INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center Street • Suite 600 • Berkeley, California 94704 • (510) 643–4274 • FAX (510) 643-7684

VISIT: An Efficient Computational Model of Human Visual Attention[1]

Subutai Ahmad
ahmad@icsi.berkeley.edu

TR-91-049

September 1991

Abstract

One of the challenges for models of cognitive phenomena is the development of efficient and flexible interfaces between low level sensory information and high level processes. For visual processing, researchers have long argued that an attentional mechanism is required to perform many of the tasks required by high level vision. This thesis presents VISIT, a connectionist model of covert visual attention that has been used as a vehicle for studying this interface. The model is efficient, flexible, and is biologically plausible. The complexity of the network is linear in the number of pixels. Effective parallel strategies are used to minimize the number of iterations required. The resulting system is able to efficiently solve two tasks that are particularly difficult for standard bottom-up models of vision: computing spatial relations and visual search. Simulations show that the network's behavior matches much of the known psychophysical data on human visual attention. The general architecture of the model also closely matches the known physiological data on the human attention system. Various extensions to VISIT are discussed, including methods for learning the component modules.

1. Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science at the University of Illinois at Urbana-Champaign, 1991.

VISIT: AN EFFICIENT COMPUTATIONAL MODEL OF HUMAN VISUAL ATTENTION

BY

SUBUTAI AHMAD

A.B., Cornell University, 1986
M.S., University of Illinois, 1988

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1991

Urbana, Illinois

VISIT: AN EFFICIENT COMPUTATIONAL MODEL OF HUMAN VISUAL ATTENTION

Subutai Ahmad, Ph.D.
Department of Computer Science
University of Illinois at Urbana-Champaign, 1991
S. Omohundro, Advisor

One of the challenges for models of cognitive phenomena is the development of efficient and flexible interfaces between low level sensory information and high level processes. For visual processing, researchers have long argued that an attentional mechanism is required to perform many of the tasks required by high level vision. This thesis presents VISIT, a connectionist model of covert visual attention that has been used as a vehicle for studying this interface. The model is efficient, flexible, and is biologically plausible. The complexity of the network is linear in the number of pixels. Effective parallel strategies are used to minimize the number of iterations required. The resulting system is able to efficiently solve two tasks that are particularly difficult for standard bottom-up models of vision: computing spatial relations and visual search. Simulations show that the network's behavior matches much of the known psychophysical data on human visual attention. The general architecture of the model also closely matches the known physiological data on the human attention system. Various extensions to VISIT are discussed, including methods for learning the component modules.

Acknowledgments

My experience doing research has so far been very pleasurable and enlightening, thanks to the support of several people. For this I am grateful to: Steve Omohundro for his constant energy, insights, and friendship (especially useful was his ability to guide me in the right direction... several times!), Gerald Tesauro for his advice on writing and doing research, Bill Baird, Brett De Schepper, Darrell Hougen, Pankaj Mehra, Bartlett Mel, Andy Moore, Bob Rafal, Terry Regier, and Andreas Weigend for helpful discussion, Andreas Stolcke for our initial discussions on the learning chapter and for being a great officemate, Bill Greenough for getting me interested in cognitive neuroscience, Jerome Feldman, Anne Treisman, Jitendra Malik, Joe Malpeli and Arthur Kramer for sharing their insights on attention and/or biological vision, and the staff at the International Computer Science Institute for creating an unusually well-run research center. I am especially obliged to Milton Epstein for helping me deposit this thesis from 3000 miles away and to Barb Cicone for her constant help in sorting out countless administrative pitfalls. Last, but not least, I want to thank Jonake Bose, Lucy Labrador, and my family for their unfailing love and encouragement.

1. Introduction

Every minute we are awake, we move our eyes in a series of rapid movements known as eye saccades. Although we are usually unaware of them, saccades can occur as often as 5 to 7 times a second. Since the fovea has a much higher resolution than the periphery, saccades let us scrutinize interesting objects everywhere in the scene. If our retina could represent every portion of the visual field with equal resolution, saccades would be unnecessary. We would save processing time but would need an order of magnitude more neurons, so there is a trade-off between time and space. Evolution has settled on saccades as a compromise.

A somewhat surprising (but experimentally well established) fact is that an analogous phenomenon occurs within our brain. Even when there are no overt movements of eye, head, or body, we are continually attending to different regions of the visual field. Experiments show that these covert shifts of attention can occur as often as 15 to 20 times a second. As with saccades, there are compelling efficiency arguments that explain the need for such a mechanism. This thesis presents a computational model of this second form of attention.

I first discuss the arguments that covert attention will be a crucial component in any realisitc visual system. I then discuss VISIT, a model of visual attention and describe a highly parallel connectionist network for implementing each of its component operations. The model is applied to two basic visual tasks: the computation of spatial relations, and recognition based on feature combinations. Attention is required to perform both of these tasks accurately in the presence of multiple objects. To understand these processes in detail VISIT is tested on two specific tasks: computing spatial relations among point clusters, and searching for a combination of features in cluttered scenes. The resulting system is efficient and flexible. The complexity of the network is linear in the number of pixels. Parallel strategies limit the number of sequential iterations required. For visual search, an efficient search strategy, SWIFT, is used to minimize the number of fixations. As a result, the network is able to solve these tasks with high resolution images.

Although the primary advantage of VISIT's design is computational efficiency, the main inspiration for the structure and function have been biological. The general architecture of VISIT matches the known physiological data on the human attention system reasonably well. The network's behavior is consistent with much of the known psychophysical experiments on human attention.

For example, there is a large body of psychological literature on visual search. VISIT combined with SWIFT is able to account for many of the interesting recent results in this domain. I discuss these aspects and review the literature within the framework of VISIT. I also describe possible extensions to the architecture, mechanisms for learning aspects of the architecture, and relationships to existing computational models of attention.

2. Why Pay Attention?

> "There are two criticisms of many of these neural net models. The first is that they don't act fast enough. Speed is a crucial requirement for animals like ourselves. Most theorists have yet to give speed the weight it deserves. The second concerns relationships. An example might help us here. Imagine that two letters - any letters - are flashed briefly on a screen, one above the other. The task is to say which is the upper one. This is easily done by older models, using the processes commonly employed in modern digital computers, but attempts to do it with parallel distributed processing appear to me to be very cumbersome. I suspect that what is missing may be a mechanism of attention. Attention is likely to be a serial process working on top of the highly parallel PDP processes."
>
> - Francis Crick, *What Mad Pursuit*, 1988.

This chapter outlines the primary arguments for implementing an attentional mechanism. There are two such lines of reasoning. Computational complexity arguments suggest that too much hardware would be required for high level vision without attention. There is also a large body of psychophysical evidence demonstrating that people need visual attention for certain tasks. In many cases the complexity arguments accurately predict the experimental results, suggesting that attention should be a fundamental component of any realistic vision system.

2.1 Computational Complexity Reasons

It is popular in the connectionist literature to use feed forward network models for vision (Goggin *et. al*, 1991; Keeler *et. al*, 1991; Le Cun *et. al*, 1990; Pomerleau, 1991; Zemel *et. al*, 1990). Typically the input layers are clamped with an image (sometimes a pre-processed image), with each input unit representing one pixel. The networks contain one output unit per item to be recognized. The output unit corresponding to the object in the image should be the most active. This general approach has certain advantages in restricted domains but breaks down when multiple objects exist

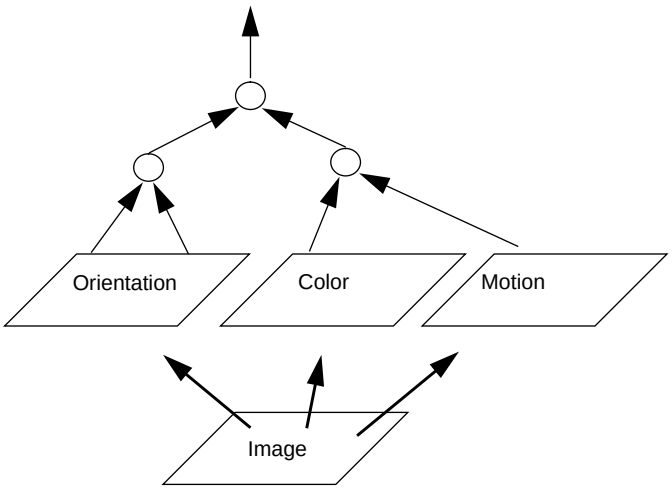


Figure 2.1. The typical feed-forward connectionist model of visual processing. A set of basic features are computed locally in parallel at every image location. Higher-level features are computed in a hierarchical fashion by explicitly combining lower levels features.

in the image, when the image has high resolution, or when the task is sufficiently complex.

To see this, consider the nature of massively parallel visual processing. In the brain, early stages contain individual neurons dedicated to computing local features at every image location in parallel (see Figure 2.1). There are neurons for detecting orientation, color, motion, depth, spatial frequency, etc. (Van Essen & Anderson, 1990). In terms of computation time this is a very efficient strategy. Due to the large number of pixels in realistic images, most researchers agree that similar parallel processing is required for computational vision. However, it is not difficult to see that this kind of encoding quickly becomes intractable at higher levels. It is not possible to explicitly represent every possible combination of features at every location. A better strategy would be to have one central process for detecting complex features that can be directed towards any region in the image.

As early as 1969, Minsky and Papert presented formal proofs for these intuitive arguments (Minsky & Papert, 1969). They showed that several simple visual predicates cannot be efficiently computed by feed forward threshold networks. For example, they proved that the connectivity predicate (the task of determining whether a curve is connected) requires an exponential number of weights. On the other hand, it is easy to envision a sequential solution using curve tracing. Minsky and Papert's arguments prove that, in general, vision cannot be performed efficiently with a strictly bottom-up system. More recently, Tsotsos (1990) has shown that general visual search is NP-complete without the use of top-down information and has suggested using visual attention to render the task tractable.

2.1.1 The Binding Problem

A basic goal of vision is to identify objects based on combinations of features (e.g. a red-horizontal bar). An immediate consequence of the complexity arguments outlined above is that this is a difficult operation to perform efficiently. Since objects can appear anywhere in the image, in any size and shape, it is infeasible have an explicit parallel representation for every possible combination of features at every location and scale. This dilemma is called the *binding problem* and appears in several different forms. Sejnowski refers to it as one of the fundamental open problems in neural computation and writes: "*the binding problem is a touchstone for testing network models that claim to have psychological validity*" (Sejnowski, 1986).

Obviously, people can associate feature combinations with objects, so there must be a way around the binding problem. We know of some operations that can be performed efficiently in parallel. For example it is possible to compute a global OR of each feature map. This operation identifies the particular features that are present in an image. If a single object were present in the image, this information would be sufficient to form the association. However, when multiple objects are present this setup will suffer from interference. If two objects are present and the red global OR is active, the system cannot identify which object is red.

Due to this interference, it is also difficult to identify the locations of individual objects when multiple objects are present. A natural solution is based on a mechanism that inhibits the representation everywhere except at the location of interest. If such a mechanism can be constructed efficiently, then the binding problem can be solved. A system using this mechanism can attend to individual objects to determine their feature combinations (Figure 2.1). The location of the object can then be determined by computing the center of mass of the active pixels.

2.2 Psychophysical Evidence

There is a large body of psychological literature on visual attention supporting the above arguments. Perhaps the most pertinent to the binding problem is the work on visual search (Treisman & Gormican, 1988). The basic experiments show that targets defined by a single feature can be detected in constant time but that targets defined by a combination of two or more features require

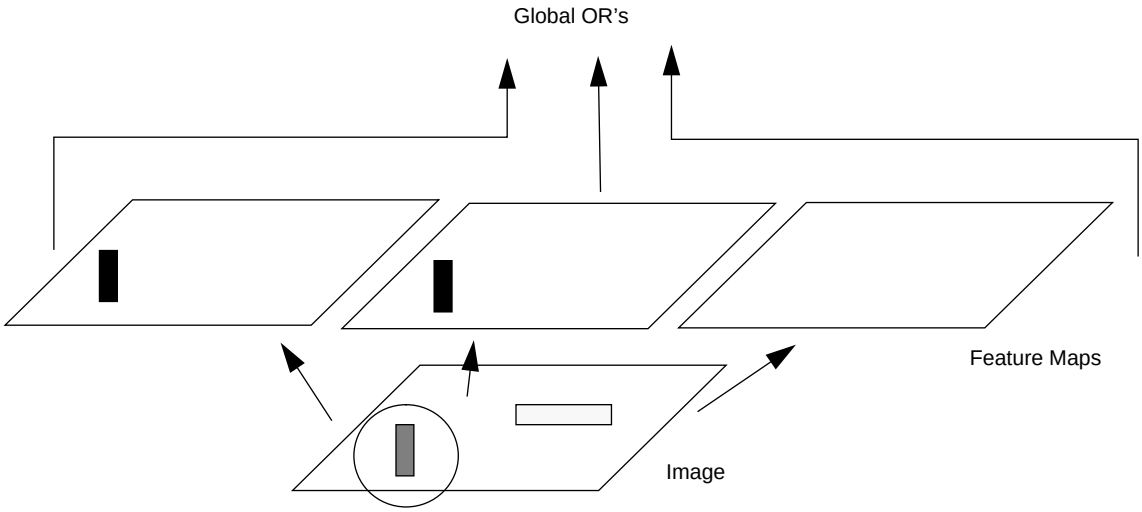


Figure 2.1. Inhibiting the feature maps everywhere except at a single region solves the binding problem.

time linear in the total number of objects.[1] Saccades are ruled out suggesting that internal processes are responsible for the sequential behavior. There have also been some physiological findings relevant to these experiments. The most convincing demonstration of attention is the experiment reported in (Desimone & Moran, 1987). They recorded from cells in the visual cortex of awake, behaving monkeys. They found that when monkeys were attending to a location in visual space, cells in the inferotemporal cortex (IT) responded only when stimuli were presented within that region. Cells in IT are retinotopically organized and normally respond to a large portion of the visual field. This experiment shows that the effective receptive field of these cells can shrink to cover only attended locations. There is a large body of research along these lines and Chapters 6 and 7 discuss these experiments in much more detail. Nevertheless the basic message is clear: visual attention exists in people and is required to perform a variety of visual tasks.

2.3 Interfacing Low and High Level Processing

The early stages of visual processing can be performed in parallel. Due to complexity reasons, later stages seem to require a sequential component. What is the nature of the interface between the two levels? The ease of our everyday visual experience suggests that it is efficient, flexible, and

1. There are some exceptions - see Chapter 7.

includes mechanisms for top-down control. Ullman (1987) has proposed an elegant framework along these lines. He argues for the existence of a small fixed set of visual control primitives. Given a task specification, an intermediate system dynamically creates a sequential *visual routine* for solving it. This would involve initiating the primitives in some order. He proposed five specific primitives, including primitives for focusing attention, for marking locations, for spreading activation, for curve tracing, and for indexing. (Jolicoeur, Ullman, & Mackay, 1986) describe some direct psychophysical evidence in support of this framework. They find that the time required to report whether two stimuli lie on the same curve increases linearly with the distance between them. The distance is not the physical distance between the stimuli on the image, but rather the distance along the curve. The authors propose that this is due to a process which searches sequentially along the curve.

As a step towards a neurally plausible realization of the full visual routines framework, the rest of this thesis describes a model of visual attention and associated control structures.

3. Implementing VISIT

The arguments favoring attention are relatively clear but constructing a feasible mechanism is non-trivial. How can a parallel connectionist network, with its fixed set of units and interconnections, implement attention efficiently and flexibly? The final network must be able to inhibit a topographic representation everywhere except at a single region. The size of the focus of attention should be continuously variable to accomodate stimuli of different sizes. The network should be able to shift the attended location dynamically. There should be a reasonable way to choose the locations which are attended - both bottom up and top down information should be incorporated. Mechanisms should exist for handling the control issues. The resulting system should also provide a flexible interface to higher level recognition systems. This chapter addresses these questions and describes VISIT[1], a network for implementing visual attention.

3.1 Global Network Structure

The above requirements can be clumped into three distinct tasks: an efficient gating scheme, a flexible method for choosing interesting locations, and sequential control. In VISIT separate networks are responsible for each operation (Figure 3.1.). This makes it possible to optimize each one without compromising efficiency. The *gating network* is responsible for suppressing all activity except that at a given region. The region is continuously variable and is used to limit the extent of high level processing in the image. The locations of interest are determined separately by a *priority network* which is free to use top-down and bottom-up information. The *control network* is responsible for sequencing and for mediating the information flow between the gating and priority networks. The model also incorporates a *working memory* for the temporary storage of relevant information. These four systems are described in the following sections.

1. A loose acronym for a network that performs VIsual Search ITeratively.

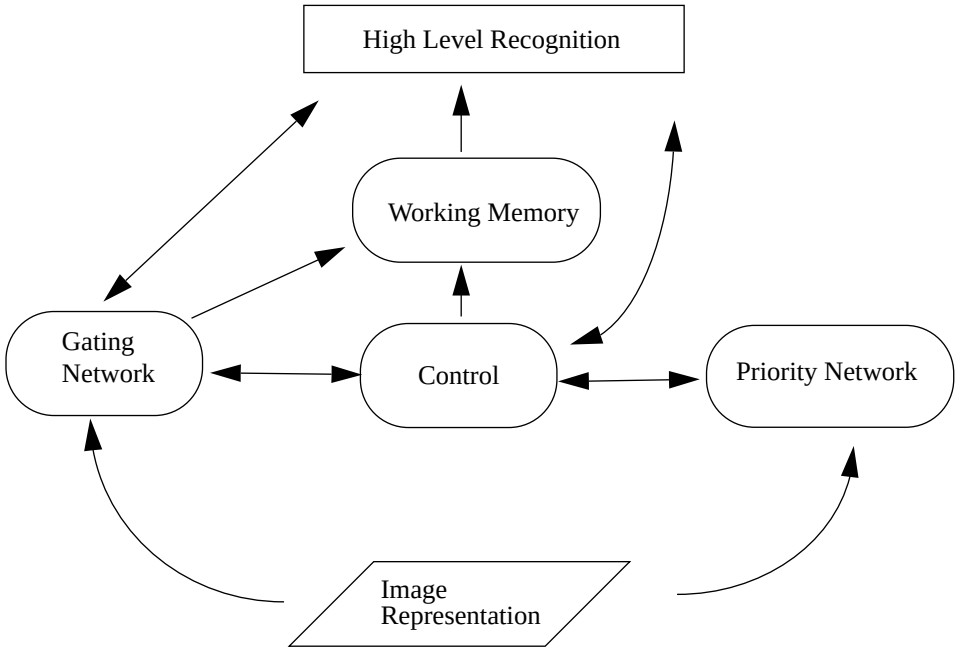


Figure 3.1. General architecture of VISIT.

3.2 The Gating Network

The gating network actually implements the focus of attenion. It is responsible for restricting subsequent processing to an arbitrary circular region. The design is simple. The network is implemented as a retinotopic layer of units, with one gate unit per pixel. Each unit encodes its position within the image. Each unit also receives input from three external units which represent the center and radius of the current circle of attention, (A_x, A_y, A_r). Each unit, *i*, computes the equation for a circle and turns on if it is *outside* the circle, i.e. if the following inequality is true:

$$(x_i - A_x)^2 + (y_i - A_y)^2 > A_r^2 \tag{3.1}$$

where (x_i, y_i) refers to the position of the unit. The output of the gating network is used to inhibit subsequent gated layers (Figure 3.2). The network continually updates the circle to reflect changes in the activity of the 3 external units. The effect is a layer of units which filters the input image according to a global control signal. The above scheme is simple and efficient. The hardware required to implement it is minimal: 3 input connections per gate unit. Once the location of the circle is known, it takes one time step to update the focus.

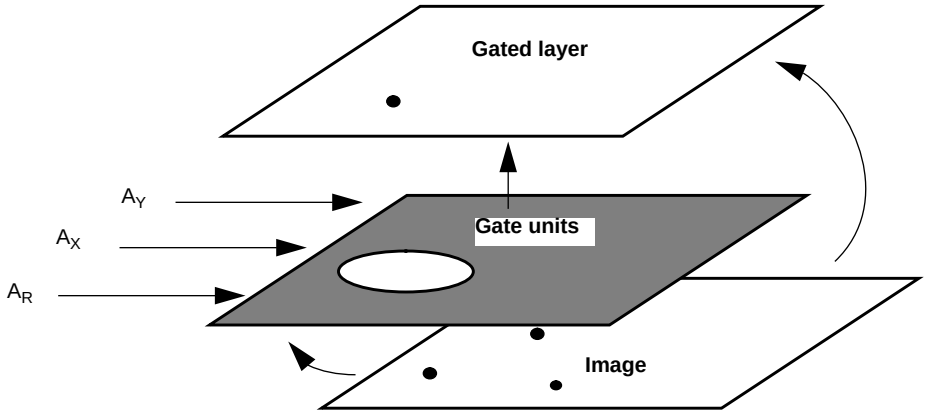


Figure 3.2. The gating network inhibits subsequent layers, restricting activity to a single circular region.

3.3 The Priority Network

The gating network relies on a good mechanism for selecting interesting locations to visit. Psychophysical experiments show that in general, this computation is highly context dependent. In some situations a bottom up method such as choosing the brightest image location is appropriate. At other times top-down information such as prior knowledge of an important location must be used.

In VISIT, the priority network is responsible for this computation. The general idea is to create a retinotopic map where activity is proportional to the "relevance" or priority of locations. Each location in the priority map corresponds to a small circular region of the image. There are three units for each location. The output of the first unit represent a priority value from 0 to 1. The priority computation can vary with the task. The only constraint is computational efficiency. Chapters 4 and 5 describe two possible schemes for computing priority.

The remaining two units are used to help the control network shift attention (Figure 3.3). These units encode the vector difference between the center of mass of the cluster of points within their receptive fields and the point (A_x, A_y). In this way each grid location encodes an "error vector" for adjusting the focus of attention. These vectors are continually updated to compensate for changes in A_x and A_y. By adding the vector with the highest priority value to A_x and A_y, the focus of attention can be shifted to the location of the most salient cluster in a single step.

The error vector representation is quite flexible. The control network has several options available to it, depending on the requirements of the current task. It can decide to move the focus of attention

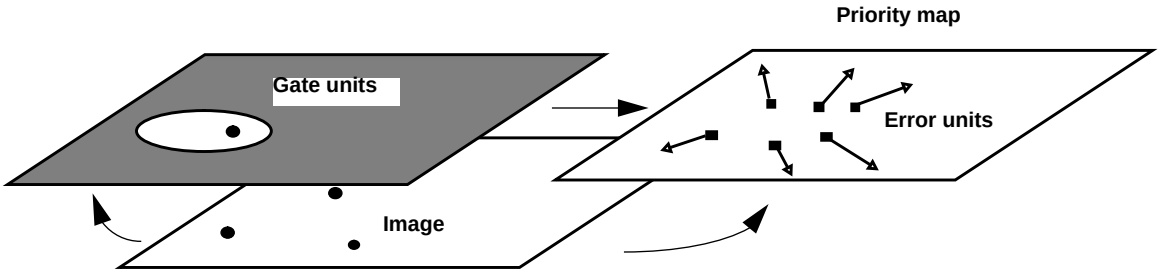


Figure 3.3. Error units. At each location, an offset vector is computed relative to the current center of attention.

to the most salient or the least salient location. It can select the location nearest the current focus by selecting the smallest error vector. To choose locations to the right, it can select a vector whose first component is positive. Or, the control network can decide to ignore the error map completely and move the focus according to some independent top-down information.[1]

It is important to note that locations in the priority map do not represent individual objects, but a fixed small region in the image. The former requires a segregation step that in general can be quite complex and time consuming. Since objects can vary in size and shape, it is highly unlikely that segmentation can be performed in a simple feed-forward network. (Several psychophysical studies suggest that even some simple aspects of object segregation may actually *require* attention (Treisman & Schmidt, 1982).) However, the representation does impose some restrictions. An object that is larger than the receptive fields of the individual priority units could cause several of them to become active. As a consequence, the location information provided by the error map is somewhat coarse. To circumvent this problem we include a fine tuning mechanism as part of the control network (Section 3.5.2).

1. The error vector representation was inspired by discoveries of a similar mechanism in the monkey superior colliculus for controlling eye saccades (see Chapter 6).The output of the confidence units is similar to the saliency map in (Koch & Ullman, 1985) (see Chapter 10), although in general many factors may contribute to the saliency of a given location.

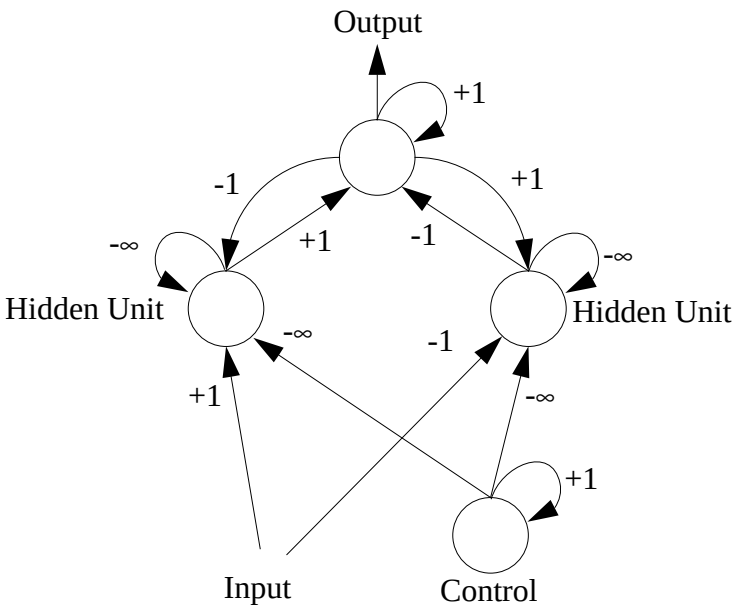


Figure 3.4. Schematic of a binding network. The network continually tracks its input signal until a control signal is sent, at which point the output is frozen to be the current signal.

3.4 The Working Memory

One of the primary motivations for visual attention is the ability to compute locations of individual objects. In our system, the location and scale can be obtained directly from the values of the units representing A_x, A_y, and A_r. Since retrieving locations is now a sequential process, we need a mechanism for capturing and temporarily storing these values. We accomplish this with small recurrent networks for each value that needs to be stored. Each of these "binding" networks continually tracks a particular unit (one of A_x, A_y, and A_r) until a control signal is sent, whereupon it freezes the output to be the current value of the unit.

The network for doing this is shown in Figure 3.4. The two hidden units have a positive linear activation function (0 if the weighted sum of its inputs, *i*, is negative; *i* otherwise). The output unit is linear. While the control unit is off, the hidden units compute the difference between the value of the assigned unit and the current output, and sends it to the output unit. The left hidden unit indicates when the output should be decreased whereas the right unit indicates when it should be increased. An inhibitory connection is included from each hidden unit to itself to ensure that after it fires, it stays off long enough to allow the output unit to adjust itself. When the control unit is turned on, the hidden units are shut off by large negative weights. An excitatory link from the control unit to itself ensures that once the control unit has fired, it stays on, preventing further adjustments. Three of these "binding networks" are used for each set of parameters that are stored.

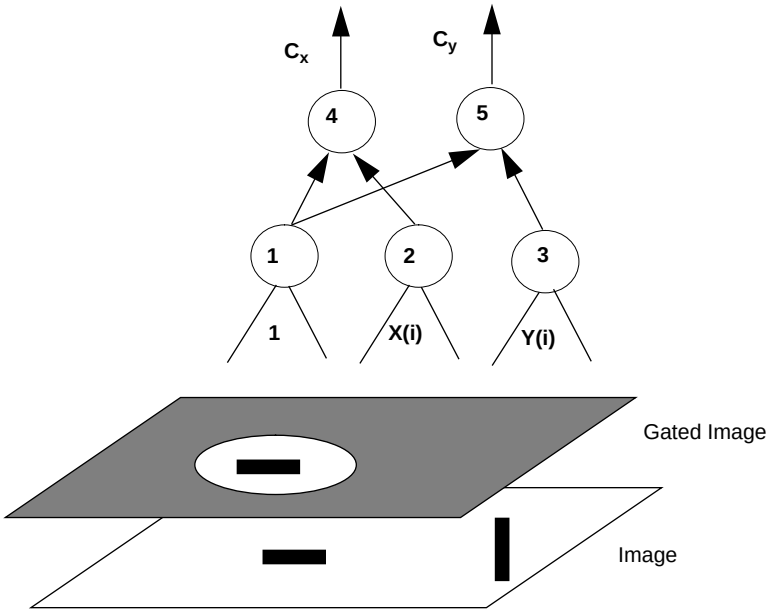


Figure 3.5. A network for computing center of mass.

3.5 The Control Network

One of the most interesting aspects of visual attention is the control process. How does the system focus on an object? How does it shift from location to location? How is it prevented from visiting the same location twice? These issues deal with the sequential aspects of attention and are often ignored in models. One of the contributions of this thesis is to make these processes explicit. There are several possible approaches, but not all of them will be efficiently implementable in neural hardware. Therefore I have tried to enforce the constraint that every aspect of the control must be implementable as a connectionist network. This section describes the function and implementation of these networks.

3.5.1 Computing Location

Although it is not possible to compute the locations of every object in the scene in a feed forward network, it is possible to compute the center of mass of the entire image. When attending to a single object, this computation will accurately localize it. This is the scheme used in VISIT (see Figure 3.6). The center of mass, (C_x, C_y), is defined as the average of the x and y coordinates of the active

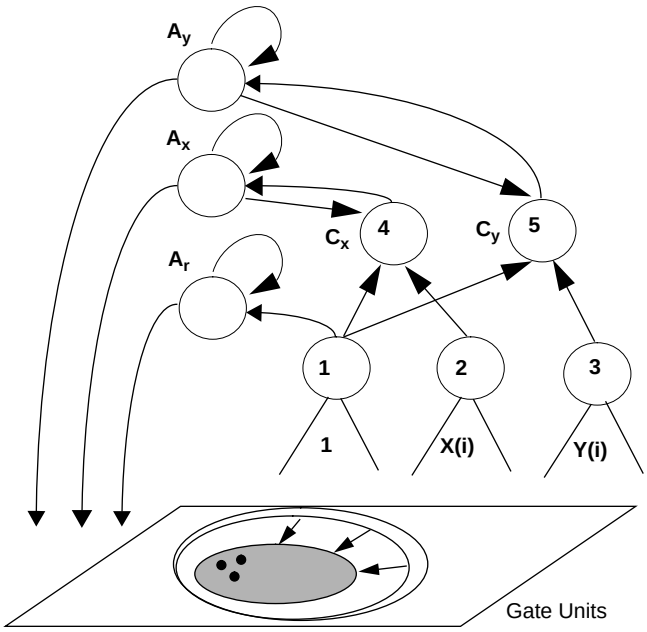


Figure 3.6. The units which continually fine tune the focus of attention.

points:

$$C_x = \frac{\sum_i X(i)\, a_i}{\sum_i a_i} \text{ and } C_y = \frac{\sum_i Y(i)\, a_i}{\sum_i a_i} \tag{3.2}$$

where $X(i)$ and $Y(i)$ denote the x and y coordinates of the i'th unit and a_i denotes its activity. $\sum a_i$ can be computed by a unit which receives input from all gate units with a weight of 1 (unit 1 in the figure). To compute the numerators we include two units with links to every gate unit (units 2 and 3). The weights from the i'th gate unit to each of these two units are $X(i)$ and $Y(i)$, so units 2 and 3 compute $\sum X(i)\, a_i$ and $\sum Y(i)\, a_i$, respectively. Two units (units 4 and 5) perform the division to obtain C_x and C_y.

3.5.2 Fine Tuning the Focus

As part of the control network, VISIT includes two mechanism for fine tuning A_x, A_y, and A_r to settle on the center and size of the cluster of points within the current focus of attention (Figure 3.6). The outputs of the center of mass network is fed to two units which compute the difference between the center of mass and the attention parameters: $(C_x - A_x, C_y - A_y)$. The units representing $A_x$and A_y receive this difference as input. By adding their own output to this difference, they accurately update the focus to center on the object within it.

To get an estimate of the size of the object the unit representing A_r continually adjusts the size of the focus to match the object within it. In the current implementation, as long as $\sum a_i$ remains constant, the unit decreases A_r by a small amount. If the sum decreases, indicating that the scale has become too small, the unit increases A_r slightly and stops. This works well for the relatively sparse images in this thesis. For more complex images a more sophisticated system would be necessary for detecting when the focus has become too small.

The two mechanisms described here allow the network to accurately compute the location and scale of the object within the focus. It is interesting to compare the temporal behavior of these mechanisms. The center of mass computation can be performed in a purely feed-forward manner and only requires a single shift in the focus (assuming the scale of the focus is initially larger than the object's size). The scale computation however is an iterative process and is typically the slowest aspect of the network. Although it may seem that VISIT has given up slow segmentation for a slow fine tuning process this is not a limiting factor. For many tasks object scale is unimportant so it is unnecessary to wait for the scaling mechanism to settle. At least one psychophysical study on attention (Kramer & Johnson, 1991) shows that the scale of an object does have an impact on performance. It would be interesting to perform experiments on human subjects to directly compare the time for computing object locations and scale.

3.5.3 Sequencing

There is another issue to consider: as the network fixates on successive objects, we would like different sets of binding networks to be instantiated. To do this we need some sort of a sequencing mechanism which will send control signals to successive binding networks. This is accomplished by the network shown in Figure 3.6. The signal unit fires when the focus of attention has stabilized for three iterations. The first time it fires, Cntrl-1 is activated; Cntrl-2 fires if Cntrl-1 has fired *and* the signal unit has gone from 0 to 1, and so on. In VISIT, the unit "Cntrl-i" in the figure corresponds to the control signal for the *i*'th binding network. Thus on successive firings of the signal unit, successive binding networks are frozen. The sequencing network is robust in the sense that it is insensitive to delays in the signal unit's firing pattern.

3.5.4 Shifting the Focus

In order to shift focus, the network must select the next location. This operation can be done using top-down or bottom-up information, or some combination of the two. It is important that compu-

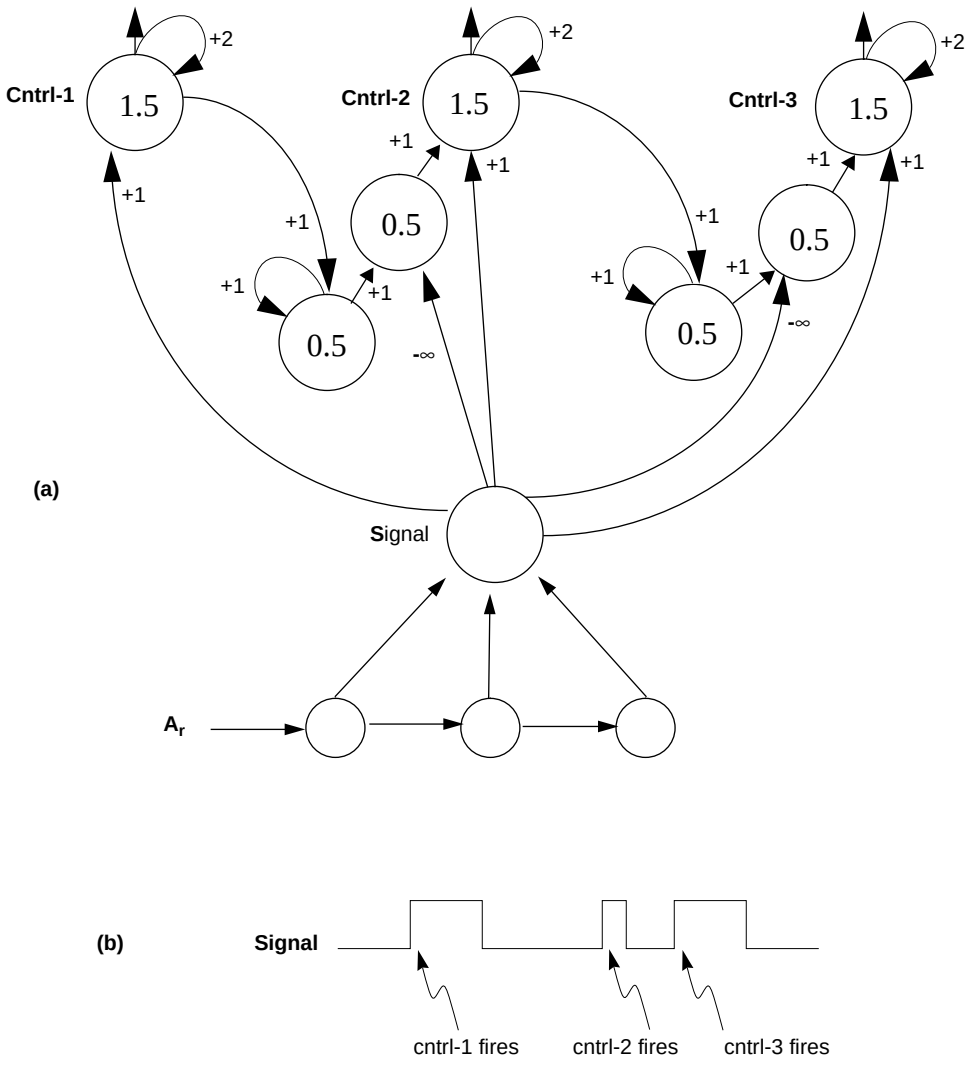


Figure 3.6. A detailed diagram of a portion of the sequencer. The units shown above sends control signals to the appropriate binding networks. The signal unit fires when all of its inputs are equal. The first time it fires, cntrl-1 starts firing. The next time, cntrl-2 fires, and so on. This structure is used to send control signals to successive binding networks.

tation time is kept as short as possible. In many situations it requires a mechanism for selecting the maximally active priority unit. This task has been studied extensively in the connectionist literature and there are several ways to do it. The current implementation of VISIT contains a unit with a built in max function, and assumes that the operation can be done in more or less constant time. This is not an unreasonable assumption. Given the complex analog processing that occurs within neuronal dendritic branches, it is quite possible that a function as complex as maximum can be computed very quickly.

An alternate way to do this is to construct a *winner-take-all* network (Feldman & Ballard, 1982) a

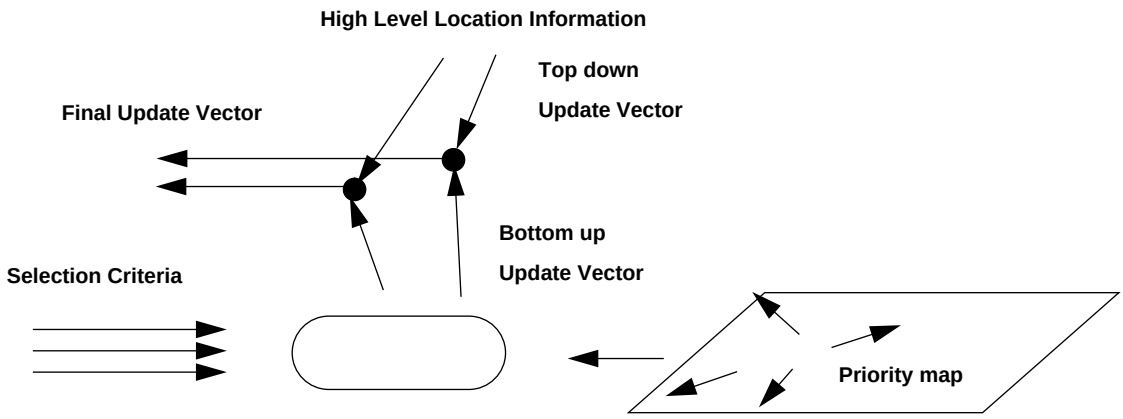


Figure 3.7. A flexible architecture for selecting new locations.

network in which each unit has mutually inhibitory links. If configured properly, such a network will settle into a state where only the maximal unit is active. However these networks can take a long time to settle, especially when the competing values are larger and very similar. It is also seems to be quite difficult to find the correct set of inhibitory weights to create a robust solution. With an analog VLSI implementation it may be possible to overcome some of these limitations (Lazzaro *et. al*, 1989).

A more robust alternative is to construct a log-depth pyramid shaped network as in (Koch and Ullman, 1985) where each node computes the maximum of node values "below" it. However, since the human visual system is dealing with a very high resolution image (about 10^6 pixels), there just isn't time for a binary log-depth network. It might be feasible if every level computed the max over a large number of neurons. For digital implementations, an attractive possibility is the mechanism described in (Srinivas and Barnden, 1989). Their network selects the maximum in time logarithmic in the number of *active* units, and not the total number of units. Given that top-down mechanisms can often eliminate many of the locations in parallel, this scheme is likely to provide a fast response time.

Incorporating Top Down Knowledge

In a realistic system, choosing the next location will be modulated by top-down knowledge. As mentioned before, error vectors allow a fair amount of flexibility. It is possible to pick the nearest relevant location, locations only to the left, etc. Prior knowledge about absolute locations in the image may also be important and should override all bottom-up information. This level of flexibility is not currently implemented, but a simple system for doing this is diagrammed in Figure 3.7. A sub-network receives inputs specifying some general criterion for selecting the next bottom-up

location (nearest, to-the-left-of, etc.). Based on this information it selects one of the error vectors. This vector is fed to two more units. If top-down location information is present, then these units will transmit it, otherwise the bottom-up error vector is used.

3.5.5 Inhibition of Return

In order to avoid oscillations, the network must have some means of preventing the focus from shifting back to a location that was just attended. An easy way to do this is to inhibit the priority units corresponding to the object currently being visited. A similar scheme is believed to exist in people and has been termed *inhibition of return* (Posner, Cohen, & Rafal, 1982). To implement this, we include an additional control signal that fires just before attention shifts to a new location. All the units that are within the focus of attention shut themselves off until a subsequent reset signal. There's one drawback to this: sometimes the system is attending to a large portion of the image, or even the entire scene. With the above scheme, if the network is in this state and shifts to a new location, it might inhibit all the priority units. This is clearly undesirable and suggests that the control signal should not be sent purely automatically, but instead should depend on the context. Interestingly, recent psychophysical studies confirm this behavior in people - inhibition of return doesn't seem to be purely automatic but is task dependent (see Section 6.1.4).

This chapter has described a number of connectionist networks for implementing a fast and flexible attentional mechanism. The model scales well in that the number of connections and units is linear in the number of pixels. The model requires a small constant number of iterations to shift focus. The next few chapters explore applications of this architecture and its relevance to biological systems.

4. Computing Spatial Relations

This chapter describes the application of VISIT to the computation of spatial relations. Consider the visual task of determining whether a set of three points form an equilateral triangle (Figure 4.1). People are very good at solving this kind of problem. Standard feed-forward networks for solving this task would not scale well to realistic image sizes. One reason for this is that local information does not contribute to the solution: there's no helpful information in the vertices themselves. In Figure 4.1, note that the only difference between the two images is that the top vertex has moved down and to the right. If each vertex was a single pixel, then the Hamming distance between the images is only two. To solve this problem, the network must extract relational information such as the distances between the vertices. The difficulties posed by this problem are common to a wide variety of visual tasks and makes it a useful touchstone problem.

The most straightforward neural representations assign a distinct unit to each pattern that must be classified. Unfortunately, the space of possible triangles is much too large for this kind of approach to be biologically possible. The optic nerve consists of about a million fibers from each eye, and so it is reasonable to consider square images which are a thousand pixels on a side. Since each of the three vertices can occupy any of these pixels, the total number of possible triangles in such an image is about $1000^6=10^{18}$. A brute force representation would require about a million times as many neurons as we have in our entire brain for just this one task. Restricting the units to represent just the set of equilateral triangles would still require about 10^{12} units. If coarse coded representations are used, these numbers can be reduced somewhat but it is still unappealing due to is lack of generality.

Techniques have been proposed for introducing translation and rotational invariance into networks (Giles et. al., 1987) which eliminate the need for independent feature detectors at every location. Unfortunately these methods require that every unit have a large (quadratic) number of connections with complicated weight linkages between them. Furthermore, positional information is lost in these representations - one cannot directly retrieve the location and orientation of the objects in the image.

These difficulties would disappear if we could directly extract the real valued coordinates of the vertices, say in the activations of *6* units. Using this representation it would be easy to construct

units which compute the distance between a pair of points. The following sections describe a simple implementation of this technique using VISIT.

4.1 VISIT and Equilateral Triangles

Figure 4.2 shows a schematic of VISIT as used to solve the equilateral triangle problem. The basic structure of the network is as described in the previous chapter. For this simple problem, the priority map just ranks local patches according to size. The working memory consists of sets of 3 binding networks arranged in succession. Each set stores the center and radius of the focus after each fixation. The sequencer determines which set of binding networks should store the current location.

The system starts off with the focus of attention covering the entire image plane. The fine tuning mechanism shrinks the focus until it is wrapped around the triangle. Once it stabilizes, a control signal is generated to store the current parameters. The error vector corresponding to the highest priority location is chosen to shift the focus to one of the vertices. The sequencer uses successive control signals to store the locations of the three vertices in the next three binding networks. As the network visits each location, the corresponding error units are inhibited, preventing the system from visiting the same location twice.

After every location has been visited, the first set of binding networks will contain the position and scale of the triangle. The next three bindings encode the positions and scales of the three vertices in the order that they were processed. These values are made available to a set of units each of

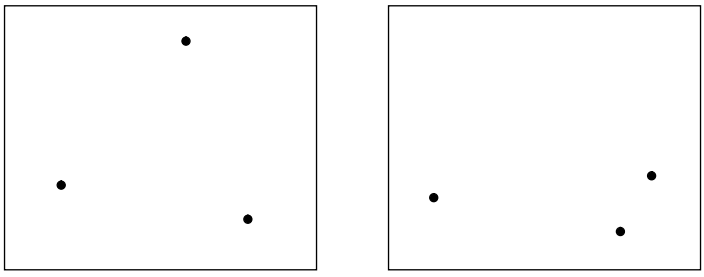


Figure 4.1. The left triangle is equilateral.The right one is not.

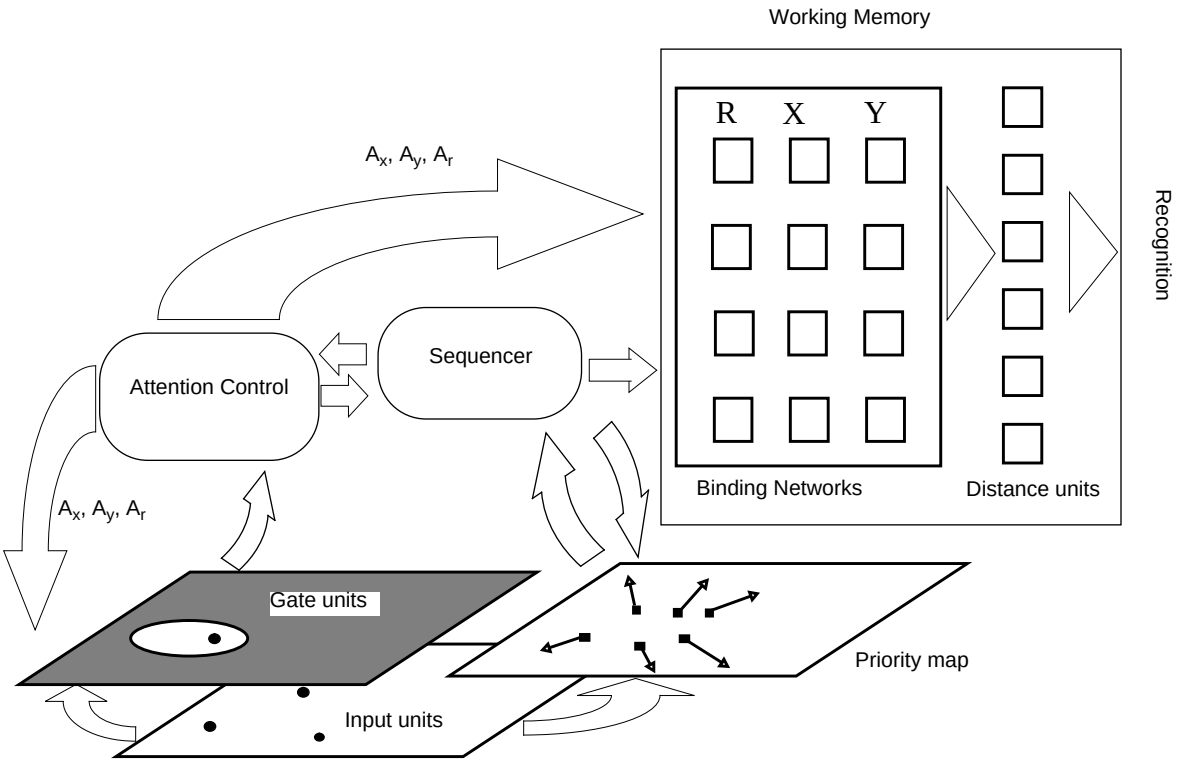


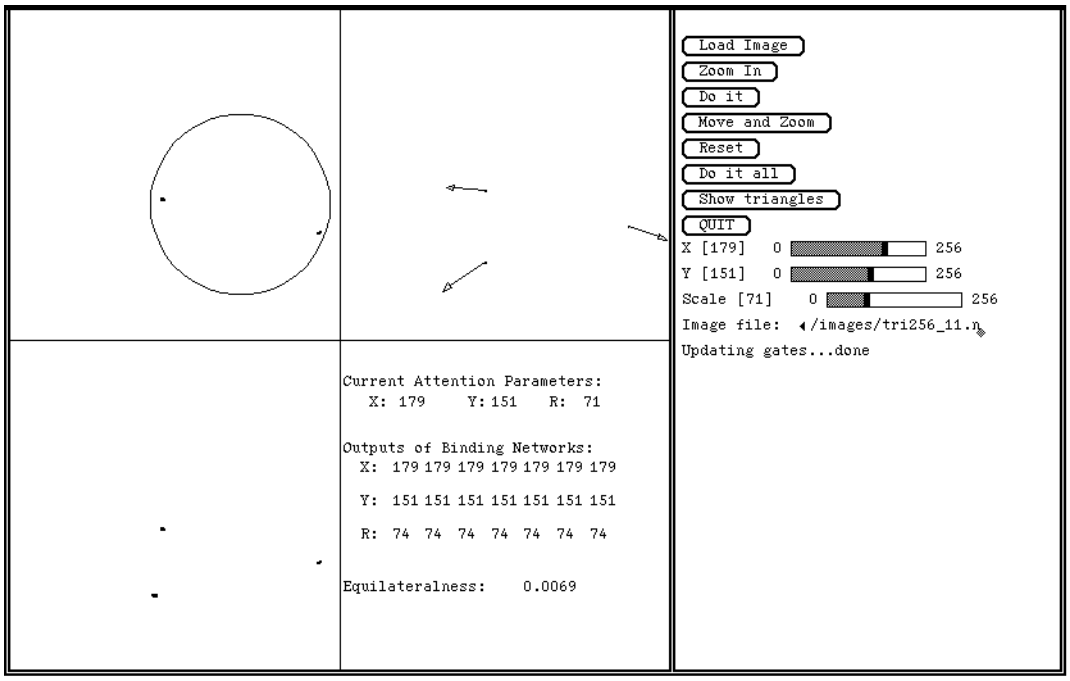
Figure 4.2. Overall architecture of the network.

which compute the distance between two of them. The outputs of the distance units are used to determine equilateralness.

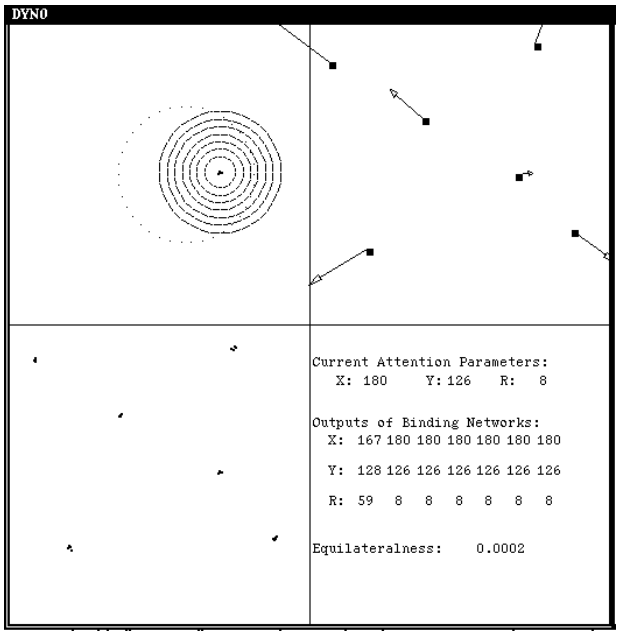
4.1.1 The Simulator

The simulations in this chapter used images with *256* by *256* pixels, or a total of *65,536* inputs. The complete network consisted of *131,912* units, or a little more than twice the number of pixels. *131,072* of these were the input and gate units; the exact number of units in the priority map varied depending on the coarseness of the map. The total number of weights was approximately *700,000*, or about *10.7* times the number of pixels. This compares rather favourably against methods which scale quadratically where about 256^4, or *4,294,967,296* weights would be required. Conversely, given a limit of *700,000* weights, such a scheme would be restricted to *29* by *29* pixel images.

Figure 4.3 shows the graphical output of the simulator. In each display, the lower left quadrant displays the current input image (three point clusters). The upper left quadrant displays the output of the gated image. The circle shows the implicit focus of attention represented by the three parameters. In Figure 4.3(a) note that only the activity within the focus is allowed to propagate. The upper



(a)



(b)

Figure 4.3. Examples of the system behavior for 256x256 bitmaps of point clusters. In both displays, the lower left quadrant shows the image; the upper left quadrant shows the output of the gate units. The error vectors are displayed in the upper right and the outputs of selected units are shown in the lower right. (a) shows a snapshot of the system with the focus of attention near one of the clusters. (b) shows the dynamic scaling behavior as the focus tries to fit the cluster of points within it.

right quadrant displays the error vectors. Each arrow represents the error vector for that location. The shaded square represents the priority value - the darker the square the higher the priority.

Figure 4.3 (b) illustrates the fine tuning scheme on the simulator. The focus of attention is initialized to a wide circle, slightly off center (dotted circle). The set of concentric bands show successive steps as the focus of attention decreases in size and shifts its location to fit the cluster inside. Note that the correct center is found in one step, whereas the scale computation requires several iterations. Figure 4.3(b) also demonstrates the sequencer and binding networks. The bottom left quadrant shows the output of the sets of three binding units. The first set of units has been frozen to *(167,128,59)* whereas the rest of them are still free to follow the attention parameters. If a control signal were to be generated now, the second set would be frozen at *(180,126,8)* to represent the current parameters.

4.1.2 Learning Spatial Relations

It is clear that VISIT can compute equilateralness with a linear number of weights, but there is an additional important advantage. The system has the ability to transfer the image representation from a pixel based representation to a highly compact one involving only locations and distances. This allows spatial relations to be *learned* easily. A system using just an image based representation will need a large number of examples to learn the correct relation. The spatial relationships that define equilateralness will have to be discovered for each position, scale, and orientation of the triangles. With a compact representation, the network can easily learn the correct function with very few training examples.

This was demonstrated by feeding the outputs of the working memory to a backpropagation network (McClelland & Rumelhart, 1986) which was used to learn to classify equilateral triangles. The training images consisted of random triangles, approximately *50%* of which were equilateral. Some noise was added around each vertex. For each image the focus of attention was initialized to cover the entire image plane. The system was allowed to run until all the units in the priority map were processed. The outputs of the working memory were then used to train the backprop net. The teacher signal was generated according to:

$$1 - \frac{|l_1 - l_2| + |l_2 - l_3| + |l_1 - l_3|}{l_1 + l_2 + l_3} \tag{4.1}$$

where l_i is the length of the i'th side. This is a function which is 1 for equilateral triangles and degrades gradually to 0 as the triangles deviate from equilateralness.

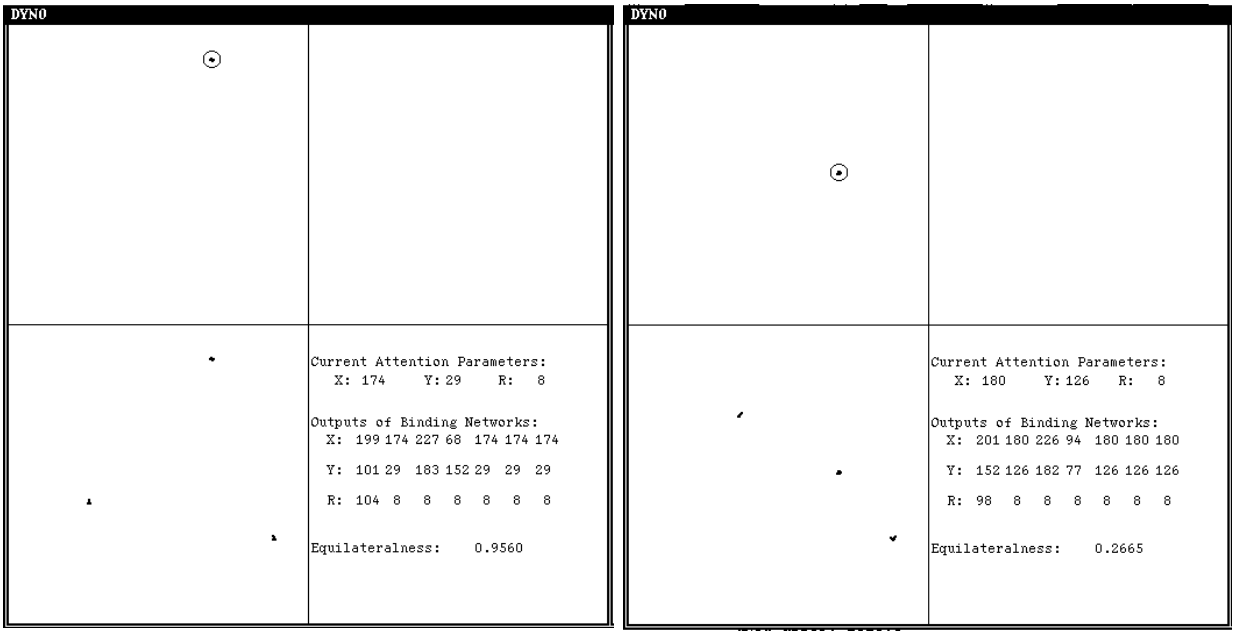


Figure 4.4. Results after parsing an equilateral triangle and a non-equilateral triangle. (Only the first four binding units are used here.**)**

Essentially, the network just had to learn the equality function of three of its inputs. With a training set of only 100 triangles the network score was consistently greater than 0.9 for equilateral triangles on independent test sets. This would be quite remarkable for a feed forward network considering the size of the input space. Furthermore, note that due to the compact representation, the number of required training examples is independent of the image size.

Figure 4.4 shows the state of the network after parsing two different triangles. The system correctly classified the left triangle as being equilateral and the right one as not being equilateral. The outputs of the binding networks show the vertex coordinates that were discovered by the network. Figure 4.5 shows a series of images from our simulator at several stages of a typical recognition sequence.

4.2 Discussion

Determining the relative locations of objects is a basic visual computation and shows up in numerous tasks. These relations are inefficiently represented in a feed forward network. The point of this chapter has been to demonstrate that with an attentional mechanism, one can efficiently extract

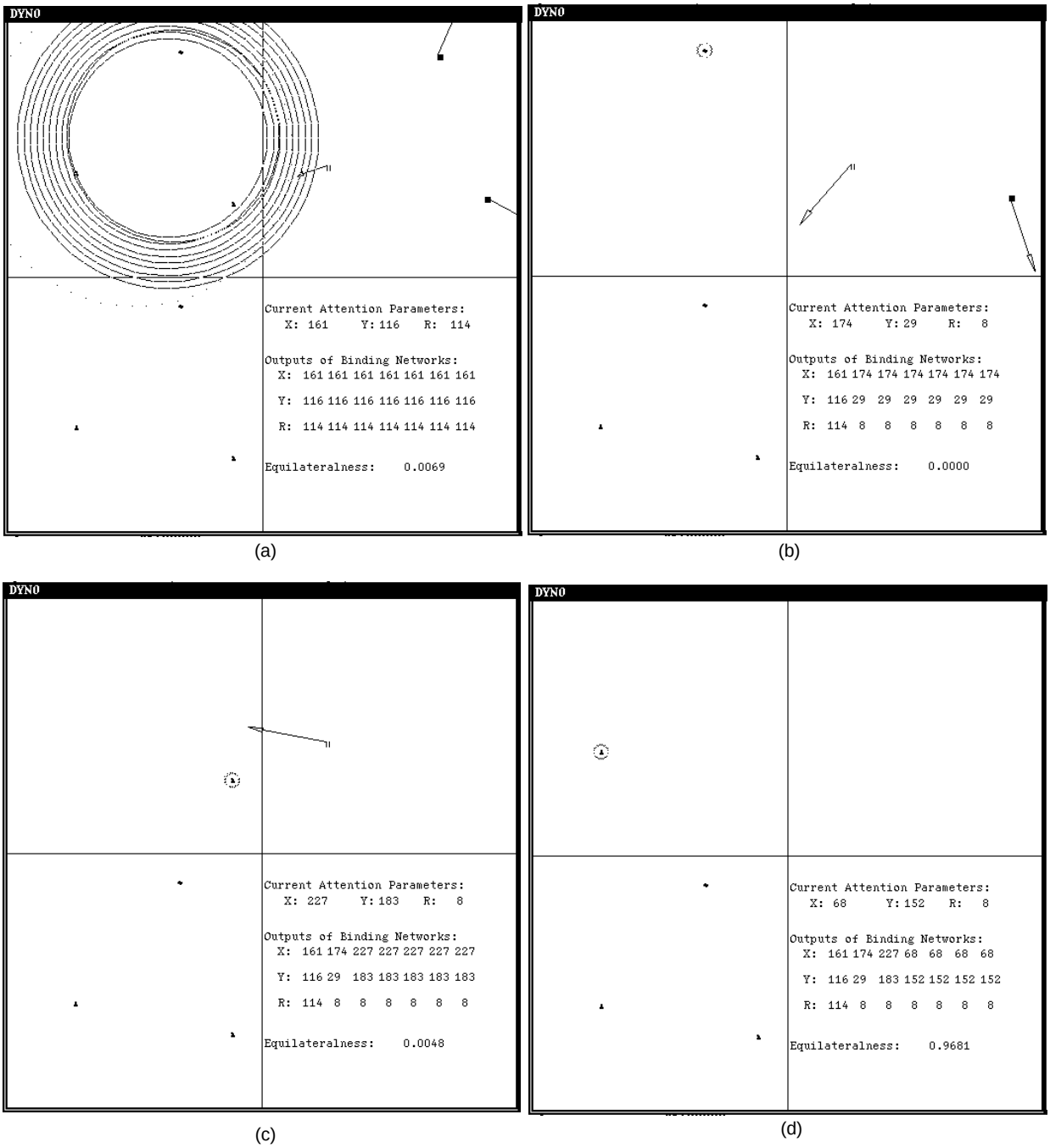


Figure 4.5. Four steps in a typical run. (a) Determining the size and location of the whole triangle. (b) - (d) Network after fixating on each of the three vertices. Note that the binding networks are updated correctly. Once all four positions are available, the network correctly classifies the triangle to be equilateral (bottom right of (d)).

such relations. The solution presented here is efficient in three different ways. The number of weights used to solve the task is linear in the image size. The number of sequential steps is on the order of the number of vertices and independent of the image size. Finally, the number of training exemplars necessary to learn the function is very small and is again independent of the image size.

5. Visual Search

What is the best way to search for an object in a cluttered scene? Consider images of the type shown in Figure 5.1. The goal is to search for a target object specified by a combination of features (e.g. a shaded horizontal figure). The task is a natural one for people, and for the last two decades psychologists have studied visual search in detail. Their experiments have shown that even in relatively simple scenes, some form of visual attention is required for the task. In fact, a significant fraction of the psychological knowledge on attention comes from visual search experiments.

There are good computational arguments to support this evidence. Suppose the image contains one object: a red horizontal bar. It is easy to construct a parallel network which detects this bars' features. With a setup as in Figure 5.2, the "red" and "horizontal" feature units will be activated at the

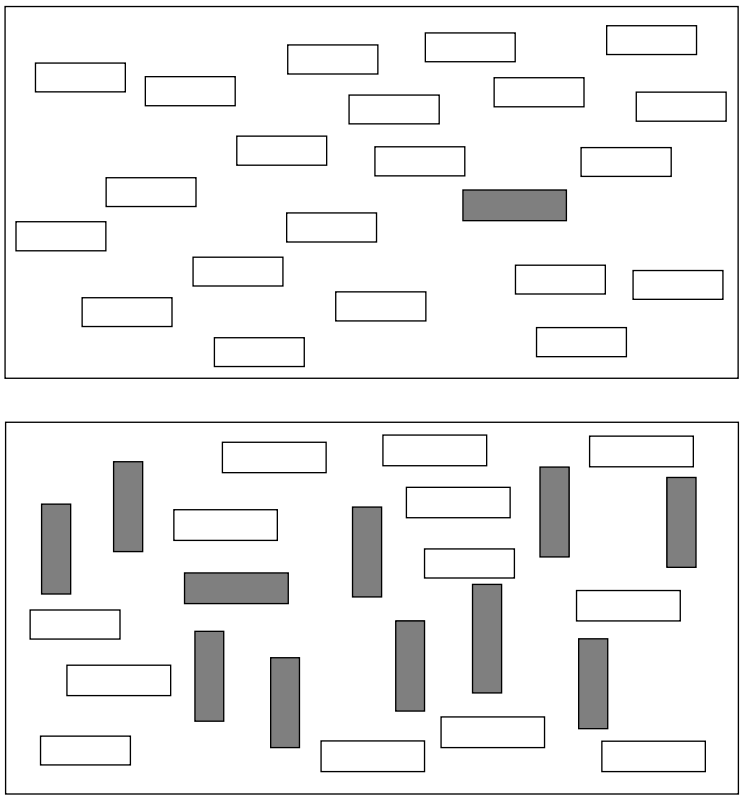


Figure 5.1. Sample images for visual search. In both figures, the task is to detect the shaded horizontal rectangle.

desired locations. We simply add one output unit per feature map. Each output unit computes a global OR of the activity in the corresponding feature map. Since there is a single object, any activity in the feature map must correspond to properties of that object. The values of the output units must correspond to the features of the object.

It is also possible to recover the location of a single object in a feed-forward manner. We can simply use the center of mass computation that was used in Chapter 4. The scheme is simple, efficient, and works well when there is a single object.

The above methods break down for scenes containing multiple objects since features from different objects can interfere with each other. An image containing one red object and one horizontal object would cause the global red and horizontal detectors to both light up. For example, both Figure 5.3(a) and Figure 5.3(b) would produce the same activity. The central problem is the inability to bind the features "red" and "horizontal" to the same location. Locations are also difficult to compute due to the crosstalk between two objects. The center of mass network would simply compute the center of mass of the whole scene.

A partial solution might include a separate detector for every possible combination of features at every location. In addition to the basic feature detectors, every location would also contain a blue-horizontal detector, a red-horizontal detector, and so on. A global OR of all the red-horizontal detectors would correctly signal the presence of a red-horizontal object. This scheme has a major

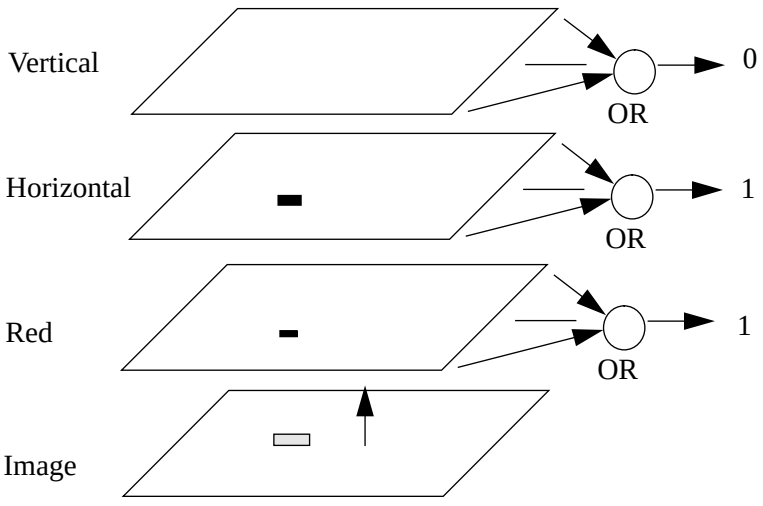
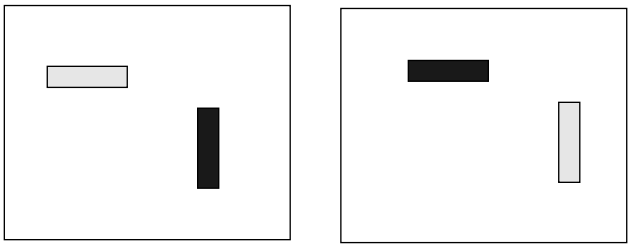


Figure 5.2. Feature maps with global OR's.



(a) (b)

Figure 5.3. Both figures would induce the same activity on the global OR units.

problem - the number of units grows combinatorially. For more complex search tasks, such as determining whether a red object appears next to a blue object, the number of units quickly become infeasible. A second problem with this scheme is that one still cannot recover the locations of individual objects.

Using visual attention, the natural solution is to temporarily inhibit all the activity in the image except at a single object. In this way, the single object method described above will work. Global red and horizontal detectors are now sufficient to detect feature conjunctions since only one object is active at a time. The cluster detection scheme can also be used to compute the object's location. Such a system would attend to each object in turn until the red and horizontal detectors both fire.

5.1 Augmenting VISIT for Visual Search

Several refinements to VISIT are necessary to perform visual search. These are discussed in the following sections. Figure 5.4 shows an overview of the network. There are two separate pathways from the image. The left pathway contains the gating network and makes decisions local to each object. The right pathway prioritizes image locations using top down information about the target and bottom-up information from the image. The control network mediates the information flow between the two. In the following sections we discuss each of the components in detail.

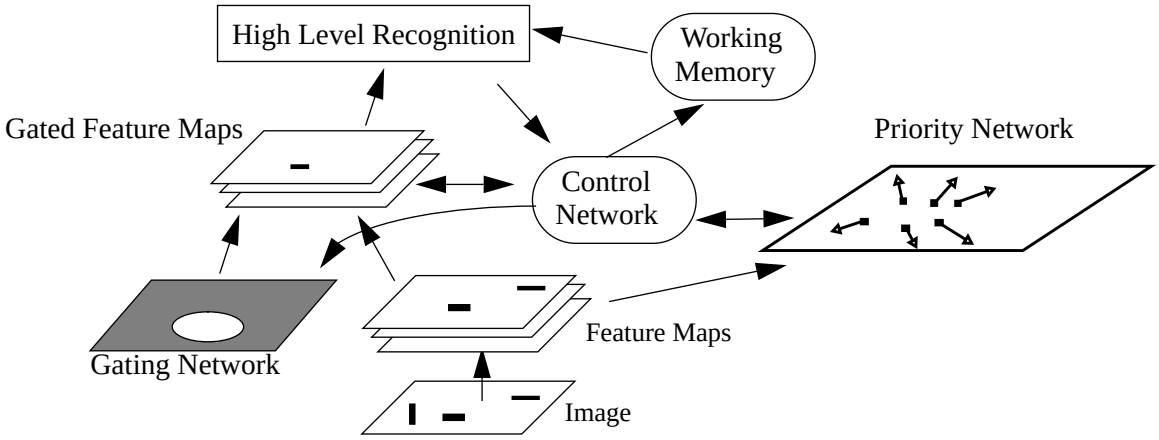


Figure 5.4. An overview of VISIT for visual search.

5.1.1 The Feature Maps

Feature maps in VISIT correspond to the topographic maps computed early in the visual system. A set of basic features (orientation, color, etc.) are detected for every location in the image in parallel (Figure 5.5). There is one unit for each feature at every location, organized topographically in layers. The activity of each unit within the layer represents the presence of a basic feature in the image at the corresponding image location. The activity of nearby units represents the presence of the feature at nearby locations. Each feature detector transmits its signal to the gating network and the priority network (described below). We also include a unit for each feature that computes a global sum of the corresponding map (Figure 5.2). In the current implementation of VISIT, two colors (red, blue) and two orientations (horizontal, vertical) are represented by feature maps. The issue of exactly which features should be computed in this way is an active area of research. For our purposes, any local feature can be included. (See Section 5.3 for more discussion on this topic.)

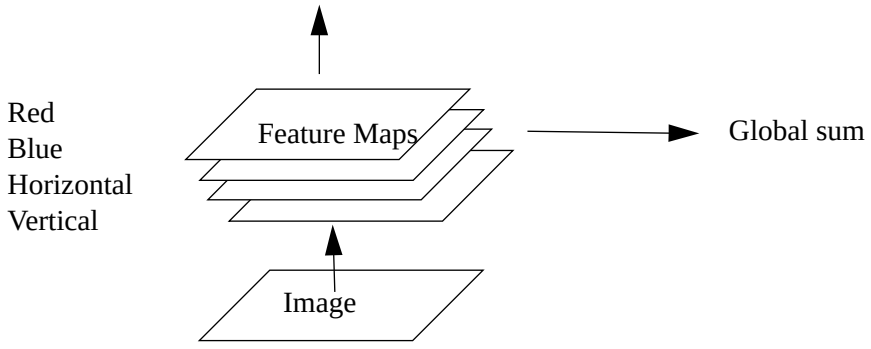


Figure 5.5. Four features are computed from the image.

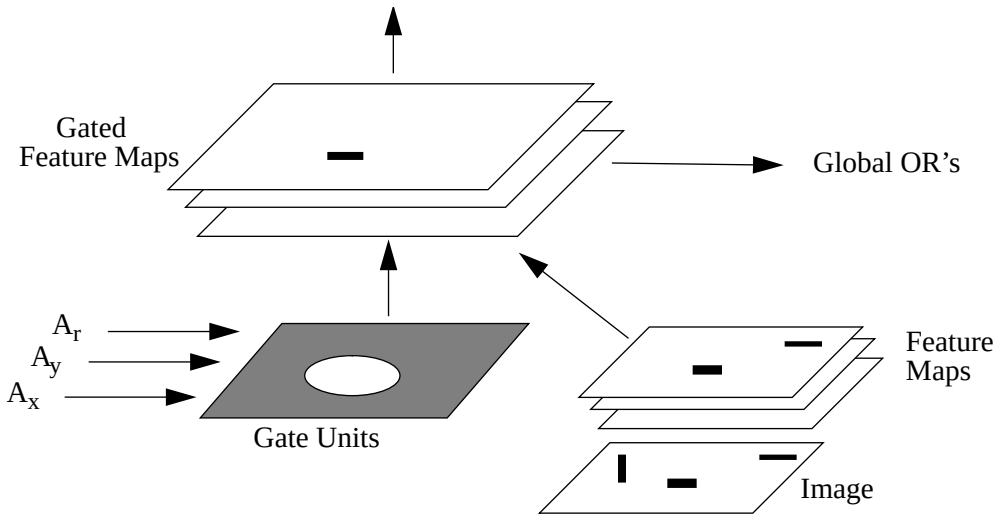


Figure 5.6. Gated feature maps.

5.1.2 The Gated Feature Maps

To recognize the individual objects, VISIT must bind the presence of features with one image location. One possibility would be to inhibit the feature maps themselves: given *(x,y,r)*, only the feature units within the circle should be active. There is a problem with this scheme: when attention is narrowly focussed, most of the image will be inhibited. As a result, the system will be unable to make global decisions based on the features. This facility is critical in VISIT for efficient search. All we really need to inhibit is the transmission of features to the recognition stage, not the entire feature map. We accomplish this by including a gated feature map for each primitive feature. Each unit within a gated feature map receives activation from the corresponding feature detector and inhibition from a gate unit (Figure 5.6). The net effect is that if the gated feature units fall within the circle they mimic the output of the feature detectors. Otherwise they are inhibited.

As with regular feature maps, the network computes a global OR for each gated feature map. Thus, to check whether an object has a certain feature combination, the network can focus on that location and check the global OR units of the corresponding gated feature maps. Using this configuration the network can efficiently access both local and global feature information simultaneously. There is some psychological evidence to support this. Even when attention is highly focused, people are able to report primitive features of objects appearing outside the focus of attention, but not combinations of features (Rock *et. al*, 1990).

5.1.3 The Priority Network

The job of the priority network is to rank image locations in order of relevance. As with the equilateral triangle network, a coarse coded error map encodes the saliency of image locations as well as an error vector for updating the focus of attention. As the focus of attention shifts to successive locations, the corresponding error units are inhibited. This prevents the focus of attention from visiting the same location twice.[1]

The main enhancement lies in the way priorities are assigned. In the previous chapter larger point clusters were always given higher priority than small ones. This was sufficient for sparse dot-images, but for cluttered, realistic scenes this simple model will result in very inefficient search sequences. Several psychophysical experiments have pointed out other possibilities. (Yantis & Jonides, 1990) have shown that stimuli which appear abruptly are attended to sooner than persistent stimuli. This can be overriden by explicitly instructing the subject to concentrate on a particular location. Experiments on visual search by (Egeth *et al*, 1984) suggest that objects with the same features or form as the target object can get higher priority than other objects. (See Chapter 6 for more detail.) All of this suggests a much more dynamic and flexible priority system than one which simply ranks the locations based on pixel density.

This sort of flexibility can be incorporated into the priority and control networks without any loss of efficiency. Within the priority network, we allow each feature map to have an independent weight. This value represents the importance of the feature map to the current task and can be dynamically adjusted by the control network (see below). The weights are represented as the activations of a set of units. The saliency at location *(x,y)*, S_{xy}, is computed as the weighted sum:

$$S_{xy} = \sum_{f \in F} W_f A_{fxy} \tag{5.1}$$

where W_f is the weight assigned to feature *f*, and A_{fxy} is the activation of the feature unit at location *(x,y)*. A coarse coded saliency unit, *i*, in the error map computes its output as:

$$A_i = G\left(\sum_{x,\, y \in RF_i} S_{xy} \right) \tag{5.2}$$

RF_i denotes the receptive field of unit *i*, and *F* is a monotonically increasing function of its input (in our implementation we use a sigmoid). When W_f is *0*, feature *f* has no effect on the locations which are attended. This allows the system to completely shut off the effect of any feature map in parallel.

1. In a continually running system, the inhibition should decay over time so that the system can return to the location later. In our system we simply reset the inhibition after each search.

5.1.4 The Working Memory

The working memory is augmented to store the features of the target object once it is selected. These features are used to set the priority levels (see below) and for recognition. Each set of binding networks now store the four feature values as well as the location. VISIT stops when the global OR values of the current gated feature maps match these stored values exactly.

5.1.5 SWIFT: A Strategy for Setting the Priority Levels

The main difference in the control network is the sub-system, named SWIFT, which controls the search process. The main function of SWIFT is to integrate top-down and bottom-up knowledge to efficiently guide the search process. Since we are performing directed visual search, the network can use previously stored information about the target object to prune the search space. Specifically, we rely on the observation that the desired object must contain *all* the features of the target object. Let F_T be the set of features attributed to the target object. Using the ability to weight feature maps differently, the SWIFT network can, in parallel, remove the influence of all but one of the features *f'* in F_T. By setting $w_{f'}$ to *1*, and all others to *0*, the system will only visit those locations which contain this feature. (Hence the name SWIFT: Search WIth Features Thrown out.)[1]

We use a second observation to select *f'*: since we are free to choose any feature in F_T as *f'*, we should set *f'* to be that feature which corresponds to the least number of objects, thus minimizing search time. In the current simulation we use boolean feature maps, so the map with the minimal total activity is most likely to contain the smallest number of objects. (If real valued features are used, outputs should be thresholded before totaling.)

SWIFT, then, goes through the following sequence in guiding the search. When presented with the target object, the network first stores all the features which belong to it. Once the image is presented, the total activity of all the feature maps are computed in parallel. The system then chooses *f'* and sets $w_{f'}$ to be *1* and all others to be *0*. Search then proceeds by sequentially visiting locations in order of their saliency. As the focus of attention stabilizes on each location, an independent network checks the features of the current object against the stored target representation. This continues until a match is found or there are no more active error units.

1. SWIFT was inspired by the experiments in (Egeth *et. al*, 1984). See Chapter 7.

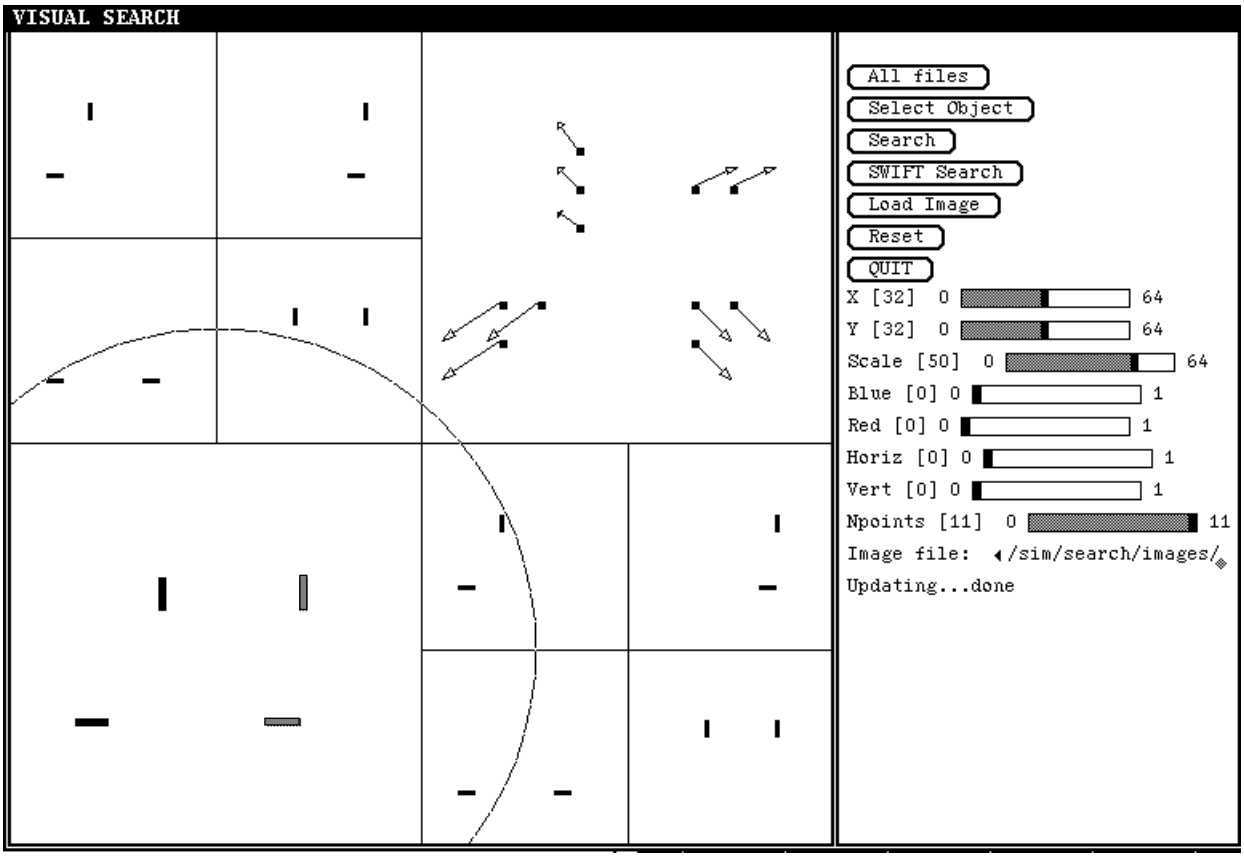


Figure 5.7. Output of the simulator.

5.2 Network Simulations

In this section we discuss the performance of VISIT on search tasks. Figure 5.7 shows the output of our simulator. The graphics display on the left shows various portions of the network. The bottom left window depicts the image. Black regions indicate blue objects and the shaded regions indicate red objects. The circle depicts the current focus of attention. In the figure, the system is currently attending to the whole image. In the current simulations, there are four features represented. The top left window shows the outputs of the four gated feature maps. In clockwise order from the top left they are: blue, red, horizontal, and vertical. When the focus of attention narrows (Figure 5.8), only those gated feature units within it will respond.

The priority network is shown in the right half of the display. The bottom right window depicts those feature maps which currently influence the priority network. In Figure 5.7 all the feature maps are currently affecting the priority network. The top right window shows the error map. As with the equilateral triangle network, each location in the error map contains two units for the error

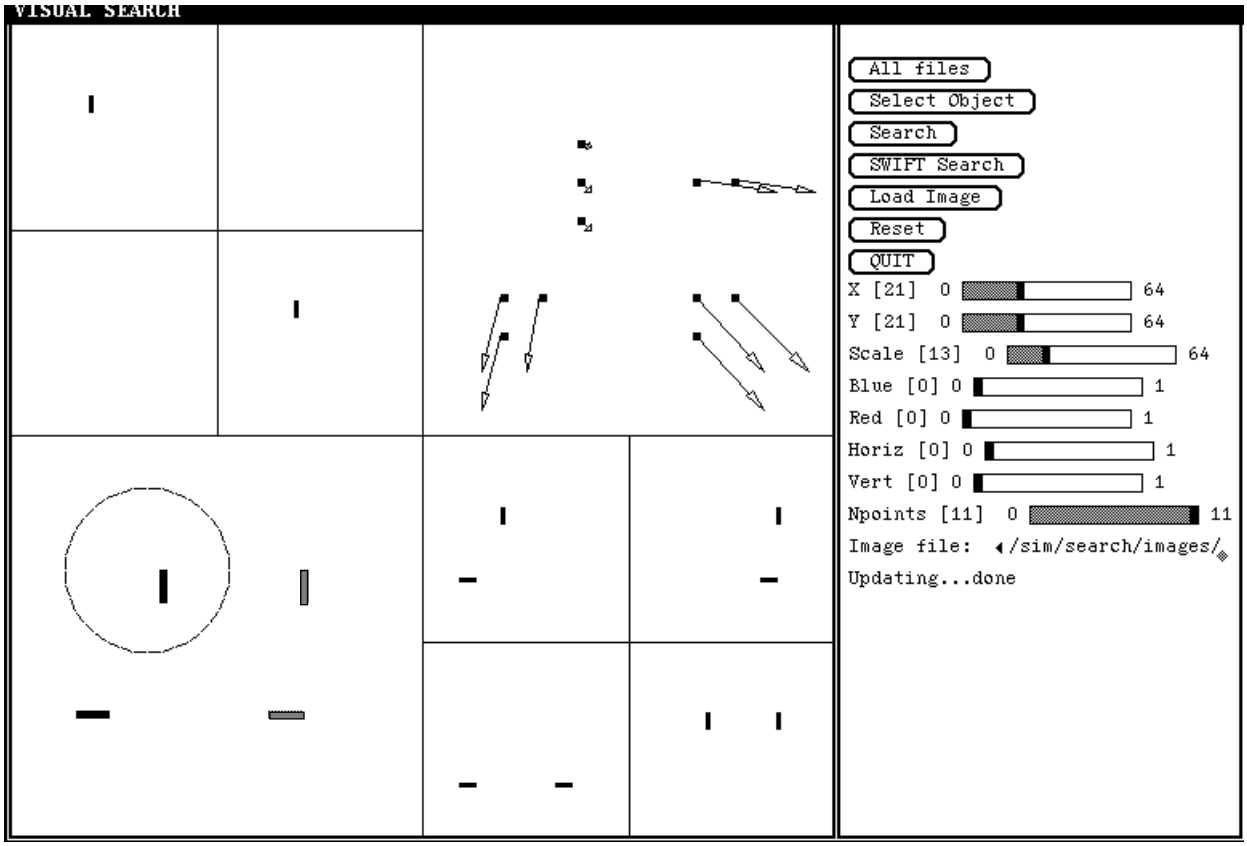


Figure 5.8. The initial display showing all possible target objects. The user is about to select the black vertical object as target.

vector and one unit for the saliency of the location. The arrows indicate the direction and magnitude of the error vector. In Figure 5.8 note that the error vectors have compensated for the change in focus of attention. The saliency is computed as described in Section 5.1.3 above. As the focus of attention visits each object, the saliency unit at that location is inhibited. (Only those locations whose saliency value is greater than .2 are shown in the display.)

The buttons on the right control various aspects of the simulation, such as loading images, stepping through the simulation, etc. The three sliders allow the user to explicitly set Ax,Ay, and Ar using the mouse.

When first started up, the system displays an image containing an instance of each object. The user narrows the focus of attention to the desired target object using the sliders. The gated OR's now correspond to its features. The user then clicks on the "Select Object" button which causes these values to be stored in working memory. During a search, when the network attends to an object, the values of the memory units are compared to the outputs of the gated OR's. If they match, then the target has been found, otherwise the network continues searching until there are no more can-

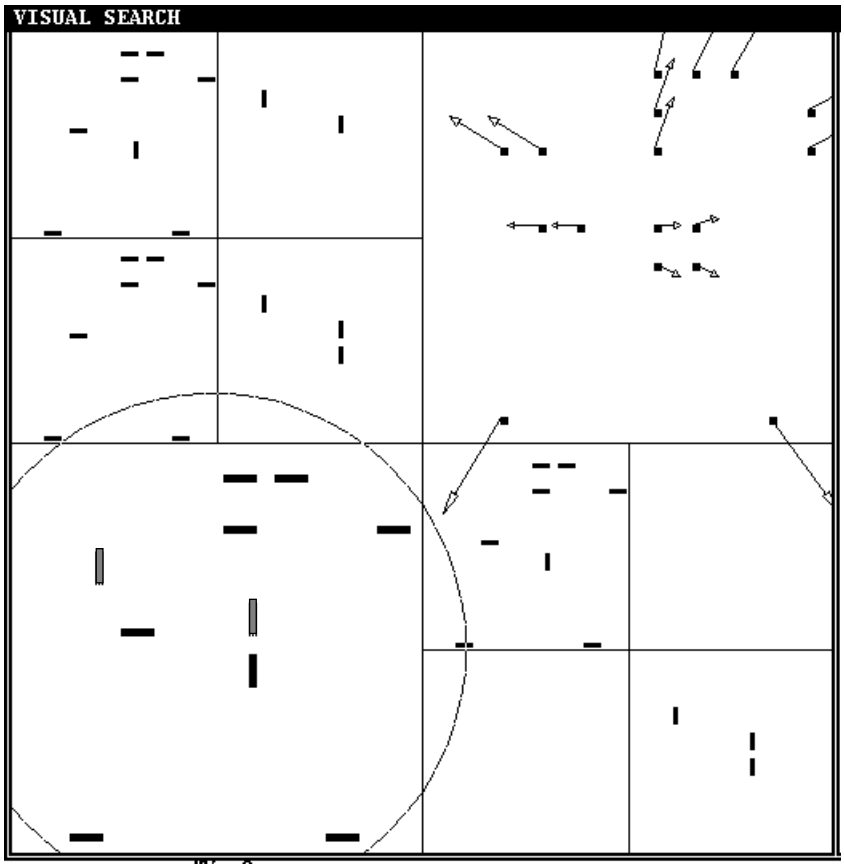


Figure 5.9. Start of the search.

didate objects.

5.2.1 An Example Search

Figure 5.9, Figure 5.10, and Figure 5.11 demonstrate a sample search sequence for a blue vertical item. Once the target object has been selected, the red and horizontal feature maps do not affect the priority network (Figure 5.9). Once the search starts, only the vertical feature map continues to influence the priority network, since there are fewer vertical objects than blue objects (Figure 5.10). In Figure 5.11 the target has been detected.

5.2.2 Computing Search Time

Note that, with SWIFT, search times do not depend on the *total* number of objects. Since SWIFT always searches the minimal feature map, the critical variable, M, that determines search time is:

$$M = \min_{f \in T} \{ O(f) \} \tag{5.3}$$

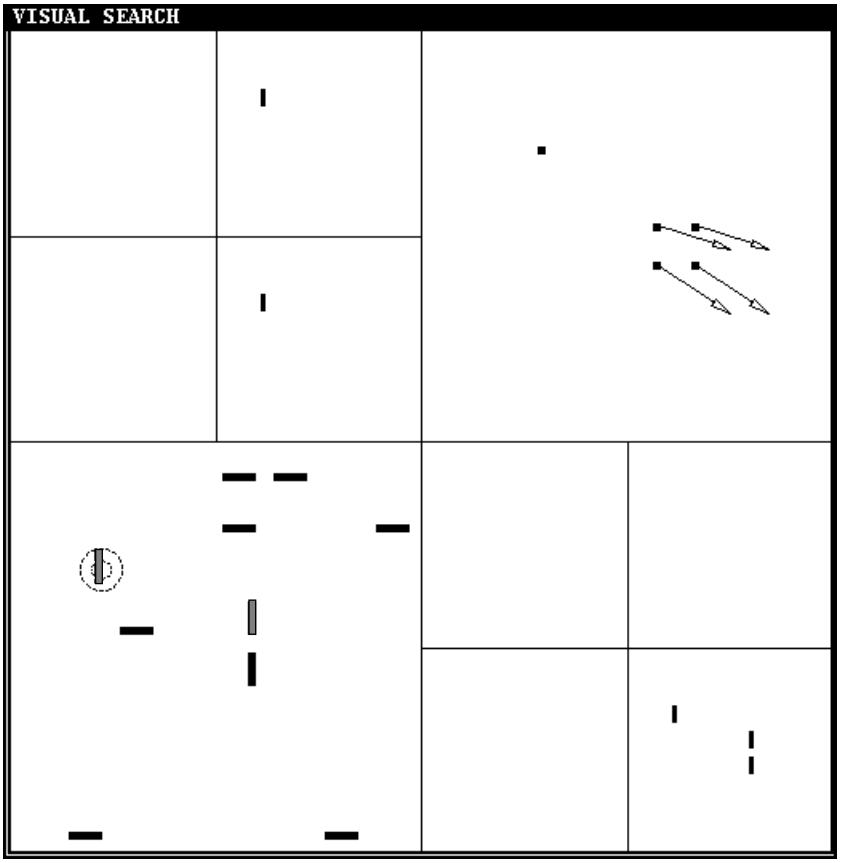


Figure 5.10. A step in the search sequence.

where *f* ranges over all the target object's features, and *O(f)* is the number of objects with feature *f*. In our architecture, search will always be linear in *M*. Note that the above equation correctly predicts the time required for the original single and conjunctive feature searches. In conjunctive feature searches (Treisman & Gelade, 1980), if the features of the distractors are chosen randomly, on average $M = \frac{1}{2}D$, thus the model predicts search time that is linear in *D*. The interesting case occurs during a conjunction search when *M* is not related to *D*, in which case search time will not be linear in *D*. Figure 5.12 and Figure 5.13 illustrate this. In the four images in Figure 5.12, *D* remains constant but *M* increases gradually. In Figure 5.14, *M* remains constant but *D* increases gradually. SWIFT predicts that in the first situation reaction time should increase linearly whereas it should remain constant in the second case.

Figure 5.14 and Figure 5.15 plot the average search time (number of fixations per image averaged over several trials) for various combinations of *M* and *D*. In Figure 5.14, the number of distractors is fixed at 40 as *M* is gradually increased. As expected, mean search time increases linearly, with an approximately 2:1 ratio in the slopes. In Figure 5.15, the graphs show that search time can remain relatively flat as *D* increases, as long as M is held constant. Again, search times for images

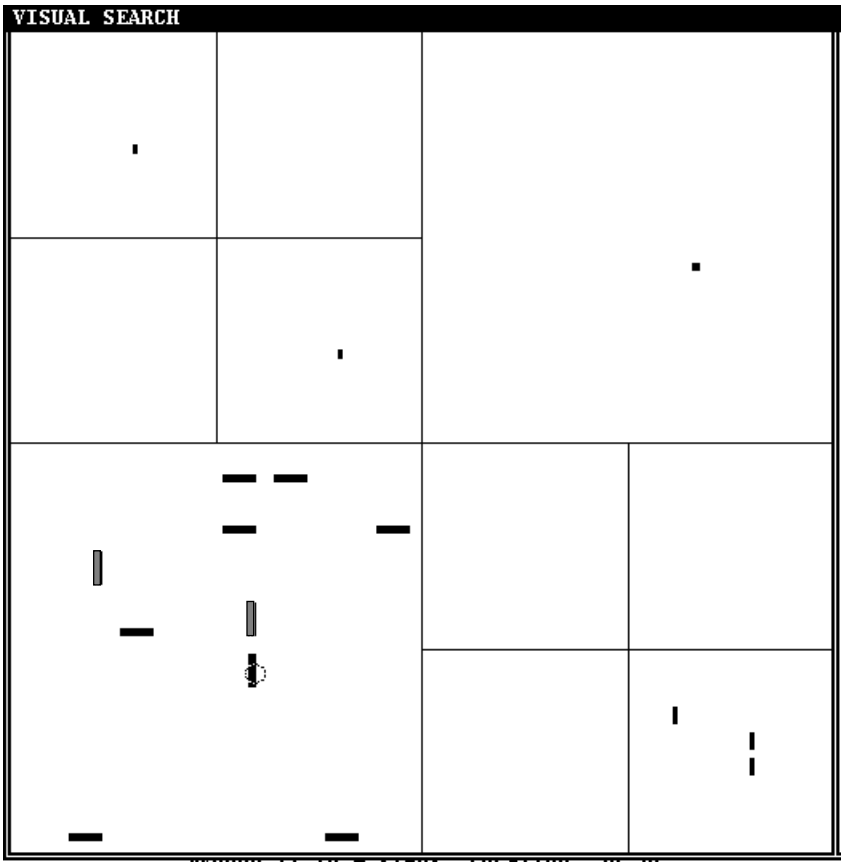


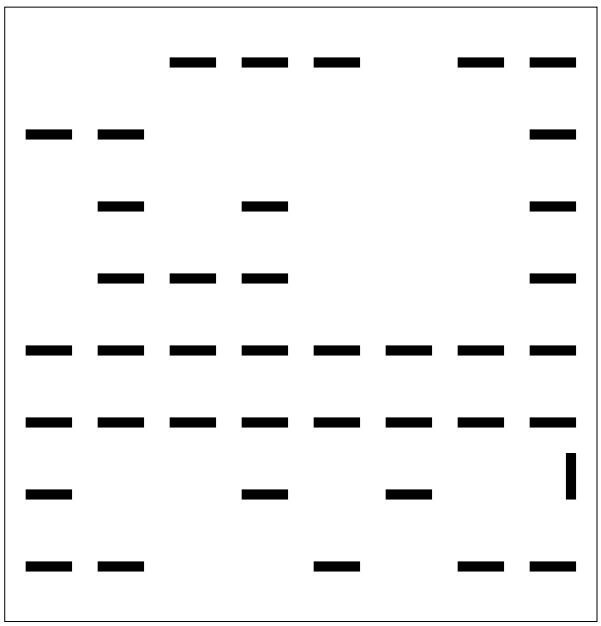
Figure 5.11. The final step in the search sequence.

with targets are approximately half that of images with no targets.

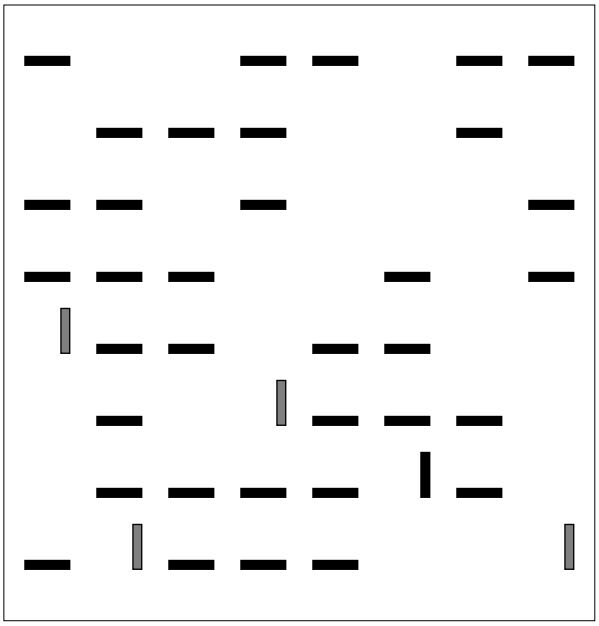
5.3 Optimizing Visual Search

5.3.1 Optimal Features for Visual Search

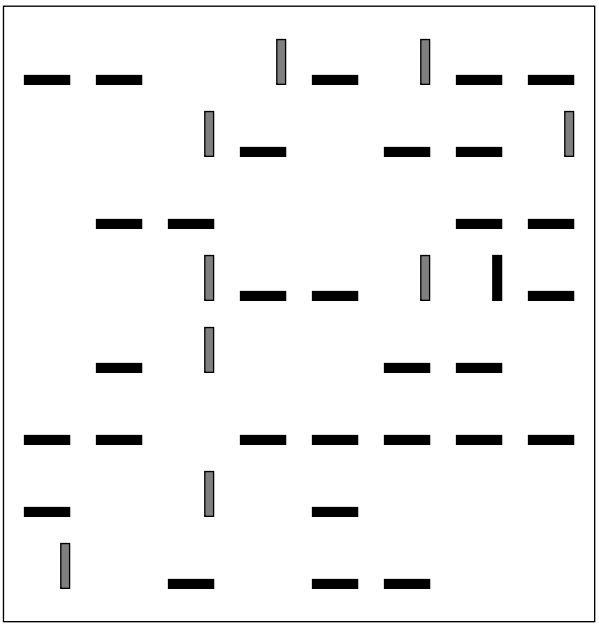
What are the best set of features for visual search? If SWIFT is used as a constraint, then we want the set of features that minimize *M* over all possible images and target objects, i.e. that best discriminate objects. It is easy to see that the optimal set of features should be maximally uncorrelated and that the distribution of feature values should be uniform over the space of possible objects. In other words, the optimal features should be the principal components of the distribution of images.



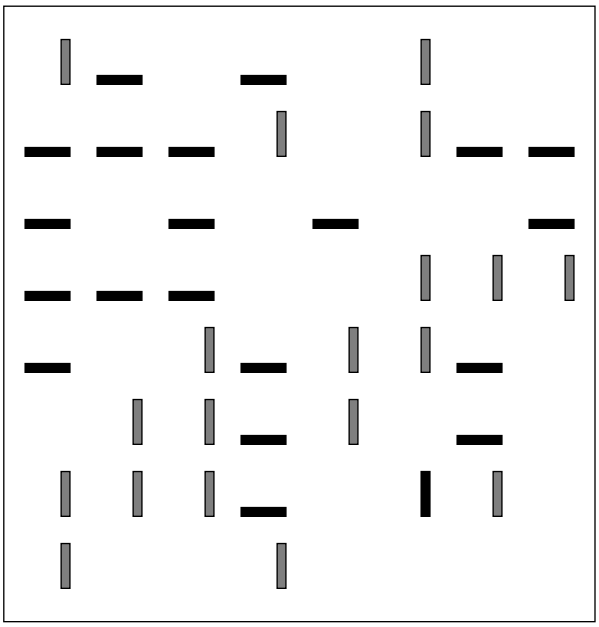
(a)



(b)

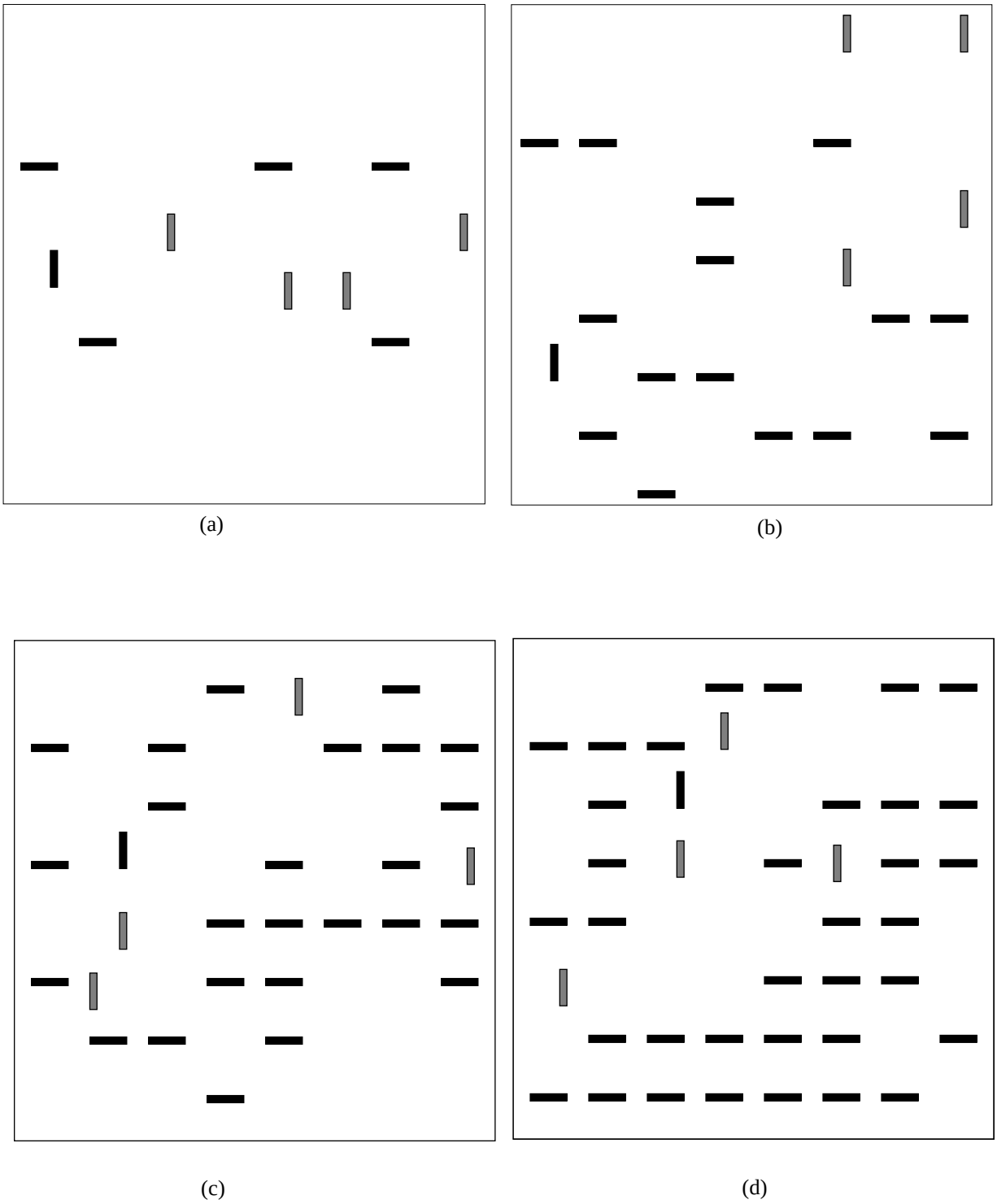


(c)



(d)

Figure 5.12. The target object in the above images is a black vertical rectangle. SWIFT predicts a linear increase in search time from image (a) M=1, (b) M=5, (c) M=10, to (d) M=20. The total number of objects is the same in all figures.



(a) (b)

(c) (d)

Figure 5.13. The target object is again a black vertical object. In (a)-(d), M remains constant at 5 but the total number of objects increases from 10 to 40. Target detection times should remain constant.

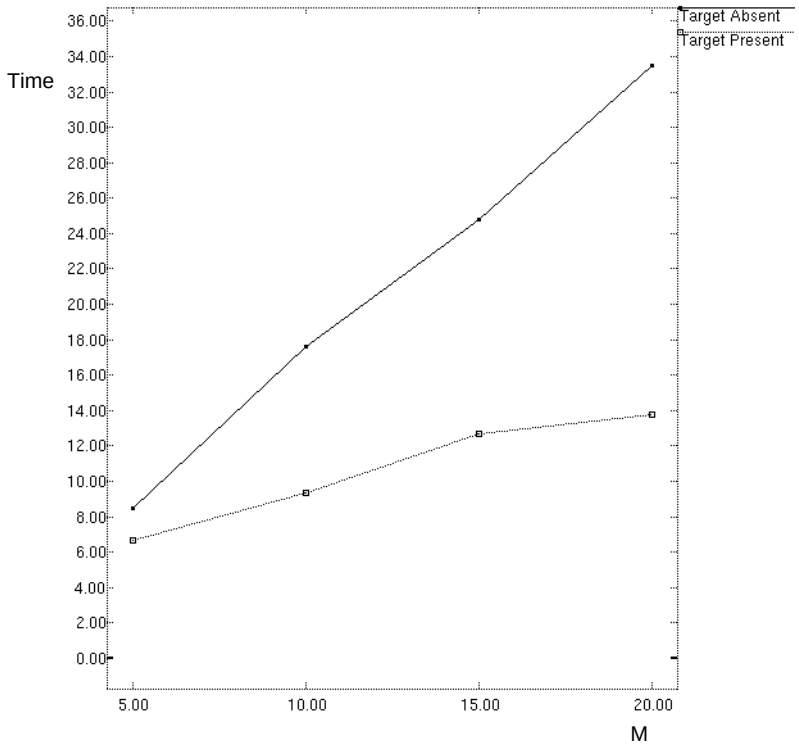


Figure 5.14. Search time vs M, with D=40.

It is interesting to note that a single Hebb neuron extracts the largest principal component of the input distribution and with inhibition, sets of Hebbian neurons can extract successively smaller components (Becker, 1991). Moreover, as some researchers have demonstrated (e.g. Linsker 1989), simple Hebbian learning can lead to features that look very similar to the features in the visual cortex. If the early features in visual cortex are in fact the principal components, then SWIFT is a simple strategy that takes advantage of it.

5.3.2 Detecting Feature Combinations in Constant Time

It turns out that, in this architecture, it is possible to implement a strategy which detects some conjunctions of features in constant time. The way to do this is simple: if the task is to search for a blue-horizontal feature, inhibit all the maps except blue and horizontal. Let the activity of the saliency units increase monotonically as the sum of all the active feature detectors within its receptive field. Then, the location with the highest activity will be the one with both blue and horizontal detectors active. Thus, if search proceeds according to highest saliency, the network should always discover the blue-horizontal object first, regardless of the distractors. This does not imply however that the binding problem can be solved in a feed-forward network. The above method still requires

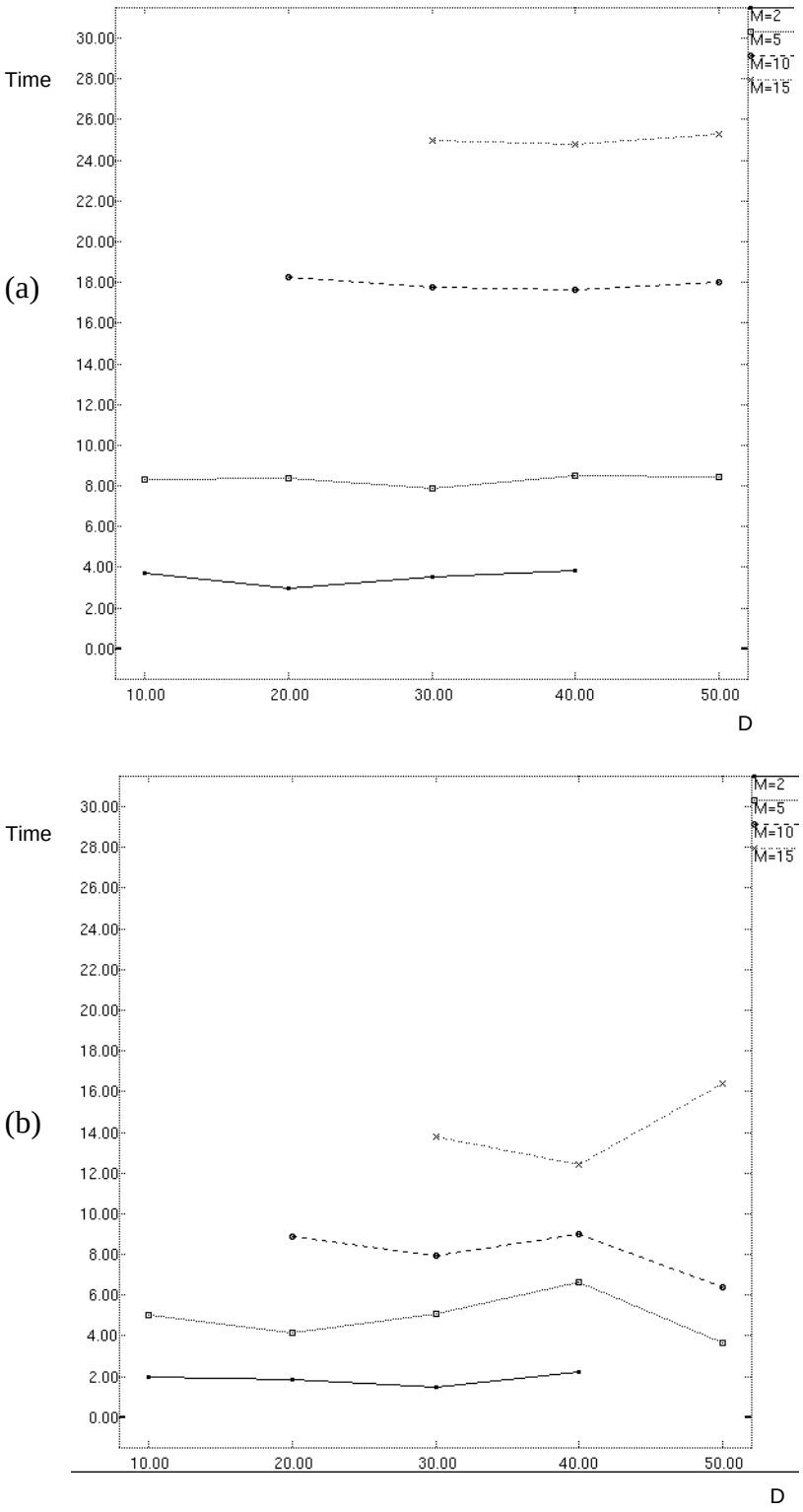
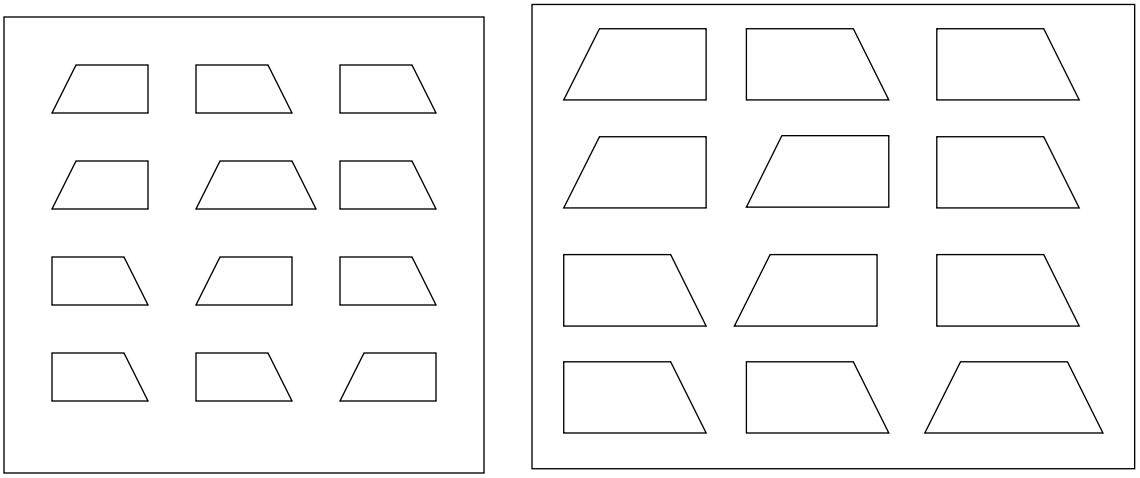


Figure 5.15. Search time vs D for various values of M. (a) Target absent. (b) Target present.



(a) (b)

Figure 5.16. An example of conjunction search where the two features are not nearby. The target is the trapezoid with two oriented sides. In this case, one can arbitrarily increase the distance between the two features (the orientation of the left and right hand sides) by increasing the size of the object.

top-down feedback to inhibit the irrelevant feature maps and still requires selective attention to recover the object's location. In any case, it seems possible to have an efficient, implementable scheme which detects feature combinations in constant time.

A problem with the above scheme is that it is not very general. It does not work if the relevant object features are not physically close together. For example, consider Figure 5.16. The target is the trapezoid with both sides at an angle. The distractors either have the left or right side at an angle. In this example, the distance between the relevant features may be arbitrarily increased. A strategy which tried to determine the co-ocurrence of the two features in parallel, would need to know the feature separation ahead of time. This can get to be quite a complex task. Note that in the figure, the separation between the objects can be smaller than the separation between the features.

5.4 Discussion

VISIT is an implemented model and makes several strong predictions about human visual attention and visual search. These predictions could be easily tested by appropriate experiments.

The model describes precisely how long search should take in different circumstances. In particular, when a target has at least one unique basic feature (when compared to the distractors), then search time should be constant. It doesn't matter whether the search is for a conjunction of several features or for a single feature. For example searching for a blue-vertical object should always take constant time if the distractors are all red and green horizontal objects. When the target does share features with the distractors, then the search time should be linear in M, the number of objects which share the least active feature in the image. Experiments which control M and D should be able to determine whether people also use the same strategy.

To my knowledge the above experiment has not been carried out. However VISIT plus SWIFT can help account for a large portion of the existing experimental results on attention and visual search. In the next chapter we discuss this and other relationships with human attention.

6. Visual Attention In People: Implementation

Much of VISIT was inspired by the psychological, and biological knowledge on attention. The next two chapters review some of this work and its relationship to VISIT. To facilitate comparison the text is divided into two broad topics: the mechanics of visual attention (how it is implemented), and its use.

6.1 Psychophysical Insights Into the Implementation

6.1.1 Evidence that Attention Exists

The first experimental evidence for covert visual attention was obtained from reaction time studies (for a review see Posner, Cohen, & Rafal, 1982). In a typical experiment, subjects are shown a display consisting of three squares. The task is to press a button as soon as the target object appears in one of the boxes and the reaction time of the subject is measured. In some of the trials, the target is preceded by a cue, such as one of the boxes becoming brighter (Figure 6.1). Valid cues predict target location with a probability of 0.8; invalid cues do so with probability 0.2. The basic result is that reaction time is significantly faster in trials consisting of a valid cue than in trials with no cue.

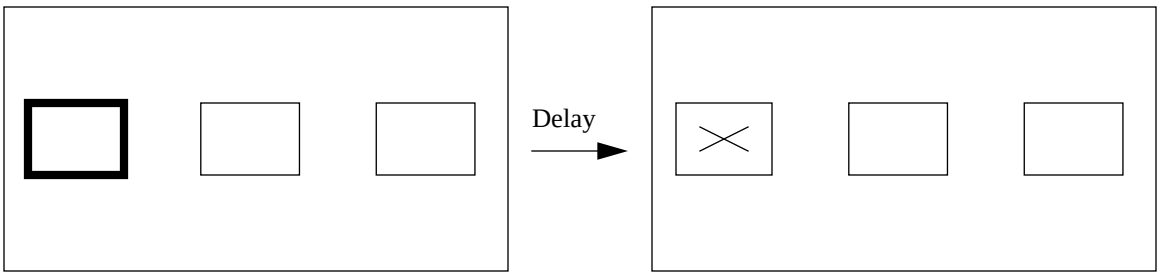


Figure 6.1. The subject is asked to press a button as soon as the target (in this case an “X”) appears in one of the boxes. Response times are facilitated in trials where a predictive cue appears (i.e. in the form of a darkened square) some time before the target.

Reaction time is significantly slower when the cue is invalid. Saccadic eye movements are ruled out in these experiments. This clearly demonstrates that an internal change in processing efficiency occurs and that the effect is spatially localized.

6.1.2 How Long Does Attention Take?

By varying the delay between the presentation of the cue and the target, (Posner, Cohen, & Rafal, 1982) were able to study the speed with which attention can work. With a delay of *0* msecs, there is no cued-side advantage, but changes in reaction time can begin as soon as *50* msecs after cue presentation. By *150* msecs, there is a marked improvement in efficiency. (Julesz & Bergen, 1987) have reported similar figures - they claim that attention only takes about *40* msecs. In tasks that require multiple shifts of attention, such as visual search, figures as low as *20-30* msecs have been reported (Treisman, 1988). Given that individual neurons can only fire once every *5-10* msecs, this leaves at most *4-10* sequential steps per shift. These values are quite astonishing considering that each shift entails disengaging attention from the current location, deciding which location to visit next, and performing the actual shift. Given these temporal constraints, each of the operations is probably performed by separate parallel modules. In VISIT, once the location is selected, the shift itself takes about *2* or *3* time steps. During visual search, new locations are selected in parallel with the shifting process. The network's behavior therefore is quite consistent with the above timing results.

6.1.3 Determining which Location to Attend Next

One of the basic components of an attentional mechanism is the task of deciding which locations to visit. There are two possible sources of information which can affect this decision: bottom-up and top-down information. In this section I review existing knowledge about these different sources.

Bottom Up Processing

Bottom-up information refers to the information obtained solely from the image. The literature shows that several types of bottom up cues can attract attention. For example, an object containing a distinctive feature often attracts attention. In addition, changes in image features are often powerful cues for attracting attention. For example, in (Posner, Cohen, & Rafal, 1982), both increases or decreases in edge intensity resulted in a cued-side advantage. (Yantis & Jonides, 1990) show that

stimuli appearing suddenly are more likely to attract attention than persistent stimuli. Follow-up experiments suggest that after a delay, stimuli which had appeared abruptly have the same priority as non-abrupt stimuli, suggestive of a priority value that decays with time (Yantis & Jones, 1990).

Top Down Processing

Top down information can also have a large influence on attended locations. In (Posner, Cohen, & Rafal, 1982), a high-level cue (such as a central arrow indicating direction) could serve to attract attention. In addition, if a cue appears consistently on the wrong side, the subject learns through experience to shift attention automatically to the opposite box (this takes about 300 msecs). The experiments in (Egeth, Virzi, & Garbart, 1984) suggest that attention can be restricted to objects containing a particular feature if the subject is told about the feature in advance. (Yantis & Jonides, 1990) provide further evidence along these lines. They report that, although abrupt stimuli seem to have higher priority over non-abrupt stimuli, this can be overridden if the subject is already focused on another task.

Integrating Top Down and Bottom Up Information

These experiments together illustrate a highly flexible system for deciding on the next location. Both top-down and bottom-up information must play a role. (Yantis & Jonides, 1990) suggest a hierarchical priority system that assigns variable priorities to spatial locations. In their scheme, locations which are currently being attended to have higher priority than abrupt onsets which in turn have higher priority than non-onsets.

VISIT implements a generalized version of the above hypothesis. The model has the ability to include a default weighting of the feature maps. For example, the motion and abrupt onset maps could be given higher priority than static maps. As in SWIFT, these weights can be dynamically adjusted according to the situation. The actual priority levels can depend both on the target object and the image. If top-down location information is available, the control network is free to ignore the priority map. The default weights are used in the absence of other information. Currently there is no mechanism for decaying priority values, but it could be added easily.

An interesting twist to these effects is discussed by (Johnston, Farnham, & Hawley, 1991). There is a well-known phenomenon that subjects respond faster to familiar words than unfamiliar words. In these experiments, a set of four words are flashed briefly (durations ranged from 33 msecs to 200

msecs), masked, and a word is presented centrally; the subject's task is to identify the location in the first display of the cued word. It is assumed that the subject must attend to the word in the first display in order to accurately localize it. They find that subjects are most accurate at localizing the cued word if it is a unique novel word in the presence of familiar words, or a unique familiar word in the presence of novel words. This effect was found even at the extremely brief duration of 33 msecs! These results suggest that a SWIFT type search strategy might even play a role for very high-level features, such as "novelty" or "familiarity".

6.1.4 Inhibition of return

(Posner, Cohen & Rafal, 1982) have compared the efficiency of cued locations against other locations of equal eccentricity after initial cue onset. The results suggest that attention shifts temporarily decrease the priority of locations after the initial visit. Termed "inhibition of return", such a mechanism is clearly useful to prevent oscillations, such as continual shifts between the two highest priority locations.

Interestingly, this mechanism seems to be quite selective. Inhibition of return does not occur with central locations but seems to co-occur only with shifts to peripheral locations (Posner, Cohen & Rafal, 1982). Experiments reported in (Rafal *et. al*, 1989) show that the inhibition does not occur with endogenously generated shifts of attention (e.g. one induced by a high-level stimuli such as an arrow) unless the subject is about to make a saccade to that location. In fact, inhibition of return always seems to accompany a saccade. Experiments in (Klein, 1988) suggest that it is active during serial visual search.

Although the exact conditions under which inhibition of return operates is not completely known, overall it seems to be used in quite a sensible manner and only when the conditions require it. Clearly it is useful for visual search. Why should it be active so predominately with eye saccades? One answer is that saccades take much longer than covert attention (about 200-300 msecs), so the cost of executing a saccade is relatively high. Also, due to the high resolution of the fovea, once we have saccaded somewhere, we have a lot of information about that location. Hence there is almost never a need to attend to such a location immediately afterwards. In fact recent experiments indicate that the inhibition may even be relative to some external frame of reference and not the retinal frame (Rafal, personal communication). This would allow the mechanism to integrate smoothly with eye saccades.

VISIT currently implements a very straightforward inhibition of return, i.e. it always inhibits the

priority units during the shift. The above rules could easily be added to the control networks.

6.1.5 The Shape of the Focus

The exact size and shape of the focus of attention is a matter of extensive debate. Spatial theories of attention argue that attention operates on a single convex region in visual space. An example is the "zoom-lens" model (Eriksen & Yeh, 1985). Eriksen and Yeh suggest that only one region can be attended to at a time. This region can vary continuously in size, from being distributed over the entire visual field to being narrowly focussed on one object. Moreover, they propose that the number of "units of information" that can be processed per unit time remains constant. Like zoom lenses on cameras, there is an inverse relationship between the resolving power and the area of focus.

In contrast to spatial theories, object based theories of attention argue that attention is allocated to perceptual groups, or high-level objects, rather than a single spatial region. Thus the shape would be determined completely by the shape of the object that is attended. As evidence to support this, Lappin (1967) reports experiments where subjects are faster at reporting multiple properties belonging to the same object than to different objects. This is true even when the objects are overlapping (Duncan, 1984), a fact that cannot be explained by a purely location based theory.

Although a detailed review of the object-based theories is beyond the scope of this thesis, it should be pointed out that object-based theories and spatial theories of attention do not necessarily conflict with each other. It is possible to explain all the results by postulating attentional mechanisms at the level of both spatial representations as well as higher-order object representations (e.g. as in the object-file model (Kahneman, Treisman, & Gibbs, 1991)). The effectiveness of high-level cues in attracting attention (in the absence of any image cues) seems to strongly argue for the existence of some sort of a spatial mechanism. Recent experiments show that, with appropriate controls, both spatial proximity and grouping effects can be demonstrated (Kramer & Jacobson, 1991). The nature of the interaction between the two forms of attention is an interesting question and currently an open one.

6.1.6 Does Attention Move Continuously or Does it Jump?

Another area of controversy involves the question of how shifts of attention take place. (Shulman, Remington, & McLean, 1979) reports evidence that attention moves continuously across the visual

field. In their experiment subjects were to indicate when any one of 4 LED's lit up. An arrow pointing left or right would appear, indicating that either the far-left or the far-right LED had a high expectation of being lit. After 200-300 msecs, the responses to the far LED on the expected side were facilitated. The main result is that responses to the near LED on the expected side were also facilitated about 150-200 msecs after the cue appeared. This facilitation was reduced after about 300msecs. The authors suggest that the spotlight of attention starts at the location of fixation and moves in an analog fashion towards the next fixation.

In (Remington & Pierce, 1984) however, the authors present contradictory evidence. In this paper, they attempt to directly measure the time required to perform this shift of attention. They did this by varying the time between the cue-onset and stimulus-onset. Their central result is that the time seems to be constant and independent of the distance. These findings imply that, either attention does not move in a continuous fashion or that the speed of the shift somehow increases with the distance. Recently, Garvin Chastain (1991, personal communication) has provided further evidence that attention does not move in a continuous fashion. In his experiments subjects attended to stimuli which moved from one location to another. Response times to random probes throughout the process was used as a measure of response efficacy at different locations. The results suggest that during a shift, the efficacy gradually dies down at one location and simultaneously increases in the other. The average efficacy at any time remains approximately constant suggesting that attention does not move continuously from one spot to another.

In summary, the evidence seems to favor a focus that jumps from one spot to another, although it is by no means definitive. It is quite possible that both types exist. It is well known that eye movements have two forms: continuous pursuit when the eye is following a moving object, and jumpy saccades as when the eye fixates on a novel object (Wurtz & Goldberg, 1989). There has been recent preliminary evidence indicating that covert attention is also used in motion computations (see Section 7.2.4) so it is quite plausible that covert attention contains analogous smooth pursuit and saccadic movements. VISIT currently implements a focus that jumps, although the gating network is fast enough to implement either mechanism by appropriately controlling the attention parameters.

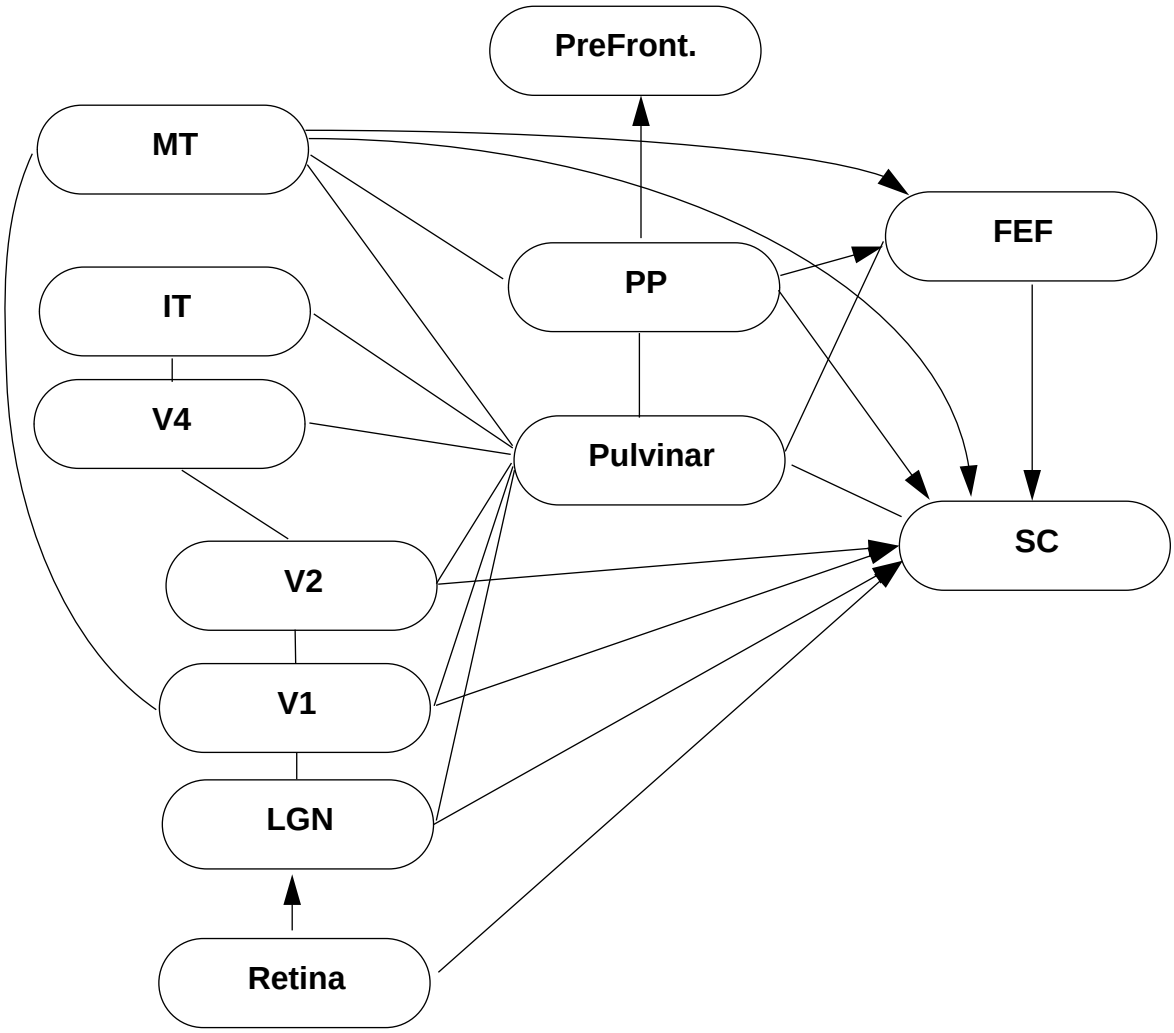


Figure 6.2. Some of the major connections between the various visual areas. Those connections without an arrow are known to be bi-directional.

6.2 Physiological Insights Into the Implementation

The last ten years have witnessed an explosion in the knowledge on the physiology of vision and visual attention. The following sections review some of this work and its relationship to VISIT[1].

After visual information is processed in the retina, it is routed to two different areas of the brain (Figure 6.5). Long axons from retinal ganglion cells form the optic nerve which terminates at the

1. A detailed review of the biology of vision is beyond the scope of this thesis. Two good books dealing with the topics covered here are (Spillman & Werner, 1990) and (Wurtz & Goldberg, 1989). For shorter reviews see (Van Essen & Anderson, 1990) and (Schiller, 1985).

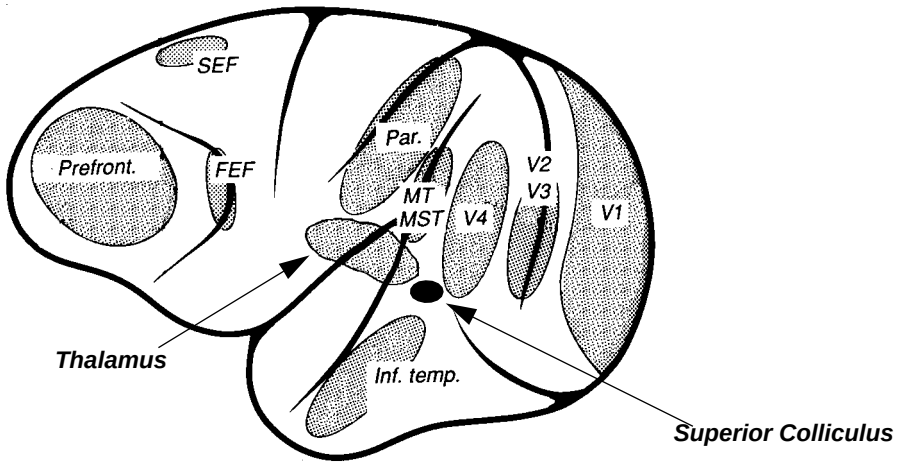


Figure 6.3. A schematic of the relative locations of the various visual areas. (Adapted from (Spillman & Werner, 1990)).

LGN. The optic radiation proceeds to the primary visual cortex. The LGN is part of the thalamus (Figure 6.3 and Figure 6.4) and is the primary target of retinal axons. There are approximately one million fibers in the optic nerve and about the same number of neurons in the LGN. There is a second path from the retina directly to the superior colliculus. This pathway plays an important role in orienting behaviors such as eye saccades and there is a good chance that it is also used for covert attention.

6.2.1 LGN, V1, and V2

Each of the areas LGN, V1, and V2 form a topographic map of retinal activity, although the details vary from area to area. The layers are connected in a roughly hierarchical fashion, with the receptive field sizes increasing as one moves up the hierarchy. It is not a strict feed-forward hierarchy, since there are pathways which skip layers (e.g. V1 to MT). Connections with other cortical areas also tend to be bi-directional and topographic. The targets of each neuron send back projections to the same neuron. In the LGN, some estimates place the number of returning fibers at about *10* times the number of outgoing fibers. Each cortical area also sends significant (reciprocal) projections to the pulvinar (see below).

These areas are analogous to the set of early feature maps in VISIT. Neurons sensitive to the orientation of stimuli, as well as cells selective to spatial frequency, velocity, binocular disparity, color, length, end-stops, curvature, texture, and motion have been found in both V1 and V2 (Hubel & Wiesel, 1968; Van Essen & Anderson, 1990). The complete set of features has yet to be determined. V1 contains about *1000* times as many neurons as the LGN. Since the LGN has a roughly

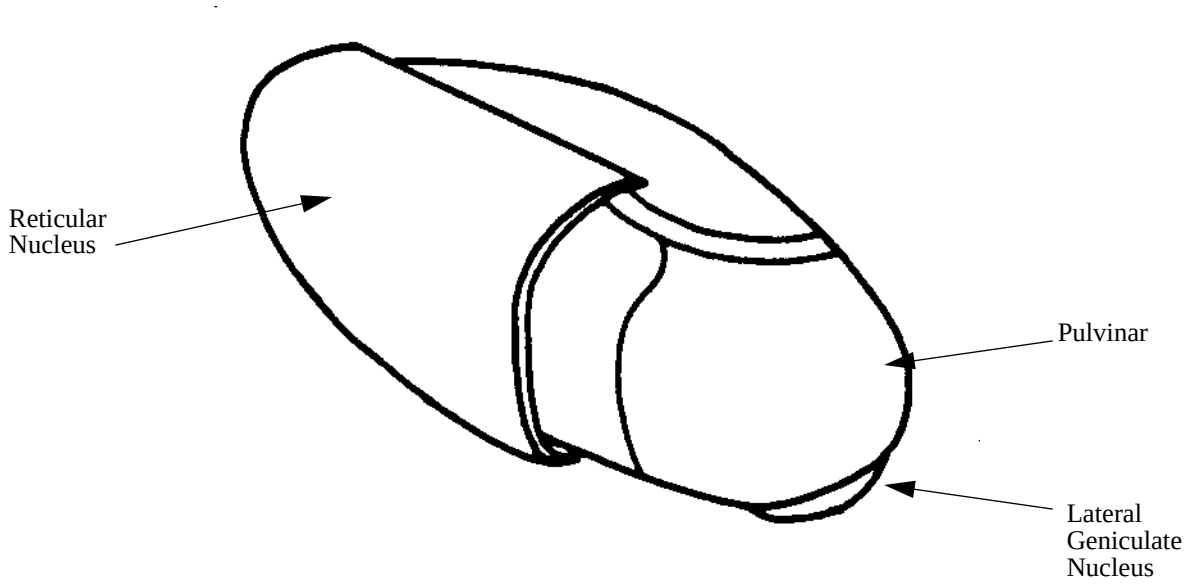


Figure 6.4. Enlarged view of the thalamus. (Adapted from (LaBerge, 1990)).

1:1 mapping with retinal ganglion cells, there may exist hundreds of different features maps in V1 alone. It is appealing to compare physiological properties of cells with the perceptual features studied by psychophysicists. Unfortunately the mapping between the two is not always clear. For example, some cells code for a combination of spatial frequency and orientation (De Valois & De Valois, 1988) which seems to be perceptually unnatural. There are also some discrepancies between psychological and physiological findings (Treisman & Sato, 1990). Nevertheless perceptual features are almost certainly derived from the responses of cells in V1 and V2. There are many correlations between perceptual features and V1 cell properties, so as a first approximation VISIT models the two as being identical.

There are a number of details which may be important in the computation of the early features but may not have much relevance to the attention process itself. VISIT does not depend on the existence feed-back links, lateral connections, magno vs. parvo cells, or an explicit feature hierarchy. The visual areas have many other complex tasks to perform - many of which may require these distinctions. Examples include stereo and motion computations, segmentation, and possibly mental imagery and learning. VISIT makes no commitments to the actual computation of the early features - the network structure can easily utilize more complex models incorporating the above features.

The structure of VISIT does impose the following restriction - focussed attention should not have any local effects on the early features. VISIT relies on the ability to simultaneously access both

global and local feature information. In particular, global information is critical in determining the location to shift to next. As far as I am aware, this restriction is consistent with the physiological knowledge. There have been some non-local eye-saccade related effects reported in V1 and V2 such as an overall increase in activity just before saccades (Goldberg & Segraves, 1989). No local attentional effects have yet been seen, although people have actively looked for it (Moran & Desimone, 1985);

6.2.2 Areas V4 and IT

In the traditional hierarchical view of the visual system, areas V4 and IT (inferotemporal cortex) would represent levels *4* and *5,* after LGN, V1, and V2. There is some evidence in favor of this viewpoint. In V4 and IT receptive field sizes continue to increase such that in IT each cell responds to more than half of the visual field. Furthermore, as predicted by the theory, cells in IT are much more selective in their preferred stimulus than cells in earlier areas (Desimone, 1991). For example, some cells respond only to hands (Gross *et. al*, 1972) whereas some respond only to faces (Bruce *et. al*, 1981), or even profiles of faces (Desimone *et. al*, 1984). However, contrary to the strictly feed-forward view, there exists strong evidence that spatial attention begins to play a major role in these areas.

The primary evidence for this stems from Moran & Desimone (1985) who report evidence that the effective sizes and locations of the receptive fields of neurons in area V4 and IT can change according to the task that the animal is trying to perform. It was found that cells in V4 whose receptive fields encompassed an attended stimulus, did not respond to unattended stimuli. They showed a normal response to attended stimuli indicating that attention does not enhance activity but tends to suppress unattended regions. However, the cells did respond to unattended stimuli if the attended stimulus was outside its receptive field. Cells in IT showed similar characteristics except that they never responded to unattended stimulus.

Lesion studies also point to a similar conclusion. Schiller and Lee (1991) examine effects of lesions in V4 on visual capacities. Five monkeys were first trained on detection and discrimination tasks. The tasks tested brightness discrimination, size, shape, color, pattern, motion, and stereoscopic depth perception. These were tested in two ways: the target object could either be more salient than the rest along the relevant dimension, or less salient. For example, the task could be to detect a bright target among dimmer distractors, or a dim target among bright distractors. After lesions to area V4 corresponding to the lower half of the visual field, performance was checked in regions

that were intact and also those that were affected by the lesions. They found that the monkeys responded much faster and with more accuracy to the more salient target than to the less salient target when stimuli were presented in the lesioned field. With salient targets the effect was mild and dropped off at the same rate as the normal field (as discriminability decreased). This occurred with bright targets among dim ones, large among small, and moving among non-stationary. Separate experiments established that there were almost no deficits in detecting singly presented targets of any type. The authors conclude that area V4 is involved in detecting less salient stimuli, a task that is thought to require attention.

6.2.3 Pulvinar

The pulvinar is a located in the dorsal part of the thalamus (Figure 6.3) and is strongly connected to just about every visual area, including the areas discussed so far, the superior colliculus (SC), frontal eye fields (FEF), and posterior parietal cortex (PP) (Robinson & McClurkin, 1989; Jones, 1985). For each projection from the pulvinar there is a corresponding reverse projection. The projections are topographic and for the most part non-overlapping. As a result the pulvinar contains several high resolution maps of visual space. Due its central location in the network of connections, several researchers have argued that this area of the thalamus must be involved in attention (Crick, 1984; LaBerge, 1990).

There is some fairly convincing evidence now that the pulvinar is in fact directly involved in the gating operation. Recordings of cells in the lateral pulvinar of awake, behaving monkeys have demonstrated a spatially localized enhancement effect tied to selective attention (Petersen *et. al*, 1985). When presented with a stimulus in their receptive fields, these cells show an increase in their firing rate when the animal attends to it. This enhancement is seen regardless of whether the animal is about to make an eye saccade to that location.

The above studies have support from alternate experimental methods. Lesion studies and PET scans suggest that the pulvinar is involved in covert attention, particularly in the gating operation. Patients with thalamic lesions have difficulty engaging attention and inhibiting crosstalk from other locations (Petersen *et. al*, 1987). In particular, these patients responded slower to cued targets even though there was sufficient time to attend. Lesioned monkeys give slower responses when competing events are present in the visual field (Posner & Petersen, 1990). LaBerge (1990) presents PET scans of human subjects taken during a letter discrimination task. By varying the difficulty of the discrimination he was able to regulate the amount of attention required. The resulting PET scans

suggested that activity within the pulvinar is significantly increased during tasks which require attention. LaBerge concludes that the pulvinar is involved in a filtering operation.

The above experiments are consistent with VISIT's gating system. The pulvinar contains separate maps representing each area it is connected to, and these areas are most active when covert attention is directed to that location. These maps could be the equivalent of VISIT's gated feature maps. If this is true, VISIT makes some strong predictions. If the pulvinar is damaged, then the ability to perform feature binding should be diminished. This is supported by the above experiments. There should also be a deficit in determining the locations of objects and computing spatial relations. In addition, VISIT predicts that if a map in the pulvinar corresponding to a particular cortical area is damaged, then there should be a corresponding deficit in the ability to bind those specific features in the presence of distractors. In the absence of distractors, the performance should remain unchanged.

6.2.4 Superior Colliculus

Like the pulvinar, the superior colliculus (SC) is a structure with converging inputs from several different areas (Figure 6.5). The SC can be split up into two distinct sets of layers: the superficial and deep layers. The superficial layer receives the bulk of its input directly from the retina (Huerta & Harting, 1984; Sparks, 1986). The deep layer receives input from a variety of different sensory modalities including visual, auditory, somatosensory, etc. There are maps for each of these modalities. It is generally believed that the SC is involved in integrating location information from these different modalities and in general orienting behavior. This is supported by the fact that the different maps are always in spatial register, i.e. that neighboring neurons in different maps are always referring to the same absolute spatial location (Jay & Sparks, 1984).

The deep layers of the superior colliculus contain error maps for eye saccades (Sparks, 1986). These neurons are laid out retinotopically in clusters, but at each location the cluster activity represents a value in motor coordinates. That is, they code the direction and amplitude of the eye saccade that would foveate on that spot. Thus the saccadic system just has to chose one of these locations and transmit the corresponding vector to the oculomotor system. VISIT uses the same representation as the error maps in superior colliculus to assist in bottom-up attentional shifts.

The superior colliculus is primarily involved in the generation of eye saccades (Wurtz & Goldberg, 1989) but it does have some relationship with covert attention. In (Posner, Cohen, & Rafal, 1982) the authors studied patients with a particular form of Parkinson's disease where the SC is damaged.

These patients are able to make horizontal, but not vertical eye saccades. The experiments showed that although the patients were still able to move their covert attention in both the horizontal and vertical directions, the speed of orienting in the vertical direction was much slower. In addition (Posner & Petersen, 1990) mention that patients with this damage shift attention to previously attended locations as readily as new ones, indicating a deficit in the inhibition of return mechanism.

The deficits are consistent with the functionality of the priority map in VISIT. Psychophysical experiments strongly suggest a linkage between the saccadic and covert attention systems (see Section 7.3.1). Since covert attention almost always shifts to a location just prior to an eye saccade (Posner & Petersen, 1990), it is possible that the same neural hardware serves as the basic priority maps for both the covert attentional and saccadic mechanisms.

6.2.5 Posterior Parietal Cortex

The posterior parietal cortex contains neurons responsive to visual stimuli. It receives a significant projection from superior colliculus and is thought to be involved in the production of voluntary eye saccades (Andersen & Gnadt, 1989). Experiments show that it is also involved in covert shifts of attention. Lesion studies have provided further evidence along these lines. (Posner *et. al*, 1984) found that damage to posterior parietal lobe led to deficits in the ability to disengage covert attention away from a target. These functions are consistent with portions of the control network in VISIT.

In (Mountcastle *et. al*, 1981) the authors tested the effect of behavioral state on light-sensitive neurons in the posterior parietal cortex (area 7) in monkeys. They found that the activity of these neurons increased when the animal was in a state of attentive fixation. In an earlier paper, in the context of eye saccades they show that these neurons start firing about *55* msecs before an actual saccade (Mountcastle *et. al*, 1975). They proposed that the posterior parietal cortex incorporated a general command center for oculomotor responses. (Mountcastle *et. al*, 1975) also found cells in posterior parietal areas that were sensitive to the *direction* of voluntary saccades, but not to spontaneous saccades. One possibility is that these areas also encode a high-level priority map and that the superior colliculus is responsible only for spontaneous shifts of attention. If these maps really are used as priority maps for covert attention, experiments should show that they fire in the same way for voluntary covert shifts as for eye saccades.

6.2.6 Other Areas

There is evidence that the frontal eye fields (FEF in Figure 6.5) are involved in saccade generation. It is thought to have a role analogous to the superior colliculus, and is particularly sensitive to high-resolution color stimuli (Schiller, 1985). It may also be involved in saccades to complex stimuli (Goldberg & Segraves, 1989). The role of the FEF in covert attention is not known. However, given the close correspondence between saccades and covert attention, a reasonable conjecture is that it plays the same role for attention.

There is also some evidence that the pre-frontal cortex is involved in saccades (Leichnetz & Goldberg, 1988). Boch and Goldberg (1987) have demonstrated cells which anticipate the appearance of stimuli. They suggest that pre-frontal cortex is involved in general goal-directed behavior. Goldman-Rakic (1991) has argued that pre-frontal cortex behaves as a temporary working memory for spatial tasks. Since there is a significant projection from posterior parietal areas to pre-frontal cortex, it is possible that the area is used in a similar fashion as the working memory in VISIT.

6.3 Discussion

In summary, I have tried to relate aspects of VISIT to the biological literature. For the most part there is a nice mapping between the functionality of the various modules and the known physiology of covert attention. Figure 6.5 displays the various visual areas again together with the proposed relationship to VISIT's modules. The literature is consistent with having the pulvinar as the gating system, the superior colliculus, frontal eye fields, and posterior parietal areas as a bottom-up priority map, the posterior parietal areas as the locus of the control networks, and the prefrontal cortex as working memory. The role of V4 and IT is not quite clear since it also seems to gate activity. It is possible that it is an additional gating system imposed on top of the early features.

(Posner & Petersen, 1990) have proposed a slightly different hypothesis. They suggest that neurons in parietal lobe disengage attention from the present focus, those in superior colliculus shift attention to the target, and neurons in pulvinar engage attention on it. It is interesting to compare the two. Their hypothesis looks at the time course of an attentional shift (disengage, move, engage) and assigns three different areas to the three different intervals within that temporal sequence. An

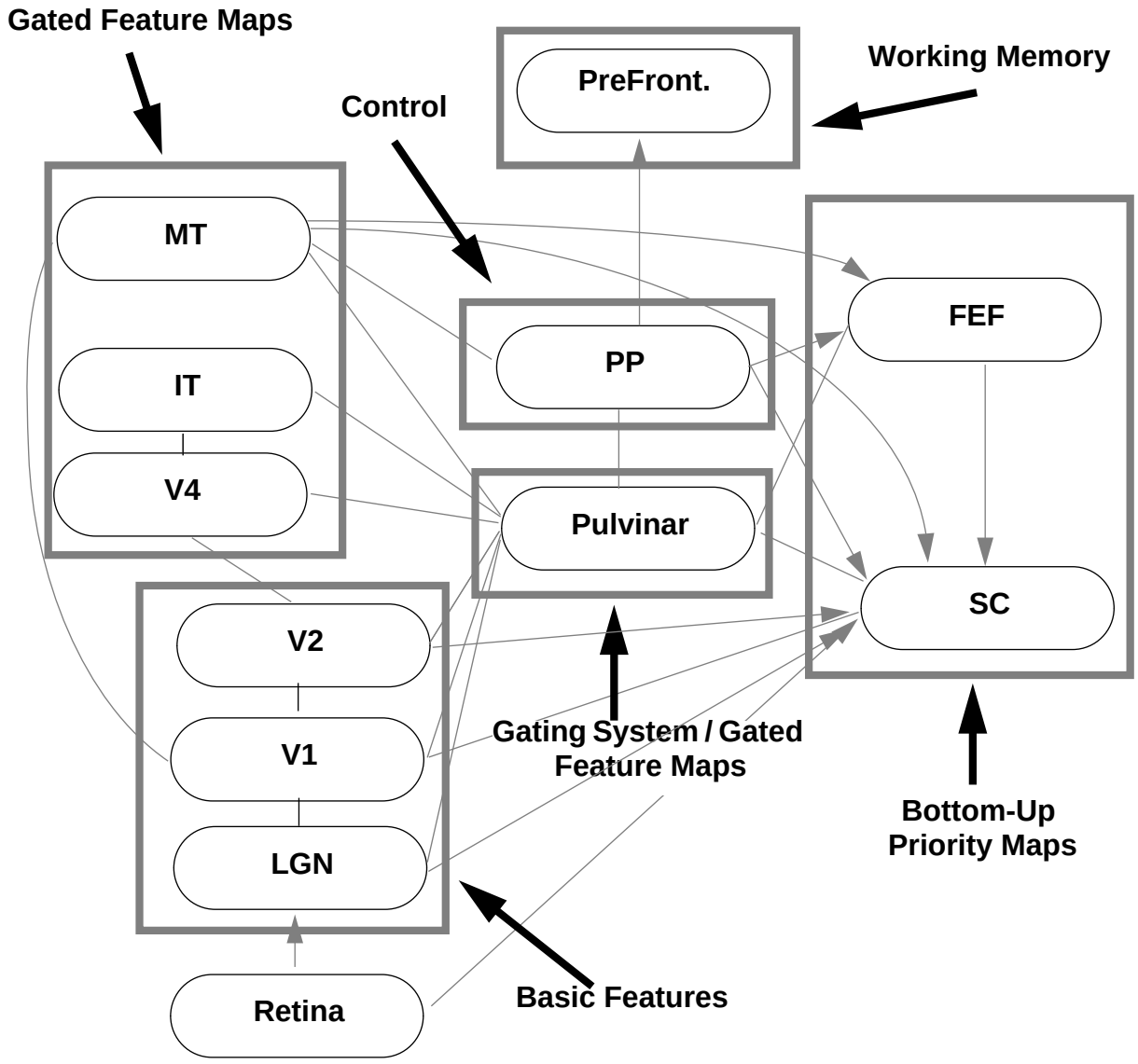


Figure 6.5. Proposed attentional functions of the different visual areas.

implemented model such as VISIT allows us to examine the underlying computations involved. The prediction is that all the areas are operating concurrently, and that all of them are involved during the shift. They just have differing computational responsibilities (i.e. compute the next location, gating, add error vector, etc.). For example, the same part of the control network is responsible for sending the disengage, move, and engage signals to the gating system. (Actually it is just one operation - transmit a new update vector.) While the gating network is being updated to a new location, the priority network and portions of the control network are already computing the next desired location. At this point the experimental work seems to be consistent with both points of

view, but experiments could be designed to distinguish between the two.

By no means is VISIT intended to be a detailed physiological model of attention. Precise modeling of even single neuron can require large amounts of computational resources. There are many physiological details that are not incorporated. However, at the macro level there seems to be a reasonable fit between the individual modules in VISIT and the known functionality of the different areas.

7. Visual Attention In People: Function

Understanding when visual attention is needed is as important as understanding its implementation. As yet there is no rigorous theory for predicting which tasks require attention, but in general there is good agreement between computational complexity arguments and the experimental results. The following sections review experimental work on determining which tasks require attention in people.

7.1 Visual Search

Over the past decade, visual search has been one of the most active areas of psychophysical research. There is strong evidence that a serial search using covert attention is required for some search tasks, whereas others require no such effort. This dichotomy makes visual search an ideal domain for studying how and why attention is used. Most of the experiments are aimed at uncovering the boundary conditions for these two modes. The following sections review some of this literature and its relationship to VISIT.

7.1.1 Single vs. Conjunctive Feature Search

In (Treisman & Gelade, 1980) the authors suggested that search for a target defined by a conjunction of features requires serial search, whereas single feature search does not. In their experiments, subjects were shown displays as in Figure 7.1 and Figure 7.2 and asked to detect the presence of a pre-defined target object (e.g. a shaded horizontal bar). They found that if the target is different from the distractor objects by a single feature (as in Figure 7.1), reaction times did not increase with the total number of objects. Conversely if a combination of two features is required (as in Figure 7.2), then reaction times did increase linearly with the total number of objects. In addition for conjunctive feature search, the slopes for target absent trials were approximately twice that of target present trials. They concluded that this was evidence for a search process that terminated as soon

(a)

(b)

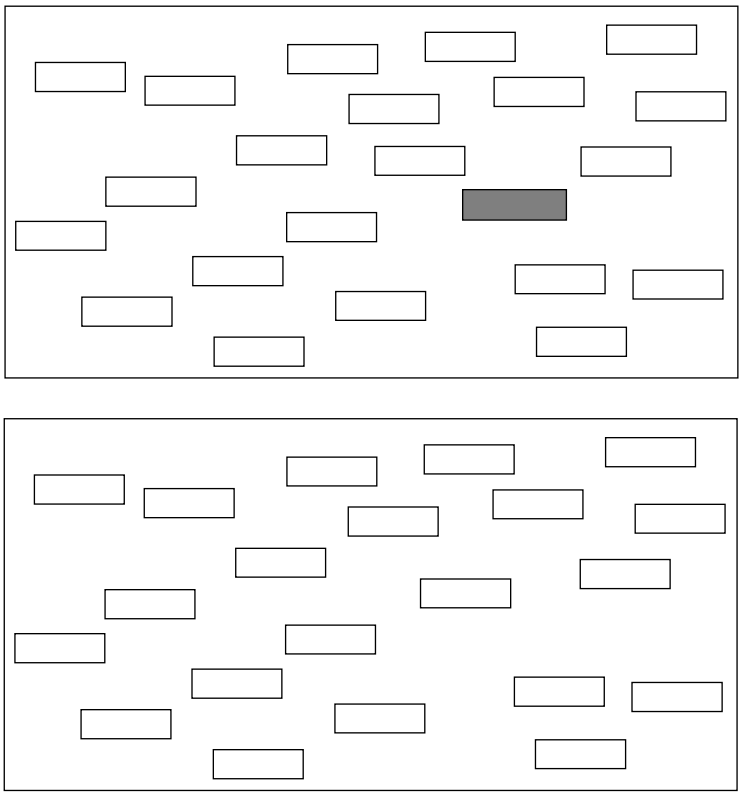


Figure 7.1. Single feature search. Target is a red (shaded) rectangle.

as the target was detected.

VISIT together with SWIFT accounts for the above results. When a feature of the target is not present in any of the other objects, SWIFT will always pick that feature map as the minimal feature. The target object will always be found on the first fixation, regardless of the number of objects in the image. (This corresponds to the case where *M=1* in Figure 5.15). In conjunction search, if the distractors are chosen randomly, then on average each feature map will contain the same number of objects. Regardless of the map chosen by SWIFT, *M* will be about one half of the number of distractors, so the search time will increase linearly with the number of objects. Since the search is self-terminating, the ratio of the slopes for the target absent and target present cases will be approximately *2:1*.

7.1.2 Search Asymmetries

There is another search paradigm where constant and linear time searches have been reported. Searching for a line oriented 18^o among vertical lines can be done in constant time, but searching

(a)

(b)

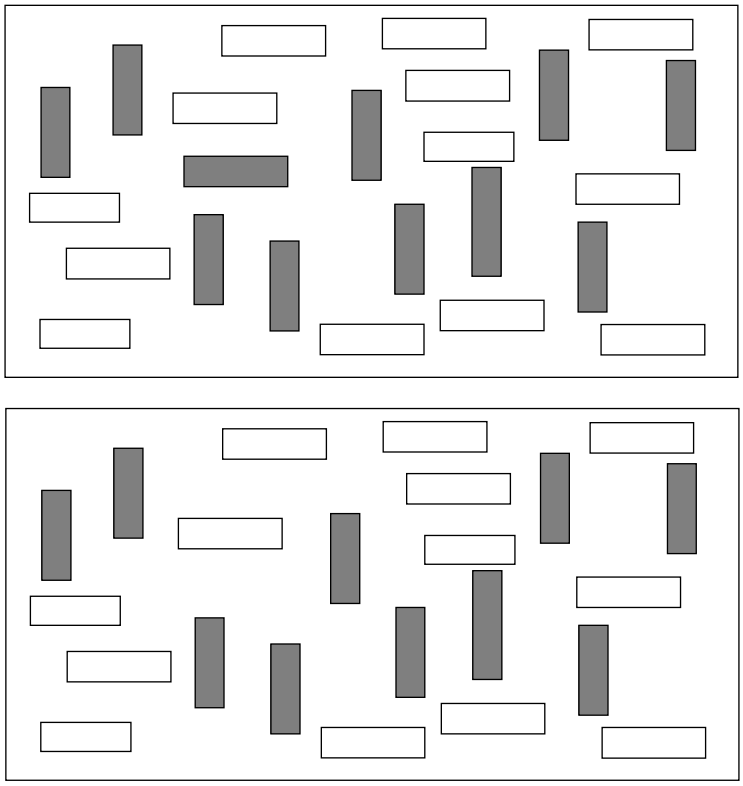


Figure 7.2. Conjunctive feature search. Target is a rectangle that is red (shaded) and horizontal.

for a vertical line among these oblique lines takes linear time (Treisman & Gormican, 1988). This asymmetry is explained by assuming that the early representation includes a finite number of orientations that are coarse coded, with units for vertical and some orientation greater than 18^o. Each oblique line is represented as a combination of activity in the vertical map and the map coding the successive orientation. Consider Figure 7.3(a), a pattern containing a single oblique line among a field of vertical lines. This image will cause several regions of activity in the vertical map but only a single region of activity in the other map. The presence of the oblique line can therefore be detected in constant time by computing a global OR. The opposite is true in Figure 7.3(b), an image of a vertical line among several oblique lines. This will generate several active regions in both maps except at one location, where only the vertical map is activated. In this case, the network must bind the presence of activity in one map with the absence of activity at the same location in another map. This requires serial search.

Similar asymmetries are present when detecting curvature, circles vs. ellipses, single vs. paired lines, etc. These can be explained as search for the presence of a feature vs. search for the absence of a feature. In fact, Treisman and Gormican (1988) argue that search asymmetries can be used as

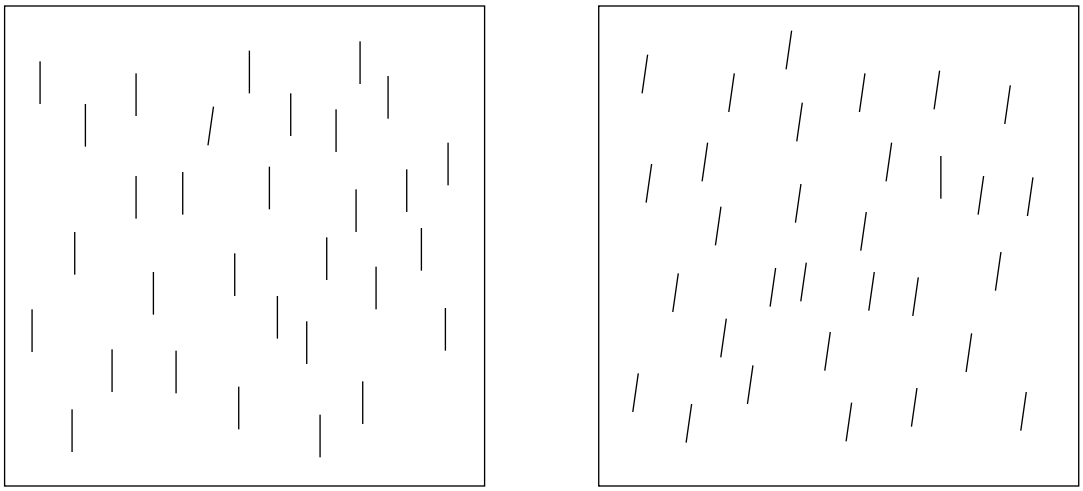


Figure 7.3. It takes less time to detect an oblique line among vertical lines (a) than to detect a vertical line among oblique lines (b).

a litmus test for features. In all of these cases, a central question is: how does the brain know what to do? The subject has no knowledge about his/her internal representations. Just knowledge about the target object is insufficient - the map that is searched depends on the particular image. The answer is simple if SWIFT is used: searching the map with the least total activity will always produce the correct results.

7.1.3 Evidence for an Efficient Search Strategy

Restricting Search to Objects with a Single Feature

In (Egeth, Virzi, & Garbart, 1984) the authors contest the claim that for conjunctive search, attention must be directed serially to each stimulus in the display. They argue that subjects can restrict search to those objects that have the same color or form as the target object. In the main experiment, subjects had to search for a red O among black O's and red N's. The number of red objects was held constant at 3 while the number of black O's was varied, analogous to Figure 5.13. The subjects were either told to attend to red objects or to O's. Reaction times for the attend-to-red subjects were flat in the number of total distractors for both target-present and target-absent conditions. In the attend-to-O case, the reaction times were flat for the target-present case.

These results are consistent with VISIT's ability to weight individual feature maps. SWIFT search suggests that these results would hold even if the subjects were not explicitly informed of the cor-

rect feature. Some evidence for this was found in a second experiment in the paper, although it was not designed to test this strategy. Experiments along the lines of the simulations in Figure 5.14 and Figure 5.15 would help to clarify this issue.

Triple Conjunction Search

Search for an object defined by a conjunction of three features results in several different search slopes (Quinlan and Humphreys, 1987). There were two situations that were tested: (a) every distractor shares exactly one feature with the target object, or, (b) every distractor shares exactly two features with the target. Both cases resulted in sequential search, but the slope in (b) was always steeper than the slope in case (a). The same results would be obtained by SWIFT. In case (a), on average the minimal feature will eliminate $\frac{2}{3}$ of the distractors. In (b), only $\frac{1}{3}$ would be eliminated on average. Thus SWIFT predicts that the slope in (a) should be about half that of (b).

Effect of Irrelevant Features

Treisman (1988) has tested single feature search where the distractors are not homogeneous. In one case the distractors varied in the relevant dimension (e.g. searching for a red target among multi-colored objects). In the second case the distractors varied along irrelevant dimensions (e.g. searching for a red target among objects of differing orientations and size, but constant color). The main result was that the first case resulted in serial search whereas the second case resulted in constant time search. This is consistent with the network. Feature maps are likely to be coarse coded, so variation within a dimension will cause the minimal feature map to have more than one spot of activity. This would necessitate a scan through the objects. Variation in irrelevant dimensions however should cause no activity in the relevant feature map, so SWIFT search will always locate the target in one step.

7.1.4 Fast Conjunction Search

Large Variances in Search Slopes

In (Cave & Wolfe, 1990) and (Treisman & Sato, 1990) the authors reported that search slopes could vary by a large amount across subjects. Conjunction searches of color and form produced wide variances. The slopes for some subjects were almost flat, whereas others were quite steep. Single feature searches produced consistently flat slopes. A difficult search task (such as search for a ran-

domly oriented T among randomly oriented L's) produced consistently steep slopes.

Conjunction Search in Constant Time

Some authors have reported conjunctive searches which always result in flat slopes. (McLeod, *et. al*, 1988) report that the detection of a moving X among static X's and moving O's can be done in parallel.[1] (Nakayama and Silverman, 1986) tested conjunction searches using the features color, motion, and depth. They found that motion-color conjunctions required serial processing, whereas depth-color and depth-motion conjunctions could be processed in parallel.

Recently (Treisman & Sato, 1990) and (Wolfe, Cave & Franzel, 1989) have suggested models where conjunctions can be detected in constant time with top-down information. Treisman and Sato suggest that if the features that are *not present* in the target inhibit the priority map then a location containing the conjunction of two features would retain the highest priority. Wolfe, Cave, and Franzel suggest an analogous mechanism using excitation instead of inhibition (see Section 10.1.2 for a review). Both of these suggestions can be modeled in VISIT by setting the feature weights appropriately (as discussed in Section 5.3.2). As mentioned before, this strategy is not very general. An equally serious problem is that it cannot explain serial search. If people can use such a general strategy for detecting feature combinations, why don't we get constant time search for all feature conjunctions? People only produce constant time search for very specific feature combinations, a fact that cannot be explained by the above mechanisms. (Cave & Wolfe, 1990) have suggested that this is due to varying amounts of noise in the system. There is no adequate explanation for the source of this noise, or why it should vary across individuals and across features.

Both of the above sets of results can be explained by VISIT with two assumptions: 1) that certain feature combinations are in fact represented explicitly, and 2) that these feature combinations are learned through experience. If such combinations are present, then the relevant map would have a single spot of activity so SWIFT would select it as minimal and locate the target in a single step. Since the features can be learned through experience, the particular combinations represented would vary from individual to individual.

---

1. Although, in what seems to be a direct contradiction to this result, (Treisman and Sato, 1990) report that conjunctions of motion and orientation produced the *steepest* slopes in their subjects.

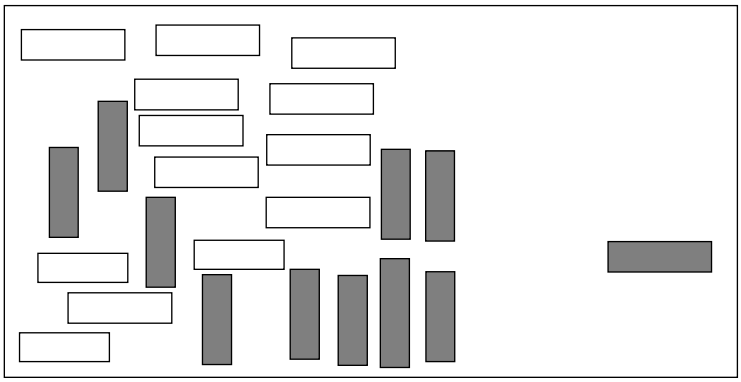


Figure 7.4. Example of grouping effects in visual search. Even though this is a conjunction search, people will find the target (shaded horizontal bar) in constant time.

Both assumptions have some evidence to support it. For example, it is known that area MT contains cells which are tuned to both direction of motion as well as orientation (Van Essen & Anderson, 1990). Karni & Sagi (1990) have shown that through training, adults can learn to improve their discrimination of certain features, and that this learning is retinally localized. (Shiu & Pashler, 1991) have further shown that learning of features is sometimes dependent on the cognitive state of the individual, and not just on the retinal inputs. Thus it is quite plausible that certain feature combinations are partially hard-wired in some people but not in others. For example, someone who is active in outdoor sports may well encode different high-level visual features than someone who spends most of their time indoors.

7.1.5 Effect of Perceptual Grouping on Search

A more puzzling phenomenon is the effect of perceptual groups on visual search. It is quite clear that the overall scene organization can have a dramatic impact on reaction time (Humphreys, Quinlan, & Riddoch, 1989; Treisman & Sato, 1990). A simple example of this can be seen in Figure 7.4, where the distractors clearly form a coherent group, separate from the target. Even though the task is a conjunction search, the total number of distractors have no effect on search time.

There is no explicit mechanism in VISIT to deal with these situations. In the current implementation, once the minimal feature is chosen, objects are just prioritized based on their size. Presented with the image in Figure 7.4, VISIT will simply go through its usual search sequence. To account for the perceptual grouping results, the system needs a smarter strategy for ranking image locations. As a first approximation, one might use multiple-scale spatial frequency detectors. (Neurons coding for spatial frequency are known to exist in V1 (De Valois & De Valois, 1988).) In Figure 7.4, a low frequency feature map would have less activity in the right half of the image, indicating

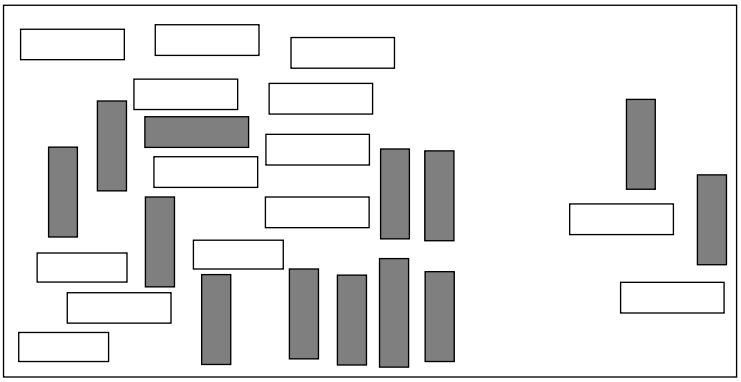


Figure 7.5. The target is now part of the larger group. The prediction is that search should now be hindered by elements in the smaller group of objects.

that the region probably contains a smaller group. A modified SWIFT strategy might therefore pick locations based on low activity in low-spatial frequency maps. The prediction based on such a strategy would be that people always check the smallest group first. This would predict that search should be hindered if the target is *not* in the smaller group (as in Figure 7.5). Although it is unlikely that this specific search strategy is used by people, experiments which control for some of these factors can help pin down the underlying computations.

7.2 Recovering General Scene Properties

7.2.1 Feature Binding

As the binding problem suggests, one of the basic tasks that should require attention is the association of a combination of features with one object. The visual search experiments provide some support for this argument. In addition (Treisman & Schmidt, 1982) present some direct evidence. Subjects were shown an image containing two letters of different colors. When the exposure time was very short, subjects would often switch the colors of the two objects. These “illusory conjunctions” did not occur when there was sufficient time to focus. It has also been shown that prior expectations can eliminate these effects (Treisman, 1988). For example subjects do not switch features if it leads to a nonsense assignment. VISIT is consistent with this. When attention is diffused over more than one object, the network cannot associate the active features with any one object. If

a higher level system tried to force such an association, then illusory conjunct errors would occur.

7.2.2 Computing Location

The binding problem also suggests that in the presence of multiple stimuli, computing the locations of individual objects should be affected. There is some evidence for this in the literature. In a review article (Wise & Desimone, 1988) mention that, even with voluntary effort, when presented with multiple targets simultaneously, the first eye saccade lands at an intermediate position. This is consistent with the notion that attention is required to individuate the locations of multiple objects. As further support, studies on visual search have shown that identification of a target is highly correlated with its accurate localization (Treisman & Gelade, 1980; Johnston & Pashler, 1990).

7.2.3 Computing Spatial Relations

Complexity arguments outlined in Chapter 4 suggest that computing spatial relations should require attention. The experimental evidence for this is not completely clear. (O'Connell & Treisman, 1991) have recently done some experiments along this line. They show that oriented dot pairs can pop out in a field of horizontal dot pairs. The result holds even if the dots are of different colors. On the other hand, oriented bi-constrast dot-pairs (pairs of black and white dots on a grey field) do not pop out. In this experiment, all the dot-pairs were close together in space. Computationally it may not be too expensive to explicitly represent some small number of relations formed by nearby objects. The fact that bi-contrast pairs do not pop out indicates that all such relations are not computed in parallel.

What about relations formed over arbitrary spatial scales? The equilateral triangle arguments suggest that attention is required to detect equilateralness with dot-triangles. This is due to the lack of local information. Interestingly, this argument predicts that detecting equilateralness for triangles made up of solid lines (Figure 7.6 (a)) can be done in parallel. This is because one simply has to check whether all angles are 60^o. If angles are represented pre-attentively, then solid line triangles should not require attention. The prediction would be that triangles as in Figure 7.6 (b) should take a longer time to process than those in Figure 7.6(a). Dot-squares (Figure 7.6 (c)) should take longer still, since one needs to attend to all four points.

Recovering more complex relations clearly requires some form of sequential processing, probably utilizing attention. Good examples are tasks like determining, in an image containing squares and

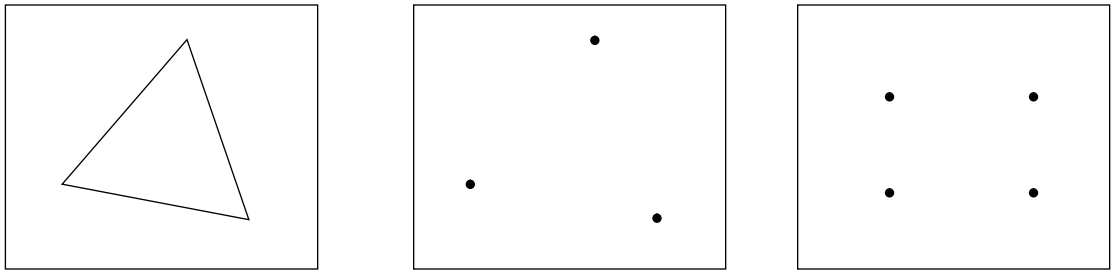


Figure 7.6. It should take longer to detect equilateralness in images like (b) than like (a). It should take even longer to detect whether a figure is square vs rectangular (c).

circles, whether there is a square next to the second largest circle (Mahoney, 1987). It is highly unlikely that we have feed forward networks for computing such predicates.

7.2.4 Computing Motion

There has been some recent evidence that covert attention is involved in the computation of motion. A simple demonstration of this is outlined in Figure 7.7. A black dot is drawn on the screen, and then, after some delay a line is flashed up. If the delay is long enough (100 msecs is plenty), then there is a clear perception that the line was drawn from the box outwards (in this case from the left to the right). With no delay, the line appears to be drawn instantaneously. The explanation in terms of visual attention is simple: when the black box is flashed up, it naturally attracts attention. When the line is subsequently drawn, attention is shifted to the opposite side. The perception that we have of a line being drawn from left to right is due solely to the *rightwards shift in attention*. When there is no delay, attention does not have time to operate exclusively on the black dot. The perception of the moving line is quite compelling and difficult to dismiss. (This sequence is easy to implement on a computer and makes an interesting demonstration of visual attention.)

An interesting hypothesis has been proposed by Cavanagh (1990). We often track a moving object by following it with our eyes and head. Although the retinal image of the object is fairly constant, the global motion of the object can be recovered by accumulating the motor commands to the eye and head. Of course, we can also sense the motion of objects while our eyes remain fixed. How is this motion perceived? There are two possibilities: 1) the perception may be due to the signals generated by low-level motion detectors or, 2) if a moving focus of attention is used to track the object, motion perception may be obtain from the signals generated by the attention system itself. (Cavan-

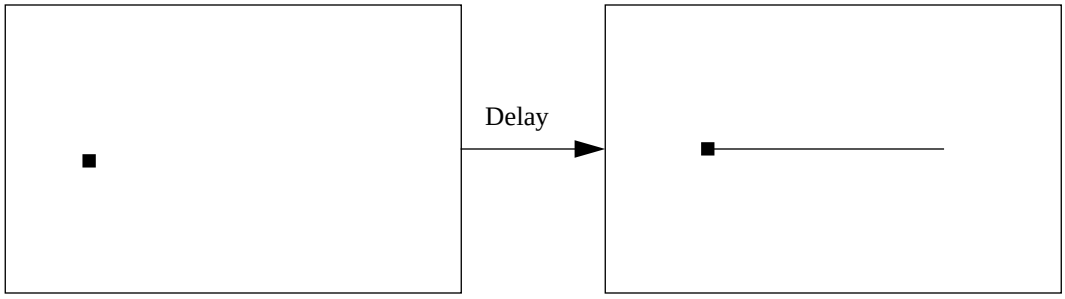


Figure 7.7. A black dot is flashed, followed by a delay, followed by a horizontal line. With a delay of 60 msecs, there is a clear impression of a line being drawn from left to right.

agh, 1990) describes evidence to support the second view. It is known that low-level motion detectors are highly sensitive to luminance but not color (Livingstone & Hubel, 1988). A circular counterfeit grating (two identical circular gratings moving in opposite directions, clockwise and anti-clockwise) was used. The key finding is that even when the two gratings were equi-luminant, subjects were still able to track their motion. This seems to rule out the low-level motion detectors since no motion information is present in the luminance channels.

If Cavanagh's hypothesis is true, it imposes an interesting constraint. The actual *movement* of the focus of attention should be recoverable. The task is made more complex by the fact that multiple shifts may be necessary in the course of tracking an object. In VISIT this operation would be simple to implement. Higher stages always have access to the attentional parameters, hence a trace of these parameters can be used to recover the motion. The focus is always updated using shifts relative to the current position. Temporal integration of the update vectors would give an accurate estimate of the net direction of motion from some reference point. By keeping track of the time as well as the updates, it is possible to recover the speed.

7.3 Interfacing with Other Forms of Attention

The above experiments suggest that covert attention is necessary for a wide range of visual tasks. Recent experiments have shown that the covert attention system seems to be related to other forms of attention as well. In this section we review some of the research which discusses this relation-

ship.

7.3.1 Relationship to Eye Saccades

Covert attention seems to be very closely related to eye saccades. The experiments described in (Posner *et. al*, 1982) show that covert attention almost always moves to the intended target of an eye saccade about 200 msecs before the eyes begin to move. Clearly the converse is not true: attention movements do not always result in eye movements. In (Posner *et. al*, 1982) patients unable to saccade in the vertical direction were also slower at shifting attention vertically. Section 6.2 discussed various areas of the brain that seem to be involved in covert attention. It is interesting to note that all of these areas are also involved in eye saccades (Wurtz & Goldberg, 1989).

These experiments imply that the two systems are highly related but can be decoupled. There might be a good computational reason for this relationship. The covert attention system is much faster than eye saccades. Covert attention shifts can occur as often as 25 times a second (Julesz & Bergen, 1987) whereas eye saccades only occur about 5-7 times per second. Since eye saccades are a more expensive operation (in terms of computation time), it makes sense to use the covert attention system to prune out candidate target locations for an eye saccade. Yarbus (1967) describes experiments where saccadic targets were highly dependent on the instructions given to the subject. For example, when asked to describe the expressions of people in a picture, the eyes fixated only on the faces. When asked to describe the furniture subjects avoided the people. Given the visual search results, it is highly unlikely that the visual system can saccade to complex features without first covertly attending to it.

7.3.2 Relationship to Auditory Attention

There is some preliminary evidence that the auditory attention system and the visual attention system are also quite related. In (Posner *et. al*, 1987) the authors investigate whether spatial attention mechanism uses the same system as auditory attention. In the first experiment subjects were asked to perform two attention-requiring tasks simultaneously. The primary task was a standard visual attention task (as described in Section 6.1.1). The secondary task was linguistic: the subject had to count the number of nouns in a list of spoken words that started with the letter “p”. They found that the cued-side advantage for the visual attention task disappeared. They argued that this was proof that the two forms of attention shared some common mechanisms. (Jay & Sparks, 1984) have provided further proof of this hypothesis. They looked at cells in superior colliculus that were respon-

sive to auditory stimuli as well as cells responsive to visual stimuli. Both the auditory and visual cells were topographically arranged, but since the eyes can move relative to the head, each should handle a different frame of reference. In fact, they found that the receptive fields of both types of cells were tuned to eye-centered coordinates. In other words, the receptive fields of auditory cells shifted coordinates dynamically to compensate for changes in eye position.

8. Extending VISIT

8.1 A More Flexible Gating Network

A fast gating network is central to VISIT. The following sections discuss various extensions to the gating network. These extensions allow a richer class of phenomena to be modeled within the same framework as VISIT.

8.1.1 A General Framework for Focus of Attention

This section describes a general mechanism for implementing attention in arbitrary input spaces. The gating network in VISIT is an example within this framework. The framework is used to demonstrate the robustness of the gating function. It is modified to include foci of different shapes, the ability to dynamically change this shape, and foci with smooth boundaries.

Locally Tuned Receptive Fields in N Dimensions

I first present a simple scheme for implementing static localized receptive fields using linear threshold units. The scheme relies on the following fact: if one maps the points in $\Re^{n-1}$ onto the paraboloid defined by $z = \sum_{i=1}^{n-1} x_i^2$, then the intersection of a hyperplane in $\Re^n$ with this paraboloid projects onto a sphere in $\Re^{n-1}$. Thus there is a mapping between planes in $\Re^n$ and spheres in $\Re^{n-1}$. To select a set of points which lie within a sphere in some space one just has to project the points onto the paraboloid and slice it with the plane corresponding to the sphere. Points which lie "beneath" the plane are within the sphere. Figure 8.1 illustrates this for $\Re^2$. Notice that the computation of a threshold unit is exactly that of deciding on which side of a hyperplane an input point lies. To encode circular receptive fields with threshold units, you just need to include an extra input: the sum of the squares of all the other inputs. An equation of the form:

$$-(\sum w_i x_i + \sum x_i^2 + const) > 0 \tag{8.1}$$

will be positive only if $\vec{x}$ lies within a spherical volume determined by the weights and constant.

Dynamic Receptive Fields

In addition to being able to select a portion of the input space, we need the ability to shift the location and size of the receptive field around quickly in response to changing demands. In the figure there are two ways to do this. The first method involves changing the slope of the hyperplane. In Figure 8.1 (a) note that as the slope increases the center of the projected circle will shift to the right. For any sphere it is possible to compute the coefficients of the hyperplane which produces that sphere. Given a plane $\vec{m} \cdot \vec{x} = \vec{m} \cdot \vec{c}$ where $\vec{m}$ and $\vec{c}$ are real-valued vectors, the projection of the intersection of the plane with the paraboloid is a sphere whose center is:

$$(a_1, a_2, \ldots, a_{n-1}) = \left(-\frac{m_1}{2m_n}, \ldots, -\frac{m_{n-1}}{2m_n}\right) \tag{8.2}$$

and whose radius is:

$$R = \frac{\sqrt{m_1^2 + m_2^2 + \ldots + m_{n-1}^2 + 4m_n(\vec{m} \cdot \vec{c})}}{2m_n} \tag{8.3}$$

In a threshold unit, changing the slope of the hyperplane corresponds to changing the weights of the inputs. So, with appropriate training one of these units can learn the correct position of its receptive field.

Since learning in neural networks typically involves several iterations, the time scale for weight changes is normally too slow to allow dynamic computations. An alternate method is to fix the plane but shift the paraboloid, by computing:

$$z = \sum_{i=1}^{n-1} (x_i - a_i)^2 + r^2 \tag{8.4}$$

This moves it a distance a_i along dimension i (changing the location of the sphere) and a distance

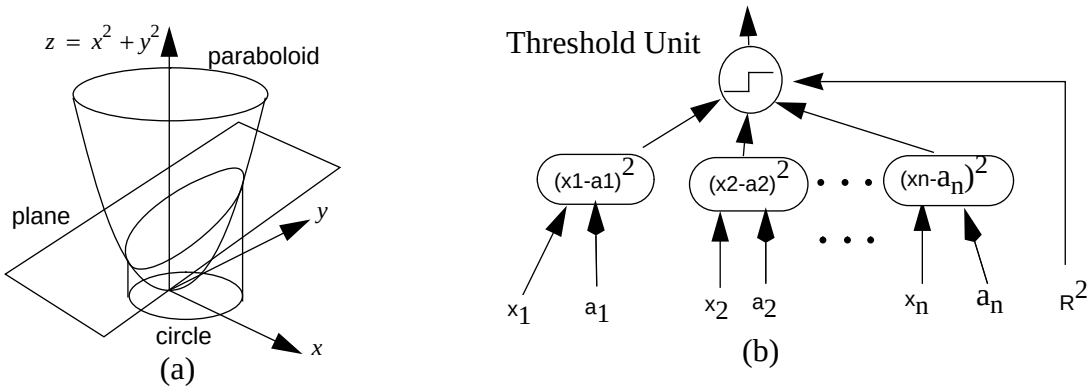


Figure 8.1(a) The plane intersects the paraboloid in a curve which projects to a circle. (b) Architecture of threshold unit computing the intersection in n dimensions.

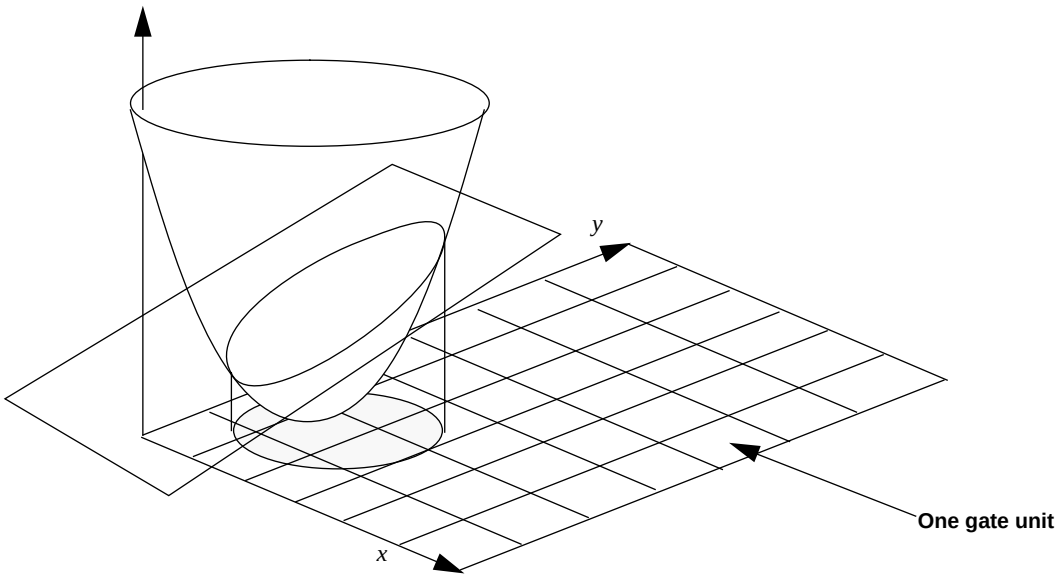


Figure 8.2. The gating layer encodes the *x*-*y* subspace.

r^2 along the z-axis (changing the radius of the sphere). If the a_i's and r^2 are available as input then the receptive field can be changed an arbitrary amount in one time step. Figure 8.1(b) shows how such a unit would be configured. The net effect is that the threshold unit will respond only when the input vector $\vec{x}$ lies within the spherical receptive field determined by $\vec{a}$ and r.

Focus of Attention with Value Coded Units

So far we have assumed an *n*-dimensional input space that is encoded as *n* analog signals. In VISIT a circular focus is implemented on a *2*-dimensional retina. The units in this representation are laid out on a flat sheet, with each unit explicitly encoding a local region in the space. In effect the gating layer encodes the *x*-*y* subspace in the figure (Figure 8.2). The inputs to the gating network represent the amount the parabola has to be shifted in order to arrive at the correct circle.

8.1.2 A Non-circular Focus

The above geometric interpretation has a number of advantages. It points out how to obtain non-circular receptive fields. By altering the non-linearity, one can obtain cross-sections with different shapes. For example, elliptical foci may be obtained by using the paraboloid: $z = \sum_{i=1}^{n-1} c_i x_i^2$ where c_i denotes the amount of stretching along each axis. In principle arbitrary shapes can be obtained by appropriately choosing the non-linearity. The shapes can be dynamically adjusted by including the appropriate parameters as additional inputs to the gate units (e.g. the c_i's in the ellipse example).

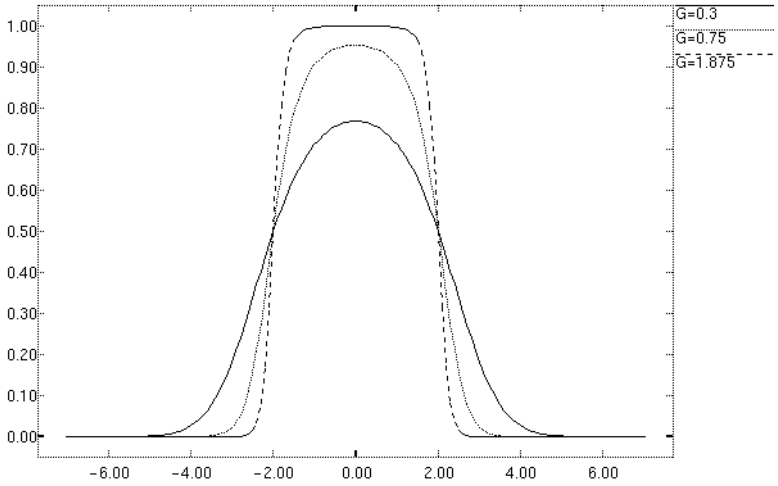


Figure 8.3. Effect of gain on locally tuned receptive fields.

8.1.3 A Smooth Decision Boundary

VISIT currently implements a circular focus with a hard boundary: pixels are either in or out of the focus. For some computations it may be important to incorporate a gradually decaying border. Such a focus may be obtained by using s sigmoidal output function in place of the threshold:

$$o = \frac{1}{1 + e^{Gz}} \tag{8.5}$$

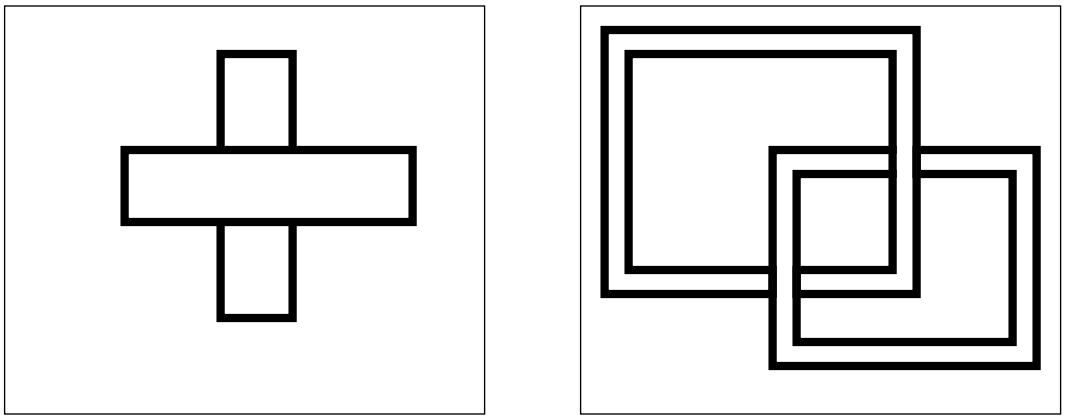
where z is the same as in Equation (8.4). This function results in a focus with a flat top, circular cross-section, and decaying boundaries. The gain parameter G controls the sharpness of the boundary. Figure 8.3 illustrates these properties in the one-dimensional case. The figure plots the function:

$$o = \frac{1}{1 + e^{G(x^2 - 4)}} \tag{8.6}$$

for three different values of G. By including G as an input to each gate unit, it is even possible to dynamically control the sharpness of the boundary.

8.1.4 Ease of Implementation

One concern with the gating network is its ease of implementation, both in silicon and biological hardware. It is quite likely that in the near future, special purpose analog chips will be manufactured to perform the tasks required by early vision[1]. It would be natural to include the gating network as part of such a chip but it is unclear whether exact quadratic computations can be performed



(a) (b)

Figure 8.4. Attending to perceived depth planes would allow the system to segregate the two objects in (a), but not the objects in (b).

in this media. The above framework demonstrates that the gating computation is actually quite robust. Exact parabola's are not necessary. If we just want some sort of a localized response in the projected space, *any* upward pointing, roughly cone like function will suffice. For example, an upward pointing Gaussian will do the job. The exact slope of the hyperplane is also not important, as long as it remains relatively stable. This robustness also makes a biological implementation of a similar gating operation more plausible.

8.2 Focus of Attention in Other Representations

The ability to extend the gating network to arbitrary dimensions allows a natural way to transfer the entire structure of VISIT to other value coded representations. This ability may prove to be quite useful in modeling other attentional phenomena. The literature on object based attention suggests that even when two objects are overlapping, people can attend to a single one (Section 6.1.5). One possible explanation is that the attention system attends to depth as well as two-dimensional regions (Duncan, 1984). For example in Figure 8.4(a) such a system would be able to segregate the horizontal object from the vertical one by attending to the near depth plane. The explanation pre-

1. For examples of current research in this direction see (Mead, 1989; Bair & Koch, 1991; and Horiuchi, et. al, 1991).

dicts that if the objects are intertwined (as in Figure 8.4 (b)), one cannot attend to each one individually. Note that since the two images are single two-dimensional images, the system must be able to gate perceived depth as well as stereoscopic depth. So a second prediction of such a system is that *perceived depth maps* exist in the visual cortex. Psychophysical experiments by (Ramachandran, 1988; Enns & Rensink, 1991) provide some preliminary evidence for this. Their experiments show that search for targets distinguished by complex 3D features such as shape from shading or orientation of simple cubes can be performed in constant time. The actual images they used were 2D images, so a disparity based feature map cannot account for the results.

Motion is another feature that seems to have similar characteristics. Preliminary experimental evidence along these lines is found in (McLeod *et al*, 1988) who show that conjunction targets defined by motion and form can be detected in constant time. (McLeod *et. al*, 1991) argue for the existence of a motion filter that can segregate moving objects by direction of motion.

Attention is also useful in non-visual modalities as well. A good example is audition, where attention is often necessary to distinguish individual sounds. Topographic maps representing frequency, amplitude, and phase are known to exist in auditory cortex (Konishi, 1983). If appropriate coordinates can be specified in these dimensions then a network like VISIT can easily implement an auditory attentional mechanism. Such a network could be used to model some of the results dealing with auditory attention (Gray & Wedderburn, 1960; Posner *et. al*, 1987). One can even imagine attentional mechanisms operating at the high level of structured representations. Any space that is value-coded is faced with the binding problem. The only requirements for VISIT are the existence of some notion of location plus a similarity metric. As long as these two conditions are met, the same network architecture can be used to implement attention in these spaces.

9. Learning To Focus Attention

Early in development, the primary visual areas self-organize to form retinotopic maps. There is evidence that only part of this process is genetically specified: the final maps are also dependent on the specific images that are experienced during development (Bear & Cooper, 1989). Network simulations with unsupervised learning rules have been used to model aspects of this developmental process (Kohonen, 1984; Obermayer *et. al*, 1991). Typically these networks are presented with an ensemble of images. Each unit learns to represent a local region such that the entire network best covers the set of input images. If a gating network exists in biological attention, it is likely to be also learned although the learning process will be more complex. In addition to forming a retinotopic map, each gate unit must learn its global position within the image. It must also learn to use this information to respond properly to its *(x, y, r)* inputs. The following sections describe a series of experiments aimed at understanding how this process might take place.

9.1 Adaptive Gate Units

First I describe how the behavior of individual gate units can be modified. Recall that in the gating network, each unit receives three inputs corresponding to the current circle of interest and responds with a "*1*" if it is outside it. The computation was implemented by units which exactly computed the equation for a circle:

$$r^2 - (x - X_i)^2 - (y - Y_i)^2 \tag{9.1}$$

where X_i and Y_i denotes the position of unit *i*. If this sum is greater than 0 then the unit is outside the circle defined by *(x,y,r)* and turns on. This quadratic computation can be modeled by second order units, a generalization of the semilinear units used in standard back propagation. The output of such a unit is given by:

$$o_i = s\left(\sum_{k,j} w_{ikj} o_k o_j\right) \tag{9.2}$$

where k and j range over all the inputs to the gating network and s is the sigmoid function:

$$s(x) = \frac{1}{1+e^{-x}} \tag{9.3}$$

We also include a bias unit (a unit whose output is always *1*) as additional input to each unit in order to generate the low order terms[1]. By expanding Eq. (9.1), we get:

$$r^2 - x^2 + 2xX_i - y^2 + 2yY_i - (X_i^2 + Y_i^2) \tag{9.4}$$

Since X_i and Y_i are constant for each gate unit, by setting the weights in Eq. (9.2) appropriately the unit can compute Eq. (9.4). The only difference is that the unit uses a sigmoid instead of a threshold which results in a soft decision boundary instead of a hard one (Section 8.1.3).

It is simple to modify back propagation to deal with second order units (Rumelhart & McClelland, 1986). Given a second order unit and an error signal δ_i, the update rule for each weight is:

$$\Delta w_{ikj} = \delta_i o_k o_j \tag{9.5}$$

where δ_i is computed as in standard back propagation. Given appropriate teacher signals, second order threshold units using back propagation learning are capable of learning the correct mapping. The central issue is to generate δ_i in a biologically plausible fashion. In the following sections, we discuss some methods for doing this.

9.2 Using a Perfect Teacher Signal

The first and simplest scenario assumes the existence of a perfect teacher. That is, given a gating layer, for each *(x,y,r)* input there is a signal available for each gate unit that specifies whether it should be on or off. With this type of a training signal, each unit receives direct information about its correct state and should rapidly learn to implement the correct mapping.

Simulation results verify this intuition. The particular network used consisted of four input units corresponding to (x,y,r) and a bias unit, plus a *10x10* array of second order units. The net input to the unit was a second order weighted sum with one weight for every pair-wise conjunction of its

1. Second order units in turn are a special case of sigma-pi units (Rumelhart & McClelland, 1986). Such units have been proposed as a model of computation in cortical neurons (Mel & Koch, 1990).

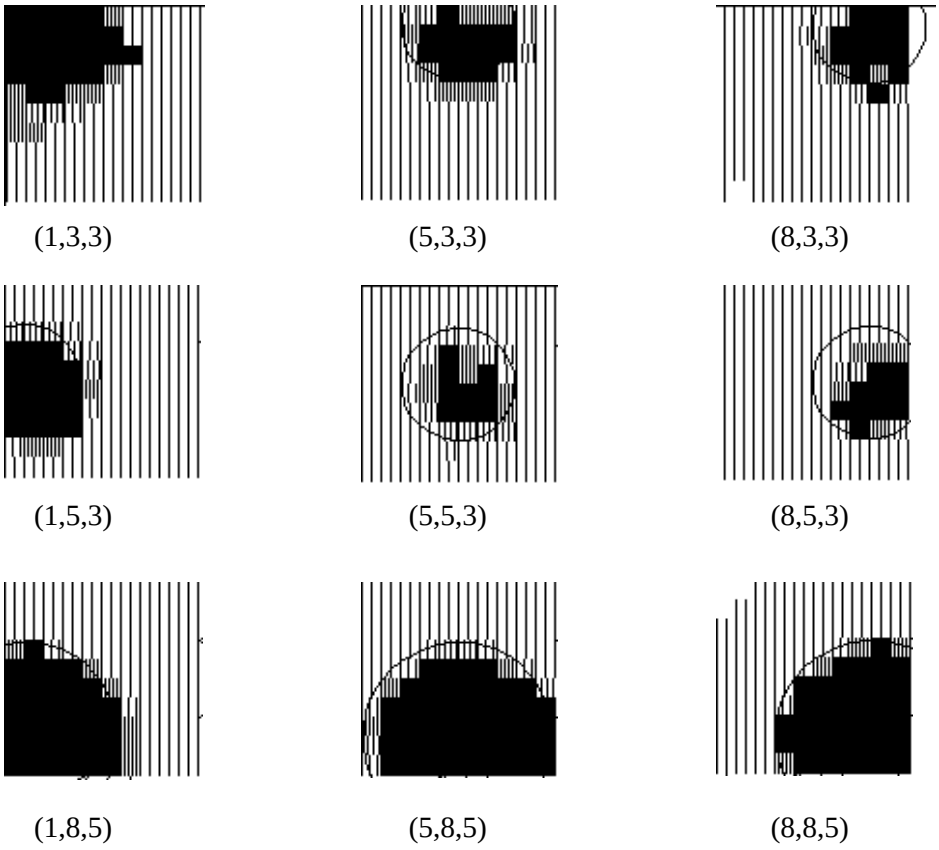


Figure 9.1. A 10X10 gating layer after being trained on a perfect training signal. Numbers below each figure are the corresponding (x,y,r) input values. The circular outline is the desired boundary. The shading represents the strength of each output signal.

inputs.

The weights were initialized to uniform random numbers in the range [-.5,.5]. The training consisted of presenting the network with randomly selected input triples. For each training pattern, the correct desired output was explicitly computed for each gate unit and used as the teacher signal. Back propagation using the update rule in Eq. (9.5) was used to update the weights after each pattern presentation. Figure 9.1 shows the output of the network after *12,000* weight updates. The figure displays the patterns of activity for several different values of *(x, y, r)*. The gating network has learned to exhibit a circular pattern of activity and correctly learned to shift it according to the input. This demonstrates that if a perfect teacher signal is available, the gating network can learn to focus attention.

9.3 Reinforcement Teacher Signals

The above scenario leads to efficient learning but is biologically implausible. It requires detailed knowledge about the internal structure of the network. In particular, the teacher must already know the locations of each of the gate units and the network must be able to propagate *different* error signals to each one. It would be more realistic to use a signal that is globally available and constant for each gate unit. This way, the same error signal can be broadcast to each gate unit from some other part of the system responsible for evaluation. A natural candidate for such a signal is the result of some general task that the network is involved in. Under this scenario, if the network completes the task correctly, then it receives positive reinforcement otherwise it receives negative reinforcement.

Visual search is a good domain for this sort of a reinforcement signal, since we know it requires attention. It is possible to generate a plausible error signal for the gating network within this domain. During a search sequence the gating network is responsible for tasks of the following sort: *"in the current image, determine the activity of a given set of features at a given location"*. If the success and failure of this task is used to train the gating network, then the teacher signals no longer depend on the internal structure of the network but can instead be derived from the results of actions on the external world. In addition, the same error signal can now be used for each gate unit. Since the task requires a focus of attention, gate units trained with it should learn the correct mapping.

9.3.1 Simulation Results

To test the above hypothesis, simulations were performed using a simplified version of visual search. The structure of the network is shown in Figure 9.2. It consists of an image, a gating layer, a single gated feature map, and a global OR. The gated feature map receives inhibition from the gating layer and positive connections from the image. The feature map simply signals the presence of activity in the corresponding locations. A global OR is computed from the activity in the gated feature map and represents the output of the network. All the weights to the gating layer are adaptive. The rest of the weights are pre-specified and kept fixed to reduce the number of free parameters (the weights to the gated feature map are all the same negative number, so there is no real need to learn them). As in the previous scenario, the gating network consists of second order units containing a weight for pair-wise conjunctions of the inputs. For this task, only those connections that

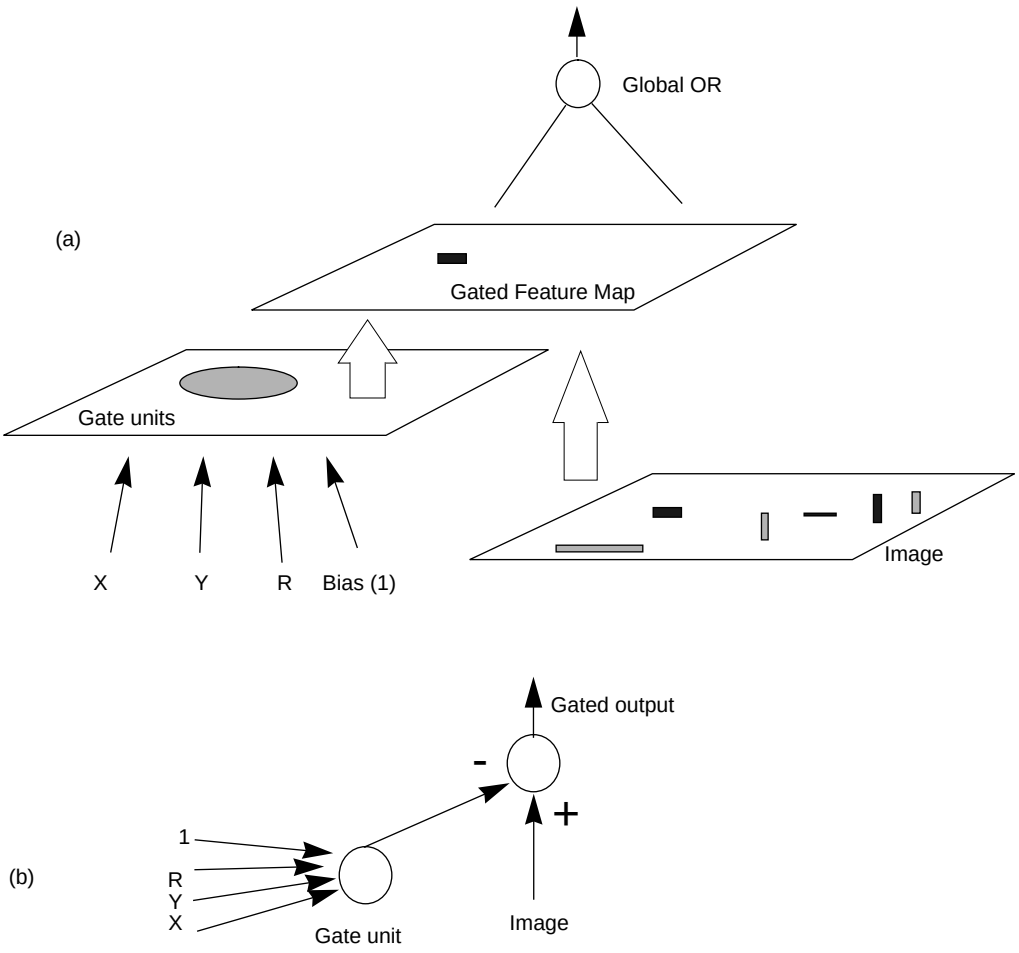
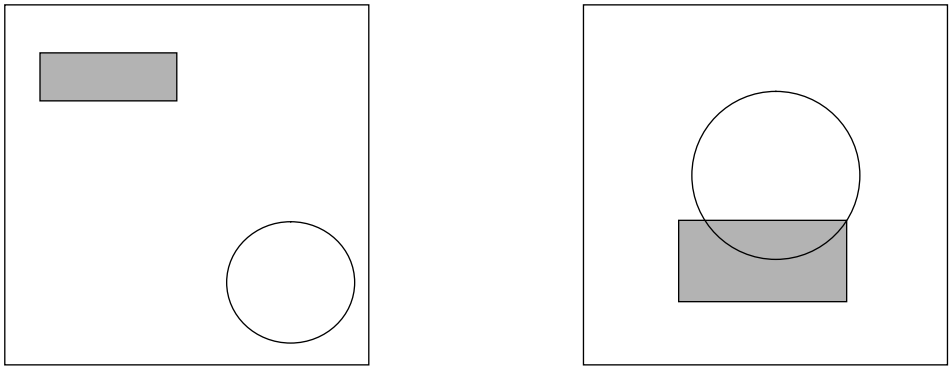


Figure 9.2. Architecture of the network. (a) The general network structure. (b) The connections at a single location.

are strictly necessary for the task are included. Note that some of the second order connections are unnecessary for implementing Eq. (9.4). In particular the weights for the terms *r, xy, rx,* and *ry* can be set to zero, which eliminates four degrees of freedom from each gate unit. This weight elimination is not necessary but makes the learning proceed much faster.

A *6x6* image was used. The weights to the gating network were initialized to uniform random values in the range [-.5,.5]. The network was trained with patterns consisting of an image and an *(x,y,r)* triplet (see Figure 9.3 for two examples). There were *5000* such training images, each with a randomly sized rectangle placed randomly within the image. The desired output of the global OR was set to *"1"* if there is any activity in the image in the region specified by the input triple, *"0"* otherwise. The error between this value and the value of the global OR was back propagated and used



Desired output = 0 Desired output = 1

(a) (b)

Figure 9.3. Two example training patterns for the reinforcement learning scenario, and their desired outputs. The circle represents the focus of attention implied by the *(x,y,r)* inputs.

as the error signal for the gating network.

Although in principle the task constraints should force the gate units to learn to focus attention, in practice this is a very difficult learning task. The reinforcement signal is extremely noisy with respect to each gate unit. For a given training pattern, a correctly behaving gate unit can still receive negative reinforcement if some of the other gate units are performing incorrectly. For example, consider a gate unit, g, in the lower left corner of the training pattern shown in Figure 9.3(a). An object is in the upper left corner and the current focus of attention is in the lower right. Since the focus does not intersect the object, the desired output is *0*. However, if any of the gate units covering the object turn on (incorrectly), then the output of the network will be incorrect and *g* will receive a negative error. Similarly, a unit with incorrect output can receive positive reinforcement if some of the other units perform correctly. For example, if *g* were to turn on incorrectly, the gated feature map output at that location will still be off since the corresponding image pixel is off. If none of the gate units within the object turn on, then the network will still produce the correct output, and *g* would receive positive reinforcement. During the initial learning phase, when the gate unit weights are random, the first of these two situations is quite likely to occur for large objects. The second situation may occur quite often with small objects. Due to this the teacher signal provides very little information, if any, to each unit.

Despite these problems, a significant amount of learning did take place. Figure 9.4 shows the behavior of the network after *250,000* iterations. It can be seen from the figures that as *x*, *y*, and *r* varied, the output of the gate units resembles a blob-like shape moving around. The output is not a

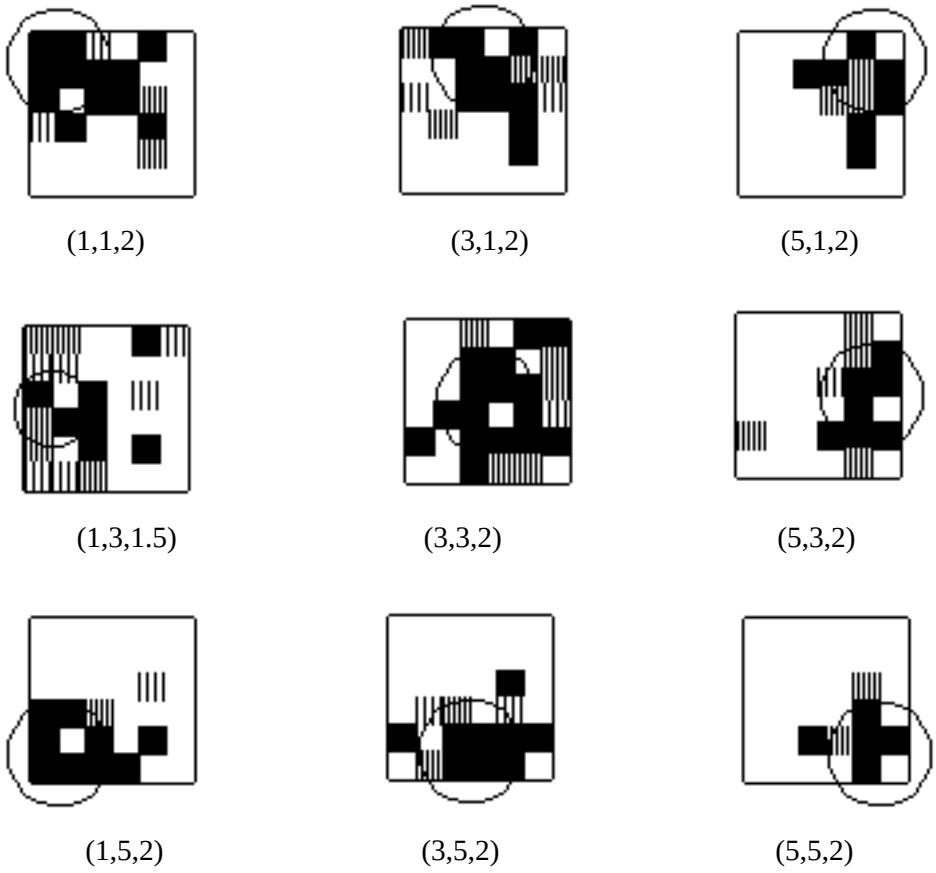


Figure 9.4. Output of a 6x6 gating network with link constraints after training with a reinforcement teacher signal.

perfect circle. There is some spurious activity, but it is possible that a smarter learning rule (e.g. one which exploited spatial regularities) could increase performance. Considering the problems mentioned above, it is quite astonishing that the network managed to learn anything at all. The resulting network can still act as a reasonable focus of attention provided the activity is blurred a little.This clearly shows that, on average, the reinforcement signal used does contain more information than noise and can be used to train the gating network.

It is interesting to examine the performance of the gating network during the training process. Initially the outputs are random and there is no relationship between the input and output signals. In the course of training, the network first learned the relevance of *r.* Figure 9.5 shows the output of the gate units for various values of *r* after *50,000* iterations. By this time the network has learned that for *r* near the maximum value, most of the gate units should be on whereas for *r* near *0,* most of the gate units should be off. This is reasonable since there is a direct correlation between the size

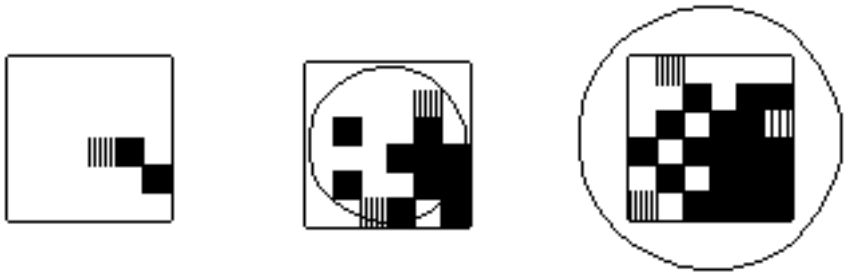


Figure 9.5. Output of a 6x6 gating network early in the learning process. Numbers below are the (x,y,r) inputs.

of the focus and the probability that the output is *1*. However, as the figure shows, the gate units have not yet learned anything about a circle. After *250,000* iterations (Figure 9.4), the network has learned that there is a correlation between *(x, y)* and the activity of the gate units. It has learned that as *x* increases, the activity should move from left to right. As y increases the activity should move from top to bottom. In effect, each gate unit has learned its global position within the image.

9.4 Discussion

Since retinotopic feature maps are learned in biological systems, the spatial attention system must also be learned. To study how this might happen, this chapter has discussed methods for training the gating network in VISIT. Experiments show that a rough blob-like focus of attention can evolve through learning using a simple, global reinforcement signal, even if that signal is quite noisy.

Ideally all aspects of the model should be learned. The learning of early visual features is getting to be well understood. There is an extensive computational literature on the development of such features. Several people have shown that unsupervised learning rules, such as the Hebb rule, can lead to units with response characteristics similar to neurons in V1 (Linsker, 1989; Becker, 1991). Recent psychophysical studies suggest that this learning continues in adults. (Karni & Sagi, 1990) showed that adult subjects can improve feature discrimination through repeated presentations and that this improvement is retinally localized. (Ling-po & Pashler, 1991) have further showed that this learning is not completely unsupervised: the subject must be actively attending to that feature

for the learning to take place.

Apart from the early feature maps and the gating network, it is unclear at present how the other modules in VISIT might be learned. The strategy used for the gating network should be applicable to some of them as well. For example, the priority map essentially acts as a feed forward system, i.e. units receive input (output of feature maps, feature weight values, plus the three attention parameters) and compute some outputs (error vectors, priority value). There are no dynamics involved, so with a perfect teacher signal, the priority map should be easy to learn. It should also be possible to construct a reinforcement learning scenario so that a global signal can be used to train the map.

A more difficult problem is training the control networks. This system is characterized by feedback pathways, interesting dynamics, and not much regularity. (Williams & Zipser, 1988) describe experiments showing how some simple control structures can be learned in recurrent networks. In general however this type of learning is not well understood. For example, it is not all clear how a search strategy such as SWIFT could evolve. There has been some research on the development of attention in people which might prove relevant (Johnson, 1990), but it is not yet at the point where detailed computational models can be formed. Methodologically, it is hard to test attentional behavior in infants which are only a few days old. Both the computational and psychophysical aspects of this issue is difficult and so this is one area where the future interaction between theory and experimental results is certain to be beneficial.

10. Related Models

VISIT builds upon, and has been inspired by, the work of a number of researchers. This chapter describes some of the work on building explicit models of attention, highlighting some of the common ground and differences with VISIT.

10.1 Two Psychological Models

There have been several psychological theories of attention in the literature. Some have been specified in more detail than others. Here I describe two: feature integration theory and guided search. Feature integration theory was chosen because it was the original model used to explain visual search, and it still seems to be the "standard" model of attention. Guided search was chosen because it has been implemented as a computer model of visual search and accounts for some of the more interesting visual search results. For additional models see (Bundesen, 1990; Duncan & Humphreys, 1989).

10.1.1 Feature Integration Theory

Feature Integration Theory was proposed by Treisman and Gelade (1980) to explain the different reaction time results for visual search. The model is outlined in Figure 10.1. The model makes use of separate feature maps which are gated by a single spotlight of attention. The spotlight is used when a conjunction of features must be detected. The location map contains all the objects in the image. The original model predicted that all conjunction searches would require a serial, self-terminating scan of every object in the image. There was no notion of priority, so conjunction searches would always result in linear search times. The model managed to unify many of the results under a single framework. This included single vs. conjunctive feature search, 2:1 search ratios, search asymmetries, and illusory conjuncts. However it was not consistent with some of the newer results, such as search within a feature, the parallel processing of conjunctions, etc. The theory has since been modified to include a notion of priority similar to that discussed in Section 5.3.2 (Treisman &

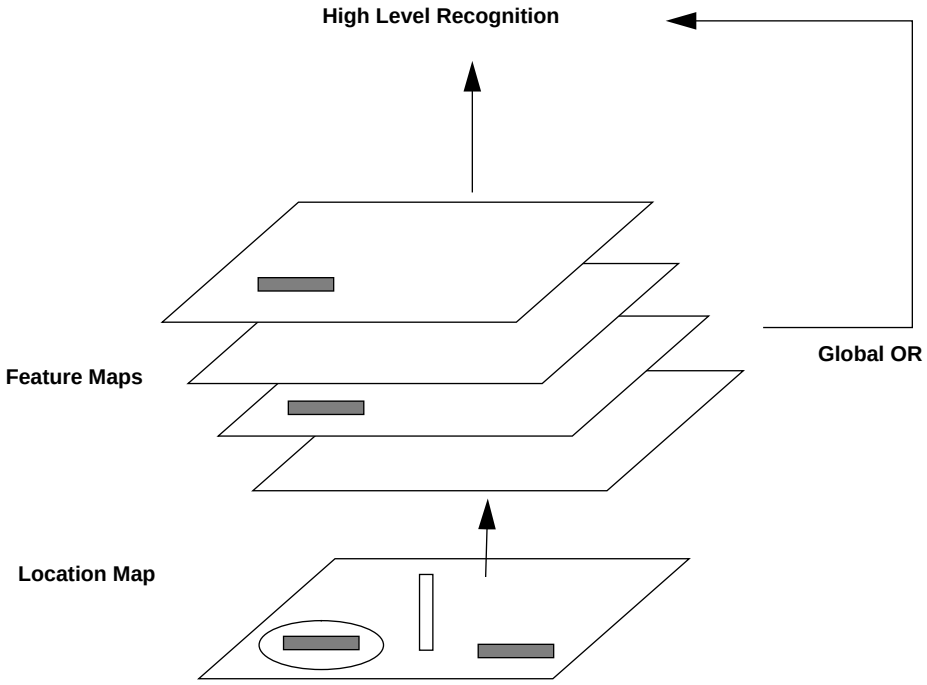


Figure 10.1. A sketch of feature integration theory.

Sato, 1990), but the mechanisms for setting the priority levels have not yet been specified clearly.

10.1.2 Guided Search

Guided Search was proposed by Cave and Wolfe (1990) to account for the results obtained in visual search experiments that conflict with Feature Integration Theory. The significant contribution of Guided Search is an explicit mechanism for computing the priorities using a combination of top-down and bottom-up information. A bottom-up priority is computed by comparing the feature values of each object against the feature values of every other object. This is done for every feature independently and summed up. This process can explain single feature search times since the target (which differs in one feature with every other object) will always have significantly higher priorities than the distractors (which only differ from the target). The top-down process adds another component which favors features belonging to the target object. This process is very similar to the scheme discussed in Section 5.3.2 and always assigns a conjunctive search target the highest priority. The model explains differing search slopes by assuming the existence of inherent noise in the priority calculations.

The authors present simulations showing that triple conjunction search, search within a feature,

effects of irrelevant distractors, constant time search for conjunctions, and varying search slopes can all be explained by Guided Search. As a model of visual search, Guided Search explains a large set of results, but cannot explain search asymmetries or the *2:1* search slopes. Furthermore, it cannot explain why only some feature conjunctions result in parallel slopes but not others. There are also three problems with implementing Guided Search as a connectionist network. The model assumes that objects have already been segmented and available in an array, with one slot per object. Second, the bottom-up computation is inefficient. It compares every location with every other location which requires $O(fn^2)$ connections where f is the number of feature maps, and n is the maximum number of elements. This is clearly not feasible for high resolution images. Finally, the model is very sensitive to the noise parameter. In order to explain the different search results, Guided Search must assume that different amounts of noise are present at different times and in different subjects.

10.2 Computational Models

Among the psychological theories of attention and visual search, Feature Integration Theory and Guided Search are notable in that they consider some of the underlying operations in some detail. They do not, however, model the implementations of these operations. There exist very few implemented models of attention. Other than VISIT, none implement every aspect within a connectionist network. The existing implementations can be partitioned into two classes: pyramid models and iterative models. The models are described below.

10.2.1 Pyramid Models

Koch and Ullman (1985) introduced the idea of a pyramid model of attention which has since been implemented by Chapman (1990) in a system called SIVS. In this model a log-depth tree is placed above a saliency map (Figure 10.2). At each level nodes receive activation from nodes below and transmit the maximum of these values to the next level. The top node of the tree computes the address and value of the most salient location. In addition, SIVS implements a fixed set of visual routines which use the tree to access image features. The main difficulty with this approach is that the focus is not continuously variable. As a result the scheme cannot handle real pixel based images

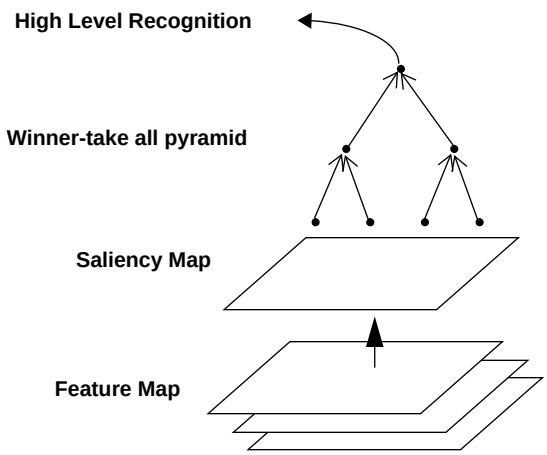


Figure 10.2. Koch and Ullman's attention pyramid.

but must assume a prior mechanism for segmenting the objects and normalizing their sizes. As with Guided Search, each object must occupy exactly one location in the saliency map. Chapman briefly discusses a way to reduce this restriction a little by overlapping several trees in a single network, but no details are specified. A further difficulty is that the time required to focus is logarithmic in the maximum number of elements. Depending on the degree of the nodes, focussing time can be relatively slow. Nevertheless, this scheme was the first concrete suggestion for implementing an attentional mechanism. The visual routines implementation, though not connectionist, represents the first attempt at modeling Ullman's framework.

Anderson and Van Essen (1987) discuss a mechanism for implementing attention using a pyramid-like structure that represents the sequence of visual areas from LGN up to IT. Each level can do one of several actions depending on some control inputs. Each level compresses the input by a factor of two. Each level can also send its input straight through, or shift it by one position to the left or right. To focus attention on a region, a series of micro-shifts are used to bring it into register with one of several discrete chunks at the bottom of the pyramid. Control signals are then used to route this region to the uppermost level. The main advantage of this scheme is that an entire region is routed to the top, so the spatial relations of objects within that region are preserved. There are a number of disadvantages. 1) It is unclear how a continuous focus of attention would be implemented. 2) The shifting operation is quite complex and time consuming. It is unclear whether this sort of a strategy can account for the extremely fast response times of human attention. 3) The scheme predicts that shifting operations occur at all levels of the hierarchy. There is as yet no physiological evidence for any sort of attentional effects below level V4.

10.2.2 Iterative Models

In contrast to the pyramid based models, (Mozer, 1988; Mozer, 1991) describes AM, a model based on iterative relaxation. In AM (as in VISIT) a region of activity in a layer of retinotopic units gates the activity from feature maps to higher levels. The current focus is determined by an iterative rule. At each time step units compare their activity with their neighbors and adjust their values according to a locally competitive update rule. In time the network selects a single region of maximal activity. The main advantage of this scheme over the pyramid models is a continuously variable region. AM also incorporates the ability to weight individual feature maps, although this capability was not used in the simulations. Sandon (1990) describes a model which also uses an iterative rule but performs the computation at several spatial scales simultaneously.

One problem with iterative models is that the settling time is quite sensitive to the size of the image as well as to specific images. It can be quite high if there are similar regions of activity that are widely separated. In simulations, AM with a 36x6 image array took anywhere from 20 to 100 iterations to settle (Mozer, 1991). A second issue is that the region selection is based completely on local information. There is no explicit mechanism for using top-down location information to shift attention. AM has been used as a model of word recognition and can model many of the perceptual effects associated with reading. Some of the visual search results can be explained under suitable noise assumptions, as in Guided Search.

(Fukushima, 1986) describes a model combining a log-depth hierarchical network and iterative relaxation. The output level of the network contains one unit per pattern to be recognized. Given an image the output nodes corresponding to the different patterns will each respond to some degree. The unit with the largest response is selected and a signal is transmitted back through the pathways activated by the input pattern. This creates a positive feedback loop which sharpens the detection of this pattern. The pathways corresponding to the other patterns gradually attenuate in the absence of this facilitation. In time the network attends to one of the input patterns. A unique property of this scheme is that the shape of the focus of attention depends only on the shape of the input patterns and thus can be arbitrary. However the process is quite slow since many iterations are required for the network to settle, and for each iteration the activity has to flow up and down a fairly deep hierarchy. The network has been used recently in the recognition of Chinese Kanji characters (Fukushima, Imagawa, & Ashida, 1991).

10.3 Discussion

VISIT incorporates a number of features of other models. The notion of a gating network and weights associated with feature maps is similar to the ideas in AM. The priority map is similar in spirit to Feature Integration Theory's location map, and Koch and Ullman's priority map. As in Guided Search, VISIT makes use of a combination of top-down and bottom-up information for the search process. The underlying constraints in VISIT are computational efficiency and the ability to work with high-resolution images. Due to this, the implementations of the above features have been very different. VISIT is the first model to separate the gating and priority computations and this results in fast run times. VISIT is also the first model to make every aspect of the control process explicit as a connectionist system. Finally, VISIT is the only system that deals well with high-resolution images.

11. Concluding Remarks

This thesis has presented a model of attention derived from computational principles and biological data. The overall design is simple, and should be easily implementable on a massively parallel computer. As far as I'm aware, VISIT is the first fully connectionist model of spatial attention (including implementation of the control processes). It is also the first model that is a useful computational model as well as being a reasonable psychological, physiological, and anatomical model of human attention.

The initial goal of this project was to build a neural bridge between early and late vision. I feel VISIT represents a step towards that goal but there is still much more to be done. It should be relatively easy to include more realistic feature maps and implement the model on parallel hardware. It should be simple to incorporate most connectionist theories of object recognition. Such a system should, for example, be able to search a sparse image for fairly complex objects in real-time. It will be more challenging to extend the system to deal with overlapping objects. Extending the system to handle multiple visual tasks will require the ability to dynamically decide what sequence of actions must be executed to complete the current task.

We are far from building a visual system with anything near the complexity of human vision. What is the best way to proceed? I personally believe that the major breakthroughs will come from those who keep both computational efficiency and biological plausibility in mind. As constraints for a model, these two might seem contradictory. What do the constraints faced by a chemical system have to do with the efficiency constraints faced by an electronic computer? My own experience argues that the interaction between the two can be extremely useful. For example, the priority map in VISIT was inspired by the existence of similar maps in monkeys for controlling eye saccades. Later I realized that error units were actually a very efficient and flexible encoding scheme. Conversely, some of the design decisions were made purely for reasons of efficiency. Examples include partitioning the network into the specific modules, and the SWIFT search strategy. I found out later that these features also matched the biological data well.

I don't believe these incidents are coincidence. At this point I am not even convinced that the two are distinct at the macro level. It seems clear to me that as our knowledge of both fields increase, the interaction between the two will lead to a remarkably rich set of models in the near future.

References

Ahmad, S. and Omohundro, S. (1991) Efficient Visual Search: A Connectionist Solution. In: *Proceedings of the 13th Annual Conference of the Cognitive Science Society,* Chicago, August, 1991.

Ahmad, S., and Omohundro, S. (1990a) Equilateral Triangles: A Challenge for Connectionist Vision. In: *Proceedings of the 12th Annual Conference of the Cognitive Science Society,* MIT, July, 1990.

Ahmad, S., and Omohundro, S. (1990b) A Connectionist System for Extracting the Locations of Point Clusters. Technical Report TR-90-011, International Computer Science Institute, Berkeley, CA.

Anastasio, T. J. (1991) A Recurrent Neural Network Model of Velocity Storage in the Vestibulo-Ocular Reflex. In: Lippman, R., Moody, J., and Touretzky, D.S. (eds.), *Advances in Neural Information Processing Systems 3,* Morgan Kaufmann, San Mateo, CA.

Anderson, C.H., and Van Essen, D.C. (1987) Shifter Circuits: A Computational Strategy for Dynamic Aspects of Visual Processing. *Proc. Natl. Academy. Sci.,* **84**:6297-6301.

Andersen, R.A., and Gnadt, J.W. (1989) Posterior parietal cortex. In: Wurtz, R.H., and Goldberg, M.E. (Eds.) *The Neurobiology of Saccadic Eye Movements.* Elsevier, New York.

Bair, W., and Koch, C. (1991) An Analog VLSI Chip for Finding Edges from Zero-crossings. In: Lippman, R., Moody, J., and Touretzky, D.S. (eds.), *Advances in Neural Information Processing Systems 3,* Morgan Kaufmann, San Mateo, CA.

Bear, M., and Cooper, L. (1989) Molecular Mechanisms for Synaptic Modification in the Visual Cortex: Interaction between Theory and Experiment. In: M. Gluck and D.E. Rumelhart (eds.), *Neuroscience and Connectionist Theory,* Lawrence Erlbaum Associates, Inc..

Becker, S. (1991) Unsupervised Learning Procedures for Neural Networks. To appear in: *International Journal of Neural Systems*, **12**.

Berger, A., Henik, A., and Rafal, R. (1991) Exogenous and Endogenous Orienting of Visual Attention. Poster presented at *Recent Advances in the Analysis of Attention*, Davis, CA.

Bruce, C.J., Desimone, R., and Gross, C.G. (1981) Visual properties of neurons in a polysensory

area in superior temporal sulcus of the macaque. *Journal of Neurophysiology,* **46**:369-384.

Bundesen, C. (1990) A Theory of Visual Attention. *Psychological Review,* **97**(4):523-547.

Cavanagh, P., Arguin, M., and Treisman, A. (1990) Effect of Surface Medium on Visual Search for Orientation and Size Features. *Journal of Experimental Psychology: Human Perception and Performance*, **16**(3):479-491.

Cavanagh, P. (1990). Pursuing Moving Objects With Attention. In: *Proceedings of the 12th Annual Conference of the Cognitive Science Society,* pp 1046-1047, MIT, July, 1990.

Cave, K.R., and Wolfe, J.M. (1990) Modeling the Role of Parallel Processing in Visual Search. *Cognitive Psychology*, **22**:225-271.

Chapman, D. (1990) *Vision, Instruction, and Action*. Ph.D. Thesis, Massachusetts Institute of Technology.

Corbetta, M., Miezin, F., Dobmeyer, S., Shulman, G., and Petersen, S. (1990) Attentional Modulation of Neural Processing of Shape, Color, and Velocity in Humans. *Science* **248**:1556-1559.

De Valois, R.L., and De Valois, K.K. (1988) *Spatial Vision*. Oxford University Press, New York.

Desimone, R., Albright, T.D., Gross, C.G., and Bruce, C. (1984) Stimulus-selective properties of inferior temporal neurons in the macaque. *Journal of Neuroscience,* **4**:2051-2062.

Desimone, Robert. (1991) Face-Selective Cells in the Temporal Cortex of Monkeys. *Journal of Cognitive Neuroscience*, **3**(1):1-8.

Duncan, J. (1984) Selective Attention and the Organization of Visual Information. *Journal of Experimental Psychology: General,* **113**(4):501-517.

Duncan, J. and Humphreys, G.W. (1989) Visual search and stimulus similarity. *Psychological Review,* **96**:433-458.

Egeth, H.E., Virzi, R.A., and Garbart, H. (1984) Searching for Conjunctively Defined Targets. *Journal of Experimental Psychology: Human Perception and Performance***, 10** (1):32-39.

Enns, J. T. and Rensink, R. A. (1991) Preattentive Recovery of Three-Dimensional Orientation from Line Drawings. To appear in *Psychological Review.*

Enns, J. T. and Rensink, R. A. (1991) A Model for the Rapid Interpretation of Line Drawings in Early Vision. In: *Visual Search II*, D. Brogan (Ed.). London: Taylor & Francis.

Feldman, J.A., and Ballard, D.H. (1982) Connectionist Models and their Properties. *Cognitive Science,* **6**:205-254.

Fukushima, K. (1986) A Neural Network Model for Selective Attention in Visual Pattern Recog-

nition. *Biological Cybernetics*, **55**:5-15.

Fukushima, K., Imagawa, T., and Ashida, E. (1991) Character Recognition With Selective Attention. In: *Proc. International Joint Conference on Neural Networks*, Seattle, 1991.

Giles, C.L., Griffin, R.D., and Maxwell, T. (1987) Encoding Geometric Invariances in Higher Order Neural Networks. In: *Proceedings of the Conference on Neural Information Processing Systems*.

Goggin, S.D., Johnson, K.M., and Gustafson, K.E. (1991) A Second-Order Translation, Rotation and Scale Invariant Neural Network. In: Lippman, R., Moody, J., and Touretzky, D.S. (eds.), *Advances in Neural Information Processing Systems 3*, Morgan Kaufmann, San Mateo, CA.

Goldberg, M.E., and Segraves, M.A. (1989) The visual and frontal cortices. In: Wurtz, R.H., and Goldberg, M.E. (Eds.) *The Neurobiology of Saccadic Eye Movements*. Elsevier, New York.

Goldman-Rakic, P. (1991) Representations and Processes in Human Prefrontal Cortex. Talk presented at the *13th Annual Conference of the Cognitive Science Society*, Chicago, August, 1991.

Goodhill, G.J., and Willshaw, D.J. (1990) Application of the elastic net algorithm to the formation of ocular dominance stripes. *Network*, **1**:41-59.

Gray, J.A., and Wedderburn, A.A.I. (1960) Grouping strategies with simultaneous stimuli. *Quarterly Journal of Experimental Psychology*, **12**:180-184.

Gross, C.G., Rocha-Miranda, C.E., and Bender, D.B. (1972) Visual properties of neurons in inferotemporal cortex of the macaque. *Journal of Neurophysiology*, **35**:96-111.

Holtzman, J.D., Sidtis, J.J., Volpe, B.T., Wilson, D.H., and Gazzaniga, M.S. (1981) Dissociation of Spatial Information for Stimulus Localization and the Control of Attention. *Brain* **104**:861-872.

Horiuchi, T., Lazzaro, J., Moore, A., and Koch, C. (1991) A Delay-Line Based Motion Detection Chip. In: Lippman, R., Moody, J., and Touretzky, D.S. (eds.), *Advances in Neural Information Processing Systems 3*, Morgan Kaufmann, San Mateo, CA.

Hubel, D.H. and Wiesel, T.N. (1968) Receptive Fields and Functional Architecture of Monkey Striate Cortex. *Journal of Physiology*, **195**:215-243.

Humphreys, G., Quinlan, P., and Riddoch, M. (1989) Grouping Processes in Visual Search: Effects with Single and Conjunctive Feature Targets. *Journal of Experimental Psychology: General*, **118**(3):258-279.

Ivry, R., & Cohen, A. (1990) Dissociation of Short and Long Range Apparent Motion in Visual Search. *Journal of Experimental Psychology: Human Perception and Performance*, **16**:317-

331.

Jay, M., and Sparks, D. (1984) Auditory receptive fields in primate superior colliculus shift with changes in eye position. *Nature* **309**(5967):345-347.

Johnson, Mark H. (1990) Cortical Maturation and the Development of Visual Attention in Early Infancy. *Journal of Cognitive Neuroscience* **2**(2):81-95.

Johnston, W.A., Farnham, J.M., and Hawley, K.J. (1991) Novel Popout: New Findings and Tentative Theory. Poster presented at *Recent Advances in the Analysis of Attention*, June 7-9, Davis, CA.

Johnston, J.C., and Pashler, H. (1990) Close Binding of Identity and Location in Visual Feature Perception. *Journal of Experimental Psychology: Human Perception and Performance*, **16**(4):843-856.

Jolicoeur, P., Ullman, S., and Mackay, M. (1986) Curve Tracing: A Possible Basic Operation in the Perception of Spatial Relations. *Memory and Cognition*, **14** (2):129-140.

Jones, E.G. (1985) *The Thalamus*. Plenum Press, New York.

Julesz, B. and Bergen, J.R. (1987) Textons, The Fundamental Elements in Preattentive Vision and Perception of Textures. In: Fischler, M.A. and Firschein, O. (Eds.) *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*. Morgan Kaufmann, San Mateo, CA.

Kahneman, D., Treisman, A., and Gibbs, B. (1991) The Reviewing of Object Files: Object Specific Integration of Information. *Cognitive Psychology,* in press.

Karni, A., and Sagi, D. (1990) Texture discrimination learning is specific for spatial location and background element orientation. Paper presented at the *Annual Meeting of the Association for Research in Vision and Opthalmology*, Sarasota, April 1990.

Keeler, J.D., Rumelhart, D.E., and Leow, W. (1991) Integrated Segmentation and Recognition of Hand-Printed Numerals. In: Lippman, R., Moody, J., and Touretzky, D.S. (eds.), *Advances in Neural Information Processing Systems 3*, Morgan Kaufmann, San Mateo, CA.

Klein, R. (1988) Inhibitory Tagging System Facilitates Visual Search. *Nature* **334**:430-431.

Koch, C., and Ullman, S. (1985) Shifts in Selective Attention: Towards the Underlying Neural Circuitry. *Human Neurobiology*, **4**:219-227.

Kohonen, T. (1984) *Self Organization and Associative Memory.* Springer, Berlin.

Konishi, M. (1983) Centrally Synthesized Maps of Sensory Space. *Trends in Neuroscience*, September 1983:370-375.

Kramer, A. F., and Jacobson, A. (1991) Perceptual Organization And Focused Attention: The Role

Of Objects And Proximity In Visual Processing. *Perception & Psychophysics* (in press).

LaBerge, D. (1990) Thalamic and Cortical Mechanism of Attention Suggested by Recent Positron Emission Tomographic Experiments. *Journal of Cognitive Neuroscience*, **2**(4):358-372.

Lappin, J. (1967) Attention in the identification of stimuli in complex visual displays. *Journal of Experimental Psychology*, **75**:321-328.

Lazzaro, K., Ryckebusch, S., Mahowald, M.A., and Mead, C.A. (1989) Winner-Take-All Networks of O(N) Complexity. In: Touretzky, D.S. (Ed.) *Advances in Neural Information Processing Systems 1*, Morgan Kaufmann, San Mateo, CA.

Le Cun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., and Jackel, L.D. (1990) Handwritten Digit Recognition with a Back-Propagation Network. In: Touretzky, D.S. (ed.), *Advances in Neural Information Processing Systems 2*, Morgan Kaufmann, San Mateo, CA.

Linsker, R. (1989) How to Generate Ordered Maps by Maximizing the Mutual Information Between Input and Output Signals. *Neural Computation*, **1**:402-411.

Livingstone, M., and Hubel, D. (1988) Segregation of Form, Color, Movement, and Depth: Anatomy, Physiology, and Perception, *Science*, **240**:740-749.

Mahoney, J, V. (1987) Image Chunking: Defining Spatial Building Blocks for Scene Analysis. MIT Artificial Intelligence Laboratory Technical Report No. 980.

McClelland, J.L., and Rumelhart, D.E. (Eds.) (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MA, MIT Press.

McLeod, P., Driver, J., and Crisp, J. (1988) Visual Search for a Conjunction of Movement and Form is Parallel. *Nature*, **332**:154-155.

McLeod, P., Driver, J., Dienes, Z., and Crisp, J. (1991) Filtering by Movement in Visual Search. *Journal of Experimental Psychology: Human Perception and Performance* **17**(1):55-64.

Mead, C. (1989) *Analog VLSI and Neural Systems.* Addison-Wesley, Reading, MA.

Mel, B.W., and Koch, C. (1990) Sigma-Pi Learning: On Radial Basis Functions and Cortical Associative Learning. In: Touretzky, D.S. (ed.), *Advances in Neural Information Processing Systems 2*, Morgan Kaufmann, San Mateo, CA.

Minsky, M. and Papert, S. (1969) *Perceptrons: An Introduction to Computational Geometry.* MIT Press, Cambridge, MA.

Moran, J., and Desimone, R. (1985) Selective Attention Gates Visual Processing in the Extrastriate

Cortex. *Science* **229**. March 1985.

Mountcastle, V.B., Lynch, J.C., Georgopoulos, A., and Sakata, H. (1975) Posterior parietal association cortex of the monkey: command function for operations within extrapersonal space. *Journal of Neurophysiology,* **38**:871-907.

Mountcastle, V.B., Anderson, R.A., and Motter, B.C. (1981) The Influence of Attention Fixation Upon the Excitability of the Light-Sensitive Neurons of the Posterior Parietal Cortex. *The Journal of Neuroscience*, **1**(11):1218-1235.

Mozer, M. (1991) *The Perception of Multiple Objects: A Connectionist Approach*. MIT Press, Cambridge, MA.

Mozer, M. (1988) A Connectionist Model of Selective Attention in Visual Perception. In: *Proceedings of the 10th Annual Meeting of the Cognitive Science Society.*

Nakayama, K., and Silverman, G. (1986) Serial and Parallel Processing of Visual Feature Conjunctions. *Nature*, **320**:264-265.

Nissen, M.J. (1985) Accessing Features And Objects: Is Location Special? In: Posner, M.I. and Marin, O.S.M., (Eds.), *Attention And Performance* XI (pp. 205-219). Erlbaum, Hillsdale, NJ.

O'Connell, K.M., and Treisman, A. (1991) Abstract Coding of Orientation as a Visual Feature. Submitted to *Perception & Psychophysics*.

Obermayer, K., Ritter, H., and Schulten, K. (1991) Development and Spatial Structure of Cortical Feature Maps: A Model Study. In: Touretzky, D.S., Lippman, R. (eds.), *Advances in Neural Information Processing Systems 3*.

Pashler, H. (1988) Cross-dimensional interaction and texture segregation. *Perception & Psychophysics*, **43**(4):307-318.

Petersen, S.E., Robinson, D.L., and Keys, W. (1985) Pulvinar nuclei of the behaving rhesus monkey: Visual responses and their modulations. *Journal of Neurophysiology,* **54**:867-886.

Petersen, S.E., Robinson, D.L., and Morris, J.D. (1987) Contributions of the pulvinar to visual spatial attention. *Neuropsychology*, **25**:97-105.

Pomerleau, D. A. (1991) Rapidly Adapting Artificial Neural Networks for Autonomous Navigation. In: Lippman, R., Moody, J., and Touretzky, D.S. (eds.), *Advances in Neural Information Processing Systems 3*, Morgan Kaufmann, San Mateo, CA.

Posner, M.I., Cohen, Y., and Rafal, R.D. (1982) Neural Systems Control of Spatial Orienting. *Phil. Trans. R. Soc. Lond.* **B 298**, pp 187-198.

Posner, M.I., Walker, J.A., Friedrich, F.J., and Rafal, R.D. (1984) Effects of Parietal Injury on Co-

vert Orienting of Attention. *The Journal of Neuroscience*, **4**(7):1863-1874.

Posner, M.I., and Petersen, S.E. (1990) The Attention System of the Human Brain. *Annual Review of Neuroscience*, **13**:25-42.

Rafal, R.D., and Inhoff, A.W. (1986) Midbrain Mechanisms for Orienting Visual Attention. In: *Proceedings of the 8'th Annual Conference of the Cognitive Science Society*, Amherst, MA.

Rafal, R.D., Posner, M.I., Friedman, J.H., Inhoff, A.W., and Bernstein, E. (1988) Orienting of Visual Attention in Progressive Supranuclear Palsy. *Brain* **111**:267-280.

Rafal, R.D., Calabresi, P.A., Brennan, C.W., and Sciolto, T.K. (1989) Saccade Preparation Inhibits Reorienting to Recently Attended Locations. *Journal of Experimental Psychology: Human Perception and Performance*, **15**(4):673-685.

Ramachandran, V. S. (1988) Perceiving Shape from Shading. In: Rock, I. (Ed.) *The Perceptual World: Readings From Scientific American*, W.H. Freeman, New York.

Robinson, D.L., and McClurkin, J.W. (1989) The visual superior colliculus and pulvinar. In: Wurtz, R.H., and Goldberg, M.E. (Eds.) *The Neurobiology of Saccadic Eye Movements*. Elsevier, New York.

Rock, I., Linnett, C.M., Grant, P., and Mack, A. (1990) Results of a New Method for Investigating Inattention in Visual Perception. Paper presented at *31'st annual meeting of the Psychonomic Society*, New Orleans, LA, November, 1990.

Rose, D., and Dobson, V.G., Eds. (1985) *Models of the Visual Cortex*. John Wiley & Sons, NY.

Sandon, P. A. (1990) Simulating Visual Attention. *Journal of Cognitive Neuroscience*, **2**(3):213-231.

Schiller, P.H., and Lee, K. (1991) The Role of the Primate Extrastriate Area V4 in Vision. *Science* **251**:1251-1253, 8 March 1991.

Schiller, P.H. (1985) A Model for the Generation of Visually Guided Saccadic Eye Movements. In: Rose, D., and Dobson, V.G. (Eds.) *Models of the Visual Cortex*. John Wiley & Sons, NY.

Sejnowski, T.J. (1986) Open Questions About Computation in Cerebral Cortex. In McClelland, J.L., and Rumelhart, D.E. (Eds.) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MA, MIT Press.

Sparks, D. L. (1986) Translation of Sensory Signals into Commands for Control of Saccadic Eye Movements: Role of Primate Superior Colliculus, *Physiological Reviews*, **66** (1).

Spillman, L., and Werner, J.S. (1990) *Visual Perception: The Neurophysiological Foundations*.

Academic Press, Inc., Berkeley, CA.

Srinivas, K., and Barnden, J. (1989) Temporal Winner Take All Networks for arbitrary selection in connectionist and neural networks. In: *Proceedings of the International Joint Conference on Neural Networks*, 1989.

Suarez, H., and Koch, C. (1989) Linking Linear Threshold Units with Quadratic Models of Motion Perception. *Neural Computation*, **1**(3):318-320.

Treisman, A., and Gelade. (1980) A Feature Integration Theory of Attention. *Cognitive Psychology*, **12**:97-136.

Treisman, A., and Schmidt, H. (1982) Illusory Conjunctions in the Perception of Objects. *Cognitive Psychology*, **14**:107-141.

Treisman, A., and Paterson, R. (1984) Emergent Features, Attention and Object Perception. *Journal of Experimental Psychology: Human Perception and Performance*, **10**(1):12-31.

Treisman, A. (1988) Features and Objects: The Fourteenth Bartlett Memorial Lecture. *The Quarterly Journal of Experimental Psychology*, **40A** (2).

Treisman, A., and Gormican, S. (1988) Feature Analysis in Early Vision: Evidence From Search Asymmetries. *Psychological Review*, **95**(1):15-48.

Treisman, A., and Sato, S. (1990) Conjunction Search Revisited. *Journal of Experimental Psychology: Human Perception and Performance*, **16** (3):459-478.

Tsotsos, J. (1990) Analyzing Vision at the Complexity Level. *Behavioral and Brain Sciences*, **13**:423-469.

Ullman, S. (1984) Visual Routines. *Cognition*, **18:**97-159.

Van Essen, D., and Anderson, C. H. (1990) Information Processing Strategies and Pathways in the Primate Retina and Visual Cortex. In: *An introduction to neural and electronic networks*, Zornetzer, S.F., Davis, J.L., and Lau, C. (Eds). Academic Press, 1990.

Williams, R.J. and Zipser, D. (1988) A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. University of California at San Diego, ICS Tech Report #8805.

Wise, S.P., and Desimone, R. (1988) Behavioral Neurophysiology: Insights into Seeing and Grasping. *Science* **242**:736-740.

Wolfe, J.M., Cave, K.R., and Franzel, S.L. (1989) Guided Search: An alternative to the modified feature integration model for visual search. *Journal of Experimental Psychology: Human Perception and Performance*, **15**:419-433.

Wurtz, R.H., and Goldberg, M.E., Eds. (1989) *The Neurobiology of Saccadic Eye Movements*.

Elsevier, New York.

Yantis, S., and Jonides, J. (1990) Abrupt Visual Onsets and Selective Attention: Voluntary Versus Automatic Allocation. *Journal of Experimental Psychology: Human Perception and Performance*, **16**(1):121-134.

Yantis, S. and Johnston, J.C. (1990) On the Locus of Visual Selection: Evidence from Focused Attention Tasks. *Journal of Experimental Psychology: Human Perception and Performance*, **16**(1):135-149.

Yantis, S., and Johnson, D. (1990) Mechanisms of Attentional Priority. *Journal of Experimental Psychology: Human Perception and Performance*, **16**(4):812-825.

Yantis, S. and Jones, E. (1991) Mechanisms of Attention Selection: Temporally modulated priority tags. *Perception & Psychophysics*, in press.

Yarbus, A. (1967) *Eye Movements and Vision*. Plenum Press, New York.

Zemel, R., Mozer, M., and Hinton, G. (1989) TRAFFIC: Recognizing Objects Using Hierarchical Reference Frame Transformations. In: Touretzky, D.S. (ed.), *Advances in Neural Information Processing Systems 2*.

Zipser, D., and Andersen, R.. (1988) A Back-Propagation Programmed Network that Simulates Response Properties of a Subset of Posterior Parietal Neurons. *Nature*, **331**:679-684.

Vita

Subutai Ahmad was born on September 8th, 1965 in Calcutta, India. After spending the first eight years of his life in India, and the next five in Hong Kong, he attended Horace Greeley High School in Chappaqua, New York, graduating in 1982. He received his Bachelor of Arts with honors in 1986 from the College of Arts and Sciences, Cornell University, majoring in Computer Science with a concentration in Cognitive Psychology. He received his M.S. from the University of Illinois at Urbana-Champaign in 1988. From 1989 to 1991, he worked on his dissertation at the International Computer Science Institute in Berkeley, California. He is currently enjoying life as a guest scientist at Siemens Corporate Research Laboratory in Munich, Germany.

Publications:

Ahmad, S. and Omohundro, S. (1991). Efficient Visual Search: A Connectionist Solution. In: *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, Chicago, 1991. Also: International Computer Science Institute Technical Report TR-91-040.

Ahmad, S. and Omohundro, S. (1990). Equilateral Triangles: A Challenge For Connectionist Vision. In: *Proceedings of the 12th Annual Conference of the Cognitive Science Society*, MIT, 1990.

Ahmad, S. and Omohundro, S. (1990). A Network for Extracting the Locations of Point Clusters Using Selective Attention. International Computer Science Institute Technical Report TR-90-011.

Ahmad, S., Tesauro, G., and He, Y. (1990). Asymptotic Convergence of Back Propagation: Numerical Experiments. In: Touretzky, D. (Ed.), *Advances in Neural Information Processing Systems 2*, Morgan Kaufmann, 1990.

Tesauro, G., He, Y., and Ahmad, S. (1989). Asymptotic Convergence of Back Propagation in Single Layer Networks. *Neural Computation*, Vol. 1, No. 3, pp 382-391.

Ahmad, S., and Tesauro, G. (1989). Scaling and Generalization in Neural Networks. In: Touretzky, D. (Ed.), *Advances in Neural Information Processing Systems* 1, Morgan Kaufmann, 1989.

Ahmad, S., and Tesauro, G. (1988). Scaling and Generalization in Neural Networks: a Case Study. In: Touretzky, D., Hinton, G., and Sejnowski, T. (Eds.), *Proceedings of the 1988 Connectionists Models Summer School*, Morgan Kaufmann, 1988.

Ahmad, S. (1988). A Study of Scaling and Generalization in Neural Networks. M.S. Thesis. Technical Report UIUCDCS-R-88-1454, Dept. of Computer Science, University of Illinois at Urbana-Champaign.

Ahmad, S. (1988). Machine Learning as a Tool for Analysis in Social Sciences. Technical Report CCSR-88-16. Center for Complex Systems Research, University of Illinois at Urbana-Champaign.