ON READING SKETCH MAPS

Alan K. Mackworth
Department of Computer Science
University of British Columbia
Vancouver, B.C., Canada V6T 1W5

Abstract

A computer program, named MAPSEE, for interpreting maps sketched freehand on a graphical data tablet is described. The emphasis in the program is on discovering cues that invoke descriptive models which capture the requisite cartographic and geographic knowledge. A model interprets ambiguously the local environment of a cue. By resolving these interpretations using a new network consistency algorithm for n-ary relations, MAPSEE achieves an interpretation of the map. It is demonstrated that this approach can be made viable even though the map cannot initially be properly segmented. A thoroughly conservative, initial, partial segmentation is described. The effects of its necessary deficiencies on the interpretation process are shown. The ways in which the interpretation can refine the segmentation are indicated.

1. Introduction

The purpose of this paper is to report on a program, MAPSEE, that reads sketch maps. The intention is not to discuss the overall goals of this research nor how it fits into current computational vision concerns except insofar as it directly impinges on them. Those issues are tackled in detail in a companion paper (Mackworth, 1977). Suffice it to say here, by way of introduction, that one of the goals is to understand how to exploit the semantics of images designed for communication as typified by sketches, in general, and sketch maps in particular.

Another goal is to transfer some of the current vision paradigm to other domains. One of the useful concepts to emerge from" earlier work was an approach to vision as a task of understanding the implications of local cues invoking models that placed constraints on the interpretation of picture elements in the neighbourhood of the cue. The Huffman-Clowes-Waltz approach (Waltz, 1972), for example, used junctions as cues, and corners as models with the constraints placed on the edges at the corners, while POLY (Mackworth, 1973, 1976) focussed on edges and surfaces. One purpose in designing MAPSEE was to demonstrate that the constraint satisfaction approach has much wider applicability than just the blocks world. This required, in part, further generalization of the so-called network consistency algorithms

Thus one focus of the current work i: to explore the limits of the cue/descriptive model approach to visior with particular emphasis on the modularity that it buys. Another focus is an aspect of the chicken-and-egg problem (Mackworth 1975b) namely, can one segment before interpreting? If so, how? - given that t complete segmentation requires prioi interpretation. In this domain, and ir many others I suspect, the semantics ar< so rich that a partial segmentation thai is <u>conservative</u> in many different ways i! sufficient to allow a bootstrap into ar interpretation. By 'rich semantics'] mean simply that there exists a large number of partially independent but mutually confirming inference paths. Furthermore, the initial interpretatior can then, in turn, refine the initial partial segmentation. (See, for example, (Yakimovsky and Feldman, 1973), (Tenenbaun and Barrow, 1976) and (Starr anc Mackworth, 1976) for other approaches tc this problem.)

2. <u>The Maps</u>

The maps chosen for this study were sketched free-hand on a graphical data tablet. No great effort was made to drav the map carefully. The map shown in Figure 1 gives many people pause before they <u>see</u> that it depicts an island on

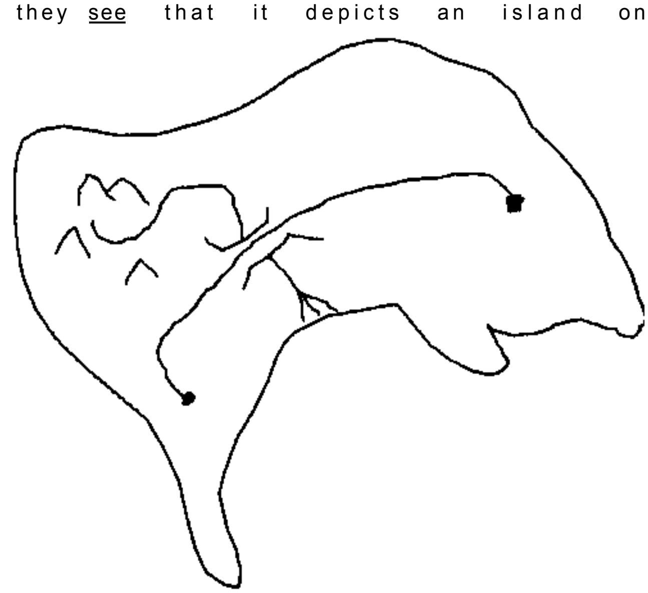


Figure 1. A Typical Sketch Map

which there are two towns connected by a road which crosses a bridge over a river which rises in a mountain range in the north-west, and runs to a delta in a bay on the southern shore. The only major possible geographical elements allowed by the current MAPSEE but missing from that map are inland lakes. Moreover, the land area need not be an island - it could cover the entire map. The cartographic elements may be arranged in any of the legal ways their corresponding geographic objects could.

3. Interpretation in Context: Cues and Models

To understand the general nature of MAPSEE the following experiment is suggested. Cut a small hole in a piece of paper and place it on the map. As you move it around the map ask yourself "What could that be?" Initially, if you're looking at a line then clearly it could be a road, a river (flowing in one direction or the other), a bridge, a mountainside or a shoreline (of a lake or of the sea, with the water on one side or the other). If on the other hand, you see a blank space, an areal element, it could be land, lake or sea. If you now temporarily remove the paper with the hole in it and see the map as a whole, you will notice that the lineal elements appear to aggregate into units of connected lines each with a uniform interpretation. These are chains. Similarly, the areal elements will aggregate into regions that have uniform interpretations.

As you resume moving the hole around the map, you will further discover a wide variety of interesting picture fragments which constrain their parts. A sharp kink in a chain, for example, rules out the possibility that it is part of a bridge. It could, on the other hand, be a mountain top, in which case the chain is a mountain and the regions on either side are both land, or it could be part of a coast line, in which case the region on one side is land, the other being sea or vice versa, or ... . If a chain stops abruptly with no Other lines anywhere in the vicinity it most certainly is not a shoreline; furthermore, the region that it stopped in must be a land region. The free end could be a river source in which case the chain is a river flowing away from the free end. (Rivers may appear out of the ground but they do not disappear into it. Rivers also start at lakes and other rivers. They empty into other rivers, lakes or the sea. They may, however, temporarily disappear under a bridge.) Or the free end could be a mountainside or ... .

These informative picture fragments are called "primary cues" because they invoke models that interpret the immediate locale of the cue thereby putting constraints on the lineal and areal components of the cue. The initial enormous ambiguity of interpretation is reduced by these local models. It is further reduced by allowing the models to talk to each other and agree upon the interpretations of picture elements that they mutually interpret. This process is handled by a network consistency algorithm that progressively eliminates interpretations of the picture primitives, the chains and regions (not the interpretations of the cues), until, if the model information is strong enough, the interpretation intended by the user remains.

A wide variety of geographical and cartographical knowledge, typified by the sample inferences given above, is captured in MAPSEE by the primary cue interpretation catalogue. The varieties of cue are shown in Figure 2, with names for their relevant component parts.

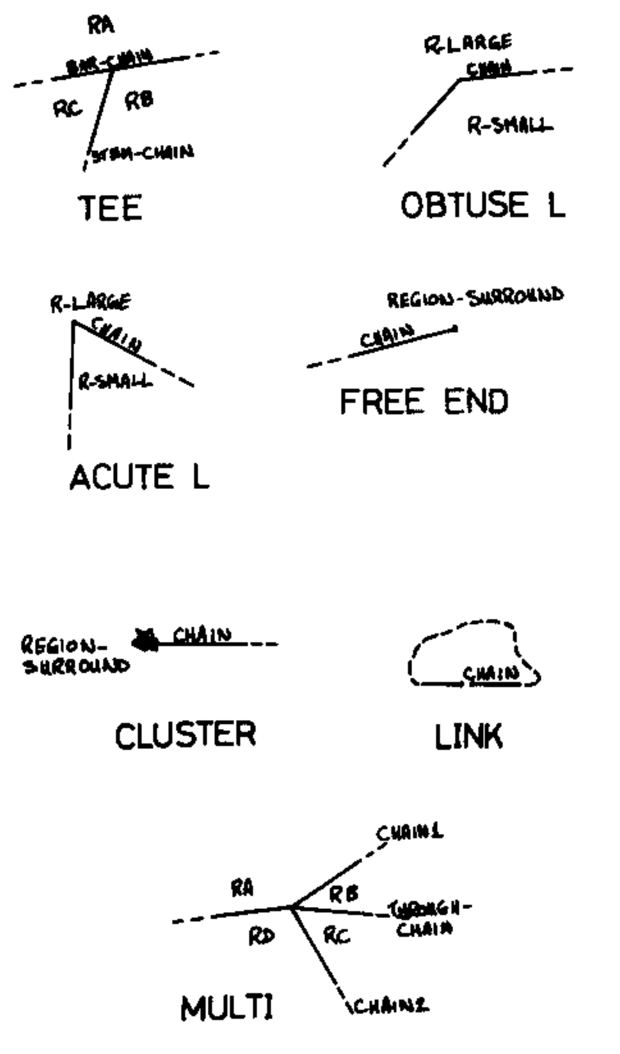


Figure 2. The Primary Cues Used by MAPSEE

Cue	Interpretations of Parts						
TEE	STEM-CHAIN	BAR-CHAIN	RA	RB	RC		
	{river}|{river*}	{shore}	{sea}	{land}	{land}		
	{river,river*}	{shore}	{lake}	{land}	{land}		
	{river,river*}	{river,river*}	{land}	{land}	{land}		
	{road}	{road}	{land}	{land}	{land}		
	{mountain}	{mountain}	{land}	{land}	{land}		
	{river,river*}	{bridge}	{land}	{land}	{land}		
OBTUSE L	CHAIN	R-LARGE	R-SMALL				
	{shore}	{lake,sea}	{land}				
	{shore}	{land}	{lake,sea}				
	{road,bridge,river,river*}	{land}	{land}				
ACUTE L	CHAIN	R-LARGE	R-SMALL				
	{shore}	{lake,sea}	{land}				
	{shore}	{land}	{lake,sea}				
	{road,mountain,river,river*}	{land}	{land}				
FREE END	CHAIN	REGION-SURROUND					
	{river} |{river*}	{land}					
	{mountain,bridge}	{land}					
CLUSTER	CHAIN	REGION-SURROUND					
	{road}	{land}					
LINK	CHAIN						
	{shore}						
MULTI	THROUGH-CHAIN	CHAIN1	CHAIN2	RA	RB	RC	RD
	{river,river*}	{river,river*}	{river,river*}	{land}	{land}	{land}	{land}
	{road}	{road}	{road}	{land}	{land}	{land}	{land}

Figure 3. The primary cue interpretation catalogue

For each cue there is a set of models as listed in Figure 3. Each model constrains the interpretation of each part of the cue to belong to the set given. The interpretations of Figure 3 are context-sensitive in that if the interpretations of a part are separated by a | then only one of them is possible. The direction of flow of a river is handled this way. A chain has associated with it the direction in which it was drawn. If , the river flows in that direction it is labelled "river" else "river*". In the first interpretation of the TEE, for example, the river can only flow into the TEE on the stem-chain

In order to use this catalogue of models we must segment the picture into chains, regions, cue instances and the bindings of their components. Unfortunately, that segmentation cannot be done perfectly, as we shall see, but it can be done with sufficient care that the models can start to make sense of the picture. That interpretation can then be used to refine the segmentation. The program MAPSEE, written in LISP, consists of the three phases: partial segmentation, network consistency, and refining the segmentation.

4. The Initial Partial Segmentation

4.1 Representations

MAPSEE receives a map in the form of a procedure for drawing it, created by the routines that track the stylus on the data tablet. That is, the input is a sequence of plotter commands where a command is move (pen up) to (x,y) or draw (pen down) to (x,y) from the current position.

There are so many points in this picture description (more than 800 for Figure 1) that one of the main priorities of all the segmentation routines is computational efficiency. There are two ways in which this is achieved. In the first place, a variety of different representations of the picture are maintained. Each is appropriate for one or more purposes. Secondly, when computing in a pictorial representation, a segmenter only works at a level of detail appropriate to its current needs.

The procedural representation gives way to a network representation which initially contains just chains (consecutive draws), line segments and segment end points. In this representation, each chain undergoes a process of generalization, as the cartographers call it, whereby at each level of detail the chain is represented to within a certain tolerance.

Finally, there is an array representation indexed by the x-y coordinates of the end points. This is quite coarse (32x32) but allows quick answers to questions such as "What are you near?" which uses a spiral search in the array. As discussed in the next section, the array representation is generalized in the process of region-finding to form a space occupation hierarchy of arrays of four elements each.

4•2 Region Segmentation

If we were to define a region as a connected subset of a 2D Euclidean space, the picture, in our domain, would always have exactly one region! Whenever the user intends to enclose a region he leaves a small (or, sometimes, not so small) gap, relying upon the map reader to divine his intention by reading his mind as well as the map. We cannot segment until we can interpret but we cannot interpret until we segment; this is the familiar AI chicken-and-egg problem. However, a initial, partial, conservative region segmentation is possible. A recursive algorithm partitions the image into empty patches: subdividing a patch of space only if it is not empty. This top-down subdivision stops well before it could lead to trouble, at a level whose patch size is much greater than any unintentional gaps in the sketch. The empty adjacent patches are then merged to form the five regions shown in Figure 4. The conservatism guarantees no leakage; no region so found will correspond to more than one 'intended' region. But some intended regions may be represented by more than one found region: the large connected land region has been split into regions 2, 3, 4 and 5. Other intended regions may not be represented at all: the two small land regions in the river delta have been missed. Moreover, the extent of the found regions is somewhat less than their actual extent. As we shall see, the consistency process is very tolerant of these necessary idiosyncracies of the region segmenter.

4.3 Cue Segmentation

Each of the cue types has its own specialized routines that discover

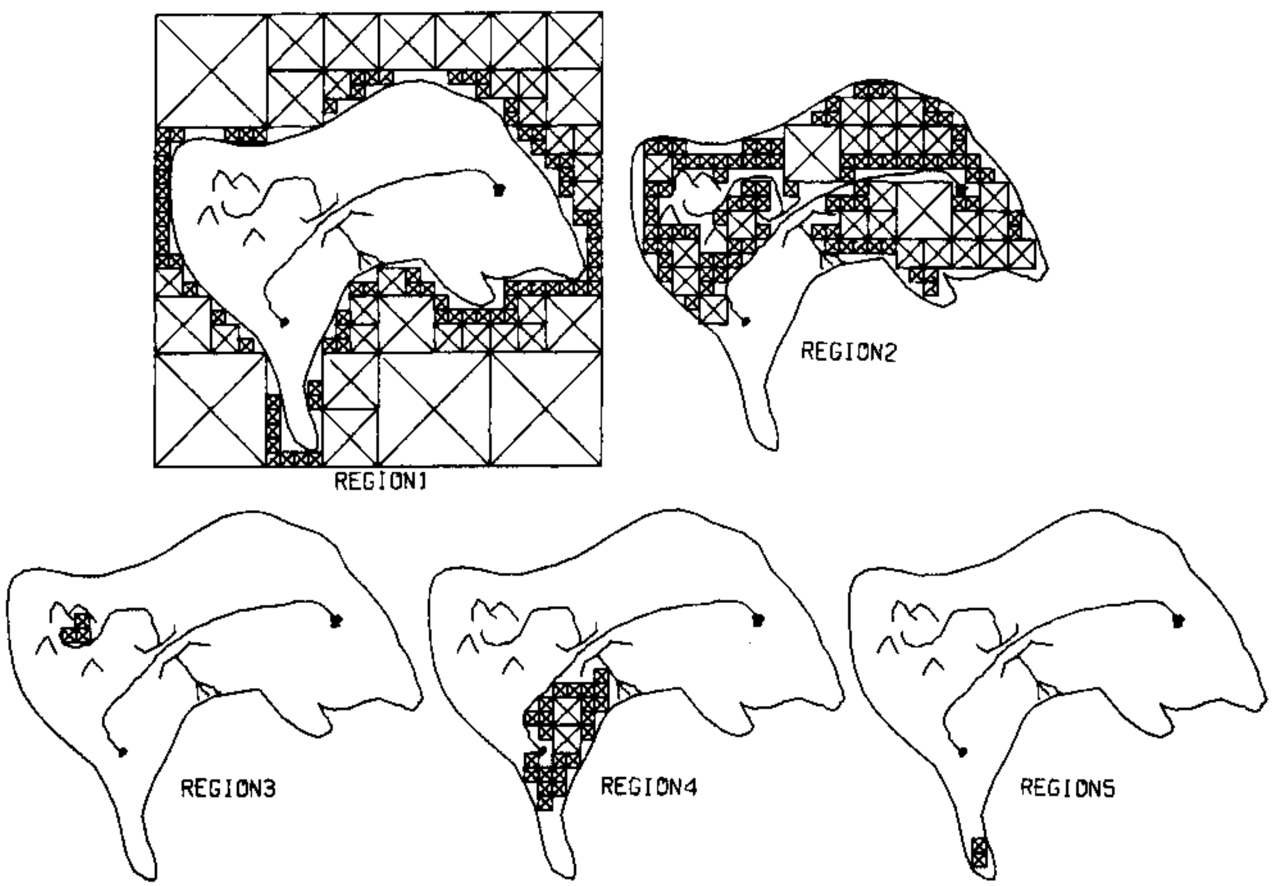


Figure 4. The Initial Region Segmentation

instances in the picture. They lean heavily on the levels of detail in the representations for efficiency. Moreover, they all have their own brand of conservatism. Each is designed to reject all border-line cue instances. As the Jolly Green Giant says, "Only the best will do!" A tentative free end, for example, must be well in the clear (relative to the minimum patch size of the region segmentation) before it is accepted as a free end. An obtuse angle must have arms longer than a given minimum, straighter than a certain tolerance, angle considerably less than pi ... . No false cues can be found so, as a result, many genuine ones are ignored. The cues found are indicated by the hexagons in Figure 5.

4.4 Fleshing Out the Cues

Each cue instance needs to bind various picture elements (chains and regions) to its internal names. Again, the segmentation process is heavily biased in favour of sins of omission rather than commission. If, for example, it is looking for the region associated in a certain direction with a cue, it crawls carefully in that direction from the initial point. If it finds a region within a very short distance, again, determined by the minimum patch size, well and good. But it it does not it will give up rather than risk returning the wrong region. It it gives up it creates a region ghost (Bobrow and Winograd, 1977)

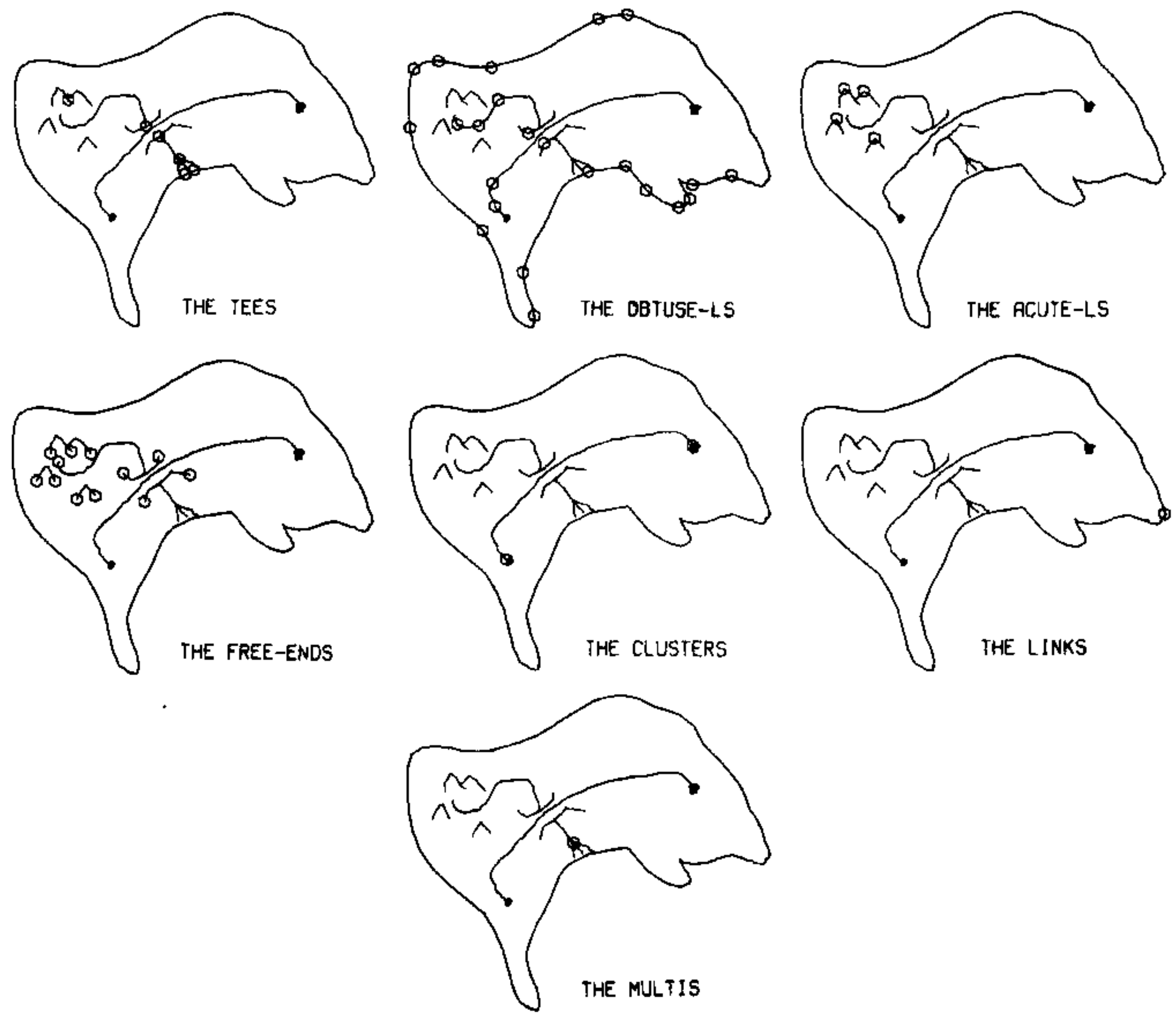


Figure 5. The Cue Instances Discovered

that stands for the region which has that relationship to the cue but cannot yet be identified. The region corresponding to the ghost may or may not exist as a found region. Eighteen region ghosts were created during the segmentation of the sample map.

5. The Consistency Phase

The picture is now partially segmented into chains, regions and partially instantiated cues. In describing the consistency process, I will ignore, for the time being, the four types of inadequacies in the segmentation (the extra regions, the missing regions, the missing cues and the region ghosts) and assume that the segmentation is perfect. Subsequently, we shall see how those inadequacies affect the consistency process.

Mackworth (1975a) discusses and extends a class of algorithms typified by Waltz's (1972) arc consistency algorithm (called AC-2, there) and Montanari's (1974) path consistency algorithm (called PC-1), designed to satisfy a set of binary relations among a set of variables each of which must be instantiated in an associated domain. Network consistency algorithms are often better than backtracking for such a task in that, by appropriate bookkeeping, they eliminate several kinds of thrashing behaviour.

In Waltz's blocks world, for example, the variables correspond to the junctions, the domains to the set of possible corners for each junction type and the binary relations to the edges, in that each edge must have the same interpretation imposed on it by each of its two corners. His network of relations was then isomorphic to the perfect line drawing being interpreted.

In MAPSEE, the "variables" are the chains and the regions (which also must be interpreted: everything need not, indeed cannot, be packed into the chain interpretations). The domains are their context-free interpretations, that is {road, river, river*, mountain, bridge, shore) for chains and {land, lake,sea) for regions. The relations are the cue instances, the constraint being the disjunction of the set of models for each cue instance.

The relations are now n-ary, not just binary, because each model relates from one to seven regions and chains. The network consistency algorithm used in MAPSEE given below is a suitably generalized version of AC-3 (Mackworth, 1975a). Note that, in lieu of network consistency, one could, of course, backtrack on the values in the domains of the chains and regions, failing back when any cue ceases to have a model which satisfied the current values; however, the following algorithm, NC, is far more efficient.

NC: An n-ary Relation Consistency Algorithm

1. Construct a queue consisting of (variable,relation) pairs in which each variable is paired with every relation that directly constrains it.
2. While the queue is not empty do steps 2.1 and 2.2.
   2.1 Remove the first pair (x,R) from the queue.
   For each value, a, in the domain of variable x, Dx, do step 2.1.1
      2.1.1 Find at least one value in the domain of each of the other variables directly constrained by relation R such that all the values, including a, simultaneously satisfy R. If such values cannot be found delete a from Dx.
   2.2 If any values were deleted from Dx in step 2.1 then do step 2.2.1
      2.2.1 Tf Dx is now empty then return failure as the result of this call else replace the queue by the union of the queue and the set of pairs obtained from all the relations other than R that constrain x, each relation paired with all the variables other than x that it constrains.
3. At this step there are three possible states of the network:
   a) If every variable has exactly one element in its domain return that set of bindings as the result of this call.
   b) If one variable, y, has k ($k > 1$) elements in its domain and the rest have exactly one element return the k solutions formed by binding y to each of its values and the other variables to their unique values.
   c) If more than one variable has more than one element in its domain then split the domain of one of those variables approximately in half and return the solutions obtained by applying the algorithm recursively to the two subproblems so generated.

The algorithm either returns failure (because some domain was exhausted) or one or more solutions each of which satisfies all the relations. The solutions are complete: no subsequent backtracking is necessary. The algorithm can be trivially modified to return just the first solution if desired. Note that the ordering of the queue is unspecified: the process converges regardless; however, it may be treated as a priority queue. For example, sorting the queue so that strongly interrelated variables are more likely to be adjacent in the queue speeds convergence.

Freuder (1976) independently generalized the consistency arguments given, for binary relations, in (Mackworth, 1975a) to apply to n-ary relations. His algorithm is very different from the one presented here in that he explicitly constructs sets of all the n-tuples of values of the variables which satisfy each relation and deletes tuples from those sets. Furthermore, he constructs similar exhaustive representations for all the implicit relations induced by the ones given up to and including the global relation that relates all the variables. As with the binary relation consistency algorithms complexity analysis of these algorithms is difficult (for anything other than worst case) making explicit comparison impossible. Rest assured, though, that they are both inherently exponential, in the worst case, in that the problem is NP-complete. For this task, however, NC requires far fewer CONS cells and operations than Freuder's algorithm. Significant contributions to the development of network consistency algorithms have also been made by Gaschnig (1974), Barrow and Tenenbaum (1976) and Rosenfeld, Hummel and Zucker (1976).

In the implementation of NC in MAPSEE each cue has a list of models associated with it. Each instance of that cue has a set of bindings for its subparts to various chains and regions (the "variables'' it constrains). In step 2.1.1 of the algorithm, a structure matcher is used to match the cue instance against each model for the cue until a model is found all of whose parts match successfully. A part of a cue instance and the corresponding part of a model match iff their domains have a non-NIL intersection unless the instance part is the particular variable x in which case the model part must have interpretation a in its domain.

For the sample map the consistency algorithm, NC, converged to unique values for all but one region in a single pass. The algorithm did not invoke itself recursively. The chain interpretations are as shown in Figure 6. The only remaining ambiguity is in the interpretation of the surrounding region, regionl, as either sea or lake. The user may have intended "sea" but the island could, of course, be in a large lake whose shore is beyond the bounds of the map. Regions 2, 3, 4 and 5 are all interpreted as land. The interpretations are, presumably, as intended by the user.

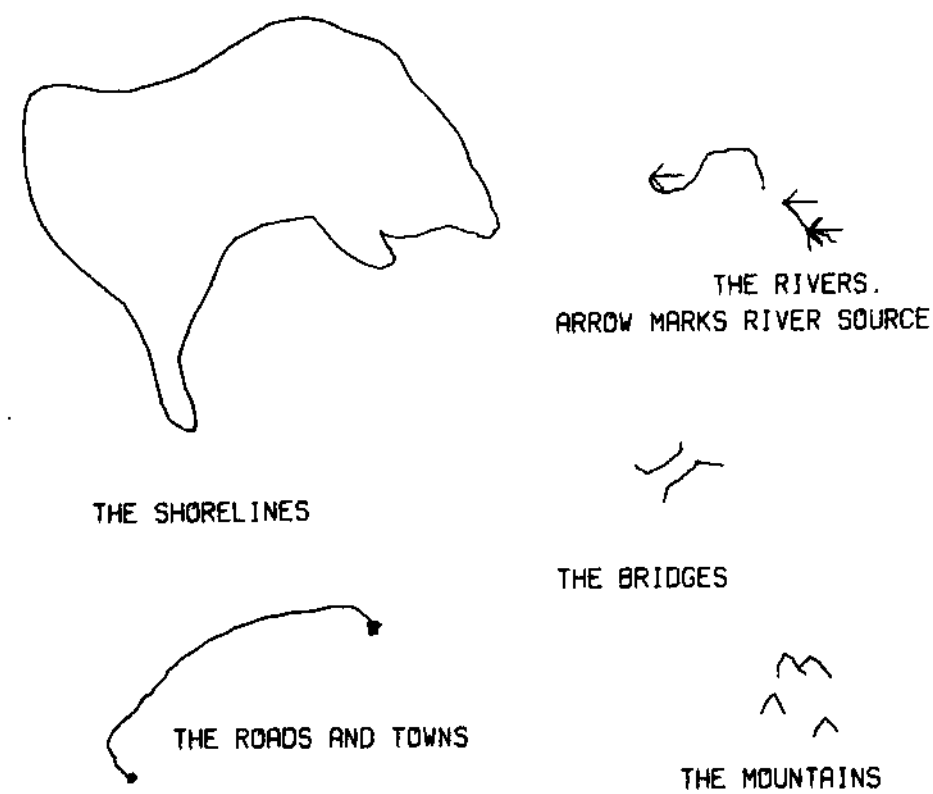


Figure 6. The Chain Interpretations

6. Refining the Initial Segmentation

In this section we will consider the effect of the segmentation deficiencies on the consistency process and then see how the results of that interpretation process can be used to refine the segmentation. Recall that the deficiencies are: the missing cues, the region ghosts, the missing regions and the extra regions.

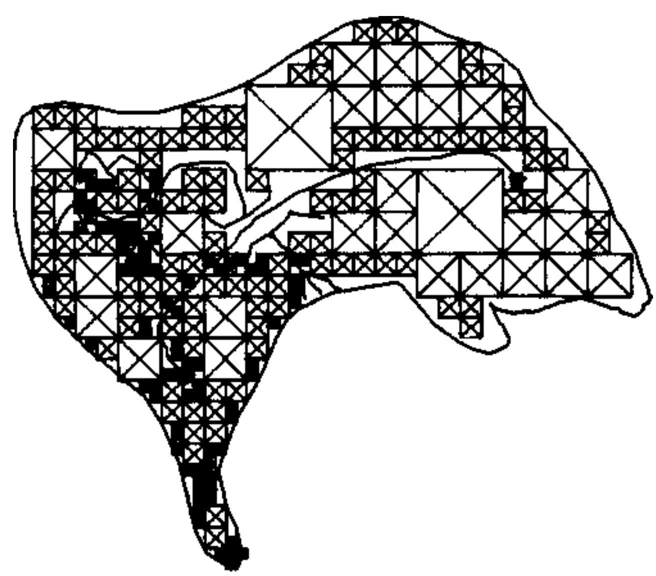
The missing cues have no serious effect on the consistency process, provided, of course, that sufficient remain. A missing cue simply fails to supply its extra constraints on the possible interpretations of the chains and regions. In this domain, however, there is such a welter of cues invoking consistent models that there is a multitude of partially independent but mutually confirming inference paths. Breaking a few of those inference paths causes no degradation in the interpretation. It is tempting to postulate that most perceptual tasks, in the real world, have the rich semantic? which give rise to this robustness property if we can but discover the appropriate language for the inferences and appropriate mechanisms for carrying them out. (The qualification "in the real world" is added because psychological experiments in the laboratory usually use meaning-deprived stimuli that rule out this phenomenon (Clowes, 1972).)

The region ghosts are, if you like, region intensions while the found regions are (imperfect) region extensions (Woods, 1975). A ghost is an intension in that it may be specified as, for example, "the region on the reflex angle side of this acute L". The intension/extension distinction forms a spectrum rather than a strict dichotomy here. Recall that a ghost arises when a cue fails to find an associated region; it may fail either because it stopped looking too soon even though there is a found region there or because there is no found region. The ghosts participate in the consistency process just as do the found regions. The single cue that created a region ghost constrains it and it is quite possible for interpretations of the ghost to be progressively ruled out. After the consistency process we still do not know the extension of a ghost but we may know more about it than before; for example, it may now be forced to have the interpretation "land".

The missing regions, as in the river delta, for example, also do not seriously affect the consistency process. The cues in the neighbourhood of a missing region will have used ghosts in its stead. But, standing in for a single missing region there will be several ghosts so the constraining effect will be weakened somewhat.

Similarly, the extra regions created by the splitting of a single intended region participate independently in the consistency process thereby exerting a weaker constraining effect than if the region had not been split. However, the semantic richness overcomes that weakening and forces the four found regions corresponding to the single intended land region (regions 2, 3, 4 and 5) to have that single interpretation. Again, as in the other cases, if the region splitting is so severe as to cut too many inference paths then the process will degrade gracefully (Marr, 1975). In that case the various found regions would not have the intended interpretation uniquely. It would simply be in the intersection of the possible interpretations of the found regions.

The third phase of MAPSEE uses the results of the consistency process to refine the initial partial segmentation. There are four ways in which this can be done: a) establishing distinctghosts with the same interpretation and location as co-extensive b) considering the merge of found regions with the same interpretation c) establishing a found region as the extension of a ghost with the same interpretation and d) discovering a new found region as the extension of one or more ghosts. These involve revisiting the picture and segmenting more purposefully, more carefully and at a finer level of detail in the particular areas concerned. Figure 7 shows the final land region that results from the successful proposed merges of the separate initial land regions.



REFINED REGION2 IS LAND

Figure 7. The Final Land Region

7. Conclusions

I cannot here discuss how this work satisfies the goals of the project nor future directions such as a) integrating still further the segmentation and interpretation phases, b) automating the generation of the primary cue interpretation catalogue by the provision of a language for describing the models so that transfer to other sketch worlds is facilitated and c) the use of schemata as procedural models. Suffice it to say that MAPSEE is an existence proof of the power of semantics in the interpretation of pictures. It demonstrates that the cue/descriptive model paradigm works in domains other than the blocks world, that the network consistency algorithms can be extended, that imperfect data can be overcome by a thoroughgoing conservatism in the segmentation process, that a partial segmentation can yield an initial interpretation, and that the interpretation can sensibly refine the initial segmentation.

8. Acknowledgements

This work is supported by the National Research Council of Canada's Operating Grant A9281 and a grant from the University of British Columbia.

9. Bibliography

Barrow, H.G. and Tenenbaum, J. M. (1976) MSYS: a System for Reasoning about Scenes. Tech. Note 121, A.I. Center, Stanford Res. Inst., Menlo Park, Calif

Bobrow, D. G. and Winograd, T. (1977) An Overview of KRL, a Knowledge Representation Language. Journal of Cognitive Science 1,1, 3-46

Clowes, M. B. (1972) Artificial Intelligence as Psychology. AISB Builetin 1, November

Freuder, E. C. (1976) Synthesizing Constraint Expressions. A. I. Memo 370, M.I.T., Cambridge, Mass.

Gaschnig, J. (1974) A Constraint Satisfaction Method for Inference Making. Proc. 12th Ann. Allerton Conf. on Circuit Theory, U. 111., Urbana-Champaign, 111., pp. 866-874

Mackworth, A. K. (1973) Interpreting Pictures as Polyhedral Scenes. Artificial Intelligence 4, 2, 121-137 also ProcT 31JCAI, Stanford, Calif., pp. 556-563

Mackworth, A. K. (1975a) Consistency in Networks of Relations. TR 75-3, Dept. of Comp. Sci., Univ. of B.C., Vancouver, and Artificial Inteligence 8 (1977), 1, 99-118

Mackworth, A. K. (1975b) How to See A Simple World. in Machine Intelligence 8, E. W. Elcock and D. Michie (eds.) (in press) and TR 75-4, Dept. of Comp. Sci., U. of B.C., Vancouver.

Mackworth, A. K. (1976) Model-driven Interpretation in Intelligent Vision Systems. Perception 5, 349-370

Mackworth, A. K. (1977) Vision Research Strategy: Black Magic, Metaphors, Mechanisms, Miniworlds and Maps. Proc Workshop on Computer Vision Systems, Amherst, Mass. (in press)

Marr, D. (1975) Early Processing of Visual Information. A. I. Memo 340, MIT, Cambridge, Mass.

Montanari, U. (1974) Networks of Constraints: Fundamental Properties and Applications to Picture Processing Information Sciences 7, 95-132

Starr, D. W. and Mackworth, A. K. (1976) Interpretation-Directed Segmentation of ERTS Images. Proc. ACM/CIPS Pacific Regional Symp., pp 69-75

Rosenfeld, A., Hummel, R. A. and Zucker, S. W. (1976) Scene Labelling by Relaxation Operations. IEEE Trans. on Systems, Man and Cybernetics, SMC-6, 420-433

Tenenbaum, M. and Barrow, H. G. (1976) IGS: a Paradigm for Integrating Image Segmentation and Interpretation. in Pattern Recognition and Artificial Intel 1igence. Academic Press

Waltz, D. L. (1972) Generating Semantic Descriptions from Drawings of Scenes with Shadows. MAC AI-TR-271, M.I.T., Cambridge, Mass.

Woods, W. A. (1975) What's in a Link, in Representation and Understanding, D. G. Bobrow and A. Collins (eds), Academic Press, pp. 35-82

Yakimovsky, Y. and Feldman, J. (1973) A Semantics-Based Decis ion-Theoretic Region Analyzer. Proc. 3IJCAI. Stanford, Calif., pp. 580-588