to appear in **Visual Attention**, ed. by L. Harris aand M. Jenkin,
Cambridge University Press

COMPLEXITY, VISION AND ATTENTION

John K. Tsotsos

Department of Computer Science
University of Toronto

What does it mean for a problem to be *complex*? One dimension of complexity is that of *computational complexity*. This chapter focuses on how this type of complexity affects the design of perceptual systems. Many natural problems have optimal solutions that are believed to be computationally intractable in any implementation, machine or neural. Thus, the computational complexity of a particular solution greatly affects its realizability, and thus its plausibility. The focus here will be on problems in vision.

This chapter will provide:

- a brief introduction to the theory of computational complexity;
- an explanation of why this discrete theory is applicable to the study of biological systems;
- several examples of formal problems within vision which are believed to be intractable;
- guidelines for how to deal with intractable problems;
- an application of those guidelines to vision;
- an overview of a model of visual attention which results from that application.

1.0 What is Computational Complexity?

Computational complexity is studied to determine the intrinsic difficulty of mathematically posed problems that arise in many disciplines[1]. Many of these problems involve combinatorial search, i.e., search through a finite but extremely large, structured set of possible solutions. Examples include the placement and interconnection of components on an integrated circuit chip, the scheduling of major league sports events, or bus routing. Complexity theory tries to discover the limitations and possibilities inherent in a problem. In the same way that the laws of thermodynamics provide theoretical limits on the utility and function of nuclear power plants, complexity theory provides theoretical limits on information processing systems. If biological vision can indeed be computationally modeled, then complexity theory is a natural tool for investigating the information processing characteristics of both computational and biological vision systems.

Using complexity theory, one can ask: For a given computational problem, how well, or at what cost, can it be solved? Before studying complexity one must define an appropriate complexity measure. Several measures are possible, but the common ones are related to the space requirements (numbers of memory or processor elements) and

[1] See Garey & Johnson (1979) and Stockmeyer & Chandra (1979) for further discussion and background.

time requirements (how long it takes to execute) for solving a problem.

In what sense are complexity results inherent to a particular problem? Certain intrinsic properties of the universe will always limit the size and speed of computers. Consider the following argument from Stockmeyer & Chandra (1988): The most powerful computer that could conceivably be built could not be larger than the known universe (less than 100 billion light-years in diameter), could not consist of hardware smaller than the proton (10^{13} cm in diameter), and could not transmit information faster than the speed of light (3×10^8 m/s). Given these limitations, such a computer could consist of at most 10^{126} pieces of hardware. It can be proved that, regardless of the ingenuity of its design and the sophistication of its program, this ideal computer would take at least 20 billion years to solve certain mathematical problems that are known to be solvable in principle. Since the universe is probably less than 20 billion years old, it seems safe to say that such problems defy computer analysis.

1.1 Some Basic Definitions

The following are some basic definitions common in complexity theory (Garey & Johnson, 1979). A *problem* is a general question to be answered, usually possessing several parameters whose values are left unspecified. A problem is described by giving a general description of all of its parameters and a statement of what properties the answer, or *solution*, is required to satisfy. An *instance* of the problem is obtained by specifying particular values for all of the problem parameters. An *algorithm* is a general step-by-step procedure for achieving solutions to problems. To solve a problem means that an algorithm can be applied to any problem instance and is guaranteed to always produce a solution for that instance.

The time requirements of an algorithm are expressed in terms of a single variable, n, reflecting the amount of input data needed to describe a problem instance. A *time complexity function* for an algorithm expresses its time requirements by giving for each possible input length, an upper bound on the time needed to achieve a solution. If the number of operations required to solve a problem is an exponential function of n, then the problem has *exponential time complexity*. If the number of required operations can be represented by a polynomial function in n, then the problem has *polynomial time complexity*. Similarly, *space complexity* is defined as a function for an algorithm that expresses its space or memory requirements. *Algorithmic complexity* is the cost of a particular algorithm. This should be contrasted with *problem complexity* which is the minimal cost over all possible algorithms. The dominant kind of analysis is *worst-case*: at least one instance out of all possible instances has this complexity.

A worst-case analysis provides an upper-bound on the amount of computation that must be performed as a function of problem size. If one knows the maximum problem size, then the analysis places an upper bound on computation for the whole problem as well. Thus, one may then claim, given an appropriate implementation of the problem solution, that processors must run at a speed dependent on this maximum in order to ensure real-time performance for all possible inputs. Worst cases do not only occur for the largest possible problem size; rather, the worst-case time complexity function for a problem gives the worst case number of computations for any problem size; this worst case may be required simply because of unfortunate ordering of computations (for example, a linear search through a list of items would take a worst-case number of comparisons if the item sought is the last one). Thus, worst-case situations in the real world may happen frequently for any given problem size.

Critical ideas in complexity theory are that of *complexity class* and, related to it,

reducibility. If a problem S is known to be efficiently transformed (or reduced) to a problem Q then the complexity of S cannot be much more than the complexity of Q. *Efficiently reduced* means that the algorithm that performs the transformation has polynomial complexity. The *class P* consists of all those problems that can be solved in polynomial time. If we accept the premise that a computational problem is not tractable unless there is a polynomial-time algorithm to solve it, then all tractable problems belong in P.

In addition to the class P of tractable problems, there is also a major class of presumably intractable problems. If a problem is in the class *NP* then there exists a polynomial p(n) such that the problem can be solved by an algorithm having time complexity of the order of $2^{p(n)}$, i.e., the time complexity function is asymptotically (as n becomes large) dominated by the polynomial p(n). A problem is *NP-Complete* if it is in the class NP, and it polynomially reduces to an already proven NP-Complete problem. The set of such problems form an equivalence class. Clearly, there must have been a first NP-Complete problem. The first such problem was that of satisfiability (Cook's 1971 Theorem). There are hundreds of NP-Complete problems. If any NP-Complete problem can be solved in polynomial time, then they all can. Most doubt the possibility that non-exponential algorithms for these problems will ever be found, so proving a problem to be NP-Complete is now regarded as strong evidence that the problem is intrinsically intractable. If an efficient algorithm can be found for any one (and hence all) NP-Complete problems, however, it would be a major intellectual breakthrough.

1.2 Dealing with NP-Completeness

NP-Completeness effectively eliminates the possibility of developing a completely optimal and general algorithm. Once a problem is seen to be NP-Complete, it is appropriate to direct efforts toward a more achievable goal. In most cases, a direct understanding of the size of the problems of interest and the size of the processing machinery is of tremendous help in determining which are the appropriate approximations. A variety of approaches have been taken when confronted with an NP-Complete problem:

(1) Develop an algorithm that is fast enough for small problems, but that would take too long with larger problems. This approach is often used when the anticipated problems are small.

(2) Develop a fast algorithm that solves a special case of the problem, but does not solve the general problem. This approach is often used when the special case is of practical importance.

(3) Develop an algorithm that quickly solves a large proportion of the cases that come up in practice, but in the worst case may run for a long time. This approach is often used when the problems occurring in practice tend to have special features that can be exploited to speed up the computation.

(4) For an optimization problem, develop an algorithm which always runs quickly but produces an answer that is not necessarily optimal. Sometimes a worst case bound can be obtained on how much the answer produced may differ from the optimum, so that a reasonably close answer is assured. This is an area of active research, with sub-optimal algorithms for a variety of important problems being developed and analyzed.

(5) Use natural parameters to guide the search for approximate algorithms. There are a number of ways a problem can be exponential. Consider the natural parameters of a problem rather than a constructed problem length and attempt to reduce the

exponential effect of the largest valued parameters.

1.3 Vision and NP-Completeness

Are there any vision problems which are provably in the class of NP-Complete problems? There are several:

i) Unbounded Visual Search using a passive sensor system is NP-Complete (Tsotsos, 1989, 1990);
ii) Unbounded Visual Search using an active sensor system is NP-Complete (Tsotsos, 1992);
iii) Polyhedral Scene Line-labeling is NP-Complete (Kirousis & Papadimitriou, 1988)
iv) Relaxation procedures for constraint satisfaction networks are P-Complete[2] (Kasif 1990)
v) Finding a single, valid interpretation of a scene with occlusion is NP-hard[3] (Cooper 1992)
vi) Curved Object Line Labelling is NP-Complete (Dendris, Kalafatis, Kirousis 1994)
vii) Unbounded Stimulus-Behavior Search is NP-hard (Tsotsos 1995)
viii) 3D Sensor Planning is NP-Complete (Ye & Tsotsos, 1996)

It is probably true that most "general" problems dealing with perception are intractable. Using neural networks does not magically provide the answer; Judd proved a wide variety of connectionist problems to be intractable (starting with the loading problem; Judd 1990). Assuming P ≠ NP, these problems cannot be solved in their general form with realizable hardware in reasonable amounts of time, and it does not matter whether the implementation is neural or silicon-based. Many use human vision as the benchmark against which one measures "general purpose" vision capabilities. But human vision cannot be solving the general problem! (Tsotsos 1990).

The above listing only scratches the surface of the literature on the topic; there are many more examples and they form quite broad and natural problem classes. It appears that any interesting problem related to human intelligence has the characteristic that it is susceptible to combinatorial explosion.

2.0 Can Perception be Modeled Computationally?

All of the above discussion and theory is relevant to perception only if it can be shown that perception can be modeled computationally. A proof of *decidability* is sufficient to guarantee that a problem can be modeled computationally (see Davis 1958,

---

[2] P-Complete is the class of problems for which no efficient parallel algorithms can be found; that is, those problems have components which are inherently sequential. This problem is included in this list for two reasons. First, the relaxation procedure itself is very common in vision and neural network models; it is used in line-labelling algorithms for example. Second, in general, it may be that perception is inherently sequential and this proof is one piece of evidence towards this conclusion.

[3] Problems which are NP-hard are as expensive to solve as those which are NP-Complete; the NP-Complete term, however, is reserved for decision problems while NP-hard is used for all others.

Davis 1965, for in-depth discussions of decidability)[4]. If it is the case that for some problem we wish to know of each element in a countably infinite set A, whether or not that element belongs to a certain set B which is a proper subset of A, then that problem can be formulated as a decision problem. Such a problem is decidable if there exists a *Turing Machine*[5] which computes yes or no for each element of A. This requires that perception, in general, be formulated as a decision problem. This formulation does not currently exist. If no sub-problem of perception can be found to be decidable, then it might be that perception as a whole is undecidable and thus cannot be computationally modeled. But, at least one decidable perceptual problem does exist. Visual search, an important sub-problem, can be formulated as a decision problem (Tsotsos 1989) and is decidable; it is an instance of the Comparing Turing Machine defined in Yashuhara 1971. More research is needed to try to formalize other sub-problems of perception in the same way; note that all of the problems listed in section 1.3 are also decidable ones even if they are intractable. Even if some other aspect of perception is determined to be undecidable, this does not mean that all of perception is also undecidable nor that other aspects of perception cannot be modeled computationally. For example, one of the most famous undecidable problems is whether or not an arbitrary Diophantine equation has integral solutions (Hilbert's 10th problem). This does not mean that mathematics cannot be modeled computationally! Similarly, another famous undecidable problem is the halting problem for Turing Machines: it is undecidable whether a given Turing Machine will halt for a given initial specification of its tape. This too has important theoretical implications, but since Turing Machines form the foundation of computation, it certainly does not mean that computation cannot exist!

Any computational paradigm is a candidate for use in constructing a biologically plausible model. Neural network approaches are not the only ones that are biologically plausible as is often believed. Neural networks are Turing-equivalent and they are subject to the same constraints of computational complexity and computational theory as any other implementation (see Judd, 1990, for further discussion and proofs of this statement). It is important to note that relaxation processes are specific solutions to search problems in large parameter spaces and nothing more. Neural networks use variations of such search procedures which in general may be termed optimization techniques. If optimization is the process by which real neurons perform some of their computation, it is subject to precisely the same considerations of computational complexity as any other search scheme.

Many argue that worst-case analysis is inappropriate for perception. Some say that relying on worst-case analysis and drawing the link to biological vision implies that biological vision handles the worst-case scenarios. This kind of inference is quite incorrect. As was shown in (Tsotsos, 1990), it is impossible for the biological (or any other) visual system to handle worst case scenarios. The whole argument exists only to prove that all worst-case scenarios cannot be handled by human vision in a bottom-up fashion and that the quest for general solutions is futile. It has also been said that biological vision systems are designed around average or perhaps best-case assumptions. However, it is far from obvious what kind of assumptions (if any) went

---

[4] Decidability should not be confused with tractability. An intractable problem may be decidable; but for an undecidable problem, one cannot determine its tractability.

[5] A Turing Machine is a hypothetical computing device (Turing 1937). A Deterministic Turing Machine consists of a finite state control, a read-write head and a two-way infinite sequence of labelled tape squares. A program then provides input to the machine, is executed by the finite state control, and computations specified by the program read and write symbols on the squares of the tape.

into the design of biological vision systems. Vision systems emerged as a result of a complex interaction of many factors including a changing environment, random genetic mutations, and competitive behavior. It is probably the case that the best we will ever be able to do under such circumstances is to place an upper bound on the complexity of the problem, and this is all worst case analysis will provide. Finally, many say that expected case analysis more correctly reflects the world that biological vision systems see. Analyses based on expected or average cases, depend critically on having a well-circumscribed domain and an algorithm. Thus the complexity measures derived reflect algorithmic complexity and not problem complexity. Only under those conditions can average or expected case analyses be performed. In general, it is not possible to define what the average or expected input is for a vision system in the world. Furthermore, the result of the analysis will be valid only for the average input, and does not place a bound on the complexity of the vision process as a whole. This also would not provide any guidance in the determination of required processing power for real-time performance.

Complexity theory is as appropriate for analysis of perception in general as any other analysis tool currently used by biological experimentalists. Experimental scientists attempt to explain their data and not just describe it; it is no surprise that their explanations are typically well-thought-out and logically motivated, involving procedural steps or events. In this way, a proposed course of events is hypothesized to be responsible for the data observed. There is no appeal to non-determinism nor to oracles (nor should there be!) that guess the right answer nor to undefined, unjustified, or undreamed-of mechanisms that solve difficult components[6]. In essence, experimental scientists attempt to provide an *algorithm* whose behavior leads to the observed data. Attempts at providing algorithmic explanations appeared even before the invention of the computer. For example, Helmholtz' unconscious inference theory (von Helmholtz 1963) is remarkably similar to the current reasoning paradigm in artificial intelligence, where reasoning is formalized as a logical process using formal mathematics. Whether a proposed algorithm or explanation is realizable depends in part on its tractability.

3.0 Visual Search

As given earlier, visual search is one perceptual problem which has been formalized as a decision problem and is shown to be decidable. Some detail is now given regarding this formulation.

3.1 Definition

Visual search is a common if not ubiquitous sub-task of vision, in both man and machine. A basic visual search task is defined as follows (Rabbitt, 1978): given a target and a test image, is there an instance of the target in the test image? One may also ask subjects to find 'odd-man-out' elements of a display, or simply to describe an image. Typically, experiments measure the time taken to reach a correct response. Region growing, shape matching, structure from motion, the general alignment problem, connectionist recognition procedures, etc., are specialized versions of visual search in

[6] The decidability of perception has far-reaching implications for the development of perceptual theories. One of those implications is that theories which appeal to oracles are probably outside the realm of science.

that the algorithms must determine which subset of pixels is the correct match to a given prototype or description. The basic visual search task is precisely what any model-based computer vision system has as its goal: given a target or set of targets (models), is there an instance of a target in the test display? Even basic vision operations such as edge-finding are also in this category: given a model of an edge, is there an instance of this edge in the test image? It is difficult to imagine any vision system which does not involve similar operations. It is clear that these types of operations appear from the earliest levels of vision systems to the highest.

3.2 Theory

It was shown in Tsotsos (1989; 1992) that unbounded visual search (no target is given, and even if given, it cannot be used to optimize search), regardless of whether the images are time-varying or the camera system is dynamically controlled (active), is NP-Complete. This is due solely to the fact that the subset of pixels in an image which corresponds to a target cannot be predicted in advance and all subsets must be considered in the worst case. The bounded problem (the target is given in advance and is used to optimize search) on the other hand, requires linear time for the search process. This qualitatively confirms all of the visual search data that has been experimentally discovered (say, by Treisman, 1988) showing a linear response time vs number of items in display relationship. The four theorems proved in (Tsotsos 1989, 1992) show that in general, a bottom-up approach to perception (as suggested by Marr, 1982) is not only computationally intractable, but biologically implausible.

The formulation of the visual search problem proceeds as follows. There is one objective function to optimize for each known object or image event model. All objective functions are considered in parallel. The search process first seeks the image subsets that satisfy each objective function, and finally, the best match among all of these. The best match is the image subset and model which exhibits the smallest matching error and the model must explain (or cover) as much of the image subset as possible. The brute-force search strategy then is to match each objective function against all possible image subsets. Given formally, the best fit of model to data is sought such that the following is satisfied:

$$\sum_{x \in M,\, j_x \in I'} |x - j_x| < \theta \quad \text{and} \quad \sum_{x \in M,\, j_x \in I'} x \cdot j_x > \phi \qquad (1)$$

The first term is the error measure while the second is the cover measure. The input is the set I, I' is a subset of I, and M is a set of values corresponding to a particular object or event in the model base. Both I and M are retinotopic representations of the same form. θ and ϕ are two thresholds. I is not necessarily the image itself but may be a collection of all features computed from a given image. A correspondence between elements of M and elements of I' can be hypothesized where element j_X in I' is the element corresponding to x in M. Each possible combination of correspondences may be considered as a separate hypothesis.

Suppose a test image is made up of 256 pixels and a target image has 64 pixels. The correspondence required above is for each element of the target image (each pixel) to be mapped onto a unique pixel of the test image. This forms a hypothesis about where exactly in the test image the target image is believed to be represented. The spatial organization of the mapping need not preserve the structure of the target stimulus, that is, pixels chosen for the mapping may be arbitrarily distributed

throughout the image. So for this example, there are $\binom{256}{64}$ such possible, bottom-up mappings; $\frac{256!}{64!\,192!}$ is approximately 10^{56}! If spatial structure is preserved and there is no rotation or scaling of the target in the test image, then there are only 81 possibilities such that the target image is entirely within the test image.

Define selective attention as follows: Selective attention is a search optimization mechanism which tunes the visual processing machinery so that it dynamically approaches an optimal configuration, based on both data-driven and task/knowledge-driven influences. Then, attention selection may determine which mapping to attempt to verify first. If the first such mapping selected is a good one, a great deal of search can be avoided, otherwise there is the potential for a very inefficient search process. For sufficiently small images and/or massive computational power, this brute force concept will work perfectly well. For the brain, this approach fails.

3.3 Implications

If the practice of complexity analysis is to lead to tangible benefits then the theorems must lead to algorithms that must be physically realizable and the physical realization must in some way be better than others with respect to time or space efficiency. No matter what the time and space complexity functions, there is an infinite space of possible variable values or problem sizes which will not be practically realizable. The fact that all computers have finite memories is sufficient to guarantee this. One cannot in practice take infinite time to read or load an infinite Turing Machine tape. Engineering design specifications always impose constraints: the amount of memory may be limited by power consumption or cost; the number of processors is likewise constrained; real-time response places a hard constraint on time complexity and thus on problem size. These constraints cannot be ignored in any complexity discussion which may eventually be used to solve real problems. What are the constraints whose satisfaction is required in order for a theory to be biologically plausible?

Biological plausibility of a theory of visual perception will be characterized in three stages. First, a theory must be sufficient to explain the observations. Second, it is important to define the size of problem which the algorithm must be able to handle, and this follows:

- the algorithm that embodies the theory is required to accept no more than the same number of input samples of the world per unit time as human sensory organs. It is a non-trivial task to determine exactly the quantitative nature of the input to the human sensory system. With respect to the visual system, there are two eyes; each has about 110-125 million rods and 6.3-6.8 million cones; each eye can discriminate over a luminance span of 10 billion to one; the spatial resolution of the system peaks at about 40 cycles/degree while the temporal resolution peaks at about 40 Hz but the two are not independent; finally, there are many inputs from other sensory and motor areas. See (Dowling, 1987) for further discussion.
- the implementation that realizes the algorithm exists in the real world and requires amounts of physical resources which exist.
- the output behavior of the implementation as a result of those stimuli is comparable both in quality, quantity and timing to human behavior. The behavioral literature on exactly what the quality, quantity, and timing of human behavior is to a variety of stimuli is immense, but far from complete. What is required however, are responses from the algorithm that agree qualitatively and quantitatively with human responses and that are generated with the same time delays as human

responses.

The third stage of the definition requires that the functions for time and space complexity require values of their variables which lead to brain-sized space requirements and behaviorally-confirmed time requirements. Issues of polynomial vs exponential do not enter the discussion of biological plausibility at all. In other words:

- solutions should require significantly fewer than about 10^9 processors operating in parallel, each able to perform one multiply-add operation over its input per millisecond;
- processor average fan-in and fan-out should be about 1000 overall; and
- solutions should not involve more than a few hundred sequential processing steps.

Any visual theory must satisfy the above characterization; and similarly, any theory of any other aspect of intelligent behavior would have a corresponding characterization of biological plausibility.

4.0 Complexity Level Analysis of Vision

The response times measured by visual search experiments are assumed to reflect the amounts and kinds of visual information processing leading to the response. Typically one sees (for many classes of experiment) response time vs. input size graphs. Recall the definition of time complexity; time complexity is given as the cost in time as the size of the input set varies. The connection to computational complexity analysis seems rather direct.

It is clear that the brain cannot be solving the intractable form of visual search. It is equally clear that everyday vision is not always directed by a task (i.e. is not always an instance of the bounded form of visual search). It is important to not rely only on the bounded, and linear, form of visual search; the unbounded form must also be carefully considered in order to determine what kind of unbounded problem the brain may be solving. It is thus appropriate to consider the guidelines of Section 1.2. Beginning from the NP-Completeness of Unbounded Visual Search, complexity level analysis attempts to follow the guidelines presented in Section 1.2 for dealing with such difficult problems[7]. In particular, here guideline 5 is used most effectively in concert with guideline 4.

In Tsotsos (1990), a sequence of modifications to the problem of Unbounded Visual Search are given, driven by the size of the perception problem in terms of number of photoreceptors, feature types, size of visual areas, connectivity of visual areas, etc., in order to transform the problem into a tractable one. The result is a visual search problem which is tractable in time, tractable in space (requires no more processing machinery than the brain may afford), but is not guaranteed to always find optimal solutions. The solutions found are approximate ones; quite acceptable most of the time, but sometimes requiring other mechanisms (such as eye movements) or sometimes lacking in precision to some degree. The claim is that this is the form of the visual search problem which the brain is actually solving; a conclusion of this sort is the only possible one since it has been proved that optimal solutions for the general problem of visual search lead to intractability.

---

[7] Other examples of this can be found in Kirousis (1990; 1993) and Dendris, Kalafatis and Kirousis (1994). Those works begin with intractability proofs for line-labelling problems and develop efficient algorithms for several restricted sub-problems.

The key approximations and optimizations revealed by complexity level analysis are quite straightforward (obvious once laid out in this way!):

- hierarchical processing and organization - this yields a logarithmic improvement in search time within the set of models;
- localized receptive field structure - this reduces the number of image subsets to consider from a function in 2^P to a polynomial of order $P^{1.5}$;
- logically separable feature maps - which permits separate selection of relevant features;
- visual attention - which permits selection in image space and in feature space for relevant image components and the inhibition of the irrelevant.

Together, these reduce the time and space requirements of the solution to visual search. The first three seem to relate more to hardware organization. Attention, on the other hand, refers to the ability of the system to be tuned in a selective fashion to permit the hardware to be locally optimized for the current task (this view is also detailed in Maunsell, 1995; see also Desimone and Duncan, 1995, for overview of neuronal mechanisms for visual attention). The remainder of this chapter will highlight aspects of the model of visual attention which resulted from these conclusions.

5.0 The Selective Tuning Model

Complexity analysis leads to the conclusion that attention must tune the visual processing architecture to permit task-directed processing. Selective tuning takes two forms: spatial selection is realized by inhibition of irrelevant connections; and feature selection is realized by inhibition of the units which compute non-selected features. Only a brief summary is presented here (a more detailed account is in Tsotsos et al. 1995). The starting point for the model was described in the formalization of the visual search problem.

The role of attention in the image domain is to localize the set I' of Equation (1) in such a way so that any interfering or corrupting signals are minimized. In doing so, attention also seeks to increase the discriminability of a particular image subset and objective function pair over other such competing pairings as quickly as possible. The search process which localizes the image subset I' is as follows. The visual processing architecture is assumed to be pyramidal in structure with units within this network receiving both feed-forward and feed-back connections (the model has this in common with the architecture developed in Van Essen et al. 1992). When a stimulus is first applied to the input layer of the pyramid, it activates in a feed-forward manner all of the units within the pyramid to which it is connected; the result is that an inverted sub-pyramid of units and connections is activated. It is assumed that response strength of units in the network is a measure of goodness-of-match of stimulus to model and of relative importance of the contents of the corresponding receptive field in the scene.

Selection relies on a hierarchy of winner-take-all (WTA) processes. WTA is a parallel algorithm for finding the maximum value in a set. First, a WTA process operates across the entire visual field at the top layer: it computes the global winner, i.e., the units with largest response. The WTA can accept guidance for areas or stimulus qualities to favor if that guidance were available but operates independently otherwise. The search process then proceeds to the lower levels by activating a hierarchy of WTA processes. The global winner activates a WTA that operates only over its direct inputs. This localizes the largest response units within the top-level winning receptive field. Next, all of the connections of the visual pyramid that do not contribute to the winner are pruned. This strategy of finding the winners within successively smaller receptive fields, layer by layer in the pyramid and then pruning away irrelevant connections is

applied recursively through the pyramid. The end result is that from a globally strongest response, the cause of that largest response is localized in the sensory field at the earliest levels. The paths remaining may be considered the pass zone while the pruned paths form the inhibitory zone of an attentional beam. The WTA does not violate biological connectivity or time constraints.

Figure 1 shows a hypothetical visual processing pyramid. There are 4 layers, each unit connected to 7 units in the layer above it and 7 units in the layer below it. The input layer (bottom layer) is numbered 1, while the output layer (top layer) is numbered 4. The two examples which follow are intended to illustrate the structure and time course of the application of attentional selection in the model. The first example shows the structure that results if a single stimulus is placed in the visual field. The second example shows the time course of attentional selection if two stimuli are placed in the visual field.

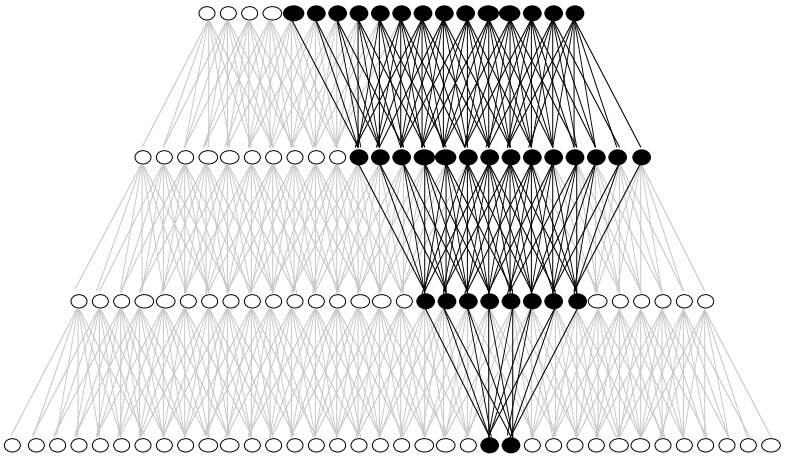

Figure 1A. A hypothetical visual processing pyramid showing the portion of the pyramid activated due to the initial feedforward stimulation of a single input stimulus.

In Figure 1A, only the feed-forward connections are shown; the feed-back connections are analogous. A stimulus which spans 2 units in the input layer is to be attended by the system; the resulting attentional beam is shown in Figure 1B. The light gray lines represent inactive connections, the dotted grey lines represent connections whose feedforward flow is inhibited by the attentional beam, and the black lines represent feedforward connections activated by the stimulus. Black units are activated solely by the black stimulus.

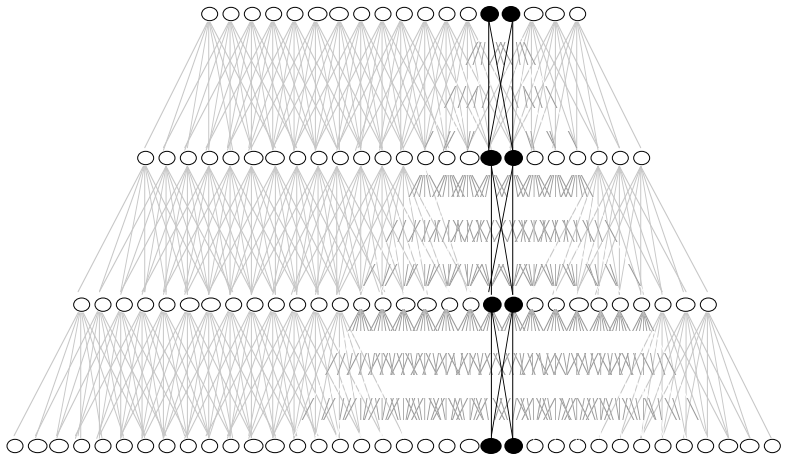

Figure 1B. The final configuration of the attentional beam reacting to a single input stimulus.

The WTA mechanism locates the peaks in the response the output layer of the pyramid, the two remainingblack units in Figure 1B. The inhibitory beam then is extended from top to bottom, pruning away the connections that might interfere with the selected units. Eventually, the two stimulus units are located in the input layer and isolated within the beam.

The important missing link in the above example is exactly how does the WTA process locate the two winners in the output layer? On the assumption that each of the units in the pyramid computes some quantity using a Gaussian weighted function across its receptive field, then the maximum responses of these computations (whatever they may be) will be exactly the two units selected in the output layer (see Tsotsos et al. 1995, for more detail on this). More importantly, with respect to attention, how does the mechanism function if there is more than one stimulus in the input, that is, with target as well as distractor elements in the visual field?

Figure 2 shows the first of a five-step sequence depicting the changes that the visual processing pyramid undergoes in such a situation. Using the same network configuration as in the previous figure and again showing only the feed-forward connections, two stimuli are placed in the visual field (input layer). They are coded black and medium gray as are the connections and units which are activated solely by them. The texture-filled units and light gray connections are those which are activated by both stimuli regardless of proportions. Note that much of the pyramid is affected by

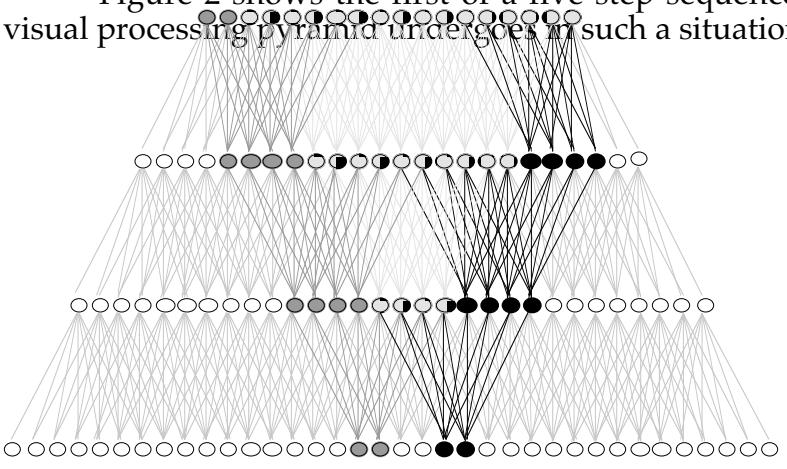

Figure 2. The visual processing pyramid at the point where the activation due to two separate stimuli in the input layer has just reached the output layer. No attentional effects are vet in evidence.

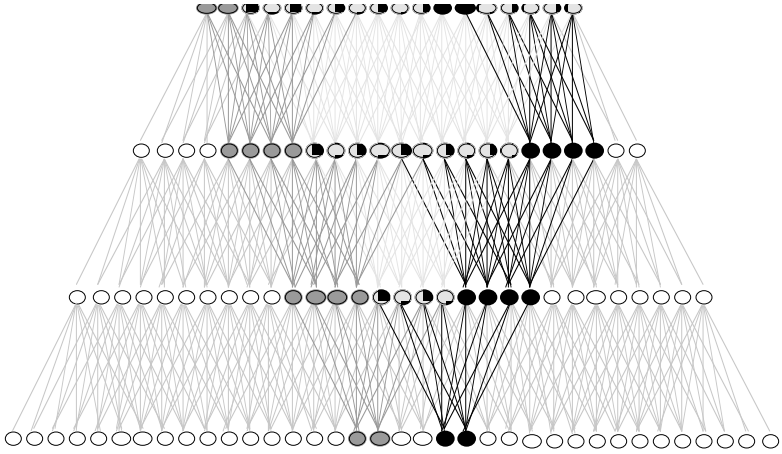

Figure 3. Attention is focussed at the location of the output layer corresponding to the location of the selected input.

Attention is focused at the black units in the output layer. This is not to say that the output of those units reflects the desired input at this point in time. Rather, these units form the root of the attentional beam as it begins its downward traversal through the visual pyramid. The next phase of the computation is to push the beam down one level further, locating the units which will be the attended ones within the beam.

The textured units response is a weak one due to the conflict that arises since each of those units 'sees' two different stimuli within its receptive field. Now the subject is directed to attend to the location of the black stimulus. Location is determined by a mechanism outside the network shown here; appropriate units corresponding to the location in the output layer are marked as shown in Figure 3.

Simultaneously, the feedforward connections from all units in the layer 3 which feed the attended units in the output layer are inhibited and are not part of the pass zone.

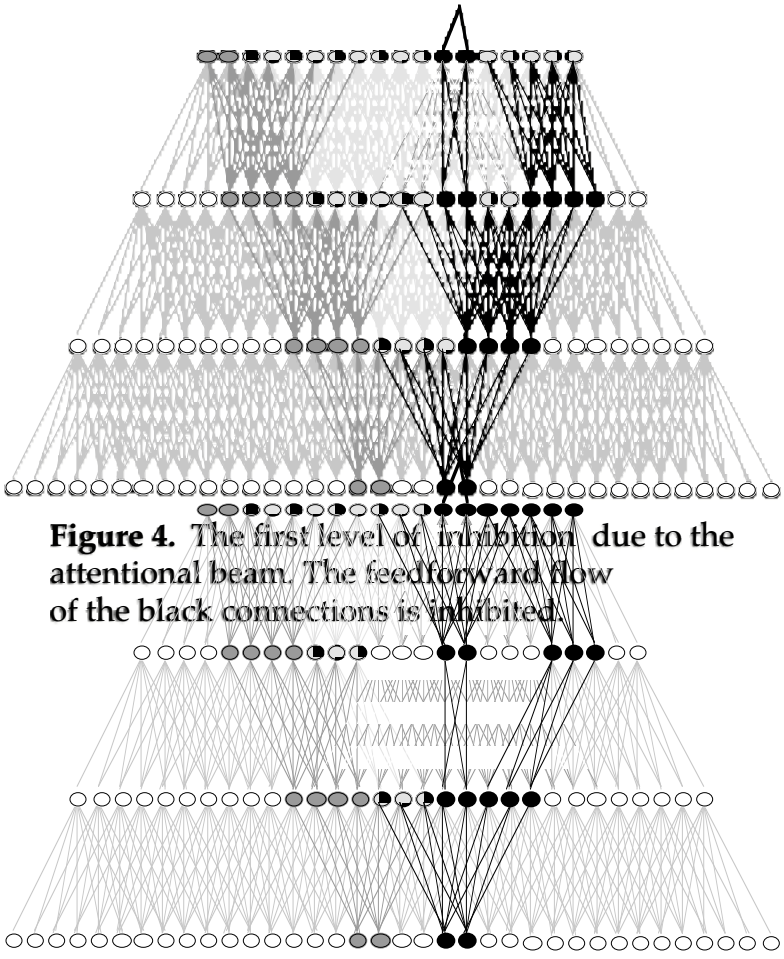

Figure 4. The first level of inhibition due to the attentional beam. The feedforward flow of the black connections is inhibited.

Figure 4 shows the connections whose feed-forward flow is explicitly inhibited by the attentional mechanism between layers 3 and 4. The interesting thing to note is that at this early stage of the application of the attentional beam, very little seems to be changing. The large scale changes come later as more of the visual pyramid is affected by the flow of the attentional beam through it. The selection of units also moves down one level to layer 3.

The next major milestone in this process is shown in Figure 5. At this point of the beam traversal, major changes can be seen. The next set of connections, those between the middle layers are inhibited. In turn, those inhibitions cause several units in 3 layer to have no active input and thus they provide no signal to the output layer. Those connections are

This change in turn causes several units in the output layer previously coded textured to be coded black, that is, they receive signals originating only from the black input stimulus.

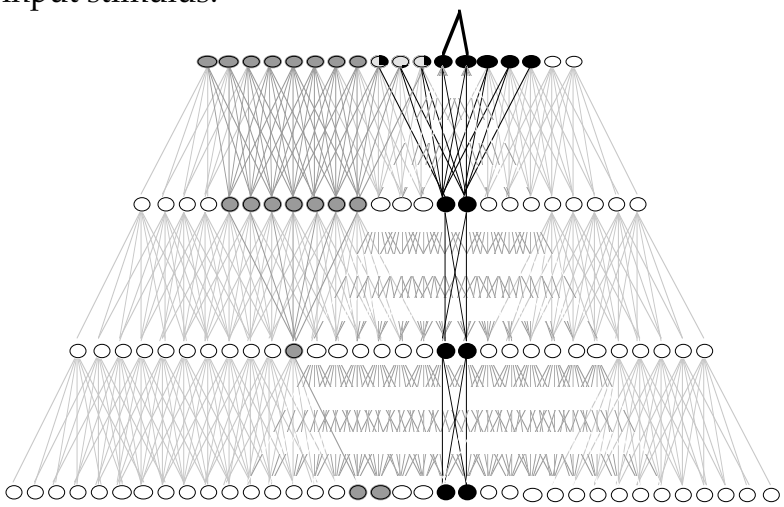

Figure 6. The third and final level of inhibition due to the attentional beam.

The final stage of the process leads to the network shown in Figure 6. After this point, the selected units in the output layer receive input only from the selected stimulus in the input layer. Note that several units in the output layer are coded in dark gray, showing that the effect of this stimulus still gets through the beam structure, in fact stronger than in the unattended case of Figure 2.

As well, a few of the weakly responding textured units still remain. The interference between stimuli evident in Figure 2 is eliminated completely with respect to the item attended, and much reduced for the unattended item. The events depicted with this set of figures would occur in the 100 to 200 ms after stimulus onset (for example, as shown in Chelazzi et al. 1993). Note the difference between the pattern of activations in Figures 1B and 6. In the former case, no location cue is given; the winner-take-all mechanism chooses the strongest responses in the output layer and inhibits the rest. Thus the set of connections and units attended forms the structure shown with the active connections being strictly those permitted by the selection mechanism. In the latter case, a location cue is given; thus, there is no inhibition within the output layer (if there were, none of the grey or textured units would survive).

Suppose one records from one of the textured 'neurons' in layer 3 of Figure 5 during the entire process. What will the response over time look like? On stimulus onset, the response will rise from zero to some level; then, as the beam is applied, will remain steady until the time corresponding to Figure 6 when the response will rise. The fact that the WTA process is not a binary one and that changes occur gradually in an iterative fashion for each level also has impact on the time course of the response. The changes in the unit's response will begin when the WTA is first applied to the configuration of Figure 5 and not only once it completes. Suppose one recorded from one of the units coloured black in layer 3 of Figure 6 throughout (this would correspond to a receptive field which was being attended). One would observe an increment in response late in the response time course here as well. This kind of increment in response well into the attentive process is observed experimentally (for example, Motter 1993; also inhibitions were reported); this is the first explanation proposed for it which also can account for inhibition effects of attention. The inhibitory effects are most clearly seen by observing the time course of units coloured white in layers 2 and 3 within the

inhibitory beam of Figure 6.

The above does not by itself account for the serial search observed in experiments of visual search. It does form, however, a part of the explanation. The rest of the explanation includes a method for inhibition of return, and for re-deployment of the beam to the next most salient targets in order. The model does in fact accomplish serial search well and these examples can be found elsewhere (Tsotsos et al. 1995). However, there is a body of single-cell recording work which this example does explain. One of those key experiments was that described by Moran and Desimone (1985). What they showed, which was surprising at the time, was that even though an effective stimulus was within a neuron's receptive field, it did not cause the neuron to fire well if the monkey was cued to attend a non-effective stimulus within the same receptive field. A more specific demonstration using the selective tuning model of that exact experimental setup appears in Tsotsos et al. (1995).

From the example above, it should be clear that the retinotopic distance between the two stimuli in the input layer is important. If the grey stimulus were one unit closer to the nlack, none of its signal would reach the output layer after the application of the attentional beam. On the other hand if it were one unit farther away, the conflict region is smaller. Since distance between stimuli is important, if the attended stimulus is near but not in the receptive field studied, the inhibitory effect of attention on the recorded neuron should be large. If it is far, the effect should disappear, and in between the inhibitory effect of attention will gradually decrease with increasing distance. This should be clear from the figures above.

Motter (1994a) concluded that the topographic representation of the neural activity in area V4 highlights potential candidates for matching to targets while minimizing the impact of any background items. In other words, the computations which create this representation seem to maximize signal-to-noise ratios for the features which are relevant to the task. Neural activity was attenuated when the stimulus did not match a cue, independent of spatial location, but was about twice as large as the attenuated value if the stimulus and cue did match. He used color and luminance as features. Interestingly, he found that neural activity was not affected due to the cueing conditions prior to presentation of stimulus arrays. This is consistent with a model which de-emphasizes connections which are not of interest. In (Motter, 1994b) he goes one step further and concludes that the attentional control system seems to be able to "shut down" the synaptic impact of all but one of many color inputs. This too is consistent with the selective tuning model and was suggested in Tsotsos (1990) as an important search optimization. Finally, Motter suggests that a sequential combination of a full field pre-attentive selection based on features which identify candidate targets, followed by a spatially restrictive focal attentive process which localizes targets, would be an interesting explanation of both his and Moran and Desimone's results; this is exactly the concept initially sketched out in Tsotsos (1990) and embodied in the selective tuning model presented here.

It is important to note that the model is implemented and runs not only in simulation but also controls the attentive behavior of a robotic stereo head. Figure 7 shows a hypothetical example of an image of letters, where the system attends to each letter in sequence. This illustrates the method using a two-dimensional example.

The selective tuning model was derived in a first principles fashion. The major contributor to those principles derives from a series of formal analyses performed within the theory of computational complexity, the most appropriate theoretical foundation to address the question "why is attention necessary for perception?" The

model not only displays performance compatible with experimental observations but also does so in a self-contained manner. That is, input to the model is a set of real, digitized images and not pre-processed data. The predictive power of the model seems broad:

• An early prediction (Tsotsos, 1990) was that attention seems necessary at any level of processing where a many-to-one mapping of neurons was found. Further, attention occurs in all the areas in concert. The prediction was made at a time when good evidence for attentional modulation was known for area V4 only (Moran and Desimone, 1985). Since then, attentional modulation has been found in many other areas both earlier and later in the visual processing stream, and that it occurs in these areas simultaneously (Kastner et al., 1998). Vanduffel and colleagues (in press) have shown that attentional modulation appears at early as the LGN. The prediction that attention modulates all cortical and perhaps even subcortical levels of processing has been borne out by recent work from several groups (e.g., Brefzynski and DeYoe, 1999; Ghandhi et al., 1999; Tootell at al., 1999).

• The notions of competition between stimuli and of attentional modulation of this competition were also early components of the model (Tsotsos, 1990) and these too have gained substantial support over the years (Desimone and Duncan, 1995; Kastner et al., 1998; Reynolds et al., 1999).

• The model predicts an inhibitory surround that impairs perception around the focus of attention (Tsotsos 1990). This too has recently gained support (Caputo and Guerra 1998; Bahcall and Kowler 1999; Vanduffel et al., in press).

• The model further implies that pre-attentive and attentive visual processing occur in the same neural substrate, which contrasts with the traditional view that these are wholly independent mechanisms. This point of view has also been gaining ground recently (Joseph et al., 1997; Yeshurun and Carrasco, 1999).

• A final prediction is that attentional guidance and control are integrated into the visual processing hierarchy, rather than being centralized in some external brain structure. This implies that the latency of attentional modulations *decreases* from lower to higher visual areas, and constitutes one of the strongest predictions of the model.

Attention is an important mechanism at any level of processing where one finds a many-to-one convergence of neural inputs and thus potential stimulus interference, a conclusion reached in (Tsotsos, 1990). This was disputed at first (Desimone, 1990); however, more recent experimental work would appear to be supportive (e.g., Kastner et al., 1998; Vanduffel et al., in press).

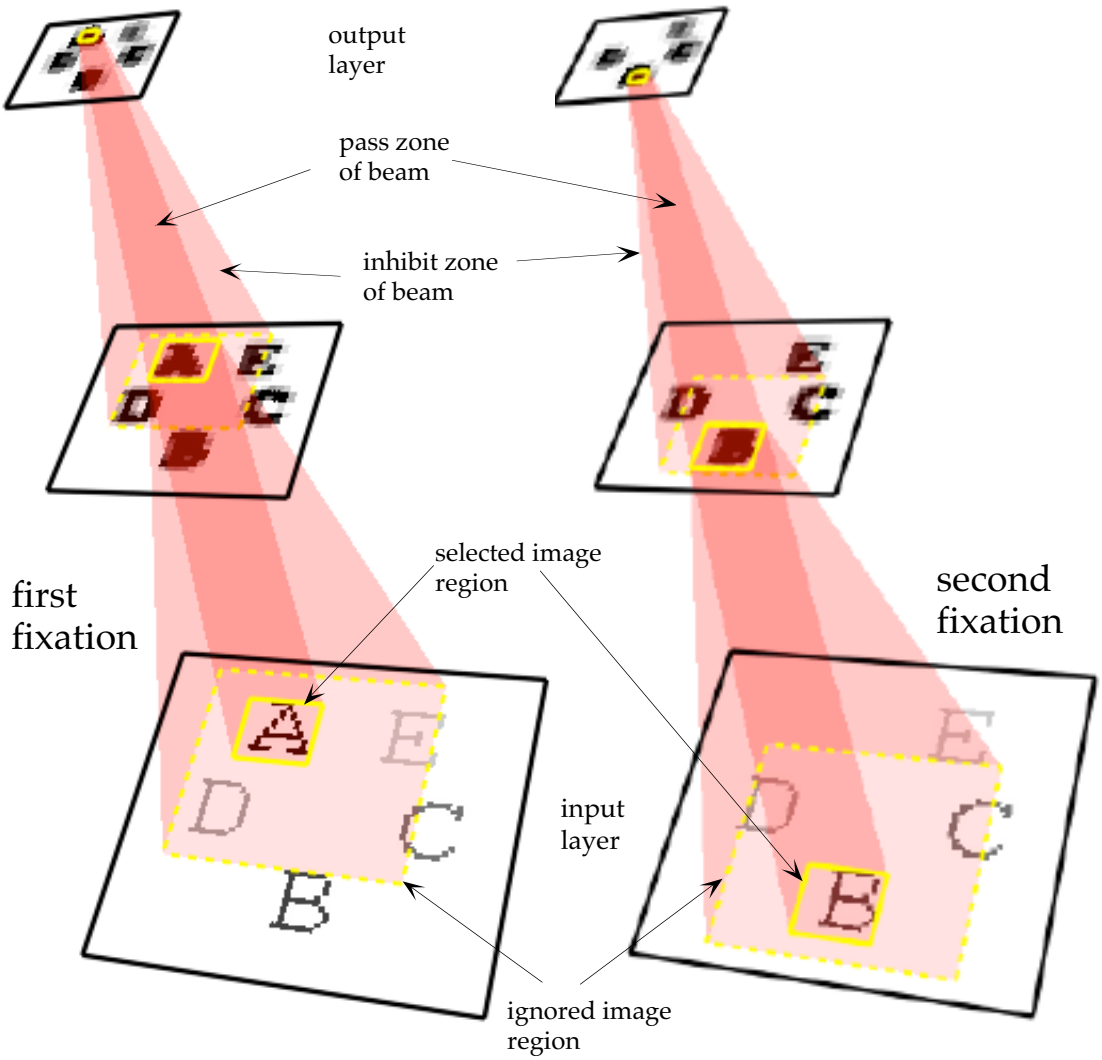

Figure 7. A hypothetical pyramid of three layers and a representation of shapes (letters) each with a different luminance. In other words, each letter is made up of pixels all of which are a uniform level of brightness and each letter is of a unique brightness. The selective tuning algorithm finds each in luminance order using the strategy described in the text. The inhibitory and pass zones of the inhibitory beam are clearly seen in the first two attentional fixations shown. The computations of each of the two layers above the input layer is based on Gaussian smoothing.

Conclusions

Marr (1982) had presented the view that a perceptual theory must be defined in three levels: the computational level, the algorithmic level and the implementation level. Further, he believed that these levels although related, could be addressed largely independently of one another. What has become very clear since Marr's classic text is that this assertion is false. The computational complexity of a proposed solution is not simply an implementation detail to worry about when coding a solution. If a proposal

at the computational level is not realizable due to inherent intractability then it serves no useful purpose. Since Marr stressed generality of solution, and since it is the case that many basic perceptual problems have already been identified as intractable, the possibility of producing unrealizable solutions following Marr's guidance is great.

The current chapter presented a brief overview of the theory of computational complexity with the goals of motivating its importance, explaining its applicability to perception in general, and arguing why this discrete theory is applicable to the study of biological systems. To strengthen the point, a number of very basic sub-problems of perception were used as examples of provably intractable problems. The ultimate claim which results is that neither the brain nor any other implementation can be solving perception in its general form in an optimal manner. A search for optimal solutions to the general problem of perception is thus arguably futile. Guidelines were presented for how to deal with intractable problems and an application of those guidelines to vision in the form of a complexity level of analysis was overviewed. The utility of such an analysis is demonstrated by the development of a model of visual attention which results from that application. A new example of the performance of that model is given which shows how attention might function in the face of conflicting stimuli from a simulated single-cell recording perspective. Several more examples using the same paradigm as well using the implemented model with real images obtained from a robot head can be found in (Tsotsos et al. 1995).

What remains? A great deal! Among other problems, complexity level analysis is expected to provide insights into:

- how much information can be extracted from a given attentional fixation?
- how are successive fixations integrated?
- how many successive fixations may be processed simultaneously?
- how does the eye movement system interact with covert attentional system?
- how is task information represented and how much of it can be used to tune the visual processing pyramid?
- how large is the visual processing pyramid in terms of numbers of layers, sizes of layers and number of units computing different visual qualities at each position?

Note that all of these open problems depend strongly on the amount of computation that can be performed in a given amount of time, or how much memory is required to store information; thus, complexity is an appropriate tool for their analysis.

Complexity analysis is by no means the most useful tool in the repertoire of the visual scientist. It is however a long neglected one, and a critical tool which can predict the realizability and performance of a given perceptual theory with respect to its neural or silicon implementation.

References

Bahcall, D., Kowler, E., (1999). Attentional Interference at Small Spatial Separations, *Vision Research 39(1)*, p 71 - 86.

Brefczynski JA, DeYoe EA, (1999). A physiological correlate of the 'spotlight' of visual attention. *Nat Neurosci*. Apr;2(4):370-4

Britten, K., (1996). Attention is Everywhere, *Nature 382*, p. 497 - 498.

Callaway, E., (1998). Local Circuits in Primary Visual Cortex of the Macaque Monkey, *Annual Review of Neuroscience 21*, p47 - 74.

Caputo, G., Guerra, S., (1998). Attentional Selection by Distractor Suppression, *Vision*

Research 38(5), p. 669 - 689.
Chelazzi, L., Miller, E., Duncan, J., Desimone, R., (1993). A neural basis for visual search in inferior temporal cortex, *Nature* , Vol. 363, p 345 - 347.
Cook, S. (1971). The complexity of theorem-proving procedures. Proceedings of the 3rd Annual ACM Symposium on the Theory of Computing, New York, NY, 151-158.
Cooper, M., (1992). **Visual Occlusion and the Interpretation of Ambiguous Pictures**, Ellis Horwood, Chicester, England.
Davis, M.. (1958). **Computability and Unsolvability,** New York.: McGraw-Hill .
Davis, M., (1965). **The Undecidable,** New York: Hewlett Raven Press.
Dendris, N.D. Kalafatis, I.A., and Kirousis, L.M., (1994). An efficient parallel algorithm for geometrically characterising drawings of a class of 3-D objects, *Journal of Mathematical Imaging and Vision 4* 375--387.
Desimone, R., (1990). Complexity at the Neuronal Level, *Behavioral and Brain Sciences 13(3)*, p 446.
Desimone, R., Duncan, J., (1995). Neural Mechanisms of Selective Attention, *Annual Review of Neuroscience 18*, p193 - 222.
Dowling, J. (1987). **The Retina: An Approachable Part of the Brain**, Harvard University Press, Cambridge, Massachusetts.
Gandhi SP, Heeger DJ, Boynton GM, (1999). Spatial attention affects brain activity in human primary visual cortex, Proc Natl Acad Sci U S A 1999 Mar 16;96(6):3314-9
Garey, M., & Johnson, D. (1979). **Computers and Intractability: A Guide to the Theory of NP-Completeness.** San Francisco: Freeman.
Joseph, J., Chun, M., Nakayama, K., (1997). Attentional Requirements in a 'Preattentive' Feature Search Task, *Nature 387*, p. 805 - 807.
Kastner, S., De Weerd, P., Desimone, R., Ungerleider, L., (1998). Mechanisms of Directed Attention in the Human Extrastriate Cortex as Revealed by Functional MRI, *Science 282*, p108 - 111.
Judd, J. S., (1990). **Neural network design and the complexity of learning.** Cambridge, MA: M.I.T. Press.
Kasif, S., (1990). On the parallel complexity of discrete relaxation in constraint satisfaction networks, *Artificial Intelligence 45*, p 275 - 286.
Kirousis, L.M. (1990) Effectively labeling planar projections of polyhedra, *IEEE Trans. Pattern Anal. Mach. Intell. 12* 123--130.
Kirousis, L.M. (1993). Fast parallel constraint satisfaction, *Artificial Intelligence 64* 147--160.
Kirousis, L.M., & Papadimitriou, C. (1988). The complexity of recognizing polyhedral scenes. *Journal of Computer and System Sciences*, 37, 14-38.
Marr, D. (1982). **Vision: A computational investigation into the human representation and processing of visual information.** San Francisco: Freeman.
Maunsell, J., (1995). The brain's visual world: Representation of visual targets in cerebral cortex, *Science 270 (5237)*, p 764 - 769.
Moran, J., Desimone, R. (1985). Selective attention gates visual processing in the extrastriate cortex, *Science 229*, p 782 - 784.
Motter, B., (1993). Focal attention produces spatially selective processing in visual cortical areas V1, V2 and V4 in the presence of competing stimuli, *J. Neurophysiology 70(3)*, p 909 - 919.
Motter, B., (1994a). Neural correlates of attentive selection of color or luminance in extrastriate area V4, *J. Neuroscience.14(4)*, p 2178 - 2189.
Motter, B., (1994b). Neural correlates of feature selective memory and pop-out in

extrastriate area V4, *J. Neuroscience.14(4),* p 2190 - 2199.
Rabbitt, P. (1978). Sorting, categorization and visual search. In E. Carterette, & M. Freidman (Eds.), **Handbook of Perception: Perceptual Processing,** (Vol. IX, pp. 85 - 136). New York: Academic Press.
Reynolds, J., Chelazzi, L., Desimone, R., (1999). Competitive Mechanisms Subserve Attention in Macaque Areas V2 and V4, *The Journal of Neuroscience, 19(5)*, p1736–1753
Schall, J., Hanes, D. (1993). Neural basis of saccade target selection in frontal eye field during visual search, *Nature 366*, p 467 - 469.
Stockmeyer, L. & Chandra, A. (1988). Intrinsically difficult problems. *Scientific American Trends in Computing,* (Vol. 1, pp. 88 - 97), New York: Scientific American Inc.
Tsotsos, J. K. (1989). The Complexity of Perceptual Search Tasks. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, Detroit, Michigan, 1571-1577.
Tsotsos, J. K. (1990). A Complexity Level Analysis of Vision. *Behavioral and Brain Sciences*, 13, 423-455.
Tsotsos, J. K. (1992). On the relative complexity of active vs passive visual search. *International Journal of Computer Vision*, 7, 127-141.
Tsotsos, J. K. (1995). Behaviorist intelligence and the scaling problem. *Artificial Intelligence 75,* p 135 - 160.
Tsotsos, J. K., Culhane, S., Wai, W., Lai, Y., Davis, N., Nuflo, F. (1995). Modeling visual attention via selective tuning, *Artificial Intelligence.*
Turing, A. (1937). On computable numbers with an application to the Entscheidungs problem. *Proceedings of the London Mathematical Society*, 2(43), 230-265.
Yashuhara, A., (1971). **Recursive Function Theory and Logic**, New York: Academic Press.
Ye, Y., Tsotsos, J.K., (1996) Sensor planning in 3D object search: its formulation and complexity, Proc. Int. Symposium on Artificial Intelligence and Mathematics, Fort Lauderdale, January 3-5.
Vanduffel, W., Tootell, R., Orban, G. (in press). "Attention-dependent suppression of metabolic activity in the early stages of the macaque visual system, *Cerebral Cortex.*
Van Essen, D., Anderson, C., Felleman, D. (1992). Information processing in the primate visual system: An integrated systems perspective, *Science 255(5043),* p 419 - 422.
von Helmholtz, H., (1963) in J.P.C.S. Southall (Ed.), **Handbook of Physiological Optics,** New York: Dover, (Originally published: 1867).
Yeshurun Y, Carrasco M, (1999). Spatial attention improves performance in spatial resolution tasks, *Vision Res*. Jan;39(2):293-306