Chapter 1

Introduction and Overview

Contents

1.1 Computational Cognitive Neuroscience

How does the brain think? This is one of the most challenging unsolved questions in science. Armed with new methods, data, and ideas, researchers in a variety of fields bring us closer to fully answering this question each day. We can even watch the brain as it thinks, using modern neuroimaging machines that record the biological shadows of thought and transform them into vivid color images. These amazing images, together with the results from many other important techniques, have advanced our understanding of the neural bases of cognition considerably. We can consolidate these various different approaches under the umbrella discipline of **cognitive neuroscience**, which has as its goal answering this most important of scientific questions.

Cognitive neuroscience will remain a frontier for many years to come, because both thoughts and brains are incredibly complex and difficult to understand. Sequences of images of the brain thinking reveal a vast network of glowing regions that interact in complex ways with changing patterns of thought. Each picture is worth a thousand words — indeed, language often fails us in the attempt to capture the richness and subtlety of it all. **Computational** models based on biological properties of the brain can provide an important tool for understanding all of this complexity. Such models can capture the flow of information from your eyes recording these letters and words, up to the parts of your brain activated by the different word meanings, resulting in an integrated comprehension of this text. Although our understanding of such phenomena is still incomplete, these models enable us to explore their underlying mechanisms, which we can implement on a computer and manipulate, test, and ultimately understand.

This book provides an introduction to this emerging subdiscipline known as **computational cognitive neuroscience**: simulating human cognition using biologically based networks of neuronlike units (**neural networks**). We provide a textbook-style treatment of the central ideas in this field, integrated with computer sim-

ulations that allow readers to undertake their own **explorations** of the material presented in the text. An important and unique aspect of this book is that the explorations include a number of large-scale simulations used in recent original research projects, giving students and other researchers the opportunity to examine these models up close and in detail.

In this chapter, we present an overview of the basic motivations and history behind computational cognitive neuroscience, followed by an overview of the subsequent chapters covering basic neural computational mechanisms (part I) and cognitive phenomena (part II). Using the neural network models in this book, you will be able to explore a wide range of interesting cognitive phenomena, including:

Visual encoding: A neural network will view natural scenes (mountains, trees, etc.), and, using some basic principles of learning, will develop ways of encoding these visual scenes much like those your brain uses to make sense of the visual world.

Spatial attention: By taking advantage of the interactions between two different streams of visual processing, you can see how a model focuses its attention in different locations in space, for example to scan a visual scene. Then, you can use this model to simulate the attention performance of normal and brain-damaged people.

Episodic memory: By incorporating the structure of the brain area called the *hippocampus,* a neural network will become able to form new memories of everyday experiences and events, and will simulate human performance on memory tasks.

Working memory: You will see that specialized biological mechanisms can greatly improve a network's *working memory* (the kind of memory you need to multiply 42 by 17 in your head, for example). Further, you will see how the skilled control of working memory can be learned through experience.

Word reading: You can see how a network can learn to read and pronounce nearly 3,000 English words. Like human subjects, this network can pronounce novel nonwords that it has never seen before (e.g., "mave" or "nust"), demonstrating that it is not simply memorizing pronunciations — instead, it learns the complex web of regularities that govern English pronunciation. And, by damaging a model that captures the many different ways that words are represented in the brain, you can simulate various forms of dyslexia.

Semantic representation: You can explore a network that has "read" every paragraph in this textbook and in the process acquired a surprisingly good understanding of the words used therein, essentially by noting which words tend to be used together or in similar contexts.

Task directed behavior: You can explore a model of the "executive" part of the brain, the *prefrontal cortex*, and see how it can keep us focused on performing the task at hand while protecting us from getting distracted by other things going on.

Deliberate, explicit cognition: A surprising number of things occur relatively automatically in your brain (e.g., you are not aware of exactly how you translate these black and white strokes on the page into some sense of what these words are saying), but you can also think and act in a deliberate, explicit fashion. You'll explore a model that exhibits both of these types of cognition within the context of a simple categorization task, and in so doing, provides the beginnings of an account of the biological basis of conscious awareness.

1.2 Basic Motivations for Computational Cognitive Neuroscience

1.2.1 Physical Reductionism

The whole idea behind cognitive neuroscience is the once radical notion that the mysteries of human thought can be explained in much the same way as everything else in science — by reducing a complex phenomenon (cognition) into simpler components (the underlying biological mechanisms of the brain). This process is just **reductionism**, which has been and continues to be the standard method of scientific advancement across most fields. For example, all matter can be reduced to its atomic components, which helps to explain the various

properties of different kinds of matter, and the ways in which they interact. Similarly, many biological phenomena can be explained in terms of the actions of underlying DNA and proteins.

Although it is natural to think of reductionism in terms of physical systems (e.g., explaining cognition in terms of the physical brain), it is also possible to achieve a form of reductionism in terms of more abstract components of a system. Indeed, one could argue that all forms of explanation entail a form of reductionism, in that they explain a previously inexplicable thing in terms of other, more familiar constructs, just as one can understand the definition of an unfamiliar word in the dictionary in terms of more familiar words.

There have been many attempts over the years to explain human cognition using various different languages and metaphors. For example, can cognition be explained by assuming it is based on simple logical operations? By assuming it works just like a standard serial computer? Although these approaches have borne some fruit, the idea that one should look to the brain itself for the language and principles upon which to explain human cognition seems more likely to succeed, given that the brain is ultimately responsible for it all. Thus, it is not just reductionism that defines the essence of cognitive neuroscience — it is also the stipulation that the components be based on the physical substrate of human cognition, the brain. This is **physical reductionism**.

As a domain of scientific inquiry matures, there is a tendency for constructs that play a role in that domain to become physically grounded. For example, in the biological sciences before the advent of modern molecular biology, ephemeral, vitalistic theories were common, where the components were posited based on a theory, not on any physical evidence for them. As the molecular basis of life was understood, it became possible to develop theories of biological function in terms of real underlying components (proteins, nucleic acids, etc.) that can be measured and localized. Some prephysical theoretical constructs accurately anticipated their physically grounded counterparts; for example, Mendel's theory of genetics anticipated many important functional aspects of DNA replication, while others did not fare so well.

Similarly, many previous and current theories of human cognition are based on constructs such as "attention" and "working memory buffers" that are based on an analysis of behaviors or thoughts, and not on physical entities that can be independently measured. Cognitive neuroscience differs from other forms of cognitive theorizing in that it seeks to explain cognitive phenomena in terms of underlying **neurobiological** components, which can in principle be independently measured and localized. Just as in biology and other fields, some of the nonphysical constructs of cognition will probably fit well with the underlying biological mechanisms, and others may not (e.g., Churchland, 1986). Even in those that fit well, understanding their biological basis will probably lead to a more refined and sophisticated understanding (e.g., as knowing the biological structure of DNA has for understanding genetics).

1.2.2 Reconstructionism

However, reductionism in all aspects of science — particularly in the study of human cognition — can suffer from an inappropriate emphasis on the process of reducing phenomena into component pieces, without the essential and complementary process of using those pieces to reconstruct the larger phenomenon. We refer to this latter process as **reconstructionism**. It is simply not enough to say that the brain is made of neurons; one must explain how billions of neurons interacting with each other produce human cognition. Teitelbaum (1967) argued for a similar complementarity of scientific processes — analysis and synthesis — in the study of physiological psychology. Analysis entails dissecting and simplifying a system to understand its essential elements; synthesis entails combining elements and understanding their interactions.

The computational approach to cognitive neuroscience becomes critically important in reconstructionism: it is very difficult to use verbal arguments to reconstruct human cognition (or any other complex phenomenon) from the action of a large number of interacting components. Instead, we can implement the behavior of these components in a computer program and test whether they are indeed capable of reproducing the desired phenomena. Such simulations are crucial to developing our understanding of how neurons produce cog-

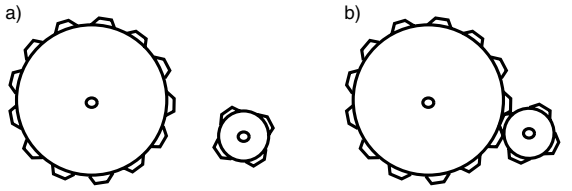

Figure 1.1: Illustration of the importance of reconstructionism — it is not enough to say that the system is composed of components (e.g., two gears as in **a**), one must also show how these components interact to produce overall behaviors. In **b**, the two gears interact to produce changes in rotational speed and torque — these effects emerge from the interaction, and are not a property of each component individually.

nition. This is especially true when there are **emergent phenomena** that arise from these interactions without obviously being present in the behavior of individual elements (neurons) — where the whole is greater than the sum of its parts. The importance of reconstructionism is often overlooked in all areas of science, not just cognitive neuroscience, and the process has really only recently become feasible with the advent of relatively affordable fast computers.

Figure 1.1 shows a simple illustration of the importance of reconstructionism in understanding how systems behave. Here, it is not sufficient to say that the system is composed of two components (the two gears shown in panel a). Instead, one must also specify that the gears interact as shown in panel b, because it is only through this interaction that the important "behavioral" properties of changes in rotational speed and torque can emerge. For example, if the smaller gear drives the larger gear, this achieves a decrease in rotational speed and an increase in torque. However, if this same driving gear were to interact with a gear that was even smaller than it, it would produce the opposite effect. This is essentially what it means for the behavior to emerge from the interaction between the two gears, because it is clearly not a property of the individual gears in isolation. Similarly, cognition is an emergent phenomenon of the interactions of billions of neurons. It is not sufficient to say that the cognitive system is composed of billions of neurons; we must instead specify how these neurons interact to produce cognition.

1.2.3 Levels of Analysis

Although the physical reductionism and reconstructionism motivations behind computational cognitive neuroscience may appear sound and straightforward, this approach to understanding human cognition is challenged by the extreme complexity of and lack of knowledge about both the brain and the cognition it produces. As a result, many researchers have appealed to the notion of hierarchical **levels** of analysis to deal with this complexity. Clearly, some levels of underlying mechanism are more appropriate for explaining human cognition than others. For example, it appears foolhardy to try to explain human cognition directly in terms of atoms and simple molecules, or even proteins and DNA. Thus, we must focus instead on higher level mechanisms. However, exactly which level is the "right" level is an important issue that will only be resolved through further scientific investigation. The level presented in this book represents our best guess at this time.

One approach toward thinking about the issue of levels of analysis was suggested by David Marr (1982), who introduced the seductive notion of **computational**, **algorithmic**, and **implementational** levels by forging an analogy with the computer. Take the example of a program that sorts a list of numbers. One can specify in very abstract terms that the computation performed by this program is to arrange the numbers such that the smallest one is first in the list, the next largest one is next, and so on. This abstract computational level of analysis is useful for specifying what different programs do, without worrying about exactly how they go about doing it. Think of it as the "executive summary."

The algorithmic level then delves into more of the details as to how sorting actually occurs — there are many different strategies that one could adopt, and they have various tradeoffs in terms of factors such as speed or amount of memory used. Critically, the algorithm provides just enough information to implement the program, but does not specify any details about what language to program it in, what variable names to use, and so on. These details are left for the implementational level — how the program is actually written and executed on a particular computer using a particular language.

Marr's levels and corresponding emphasis on the computational and algorithmic levels were born out of the early movements of **artificial intelligence**, **cognitive psychology**, and **cognitive science**, which were based on the idea that one could ignore the underlying biological mechanisms of cognition, focusing instead on identifying important computational or cognitive level properties. Indeed, these traditional approaches were based on the assumption that the brain works like a standard computer, and thus that Marr's computational and algorithmic levels were much more important than the "mere details" of the underlying neurobiological implementation.

The **optimality** or **rational analysis** approach, which is widely employed across the "sciences of complexity" from biology to psychology and economics (e.g., Anderson, 1990), shares the Marr-like emphasis on the computational level. Here, one assumes that it is possible to identify the "optimal" computation or function performed by a person or animal in a given context, and that whatever the brain is doing, it must somehow be accomplishing this same optimal computation (and can therefore be safely ignored). For example, Anderson (1990) argues that memory retention curves are optimally tuned to the expected frequency and spacing of retrieval demands for items stored in memory. Under this view, it doesn't really matter how the memory retention mechanisms work, because they are ultimately driven by the optimality criterion of matching expected demands for items, which in turn is assumed to follow general laws.

Although the optimality approach may sound attractive, the definition of optimality all too often ends up being conditioned on a number of assumptions (including those about the nature of the underlying implementation) that have no real independent basis. In short, optimality can rarely be defined in purely "objective" terms, and so often what is optimal in a given situation depends on the detailed circumstances.

Thus, the dangerous thing about both Marr's levels and these optimality approaches is that they appear to suggest that the implementational level is largely irrelevant. In most standard computers and languages, this is true, because *they are all effectively equivalent at the implementational level*, so that the implementational issues don't really affect the algorithmic and computational levels of analysis. Indeed, computer algorithms can be turned into implementations by the completely automatic process of compilation. In contrast, in the brain, the neural implementation is certainly not derived automatically from some higher-level description, and thus it is not obviously true that it can be easily described at these higher levels.

In effect, the higher-level computational analysis has already *assumed* a general implementational form, without giving proper credit to it for shaping the whole enterprise in the first place. However, with the advent of parallel computers, people are beginning to realize the limitations of computation and algorithms that assume the standard serial computer with address-based memory — entirely new classes of algorithms and ways of thinking about problems are being developed to take advantage of parallel computation. Given that the brain is clearly a parallel computer, having billions of computing elements (neurons), one must be very careful in importing seductively simple ideas based on standard computers.

On the other end of the spectrum, various researchers have emphasized the *implementational* level as primary over the computational and algorithmic. They have argued that cognitive models should be assembled by making extremely detailed replicas of neurons, thus guaranteeing that the resulting model contains all of the important biological mechanisms (e.g., Bower, 1992). The risk of this approach is complementary to those that emphasize a purely computational approach: without any clear understanding of which biological properties are functionally important and which are not, one ends up with massive, complicated models that are difficult to understand, and that provide little insight into the critical properties of cognition. Further, these models inevitably fail to represent *all* of the biological mechanisms in their fullest possible detail, so one can never be quite sure that something important is not missing.

Instead of arguing for the superiority of one level over the other, we adopt a fully **interactive, balanced** approach, which emphasizes forming connections between data across all of the relevant levels, and striking a reasonable balance between the desire for a simplified model and the desire to incorporate as much of the

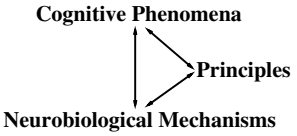

Figure 1.2: The two basic levels of analysis used in this text, with an intermediate level to help forge the links.

known biological mechanisms as possible. There is a place for both **bottom-up** (i.e., working from biological facts "up" to cognition), **top-down** (i.e., working from cognition "down" to biological facts), and, most important, interactive approaches, where one tries to simultaneously take into account constraints at the biological and cognitive levels.

For example, it can be useful to take a set of facts about how neurons behave, encode them in a set of equations in a computer program, and see how the kinds of behaviors that result depend on the properties of these neurons. It can also be useful to think about what cognition should be doing in a particular case (e.g., at the computational level, or on some other principled basis), and then derive an implementation that accomplishes this, and see how well that characterizes what we know about the brain, and how well it does the cognitive job it is supposed to do. This kind of interplay between neurobiological, cognitive and principled (computational and otherwise) considerations is emphasized throughout the text.

To summarize our approach, and to avoid the unintended associations with Marr's terminology, we adopt the following hierarchy of analytical levels (figure 1.2). At its core, we have essentially a simple bi-level physical reductionist/reconstructionist hierarchy, with a lower level consisting of *neurobiological mechanisms,* and an upper level consisting of *cognitive phenomena*. We will reduce cognitive phenomena to the operation of neurobiological mechanisms, and show, through simulations, how these mechanisms produce emergent cognitive phenomena. Of course, our simulations will have to rely on simplified, abstracted renditions of the neurobiological mechanisms.

To help forge links between these two levels of analysis, we have an auxiliary intermediate level consisting of *principles* presented throughout the text. We do not think that the brain nor cognition can be fully described by these principles, which is why they play an auxiliary role and are shown off to one side of the figure. However, they serve to highlight and make clear the connection between certain aspects of the biology and certain aspects of cognition. Often, these principles are based on computational-level descriptions of aspects of cognition. But, we want to avoid any implication that these principles provide some privileged level of description (i.e., like Marr's view of the computational level), that tempts us into thinking that data at the two basic empirical levels (cognition and neurobiology) are less relevant. Instead, these principles are fundamentally shaped by, and help to strike a good balance between, the two primary levels of analysis.

The levels of analysis issue is easily confused with different levels of *structure* within the nervous system, but these two types of levels are not equivalent. The relevant levels of structure range from molecules to individual neurons to small groups or columns of neurons to larger areas or regions of neurons up to the entire brain itself. Although one might be tempted to say that our cognitive phenomena level of analysis should be associated with the highest structural level (the entire brain), and our neurobiological mechanisms level of analysis associated with lower structural levels, this is not really accurate. Indeed, some cognitive phenomena can be traced directly to properties of individual neurons (e.g., that they exhibit a fatiguelike phenomenon if activated too long), whereas other cognitive phenomena only emerge as a result of interactions among a number of different brain areas. Furthermore, as we progress from lower to higher structural levels in successive chapters of this book, we emphasize that specific computational principles and cognitive phenomena can be associated with each of these structural levels. Thus, just as there is no privileged level of analysis, there is no privileged structural level — all of these levels must be considered in an interactive fashion.

1.2.4 Scaling Issues

Having adopted essentially two levels of analysis, we are in the position of using biological mechanisms op-

erating at the level of individual neurons to explain even relatively complex, high-level cognitive phenomena. This raises the question as to why these basic neural mechanisms should have any relevance to understanding something that is undoubtedly the product of millions or even billions of neurons — certainly we do not include anywhere near that many neurons in our simulations! This **scaling issue** relates to the way in which we construct a scaled-down model of the real brain. It is important to emphasize that the need for scaling is at least partially a pragmatic issue having to do with the limitations of currently available computational resources. Thus, it should be possible to put the following arguments to the test in the future as larger, more complex models can be constructed. However, scaled-down models are also easier to understand, and are a good place to begin the computational cognitive neuroscience enterprise.

We approach the scaling problem in the following ways.

- The target cognitive behavior that we expect (and obtain) from the models is similarly scaled down compared to the complexities of actual human cognition.
- We show that one of our simulated neurons (units) in the model can approximate the behavior of many real neurons, so that we can build models of multiple brain areas where the neurons in those areas are simulated by many fewer units.
- We argue that information processing in the brain has a **fractal** quality, where the same basic properties apply across disparate physical scales. These basic properties are those of individual neurons, which "show through" even at higher levels, and are thus relevant to understanding even the large-scale behavior of the brain.

The first argument amounts to the idea that our neural network models are performing essentially the same type of processing as a human in a particular task, but on a reduced problem that either lacks the detailed information content of the human equivalent or represents a subset of these details. Of course, many phenomena can become qualitatively different as they get scaled up

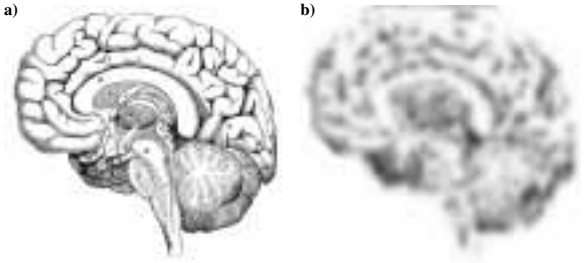

Figure 1.3: Illustration of scaling as performed on an image — the original image in (a) was scaled down by a factor of 8, retaining only 1/8th of the original information, and then scaled back up to the same size and averaged (blurred) to produce (b), which captures many of the general characteristics of the original, but not the fine details. Our models give us something like this scaled-down, averaged image of how the brain works.

or down along this content dimension, but it seems reasonable to allow that some important properties might be relatively scale invariant. For example, one could plausibly argue that each major area of the human cortex could be reduced to handle only a small portion of the content that it actually does (e.g., by the use of a $16x16$ pixel retina instead of 16 million x 16 million pixels), but that some important aspects of the essential computation on any piece of that information are preserved in the reduced model. If several such reduced cortical areas were connected, one could imagine having a useful but simplified model of some reasonably complex psychological phenomena.

The second argument can perhaps be stated most clearly by imagining that an individual unit in the model approximates the behavior of a population of essentially identical neurons. Thus, whereas actual neurons are discretely spiking, our model units typically (but not exclusively) use a continuous, graded activation signal. We will see in chapter 2 that this graded signal provides a very good approximation to the average number of spikes per unit time produced by a population of spiking neurons. Of course, we don't imagine that the brain is constructed from populations of identical neurons, but we do think that the brain employs overlapping distributed representations, so that an individual model unit can represent the centroid of a set of such repre-

sentations. Thus, the population can encode much more information (e.g., many finer shades of meaning), and is probably different in other important ways (e.g., it might be more robust to the effects of noise). A visual analogy for this kind of scaling is shown in figure 1.3, where the sharp, high-resolution detail of the original (panel a) is lost in the scaled-down version (panel b), but the basic overall structure is preserved.

Finally, we believe that the brain has a fractal character for two reasons: First, it is likely that, at least in the cortex, the effective properties of long-range connectivity are similar to that of local, short-range connectivity. For example, both short and long-range connectivity produce a balance between excitation and inhibition by virtue of connecting to both excitatory and inhibitory neurons (more on this in chapter 3). Thus, a model based on the properties of short-range connectivity within a localized cortical area could also describe a larger-scale model containing many such cortical areas simulated at a coarser level. The second reason is basically the same as the one given earlier about averaging over populations of neurons: if on average the population behaves roughly the same as the individual neuron, then the two levels of description are self-similar, which is what it means to be fractal.

In short, these arguments provide a basis for optimism that models based on neurobiological data can provide useful accounts of cognitive phenomena, even those that involve large, widely distributed areas of the brain. The models described in this book substantiate some of this optimism, but certainly this issue remains an open and important question for the computational cognitive neuroscience enterprise. The following historical perspective on this enterprise provides an overview of some of the other important issues that have shaped the field.

1.3 Historical Context

Although the field of computational cognitive neuroscience is relatively young, its boundaries are easily blurred into a large number of related disciplines, some of which have been around for quite some time. Indeed, research in any aspect of cognition, neuroscience, or computation has the potential to make an important contribution to this field. Thus, the entire space of this book could be devoted to an adequate account of the relevant history of the field. This section is instead intended to merely provide a brief overview of some of the particularly relevant historical context and motivation behind our approach. Specifically, we focus on the advances in understanding how networks of simulated neurons can lead to interesting cognitive phenomena, which occurred initially in the 1960s and then again in the period from the late '70s to the present day. These advances form the main heritage of our approach because, as should be clear from what has been said earlier, the neural network modeling approach provides a crucial link between networks of neurons and human cognition.

The field of **cognitive psychology** began in the late 1950s and early '60s, following the domination of the behaviorists. Key advances associated with this new field included its emphasis on *internal mechanisms* for mediating cognition, and in particular the use of *explicit computational models* for simulating cognition on computers (e.g., problem solving and mathematical reasoning; Newell & Simon, 1972). The dominant approach was based on the **computer metaphor**, which held that human cognition is much like processing in a standard serial computer.

In such systems, which we will refer to as "traditional" or "symbolic," the basic operations involve **symbol manipulation** (e.g., manipulating logical statements expressed using dynamically-bound variables and operators), and processing consists of a sequence of **serial**, **rule-governed** steps. **Production systems** became the dominant framework for cognitive modeling within this approach. **Productions** are essentially elaborate if-then constructs that are activated when their if-conditions are met, and they then produce actions that enable the firing of subsequent productions. Thus, these productions control the sequential flow of processing. As we will see, these traditional, symbolic models serve as an important contrast to the neural-network framework, and the two have been in a state of competition from the earliest days of their existence.

Even though the computer metaphor was dominant, there was also considerable interest in neuronlike processing during this time, with advances like: (a) the

McCulloch and Pitts (1943) model of neural processing in terms of basic logical operations; (b) Hebb's (1949) theory of **Hebbian learning** and the **cell assembly**, which holds that connections between coactive neurons should be strengthened, joining them together; and (c) Rosenblatt's (1958) work on the **perceptron** learning algorithm, which could learn from **error signals**. These computational approaches built on fundamental advances in neurobiology, where the idea that the neuron is the primary information processing unit of the brain became established (the "neuron doctrine"; Shepherd, 1992), and the basic principles of neural communication and processing (action potentials, synapses, neurotransmitters, ion channels, etc.) were being developed. The dominance of the computer metaphor approach in cognitive psychology was nevertheless sealed with the publication of the book *Perceptrons* (Minsky & Papert, 1969), which proved that some of these simple neuronlike models had significant computational limitations — they were unable to learn to solve a large class of basic problems.

While a few hardy researchers continued studying these neural-network models through the '70s (e.g., Grossberg, Kohonen, Anderson, Amari, Arbib, Willshaw), it was not until the '80s that a few critical advances brought the field back into real popularity. In the early '80s, psychological (e.g., McClelland & Rumelhart, 1981) and computational (Hopfield, 1982, 1984) advances were made based on the activation dynamics of networks. Then, the backpropagation learning algorithm was rediscovered by Rumelhart, Hinton, and Williams (1986b) (having been independently discovered several times before: Bryson & Ho, 1969; Werbos, 1974; Parker, 1985) and the *Parallel Distributed Processing (PDP)* books (Rumelhart et al., 1986c; McClelland et al., 1986) were published, which firmly established the credibility of neural network models. Critically, the backpropagation algorithm eliminated the limitations of the earlier models, enabling essentially any function to be learned by a neural network. Another important advance represented in the PDP books was a strong appreciation for the importance of **distributed representations** (Hinton, McClelland, & Rumelhart, 1986), which have a number of computational advantages over symbolic or localist representations.

Backpropagation led to a new wave of cognitive modeling (which often goes by the name **connectionism**). Although it represented a step forward computationally, backpropagation was viewed by many as a step backward from a biological perspective, because it was not at all clear how it could be implemented by biological mechanisms (Crick, 1989; Zipser & Andersen, 1988). Thus, backpropagation-based cognitive modeling carried on without a clear biological basis, causing many such researchers to use the same kinds of arguments used by supporters of the computer metaphor to justify their approach (i.e., the "computational level" arguments discussed previously). Some would argue that this deemphasizing of the biological issues made the field essentially a reinvented computational cognitive psychology based on "neuronlike" processing principles, rather than a true computational cognitive neuroscience.

In parallel with the expanded influence of neural network models in understanding cognition, there was a rapid growth of more biologically oriented modeling. We can usefully identify several categories of this type of research. First, we can divide the biological models into those that emphasize learning and those that do not. The models that do not emphasize learning include detailed biophysical models of individual neurons (Traub & Miles, 1991; Bower, 1992), information-theoretic approaches to processing in neurons and networks of neurons (e.g., Abbott & LeMasson, 1993; Atick & Redlich, 1990; Amit, Gutfreund, & Sompolinsky, 1987; Amari & Maginu, 1988), and refinements and extensions of the original Hopfield (1982, 1984) models, which hold considerable appeal due to their underlying mathematical formulation in terms of concepts from statistical physics. Although this research has led to many important insights, it tends to make less direct contact with cognitively relevant issues (though the Hopfield network itself provides some centrally important principles, as we will see in chapter 3, and has been used as a framework for some kinds of learning).

The biologically based learning models have tended to focus on learning in the early visual system, with an emphasis on Hebbian learning (Linsker, 1986; Miller, Keller, & Stryker, 1989; Miller, 1994; Kohonen, 1984; Hebb, 1949). Importantly, a large body of basic neu-

roscience research supports the idea that Hebbian-like mechanisms are operating in neurons in most cognitively important areas of the brain (Bear, 1996; Brown, Kairiss, & Keenan, 1990; Collingridge & Bliss, 1987). However, Hebbian learning is generally fairly computationally weak (as we will see in chapter 5), and suffers from limitations similar to those of the 1960s generation of learning mechanisms. Thus, it has not been as widely used as backpropagation for cognitive modeling because it often cannot learn the relevant tasks.

In addition to the cognitive (connectionist) and biological branches of neural network research, considerable work has been done on the computational end. It has been apparent that the mathematical basis of neural networks has much in common with statistics, and the computational advances have tended to push this connection further. Recently, the use of the Bayesian framework for statistical inference has been applied to develop new learning algorithms (e.g., Dayan, Hinton, Neal, & Zemel, 1995; Saul, Jaakkola, & Jordan, 1996), and more generally to understand existing ones. However, none of these models has yet been developed to the point where they provide a framework for learning that works reliably on a wide range of cognitive tasks, while simultaneously being implementable by a reasonable biological mechanism. Indeed, most (but not all) of the principal researchers in the computational end of the field are more concerned with theoretical, statistical, and machine-learning kinds of issues than with cognitive or biological ones.

In short, from the perspective of the computational cognitive neuroscience endeavor, the field is in a somewhat fragmented state, with modelers in computational cognitive psychology primarily focused on understanding human cognition without close contact with the underlying neurobiology, biological modelers focused on information-theoretic constructs or computationally weak learning mechanisms without close contact with cognition, and learning theorists focused at a more computational level of analysis involving statistical constructs without close contact with biology or cognition. Nevertheless, we think that a strong set of cognitively relevant computational and biological principles has emerged over the years, and that the time is ripe for an attempt to consolidate and integrate these principles.

1.4 Overview of Our Approach

This brief historical overview provides a useful context for describing the basic characteristics of the approach we have taken in this book. Our core mechanistic principles include both backpropagation-based error-driven learning and Hebbian learning, the central principles behind the Hopfield network for interactive, constraint-satisfaction style processing, distributed representations, and inhibitory competition. The neural units in our simulations use equations based directly on the ion channels that govern the behavior of real neurons (as described in chapter 2), and our neural networks incorporate a number of well-established anatomical and physiological properties of the neocortex (as described in chapter 3). Thus, we strive to establish detailed connections between biology and cognition, in a way that is consistent with many well-established computational principles.

Our approach can be seen as an integration of a number of different themes, trends, and developments (O'Reilly, 1998). Perhaps the most relevant such development was the integration of a coherent set of neural network principles into the *GRAIN* framework of McClelland (1993). GRAIN stands for graded, random, adaptive, interactive, (nonlinear) network. This framework was primarily motivated by (and applied to) issues surrounding the dynamics of activation flow through a neural network. The framework we adopt in this book incorporates and extends these GRAIN principles by emphasizing learning mechanisms and the architectural properties that support them.

For example, there has been a long-standing desire to understand how more biologically realistic mechanisms could give rise to error-driven learning (e.g., Hinton & McClelland, 1988; Mazzoni, Andersen, & Jordan, 1991). Recently, a number of different frameworks for achieving this goal have been shown to be variants of a common underlying error propagation mechanism (O'Reilly, 1996a). The resulting algorithm, called *GeneRec,* is consistent with known biological mechanisms of learning, makes use of other biological properties of the brain (including interactivity), and allows for realistic neural activation functions to be used. Thus,

this algorithm plays an important role in our integrated framework by allowing us to use the principle of backpropagation learning without conflicting with the desire to take the biology seriously.

Another long-standing theme in neural network models is the development of inhibitory competition mechanisms (e.g., Kohonen, 1984; McClelland & Rumelhart, 1981; Rumelhart & Zipser, 1986; Grossberg, 1976). Competition has a number of important functional benefits emphasized in the GRAIN framework (which we will explore in chapter 3) and is generally required for the use of Hebbian learning mechanisms. It is technically challenging, however, to combine competition with distributed representations in an effective manner, because the two tend to work at cross purposes. Nevertheless, there are good reasons to believe that the kinds of sparse distributed representations that should in principle result from competition provide a particularly efficient means for representing the structure of the natural environment (e.g., Barlow, 1989; Field, 1994; Olshausen & Field, 1996). Thus, an important part of our framework is a mechanism of neural competition that is compatible with powerful distributed representations and can be combined with interactivity and learning in a way that was not generally possible before (O'Reilly, 1998, 1996b).

The emphasis throughout the book is on the facts of the biology, the core computational principles just described, which underlie most of the cognitive neural network models that have been developed to date, and their interrelationship in the context of a range of well-studied cognitive phenomena. To facilitate and simplify the hands-on exploration of these ideas by the student, we take advantage of a particular implementational framework that incorporates all of the core mechanistic principles called *Leabra* (**l***ocal,* **e***rror-driven and* **a***ssociative,* **bi***ologically* **r***ealistic* **a***lgorithm*). Leabra is pronounced like the astrological sign Libra, which emphasizes the *balance* between many different objectives that is achieved by the algorithm.

To the extent that we are able to understand a wide range of cognitive phenomena using a consistent set of biological and computational principles, one could consider the framework presented in this book to be a "first draft" of a coherent framework for computational cognitive neuroscience. This framework provides a useful consolidation of existing ideas, and should help to identify the limitations and problems that will need to be solved in the future.

Newell (1990) provided a number of arguments in favor of developing unified theories of cognition, many of which apply to our approach of developing a coherent framework for computational cognitive neuroscience. Newell argued that it is relatively easy (and thus relatively uninformative) to construct specialized theories of specific phenomena. In contrast, one encounters many more constraints by taking on a wider range of data, and a theory that can account for this data is thus much more likely to be true. Given that our framework bears little resemblance to Newell's SOAR architecture, it is clear that just the process of making a unified architecture does not guarantee convergence on some common set of principles. However, it is clear that casting a wider net imposes many more constraints on the modeling process, and the fact that the single set of principles can be used to model the wide range of phenomena covered in this book lends some measure of validity to the undertaking.

Chomsky (1965) and Seidenberg (1993) also discussed the value of developing *explanatory* theories that explain phenomena in terms of a small set of independently motivated principles, in contrast with *descriptive* theories that essentially restate phenomena.

1.5 General Issues in Computational Modeling

The preceding discussion of the benefits of a unified model raises a number of more general issues regarding the benefits of computational modeling[1] as a methodology for cognitive neuroscience. Although we think the benefits generally outweigh the disadvantages, it is also important to be cognizant of the potential traps and problems associated with this methodology. We will just provide a brief summary of these advantages and problems here.

[1] We consider both models that are explicitly simulated on a computer and more abstract mathematical models to be computational models, in that both are focused on the computational processing of information in the brain.

Advantages:

Models help us to understand phenomena. A computational model can provide novel sources of insight into behavior, for example by providing a counterintuitive explanation of a phenomenon, or by reconciling seemingly contradictory phenomena (e.g., by complex interactions among components). Seemingly different phenomena can also be related to each other in nonobvious ways via a common set of computational mechanisms.

Computational models can also be lesioned and then tested, providing insight into behavior following specific types of brain damage, and in turn, into normal functioning. Often, lesions can have nonobvious effects that computational models can explain.

By virtue of being able to translate between functional desiderata and the biological mechanisms that implement them, computational models enable us to understand not just how the brain is structured, but *why* it is structured in the way it is.

Models deal with complexity. A computational model can deal with complexity in ways that verbal arguments cannot, producing satisfying explanations of what would otherwise just be vague hand-wavy arguments. Further, computational models can handle complexity across multiple levels of analysis, allowing data across these levels to be integrated and related to each other. For example, the computational models in this book show how biological properties give rise to cognitive behaviors in ways that would be impossible with simple verbal arguments.

Models are explicit. Making a computational model forces you to be explicit about your assumptions and about exactly how the relevant processes actually work. Such explicitness carries with it many potential advantages.

First, explicitness can help in deconstructing psychological concepts that may rely on *homunculi* to do their work. A homunculus is a "little man," and many theories of cognition make unintended use of them by embodying particular components (often "boxes") of the theory with magical powers that end up doing all the work in the theory. A canonical example is the "executive" theory of prefrontal cortex function: if you posit an executive without explaining how it makes all those good decisions and coordinates all the other brain areas, you haven't explained too much (you might as well just put pinstripes and a tie on the box).

Second, an explicitly specified computational model can be run to generate novel predictions. A computational model thus forces you to accept the consequences of your assumptions. If the model must be modified to account for new data, it becomes very clear exactly what these changes are, and the scientific community can more easily evaluate the resulting deviance from the previous theory. Predictions from verbal theories can be tenuous due to lack of specificity and the flexibility of vague verbal constructs.

Third, explicitness can contribute to a greater appreciation for the complexities of otherwise seemingly simple processes. For example, before people tried to make explicit computational models of object recognition, it didn't seem that difficult or interesting a problem — there is an anecdotal story about a scientist in the '60s who was going to implement a model of object recognition over the summer. Needless to say, he didn't succeed.

Fourth, making a computational model forces you to confront aspects of the problem that you might have otherwise ignored or considered to be irrelevant. Although one sometimes ends up using simplifications or stand-ins for these other aspects (see the list of problems that follows), it can be useful to at least confront these problems.

Models allow control. In a computational model you can control many more variables much more precisely than you can with a real system, and you can replicate results precisely. This enables you to explore the causal role of different components in ways that would otherwise be impossible.

Models provide a unified framework. As we discussed earlier, there are many advantages to using a single computational framework to explain a range of phenomena. In addition to providing a more stringent test of a theory, it encourages parsimony and

also enables one to relate two seemingly disparate phenomena by understanding them in light of a common set of basic principles.

Also, it is often difficult for people to detect inconsistency in a purely verbal theory — we have a hard time keeping track of everything. However, a computational model reveals inconsistencies quite readily, because everything has to hang together and actually *work*.

Problems:

Models are too simple. Models, by necessity, involve a number of simplifications in their implementation. These simplifications may not capture all of the relevant details of the biology, the environment, the task, and so on, calling into question the validity of the model.

Inevitably, this issue ends up being an empirical one that depends on how wrong the simplifying assumptions are and how much they influence the results. It is often possible for a model to make a perfectly valid point while using a simplified implementation because the missing details are simply not relevant — the real system will exhibit the same behavior for any reasonable range of detailed parameters. Furthermore, simplification can actually be an important benefit of a model — a simple explanation is easier to understand and can reveal important truths that might otherwise be obscured by details.

Models are too complex. On the flip side, other critics complain that models are too complex to understand why they behave the way they do, and so they contribute nothing to our understanding of human behavior. This criticism is particularly relevant if a modeler treats a computational model as a theory, and it points to the mere fact that the model reproduces a set of data as an explanation of this data.

However, this criticism is less relevant if the modeler instead identifies and articulates the critical principles that underly the model's behavior, and demonstrates the relative irrelevance of other factors. Thus, a model should be viewed as a concrete instantiation of broader principles, not as an end unto itself, and the way in which the model "uses" these principles to account for the data must be made clear. Unfortunately, this essential step of making the principles clear and demonstrating their generality is often not taken. This can be a difficult step for complex models (which is, after all, one of the advantages of modeling in the first place!), but one made increasingly manageable with advances in techniques for analyzing models.

Models can do anything. This criticism is inevitably leveled at successful models. Neural network models do have a very large number of parameters in the form of the adaptable weights between units. Also, there are many degrees of freedom in the architecture of the model, and in other parameters that determine the behavior of the units. Thus, it might seem that there are so many parameters available that fitting any given set of behavioral phenomena is uninteresting. Relatedly, because of the large number of parameters, sometimes multiple different models can provide a reasonable account of a given phenomenon. How can one address this *indeterminacy* problem to determine which is the "correct" model?

The general issues of adopting a principled, explanatory approach are relevant here — to the extent that the model's behavior can be understood in terms of more general principles, the success of the model can be attributed to these principles, and not just to random parameter fitting. Also, unlike many other kinds of models, many of the parameters in the network (i.e., the weights) are determined by principled learning mechanisms, and are thus not "free" for the modeler to set. In this book, most of the models use the same basic parameters for the network equations, and the cases where different parameters were used are strongly motivated.

The general answer to the *indeterminacy* problem is that as you apply a model to a wider range of data (e.g., different tasks, newly discovered biological constraints), and in greater detail on each task (e.g., detailed properties of the learning process), the models will be much more strenuously tested. It thus becomes much less likely that two different models

can fit all the data (unless they are actually isomorphic in some way).

Models are reductionistic. One common concern is that the mechanistic, reductionistic models can never tell us about the real essence of human cognition. Although this will probably remain a philosophical issue until very large-scale models can be constructed that actually demonstrate realistic, humanlike cognition (e.g., by passing the *Turing test*), we note that *reconstructionism* is a cornerstone of our approach. Reconstructionism complements reductionism by trying to reconstruct complex phenomena in terms of the reduced components.

Modeling lacks cumulative research. There seems to be a general perception that modeling is somehow less cumulative than other types of research. This perception may be due in part to the relative youth and expansive growth of modeling — there has been a lot of territory to cover, and a breadth-first search strategy has some obvious pragmatic benefits for researchers (e.g., "claiming territory"). As the field begins to mature, cumulative work is starting to appear (e.g., Plaut, McClelland, Seidenberg, & Patterson, 1996 built on earlier work by Seidenberg & McClelland, 1989, which in turn built on other models) and this book certainly represents a very cumulative and integrative approach.

The final chapter in the book will revisit some of these issues again with the benefit of what comes in between.

1.6 Motivating Cognitive Phenomena and Their Biological Bases

Several aspects of human cognition are particularly suggestive of the kinds of neural mechanisms described in this text. We briefly describe some of the most important of these aspects here to further motivate and highlight the connections between cognition and neurobiology. However, as you will discover, these aspects of cognition are perhaps not the most obvious to the average person. Our introspections into the nature of our own cognition tend to emphasize the "conscious" aspects (because this is by definition what we are aware of), which appear to be *serial* (one thought at a time) and *focused* on a subset of things occurring inside and outside the brain. This fact undoubtedly contributed to the popularity of the standard serial computer model for understanding human cognition, which we will use as a point of comparison for the discussion that follows.

We argue that these conscious aspects of human cognition are the proverbial "tip of the iceberg" floating above the waterline, while the great mass of cognition that makes all of this possible floats below, relatively inaccessible to our conscious introspection. In the terminology of Rumelhart et al. (1986c), neural networks focus on the *microstructure* of cognition. Attempts to understand cognition by only focusing on what's "above water" may be difficult, because all the underwater stuff is necessary to keep the tip above water in the first place — otherwise, the whole thing will just sink! To push this metaphor to its limits, the following are a few illuminating shafts of light down into this important underwater realm, and some ideas about how they keep the "tip" afloat. The aspects of cognition we will discuss are:

- Parallelism
- Gradedness
- Interactivity
- Competition
- Learning

Lest you get the impression that computational cognitive neuroscience is unable to say anything useful about conscious experience, or that we do not address this phenomenon in this book, we note that chapter 11 deals specifically with "higher-level cognition," which is closely associated with conscious experience. There we present a set of ideas and models that provide the bridge between the basic mechanisms and principles developed in the rest of the book, and the more sequential, discrete, and focused nature of conscious experience. We view these properties as arising partly due to specializations of particular brain areas (the prefrontal cortex and the hippocampus), and partly as a result of the *emergent phenomena* that arise from the basic properties of neural processing as employed in a coordinated

processing system. This chapter emphasizes that there is really a continuum between what we have been referring to as conscious and subconscious processing.

1.6.1 Parallelism

Everyone knows the old joke about not being able to walk and chew gum at the same time. This is a simple case of processing multiple things in parallel (doing more than one thing at the same time). In our everyday experience, there are lots of examples of a situation where this kind of parallel processing is evident: having a conversation while driving or doing anything else (cooking, eating, watching TV, etc.); hearing your name at a cocktail party while talking to someone else (the aptly named "cocktail party effect"); and speaking what you are reading (reading aloud), to name just a few.

What may come as a surprise to you is that each of the individual processes from the above examples is itself the product of a large number of processes working in parallel. At the lowest level of analysis, we know that the human brain contains something like 10 *billion* neurons, and that each one contributes its little bit to overall human cognition. Thus, biologically, cognition must emerge from the parallel operation of all these neurons. We refer to this as parallel *distributed* processing (PDP) — the processing for any given cognitive function is distributed in parallel across a large number of individual processing elements. This parallelism occurs at many different levels, from brain areas to small groups of neurons to neurons themselves.

For example, when you look at a visual scene, one part of your brain processes the visual information to identify *what* you are seeing, while another part identifies *where* things are. Although you are not aware that this information is being processed separately, people who have lesions in one of these brain areas but not the other can only do one of these things! Thus, the apparently seamless and effortless way in which we view the world is really a product of a bunch of specialized brain areas, operating "under the hood" in a tightly coordinated fashion. As this hood is being opened using modern neuroimaging techniques, the parallelism of the brain is becoming even more obvious, as multiple brain areas are inevitably activated in most cognitive tasks.

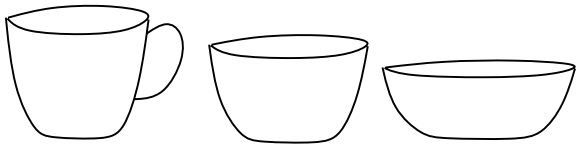

Figure 1.4: Example of graded nature of categorical representations: Is the middle item a cup or a bowl? It could be either, and lies in between these two categories.

Parallel processing can make it challenging to understand cognition, to figure out how all these subprocesses coordinate with each other to end up doing something sensible as a whole. In contrast, if cognition were just a bunch of discrete sequential steps, the task would be much easier: just identify the steps and their sequence! Instead, parallelism is more like the many-body problem in physics: understanding any pairwise interaction between two things can be simple, but once you have a number of these things all operating at the same time and mutually influencing each other, it becomes very difficult to figure out what is going on.

One virtue of the approach to cognition presented in this book is that it is based from the start on parallel distributed processing, providing powerful mathematical and intuitive tools for understanding how collective interactions between a large number of processing units (i.e., neurons) can lead to something useful (i.e., cognition).

1.6.2 Gradedness

In contrast with the discrete boolean logic and binary memory representations of standard computers, the brain is more **graded** and analog in nature. We will see in the next chapter that neurons integrate information from a large number of different input sources, producing essentially a *continuous, real valued* number that represents something like the relative *strength* of these inputs (compared to other inputs it could have received). The neuron then communicates another graded signal (its rate of firing, or *activation*) to other neurons as a function of this relative strength value. These graded signals can convey something like the *extent* or *degree* to which something is true. In the example in

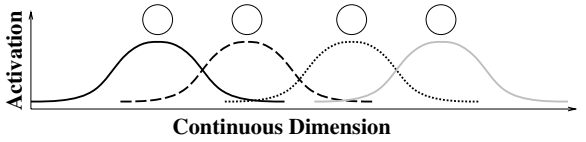

Figure 1.5: Graded activation values are important for representing continuous dimensions (e.g., position, angle, force, color) by coarse coding or basis-function representations as shown here. Each of the four units shown gives a graded activation signal roughly proportional to how close a point is along the continuous dimension to the unit's preferred point, which is defined as the point where it gives its maximal response.

figure 1.4, a neuron could convey that the first object pictured is almost definitely a cup, whereas the second one is maybe or sort-of a cup and the last one is not very likely to be a cup. Similarly, people tend to classify things (e.g., *cup* and *bowl*) in a graded manner according to how close the item is to a *prototypical* example from a category (Rosch, 1975).

Gradedness is critical for all kinds of perceptual and motor phenomena, which deal with continuous underlying values like position, angle, force, and color (wavelength). The brain tends to deal with these continua in much the same way as the continuum between a cup and a bowl. Different neurons represent different "prototypical" values along the continuum (in many cases, these are essentially arbitrarily placed points), and respond with graded signals reflecting how close the current exemplar is to their preferred value (see figure 1.5). This type of representation, also known as *coarse coding* or a *basis function* representation, can actually give a precise indication of a particular location along a continuum, by forming a *weighted estimate* based on the graded signal associated with each of the "prototypical" or basis values.

Another important aspect of gradedness has to do with the fact that each neuron in the brain receives inputs from many thousands of other neurons. Thus, each individual neuron is not critical to the functioning of any other — instead, neurons contribute as part of a graded overall signal that reflects the number of other neurons contributing (as well as the strength of their individual contributions). This fact gives rise to the phenomenon of *graceful degradation*, where function degrades "gracefully" with increasing amounts of damage to neural tissue. Simplistically, we can explain this by saying that removing more neurons reduces the strength of the signals, but does not eliminate performance entirely. In contrast, the CPU in a standard computer will tend to fail catastrophically when even one logic gate malfunctions.

A less obvious but equally important aspect of gradedness has to do with the way that processing happens in the brain. Phenomenologically, all of us are probably familiar with the process of trying to remember something that does not come to mind immediately — there is this fuzzy sloshing around and trying out of different ideas until you either hit upon the right thing or give up in frustration. Psychologists speak of this in terms of the "tip-of-the-tongue" phenomenon, as in, "its just at the tip of my tongue, but I can't quite spit it out!" Gradedness is critical here because it allows your brain to float a bunch of relatively weak ideas around and see which ones get stronger (i.e., *resonate* with each other and other things), and which ones get weaker and fade away. Intuition has a similar flavor — a bunch of relatively weak factors add up to support one idea over another, but there is no single clear, discrete reason behind it.

Computationally, these phenomena are all examples of **bootstrapping** and **multiple constraint satisfaction**. Bootstrapping is the ability of a system to "pull itself up by its bootstraps" by taking some weak, incomplete information and eventually producing a solid result. Multiple constraint satisfaction refers to the ability of parallel, graded systems to find good solutions to problems that involve a number of constraints. The basic idea is that each factor or constraint pushes on the solution in rough proportion to its (graded) strength or importance. The resulting solution thus represents some kind of compromise that capitalizes on the convergence of constraints that all push in roughly the same direction, while minimizing the number of constraints that remain unsatisfied. If this sounds too vague and fuzzy to you, don't worry — we will write equations that express how it all works, and run simulations showing it in action.

1.6.3 Interactivity

Another way in which the brain differs from a standard serial computer is that processing doesn't just go in only one direction at a time. Thus, not only are lots of things happening at the same time (parallelism), but they are also going both forward and backward too. This is known as *interactivity*, or *recurrence*, or *bidirectional connectivity*. Think of the brain as having hierarchically organized processing areas, so that visual stimuli, for example, are first processed in a very simple, low-level way (e.g., in terms of the little oriented lines present in the image), and then in subsequent stages more sophisticated features are represented (combinations of lines, parts, objects, configurations of objects, etc.). This is at least approximately correct. In such a system, interactivity amounts to simultaneous *bottom-up* and *top-down* processing, where information flows from the simple to the more complex, and also from the more complex down to the simple. When combined with parallelism and gradedness, interactivity leads to a satisfying solution to a number of otherwise perplexing phenomena.

For example, it was well documented by the 1970s that people are faster and more accurate at identifying letters in the context of words than in the context of random letters (the *word superiority effect*). This finding was perplexing from the unidirectional serial computer perspective: Letters must be identified before words can be read, so how could the context of a word help in the identification of a letter? However, the finding seems natural within an interactive processing perspective: Information from the higher word level can come back down and affect processing at the lower letter level. Gradedness is critical here too, because it allows weak, first-guess estimates at the letter level to go up and activate a first-guess at the word level, which then comes back down and resonates with the first-guess letter estimates to home in on the overall representation of the word and its letters. This explanation of the word superiority effect was proposed by McClelland and Rumelhart (1981). Thus, interactivity is important for the bootstrapping and multiple constraint satisfaction processes described earlier, because it allows constraints from all levels of processing to be used to bootstrap and converge on a good overall solution.

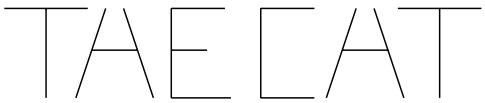

Figure 1.6: Ambiguous letters can be disambiguated in the context of words (Selfridge, 1955), demonstrating interactivity between word-level processing and letter-level processing.

There are numerous other examples of interactivity in the psychological literature, many of which involve stimuli that are ambiguous in isolation, but not in context. A classic example is shown in figure 1.6, where the words constrain an ambiguous stimulus to look more like an H in one case and an A in the other.

1.6.4 Competition

The saying, "A little healthy competition can be a good thing," is as true for the brain as it is for other domains like economics and evolution. In the brain, competition between neurons leads to the *selection* of certain representations to become more strongly active, while others are weakened or suppressed (e.g., in the context of bootstrapping as described above). In analogy with the evolutionary process, the "survival of the fittest" idea is an important force in shaping both learning and processing to encourage neurons to be better adapted to particular situations, tasks, environments, and so on. Although some have argued that this kind of competition provides a sufficient basis for learning in the brain (Edelman, 1987), we find that it is just one of a number of important mechanisms. Biologically, there are extensive circuits of *inhibitory interneurons* that provide the mechanism for competition in the areas of the brain most central to cognition.

Cognitively, competition is evident in the phenomenon of *attention*, which has been most closely associated with perceptual processing, but is clearly evident in all aspects of cognition. The phenomenon of *covert* spatial attention, as demonstrated by the Posner task (Posner, 1980) is a good example. Here, one's attention is drawn to a particular region of visual space by a *cue* (e.g., a little blinking bar on a computer screen), and then another stimulus (the *target*) is presented shortly thereafter. The target appears either near

the cue or in the opposite region of space, and the subject must respond (e.g., by pressing a key on the computer) whenever they detect the onset of the target stimulus. The target is detected significantly faster in the cued location, and significantly slower in the noncued location, relative to a baseline of target detection without any cues at all. Thus, the processing of the cue competes with target detection when they are in different locations, and facilitates it when they are in the same location. All of this happens faster than one can move one's eyes, so there must be some kind of internal ("covert") attention being deployed as a result of processing the cue stimulus. We will see in section 8.5 that these results, and several other related ones, can be accounted for by a simple model that has competition between neurons (as mediated by the inhibitory interneurons).

1.6.5 Learning

The well-worn nature versus nurture debate on the development of human intelligence is inevitably decided in terms of both. Thus, both the genetic configuration of the brain and the results of learning make important contributions. However, this fact does nothing to advance our understanding of exactly *how* genetic configuration and learning interact to produce adult human cognition. Attaining this understanding is a major goal of computational cognitive neuroscience, which is in the unique position of being able to simulate the kinds of complex and subtle interdependencies that can exist between certain properties of the brain and the learning process.

In addition to the developmental learning process, learning occurs constantly in adult cognition. Thus, if it were possible to identify a relatively simple learning mechanism that could, with an appropriately instantiated initial architecture, organize the billions of neurons in the human brain to produce the whole range of cognitive functions we exhibit, this would obviously be the "holy grail" of cognitive neuroscience. For this reason, this text is dominated by a concern for the properties of such a learning mechanism, the biological and cognitive environment in which it operates, and the results it might produce. Of course, this focus does not diminish the importance of the genetic basis of cognition. Indeed, we feel that it is perhaps only in the context of such a learning mechanism that genetic parameters can be fully understood, much as the role of DNA itself in shaping the phenotype must be understood in the context of the emergent developmental process.

A consideration of what it takes to learn reveals an important dependence on gradedness and other aspects of the biological mechanisms discussed above. The problem of learning can be considered as the problem of *change*. When you learn, you change the way that information is processed by the system. Thus, it is much easier to learn if the system responds to these changes in a graded, proportional manner, instead of radically altering the way it behaves. These graded changes allow the system to try out various new ideas (ways of processing things), and get some kind of graded, proportional indication of how these changes affect processing. By exploring lots of little changes, the system can evaluate and strengthen those that improve performance, while abandoning those that do not. Thus, learning is very much like the *bootstrapping* phenomenon described with respect to processing earlier: both depend on using a number of weak, graded signals as "feelers" for exploring possibly useful directions to proceed further, and then building on those that look promising.

None of this kind of bootstrapping is possible in a discrete system like a standard serial computer, which often responds catastrophically to even small changes. Another way of putting this is that a computer program typically only works if everything is right — a program that is missing just one step typically provides little indication of how well it would perform if it were complete. The same thing is true of a system of logical relationships, which typically unravels into nonsense if even just one logical assertion is incorrect. Thus, discrete systems are typically too *brittle* to provide an effective substrate for learning.

However, although we present a view of learning that is dominated by this bootstrapping of small changes idea, other kinds of learning are more discrete in nature. One of these is a "trial and error" kind of learning that is more familiar to our conscious experience. Here, there is a discrete "hypothesis" that governs behavior during a "trial," the outcome of which ("error") is used

to update the hypothesis next time around. Although this has a more discrete flavor, we find that it can best be implemented using the same kinds of graded neural mechanisms as the other kinds of learning (more on this in chapter 11). Another more discrete kind of learning is associated with the "memorization" of particular discrete facts or events. It appears that the brain has a specialized area that is particularly good at this kind of learning (called the *hippocampus*), which has properties that give its learning a more discrete character. We will discuss this type of learning further in chapter 9.

1.7 Organization of the Book

This book is based on a relatively small and coherent set of mechanistic principles, which are introduced in part I of the text, and then applied in part II to understanding a range of different cognitive phenomena. These principles are implemented in the Leabra algorithm for the exploration simulations. These explorations are woven throughout the chapters where the issues they address are discussed, and form an integral part of the text. To allow readers to get as much as possible out of the book without doing the simulations, we have included many figures and have carefully separated the procedural aspects from the content using special typesetting.

Because this book emphasizes the linkages and interactions between biology, computational principles, and a wide variety of human cognitive phenomena, we cannot provide exhaustive detail on all potentially relevant aspects of neuroscience, computation, or cognition. We do attempt to provide references for deeper exploration, however. Relatedly, all of the existing supporting arguments and details are not presented for each idea in this book, because in many cases the student would likely find this tedious and relatively uninformative. Thus, we expect that expert neuroscientists, computational/mathematical researchers, and cognitive psychologists may find this book insufficiently detailed in their area of expertise. Nevertheless, we provide a framework that spans these areas and is consistent with well-established facts in each domain.

Thus, the book should provide a useful means for experts in these various domains to bridge their knowledge into the other domains. Areas of current debate in which we are forced to make a choice are presented as such, and relevant arguments and data are presented. We strive above all to paint a coherent and clear picture at a pace that moves along rapidly enough to maintain the interest (and fit within the working memory span) of the reader. As the frames of a movie must follow in rapid enough succession to enable the viewer to perceive motion, the ideas in this book must proceed cleanly and rapidly from neurobiology to cognition for the coherence of the overall picture to emerge, instead of leaving the reader swimming in a sea of unrelated facts.

Among the many tradeoffs we must make in accomplishing our goals, one is that we cannot cover much of the large space of existing neural network algorithms. Fortunately, numerous other texts cover a range of computational algorithms, and we provide references for the interested reader to pursue. Many such algorithms are variants on ideas covered here, but others represent distinct frameworks that may potentially provide important principles for cognition and/or neurobiology. As we said before, it would be a mistake to conclude that the principles we focus on are in any way considered final and immutable — they are inevitably just a rough draft that covers the domain to some level of satisfaction at the present time.

As the historical context (section 1.3) and overview of our approach (section 1.4) sections made clear, the Leabra algorithm used in this book incorporates many of the important ideas that have shaped the history of neural network algorithm development. Throughout the book, these principles are introduced in as simple and clear a manner as possible, making explicit the historical development of the ideas. When we implement and explore these ideas through simulations, the Leabra implementation is used for coherence and consistency. Thus, readers acquire a knowledge of many of the standard algorithms from a unified and integrated perspective, which helps to understand their relationship to one another. Meanwhile, readers avoid the difficulties of learning to work with the various implementations of all these different algorithms, in favor of investing effort into fully understanding one integrated algorithm at a practical hands-on level. Only algebra and simple calculus concepts, which are reviewed where necessary,

are required to understand the algorithm, so it should be accessible to a wide audience.

As appropriate for our focus on cognition (we consider perception to be a form of cognition), we emphasize processing that takes place in the human or mammalian **neocortex**, which is typically referred to simply as the **cortex**. This large, thin, wrinkled sheet of neurons comprising the outermost part of the brain plays a disproportionally important role in cognition. It also has the interesting property of being relatively homogeneous from area to area, with the same basic types of neurons present in the same basic types of connectivity patterns. This is principally what allows us to use a single type of algorithm to explain such a wide range of cognitive phenomena.

Interactive, graphical computer simulations are used throughout to illustrate the relevant principles and how they interact to produce important features of human cognition. Detailed, step-by-step instructions for exploring these simulations are provided, together with a set of exercises for the student that can be used for evaluation purposes (an answer key is available from the publisher). Even if you are not required to provide a written answer to these questions, it is a good idea to look them over and consider what your answer might be, because they do raise important issues. Also, the reader is strongly encouraged to go beyond the step-by-step instructions to explore further aspects of the model's behavior.

In terms of the detailed organization, part I covers *Basic Neural Computational Mechanisms* across five chapters (*Individual Neurons*, *Networks of Neurons*, and three chapters on *Learning Mechanisms*), and part II covers *Large-Scale Brain Area Organization and Cognitive Phenomena* across five chapters (*Perception and Attention*, *Memory*, *Language*, and *Higher-Level Cognition*, with an introductory chapter on *Large-Scale Brain Area Functional Organization*). Each chapter begins with a detailed table of contents and an introductory overview of its contents, to let the reader know the scope of the material covered. When key words are defined or first used extensively, they are highlighted in **bold** font for easy searching, and can always be found in the index. `Simulation terms` are in the font as shown.

↪ Procedural steps to be taken in the explorations are formatted like this, so it is easy to see exactly what you have to do, and allows readers who are not running the model to skip over them.

Summaries of the chapters appear at the end of each one (this chapter excluded), which encapsulate and interrelate the contents of what was just read. After that, a list of references for further reading is provided. We hope you enjoy your explorations!

1.8 Further Reading

The original PDP (parallel-distributed processing) volumes, though somewhat dated, remain remarkably relevant: Rumelhart, McClelland, and PDP Research Group (1986c), McClelland, Rumelhart, and PDP Research Group (1986).

An excellent collection of the important early papers in neural networks can be found in Anderson and Rosenfeld (1988).

For other views on the basic premises of cognitive neuroscience and levels of analysis, we suggest: Marr (1982), chapter 1; Sejnowski and Churchland (1989); Shallice (1988), chapter 2; Posner, Inhoff, Friedrich, and Cohen (1987); Farah (1994); Kosslyn (1994).

For a developmentally-focused treatment of computational neural network modeling, see: Elman et al. (1996) and Plunkett and Elman (1997).

For other treatments of computational modeling using artificial neural networks, see: Hertz, Krogh, and Palmer (1991), Ballard (1997), Anderson (1995), McCleod, Plunkett, and Rolls (1998), and Bishop (1995).

For an encyclopedic collection of computational neural network models and more general brain-level theories, see Arbib (1995).