# Integrating Top-Down Control with Intermediate-Level Vision: A Case Study

Bruce A. Draper
J. Ross Beveridge
Edward M. Riseman

**COINS TR 89-45**

May 1989

# Integrating Top-Down Control with Intermediate-Level Vision: A Case Study

Bruce A. Draper  J. Ross Beveridge  Edward M. Riseman

April 26, 1989

### Abstract

[1] The AI approach to vision has been heralded as reducing the computational burden of traditional bottom-up systems by applying knowledge-based control. AI-style systems use knowledge to focus attention and processing resources on the most promising hypotheses and combine information from multiple knowledge sources and/or sensors. Our case study compares the complexity of an intermediate-level grouping task with and without top-down control. The results provide clear empirical support for the claim that knowledge-directed control reduces the computation required for object identification.

The Rectilinear Line Grouping System (RLGS) is a bottom-up line grouping system designed to extract man-made structures from static images. The Schema System is a knowledge-based system shell for controlling computer vision tasks. In this paper we consider the task of finding instances of two objects (telephone poles and road signs) in complex natural scenes. First we apply the RLGS in the original bottom-up manner in which it was designed, noting the number of line relations that must be computed, as well as the complexity of the graph matching task that must be performed. Then we place the RLGS primitives under the direction of the Schema System, noting the reduction in required computation.

## 1 Introduction

The strictly bottom-up approach to the interpretation of complex images is infeasible as either a biological or a computational model. Tsotsos applied biological and physiological constraints to show that massive parallelism alone can not explain human visual performance; other mechanisms, including knowledge-based prediction, are required ([6]). The inadequacy of bottom-up interpretation as a computational model has been documented in

[1] This research was supported by the National Science Foundation under grant NSF DCR-8500332 and by the Defense Advanced Research Projects Agency under grants ETL DACA76-85-C-0008 and RADC F30602-87-C-0140.

two ways: first, by proving that tasks involved in some approaches to the image interpretation process are NP-complete, such as subgraph isomorphism and constraint propagation ([4]); second, by the observations of those who have built large scene interpretation systems (e.g. [3]).

The AI approach to vision has been heralded as reducing the computational burden of traditional bottom-up systems by applying sophisticated, knowledge-based control. Object predictions based on domain knowledge, interpretation results of previous images in the sequence, and the evolving interpretation of the current image (among other sources) can be used to constrain the search for an object to restricted portions of the image. Common views of 3D objects can provide efficient 2D predictions, and automatic analysis of the knowledge base can determine the most efficient sequence of tasks to identify an object.

This paper reports a case study in using top-down knowledge and control to constrain bottom-up processing. The task is to find instances of two objects (telephone poles and warning signs) in two complex natural scenes. A measure of the effectiveness of knowledge-directed control is made by comparing the computation required to recognize the objects with and without knowledge-based control. In both cases, the underlying recognition tools are the same – the rectilinear line grouping system (RLGS [5]), which groups lines according to fundamental geometric relations, and a subgraph isomorphism routine ([7]) which finds instances of the objects' 2D models in the line groups. The difference is that in the second case, a top-down control system called the schema system ([2]) uses knowledge of expected region characteristics (color, texture and shape) to constrain the line grouping process.

The reader should understand that the point of this paper is not criticism of the RLGS or systems like it. Quite the opposite, this effort is the result of the success of the RLGS at locating certain objects, not only in aerial photography (for which is was designed), but in the less structured domain of New England road scenes. The line relations and connected components algorithm of the RLGS were obviously useful tools for the image interpretation task. The RLGS's control component was inadequate, but that was not the focus of the research. This paper grew from an effort to integrate the RLGS primitives into the intelligent control framework of the schema system.

# 2 The Bottom-Up System

## 2.1 The Rectilinear Line Grouping System

The RLGS was first presented in [5]. It is a bottom-up perceptual organization system that suggests the presence of man-made events by detecting sets of related line segments. Three distinct binary relations are used to form these sets. The relations are: spatially proximate collinear, spatially proximate parallel and spatially proximate orthogonal. The collinear relation indicates that one line is a good collinear continuation of the other. The

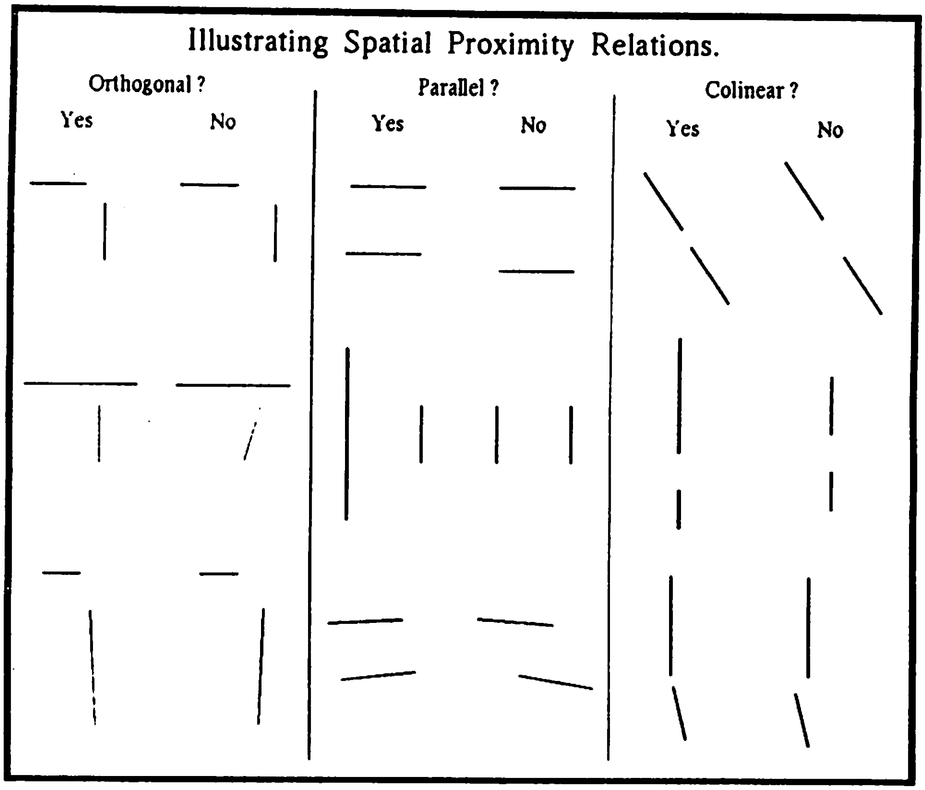

Figure 1: Illustrating when pairs of line segments satisfy the individual spatial proximity relations. Note subtle changes in distances between lines (scaled as a function of line length) as well as relative orientations.

parallel relation indicates that the two lines bound a rectangular area. The orthogonal relation indicates that the two lines form a corner or 'T'. Examples illustrating each type of relation are shown in Figure 1. Formal definitions are provided in [5].

The bottom-up line groups produced by the RLGS are best conceptualized as connected components in a graph. Line segments correspond to the nodes of the graph, while pairs of nodes satisfying one of the three proximity relations are linked. Because the binary relations are not transitive, connected components are confined to sub-graphs. The sub-graphs insure that the orientation of the lines within a single group are consistent with rectilinear structure. For example, a sequence of collinear pairs forming a semi-circle will not be part of a single sub-graph and therefore cannot form a single group.

In practice, grouping is restricted to eight sub-graphs. Each sub-graph contains line segments with orientation within $\pm\theta$ of either of two orthongal axes. In order to subdivide the range $0 - \pi/2$ into eight sub-ranges overlapping by 50%, a value of $(\pi/2)/8$ radians (11.25 degrees) is selected for $\theta$. These sub-graphs, called ***orthogonality ranges***, are designated by number. Orthogonality range $i$ contains all lines whose orientation is within $\theta$ of $(i * \theta)$ or $((i * \theta) + \pi/2)$. Hence, orthogonality range 0 contains roughly horizontal and vertical lines. Orthogonality range 4 contains roughly diagonal lines.

The overlapping of orthogonality ranges insures that interesting groups, if they exist, will be found by grouping within at least one range. However, it introduces a complication. If a large group is found within a given range, fragments from this group may be present in the two adjoining ranges. In [5] a voting scheme was presented for resolving this type of

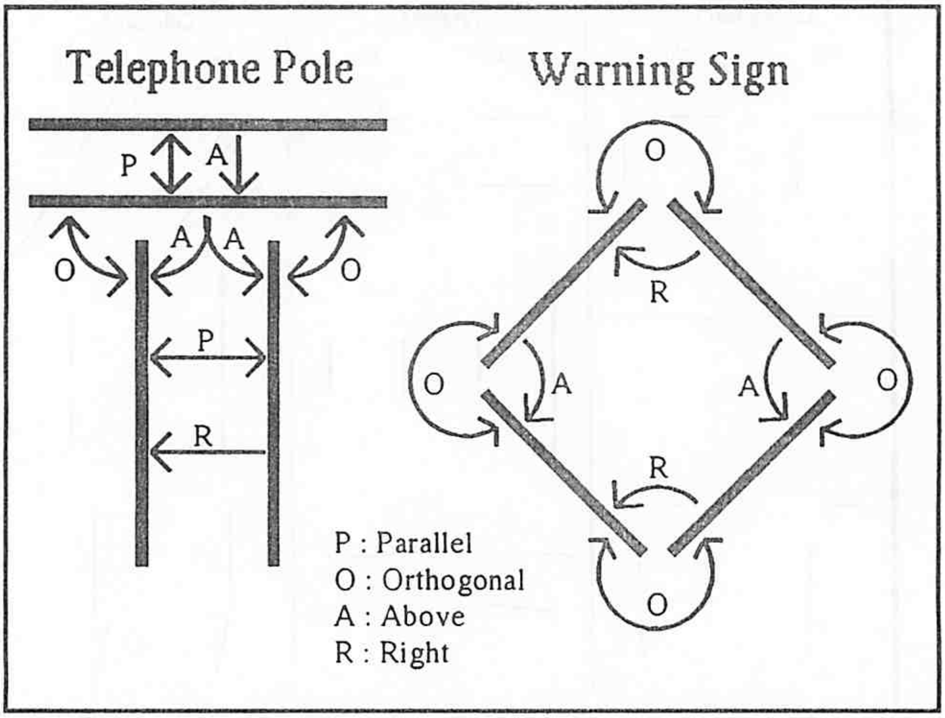

Figure 2: Illustrating relation model for telephone pole and warning sign. Note that orthogonal and parallel are symetric relations while above and right are not. Also note that the parallel relation appears only in the telephone pole model.

duplication. In most, but not all cases, the redundancy was adequately resolved by voting. However, as we shall show, top-down control removes entirely the need for overlapping orthogonality ranges.

## 2.2 The Graph Matcher

Once line groups have been extracted, objects are located in the image by finding correspondences between object models and pieces of line groups. Object models are expressed in terms of RLGS primitives. To be precise, an object model is a set of predicted line segments, grouped by pairwise collinear, parallel and orthogonal relations. Figure 2 shows the telephone pole and warning sign models used in this study. Objects are identified in the image by finding an isomorphism between an object model, called the *model graph*, and a RLGS line group, refered to as the *data graph*.

The subgraph isomorphism routine is based on the commonly-used algorithm of Ullman ([7]), with minor modifications for vision. Ullman's algorithm matches model lines to data lines using depth-first search. Every time a data line is assigned to a model line, all potential bindings for each remaining model line are tested, and any that are incompatible with the new assignment are removed from the model line's set of potential bindings. In AI terminology, binary constraint propagation is performed every time a data line is bound to a model line during the depth-first search.

Our implementation of Ullman's algorithm has been modified to allow directional links and on-demand (lazy) evaluation of node links. Directional links allow the expression of non-commutative relations such as "above" which we have found necessary. In our telephone pole model, for example, it is important that the crossbar be "above" (2D) the upright. On-demand computation removes the necessity of precomputing all possible line relations.

Although Ullman's algorithm is very good, it cannot escape the inherent intractability of the subgraph isomorphism problem. The algorithm's time complexity is exponential in the number of lines. As a result, the size of the graphs is the dominant factor in determining the over-all run-time of the system. Both of the models in this paper have four nodes in their model graphs. Therefore, we compare the complexity of the model matching task in the bottom-up and top-down scenarios by comparing the size of the respective data graphs.

# 3 The Schema System

The schema system is a knowledge-based image interpretation system for recognizing objects in natural scenes. It differs from other knowledge-based vision systems in that 1) it integrates many distinct methods of object interpretation; 2) it makes explicit control decisions; and 3) both its declarative knowledge base and its run-time processing are organized according to perceptual objects, rather than by level of abstraction. We will focus on the integration of multiple interpretation methods here; readers interested in the motivations for and consequences of the other points are encouraged to see [2].

Visual processing is highly object-specific. Much of the recent model-based vision research has limited itself to studying objects for which we have rigid models (e.g. CAD/CAM). Typically, an attempt is made to find a single algorithm which will recognize all such objects. Unfortunately, not everything can be identified through rigid object models. Some objects, such as telephone wires, are not rigid. Others, for example trees, are nearly rigid, but are irregular in that each instance has a unique shape. making fixed models infeasible. Still others (e.g. clouds, human beings) are neither rigid nor regular, while some (e.g. the sky) defy the very concept of a shape based representation. Many object recognition techniques are necessary for recognizing different objects. Even when a single interpretation mechanism is applicable to multiple objects, object-specific knowledge of contexts, common viewpoints, etc., can be used to speed the recognition process.

The point is not that rigid models should be abandoned - far from it, they provide the basis for recognition techniques that are useful for a wide range of objects and contexts. The point is that no single recognition algorithm will suffice for all objects under all circumstances. To be general purpose, a high-level vision system must provide a multitude of object-specific interpretation strategies.

The schema system views object recognition as a problem of search through the space of visual operators. Each operator seeks to extract evidence supporting the presence of an

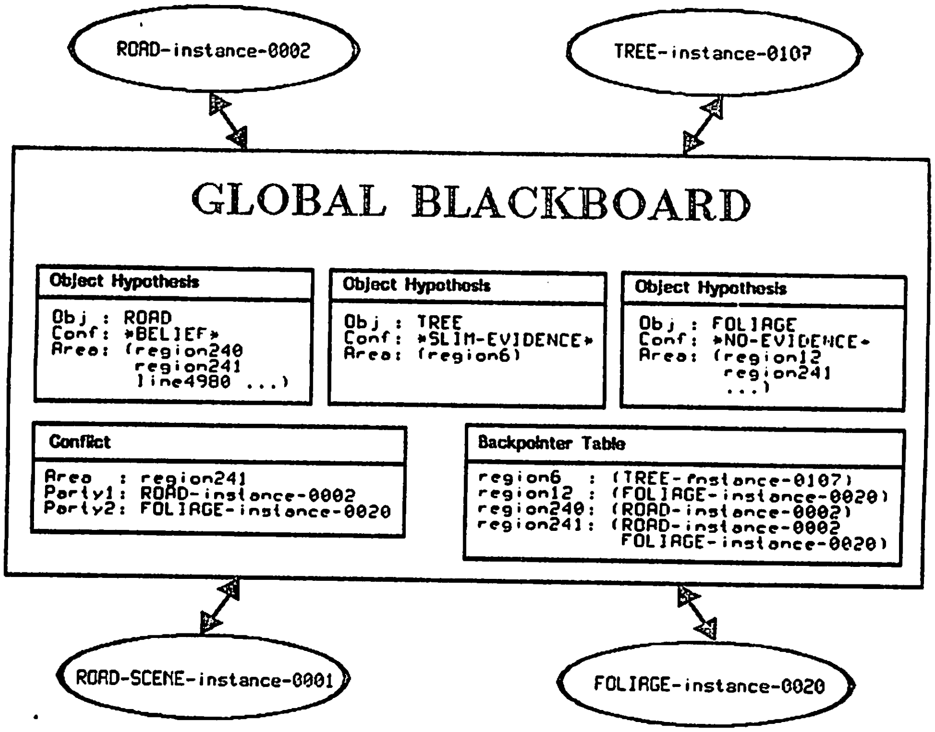

Figure 3: Illustrating the Schema System Blackboard.

object class instance. Examples of visual operators include color- and texture-based pattern classifiers, predictors of 2D views from 3D models and the subgraph isomorphism routine mentioned earlier. For each object class in the domain, a specialized schema is built which will guide the search for instances of that object class. Finding the appropriate sequence of operations to efficiently locate an object instance in any given context is the task of a schema. The schema system consists of many object schemas, each of which operate concurrently when interpreting an image. Inter-object relations are computed from object hypotheses written to the global blackboard by the various schemas. Figure 3 illustrates the state of the global blackboard during an interpretation.

The knowledge engineer must provide a schema for each object class in the domain. The schema declaration consists of 1) the list of relevant "knowledge sources" and 2) a function for combining symbolic evidence. The schema system then automatically compiles the optimal strategy for the object by exhaustive search of the (object-specific) space of knowledge source sequences. For the schemas in our case study, the relevent knowledge sources are parallel and orthogonal line grouping, graph matching, region color and texture classification, and region shape analysis. In order to be comparable with the purely bottom-up interpreation system, we used a very simple evidence combination function: any hypothesis which matches the model line graph is accepted as true, while any hypothesis that does not is rejected. In more complicated interpretation experiments complex evidence combination functions are used to allow for multiple views and scales, missing data, etc.

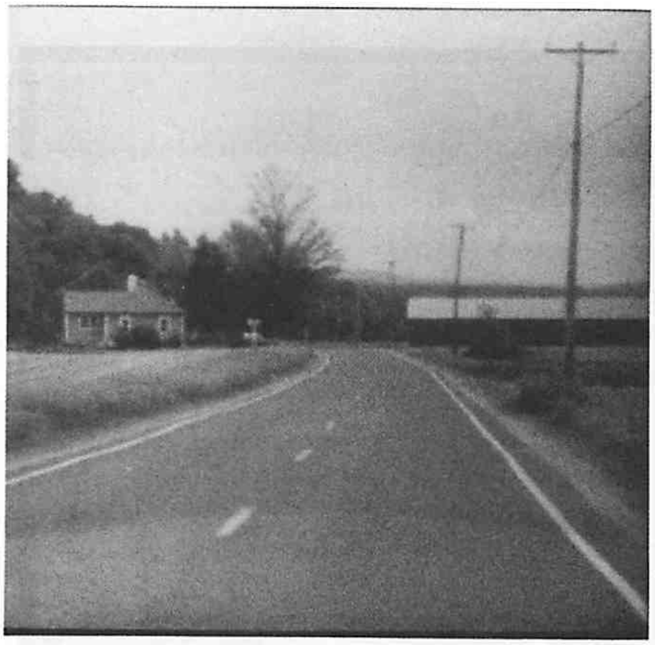

Figure 4: Road scene image 1.

# 4 Comparing Bottom-Up and Knowledge-Directed Execution

We applied the bottom-up and knowledge-directed verions of our system to two images, one of which contained a telephone pole, the other a warning sign. Figures 4 and 5 are black-and-white copies of the original color images. Both the bottom-up and knowledge-directed versions of the experiments are started with a set of straight lines extracted from an image using the algorithm in [8] (Figures 6 and 7). The lines were filtered to remove short and low contrast lines. Because the goal is to extract instances of telephone poles and warning signs from the image, relations which do not appear in either of these models need not be computed; in particular, the RLGS's spatially proximate collinear relation is not used[2]. Also, because the direction of gravity is known for this domain and all of the lines in the models are either horizontal, vertical or diagonal, lines at other orientations can be discarded. For the bottom-up RLGS, which divides the lines into 8 sets according to orientation ([5]), this corresponds to only grouping lines in 2 of the 8 sets (see section 2.1).

## 4.1 Bottom-Up Interpretation

The RLGS extracts all possible instances of the spatially proximate parallel and orthogonal line relations (Figures 8 and 9 show the line-pairs found for image 1). A connected

[2] Note that a subgraph isomorphism is a one-to-one mapping between data and model lines. Therefore, the collinear relation cannot be used to map multiple line segments to a single model line.

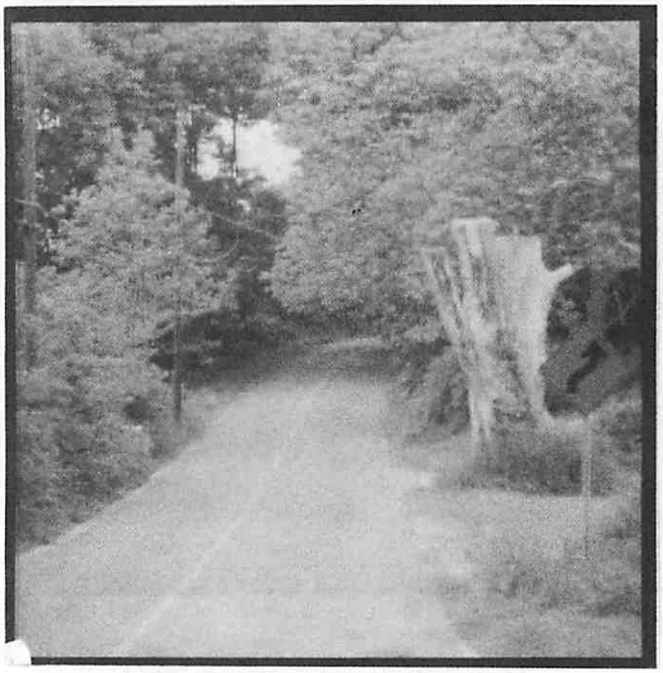

Figure 5: Road scene image 2.

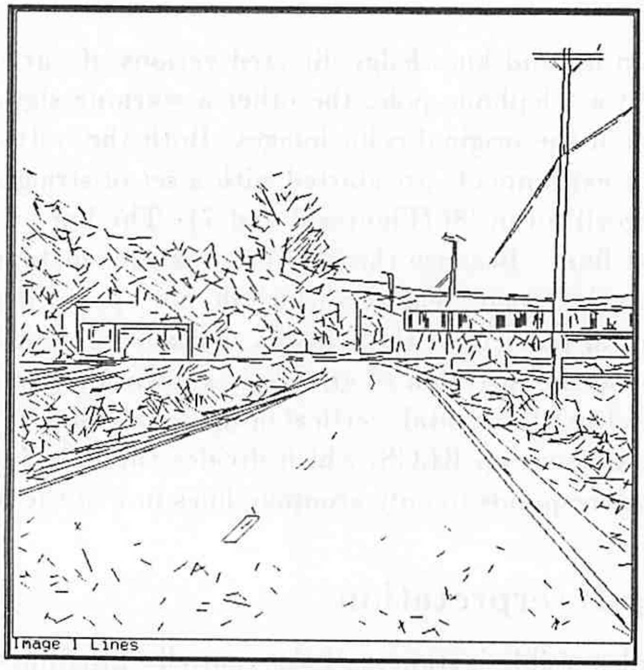

Figure 6: Straight line extracted from image 1 using [8]

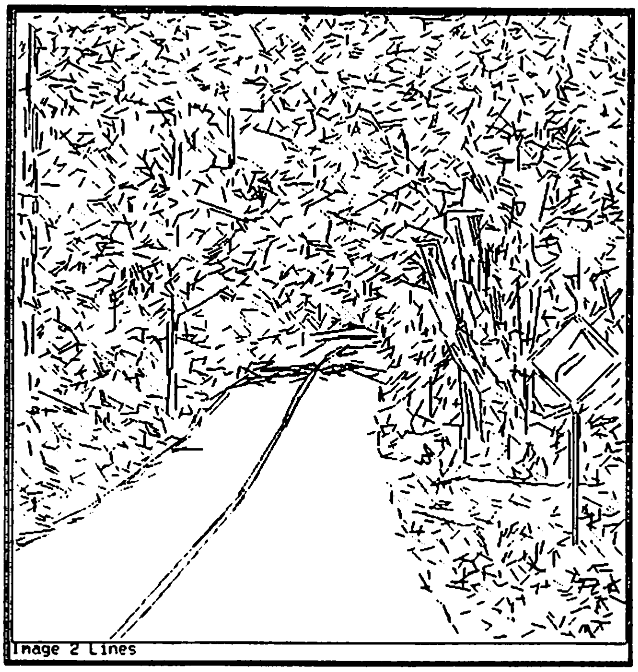

Figure 7: Straight line extracted from image 2 using [8]

components algorithm is then applied to find groups of related lines. Two types of line groups are created, the first by computing connected components over just the orthogonal relation, the second by computing connected components over both the orthogonal and parallel relations. The graph matcher can then search all diagonal orthogonal groups for instances of the warning sign model, and all horizontal/vertical orthogonal and parallel groups for instances of the telephone pole model.

## 4.2 Knowledge-Directed Interpretation

Figures 10 and 11 show the orthogonal and parallel relations computed under the control of the schema system for image 1. The schema system uses features derived from the region segmentation and knowledge of the expected color and texture of objects to control the computation of line pairs and groups. Because warning signs are saturated yellow, region-based recogntion strategies perform well. Warning signs rare.y fragment in the segmentation, and they can be distinguished from other objects based on the color features. For warning signs, the line data is used to confirm region-based hypotheses. The schema therefore selects the set of lines that intersect the hypothesis, and directs the RLGS to compute the orthogonal relation over this small set of lines. These lines are then passed as the data graph to the graph matcher, without ever calling the RLGS's connected components routine.

Telephone poles are recognized differently, since unlike warning signs they cannot be counted on to segment well. Tall, narrow objects tend to be fragmented by the region segmentation process. Moreover, although telephone poles have an expected color and

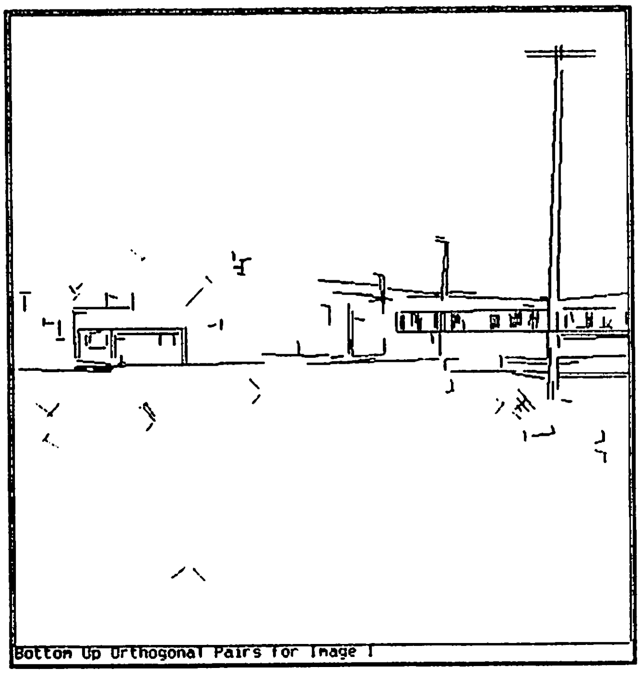

Figure 8: Spatially proximate orthogonal pairs found bottom-up for Image 1.

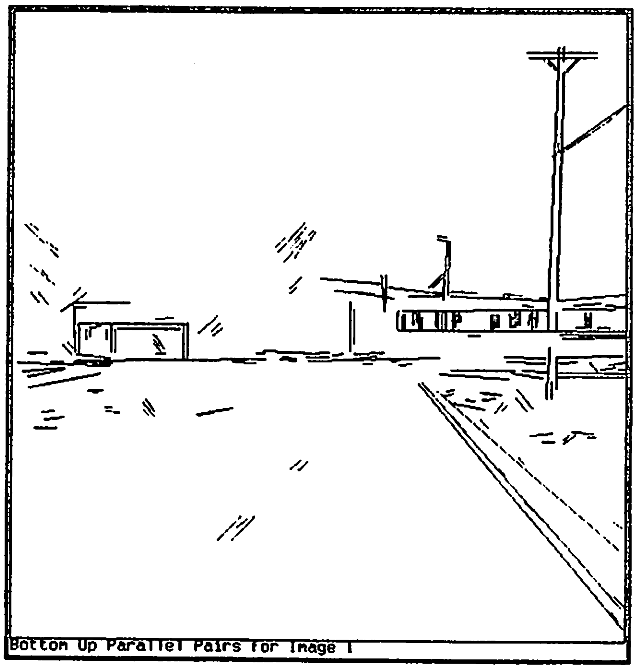

Figure 9: Spatially proximate parallel pairs found bottom-up for Image 1.

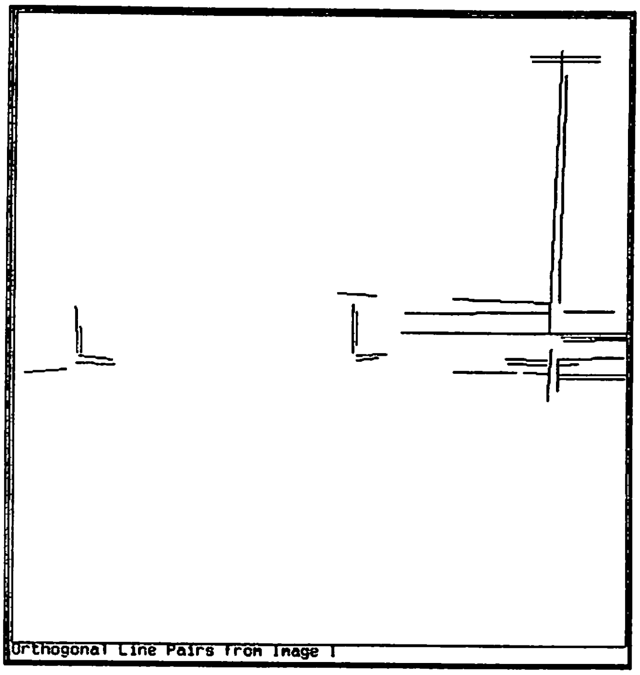

Figure 10: Spatially proximate orthogonal pairs found by schema system for Image 1.

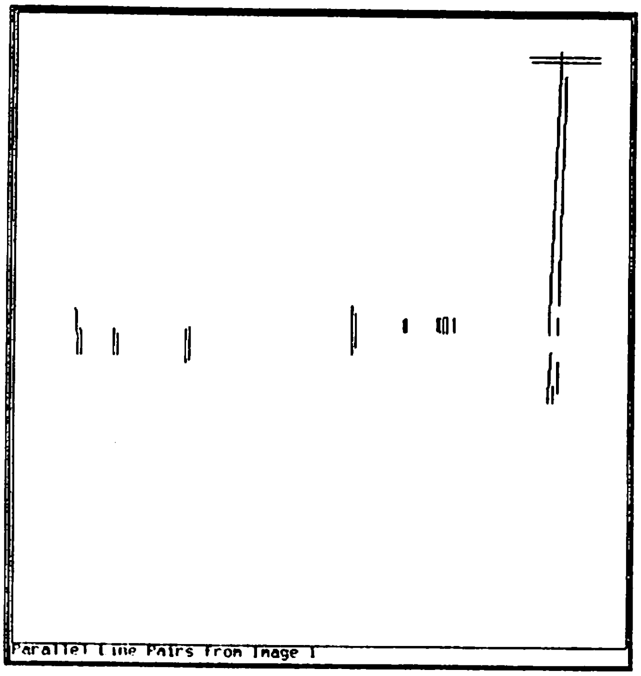

Figure 11: Spatially proximate parallel pairs found by schema system fcr Image 1.

|  | Image 1 |  | Image 2 |  |
|---|---|---|---|---|
|  | Parallel | Orthogonal | Parallel | Orthogonal |
| Bottom-up | 75 | 272 | 71 | 235 |
| Top-down | 15 | 50 | 29 | 27 |

Table 1: Relations computed with and without top-down knowledge

|  | Image 1 |  | Image 2 |  |
|---|---|---|---|---|
|  | O-groups | PO-groups | O-groups | PO-groups |
| Bottom-up | 150 | 224 | 193 | 318 |
| Top-down | 12 | 80 | 18 | 20 |

Table 2: # of lines involved in orthogonal and parallel/orthogonal groups

texture, it is not as distinctive as that of a warning sign. Thus the primary mechanism for recognizing telephone poles is the line data model-match; the region data merely suggests portions of the image to be considered. The search is applied everywhere there is either 1) a region that roughly matches the expected color and texture of telephone pole or 2) a highly elongated region. The schema collects all lines that are near such key regions, and directs the RLGS to find any lines that are collinear extensions of these lines (the lines, although more robust than the regions, may also be fragmented). Next, the schema computes the spatial window defined by the original and collinear lines, expands it slightly and collects all the horizontal and vertical lines within the window. These lines are then handed to the RLGS to be grouped according to the spatially proximate parallel and orthogonal relations. All line groups found by the RLGS with four or more lines are then tested with the graph matcher. Whenever a match is found, a new region is created from the interstitial pixels and (assuming its color is not totally unreasonable) hypothesized to be a telephone pole.

## 4.3 Results of the Case Study

Not surprisingly, the knowledge-directed version ran more efficiently. Table 1 shows the number of line pairs computed by the two systems; it shows that the schema system computed a factor of five fewer relations than its bottom-up counterpart. Table 2 shows the number and size of the line groups produced. Since the line groups become the data graphs for the exponentially expensive graph matching algorithm, the size of the line graphs is the dominant factor in determining the running time of the over-all system. Once again, the schema system proves more efficient: the line groups computed top-down are an order of magnitude smaller than the line groups produced bottom up.

For image 1, the pairs computed using the schema system cluster in areas where the schema found promising telephone pole regions; not surprisingly, it considered several

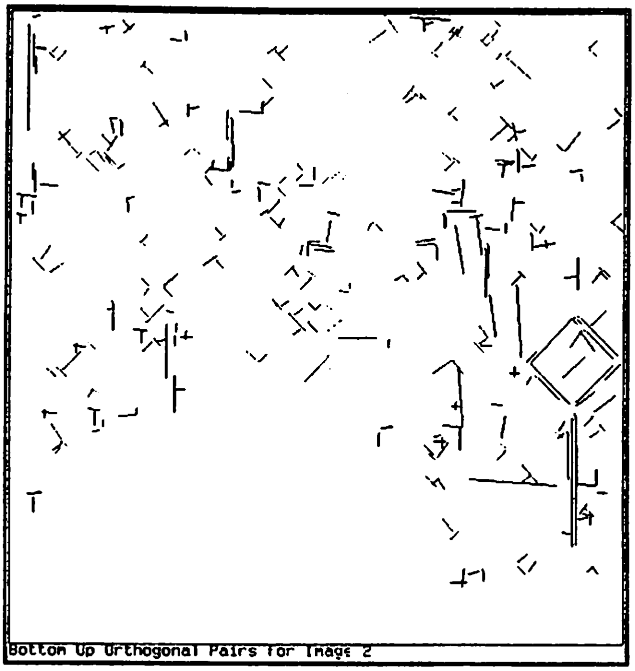

Figure 12: Spatially proximate orthogonal pairs found bottom-up for Image 2.

incorrect regions as well as the correct one. On the other hand, no relations were computed using diagonal lines in this image. The possibility of a warning sign being present was rejected on the basis of the region data alone.

Image 2 presents a different kind of problem. At first glance, one would expect fewer pair-wise relations to be present, because of the relative lack of man-made structure. Unfortunately, hundreds of instances of almost any pairwise relation can be found in the random chaos of the trees. The need to constrain the grouping process according to area (region) attributes is therefore just as important. Figures 12, 13, 14 and 15 show the orthognal and parallel pairs found by the two methods. Not only does the schema system do less work, it lessens the likelihood of finding false model matches in the trees.

There are disadvantages to the schema approach, however. Exhaustive search of the line data guarantees that all instances of the model in the line data will be found, whereas a failure in the segmentation system could cause the schema system to miss an instance (for example, if no part of the telephone pole was seperated from the background). In addition, although the amount of computation on the line data has been greatly reduced, the costs of region segmentation and region feature extraction must be added to the total cost of the interpretation. Nonethless, the computation saved through using knowledge may make an otherwise infeasible recognition task feasible.

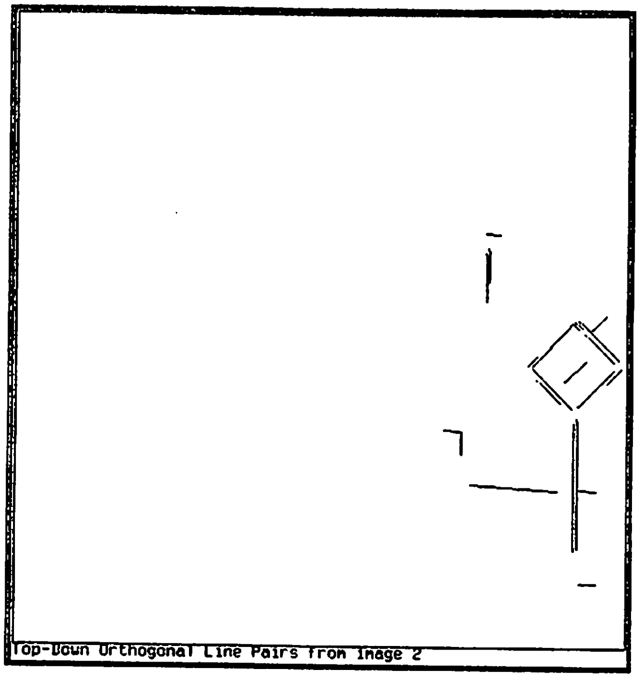

Figure 13: Spatially proximate orthogonal pairs found by schema system for Image 2.

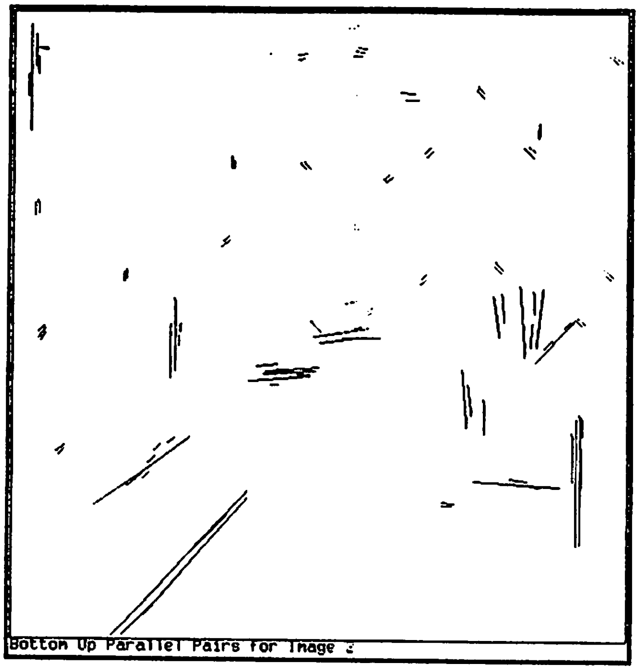

Figure 14: Spatially proximate parallel pairs found bottom-up for Image 2.

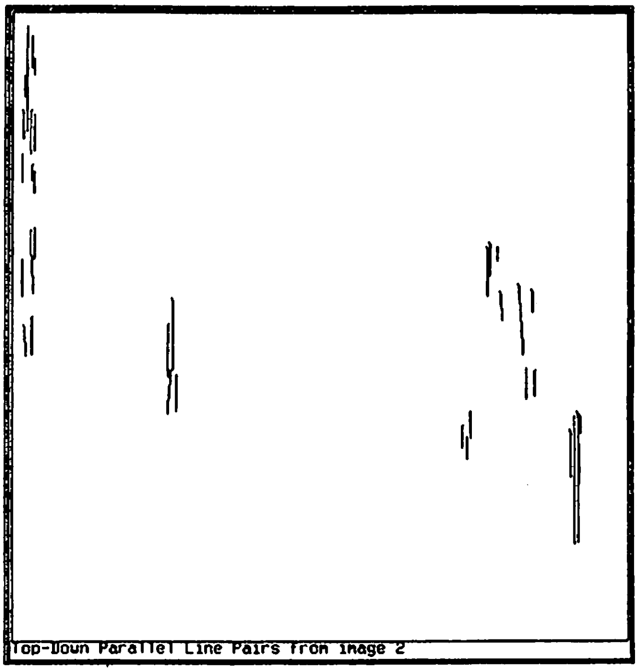

Figure 15: Spatially proximate parallel pairs found by schema system for Image 2.

# 5 Conclusion

Our case study gives clear empirical support for the claim that using knowledge saves computation. We compared the cost of object recognition with and without knowledge-directed control. The use of knowledge reduced the number of intermediate-level relations computed by a factor of five and the complexity of the resulting graph-matching problem by an order or magnitude. This is not meant to indict all bottom-up systems. Many low-level tasks, such as straight line extraction, are ideally suited for parallel, bottom-up processing. Moreover, bottom-up systems such as the RLGS often teach us what relations need to be extracted from an image. What is being assailed is the exclusive use of the bottom-up control strategy. It is hard to imagine circumstances in which knowledge of the interpretation goal could not in some manner constrain the application of visual operators, except at the lowest levels of the interpretation task. By adding the RLGS's relations to the schema system's library, the same image abstractions can be computed more efficiently. Thus the schema system is able to subsume the operators of the RLGS, both improving their efficiency and integrating them with other image interpretation techniques.

# References

[1] J. Ross Beveridge, Joey Griffith, Ralf R. Kohler, Allen R. Hanson and Edward M. Riseman, "Segmenting Images Using Localized Histograms and Region Merging" , to appear *International Journal of Computer Vision*

[2] Bruce A. Draper, Robert T. Collins, John Brolio, Allen R. Hanson and Edward M. Riseman. "The Schema System". To appear in *International Journal of Computer Vision.*

[3] Allen R. Hanson and Edward M. Riseman. "VISIONS: A Computer System for Interpreting Scenes" in *Computer Vision Systems*, Hanson and Riseman (eds.), Academic Press, New York. 1978.

[4] Lefteris M. Kirousis and Christos H. Papadimitriou, "The Complexity of Recognizing Polyhedral Scenes", *Proc. of 26th FOCS* (1985) pp. 175-185.

[5] George Reynolds and J. Ross Beveridge, "Searching for Geometric Structure in Images of Natural Scenes" *Proc. of the DARPA Image Understanding Workshop*, Los Angeles, CA., Feb. 1987. (Morgan Kaufman Publishers, Inc., New York) pp. 257-271. Also COINS Technical Report 87-03, University of Massachusetts at Amherst, January 1987.

[6] John K. Tsotsos. "A 'Complexity Level' Analysis of Vision" *Proc. of the Int. Conf. of Computer Vision*, London, 1987. pp. 346-355.

[7] J.R. Ullman, "An Algorithm for Subgraph Isomorphism", *Journal of the ACM* 23(1) (Jan. 1976) pp. 31-42.

[8] Richard Weiss and Michael Boldt. "Geometric Grouping Applied to Straight Lines", *Proc. of CVPR*, Miami, FL. June 22-26 1986. pp. 489-495.