

which is contradicted by Fig. 10.15, given the previous definition of ABOVE.

One common way around these problems is to associate quantitative, "continuous" information with relations (section 10.3.2 and later examples).

10.3 SEMANTIC NET EXAMPLES

Examples of semantic nets abound throughout Part IV. Two more examples illustrate the power of the notions. The first example is described very generally, the second in detail.

10.3.1 Frame Implementations

Frame system theory [Minsky 1975] is a way of explaining our quick access to important aspects of a (perhaps perceptual) situation. It is a provocative and controversial idea, and the reader should consult the References for a full treatment. Implementationally, a frame may be realized by a partition; a frame is a "chunk" of related structure.

Associating related "chunks" of knowledge into manipulable units is a powerful and widespread idea in artificial intelligence [Hayes 1980; Hendrix 1979] as well as psychology. These chunks go by several names: units, frames, partitions, schemata, depictions, scripts, and so forth [Schank and Abelson 1977; Moore and Newell 1973; Roberts and Goldstein 1977; Hayes* 1977; Bobrow and Winograd 1977, 1979; Stefik 1979; Lehnert and Wilks 1979; Rumelhart et al. 1972].

Frames systems incorporate a theory of associative recall in which one selects frames from memory that are relevant to the situation in which one finds oneself. These frames include several kinds of information. Most important, frames have *slots* which contain details of the viewing situation. Frame theory dictates a strictly specific and prototypical structure for frames. That is, the number and type of slots for a particular type of frame are immutable and specified in advance. Further, frames represent specific prototype situations; many slots have default values; this is where expectations and prior knowledge come from. These default values may be disconfirmed by perceptual evidence; if they are, the frame can contain information about what actions to take to fill the slot. Some slots are to be filled in by investigation. Thus a frame is a set of expectations to be confirmed or disconfirmed

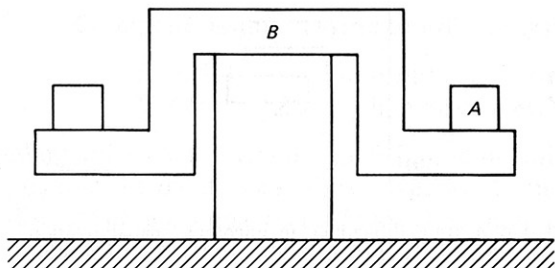


Fig. 10.15 A counterexample to $\text{SUPPORTS}(B, A) \Rightarrow \text{ABOVE}(A, B)$.

and actions to pursue in various contingencies. One common action is to “bring in another frame.”

The theory is that based on a partial match of a frame’s defining slots, a frame can be “brought to mind.” The retrieval is much like jumping to a conclusion based on partial evidence. Once the frame is proposed, its slots must be matched up with reality; thus we have the initial major hypothesis that the frame represents, which itself consists of a number of minor subhypotheses to be verified. A frame may have other frames in its slots, and so frames may be linked into “frame systems” that are themselves associatively related. (Consider, for example, the linked perceptual frames for being just outside a theater and for being just inside.) Transformations between frames correspond to the effects of relevant actions. Thus the hypotheses can suggest one another. “Thinking always begins with suggestive but imperfect plans and images; these are progressively replaced by better—but usually still imperfect—ideas” [Minsky 1975].

Frame theory is controversial and has its share of technical problems [Hinton 1977]. The most important of these are the following.

1. Multiple instances of concepts seem to call for copying frames (since the instances may have different slotfillers). Hence, one loses the economy of a preexisting structure.
2. Often, objects have variable numbers of components (wheels on a truck, runways in an airport). The natural representation seems to be a rule for constructing examples, not some specific example.
3. Default values seem inadequate to express legal ranges of slot-filling values or dependencies between their properties.
4. Property inheritance is an important capability that semantic nets can implement with “is a” or “element-of” hierarchies. However, such hierarchies raise the question of which frame to copy when a particular individual is being perceived. Should one copy the generic Mammal frame or the more specific Camel frame, for instance. Surely, it is redundant for the Camel frame to duplicate all the slots in the Mammal frame. Yet our perceptual task may call for a particular slot to be filled, and it is painful not to be able to tell where any particular slot resides.

Nevertheless, where these disadvantages can be circumvented or are irrelevant, frames are seeing increasing use. They are a natural organizing tool for complex data.

10.3.2 Location Networks

This section describes a system for associating geometric analogical data with a semantic net structure which is sometimes like a frame with special “evaluation” rules. The system is a geometrical inference mechanism that computes (or infers) two-dimensional search areas in an image [Russell 1979]. Such networks have found use in both aerial image applications [Brooks and Binford 1980; Nevatia and Price 1978] and medical image applications [Ballard et al. 1979].

A *location network* is a network representation of geometric point sets related by set-theoretic and geometric operations such as set intersection and union, distance calculation, and so forth. The operations correspond to restrictions on the location of objects in the world. These restrictions, or rules, are dictated by cultural or physical facts.

Each internal node of the location network contains a geometric *operation*, a list of *arguments* for the operation, and a *result* of the operation. For instance, a node might represent the set-theoretic union of two argument point sets, and the result would be a point set. Inference is performed by *evaluating* the net; evaluating all its operations to derive a point set for the top (root) operation.

The network thus has a hierarchy of ancestors and descendants imposed on it through the argument links. At the bottom of this hierarchy are *data nodes* which contain no operation or arguments, only geometric data. Each node is in one of three states: A node is *up-to-date* if the data attached to it are currently considered to be accurate. It is *out-of-date* if the data in it are known to be incomplete, inaccurate, or missing. It is *hypothesized* if its contents have been created by net evaluation but not verified in the image.

In a common application, the expected relative locations of features in a scene are encoded in a network, which thus models the expected structure of the image. The primitive set of geometric relations between objects is made up of four different types of operations.

1. *Directional* operations (left, reflect, north, up, down, and so on) specify a point set with the obvious locations and orientations to another.
2. *Area* operations (close-to, in-quadrilateral, in-circle and so on) create a point set with a non-directional relation to another.
3. *Set* operations (union, difference and intersection) perform the obvious set operations.
4. *Predicates* on areas allow point sets to be filtered out of consideration by measuring some characteristic of the data. For example, a predicate testing width, length, or area against some value restricts the size of sets to be those within a permissible range.

The location of the aeration tank in a sewage treatment plant provides a specific example. The aeration tank is often a rectangular tank surrounded on either end by circular sludge and sedimentation tanks (Fig. 10.16). As a general rule, sewage flows from the sedimentation tanks to aeration tanks and finally through to the sludge tanks. This design permits the use of the following types of restrictions on the location of the aeration tanks.

Rule 1: "Aeration tanks are located somewhere close to both the sludge tanks and the sedimentation tanks."

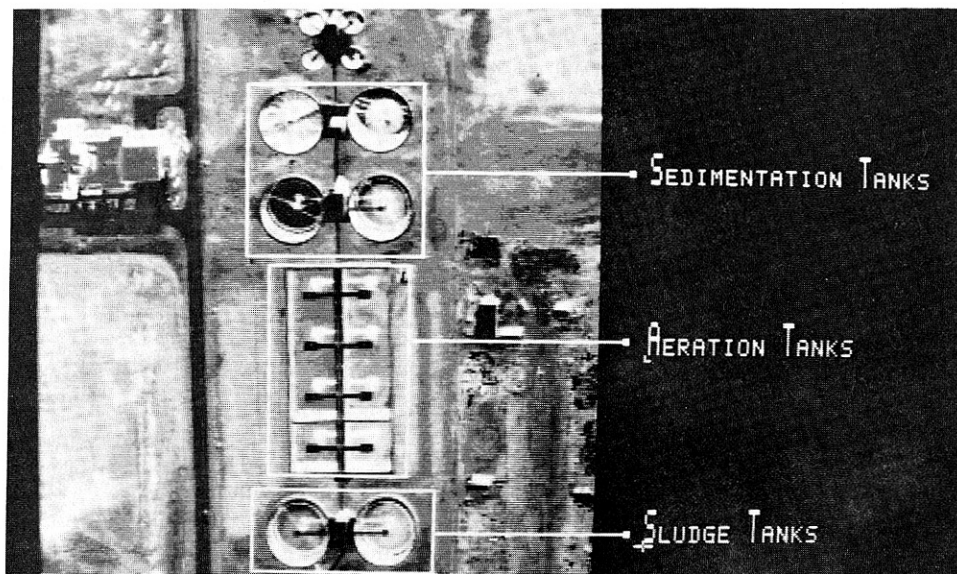


Fig. 10.16 Aerial image of a sewage plant.

The various tanks cannot occupy the same space, so:

Rule 2: "Aeration tanks must not be too close to either the sludge or sedimentation tanks."

Rule 1 is translated to the following network relations.

CLOSE-TO(Union (LocSludgeTanks, LocSedTanks), Distance X)

Rule 2 is translated to

NOT-IN(Union (LocSludgeTanks, LocSedTanks), Distance Y)

The network describing the probable location of the aeration tanks embodies both of these rules. Rule 1 determines an area that is close to both groupings of tanks and Rule 2 eliminates a portion of that area. Thinking of the image as a point set, a set difference operation can remove the area given by Rule 2 from that specified by Rule 1. Figure 10.17 shows the final network that incorporates both rules.

Of course, there could be places where the aeration tanks might be located very far away or perhaps violate some other rule. It is important to note that, like the frames of Section 10.3.1, location networks give prototypical, likely locations for an object. They can work very well for stereotyped scenes, and might fail to perform in novel situations.

The Evaluation Mechanism

The network is interpreted (evaluated) by a program that works top-down in a recursive fashion, storing the partial results of each rule at the topmost node as-

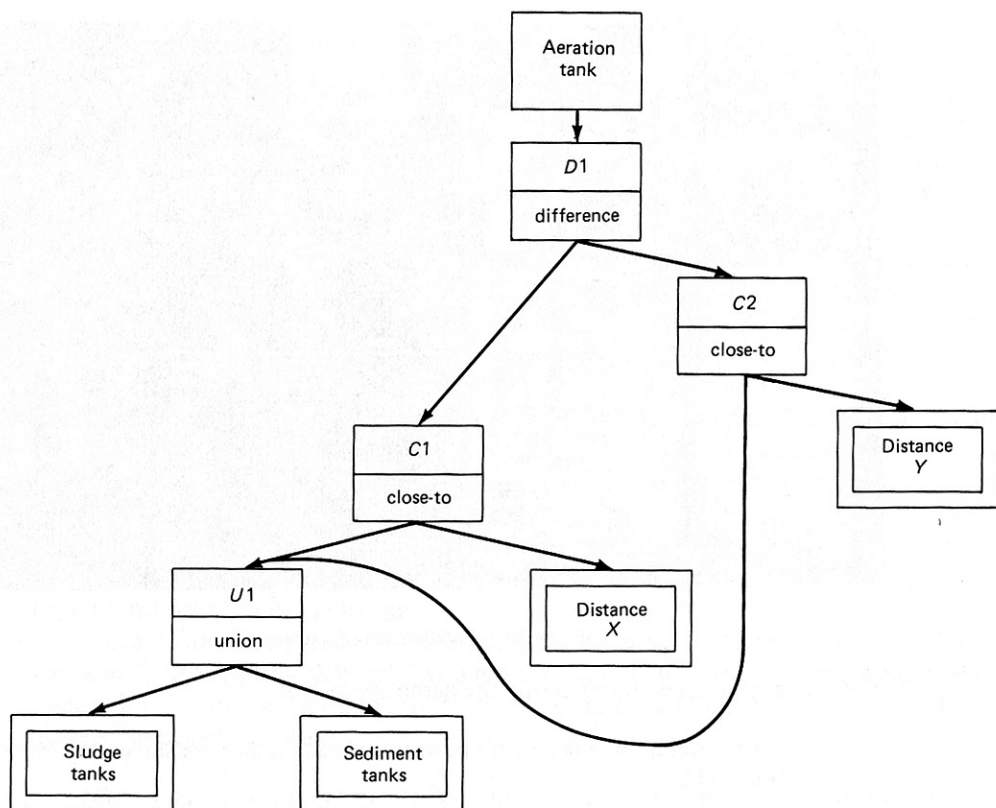


Fig. 10.17 Constraint network for aeration tank.

sociated with that rule (with a few exceptions). Evaluation starts with the root node. In most networks, this node is an operation node. An operation node is evaluated by first evaluating all its arguments, and then applying its operation to those results. Its own result is then available to the node of the network that called for its evaluation.

Data nodes may already contain results which might come from a map or from the previous application of vision operators. At some point in the course of the evaluation, the evaluator may reach a node that has already been evaluated and is marked up-to-date or hypothesized (such a node contains the results of evaluation below that point). The results of this node are returned and used exactly as if it were a data node. Out-of-date nodes cause the evaluation mechanism to execute a low-level procedure to establish the location of the feature. If the procedure is unable to establish the status of the object firmly within its resource limits, the status will remain out-of-date. At any time, out-of-date nodes may be processed without having to recompute any up-to-date nodes. A node marked hypothesized has a value, usually supplied by an inference process, and not verified by low-level image analysis. Hypothesized data may be used in inferences: the results of all inferences based on hypothesized data are marked hypothesized as well.

If a data node ever has its value changed (say, by an independent process that adds new information), all its ancestors are marked out-of-date. Thus the root node will indicate an out-of-date status, but only those nodes on the out-of-date path must be reevaluated to bring the network up to date. Figure 10.18 shows the operation of the aeration tank network of Fig. 10.17 on the input of Fig. 10.16. In this case the initial feature data were a single sludge tank and a single sedimentation tank. Suppose additional work is done to find the location of the remaining sludge and sediment tanks in the image. This causes a reevaluation of the network, and the new result more accurately reflects the actual location of the aeration tanks.

Properties of Location Networks

The location network provides a very general example of use of semantic nets in computer vision.

1. It serves as a data base of point sets and geometric information. The truth status of items in the network is explicitly maintained and depends on incoming information and operations performed on the net.
2. It is an expansion of a geometric expression into a tree, which makes the order of evaluation explicit and in which the partial results are kept for each geometric calculation. Thus it provides efficient updating when some but not all the partial results change in a reevaluation.
3. It provides a way to make geometrical inferences without losing track of the hypothetical nature of assumptions. The tree structure records dependencies among hypotheses and geometrical results, and so upon invalidation of a geometric hypothesis the consequences (here, what other nodes have their values affected) are explicit. The record of dependencies solves a major problem in automated inference systems.
4. It reflects implicit universal quantification. The network claims to represent true relations whose explicit arguments must be filled in as the network is "instantiated" with real data.
5. It has a "flat" semantics. There are no element-of hierarchies or partitions.
6. The concept of "individual" is flexible. A point set can contain multiple disconnected components corresponding to different world objects. In set operations, such an assemblage acts like an explicit set union of the components. An "individual" in the network may thus correspond to multiple individual point (sub)sets in the world.
7. The network allows use of partial knowledge. A set-theoretic semantics of existence and location allows modeling of an unknown location by the set-theoretic universe (the possible location is totally unconstrained). If something is known not to exist in a particular image, its "location" is the null set. Generally, a location is a point set.
8. The set-theoretic semantics allows useful punning on set union and the OR operation, and set intersection and the AND operation. If a dock is on the

shoreline AND near a town, the search for docks need only be carried out in the intersection of the locations.

10.4 CONTROL ISSUES IN COMPLEX VISION SYSTEMS

Computer vision involves the control of large, complex information-processing tasks. Intelligent biological systems solve this control problem. They seem to have complicated control strategies, allowing dynamic allocation of computational resources, parallelism, interrupt-driven shifts of attention, and incremental behavior modification. This section explores different strategies for controlling the complex information processing involved in vision. Appendix 2 contains specific

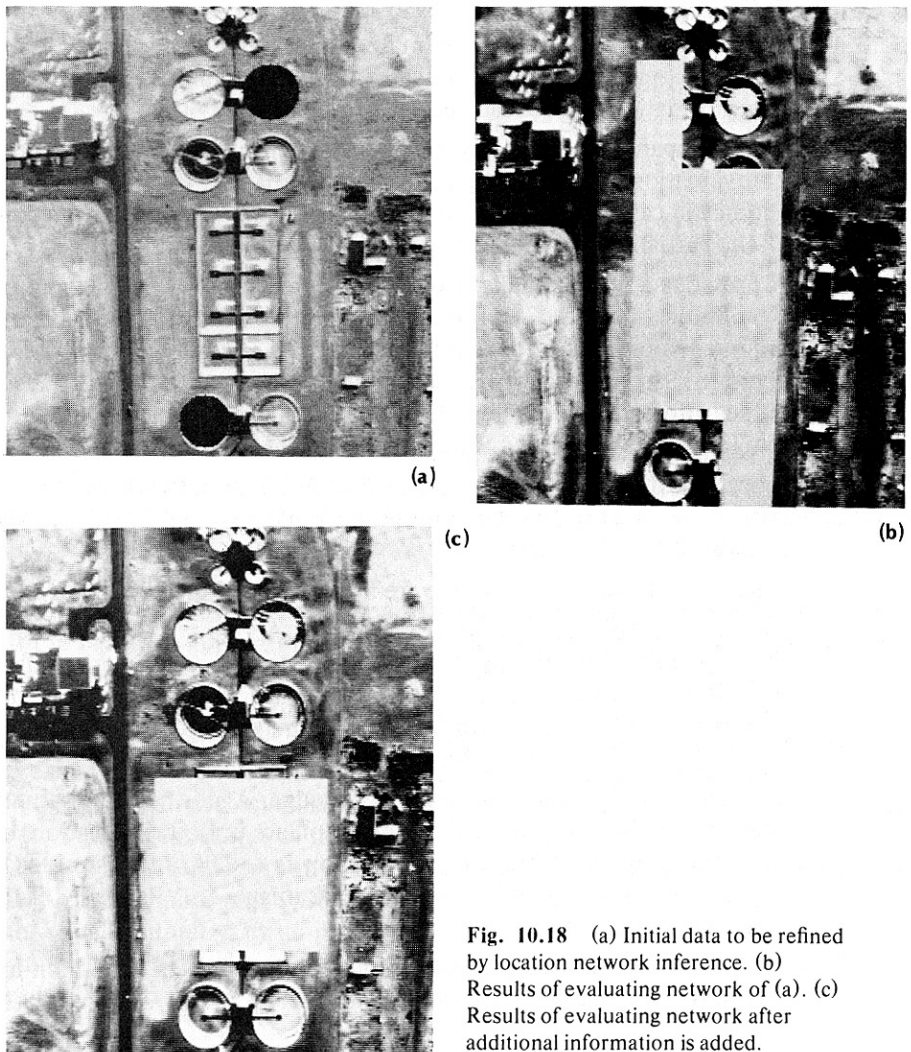


Fig. 10.18 (a) Initial data to be refined by location network inference. (b) Results of evaluating network of (a). (c) Results of evaluating network after additional information is added.