BRAIN MECHANISMS

Linking Cognitive Phenomena to Neuron Activity

L. ANDREW COWARD

CONTENTS

CHAPTER 3

The different architectures of computers and brains

There are no direct resemblances between the brain and any current computer system. The physical resources of the brain that perform information processing are neurons, while the physical resources that perform information processing in computers are transistors. The types of information processes performed by neurons and transistors are very different. The organization of these physical resources to carry out the required features is completely different in the two types of systems. Nevertheless, the way we go about understanding computers has some important lessons for how we can understand the brain. As discussed in chapter 2, there are methods used to organize the information about computer systems with billions of components in such a way that human designers can understand those systems. These methods involve the creation of hierarchies of description with some very specific properties. Such hierarchies can be developed to organize our knowledge of brains with billions of neurons in such a way that we can understand how physiological processes result in human cognition.

In this chapter we will develop the framework for the highest-level descriptions of the brain. This is the level of system architecture. To introduce the approach, we will begin by discussing the architectures used to understand computers at the highest level of description.

Both computers and brains can be viewed as control systems. As described in detail in appendix 1, practical considerations, like the need to limit physical information processing resources, constrain the way in which the resources of any complex control system are organized. At the highest level, the organization of information processing resources in complex control systems is constrained into one of just two general architectural forms. One of these forms is ubiquitous in computers; the other form is ubiquitous in brains.

What is an architecture?

An architecture breaks up a system into subsystems, specifies what each subsystem does, and describes how the subsystems interact in order to carry out system tasks. This organization into subsystems helps in understanding the system. Subsystems are sometimes called components or modules. There are different ways in which the same system could be organized into subsystems, and there are two ways in particular that are important for understanding computers.

Computer architectures

Two key ways in which computer systems are organized into subsystems are the functional architecture and the physical or system architecture. They are illustrated in figure 3.1.

THE FUNCTIONAL ARCHITECTURE OF A COMPUTER

A user interacts with their computer via applications. These applications can be thought of as subsystems. Application subsystems include different web browsers; Safari and Firefox are two examples. Other applications process text, images, sounds, or videos. Yet other applications support playing different games like Chess or Go. Figure 3.1-I shows the appearance of a computer screen when we display icons for a number of these applications. Each application is a group of behaviours that are similar or related in some way from the point of view of a user. All the behaviours in a group act on similar types of information and perform similar processes on that information.

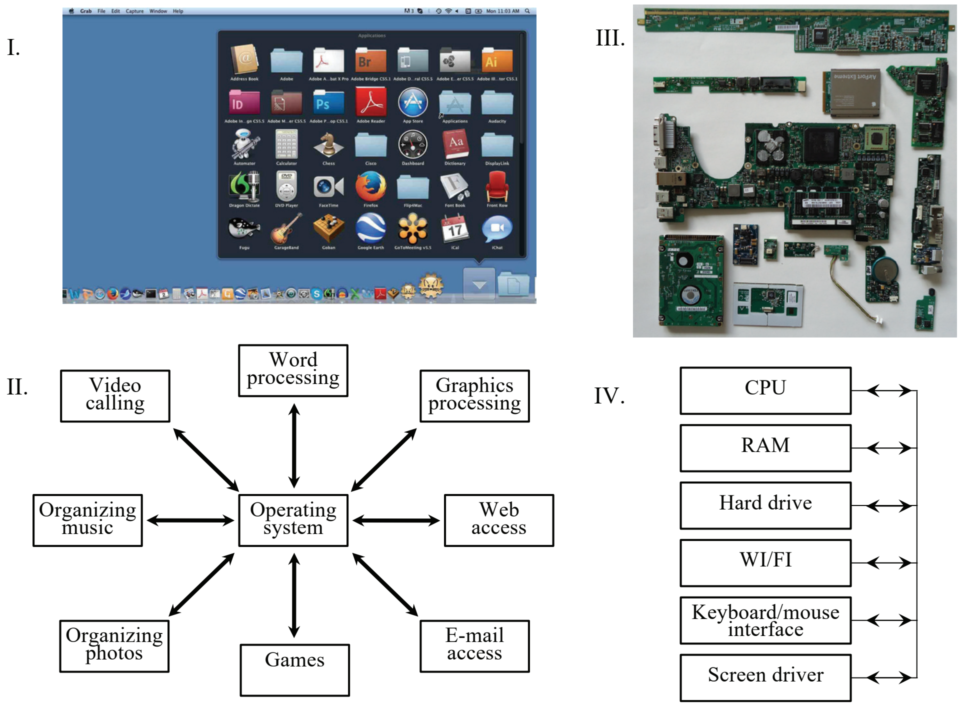

FIGURE 3.1. The high-level functional and system architectures of a personal computer. The functional architecture describes the applications and features of the computer and the interactions between them. The applications can only access computer resources and exchange information with each other via the operating system. The system architecture describes the physical units of the computer, such as the central processing unit (CPU), memory, etc. and their interactions. All these physical units communicate with each other over a common pathway, called a bus. I. Display of the application components. II. Functional architecture connecting the application components. III. Separate physical system modules. IV. System architecture that connects the physical system modules.

It is important to note that the information being processed within one application is readily available elsewhere in that application but a bit less available to any other application. If I am working in an email application, I can immediately open past emails and copy their text to paste it into a new email. However, to get at text in some other application like Word, I need to open the Word application. If I want to move that text across to an email, I must copy it in the Word application, move to the email application, and paste it into the appropriate location. Some information, such as the formatting of the text, may be lost in this process. In practice this is not particularly burdensome, but what is happening is that in order to exchange information between two applications, the first application must hand the information over to something called the operating system of the computer, and the second application can then get it from the operating system.

This is not the only role the operating system plays with respect to the applications. The operating system controls all the resources of the computer, like memory, access to the screen, access to the Wi-Fi, or access to the CPU. For an application to perform a task, it must request these resources from the operating system.

Figure 3.1-II captures these relationships in what is often called the functional architecture. The subsystems in this architecture are the applications and the operating system. The relationships between these subsystems are illustrated by the two-way arrows between the operating system and each application. There are no direct arrows between the applications because applications cannot directly exchange information; such information can only go via the operating system.

At a more detailed level of the functional architecture, the way in which all the features of each application work can be described. Hence another name for the functional architecture could be the user manual.

THE SYSTEM ARCHITECTURE OF A COMPUTER

If you open up a computer and look inside, you can see a very different type of subsystem. As illustrated in figure 3.1-III, there are a number of flat, roughly rectangular pieces of green plastic covered with various types of more detailed electronic devices like integrated circuits. The green plastic pieces are called printed circuit boards, and they connect together all their devices. The printed circuit board plus its devices is called a printed circuit assembly, and these printed circuit assemblies are the major physical subsystems of the computer. In an actual computer these printed circuit assemblies are connected together, but for the figure they have been separated and roughly arranged in rows.

What is the role of these printed circuit assembly subsystems? Each subsystem performs a different group of information processes. One subsystem performs the information storage and retrieval processes of random-access memory (RAM). Another subsystem carries out the transmission and receiving of information over wireless (Wi-Fi). Another subsystem called the central processing unit (CPU) performs all the logic and arithmetic processes. Yet another subsystem performs the processes associated with controlling what appears on the computer screen. We will call these physical subsystems *modules*, where a module is a physical structure that performs a group of similar information processes. All of the information processes performed by one module are sufficiently similar that they can be performed with the same hardware.

Some of these modules and their relationships with each other are illustrated in figure 3.1-IV. This figure is the physical or system architecture of the computer. It shows how all the different modules perform different groups of information processes but all communicate with each other over a shared common channel (called a bus). The vast majority of computer systems have this architecture, with separate memory, CPU, and various input/output modules connected by a common bus. Early on in the computer age other architectures were used, such as analogue computers. Each such analogue computer was designed to carry out a very limited range of similar user tasks. Today, analogue computers may be (very rarely) used for extremely specific tasks, but if a computer needs to perform a range of different applications, it invariably has this memory, processing, input, and output architecture.

Information exchange within the computer system architecture

What distinguishes one module from another in the system architecture? Each module very efficiently carries out a different group of information processes, and a key factor is that each module must be able to carry out its information processes relatively independently of the information processes being performed by other modules. If a module required large amounts of information generated by other modules in order to carry out its processes, it would be constantly waiting for the other modules to provide that information. To achieve efficiency, the information exchanged between modules in the course of carrying out applications must be relatively small.

What does “relatively” mean? It means that most of the information used by a module must be generated within that module, and much less of this information must come from other modules. In other words, the information exchange between two printed circuit assembly modules over the common bus must be much less than the information exchange within one printed circuit assembly. This is also true at a more detailed level: much less information exchange must occur between two integrated circuits on the same printed circuit assembly than within an integrated circuit.

Organization of resources into a modular hierarchy

The physical resources of a computing system are organized into a modular hierarchy on the basis of information process similarity. Some of the different levels of modules in the hierarchy are printed circuit assemblies, integrated circuits, cells within integrated circuits, logic gates within cells, and transistors. A printed circuit assembly performs a group of similar information processes. An integrated circuit performs a group of even more similar information processes that contribute to the processes performed by its printed circuit assembly. A cell within an integrated circuit is made up of a group of logic gates and other components that together perform a group of very similar information processes. These processes contribute to the processes performed by the integrated circuit. A logic gate within a cell performs a group of even more similar processes that contribute to the processes performed by the cell. A transistor within a logic gate performs a group of extremely similar processes that contribute to the processes performed by the logic gate.

RELATIONSHIP BETWEEN THE COMPUTER FUNCTIONAL AND SYSTEM ARCHITECTURES

The functional and physical architectures look so different that an obvious question is, what is the relationship between them? One question which might be asked is, do any physical modules in the system architecture correspond with applications or features in the functional architecture? The answer is that it is very unlikely that there will be any correspondences of this type.

The reason is that a key driving force on the system architecture is the need to conserve physical information processing resources. The features of any application are carried out by long sequences and combinations of different information processes. Even between two completely different applications, some of the individual information processes are similar. Almost any application will require processes to retrieve stored information. If two information processes are similar, they can be performed on the same physical resources. When resources are shared in this way, the total resources required by the system are reduced.

Hence the physical architecture is created by identifying different groups of information processes that are similar but used by many different applications. Modules are designed that are customized to perform each group very efficiently. To perform all of its features, any one application will generally need information processes of the types performed by most, if not all, of the physical modules. For example, an email application will need CPU processes, memory processes, mouse/keyboard interface processes, Wi-Fi access processes, screen control processes, etc.

Any one physical module will therefore need to support most, if not all, applications. It is of course possible that some application, perhaps a video game, might need a specialized type of information process, perhaps supporting very high-speed responses. A separate physical module to support just that process type would then be used only for the one application. However, the game application would still use most of the other physical modules, and in practice it is highly likely that a few other applications, perhaps other games, will benefit from using the new process type.

Hence in general, any module in the physical architecture will support multiple applications, and any application in the functional architecture will require the support of many physical modules. There will be no simple correspondences between physical modules and component applications.

General types of information process within a computer system

Another important property of the computer physical architecture is that all of its information processes can be classified into one of two general types. One type is the instruction or command. The CPU performs long sequences of instructions. The other type is data reading and writing, often performed by the memory. These two types of process are connected: generally an instruction is carried out if some data from the current environment corresponds with data stored in memory. One name for this architecture is the *instruction* architecture.

At a user level we can also view any interaction with our computer in terms of instructions and data. If I want to access a website, I first provide some *data* by typing the address of the site or by finding the address in some other web page. Then by typing "return" or by clicking on the appropriate address I effectively issue an *instruction* to go to that site.

Significantly, if we look in more detail at how the computer responds, each instruction or data operation at this user level is carried out by combinations and sequences of instruction and data operations at more detailed levels.

At an intermediate level, a high-level instruction is implemented by combinations and sequences of software code that specify more detailed instructions and data reads and writes. At an even more detailed level, software code is implemented by combinations and sequences of machine code, which is also in the form of instructions and accessing and recording data. At a yet more detailed level we can view machine instructions as combinations and sequences of device-level code like "open the gate of that transistor" or "access the state of that transistor." In other words, instructions and data read/writes at the device level.

This ubiquitous presence of instructions and data read/writes is critical for understanding computers in the sense of being able to create or modify their design. The existence of these two types of information process make it possible to create the hierarchies of description required for understanding of computer systems design as discussed in the previous chapter.

Understanding computers

Both the functional and system architectures aid in understanding how a computer works but from very different points of view. The functional architecture helps us understand how to operate a computer, while the system architecture helps us understand how to design a computer. However, there is no simple correspondence between applications in the functional architecture and modules in the system architecture.

Hence although the functional architecture (or user manual) can be very useful for understanding how to operate a computer, it is of little help if the need is to understand the design.

Why do computers almost always have the instruction architecture?

We mentioned earlier that computers almost invariably have the instruction architecture. The reason for this is that there are some practical considerations that apply to any complex control system, and together these practical considerations tend to constrain computer physical architectures into the instruction architecture form.

We have already mentioned resource economy. Computers contain information processing resources like transistors. Although it is true that modern computer systems can contain billions of transistors, it is also true that with inefficient design this number would be vastly larger. There is a practical pressure to keep the number as small as possible.

A second consideration is the need for modifiability to easily make design changes to user features. All features are ultimately carried out by the system hardware. One problem here is that if a modification to a feature requires any changes to that hardware, the modification may introduce unanticipated and probably undesirable changes to other features because any hardware is used for many different features. In computer systems, software manages much of the access to hardware, and feature changes can often be achieved by software changes. However, although changes are easier to make in software, this does not avoid the side effect problem: a software change to implement a feature change risks introducing undesirable side effects into other features.

A third consideration is repairability. When a system fails, it is desirable to be able to identify the component that caused the failure on as detailed a level as possible. This failed component can then be replaced at reasonable cost, rather than having to replace the whole system when one component fails.

A fourth consideration is constructability. If the manufacturing process for the computer system is very complex, there is an increased risk of errors that will be expensive to correct.

The final consideration is in some ways a specific case of resource economy, but it is so critical that it is worth identifying separately. This consideration is the need for synchronicity. Let me illustrate the synchronicity problem with an example. Suppose that you are downloading two different websites from the internet at the same time. The information to display each website is broken down into small "packets" that are transmitted separately to your computer, quite likely over different physical routes. At your computer, each packet must be processed to extract its information. This processing is similar between packets and will therefore use much the same information processing resources. The synchronicity problem is to make sure that information coming from the packets describing one website is kept separate from information about the other website, even though all the information is being processed on the same resources.

The five considerations are thus resource economy, modifiability, repairability, constructability and synchronicity. These five considerations tend to constrain computer physical architectures into the instruction architecture form. We will touch on the reasons for this constraint at a number of points in the book, and a complete but more technical description of the reasons is provided in appendix 1. This constraint is a tendency, not an absolute requirement. If there were just one or a few similar user features and no limits on available resources, many architectural forms would be possible. But as the number of features that need to be performed using a given set of resources increases, the physical architecture will be more and more tightly constrained into the instruction architecture form. The ubiquity of this instruction architecture form indicates that in practice the constraints are almost invariably strong.

Brain architectures

In brains, we can also identify two architectures that are analogous with the functional architecture and the physical architecture of computers. The two types are illustrated in figure 3.2.

THE FUNCTIONAL ARCHITECTURE OF THE BRAIN

When we observe human cognition, we find that behaviours can be classified into groups. Behaviours in one group perform similar tasks with similar types of information. The behaviours of remembering many different past events in our own lives, including how we felt during those events, form one group, which we call episodic memory. Another group includes recalling many different facts about the world, such as the meanings of words. We call this group semantic memory. These different groups are analogous with the different application components in a computer. From the point of view of a user, each application performs similar processes on different sets of information.

On this basis of groups of similar tasks acting on different information we can organize human cognition into a functional architecture as illustrated in figure 3.2-I. A key part of this organization is the different types of memory. A number of memory types have been identified by cognitive science. We identified two of these in the previous paragraph: *semantic* memory and *episodic* memory. A third type is short-term storage of information for immediate processing. The information potentially available from semantic or episodic memory is not all available for immediate use. Small parts of this information can be retrieved, with those parts remaining active for a short time. During this time, they can be used, for example, to drive speech that describes the retrieved facts or memories. A little later the information is no longer active, and if required again it must be retrieved again. This ability to keep relatively small pieces of information actively available for a short time is called *working* memory.

A fourth type of memory is *procedural*. This is the memory for how to perform skilled (often physical) behaviours, such as riding a bicycle. One notable difference between procedural memory and the semantic and episodic types is that it is hard to describe verbally exactly how we perform the skill, but a major way in which we make use of memories for facts and events is verbal description.

The fifth type of memory is less familiar. If we are very briefly exposed to some visual or verbal information, such an exposure can influence our behaviour a few minutes later, even though we were not consciously aware of the information at the time. This type of memory, called *priming*, achieved some notoriety a number of years ago from the idea that flashing the message "Drink Coca-Cola" on a movie screen, too briefly for the audience to be aware of it, could nevertheless result in an increase in Coke sales a little later. A less artificial example would be that if we catch several brief partial glimpses of an animal hidden in the grass, even though none of the individual glimpses were sufficient to identify the animal or even to consciously recognize that something is there, we can carry information across between glimpses and combine it into a recognition.

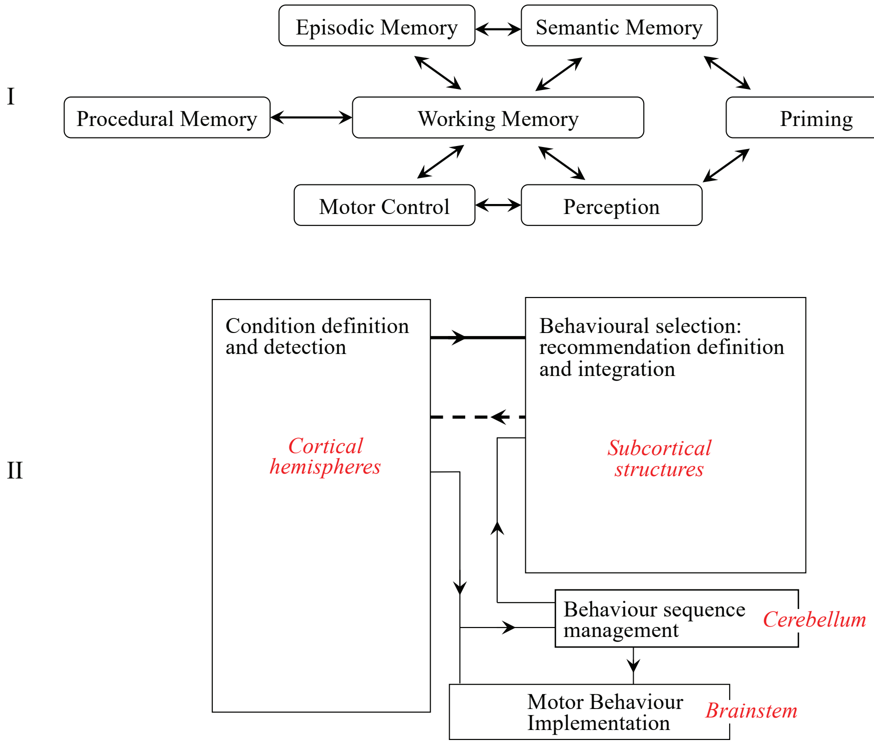

FIGURE 3.2. The high-level functional and system architectures of the brain. I. The functional architecture shows the cognitive components of the brain and the interactions between them. II. The system architecture shows the anatomical modules of the brain, such as the condition definition and detection module (the cortex), the behavioural recommendation definition and integration module (various subcortical structures), the behavioural sequence management module (the cerebellum), and the major interaction pathways.

Another key functional component at this high level is *perception*, the processing of sensory inputs to detect the presence of cognitively relevant circumstances. We need to be able to look at a visual scene and identify different objects and groups of objects that may require a response. The other functional component in figure 3.2-I is *motor control*, which generates appropriate body movements.

There are of course many different complex human behaviours that combine the use of these functional components. Understanding speech and production of speech are two important examples. Another type of behaviour is the ability to recognize faces, and to remember the names associated with the faces. Other types include the abilities to organize complex bodies of knowledge or to play complex games such as Chess or Go.

The functional architecture illustrated in figure 3.2-I is a convenient and useful way to think about different human capabilities. But as in the case of a computer, if you open up a brain, you get a very different picture. The brain is made up of lots of cells called neurons, lots of connections between neurons called axons, and a large number of smaller glia cells that mostly provide support functions to the neurons and axons. An average neuron has of the order of ten thousand input connections[1] from other neurons that are the sources of its inputs. A neuron also has just one axon, and on average an axon makes ten thousand connections onto other target neurons. A neuron integrates any inputs from its source neurons to determine if it will produce its own outputs and, via its axon, communicates any such outputs to all its target neurons. These outputs are in the form of electrical spikes that propagate along the axon to all the targets. Some neurons, called excitatory, encourage their target neurons to produce outputs. Other neurons, called inhibitory, discourage outputs by their targets. Yet other neurons modulate their target neurons in various ways.

The neurons in the brain are clustered together in different ways to form a number of interconnected anatomical structures. Some of the most obvious structures are the cortex, some subcortical structures, and the cerebellum. All these structures have substructures.

PRACTICAL CONSTRAINTS ON THE BRAIN

Earlier in the chapter we talked about how the needs for resource economy, modifiability, repairability, constructability, and synchronicity tend to constrain complex computer systems into the instruction architecture form. Analogous practical needs also exist for brains.

If the brain of one species can learn to perform a set of behaviours using fewer neurons than the brain of another species, the brain that requires less food to support fewer neurons will have a natural selection advantage. A brain that can learn without damaging its ability to perform behaviours learned in the past will have a natural selection advantage over a brain in which new learning interferes with earlier learning. A brain that can recover from

---

[1] Although the average number of input connections is about ten thousand, for individual neurons the number can vary from two or three up to over a hundred thousand.

physical damage to some of its components will also have a natural selection advantage. A brain that develops under DNA guidance by a relatively simple process will have a natural selection advantage over a brain for which the development process is more complex and therefore more error-prone. A brain must be able to look at several different objects in quick succession but will need time to fully process the information from any one object. A brain that can simultaneously process information derived from a sequence of several objects, sharing the processing resources but not confusing information derived from different objects, will also have a natural selection advantage.

The architecture of the brain is therefore subject to the same resource limitation, modifiability, repairability, constructability, and synchronicity pressures as computers. Furthermore, there is an additional pressure on brain architecture. Evolution proceeds by mutations to DNA information. Some mutations are harmful, but sometimes a mutation has a natural selection advantage. If all possible beneficial mutations also had fatal side effects, evolution would not be possible, and a species could not adapt to environmental changes. There is therefore a practical pressure in favour of evolvability—in other words, brains in which mutations are not necessarily fatal.

These practical considerations have placed strong constraints on the physical architecture of the brain. However, brains must learn most of their behaviours from experience, while the behaviours of computers are specified under external intellectual control by a designer. The architectures that result from the analogous practical considerations are therefore qualitatively different between computers and brains. For computer systems the architecture is the well-known instruction architecture. For brains the practical needs result in the physical architecture being constrained into a form called the recommendation architecture.

Although the two architectures are qualitatively different, there are some important parallels. In computer systems there are two general types of information process, data read/write and instruction, and two major subsystems that specialize in the two types, memory and processor, respectively. In the recommendation architecture there are also two general types of information process, but in this case they are called condition definition/detection and behavioural recommendation definition/integration. There are also two major subsystems specializing in the two types of process. In the brain, these two subsystems are the cortex and the subcortical structures.

We will touch on some of the reasons the brain is constrained into this recommendation architecture form at different points in this book, but a full description of these reasons is contained in appendix 1.

THE PHYSICAL ARCHITECTURE OF THE BRAIN

If the skull was completely removed so that the outer part of the brain could be seen, it might look a bit like figure 3.3. Three major anatomical modules of the brain are visible: the cerebrum, the brainstem, and the cerebellum. The cerebrum is a roughly spherical structure in the upper front of the brain. Protruding out of the bottom of the cerebrum is a bundle of axons and various groups of neurons called the brainstem. Lower down, the brainstem becomes the spinal cord that carries communications between brain and body. In the lower back of the brain, connected to the brainstem, is another roughly spherical structure called the cerebellum.

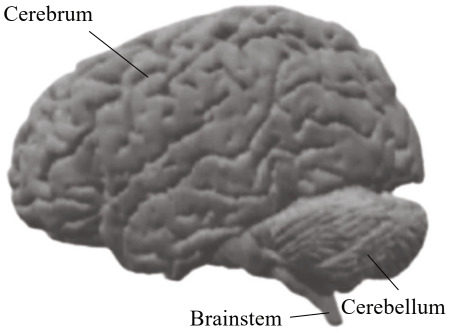

FIGURE 3.3. Appearance of the outside of the brain. Three major modules are visible: the cerebrum, the cerebellum, and the brainstem.

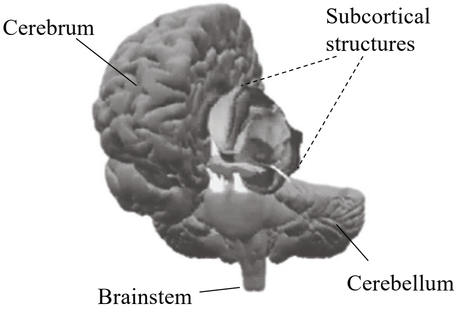

FIGURE 3.4. Appearance of the brain with one cerebral hemisphere removed. A fourth major module is visible: the subcortical structures.

Most structures in the brain come with two copies, a left copy and a right copy. The cerebrum and the cerebellum are both made up of two halves, or hemispheres. The two halves of the cerebrum are called the cerebral hemispheres, and the two halves of the cerebellum the cerebellar hemispheres.

Wrapped around the outside of the cerebrum, including much of the area where the two hemispheres face each other, is a relatively thin sheet of neurons called the cortex. Similarly, wrapped around the outside of the cerebellum is a relatively thin sheet of neurons called the cerebellar cortex. In both cases the neurons are organized into different layers going through the sheet. Inside of both the cerebral and cerebellar hemispheres is a large volume of what is called "white matter." This white matter is made up of a very large number of axons connecting between cortical neurons.

If one cerebral hemisphere were removed, as illustrated conceptually in figure 3.4, a group of subcortical structures called nuclei that are buried between the hemispheres would be visible. Nuclei are made up of clumps of neurons separated from other clumps by regions containing rather fewer neurons. These subcortical structures make up the fourth major brain module. The organization of these four modules in the system architecture of the brain is illustrated in figure 3.2-II, including the major connectivity paths between them. Each of these four major anatomical modules performs a different group of information processes.

The cortex detects specific circumstances, or conditions, within information derived from the senses and from the internal state of the brain itself. These conditions are defined by the cortex from experience, and condition definitions are constantly evolving. If any defined condition is present, its detection is signalled.

Some of the currently detected conditions are communicated to the subcortical structures, where each such detection is interpreted as a set of recommendations in favour of a range of different behaviours, each recommendation having its own weight. The subcortical structures determine and implement the behaviour with the largest total weight across all currently detected conditions. The subcortical structures also modify the recommendation weights on the basis of whether a selected behaviour had favourable or unfavourable results.

We will discuss the nature of conditions in a lot more detail in later chapters. However, at this point it is important to note that although conditions in the cortex are ultimately combinations of sensory information, in cognitive terms they are ambiguous. In other words, they do not correspond precisely with cognitive categories or specific behaviours.

Consider some cortical condition, defined by a combination of visual information, that is detected on one occasion when a cat is seen. In general, such a condition will not be detected every time a cat is seen and will sometimes be detected when other types of object (such as dogs or racoons) are seen. For example, input derived from looking at the head of an animal might lead to detection of a condition roughly corresponding with an approximate sphere with two pointy protrusions (ears). Such a condition would often be present when looking at a cat or racoon but less often for dogs, which sometimes have floppy ears. Another condition might roughly correspond with an approximate sphere with one blunt protrusion (muzzle). This condition would often be present in a dog or racoon but less often for a cat. Combinations of many such partially ambiguous conditions usually make it possible to unambiguously identify the animal.

Because there are no conditions that are always detected when an example of one category is seen and never detected in response to examples of other categories, discrimination between different categories is on the basis of detecting *groups* of conditions. Conditions are defined in such a way that the groups of conditions detected in response to an object of one category are sufficiently similar to the groups detected in response to other objects of the same type, and sufficiently different from the groups detected in response to objects of any other category, that it is possible to assign recommendation strengths in such a way that high integrity category identification is almost always achieved.

There are two major types of implemented behaviours. For one type, an implementation is the release of information into and/or out of some regions in the cortex. Such a release could be a subset of current sensory inputs into the cortex for processing. This subset might correspond with one object in the environment, and the release behaviour could be labelled *paying attention to that object*. Alternatively, a release could be between two cortical regions as a step in cognitive processing. Finally, a release could be out of the cortex to the brainstem and spinal cord to drive a selected motor behaviour.

The other type of implementation is a reward behaviour that changes recently used recommendation weights. Reward behaviours are also recommended by cortical condition detections, with the recommendation weights recorded in the subcortical structures. Reward behaviours modify the recommendation weights that determined recently selected behaviours, changing the probability that such behaviours will be selected again in the future in similar circumstances. Release or reward type behaviours will only be implemented if currently detected cortical conditions have sufficient total recommendation strength in the subcortical structures.

In general, learning a new behaviour will require changes to both condition definitions and recommendation weights. These changes take place in the cortex and the subcortical structures, respectively. However, if a sequence of behaviours is often appropriate in some circumstances, and in such circumstances the sequence always needs to be performed in the exactly same order, then once the sequence has been learned it will need very little new learning in the future. Examples of behaviours requiring such sequences include walking, climbing stairs, bicycling, playing a musical instrument, and producing speech. In each case there are portfolios of different muscle movement sequences that are needed on many different occasions. For climbing stairs, there is a sequence of leg, ankle, and foot muscle movements that are required to lift one foot by one stair. A different sequence is needed to lift the other foot to the next stair, then the first sequence is needed again, and so on. For speech, there are many frequently used sequences of mouth, tongue, lips, and throat muscles needed to generate often-used words and phrases.

The role of the cerebellum is to take over the implementation of such sequences, carrying them out accurately and rapidly. In the cerebellum, the speed with which the sequence is carried out can be adjusted, but if any other changes to the sequence are required, such changes must be managed back in the cerebral cortex and subcortical structures.

The brainstem and spinal cord map from the cortical outputs defining the selected motor behaviour to detailed muscle movements by the body to perform the behaviour.

All animal brains have the architecture illustrated in figure 3.2-II, called the recommendation architecture. This recommendation architecture has one structure defining and detecting conditions, a second structure determining the behaviour most strongly recommended by the current condition detections, and a third structure managing frequently used sequences of behaviours. As we describe in detail in appendix 1, any system which must *learn* a complex combination of behaviours tends to be constrained into this recommendation architecture form, just as any system which is *designed* to perform a complex combination of functions tends to be constrained into the instruction architecture form.

Information exchange within the brain system architecture

As we pointed out earlier, in a computer there is much more information exchange within a physical module than between two modules. It is striking that in the brain the same arrangement can be seen. We can use the degree of connectivity as a proxy for degree of information exchange. There is in general much more connectivity within an anatomical module than between different modules.

The cortex is a particularly clear example. The white matter inside the cerebral hemispheres connects neurons together. Of this connectivity, 99.7% is between neurons in the cortex. Only the remaining 0.3% is between a cortical neuron and a neuron in some structure outside the cortex.

For the other major anatomical structures, it is also the case that there is much more connectivity within the structure than between different structures. The implication is that, as for computers, there is much more information exchange within an anatomical module than between modules. In a computer there is also much more information exchange between the submodules of a major module that between the major modules. As we will see in the next section, the same is true of the brain.

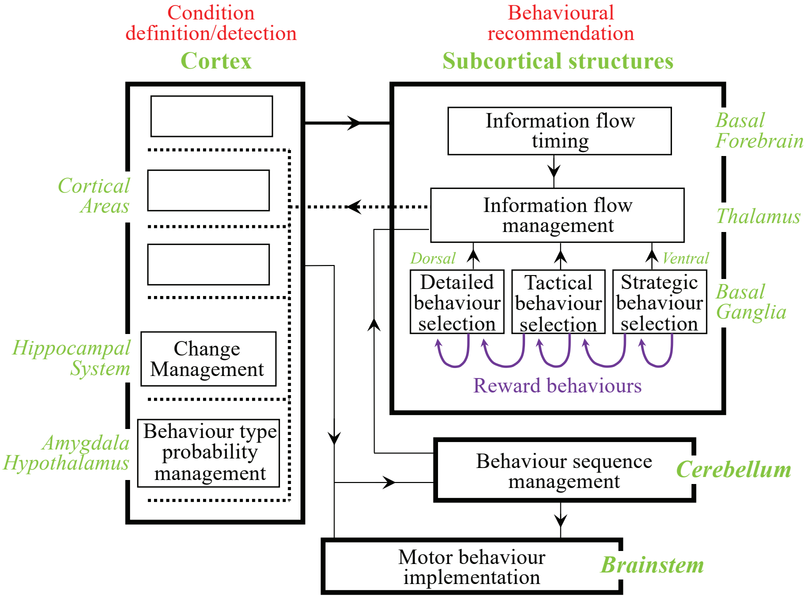

FIGURE 3.5. A more detailed view of the system architecture of the brain. As described in appendix 1, the brain is constrained into this architectural form by a group of practical considerations. Different cortical areas define and detect conditions on different levels of complexity. The hippocampal system manages the condition definition process to ensure that resources are used efficiently, and any changes have as small an effect as possible on previously learned behaviours. Technically, part of the hippocampal system performs behaviour selection processes, selecting which changes to cortical conditions will be implemented. The amygdala and hypothalamus define and detect conditions appropriate for recommending different general types of behaviour (such as aggressive or food-seeking behaviours). Again, technically, part of the amygdala performs behaviour selection processes. The primary behaviour selection module is the basal ganglia. The basal ganglia get many inputs from the cortex, interpreted as behavioural recommendations in favour of releases of cortical information or changes to recommendation weights. Cortical inputs to one end of the structure (called the ventral basal ganglia) are interpreted as recommendations in favour of strategic behaviours, while cortical inputs from different parts of the cortex to the other end (the dorsal basal ganglia) are interpreted as recommendations in favour of detailed behaviours. Cortical information release selections are communicated to the thalamus, which implements the behaviours. The exact timing of these releases can be critical, especially for managing changes to conditions, and this timing is managed by the basal forebrain. The cerebellum implements previously learned and frequently used sequences of behaviours by targeting the thalamus to release cortical outputs or by targeting the brainstem and spinal cord directly.

Pervasiveness of condition definition/detection and behavioural recommendation processes in the brain

We noted that in computers there are two general types of information processes and that we can describe user processes and all the more detailed electronic processes in terms of combinations of these process types.

In the case of the brain there are again two general types of information processes: condition definition/detection and behavioural recommendation definition/integration. Just as for electronic systems, it is the existence of these two types of information process that makes it possible to create the hierarchies of description needed for understanding. We will develop later in the book how cognitive tasks can be described in terms of these same two types of process on every level through anatomy and physiology to neurochemistry.

MORE DETAILED BRAIN SYSTEM ARCHITECTURE

All of the major anatomical modules of the brain have submodules. Furthermore, there are submodules of all those submodules and yet deeper "sub-submodules." At this point we will just consider some of the submodules of the major structures, the major connectivity paths between them, and the information processes performed by those submodules. How those information processes are performed requires discussion of yet more detailed anatomical structures, which we will come to later.

The cortex sheet is divided up into over 150 different areas duplicated in the left and right cerebral hemispheres, and these areas are the major submodules of the cortex. All areas have the same types of neurons but differ in numbers of neurons and the detailed arrangement of those neurons into layers. There is much more connectivity within an area than between areas.

The subcortical structures are a group of nuclei, and like the cortex they are duplicated on the left and right sides of the brain. The largest of these structures are the thalamus, the basal ganglia, the hypothalamus, the basal forebrain, the amygdala, and the hippocampus. The hippocampus forms part of a larger hippocampal system that also includes several nearby cortical areas. We will defer discussion of the substructures of the cerebellum and brainstem until later.

Figure 3.5 is a more detailed view of the same recommendation architecture illustrated in figure 3.2-II, but with these brain substructures and their information processes illustrated. We will discuss each of the more detailed modules in turn.

Cortical areas

Within the condition definition and detection subsystem illustrated in figure 3.5, modules are all the different cortical areas. Each area defines and detects conditions that are combinations of the inputs to the area. Depending on the area, most of the inputs come from one of the senses (eyes, ears, etc.) or from some relatively small combination of other areas.

A conceptual view of this connectivity situation is illustrated in figure 3.6. Some areas get inputs direct from one of the sensory organs, and the conditions defined in those areas are combinations of the sensory inputs. Such areas are called primary sensory areas. Other areas get inputs derived from only one sense but via intermediate areas. These areas are called monomodal, and the conditions within such areas are defined by combinations of conditions defined by earlier areas. Yet other areas get inputs ultimately derived from more than one sense. These areas are called polymodal, and the conditions within such areas are defined by combinations of information from the contributing senses. The areas that get inputs from the senses directly or via one or just a few intermediate areas are called lower areas, and areas that get inputs from the senses via many intermediate areas are called higher areas.

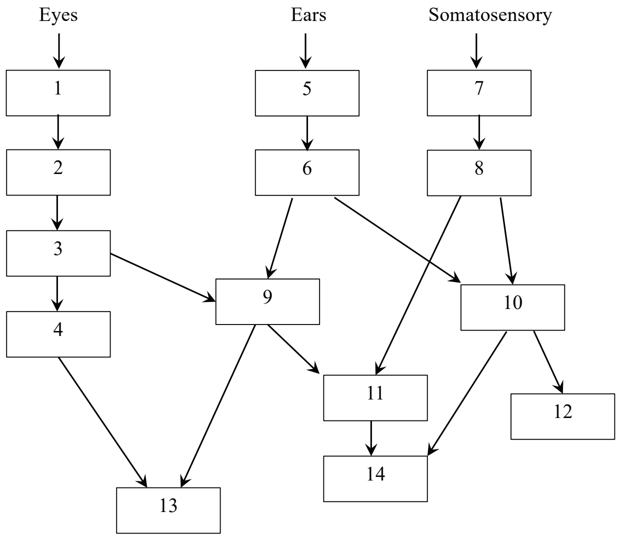

FIGURE 3.6. Simplified conceptual view of the connectivity between different cortical areas. Each area defines conditions that are combinations of its inputs. Most inputs to an area come from one or a few sources. A source is either a sensory organ or another cortical area. Monomodal areas get their inputs from one sense or from areas that derive their inputs from one sense. Polymodal areas derive their inputs (via intermediate areas) from multiple senses. In the diagram, areas 1 – 8 are monomodal and areas 9 – 14 polymodal. Because the conditions defined by one area are combinations of the conditions defined by its input areas, the complexity of the conditions is greater in the target area than in the input areas. In this context, greater complexity means that the total number of raw sensory inputs that contribute to a condition, directly or via intermediate conditions, is larger. Conditions on different levels of complexity are effective for recommending different types of behaviours.

If the *complexity* of a condition is defined as the total number of raw sensory inputs that ultimately contribute to the definition of that condition, directly or via intermediate areas, the complexity is different for each area. Note that the same raw sensory input may contribute multiple times to a higher condition via different intermediate conditions.

Conditions recommend behaviours, and conditions with different complexities are appropriate for recommending different types of behaviours. The appropriate behaviour in response to one isolated object may be different from the appropriate behaviour in response to a group of objects even if the group includes that one object. Our behaviour on meeting a group of friends will often be different from our behaviour when meeting one of them alone. Conditions defined by combinations of some relatively small number of visual inputs may be a good complexity for recommending appropriate responses to individual visual objects. Higher complexity conditions defined by combinations of those "object" conditions may be a good complexity for recommending appropriate responses to different groups of visual objects.

The brain has a set of cortical areas, each with a different range of condition complexity. Natural selection has ensured that for all essential behaviours, conditions in one or more of these ranges will be appropriate for recommending those behaviours.

Managing cortical information flows

As mentioned earlier, selected behaviours are often implemented by releases of information between different cortical areas. Attention behaviours are releases of subsets of current sensory information for detailed processing by the brain. An attention behaviour might single out information derived from just one object. Motor behaviours are releases of a subset of current cortical condition detections to drive physical movements. Cognitive processing behaviours are releases of particular subsets of condition detections from one cortical area to another. A simple example would be the behaviour of releasing the conditions detected within three different sensory objects to an area that detects conditions able to discriminate between different groups of objects.

Cortical connectivity is hardwired by axons. So how can there be a "release" of information if the connectivity carrying the information is always in place? The answer is that release of information is achieved by placing a frequency modulation on the subset of the information to be released. The effect of such a frequency modulation is that the output spikes from the neurons corresponding with the selected condition detections are bunched together in time and have a much larger effect on their neuron targets than detections that arrive more scattered in time. The modulation is imposed at a frequency of the order of about forty cycles/second, which means that bursts of spikes tend to arrive once every twenty-five milliseconds. This forty cycles/second, or 40 Hz, is the gamma band electrical frequency observed in an electroencephalogram (EEG) of the cortex when the brain is paying attention to something. The detailed mechanisms of this frequency modulation will be described later in the book.

Receptive field terminology

A condition is defined by a combination of simpler conditions. Hence the inputs to a module like a cortical area or a cortical neuron are simpler conditions, and the outputs from the module are more complex conditions. These more complex conditions that are outputs from a module are called receptive fields. In origin, the term receptive field of a module like a neuron referred to the area of the retina of the eye that provided input to the neuron, directly or via other neurons. The term receptive field has been generalized to refer to the set of circumstances in which a brain module such as a neuron produces outputs. In other words, the receptive field is the set of conditions defined and detected by a module.

Of course, the receptive field of one module is also a condition that is an input to other modules. In this sense there is no difference between a receptive field and a condition. However, we will refer to module outputs as receptive field detections. It is receptive fields that acquire recommendation strengths in favour of different behaviours in the subcortical structures.

Organization of cortical resources into a modular hierarchy

Within a cortical area, condition definition and detection by the cortex is further structured. The most common type of neuron in the cortex is the pyramidal neuron, named because of the roughly pyramidal shape of its central body. The cortex sheet is made up of a number of parallel layers of pyramidal neurons. Perpendicular to the cortex and penetrating through all the layers are structures called columns. Columns are therefore made up of a sequence of layers. Each layer is made up of a number of pyramidal neurons. The inputs to a pyramidal neuron arrive on a structure called a dendritic tree. This tree has a number of individual dendrites, and each dendrite has many branches.

Hence cortical modules can be observed on a number of levels: area, column, column-layer, pyramidal neuron, dendrite, and dendritic branch. A dendritic branch detects a group of very similar conditions defined by different combinations of its inputs from other pyramidal neurons. A dendrite detects a group of conditions defined by different combinations of its branches. A pyramidal neuron detects a group of slightly less similar conditions defined by combinations of its dendrite conditions. A column-layer detects a group of somewhat less similar conditions defined by combinations of its neuron conditions, and so on. In addition, there is much more information exchange within a module on one level than between the module and other modules on the same level.

This arrangement of cortical modules is analogous with the modular hierarchy of hardware in a computer system: printed circuit assemblies, integrated circuits, cells, logic gates, and transistors, with much more information exchange within a module than between modules. In both cases, the modular hierarchy reflects the resource economy achieved by organizing information processes into groups on the basis of similarity and customizing physical resources to perform each group very efficiently. However, the types of information processes organized in computers and brains are qualitatively different.

Hippocampal System

Learning requires changes to receptive field definitions. There are two major problems that must be addressed. The first is that the resources needed to record all the receptive field definitions cannot be excessively large. There are huge numbers of different receptive fields that *could* be defined, but defining a receptive field requires resources to record its definition. It is therefore critical to define receptive fields only when needed, and such receptive fields must be as valuable as possible for recommending behaviours.

The second problem is that because any one receptive field could recommend a wide range of different behaviours, any changes to its definition can jeopardize the integrity of all those existing recommendation strengths. Changes to existing receptive fields must be minimized as much as possible.

At each point in time the information processes performed by the hippocampal system address these two problems. The hippocampal information processes determine whether receptive field changes will occur and where in the cortex any changes will take place.

In figure 3.5, the hippocampal system is located in the condition definition subsystem. However, although some parts of the hippocampal system are in fact special purpose cortical areas, other parts are anatomically separate and perform behaviour selection functions. The special cortical areas define and detect receptive fields appropriate for recommending receptive field change behaviours, and the hippocampus proper selects the changes that will actually be made.

Amygdala and Hypothalamus

In certain situations, a general type of behaviour is urgently needed. If there is a threat to self or property, an aggressive behaviour of some kind is appropriate. If the threat exceeds our probable ability to handle it, an avoidance behaviour to escape the situation is appropriate. If we have not eaten for a while, food-seeking behaviour is appropriate. There is therefore value in the definition of receptive fields that can recommend general types of behaviour. Such receptive fields do not themselves recommend a specific behaviour; rather, they increase the probability that recommendations in favour of specific behaviours of the general type will be generated and accepted.

These general recommendations are subjectively experienced as emotions. Anger is the subjective experience of recommendation strength in favour of aggressive behaviours in general. Fear is the experience of recommendation strength in favour of avoidance behaviours. Hunger is the experience of a general recommendation in favour of food-seeking behaviours. The amygdala and hypothalamus perform the information processes required for this behaviour type selection function. However, once again figure 3.5 is somewhat simplified. The hypothalamus and part of the amygdala detect receptive fields appropriate for recommending different general types of behaviours. Another part of the amygdala makes the selections.

Basal ganglia, thalamus, and basal forebrain

At each point in time, current receptive field detections by the cortex target the basal ganglia. In the basal ganglia there are many different small groups of neurons that correspond with different behaviours. If a cortical output targets one such group, by that targeting it has recommendation weight in favour of the corresponding behaviour. The magnitude of the weight can be varied. Within the basal ganglia there is a competition between all the different groups to determine which behaviour has the largest total current weight, and this behaviour is implemented.

The neuron groups of the basal ganglia are physically arranged on a long structure, with the two ends called dorsal and ventral. At the ventral end the behaviours are strategic; at the dorsal end they are very specific. To give an example, imagine the behaviours required by a musician to perform at a concert. The behaviour of choosing to perform at a particular time and place is strategic. The behaviour of choosing which pieces of music to perform is more tactical. The behaviours of body language and comments to the audience are more specific, while the muscle movements to operate the musical instrument from moment to moment are very detailed.

In many cases, the selected behaviour is implemented by release of cortical receptive field detections from one area to another, or in the case of very specific behaviours by release to the brainstem and spinal cord. Such releases are performed by the thalamus. The basal ganglia communicate the selected behaviour to the thalamus, which then performs the required information release. The thalamus achieves this by imposing a frequency modulation on the signals to be released, as mentioned earlier.

Sometimes, especially for the behaviours of changing cortical receptive fields, the exact timing of the release is critical. Such precise timing is performed by the basal forebrain. As in the case of release by the thalamus discussed earlier, release timing is also achieved by frequency modulation, but in this case the basal forebrain imposes a modulation in the 8 Hz range on top of the thalamic gamma band modulation. This 8 Hz modulation shows up as the theta band frequency in the EEG.

Relationship between the brain functional and system architectures

The anatomical or system architecture of the brain has no resemblance to a computer architecture. However, natural selection favouring brains that make efficient use of resources has constrained the anatomical resources of the brain into a form in which different modules support different types of information process. This selection pressure means that just as the physical modules of a computer do not correspond with user applications or features, the anatomical components of the brain do not correspond with types of cognitive behaviours.

There is a long history, especially among philosophically oriented researchers, of looking for anatomical modules in the brain that correspond with psychological skills or features. This effort has been without success. There are many examples of the disconnect between anatomical structures and cognitive features.

One example is that observation of brain activity during the recall of past events shows that many different anatomical structures are involved, including different cortical areas plus the hippocampus, thalamus, basal ganglia, basal forebrain, amygdala, hypothalamus, and cerebellum. Conversely, the cortical areas active when imagining an event that has never taken place are almost the same as those active during recall of a real event. Furthermore, there are a number of cortical areas that are known to be active during a very wide range of cognitive tasks. The same half a dozen areas are active during all sorts of different types of memory tasks, such as remembering events, remembering facts, and short-term working memory.

A critical human skill is the ability to recognize faces. Most of us can come up with the name of any one of hundreds of different people just by seeing their face, and at the same time remember something about the person. If a brain area called the fusiform face area (FFA) is damaged it can severely reduce this ability. The FFA has in the past been offered as an example of an anatomical module performing a cognitive function. However, if activity in the brain is observed during a face recognition task, it is found that many other areas are also active. In addition, it is found that experts in some other visual tasks make use of the FFA even though the average person makes no use of the area for the same tasks. An expert in birds makes use of the area for recognizing different species, but the average person makes no use of the area for this purpose. Someone who can identify the make and year of large numbers of cars also makes use of the FFA for such identifications. It is also interesting that the area of the fusiform cortex used for face recognition is in the right hemisphere of the cortex, and the corresponding area in the left cortex appears to be critical for recognition of words presented visually.

For many cognitive tasks, the visual differences between objects with different behavioural implications are substantial. The visual differences between a dog, a watch, or a chair are considerable. We can often immediately recognize our dog, our watch, or our chair, but it is rarely necessary to be able to individually identify very large numbers of different individual dogs, watches, or chairs. The problem in identifying human faces is that faces are all fairly similar, but each one of a very large number must be individually identified because the appropriate behaviours in response are quite different. Similarly, two different written words can be quite similar visually but must be clearly distinguished from each other. The visual differences between two bird species may be relatively slight, but an expert must be able to distinguish between perhaps thousands of species.

The implication of all these observations is that the FFA is not a structure performing face recognition but a structure specializing in information processes that can discriminate between large numbers of visual images that are very similar to each other, when such discrimination is behaviourally important. When a human brain needs to learn a skill requiring such discrimination capabilities, the FFA is called into service to provide the necessary information processes.

Chapter summary

In computers there are two key architectures: the functional architecture, which is critical for understanding how to use a computer, and the system architecture, which is critical for understanding the design of the computer.

The functional architecture divides up the system into user feature components called applications. The system architecture divides up the system into physical modules that perform different groups of information processes, with the information exchange between modules minimized as much as possible. The information processes are of two general types: instructions and data read/writes, and two major modules in the system architecture are the processor that specializes in instructions and the memory that specializes in data read/writes. The modules perform all the information processes required to carry out the applications, but there are no one-to-one correspondences between applications and modules.

Although brains are qualitatively different from computers, brains and computers have some analogous architectural features. Brains also have a functional architecture and a system architecture. The functional architecture organizes cognitive capabilities into components like perception and various types of memory. The system architecture divides up brain anatomy into major modules that perform different groups of information processes, with much more information exchange within a module than between modules. The information processes are again of two types, but in the case of the brain the two types are condition definition/detection and behavioural recommendation definition/integration. These processes are qualitatively different from the process types in computers, but as in computers there are two major modules specializing in the two types. The cortex specializes in condition definition/detection processes and the subcortical structures in behavioural recommendation processes.

Also analogously with computers, there are no one-to-one correspondences between cognitive capabilities and anatomical modules. Attempts to establish such correspondences are misleading. One cognitive capability may show a severe deficit following damage to a particular structure, but that does not demonstrate that the capability is performed by that structure alone. It simply demonstrates that the one capability cannot be performed effectively without the information processes carried out by that structure. Other capabilities will also need the processes carried out by that structure, and the one capability will also need processes performed by other structures.

INDEX