# Using Generic Knowledge in Analysis of Aerial Scenes: A Case Study

Andres Huertas, William Cole and Ramakant Nevatia *
Institute for Robotics and Intelligent Systems
Departments of Electrical Engineering and Computer Science
University of Southern California
Los Angeles, California 90089-0273

## Abstract

Our goal is to produce high-quality symbolic descriptions from aerial scenes. We have chosen to work in the domain of large commercial airport complexes. Such scenes have a variety of features such as the transportation network, building structures, and mobile objects. This paper concentrates on detection and description of the transportation network (runways and taxiways). We illustrate the complexities of this problem and how it can be solved by using geometrical context and *generic* airport domain knowledge.

## 1 Introduction

Automatic analysis of complex aerial images is an important and challenging problem. Our aim is to compute rich, symbolic descriptions from an image which may be used for a variety of tasks such as making a cartographic map, change detection and guidance.

Aerial images are highly complex as they contain a large number (possibly hundreds) of both man-made and natural objects of a large variety, only some of which may be of interest. The image is also very complex, from a signal point of view, and sophisticated processing techniques are necessary. But, the scene can not be analyzed in terms of signal alone; we must also use our cultural and world knowledge to reduce the ambiguities inherent in images. The interesting issue, then, is what kind of knowledge is to be used and how.

We have chosen major commercial airport complexes as a test domain. Airports contain a variety of objects, such as the transportation network (runways, taxiways, and roads), building structures (hangars, terminals, storage warehouses, fuel storage farms), and mobile objects (automobiles, aircraft, humans). The airport complexes are under continual changes, usually due to expansion. The images themselves are rather complex due to the large number of objects present in them. However, a variety of such images are available with only moderate effort.

*This research was supported by the Defense Advanced Research Projects Agency, monitored by the Air Force Wright Aeronautical Laboratories under contract F33615-87-C-1436

In our analysis, we do not assume specific knowledge of the scene, such as would be given by a detailed, current map of the specific airport complex. Instead, we only have *generic* information that the scene being viewed is an airport complex. We do have, and use, the knowledge about airports and the structures contained in them.

Our approach in the design of the system is that it must be modular and that the modules interact mostly at high, symbolic levels. In airports, for example the modules may be for detecting and describing the transportation network, the buildings and the mobile objects. Detection of one type of object, such as a taxiway, may aid in increasing the confidence of a structure believed to be a passenger terminal (and *vice-versa).* However, we believe that such interaction takes place at a high level, after symbolic, object level hypotheses have been formed. This process can be considered to be hierarchical; each module has sub-modules that operate in a similar way. Thus, the transportation network module may consist of runway, taxiway and road modules; each of which operates somewhat independently but uses context provided by the detection of other structures. Some structures may be more prominent and easier to detect, for example, runways are easier to detect than taxiways. In that case, the former provides the context for detection of the latter.

In this paper, we will concentrate on the problem of detecting and describing the transportation network in an airport complex, specifically the runways and taxiways, to illustrate our methodology. (We have also developed methods of detecting and describing three-dimensional structures such as buildings [Huertas and Nevatia, 1988, Mohan and Nevatia, 1988] however, we will not discuss those here due to lack of space). The runways and taxiways may appear to be rather obvious and prominent features to us, modeled easily as long, thin rectangular strips of uniform brightness. However, such is not the case. Runways and taxiways contain a number of surface markings: some to aid a pilot and others caused by dirt, oil spots, exhaust fumes etc. Surface composition is not always uniform when runways and taxiways are extended or repaired. Presence of other objects, such as vehicles and airplanes, on or in the vicinity of the taxiways and runways further violates the simple model. Due to these and other reasons, the low-level segmentation of such scenes turns out to be highly frag-

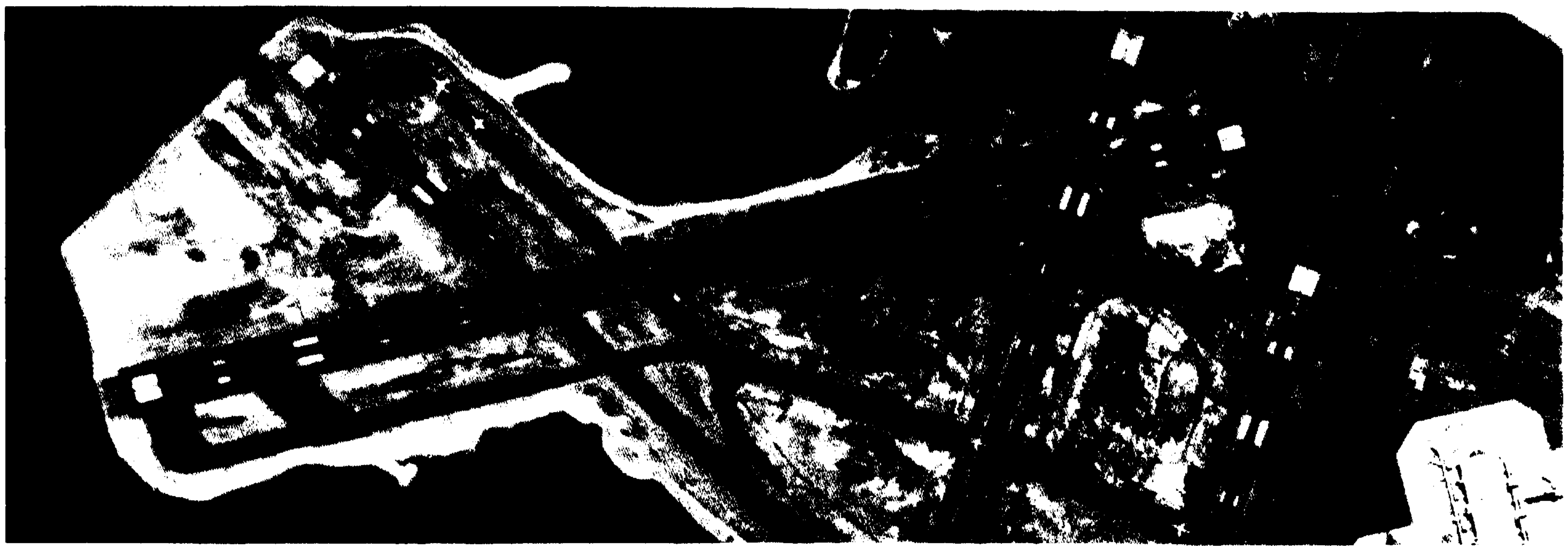

Figure 1: Logan International Airport image (LOGAN)

mented with a very large number of low-level features detected, only a small number of which are relevant to our goal. Our task now is to use the generic knowledge of runways and taxiways to extract them from this mass of confusing data.

Our approach is basically one of "hypothesize and verify". Various grouping operations relying on geometry, object shape and context form hypotheses that are then verified according to some desired attributes. Our system detects runways first, as they are more prominent and can provide the needed context for detection of taxiways (and many other objects in the scene). Runway hypotheses are formed by utilizing grouping operations of continuity, collinearity, parallelism and symmetry. Verification consists of finding the appropriate markings (such as center lines, "threshold" marks etc.) that are expected on the runways in a major airport. Once some of the marks are found, the context provided by them can be used to look for additional marks that may be too faint to find otherwise.

Detection of taxiways is somewhat similar to that of runways, however, the taxiways are much less constrained in shape and appearance and the context of already detected runways helps in detection and verification of taxiways. The converse is also true, *i.e.* finding taxiways connected to a runway can help increase the confidence of the detected runway.

We describe our technique for runways detection briefly in section 2; further details of earlier work are given in [Huertas *et* a/., 1987]. Detection of taxiways and junctions are described in more detail in sections 3 and 4.

There have been relatively few efforts in recent years to analyze complex, cultural aerial scenes. McKeown and his associates [McKeown *et* a/., 1985, McKeown and Harvey, 1987] at CMU represent an exception. Our work is related to theirs, but is largely complementary. The major difference is perhaps in the way domain knowledge is used. We believe that our approach is much more modular and the use of domain knowledge in our system is at much higher levels, with the lower levels relying on much more complex geometrical knowledge.

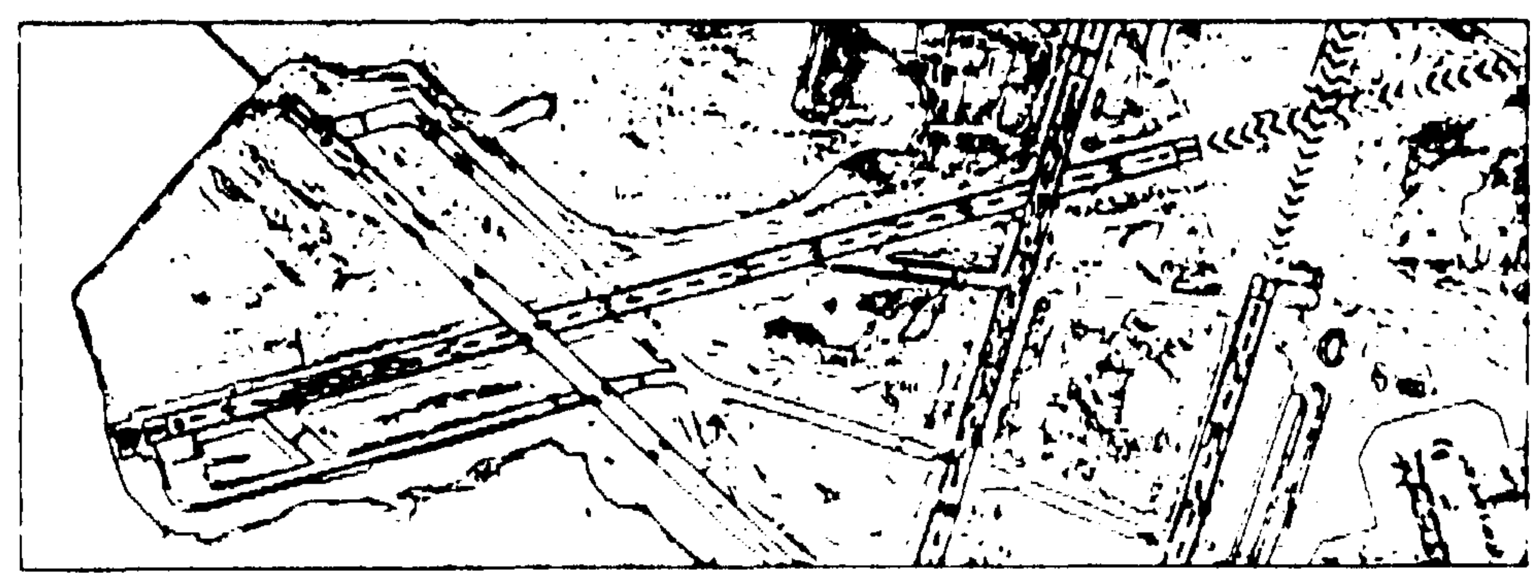

Figure 2: Line Segments from LOGAN image

## 2 Detection of Runways

Runways are perhaps the most prominent structure in an airport scene. In our system, no external context (from other objects) is available for detecting runways, though this module use geometrical context internally.

Figure 1 shows a portion (LOGAN:800 x 2200 resolution) of Logan International Airport in Boston. We first use our LINEAR software [Nevatia and Babu, 1980, Canny, 1986] to compute line segments (figure 2) and "anti-parallels" (parallel lines of opposite contrast, we will call them *apars)* (figure 3). We estimate the dominant apar orientations and widths (focus of attention) by length-weighted histograms. We then extract and join potential runway fragments into runway hypotheses by a number of grouping steps using continuity and collinearity. These hypotheses are verified by looking for the markings that they are supposed to have [FAA, 1980]. We look for centerlines, sidestripes, threshold markings, touchdown markings, large distance markings, small distance markings, and blast pad markings (figure 4). Our system uses a feedback mechanism to allow further search for evidence of centerlines and blastpad markings, using the previously detected markings as context. Runways are described in terms of position, length, width and orientation, and associated markings.

## 3 Detection of Taxiways

Taxiways are much more complex objects than runways, as they can have a wider range in their geometrical pa-

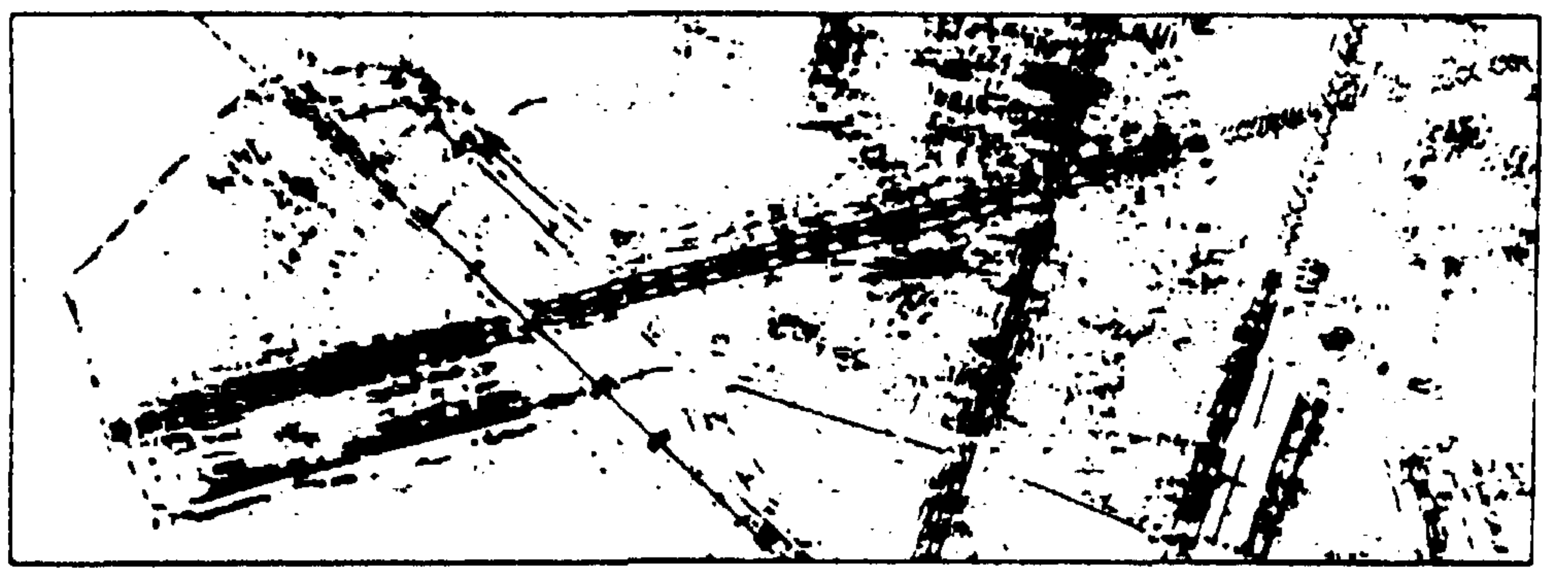

Figure 3: Anti-parallels from segments in LOGAN image

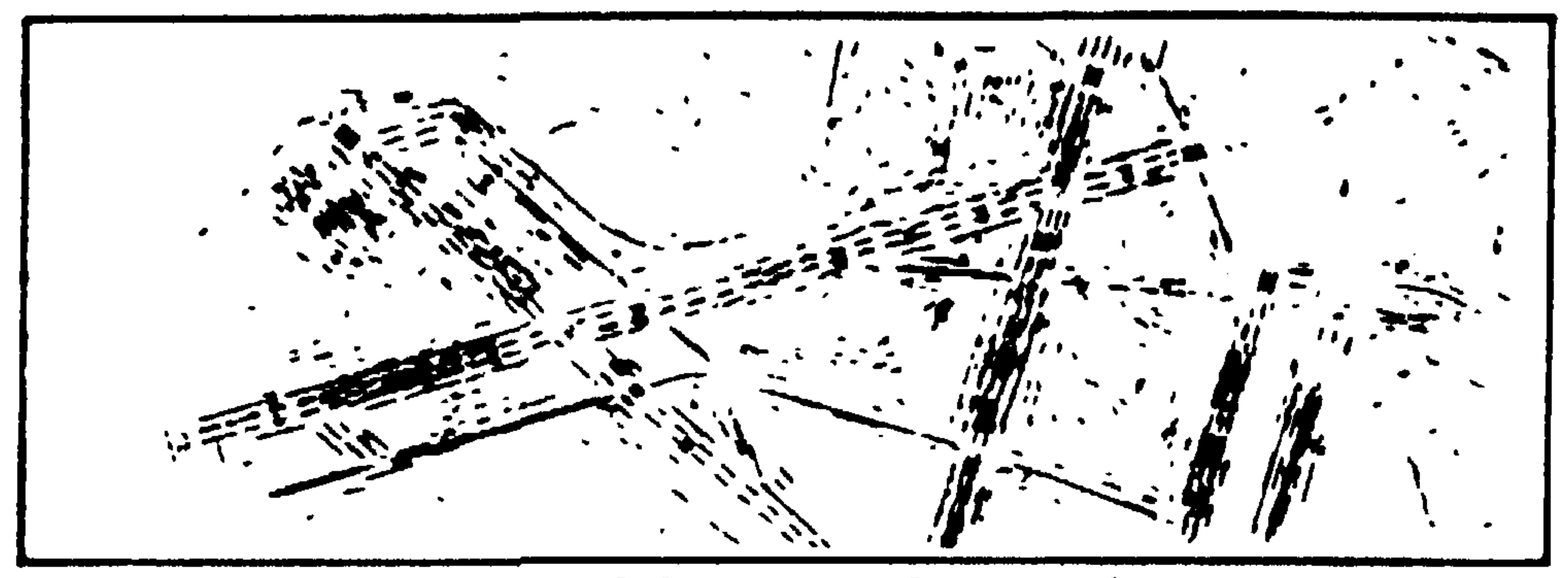

**Figure 5: Apars in width group for Taxiways in LOGAN**

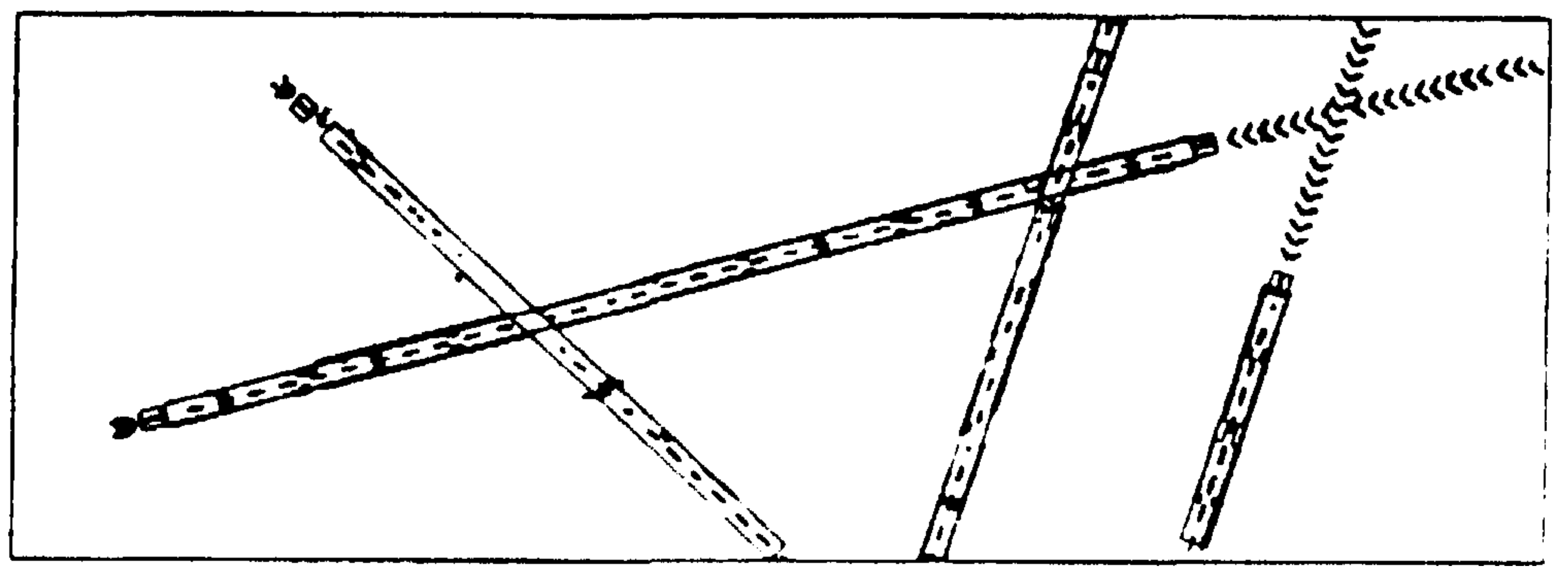

Figure 4: Verified Runways with Markings detected

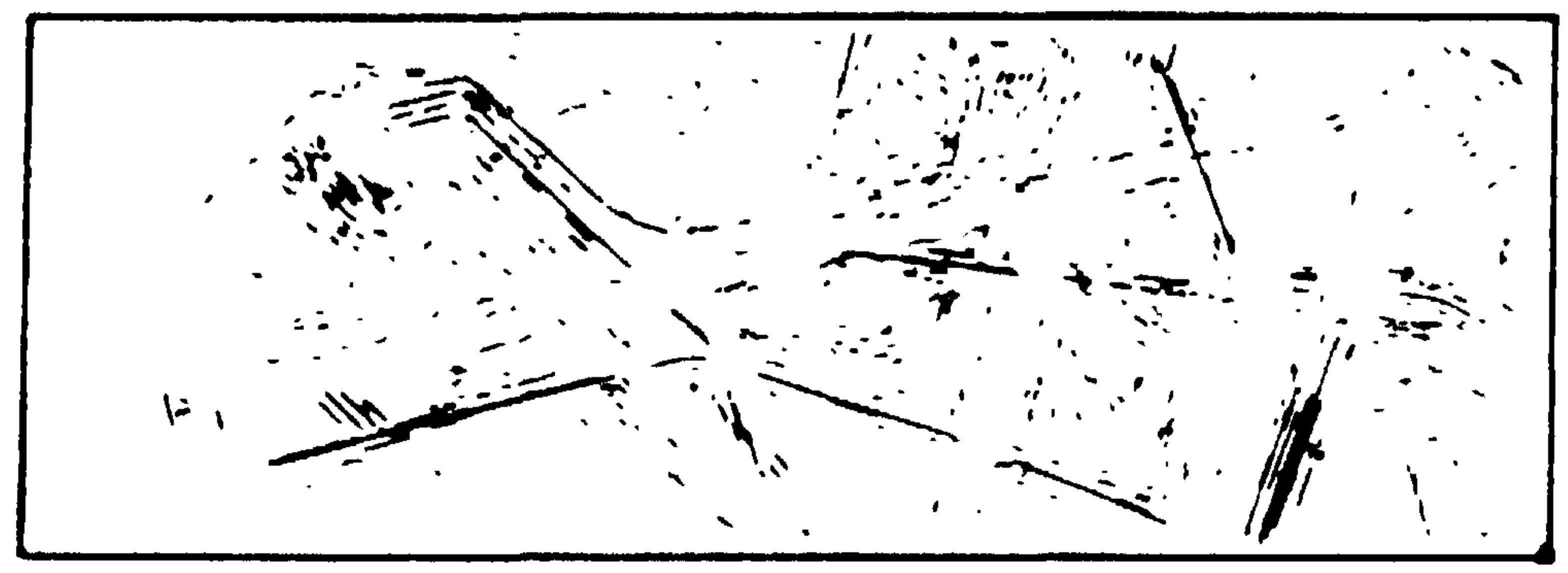

Figure 6: Apars joined on collinearity in LOGAN

rameters (see figure 1); they can be short or long, have a variety of widths, be straight or curved, and connect a variety of airport components. However, besides generic knowledge we make use of the context provided by the detected runways to help detect the taxiways. Our module first finds the "long" straight portions of taxiways and then, it attempts to extend these portions also based on context.

Taxiways are verified by looking for two types of surface markings: the continuous sidestripes that bound them, and the continuous centerline down the middle of the roadway. Although our method to detect taxiways is similar to that for runway detection, below we give details of the method as this module is new and has not been published elsewhere.

### 3.1 Hypotheses Formation

As mentioned above, this process is aided by the descriptions of previously detected runways, and also by knowledge on the constraints imposed by airport design procedures [Ashford and Wright, 1984]. We know for instance, the minimum acceptable distance between a taxiway and a runway if they are parallel, or the minimum angle that a taxiway may form with a runway. We also know that taxiways do not cross but join runways. Taxiway crossings however, are allowed.

The first step in detecting taxiways is to find long fragments which may correspond to fragments of taxiways. The apars representing these fragments (figure 5) are selected from the apars shown in figure 3 in a manner analogous to the selection of potential runway fragments (see [Huertas *et al*., 1987]): they have a range of widths, and either are parallel to a runway or, form an angle greater than 25° with a runway. If the distance between an apar and the runway is greater than the allowed distance between parallel runways and taxiways, the angle constraint is not applied.

Next, the selected apars are joined on continuity along anti-parallel segments, and then they are joined on collinearity (figure 6).

The second step attempts to extend long portions of taxiway fragments. It is known that the purpose of taxiways is to facilitate aircraft moving from one section of the airport to another, thus taxiways do not arbitrarily end as do runways. We attempt to extend these taxiway fragments by invoking the following context dependent processes:

1. Extension based on Aircraft Support: A large aircraft on a taxiway will cause the taxiway hypothesis to fragment, thus to extend the taxiway fragments, we first try to detect aircraft by looking for symmetries due to the aircraft wings and fuselage at each fragment end. If an aircraft is detected, the taxiway is extended the length of the aircraft.
2. Extension based on Runway Context: We attempt to extend or discard taxiway hypotheses fragments based on their spatial relationships to verified runways in the scene. The following steps are taken:
   (a) Fragment intersects runway: The taxiway hypothesis fragments are extended until they intersect a runway. If the intersection angle is greater than the minimum intersection angle and the distance between the taxiway hypothesis fragment and the runway intersection point is small, we look for additional evidence to extend the fragment into the runway. This evidence includes checking for shorter apars collinear to the taxiway in this region and, failing this, the detection of aircraft in this region. If we find sufficient evidence, the taxiway hypothesis is extended into the runway.
   (b) Fragment is parallel to runway: If the taxiway fragments are parallel to one of the verified runways, we look for small wide apars joining the end of the taxiway fragment to the runway indicating the presence of a taxiway apron.

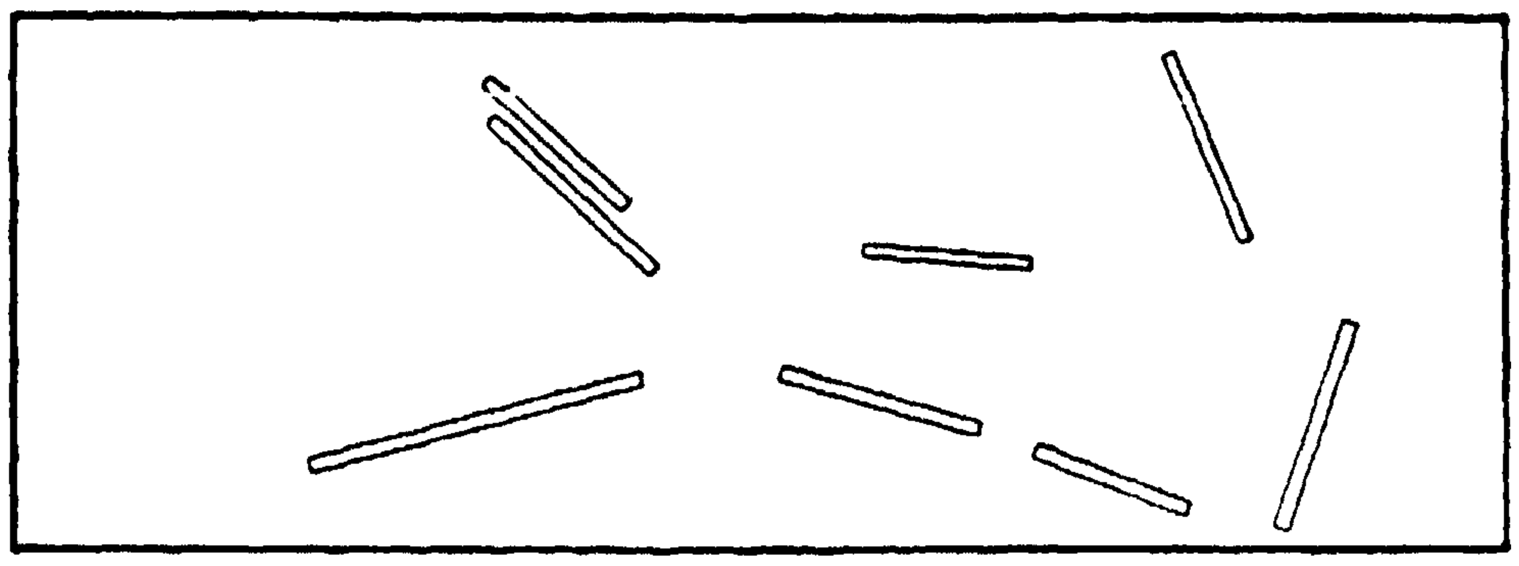

Figure 7: Apars representing hypotheses of straight por tions of taxiways

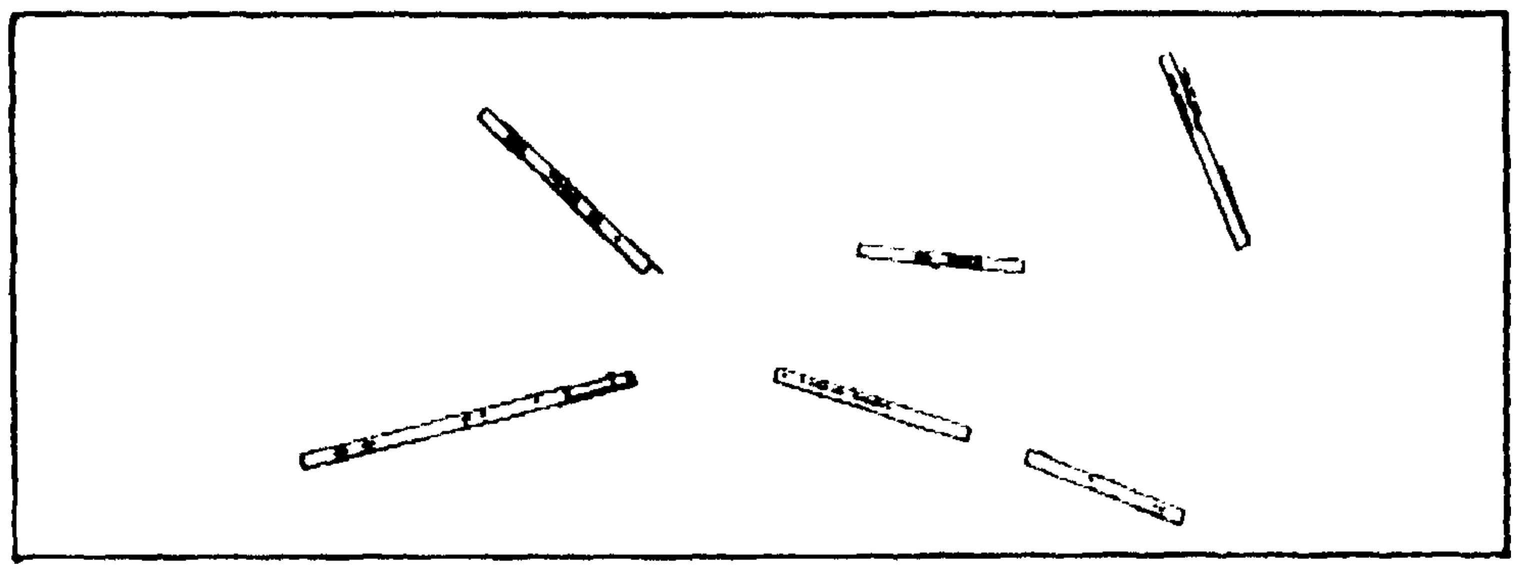

Figure 8: Verified taxiways and markings

(c) Extension based on Taxi way Intersection: (see next section)

(d) Extension based on Resegmentation: It is possible that a material change in the taxiway caused problems for the initial grouping processes. We attempt to extend the taxiway fragments by resegmenting image windows extending beyond the fragments' ends, and looking for evidence of taxiway continuation. This process is continued until no further evidence is found. At this point, we repeat steps (a) and (b).

The apars representing hypotheses of straight portions of taxiways are shown in figure 7. In this result, only process 1 above was applied. Extension of taxiways based on intersections (process 2c) is described below.

### 3.2 Hypotheses Verification

We verify taxiway hypotheses by looking for evidence of markings along the roadway. We expect to find sidestripes and a continuous centerline, indicating the allowed pathways. We look for evidence of markings in the set of thin bright apars. As with the runways, a resegmentation step can be applied to locate further evidence of markings. The verified taxiway hypotheses are shown in figure 8.

## 4 Description of Junctions and Connections

The use of context is essential in this task. The network of runways and taxiways in a major commercial airport can be very complex. This module attemps to describe the junctions among them by explicitly locating the boundaries, or portions of the boundaries, of the sections of roadways that connect the previously detected runways and straight portions of taxiways.

Figure 9: Complex junction at LOGAN

The accurate description of the junctions between pathways also help determine their function. Some are used as holding aprons (usually wide and normal to runway); some are exit ramps (the closer to the end of the runway, the smaller the angle between them. The angle itself determines the allowed exit speed.); some are merely connecting pathways; the continuous centerline determines the "legal" turns and paths, and so on.

Junctions among taxiways and between taxiways and runways can vary widely in their complexity and configurations. The image in figure 9, a portion of the image previously shown in figure 1, shows the intersection of four taxiways, and connections between taxiways and runways when these are not parallel to each other. An example of another configuration is given later.

To process these complex junctions we rely on the *context:* The previously detected runways and taxiways provide a very rich set of geometric constraints. The search for junction boundaries — straight or curved — thus, is facilitated by the geometric interrelationships among the nearby elements (position, length, width, and orientation of nearby runways and taxiways), as well as by the geometric constraints imposed by airport design procedures. Details are given below, as these module is new and has not been described elsewhere.

In our method we first form junction hypotheses by looking at the underlying intensity edges for evidence of portions of boundaries. The valid connections are then determined by looking for evidence of markings (centerline) associated with taxiways.

### 4.1 Hypotheses Formation

Each pair of elements (runways or taxiways) determines two search windows where we look for the junction boundaries. The shape of the window is constrained by the available context, that is, the previously detected runways and taxiways. Our method distinguishes two types of junctions: 1-junctions (typically among taxiways), and t-junctions (typically between taxiways and runways). More complex junctions are viewed as over-

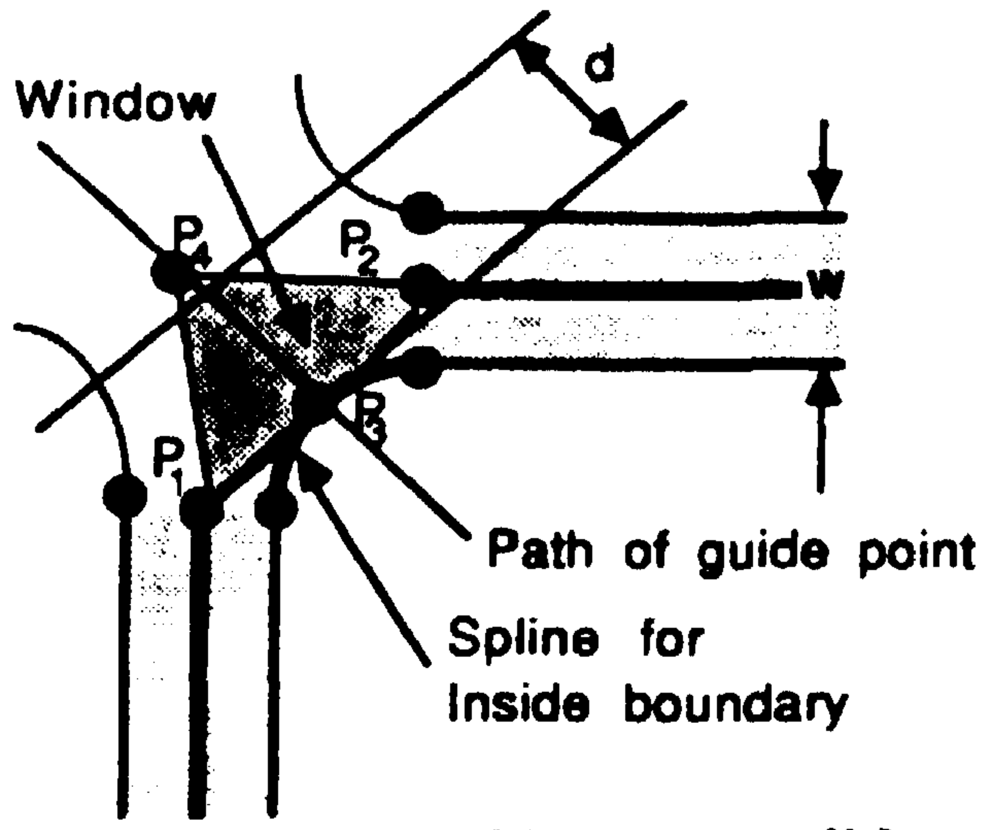


Figure 12: Search for markings to validate l-junction hypotheses

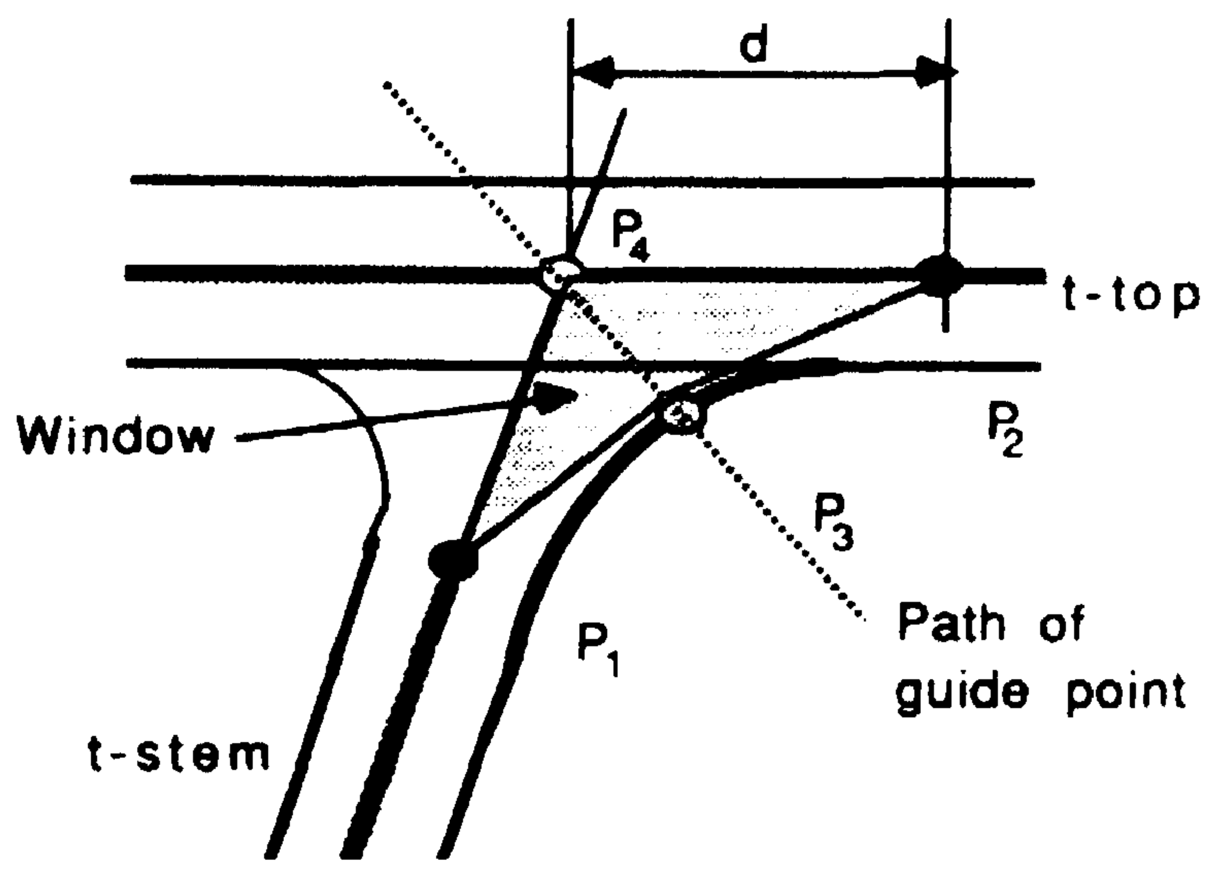


Figure 13: Search for markings to validate t-junction hypotheses

is the middle point between PI and P2. P4 is the intersection of the apars representing the two elements. For a t-junction (figure 13) P2 is computed on the t-top element, at a distance *d* from P4. The distance *d* is proportional to the width of the t-stem element.

3. Look for marking boundary: The search process is similar to that of searching for inside boundaries. We compute a series of splines using the anchor points and varying the guide point. In addition, we look for the centerline along the straight path from PI and P4.

### 4.3 Results

We now present results for two examples. The first deals with the intersection of taxiways shown previously in figure 9. Figure 14 shows the original, thresholded underlying edges, the taxiway and runway context (shaded areas), and all possible connections among the taxiways. These correspond to the splines intersecting the largest number of underlying edges within each window searched for potential inside and outside boundaries.

Figure 15 shows the connections that meet the hypotheses criteria discussed above. Figure 16 shows the resegmented edges, the inside boundaries, and the evidence of centerlines, and thus, the verified hypotheses and alowed paths.

The image in figure 17 shows the connecting paths between a taxiway and a runway when they are parallel

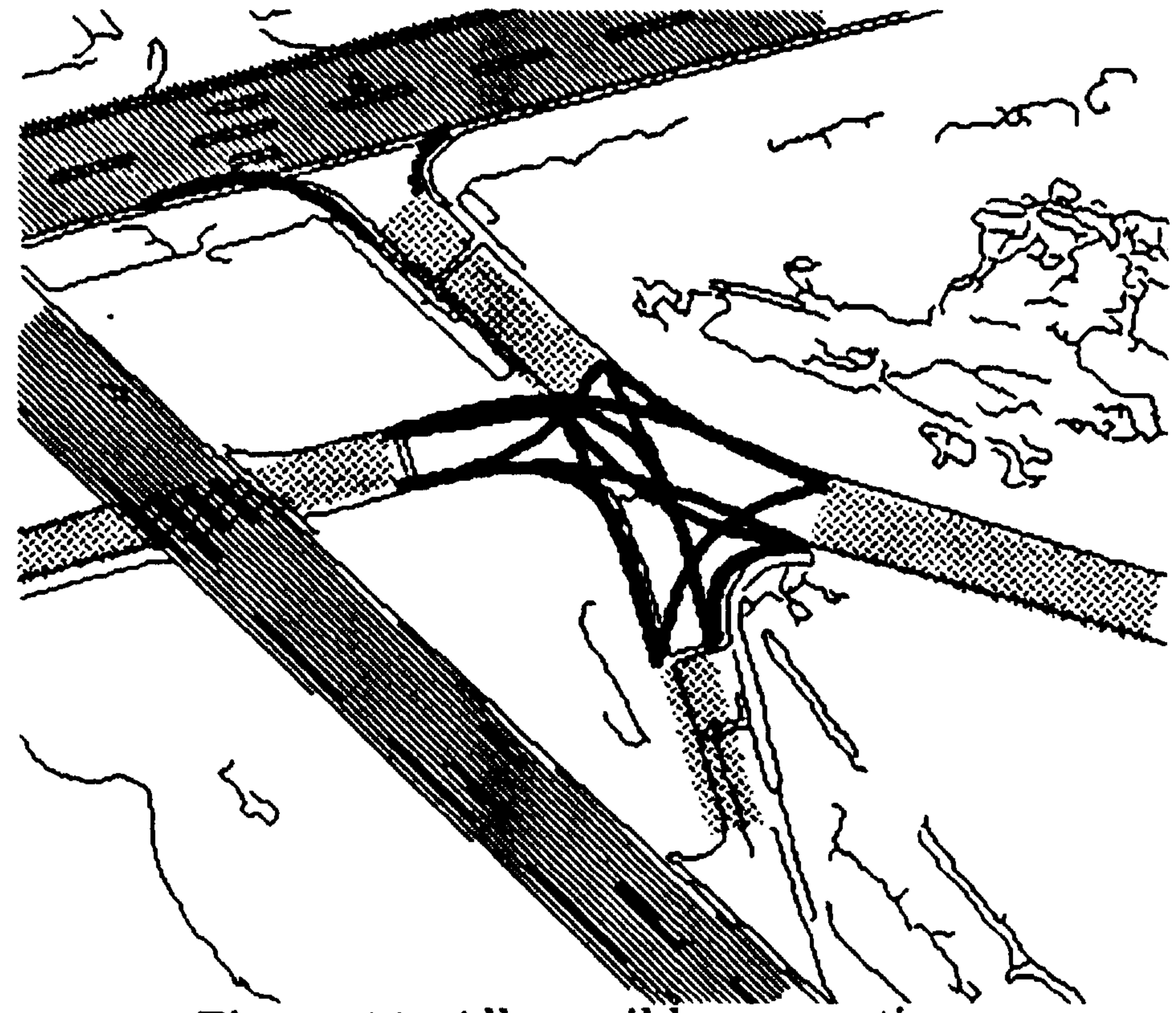

Figure 14: All possible connections

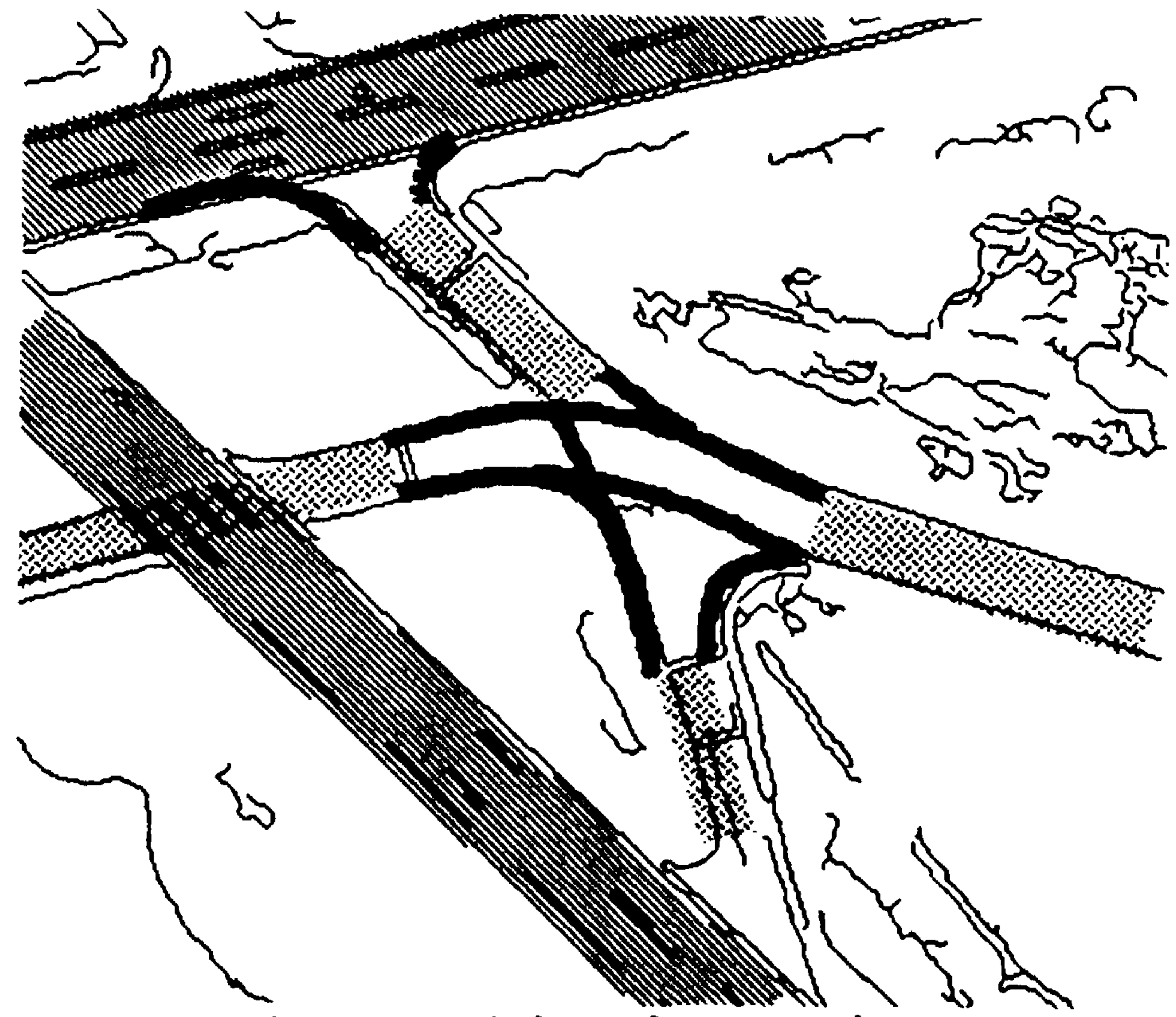

Figure 15: Selected connections

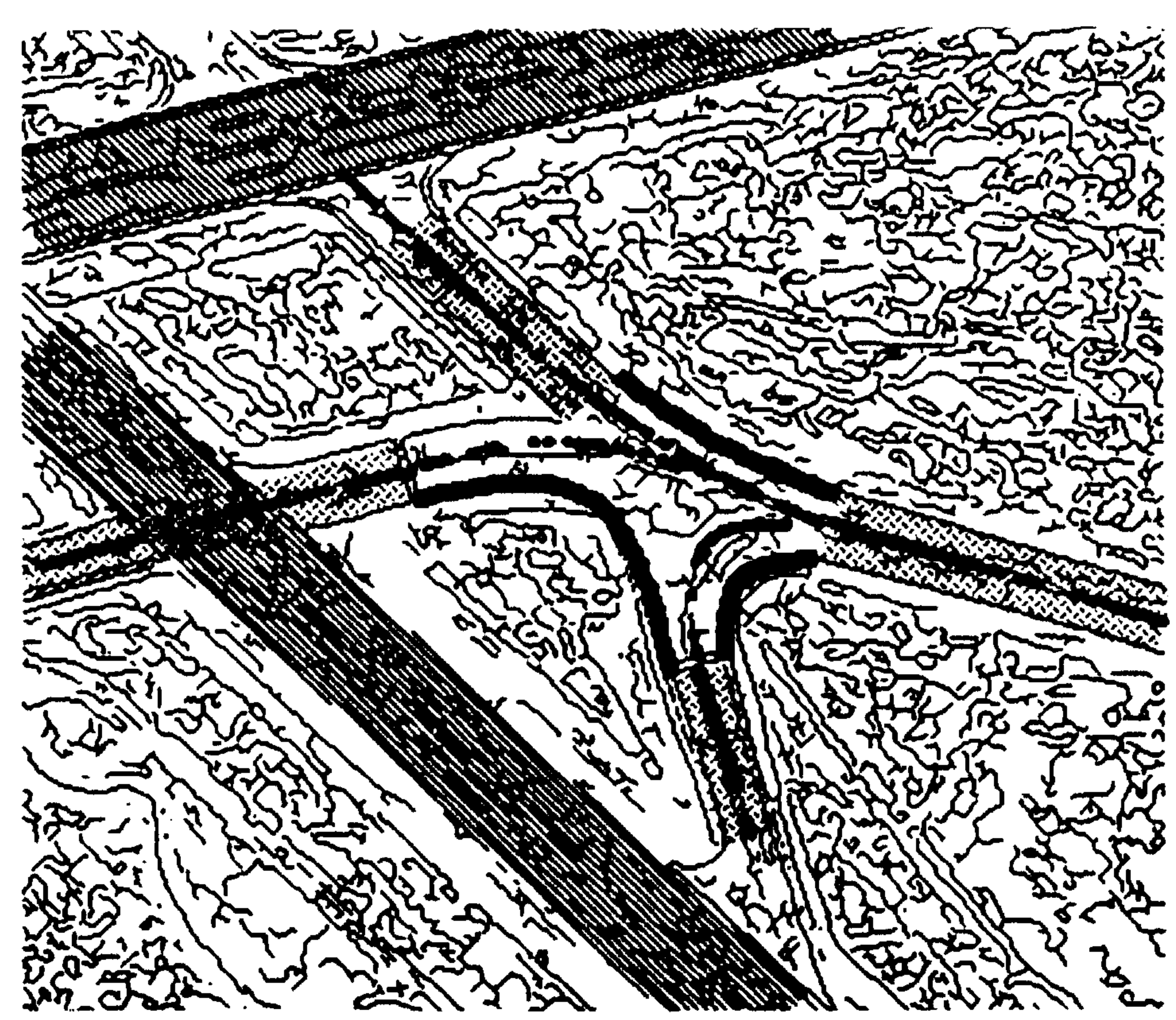

Figure 16: Underlying edges and evidence of centerline

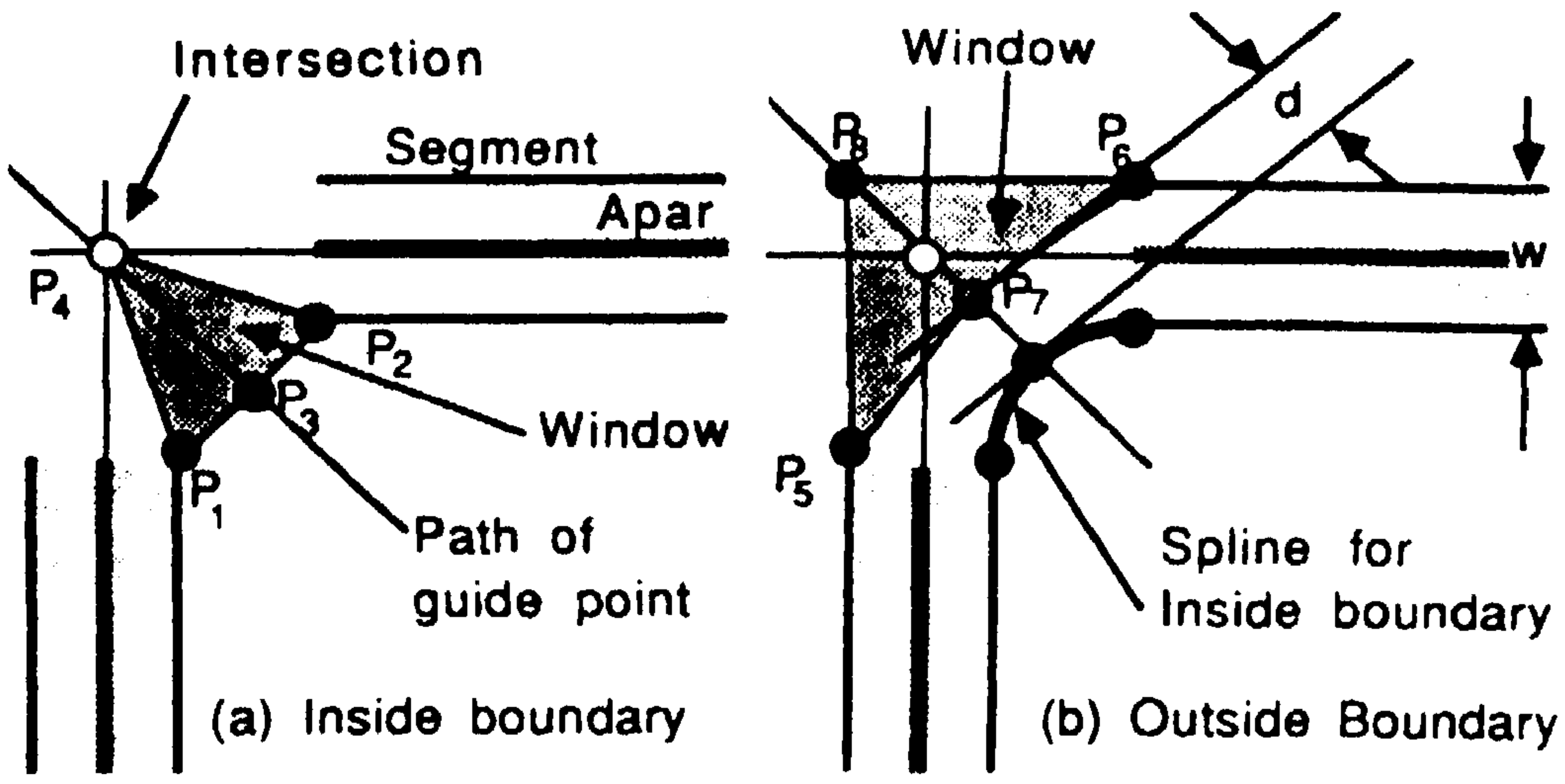


Figure 10: Search Windows for 1-junctions

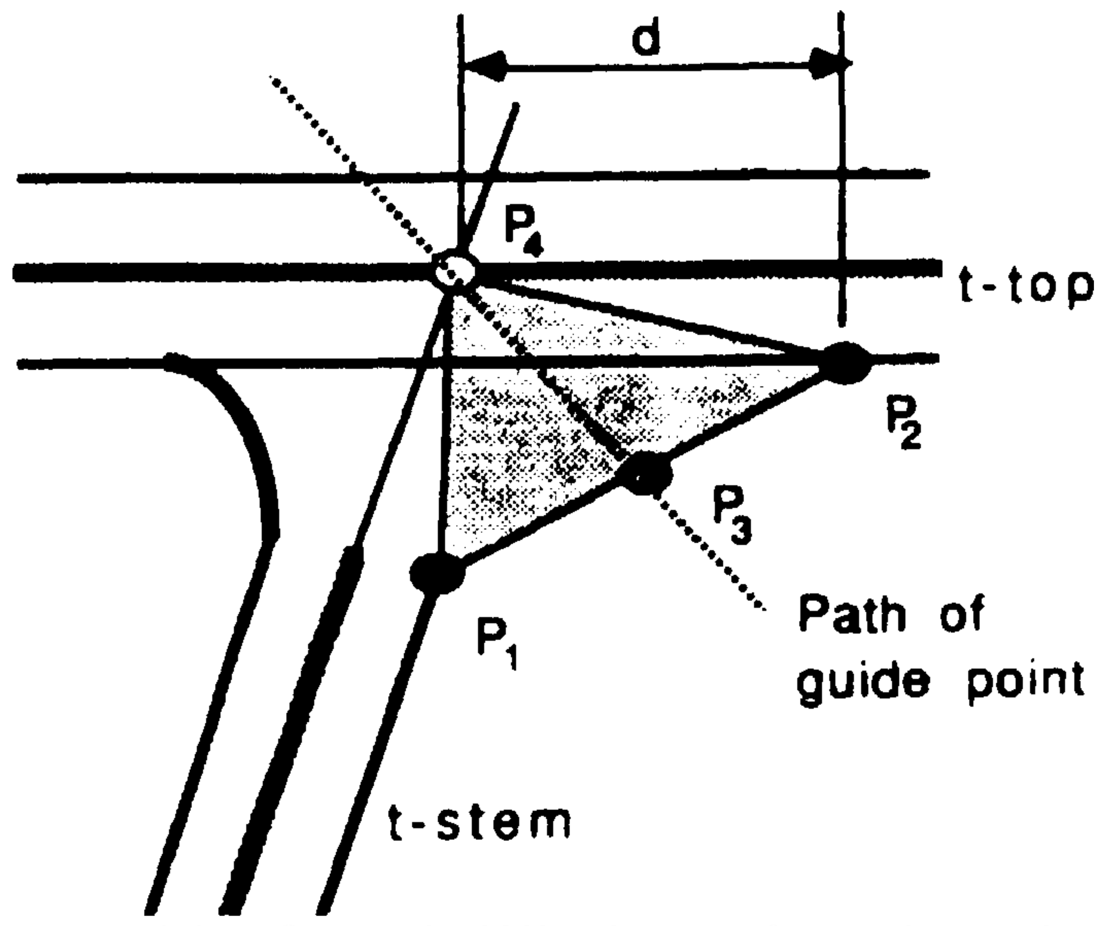


Figure 11: Search Windows for t-junctions

lapping 1-junctions. Note that there are no junctions between crossing runways; they overlap.

Figure 10 shows an 1-junction. For each pair of potential "joinable" fragments we distinguish an "inside" and an "outside" boundary. The inside boundary, if it exits, would be found on the side where we measure the smaller angle between the two elements. On the other hand, t-junctions are considered to have two "inside" boundaries. The search window for one of these boundaries is shown in figure 11.

A second classification involves the boundaries themselves. Some are curved while others are straight. The curved boundaries — in airport design — actually consist of circular or parabolic sections. However, to deal with imperfect segmentation of boundaries, we model the straight boundaries as two straight lines, and the curved boundaries by means of cubic splines. For each boundary we apply both models and then the choose the better fit (see below).

We look first for the inside boundary, and then for the outside boundary, (there is always an inside boundary. At complex intersections however, there may not be an outside boundary). If there is no evidence of an inside boundary, we do not look for an outside boundary. The method is as follows:

1. Collect context information: We use the intensity edges used above to find runways and taxiways, and the apars corresponding to verified runways and taxiways. These are shown shaded in the figures below.
2. Compute search window for inside boundary: Determine the points P1, P2, P3, and P4. For an 1-junction (figure 10, PI and P2 are located at the ends of the two elements. P3 is the middle point between PI and P2. P4 is the intersection of the apars representing the two elements. For a t-junction (figure 11) P2 is computed on the t-stem inside boundary at a distance *d* from P4. The distance *d* is proportional to the average width of the two apars.
3. Look for inside boundary: We compute a series of splines using three points: two anchors and a guide point. The anchor points are PI and P2. The guide point varies from P3 to P4. For each spline, we compute the intersection of the spline with the underlying intensity edges. The spline that returns the highest number of edges is taken as a hypotheses (possible inside boundary) if the following criteria are met:
   (a) The length of the underlying boundary (or boundary fragments) is at least one half of the length of the spline. On other words, allow 50% boundary fragmentation and/or partial spline-to-boundary fit.
   (b) The "junction" between the spline and the element boundaries (at PI and P2) is smooth (15° tolerance). That is, the tangent to the spline at the anchor points is similar to the direction of the edge.
4. Compute search window for outside boundary: Determine the points P5, P6, P7, and P8. P5 and P6 are located at the ends of the two elements. P7 is computed to be along the line joining the guide point of the inside spline and P8, at a distance *d* from the guide point, *d* is one half the average of the withds of the two apar elements. P8 is the intersection of the outside boundaries of the two elements.
5. Look for outside boundary: Similar process as for inside boundaries. The anchor points are P5 and P5, and the guide point varies from P7 to P8. As above, we compute the intersection of each the spline with the underlying intensity edges. The spline that returns the highest number of edges is taken as a hypotheses (possible outside boundary) if similar criteria are met.

### 4.2 Hypotheses Verification

As before, verification consists of finding the markings we expect. Our method looks for the centerlines associated with taxiways.

1. Resegment and collect context information: We resegment the image to include all intensity edges in the neighborhood of the junction. For context we use the apars corresponding to verified runways and taxiways, and a description of the hypothesized junction *inside* boundary (the guide point for the spline or a pair of straight lines). See figure 12.
2. Compute search window for centerline boundary: Determine the points PI, P2, P3, and P4. For an 1-junction (figure 10) PI and P2, the anchor points, are located at the ends of the two elements. P3

Figure 17: Parallel runway and taxiway

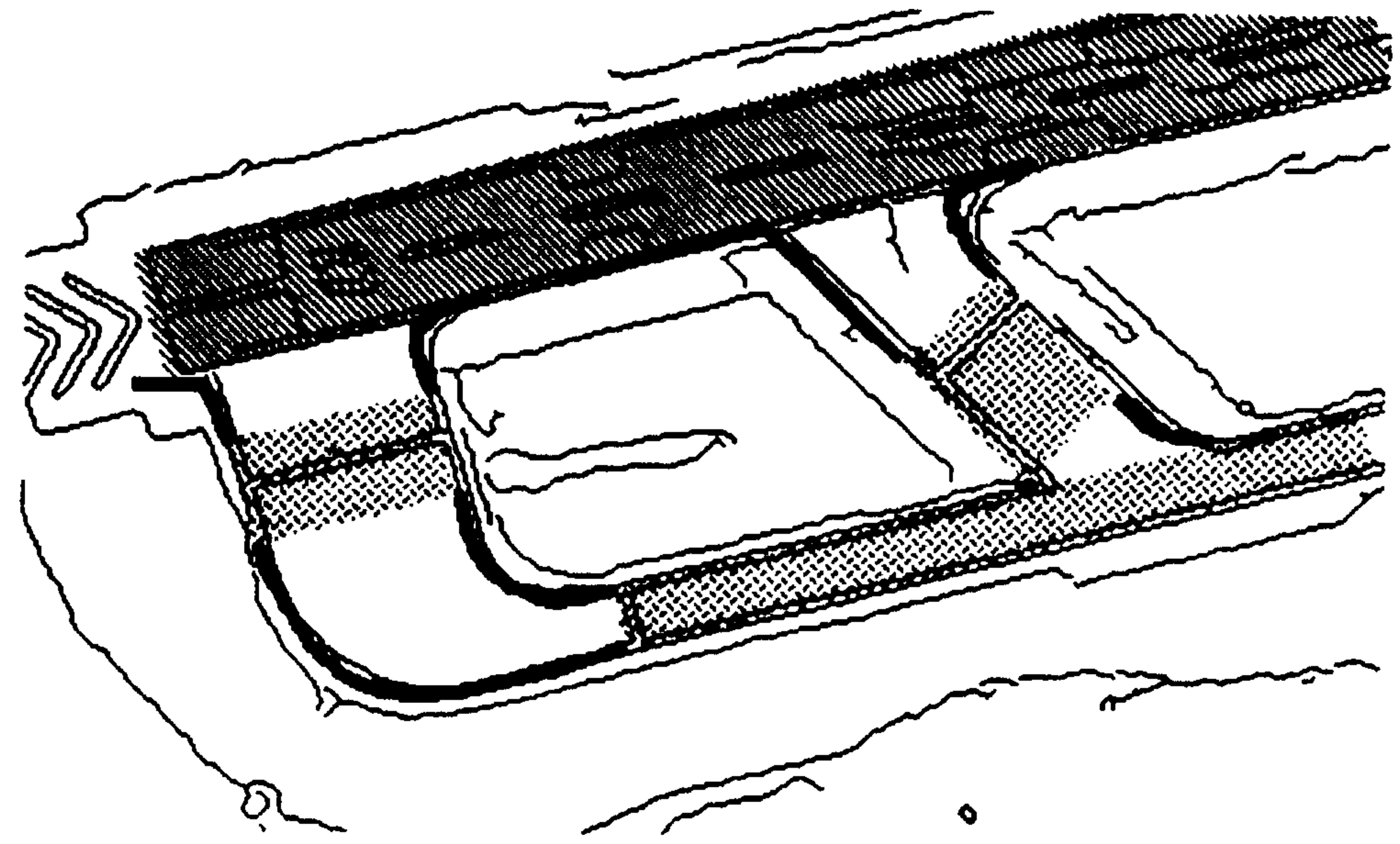

Figure 19: Boundary hypotheses

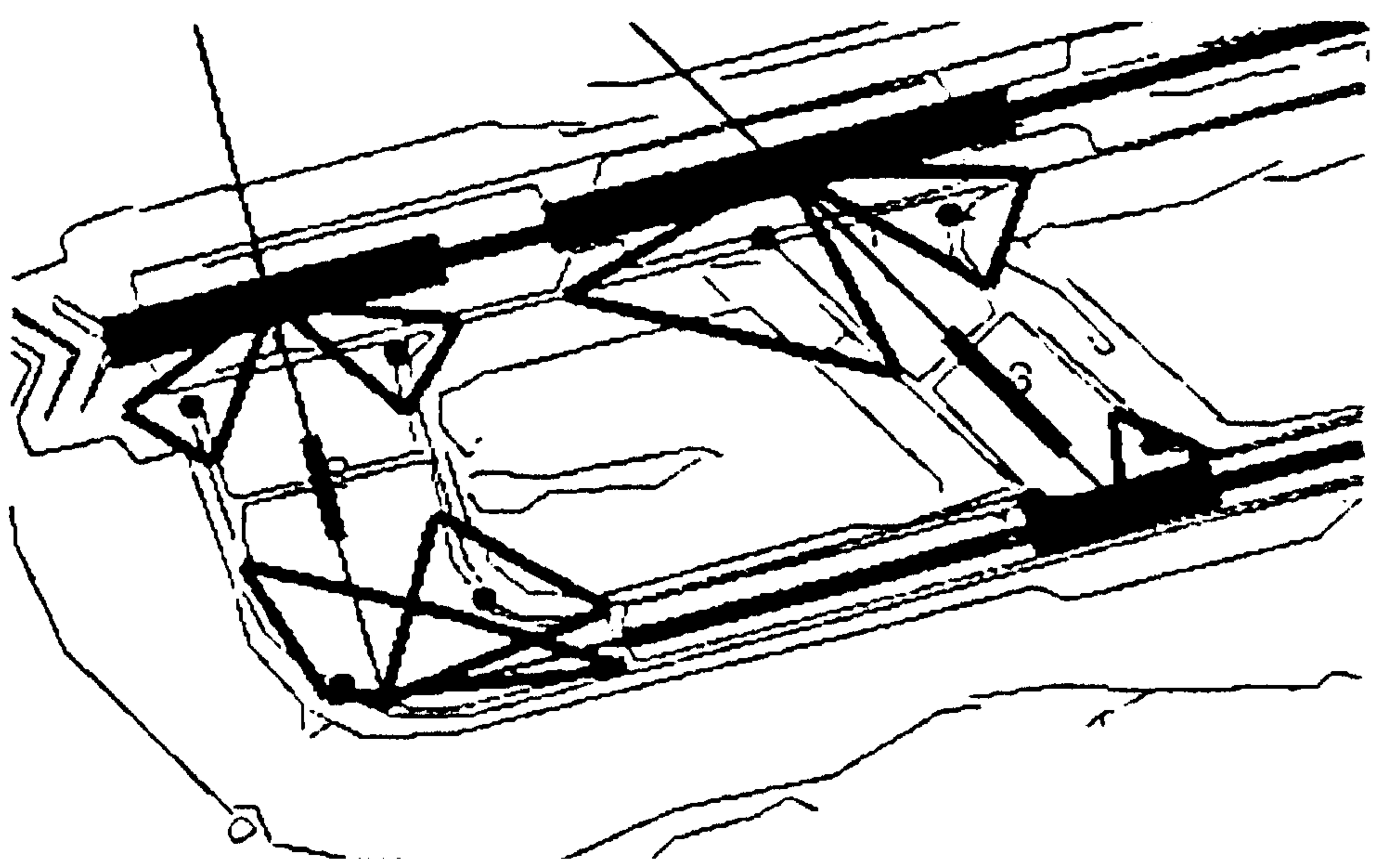

Figure 18: Search windows

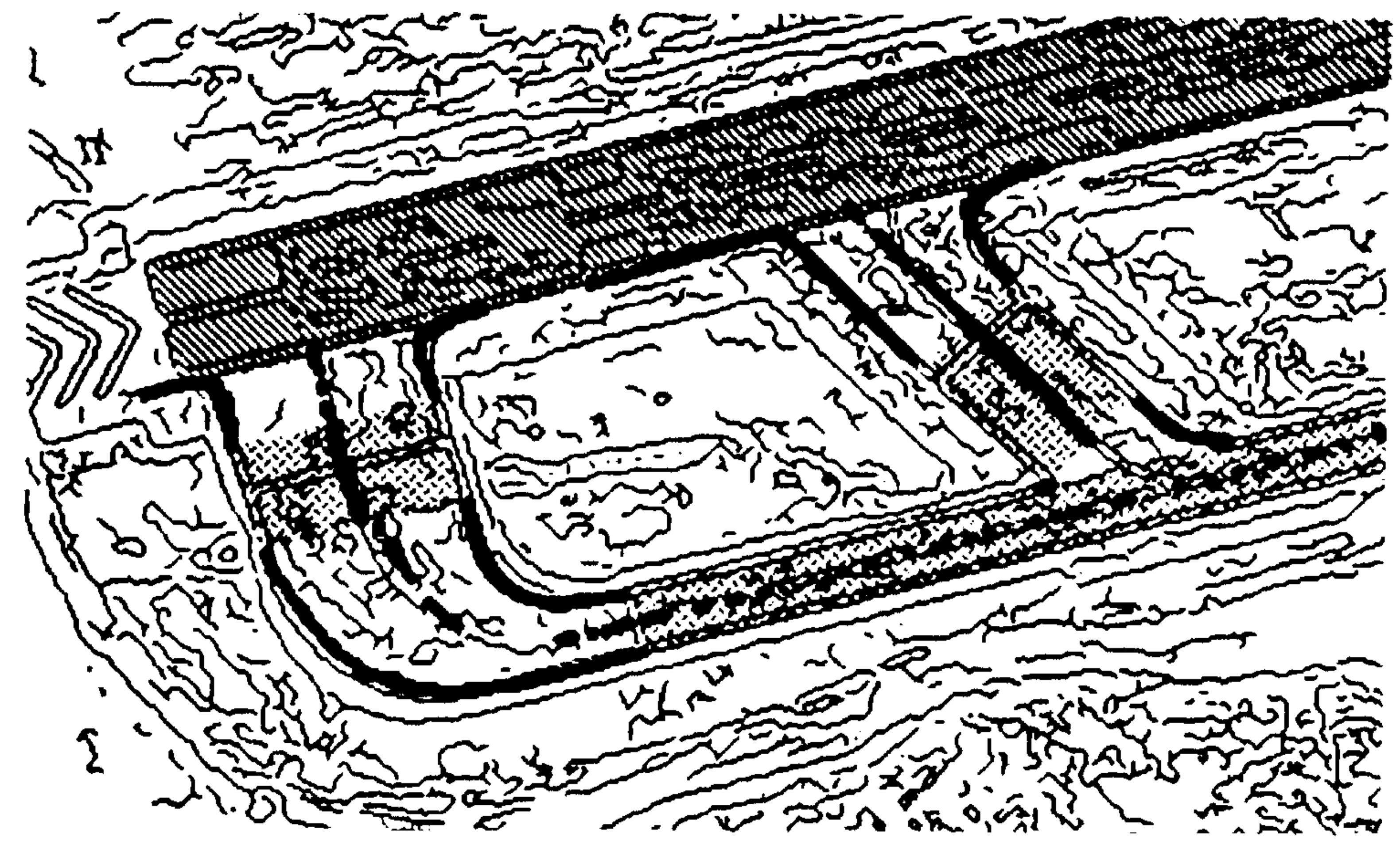

**Figure 20: Evidence of markings and valid hypotheses**

to each other. The underlying edges and the runway and apar context are shown in figure 18. We also show in this figure the extent of the search windows. The boundary hypotheses are shown in figure 19, and the detected evidence of markings in figure 20.

## 5 Conclusion

We have described our method for detecting and describing runways and taxiways in a major, commercial airport scene. We believe that this is an important application in itself. However, we hope that it has also served to illustrate how geometrical context can be used to aid in a difficult image understanding task, without requiring complete and specific *a priori* knowledge of the scene being viewed.

## References


[Ashford and Wright, 1984] N. Ashford and P.H. Wright. Airport Engineering, 2nd Ed. Wiley and Sons, 1984.

[Canny, 1986] J. Canny . A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 8, November 1986, pp. 679-698.

[FAA, 1980] Marking P aved Areas on Airports *FAA Advisory Circular AC 150/5340-1E,* November 4, 1980.

[Huertas and Nevatia, 1988] A. Huertas and R. Nevatia. Detecting Buildings in Aerial Images. *Computer Vision, Graphics and Image Processing,* Vol. 41, number 2, February 1988, pp 131-152.

[Huertas *et* a/., 1987] A. Huertas, W. Cole and R. Nevatia. Detecting Runways in Aerial Images. *Proceedings of the DARPA Image Understanding Worshop,* Los Angeles, California, February 1987.

[Mohan and Nevatia, 1988] R. Mohan and R. Nevatia. Perceptual Groupings for the Detection and Description of Structures in Aerial images *Proceedings of the DARPA Image Understanding Workshop,* Boston, Massachusetts, April 1988.

[Nevatia and Babu, 1980] R. Nevatia and R. Babu. Linear Feature Extraction and Description. *Computer Vision, Graphics and Image Processing,* Vol. 13, 1980, pp. 257-269.

[McKeown and Harvey, 1987] D.M. McKeown, and W.A. Harvey. Automatic Knowledge Acquisition for Aerial Image Interpretation. *Proceedings DARPA Image Understanding Workshop,* Vol. 1, February 1987,pp. 205-226.

[McKeown *et al,* 1985] D.M. McKeown, W.A. Harvey and J. McDermott J. Rule-Based Interpretation of Aerial Imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 7, September 1985, pp. 570-585.