THE COMPLEXITY OF VISUAL SEARCH TASKS

John K. Tsotsos

Dept. of Computer Science, and
Centre for Vision Research
York University,
Toronto, Ontario, Canada

THEORETICAL FOUNDATIONS

One of the most frustrating things about studying attention is that research is so often is accompanied by vague discussions of *capacity limits, bottlenecks, resource limits, allocations of attentional resources*, and the like. How can these notions be made more concrete? The sub-area of computer science known as Computational Complexity is concerned with the theoretical issues dealing with the cost of achieving solutions to problems in terms of time, memory and processing power as a function of problem size. How much of attentional behaviour can be explained using this viewpoint?

A reasonable question to ask is does computational complexity have relevance for real problems, particularly where neurobiology is concerned? Stockmeyer & Chandra[1]present a compelling argument for the relevance of complexity. 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 x 108 m/s). Given these limitations, such a computer could consist of at most 10126 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. There do exist real problems for which this argument applies (see Garey & Johnson[2] for a catalog).

With respect to neurobiology, many have considered complexity constraints in the past, including Simon[3], Uhr[4] , Feldman & Ballard[5] , Tsotsos[6,7] , Judd[8] . All roads lead to the same conclusions: the brain cannot fully process all stimuli in parallel in the observed response times. But this is like saying there is a capacity limit :*This does not constrain a solution*

Can Human Vision be Modeled Computationally?

This is a non-trivial question often asked and it is important that an answer be provided. The issue is important because if it could be proved that human brain processes cannot be modelled computationally (and this does not refer to a computer's hardware architecture only), then a lot of people should shift research focus. Computer science is the discipline that deals with such questions. What we know is that a proof of *decidability* is sufficient to guarantee that a problem can be modeled computationally (Davis [9,10]). This requires that vision be formulated as a decision problem and that a Turing Machine is defined to provide solution. This formulation does not currently exist. If no sub-problem of vision can be found to be decidable, then it might be that perception as a whole is undecidable and thus cannot be computationally modeled. However, at least one decidable perceptual problem does exist. Visual search can be formulated as a decision problem (Tsotsos[11]) and is decidable; it is an instance of the Comparing Turing Machine (Yashuhara[12]).

This however is not a proof that human vision can be modeled computationally. All this says is that one sub-problem, albeit a ubiquitous one, can indeed be modelled on a computer. What if there are other undecidable vision sub-problems? Even if some other aspect of vision is determined to be undecidable, this does not mean that all of vision is also undecidable or that other aspects of perception cannot be modeled computationally. Hilbert's 10th problem in mathematics or the halting problem for Turing Machines are two examples of famous undecidable problems. The former does not imply that mathematics is not possible while the latter does not mean that computers are impossible. It seems that most domains feature both decidable as well as undecidable sub-problems and these co-exist with no insurmountable difficulty. Decidability should thus not be confused with tractability. Tractability refers to the sort of problem Stockmeyer and Chandra described earlier, namely, can enough resources be found and enough time so that the problem can be solved reasonably. An intractable problem may be decidable; but for an undecidable problem, one cannot determine its tractability. Intractable problems are those (we believe!) that have exponential complexity in space and/or time; there are several classes of such problems with differing characteristics. NP-Complete is one of those.

In order to show the decidability of visual search, two theoretical abstract problems have been defined and proofs of their complexity were presented[11]. The first is *Unbounded*

Visual Search. This was intended to model recognition where no task guidance to optimize search is permitted. It corresponds to recognition with all top-down connections in the visual processing hierarchy removed or disabled. In other words, this is pure data-directed vision, as Marr believed was possible[13]. The second problem is *Bounded Visual Search.* This is recognition with knowledge of a target and task in advance, and that knowledge is used to optimize the process. The basic theorems proved by Tsotsos[11] and then later confirmed by Rensink[14] are:

Theorem 1: Unbounded Visual Search is NP-Complete.

Theorem 2: Bounded Visual Search has time complexity linear in the number of test image pixel locations.

The results are broad and powerful. The first tells us that the pure data-directed approach to vision (and in fact to perception in any sensory modality) is computationally intractable in the general case. Marr's 'in principle' solution to vision can not be put into practice and his implementation level of description is not feasible[13]. The second tells us that visual search takes time linearly dependent of the size of the input, something that has been observed in a huge number of experiments.

If a problem is NP-Complete, how is it possible to deal with it? A classic volume by Garey & Johnson[2] provides the guidance we seek. NP-Completeness eliminates the possibility of developing a completely optimal and general algorithm. A direct understanding of the size of the problems of interest and the size of the processing machinery may help in determining the appropriate approximations.

(1) Develop an algorithm that is fast enough for small problems, but that would take too long with larger problems (good if anticipated problems are small).

(2) Develop a fast algorithm that solves a special case of the problem, but does not solve the general problem (assumes special cases have 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.

(4) For an optimization problem, develop an algorithm that always runs quickly but produces an answer that is not necessarily optimal.

(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.

For the vision problem as defined here, strategy 5 will be our guide.

Many objections can be raised over this analysis and these must be addressed. Perhaps the most obvious objection is that an analysis relying on worst-case analysis and drawing the link to biological vision implies that biological vision handles the worst-case scenarios. It should be pointed out that 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.

A different sort of objection has its roots in evolutionary hypotheses. Some claim that biological vision systems are designed around average or best-case assumptions; in any case, expected case analysis more correctly reflects the world biological systems. As well be seen in the following paragraphs, there is some theoretical evidence that there is no difference between worst-case and median-case analysis in vision.

Parodi, Lancewicki, Vijh & Tsotsos[15] addressed the criticism that the brain did not evolve to solve worst case problems. They developed algorithm for generating random instances of polyhedral scenes (trihedral and LegoLand). They examined median case complexity of labelling the scenes with a number of algorithms. The key results are the following:

A. blind depth-first search with backtracking, AC-1 relaxation has median-case time complexity that is exponential in the number of junctions

B. informed best-first search with backtracking, AC-1 relaxation has median-case time complexity that is linear in number of junctions.

These results were achieved empirically with the following experimental strategies.

In the first experiment, they investigated the median-case complexity of “blind” search”, that is, simple depth-first search with backtracking. They achieved arc-consistency by the relaxation algorithm AC-1 of Mackworth and Freuder [16] and then performed a depth-first search on the constraint network. Arc consistency requires that for each pair of adjacent junctions J1 and J2, remove all labelings of J1 for which there is no labeling in J2 which is compatible with it, and vice versa. Repeat the operation for all pairs of junctions until we have gone through all the pairs of junctions once without deleting a single labeling. This procedure takes in the worst case time O(n 2),where n is the number of segments (or of junctions) in the line drawing.

Time for the search stage is computed as the number of times that the depth-first-search stack containing all nodes which have been visited (that is, touched at least once) but not explored (that is, such that all the nodes adjacent to them have not been visited) is

updated. 100 different random scenes were generated[1] for each size of scene (total number of junctions in the scene) and the median number of algorithmic steps computed for each set. The result was fit to a straight line in logarithmic space as a function of number of scene junctions, thus demonstrating exponential behavior even in the median case.

In the second experiment, they tested the median case complexity for informed best-first search. This search exploits the following heuristic rules:

• At a given point during the search, nodes can be subdivided into three classes: unvisited, visited, and explored. Unvisited nodes are those that have yet to be considered by the search; visited nodes are those that have been touched but the nodes that they are adjacent to have not been visited yet; explored nodes are those that have been visited and the nodes adjacent to them have been visited as well. The nodes that have been visited have already been labeled. The set of those nodes which are adjacent to visited nodes is the set in which the next node to be tentatively labeled must be chosen. Best-first search chooses the node that has the smallest set of legal labelings, breaking ties arbitrarily.

• T-junctions have six possible labelings. Four of them are common to the basic and the extended trihedral world, the other two only appear in the extended trihedral world, and they appear more seldom. Therefore, it is convenient to try first the "conventional" labelings and only successively the remaining two labelings. For E-junctions, we try first the labeling with the middle-segment labeled as a convex segment and the remaining segments labeled as arrows. Thus the structure of the domain is used to guide search.

This is not the same as domain knowledge; it is more closely related to Marr's and Richard's natural constraints.

As a result of these two experiments (plus more that can be found in the paper), the objection that worst-case analysis is not appropriate is defused. Even in median case, they have shown that complexity is exponential if no knowledge is used to tune search while even very modest amounts of general knowledge can convert an exponential process to a linear one.

CONSTRAINTS ON A MODEL

In Tsotsos [6,7], 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 that the brain is actually solving; a conclusion of this sort is the only possible one since it has been proved

[1] See Parodi et al. [15] for the algorithm that generates random scenes of polyhedral line drawings.

that optimal solutions for the general problem of visual search lead to intractability.

In unbounded visual search problem, time complexity is $O(N2^{PM})$. The natural parameters of this computation are N (number of prototypes), P (size of image in pixels) and M (number of features computed at each pixel).

N - very large - any reduction leads to linear improvements
P - photoreceptors - any reduction leads to exponential improvements
M - number of visual features - any reduction leads to exponential improvements

Architecture

This can be reduced to at least $O(P^{1.5}\ 2M\ \log_2 N)$ using a number of simple optimizations and approximations:

1. Hierarchical organization takes search of model space from $O(N)$ to $O(\log_2 N)$

2. Pyramidal abstraction - rather than searching a full image, the search takes place over a smaller more abstract image, and is then refined locally (idea due to Leonard Uhr[4]) P is reduced.

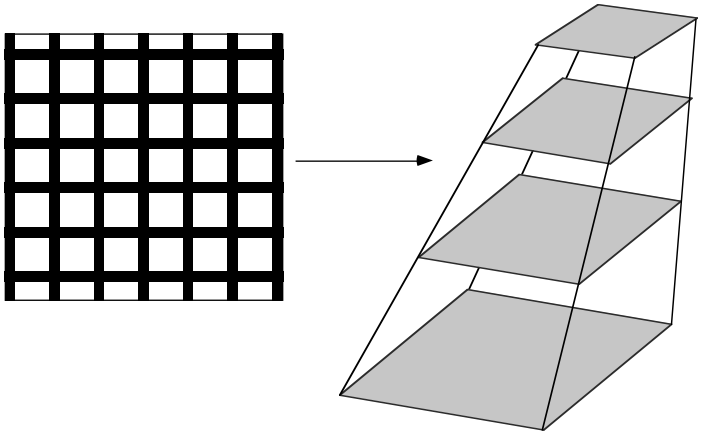

Figure 1. If images are inspected using a coarse-to-fine strategy, as exemplified by a pyramidal representation, the number of units searched may be significantly reduced.

3. Logically (not necessarily physically) separate visual maps - permits selection of features of relevance and thus reduces M

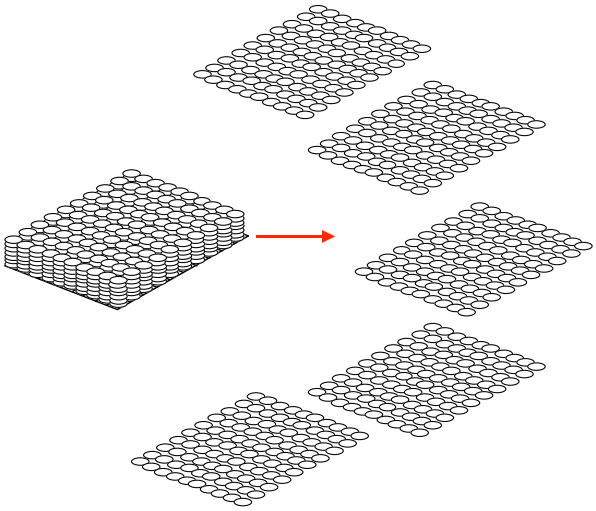

Figure 2. Rather than have all features retinotopically organized and addressable only by location, logical separation into feature maps would permit access by location and by feature type.

4. Spatio-temporally localized receptive fields reduce number of possible receptive fields from 2^P to $O(P^{1.5})$ (this assumes a hexagonal grid, and hexagonal, contiguous receptive fields of all sizes and centered at all locations in the image array)

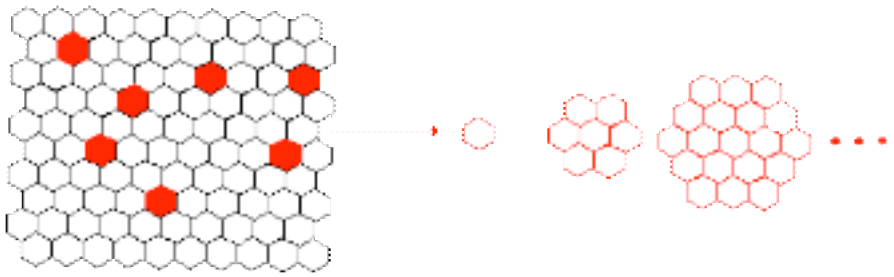

Figure 3. If a processing strategy sacrifices arbitrary receptive fields for locally spatial ones only, enormous savings in complexity of search are possible. This is at the expense of fine grain selection of locations. The figure shows this difference.

After applying the 4 constraints, $O(P^{1.5}\ 2^M\ \log_2 N)$ is the worst-case time complexity. Where does attention come in? Attention can further reduce the $P^{1.5}$ term if one selects the receptive field that is to be processed. This is not a selection of location, but rather a selection of a local region and its size at a particular location. This represents spatial selectivity. Using feature selectivity can reduce the M term, that is, which subset of all possible features actually is represented in the image or is important for the task at hand. Object selectivity can reduce the N term, reflecting once again task-specific information.

A final source of constraints is due to connectivity limits. Suppose a unit in the output layer requires connection to each receptive field in the input layer. Say the input layer has 10^6 pixels. Then, there would be 10^9 receptive fields in this layer using the above

calculations. Thus, connectivity between the output and input layers is much too high to be biologically plausible.

Information Routing

It is important to consider how information moves around in this pyramidal representation. Data is represented at low resolution in output layer, and high resolution at the input layer. Connectivity in terms of neural connections is not unlimited and without cost. The Minimum Cost Principle suggest that overall connectivity should be minimized. One strategy that satisfies this principle is that units at the output layer achieve access to high-resolution representations through the pyramid rather than connecting directly to the layer of interest.

Pyramidal representations have an additional problem - signal interference of at least 2 forms. The first type is the result of interference within a sub-pyramid and is solved by eliminating interfering signal. The second is interference between sub-pyramids and is solved by using a single attentional focus sequentially attending to each item. Finally, there is the ambiguity resulting from convergence

Vision Problem is Re-Shaped

The brain is **not** *solving the general vision problem!* Why is the problem solved by the new architecture not the same as the general visual search problem?

1. Hierarchical organization - Although this will optimize search, it does not affect the nature of the problem solved.
2. Pyramidal abstraction - This form of abstraction does affect the problem through the loss of information and signal combination.
3. Separate visual maps - there is no effect on the nature of the problem.
4. Spatio-temporally localized receptive fields - arbitrary combinations of locations dis-allowed, forced to look at features across a receptive field instead of finer grain combinations
5. Attention further limits what is processed in space and in feature domain

Let us also not forget the purpose of vision - if there is no output from the system (action, response), it is difficult if not impossible to know what has been perceived. In order to produce a response, perfect percepts are not necessarily required. “Good enough” percepts may suffice. This means that in many cases, if not most, the vision system need only compute sufficiently correct percepts as required by the task at hand.

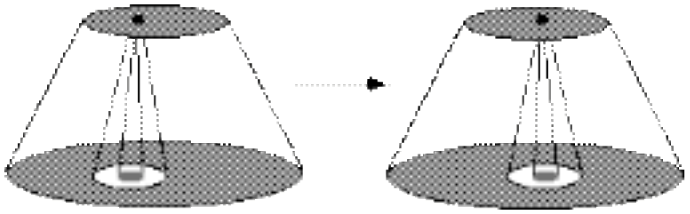

Figure 5. The top part of the figure shows a hypothetical 'neuron', say in one of the higher order visual areas, whose receptive field spans a circle of stimulus elements such as those commonly found in a typical visual search experiment. If the subject is to respond to one particular element, but neurons see a convergence of all the elements, the response of the neuron is necessarily ambiguous. The bottom part of the figure shows the kind of attentional inhibition that would remedy this ambiguity by eliminating the convergence of irrelevant inputs to that neuron.

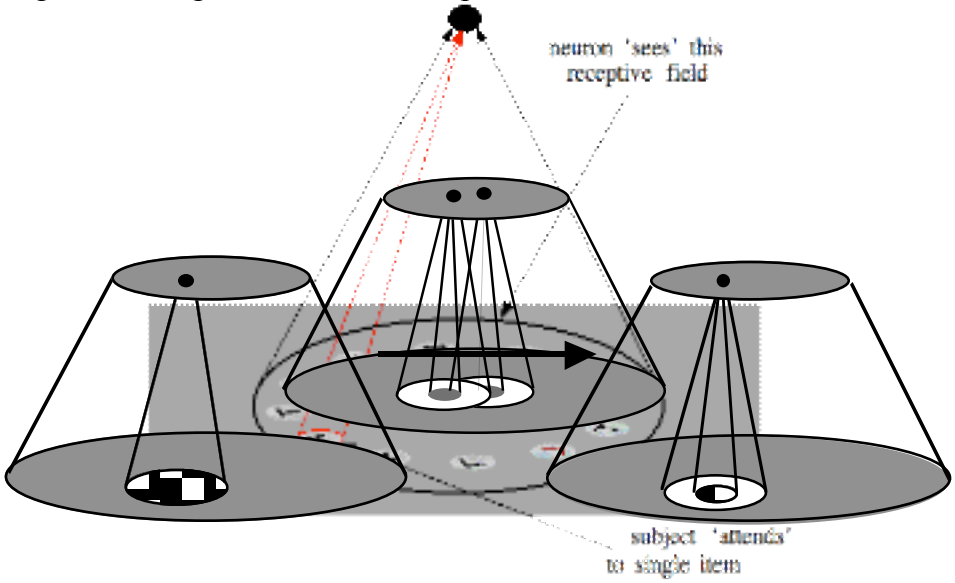

Figure 6. The top part of this figure shows how two stimuli in the input produce interfering activation cones leading to poor responses. The bottom half of the figure symbolizes a solution, namely selective attentive selection of stimuli that eliminates this problem.

CONCLUSIONS

This research began with the goal of trying to make notions such as capacity limits concrete, and tried to discover constraints on plausible solutions to vision. Some progress was made. It is important to stress that this viewpoint is not intended to address all aspects of attention. Certainly a precise definition of capacity is not provided; however it is possible through the proofs and architectural changes to find architectures that plausibly do not violate biological constraints on numbers of neurons and connections. Important problems such as routing and interference can thus be addressed and a new rationale for the existence of attention is shown. Perhaps the most important conclusion is that the problem the brain is actually solving is not the generic vision problem. That is intractable. Rather,

the problem is re-shaped through approximations so that it becomes solvable by the amount of processing power available for vision. There is much more to be done in this regard, however, the Selective Tuning Model for Visual Attention resulted from these 'in principle' investigations.

REFERENCES

1. L. Stockmeyer, A. Chandra. Intrinsically difficult problems. *Scientific American Trends in Computing,* (Vol. 1, pp. 88 - 97), New York: Scientific American Inc., (1988).
2. M. Garey, D. Johnson. **Computers and Intractability: A Guide to the Theory of NP-Completeness.** San Francisco: Freeman. (1979).
3. H. Simon. The architecture of complexity, Proc. American Philosophical Society 106, 467 - 482. (1962).
4. L. Uhr. Layered `recognition cone' networks that preprocess, classify and describe, *IEEE Transactions on Computers*, 758-768. (1972).
5. J. Feldman, D. Ballard. Connectionist models and their properties, Cognitive Science 6, 205 - 254. (1982).
6. J.K. Tsotsos. "A `Complexity Level' Analysis of Vision", Proceedings of International Conference on Computer Vision: Human and Machine Vision Workshop, London, England, June (1987).
7. J.K. Tsotsos. A Complexity Level Analysis of Vision. *Behavioral and Brain Sciences*, 13, 423-455. (1990).
8. J.S. Judd. **Neural network design and the complexity of learning.** Cambridge, MA: M.I.T. Press. (1990).
9. M. Davis. **Computability and Unsolvability,** New York.: McGraw-Hill . (1958).
10. M. Davis. **The Undecidable,** New York: Hewlett Raven Press. (1965).
11. J.K. Tsotsos. The Complexity of Perceptual Search Tasks. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, Detroit, Michigan, 1571-1577. (1989).
12. A. Yashuhara. **Recursive Function Theory and Logic**, New York: Academic Press. (1971).
13. D. Marr. **Vision: A computational investigation into the human representation and processing of visual information.** San Francisco: Freeman. (1982).
14. R. Rensink. A new proof of the NP-Completeness of Visual Match, TR 89-22, Dept. of Computer Science, University of British Columbia. (1989).
15. P. Parodi, R. Lancewicki, A. Vijh, A., J.K. Tsotsos. "Empiricaly-Derived Estimates of the Complexity of Labeling Line Drawings of Polyhedral Scenes", *Artificial Intelligence105*, p47 - 75, (1998).
16. A.K. Mackworth, E.C. Freuder. The complexity of some polynomial network consistency algorithms for constraint satisfaction problems, *Artificial Intelligence 25* p65–74. (1985).