry possess *nonstandard scaling symmetry*. Certain mathematical constructs such as the Koch curve, which can be made by superimposing smaller and smaller triangles, have *exact scale invariance*. However, most objects are only *statistically scale invariant*, since they are only invariant in an average sense (magnifications of coastlines are qualitatively identical, not quantitatively). Recently, a host of statistically scale-invariant fractals derived from the iteration of complex functions has been described (*Julia sets*). Algorithms for the generation of these beautiful and complicated structures, as well as color computer graphics, are being studied, and their popularity is evidenced by the proliferating number of articles in the scientific and popular literature (see Part III of this book). As will be shown later, speech waveforms at sentence time scales can be classified as a statistically scale-invariant natural phenomenon.

4.12.4 Speech Fractal Dimension

4.12.4.1 Coastlines and Speech Compared

What do coastlines and speech time waveforms have in common, and how can they be visually and analytically compared? Figure 4.15 shows the coastline of England and a "speech-island" side-by-side to facilitate comparison. The speech island was computed by mapping the amplitude of the speech wave into radius and time into angle. For each sampled time point a line is drawn from zero radius to that point, thereby tending to fill the figure; this places a visual emphasis on the "texture" of the edge and facilitates visual comparison with other "closed-curve" natural objects (e.g., islands, clouds, leaves). This island comprises a sentence containing 2 seconds of speech (20,000 points at the digitization rate of 10,000 samples/second). Notice that both England and the speech graph have a highly convoluted surface, but that the speech looks much rougher. The fractal dimension D, in a sense, quantifies roughness, and is discussed in the following section.

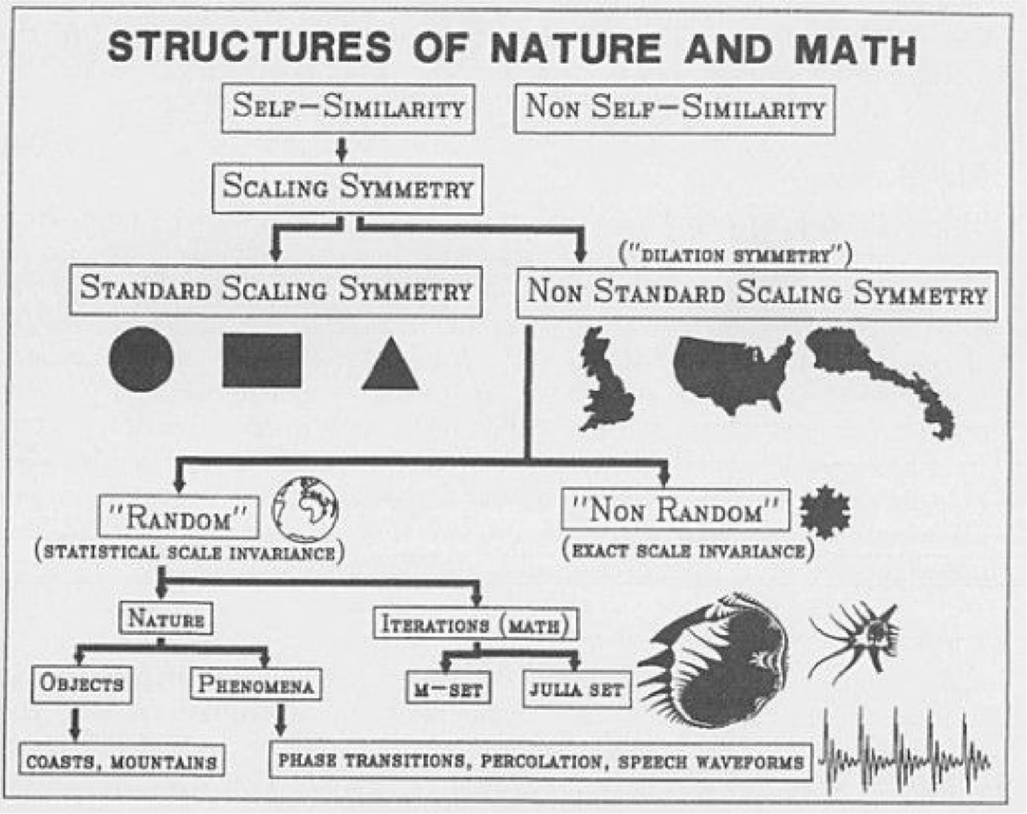


Figure 4.14. *Classification of the shapes of nature and math. Self-similar structures, when magnified, look like the original shape. "Random" self-similar structures are self-similar only in the statistical sense so even though they don't repeat their pattern exactly, they clearly have the same look at different magnifications. Speech waveforms, at sentence time scales, are statistically self-similar.*

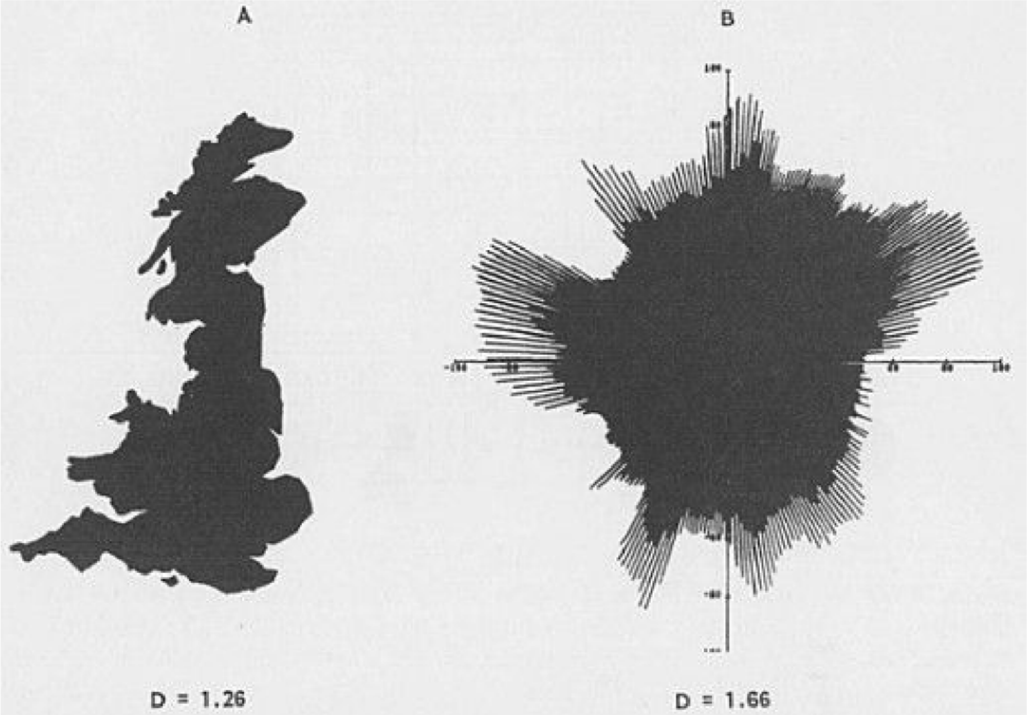


Figure 4.15. Comparison between coastlines and speech waveforms. Notice that the coast of England (A) and speech (B) have a highly convoluted surface, but speech looks somehow rougher. Mandelbrot's fractal dimension D quantifies the degree to which such irregular curves "fill space," and D = 1.26 for England's coast. For speech waveform graphs, D ~ 1.66 (here the amplitude trace is plotted in polar space with the middle filled in to focus visual attention on the "texture" of its edge and to facilitate comparison with other natural forms).

4.12.4.2 Speech-Wave Structure

If the waveforms were statistically self-similar, they could be characterized with just one number, *D*. Though the irregularity continues with each magnification, the degree of roughness seems to fluctuate slightly when looking at isolated magnifications. This does not invalidate the self-similarity judgement, since self-similarity applies only in the statistical sense and cannot be captured in just a few sample magnifications. In order to solve this problem, an approach like the one suggested by Equation (4.5) is used. The waveform graph is laid upon a grid (by computer), and the number of grid points (*N*) intersected is measured as a function of grid size, . Plots of log *N* vs. log are calculated; the slope determined by a least squares line fit to the data gives an estimate of — *D* (Figure 4.16). This relationship gives the same information as the measuring stick relationships, but is easier for the computer to calculate. The log plot for different values of were computed for the utterance, “Nine men were hired to dig the ruins.” “Self-similarity” was indicated by the remarkable straightness of the plotted line, and the slope indicated a fractional dimension *D* of 1.6 in the range studied (~ 2 s to ~ 10 ms). Note that technically, because the speech waveform’s axes have different physical meanings, the term “self-affinity” might be used; in this section the term “self-similarity” is used in a general way to suggest the *scale invariance of a time-waveform when initially scaled to fit (graphically) in a square box.* The *D*-computation is also based on the square-box scaling.

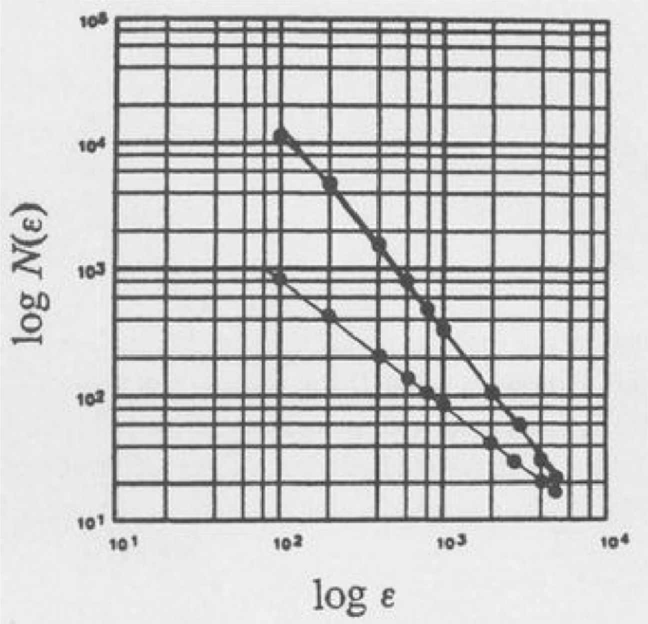


Figure 4.16. *Example speech graph. A plot of log N(e) vs. log is shown for a male human speaker, superimposed with a plot for a digitized diagonal line (D = 1). The slope of these plots gives an estimate for the fractal dimension, D. The sentence plotted is: “Nine men were hired to dig the ruins.” The axis is in units of the sampled data, so at 10,000 samples / second, “100” corresponds to 10 ms.*

4.12.4.3 Numerical Evaluation

For Richardson’s coastline measurements, published posthumously in 1960, a computer was not used, and the labor-intensive nature of his work is evident:

> *“At first I tried to measure the frontiers by rolling a wheel of 1.8 centimeters diameter on maps; but there is often fine detail which the wheel cannot follow...considerable skill would be needed to guide the wheel...Much more definite measurements have been made by walking a pair of dividers along a map of the frontier...”*

Note that as the measure gets smaller, measurement accuracies become critical. For example, in tracing a map boundary with a pair of dividers, the difficulty associated with accurately hitting the border line can greatly affect the measurement. A computer-based numerical approach helps to overcome this problem. Using the methods described above, *D* was computed for a variety of acoustic signals. For human speech, by averaging different sentence utterances for both male and female speakers of ~ 2 seconds in duration, $D = 1.66 \pm .05$ This remarkable invariance suggests that *D* is a characteristic parameter for speech (at least for the simple declarative English sentences studied). *D* seems to be unaffected by pitch (fundamental frequency), which is useful because it facilitates comparing different voices.

In humans, vocal stress is produced by increasing the subglottal air pressure via the lungs and is signaled by increased effort on the speaker’s part and usually by increased intensity and pitch (overall intensity and pitch do not show up in the *D* calculation). One motivation for this work was the comparison of natural and synthetic speech with the goal of improving synthetic speech. For this work, synthetic speech was produced using a digital speech synthesizer (see Section 3.2). Interestingly, synthesized speech sentences gave a *D* very close to human speech ($D = 1.57 \pm .03$), suggesting that the global structure of the human and synthetic speech is very similar. However, when *D* for vocal stress was calculated just for vowels, *D* monotonically increased with increasing stress for humans (*D* climbed to ~ 1.88 over the range of stress studied) but decreased or stayed the same for the synthesizer, suggesting a problem with vocal stress modelling for this synthesizer.

In human speech, the fractal dimension for nasalized and non-nasalized vowels is significantly different, though unlike the case for vocal stress, the direction of change differs with the vowel (e.g. for the “ah” vowel sound in father, *D* decreases by a few tenths, but for “ee” vowel sound in meet, *D* increases).[13] In the synthesizer, no change in *D* is detected. Again, the fact that the *D* trend for human and synthetic speech differs suggests that the nasalization method employed is not sufficient to adequately simulate nasality in human speech.

4.12.5 Catalog of Other Acoustic Sounds

The diversity and range of *D* for speech and other natural sounds is not fully known at this time. *D* computations were made for animal sounds (of about 2 seconds duration) and are: dolphin, $D = 1.90$; cat, $D = 1.74$; angry cat, $D = 1.78$; and very angry cat, $D = 1.91$. Whispered human speech gave $D = 1.49$.

It should be noted that *D* gives information which spectrograms do not provide (see Pickover and Khorasani, 1986). A power spectral slope cannot be directly related to *D* because it omits phase information which describes the *arrangement* of bumps in the graph of an object. In fact, two objects with different *D*s can have identical power spectra. For some educational demonstrations of this, see Pickover and Khorasani (1986e). A provocative area for future research would be to assess the extent to which *D* sensitivity to nasalization could be useful in providing feedback or diagnostic capabilities for certain speech pathologies where hyper/hyponasality can indicate anatomical or neurological difficulties in velar control or closure. Deaf speakers also produce inappropriate degrees of nasal resonance since they cannot hear the oral-nasal distinctions made by hearing speakers. (Pick86e and Khorasani, *Computers and Graph.;* Pick86f, *Computer Graph. Forum*.)

4.13 A Monte Carlo Approach to Fractal Dimension

Unfortunately, large amounts of computer time are sometimes needed to accurately do self-similarity studies. It is possible to develop fast computer techniques for the characterization of self-similar shapes and signals based upon Monte Carlo methods. The algorithm is specifically designed for digitized input (e.g. pictures, acoustic waveforms, analytic functions) where the self-similarity is not obvious from visual inspection of just a few sample magnifications. Pickover (1986) has several visual aids for conceptualizing the Monte Carlo process (see reference above).

In brief, a Monte Carlo approach can be used for choosing grid points (as in the last section) which can speed the *D* computation considerably. The name “Monte Carlo” conveys the idea of chance or randomness inherent in a method. Monte Carlo calculations are often used in physics when modelling complex phenomena requiring an exorbitant number of terms and calculations if done explicitly, for example using only a percentage of the number of atoms in a molecule when simulating x-ray scattering phenomena. For fractals, instead of searching every grid square of a picture or graph at a particular value of *e*, only a small random subset of grid squares is studied to represent the entire population of grid elements. This was tested for many speech sentence waveforms for different speakers, using different random numbers, and the Monte Carlo method gave *D* values very close to *D* determined by the explicit method. *D* was computed from the slope of a line fit by linear regression in plots of log *N*( ) *vs*. log . Normally intractable problems, such as the assessment of self-similarity for several minutes of speech, now become manageable. (Pick86f, *Computer Graph. Forum*.)

4.14 Other Speech Graphics, Wigner Distributions, FM Synthesis

Other interesting ways to display and synthesize speech are described in the above references which discuss, among other things, Wigner distributions and FM synthesis of speech sounds. I present some technical notes on these areas below, and the reader is directed to the published papers for a more detailed account. Readers not intimate with signal processing theory should skip the next section.

4.14.1 Wigner Distribution

The Wigner distribution, originally used in quantum statistical mechanics, can be used to create trivariate representations for speech signals which are similar to spectrograms. [In the past few years there has been considerable interest in a class of joint distributions which potentially offer a powerful technique for the study of signals. The unique feature of these distributions is that they satisfy the correct marginals,

$$\int_{-\infty}^{\infty} F(t, \omega)\, d\omega = |s(t)|^2$$

(4.7)

$$\int_{-\infty}^{\infty} F(t, \omega)\, dt = |S(\omega)|^2$$

(4.8)

where *F*(*t*,) is the joint distribution, *s*(*t*) the signal, and *S*( ) the spectrum:

$$S(\omega) = \frac{1}{\sqrt{(2\pi)}} \int_{-\infty}^{\infty} s(t) e^{-it\omega}\, dt$$

(4.9)

Furthermore the joint distribution is uniquely related to the signal s(t):

$$s(t)\, s^*(t') = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{F(t'', \omega)}{f_N(\theta, t - t')}\; e^{i\theta(t''-t+t'/2) + (t-t')\omega} dt''\, d\omega\, d\theta$$

(4.10)

where the signal *s*(*t*) is obtained by taking any convenient value for *t'* such as zero. The standard spectrogram does not posses either of these two properties. In particular, we studied bilinear distributions. All bilinear distributions can be obtained from

$$F(t, \omega) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-i\theta t - i\tau\omega + i\theta u} f(\theta, \tau) \bullet s^*(u - \tau/2)\, s(u + \tau/2) du\, d\tau\, d\theta$$

(4.11)

and particular distributions are obtained by choosing different functions for the kernel *f*(,) such that *f*(0,) = *f*(, 0) = 1. Using *f*(,) = 1, cos(/2), e$^{i\;\;|\;\;|/2}$, one obtains the Wigner-Ville, the Margenau-Hill-Rihaczek, and the Page distributions respectively.] Our research compares several joint time-frequency distributions for speech (Cohen and Pickover, 1986c, *IEEE Int. Conf. on Circuits & Syst.*).

4.14.2 FM Synthesis of Speech

In 1985, the Yamaha Corporation offered the first commercial music synthesizer based on FM synthesis. FM synthesis involves the modulation of one pure sine wave with another; this produces a variety of complex and natural-sounding waveforms. In conventional music (and speech) synthesizers, basic input waveforms (such as sawtooth and rectangular) are filtered to create the final output sounds. In our research, we have applied the FM synthesis approach to speech sounds (Pick87d). See also Chowning (1973) for pioneering work in this field. Note that (Pick85f) describes a system called "TUSK" with 30 different displays for analyzing speech.

4.15 Molecular Genetics: DNA Vectorgram

4.15.1 Background

As we said in "Cancer Genes (DNA Waveforms)" on page 33, DNA contains the basic genetic information of all living cells. The sequences of bases of DNA (adenine, cytosine, guanine, and thymine — A,C,G, and T) may hold information concerning protein synthesis as well as a variety of regulatory signals. For example, specific AT-rich regions are thought to be codes for beginning transcription. Also, certain *specific* viral sequences elicit cancerous changes in cells in artificial media and in animals. In addition to containing such regulatory codes and tumor-promoting codes, DNA sequence and composition are often correlated with physical properties of the DNA such as the DNA melting temperature.

Fairly detailed comparisons between DNA sequences are useful and can be achieved by a variety of brute-force statistical computations, but sometimes at a cost of the loss of an intuitive feeling for the structures. Differences between sequences may obscure the similarities. Even determining whether a particular sequence is *random* is curiously difficult. The approaches described in this section provide a way for simply representing and comparing random and DNA sequences in such a way that several sequence features may be detected by the analyst's eye.

4.15.2 DNA Vectorgram

The "vectorgrams" sometimes look like the steps a drunkard would take wandering in an open field. They can also be used to search for patterns in the sequence of bases in DNA. The method involves the conversion of the DNA sequence to binary data and subsequent mapping of the data to a two-dimensional pattern on a cellular lattice. For the example presented in this section, triply bonded bases (GC) are differentiated from doubly bonded bases (AT) by assigning nucleotide input values as follows: G=1, C=1, A=0, T=0. As region another combination which is the logical inverse (e.g. G and A interchanged; 010 vs. 101), then the net movement will be zero. A repeating sequence such as ...GGGGAAGAATACGAGGGGAA... generates a trace that returns to its starting point.

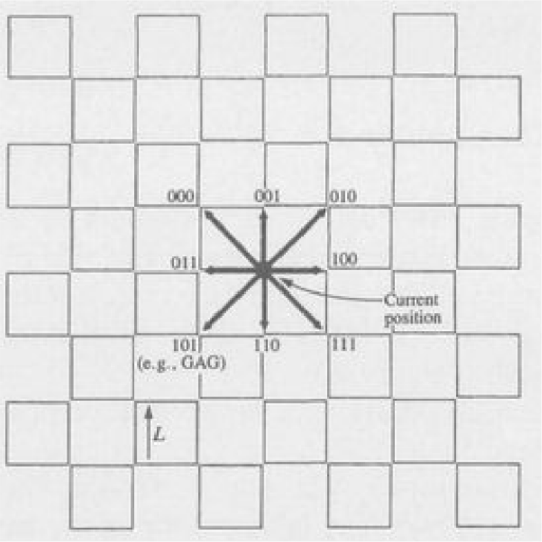


Figure 4.18. DNA transformation. The mapping of the digit strings into characteristic two-dimensional patterns traced out on a cellular lattice of cell length L. Each of the three digit combinations causes a vector to be drawn from a point on the lattice to one of the eight points immediately adjacent according to the coding system shown.

Figure 4.19 shows a DNA vectorgram for a random input sequence. This is useful for comparison with the DNA sequence to follow which is visually far from random. The radius of the circle centered at the origin indicates how far the sequence is expected to travel by chance.

It was quite startling to see such a large difference in the vectorgram produced by a real DNA sequence as compared with the vectorgram in Figure 4.19. An example of the output of the graphics system for a large DNA sequence represented by the dots in Figure 4.17 is presented in Figure 4.20. The calculation was performed for a human bladder oncogene consisting of about 4000 bases (Reddy, 1983). The vectorgram, far from being random, travels a mostly downward course indicating strings containing a predominance of 1’s (011,101,111). The most prominent feature on the map is the “kink” (global shift in direction) at about 1350, and interestingly this feature corresponds to a biologically important area of the DNA sequence. Magnifications of the fine structure of the vectorgram reveal additional interesting patterns (loops, hairpins, etc.). The authors’s current work using 3-D lattices indicates the usefulness of intricate 3-D spatial patterns to represent genetic sequences and other kinds of data (Figure 4.21).

One can study a number of other cancer genes with this approach and examine the usefulness of the vectorgram in capturing patterns not easy to find using other traditional approaches. (See: Pick87a, *IBM J. of Res. and Dev*. for magnifications of the regions in bubbles in Figure 4.20. Also see: Pick89, *Speculations in Science and Tech*.)

4.16 Reading List for Chapter 4

Several excellent reference books describing prior work in the field of unusual graphic representations were listed in the beginning of this chapter. In addition, there is a growing literature on the Chernoff face representation. For some good references, see: Chernoff, H. (1973), Chernoff and Rizvi (1975), Flury and Riedwyl (1981), and Jacob et al. (1976).

For more information on the fractal characterization of natural objects, see Mandelbrot’s work (Mandelbrot, 1983). For information on 1/ noise in music and speech, see the various papers by Voss (e.g., Clark and Voss (1975), Voss and Clarke (1978), and Voss (1979)). The bibliography at the end of this book has additional references.

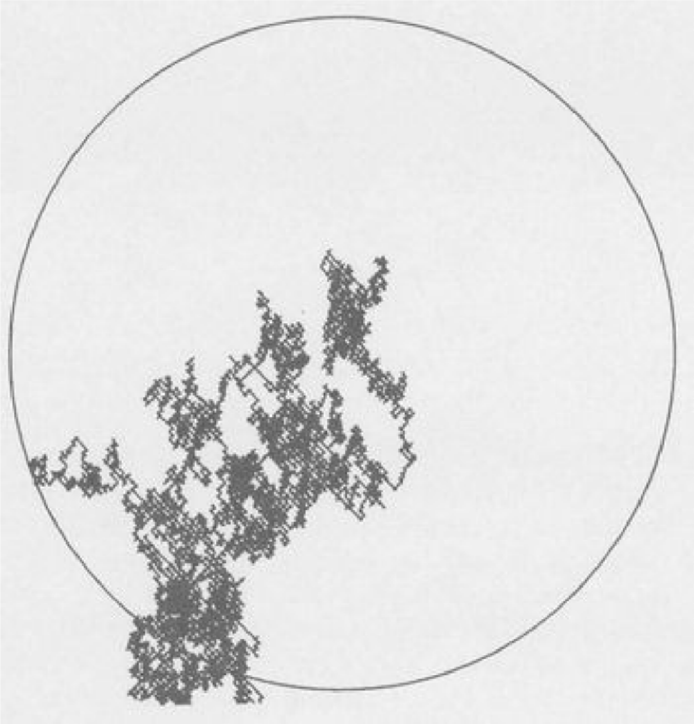


***Figure 4.19**. Random nucleotide-base input sequence. This is useful for visual comparison with the DNA sequence in Figure 4.20, which is far from random.*

Index