A Reprint from the

PROCEEDINGS

Of SPIE-The International Society for Optical Engineering

Volume 938

Digital and Optical Shape Representation and Pattern Recognition

4–6 April 1988
Orlando, Florida

Model-based vision by cooperative processing of evidence and hypotheses using configuration spaces

Yoshinori Kuno, Katsushi Ikeuchi, Takeo Kanade
The Robotics Institute and Computer Science Department
Carnegie Mellon University, Pittsburgh, Pennsylvania 15213

Model-Based Vision by Cooperative Processing of Evidence and Hypotheses Using Configuration Spaces

Yoshinori Kuno, Katsushi Ikeuchi and Takeo Kanade

The Robotics Institute and Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

This paper presents a model-based object recognition method which combines a bottom-up evidence accumulation process and a top-down hypothesis verification process. The hypothesize-and-test paradigm is fundamental in model-based vision. However, research issues remain on how the bottom-up process gathers pieces of evidence and when the topdown process should take the lead. To accumulate pieces of evidence, we use a configuration space whose points represent a configuration of an object (ie. position and orientation of an object in an image). If a feature is found which matches a part of an object model, the configuration space is updated to reflect the possible configurations of the object. A region in the configuration space where multiple pieces of evidence from such feature-part matches overlap suggests a high probability that the object exists in an image with a configuration in that region. The cost of the bottom-up process to further accumulate evidence for localization, and that of the topdown process to recognize the object by verification, are compared by considering the size of the search region and the probability of success of verification. If the cost of the top-down process becomes lower, hypotheses are generated and their verification processes are started. The first version of the recognition program has been written and applied to the recognition of a jet airplane in synthetic aperture radar (SAR) images. In creating a model of an object, we have used a SAR simulator as a sensor model. so that we can predict those object features which are reliably detectable by the sensors. The program is being tested with simulated SAR images, and shows promising performance.

1. Introduction

Hypothesize-and-test. or a combination of bottom-up and top-down processes, is a paradigm commonly used in model-based vision[1,2,3]. The method is important for achieving efficient and robust vision. The research issues are how the bottom-up process gathers pieces of evidence to generate hypotheses and when the hypotheses should be generated for the top-down process to start. Conventional methods are rather rigid in terms of how and when to generate hypotheses. For example, the "local-feature-focus" recognition strategy proposed by Bolles. Horaud and Cain[2] requires that "focus features" be determined in advance. A hypothesis is generated if exactly those "focus features" have been detected.

We propose a model-based object recognition method which combines a bottom-up process and a top-down process efficiently by using configuration spaces. A point in a configuration space represents a configuration of an object (ie. position and orientation of an object in an image). If a feature is found which matches a part of an object model, the configuration space is updated to reflect the possible configurations of the object. A region in the configuration space where multiple pieces of evidence from such feature-part matches overlap suggests a high probability that the object exists in an image with a configuration in that region. The cost of the bottom-up process to further accumulate evidence for localization. and that of the top-down process to recognize the object by verification, are compared by considering the size of the search region and the probability of success of verification. If the cost of the top-down process is lower, hypotheses are generated and their verification processes are started.

The fundamental idea of the bottom-up process is similar to the Hough transform[4]. Thompson and Mundy[5], Hwang[6] and Sato et al.[7] used a similar approach. What is new in our method is that we consider the certainty of an object's existence and the uncertainty of the object's position and orientation by using the configuration spaces. The certainty is represented by a certainty value of each point of the region in the configuration spaces. The value is basically decided by what features support the region. The positional uncertainty is represented by the area or volume of the region. The method decides when to change the process by comparing the process costs obtained from the information in the configuration spaces. If the object's existence certainty is high and the positional uncertainty is small, the top-down cost is low and it takes the lead.

We have applied the method to synthetic aperture radar (SAR) image recognition. In designing the recognition method, we have considered a sensor model as well as object models. Ikeuchi and Kanade[8] proposed the use of sensor models to predict the detectability and reliability of each feature in order to establish a robust recognition method. In creating a model of an object, we have used a SAR simulator

as a sensor model, so that we can predict those object features which are reliably detectable by the sensors.

2. Overview of the Recognition Method

We use the aspect method[9] to deal with the appearance of 3-D objects in 2-D images. An aspect is a representative view of an object. An object model is represented as a set of aspect models which cover all possible appearances. Thus, the 3-D object recognition problem becomes a problem of finding a 2-D object in an image which matches one of the aspect models. Section 6 will show why this is sufficient for SAR image recognition.

Object recognition can be considered as finding the configuration (i.e. position and orientation in an image) of an object. An important issue in a recognition method is how to maintain information about what configurations an object can take. We use configuration spaces for this purpose. A point in a configuration space represents a configuration of an object or a part of an object. Before an object recognition process starts, an object may be in any configuration, and the probability of any given configuration is very low. If a feature is found which matches a part of the object, it constrains the possible configurations of the object, thus giving the possible configurations a higher probability. We consider a subspace of these high probability configurations as evidence which support the existence of the object. If another piece of evidence is found, and if both subspaces overlap in the configuration space, the overlapped region forms a new subspace, where the number of the possible configurations is smaller and the probability is higher. This process shows that the accumulation of evidence decreases the uncertainty of the configuration and increases the probability of the existence of the object. In the configuration space, the information is maintained by a thick "cloud" where several thin clouds overlap. This means that the subspaces of evidence overlap to form a smaller subspace with a higher probability.

Figure 1 shows an overview of the method. An object model is composed of parts. Each part corresponds to a primitive feature such as a point or a line segment which can be extracted in images. Here, a point means a round blob feature and a line segment is a elongated blob feature with narrow width. Each part has some attributes such as area, intensity, and ratio of the maximum to the minimum inertia[9]. To each part, two configuration spaces are attached to represent its positional relation to the whole object.

One configuration space is a model part configuration space ("model PCS"). This configuration space shows the possible positions and orientations of each part with respect to the position and orientation of the whole object. The other configuration space for each part is a model object configuration space ("model OCS"). It holds the reverse relation of the model PCS; that is, it gives the possible positions and orientations of the whole object viewed from the location of the part.

Several feature extractors search in an image for features. When a feature is matched against a part in the model base using some attributes, the match is considered as a piece of evidence supporting the existence of the object which has the part. This evidence is accumulated in an evidence accumulation object configuration space ("evidence accumulation OCS") for the object by establishing a cloud which shows the possible configurations of the object. The cloud can be obtained from the model OCS of the part.

If there happens to be a cloud in an evidence accumulation OCS where supports overlap from many feature-part matches, the cloud represents a higher probability that the object exists in those configurations. The more pieces of evidence that support the cloud, the higher is the probability and the smaller is the size of the cloud. The latter means that the positional uncertainty of the object is reduced by accumulating evidence.

If the probability is high and the size is small, it might be more efficient to use a top-down process instead of the bottom-up or evidence accumulation process. This is because the high probability assures a high success rate in verifying the hypotheses, and the small size means that a top-down process needs to search only in a small area. The method decides when to change the process by comparing the cost to continue the bottom-up process until the object is detected and the cost to detect the object by shifting to the top-down process.

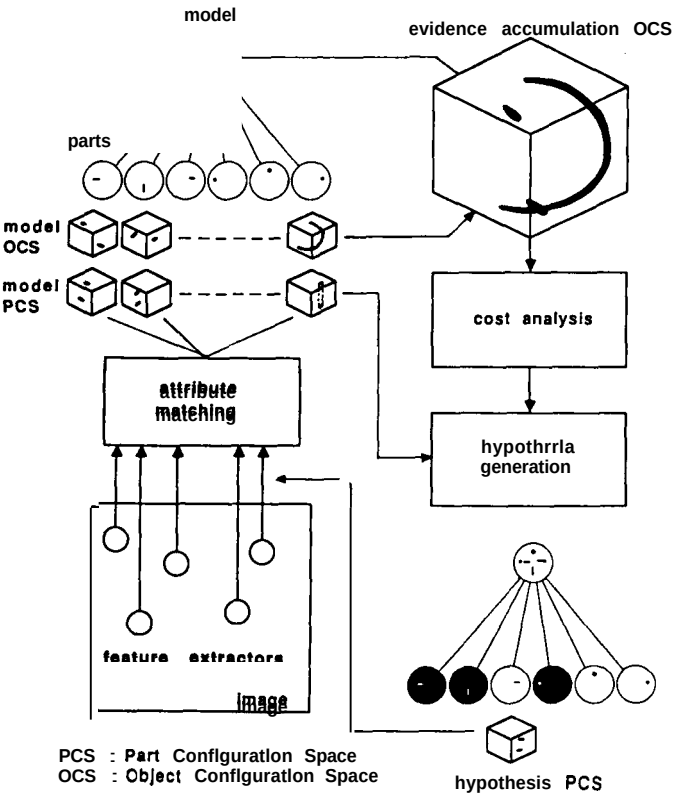


Figure 1: Overview of the recognition method.

If the cost of the top-down process is lower, hypotheses for the object and undetected parts are generated. The positions and orientations which each part can take are computed from the cloud in the evidence accumulation OCS and a model PCS of each part. These configurations are stored in a hypothesis part configuration space ("hypothesis PCS") of each undetected part hypothesis.

Each hypothesis directs the feature extractors to examine the region specified in the hypothesis PCS. If enough of the part hypotheses have been verified, the object is verified.

Though a feature matches one part in the model base in the above explanation, a feature may match multiple parts in real applications. The basic assumption is that the method can get a cloud with a high probability corresponding to a correct object even if each feature makes some mismatches. We can make a network connecting similar parts which may match the same type of feature. In this case, all the connected parts match a feature simultaneously when it is found by the feature extractors.

3. Recognition and Configuration Spaces

The method uses four kinds of configuration spaces:

1. Model part configuration spaces ("model PCS")
2. Model object configuration spaces ("model OCS")
3. Evidence accumulation object configuration spaces ("evidence accumulation OCS")
4. Hypothesis part configuration spaces ("hypothesis PCS")

The first two configuration spaces are stored with the model. The evidence accumulation OCS's are used in the bottom-up process. The hypothesis PCS's are created when hypotheses are generated. The OCS's show possible configurations of an object, while the PCS's show those of a part.

This section describes the information that each configuration space maintains by using a simple example. Figure 2 shows an example of an object consisting of three points and three line segments. We define a coordinate system to describe the position and orientation of each object or part. A coordinate system is represented by the position of the origin and the direction of the x-axis.

Model PCS

This configuration space shows the possible positions and orientations of each part with respect to the coordinate system for the whole object. Figure 3 (a) shows the configuration space for the line-like part P1 in Figure 2. There are two regions or clouds in the space because the orientation of a line segment has two possible values with a difference of π. Figure 3 (b) shows the configuration space for the point P6. Since a

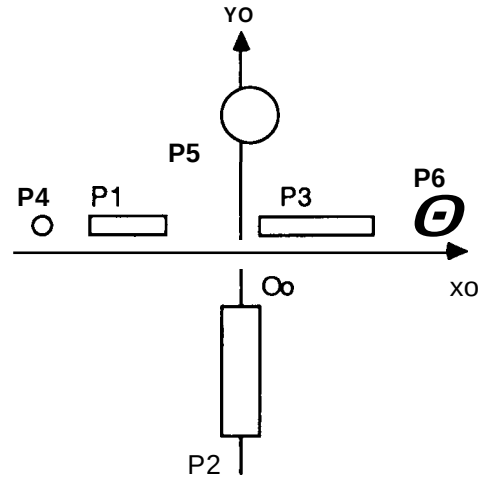
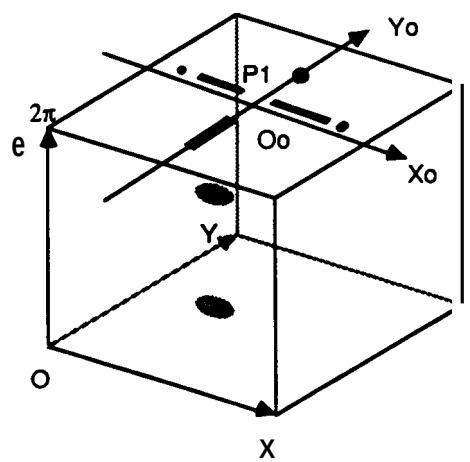
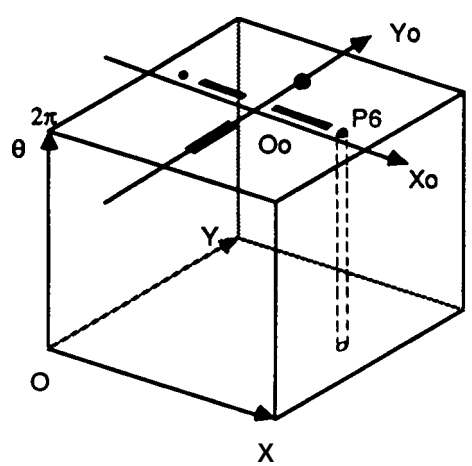


Figure 2: Object model.



(a) Line part.



(b) Pointpart.

Figure 3: Model part configuration spaces.

point has no direction attribute, we can ignore the θ dimension or consider that the region is like a cylinder in Figure 3 (b).

A cloud describes the position and orientation constraints in configuration so that we can allow some position and orientation tolerance for each part in the model as well as measurement uncertainty in the feature extraction. The size and shape of each cloud depend on the amount of tolerance and uncertainty.

Model OCS

This configuration space holds the reverse relation of the model PCS. It shows possible regions in which the object can exist when the part is put at the origin with zero degrees of rotation. Figure 4 (a) is a model OCS of the line segment P1 in Figure 2. There are two clouds in the configuration space corresponding to the two possibilities for the object's position and orientation. These are indicated by the two object coordinate systems, $X1-Y1$ and $X2-Y2$, in Figure 4 (a).

The shape of a cloud for a point feature is a little complicated. Figure 4 (b) shows an example for P6. If a point feature matches a point part. the position of the object should be somewhere on a circle as shown on the top face of the box. The object orientation is also constrained. If the object is located at a certain point on the circle indicated by the end of an arrow, the object's orientation must be in the direction of the arrow. This relation is represented as a spiral in the three-dimensional configuration space as shown in Figure 4 @).

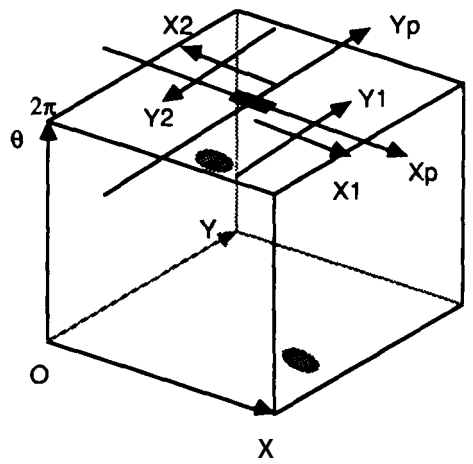
Evidence accumulation OCS

This configuration space is used to accumulate pieces of evidence for each object

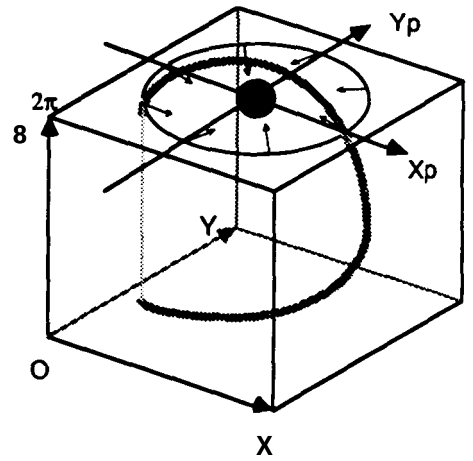
If a feature is extracted which matches a part of an object, the model OCS for the part is translated and rotated so that the position of the origin and the direction of the x-axis are equal to those of the extracted feature. Next, the clouds in the model OCS are translated along the 8 direction by the value of the direction of the feature. Then. they are convolved with the uncertainty of the measurement of the position and orientation of the detected feature. The resultant clouds are written in the evidence accumulation OCS for the object. Figure 5 shows the clouds after detecting two features which match P1 and P6. The region where the two clouds overlap indicates a high probability of the existence of the object in this region. We will discuss the details in the next section.

Hypothesis PCS

When an object's existence is hypothesized, the hypothesis PCS is created for each undetected part to specify the positions and orientations where the parts should exist, in other words, the search area in the top-down process. This space can be obtained by convolving the model PCS of the part with the clouds in the evidence accumulation OCS. Figure 6 shows an



(a) Line part.



(b) Pointpart.

Figure 4: Model object configuration spaces.

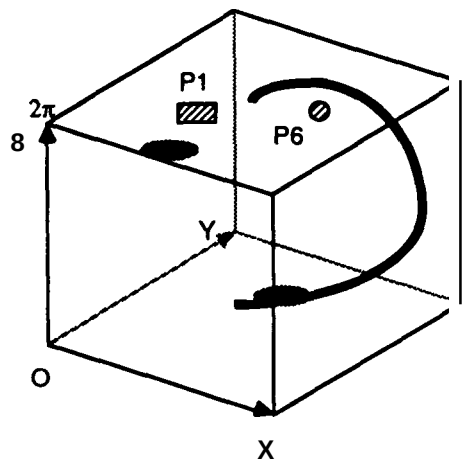


Figure 5: Evidence accumulation object configuration space.

example for **P3.** The existence **of** the object is hypothesized after detecting three feature-part matches indicated by the regions filled with diagonal lines. The possible positions and orientations of the line segment part **P3** form two ellipsoid clouds in Figure 6.

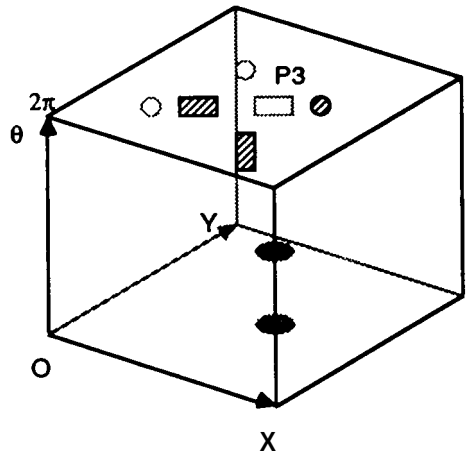


Figure 6: Hypothesis part configuration space.

4. Evidence Accumulation into the Configuration Space

The evidence accumulation process is performed by using an evidence accumulation OCS. We will explain the method by considering a 1-D example. Figure 7 shows an object model consisting of four points. The position of the object is represented by the position of the origin O,. Assume that each part has the same positional tolerance of $2r_a$.

A cloud in an evidence accumulation OCS has two attributes. One is its shape and size, which determine the possible configurations of the object. The shape and size can be obtained by convolving the tolerance of the position and orientation of a part with the measurement uncertainty of an extracted feature. The other is the value of each point in the cloud, which is related to the certainty that the object exists at the position and orientation specified by the point. The certainty depends on the combination of the parts supporting the cloud, the measurement uncertainty in the feature extraction, and the amount of noise in the image. Thus, the cloud can be considered as a certainty density distribution in the evidence accumulation OCS.

In this 1-D case, a cloud is represented by a function of x. If a feature F1 matching P1 is detected, the certainty of the existence of the object has a distribution indicated by the solid line in Figure 8 (a), considering measurement uncertainty of the feature in addition to the positional tolerance of the part. Then, F2 is detected as P2. Another distribution is obtained as the dashed line in Figure 8 (a). Since P2 is assumed to be a part which is important for distinguishing this object from other objects. its detection indicates a higher certainty of the object's existence. Then, **F3** is detected as **P3,** and another distribution is obtained, shown by the dotted line. After this evidence has been accumulated, the three clouds, when summed, can be considered to form a cloud showing a total certainty density distribution as in Figure 8 (b).

If we establish a region of length $2r_k$ as shown in Figure 8, the area of the hatched region shows the certainty that the object exists in the region. If the top-down process starts at this moment, and a hypothesis for the undetected part **P4** is

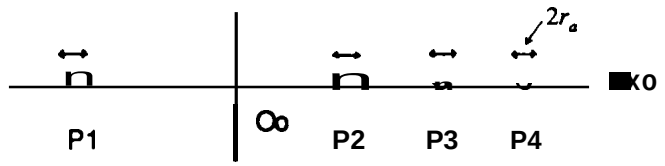


Figure 7: Object model in 1-D case.

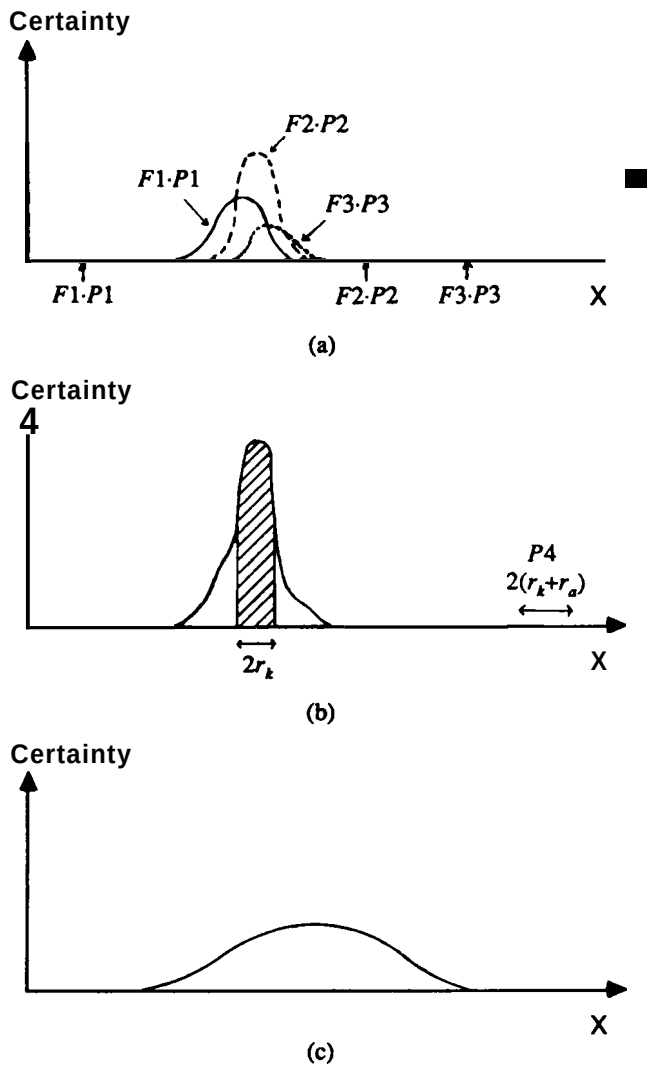


Figure 8: Certainty density distributions in the evidence accumulation object configuration space.

generated, the process must examine an area of $2(r_k+r_a)$ to verify the hypothesis, because we assume that the object exists somewhere in the $2r_k$ region, and each part has a positional tolerance $2r_a$.

The probability that the top-down process succeeds in finding the object within the region is proportional to the certainty of the object's existence accumulated over the region in the configuration space. The examination cost of the top-down process to verify an object hypothesis is related to $2r_k$. Suppose we get a dismbution as in Figure 8 (c). As we set a larger r_k, the probability of success becomes larger. However, at the same time, the cost of the top-down process becomes higher, because the top-down process must examine a larger area. On the other hand, if we set a smaller r_k, the probability of success is lower, though the search cost is smaller. A narrow dismbuaon in which we can take a small r_k yielding a large area (a high success probability), like in the case of Figure 8 (b), is necessary to initiate the top-down process. Note that r_k is not a parameter which needs to be determined in advance. This will be explained in the next section.

5. Comparison of the Process Costs

A combination of a bottom-up (data-driven) process and a top-down (model-driven) process is important to make object recognition efficient and robust. The question is when to switch from the bottom-up process to the top-down process. Suppose k pieces of evidence $e_1, e_2, \ldots, e_k$ have been found for an object which consists of m parts. The decision can be made by comparing the cost to continue the bottom-up process until the target object is found with the cost to detect the object by the top-down verification process. We make a simple analysis to estimate the costs to get a general idea of the appropriate time to switch. We can modify the results of the analysis through experiments when we apply the recognition method to a real application.

In the bottom-up process the feature extractors have to search A/m pixels on average to find a feature matching a part of the target object, where A is the total number of pixels in an image. During that period they might find f_d features including the necessary feature, each of which has the cost of finding a matching part. The cost of completing the recognition by the bottom-up process, that is, the cost to detect $m-k$ features by the process is

$$E_{bk} = e_b f_d (m-k)\frac{A}{m},$$

where e_b is the average cost of examining a pixel by the bottom-up process when only the necessary features are in the image.

Next we consider the cost of the top-down process. Though the cloud in the configuration space is three-dimensional, let us consider only the extent of the x and y dimensions of the cloud, because we assume that the cost depends only on the search area. In other words, if a feature exists at some point, the process can find it without regard to its orientation. To make the matter simple, we assume that the cloud is a circle of radius r_k. We also assume that the positional tolerance for each part is the same for all the parts and is described by a circle of radius r_a. Finally, we assume that the cost to examine a pixel, e_t, is the same for all the parts. The cost to examine $m-k$ hypotheses is

$$E_{tk} = e_t(m-k)\pi(r_k+r_a)^2.$$

When the above cost is paid, the probability of success is

$$C = \int_{\pi(r_k)} c(obj=x,y|e_1 e_2 \ldots e_k)\,dx\,dy.$$

This is the certainty of the existence of the object when the set of features *are* observed in an image. It corresponds to the hatched area in Figure 8 (b) in the 1-D case. Here we consider this certainty as the probability of the object's existence. Therefore, the expected cost of the top-down process until the process succeeds is

$$E_{tk} = \sum_{n=1}^{\infty} n E_{tk1} C(1-C)^{n-1}$$
$$= E_{tk1}/C.$$

In a real application we can terminate the search in the middle of the process if some of the hypotheses have been refuted. Thus the real cost of the topdown process is less than the value obtained by the above equation.

We use the ratio of E_t to E_{bk} to decide when to change the process. The ratio is

$$S = \frac{E_{tk}}{E_{bk}} = \frac{e_t}{e_b f_d}\frac{\pi(r_k+r_a)^2}{A/m}\frac{1}{C}.$$

If S is less than 1, it is better to use the top-down process. The first term of the equation shows the ratio of the costs of examining a pixel by the top-down and the bottom-up processes. The second is the ratio of the areas to be searched. For example, suppose $e_t=e_b$, $f_d=1.2$, $m=10$, $A=64^2$ and $r_a=4$, then C should be over 0.52 to start the top-down process when r_k is 5 and over 0.31 when r_k is 3.

Note that r_k and C are not independent and that r_k is not a predetermined value. As explained in Section 4, C can be calculated for any given r_k. Therefore. to be precise. the cost comparison says that the topdown process should start if there is a subspace of radius r_k in the cloud which makes S smaller than 1. However, this requires the evaluation of S for various r_k values. Therefore, in the implementation described in Section 6, a predetermined threshold is used to extract such a subspace. In other words. points whose certainty value is greater than the threshold are extracted. S is computed from the size of the subspace and the certainty over the subspace.

In reality. the value of C is difficult to obtain. It depends on

the uniqueness of the set of features. It has a high value if the set of features are very specific to the object among the objects which are expected to appear in an image. The value also depends on the noise in an image, which is represented by f_d. If an image is noisy, the probability decreases. especially for a simple combination of features, because such a simple combination of features might often happen by accident due to noise. We have to consider the above matters when we use the method for a real application.

6. Application: Object Recognition in SAR images

6.1. SAR sensor and object models

We apply the method which has been described to object recognition in SAR images. Ikeuchi and Kanade[8] indicated the importance of using sensor models in addition to object models to create an efficient and robust recognition method by considering what kind of features can be reliably detected. In this application, we use a SAR simulator developed at The Analytic Sciences Corporation (TASC)[10] as a SAR sensor model.

We deal with images of objects on the ground taken at the same elevation boresight ϕ as shown in Figure 9. We do not need to consider scale change in the SAR images. This makes the variation in the objects' appearance only dependent on the orientation θ of the objects with respect to the sensor direction. Figure 10 shows SAR images of a jet airplane with four orientations. The appearance changes according to the orientation. The bright, point-shaped regions and line-shaped regions are the primitive features which we use for recognition in the SAR images.

At the moment, instead of grouping appearances into aspects by considering their similarity, we just divide the orientation into equal intervals and prepare a model for each orientation. Each orientation model consists of the parts described by a point or a line feature with some attributes such as area, intensity, direction and ratio of the maximum to the minimum inertia. Those parts and their attributes are obtained by the SAR simulator. The intensity atmbute is related to the

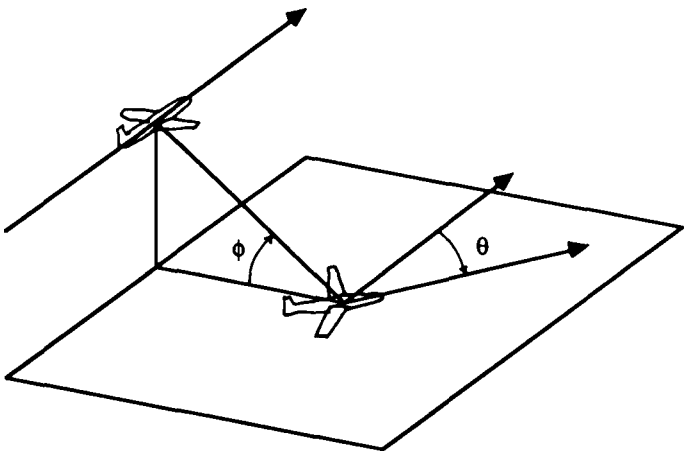
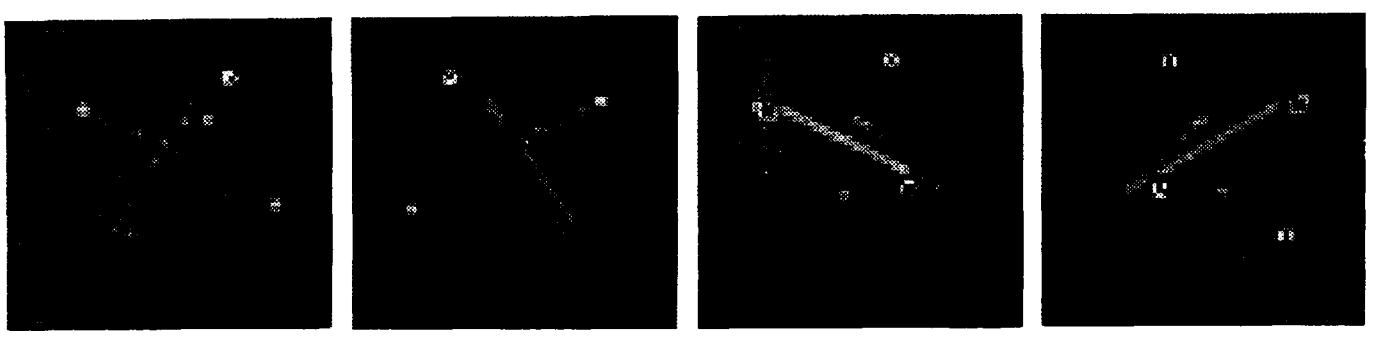


Figure 9: SAR image acquisition setting.

reliability in the feature extraction step. The parts with high intensity can be reliably detected even in a noisy image. Therefore, the feature extractors first search for features with high intensity which might match the parts with high intensity in the bottom-up process. If an object hypothesis is generated from those features, the feature extractors try to find lower intensity features in the top-down process. This decreases the recognition cost and increases the robustness, which are the merits of using the sensor model.

The model PCS and the model OCS for each part are generated from the SAR simulator. The size and shape of the clouds in the configuration spaces depend on the positional tolerance of each aspect and the measurement uncertainty in the bottom-up process. The latter is estimated by using the intensity and the shape of each part. If the intensity is high, the uncertainty is small, because such a feature can be detected reliably with less positional error. If the shape is line-like, the uncertainty is large along the line direction and small in the direction perpendicular to it, because end positions of a line usually have some error. The size and shape of the uncertainty are empirically determined for the moment.

Though the configuration spaces used here are three-dimensional, they can be treated as two-dimensional spaces for



(a) $\theta = -30°$. (b) $\theta = 30°$. (c) $\theta = -120°$. (d) $\theta = 120°$.

Figure 10: SAR images with different orientations.

each orientation model. This means that each model uses one θ-plane in the three-dimensional spaces. Since the appearance depends only on 8, and since models are given for each orientation, only the translation of each model needs to be considered in recognition. In other words, a whole set of the orientation models for an object can be considered to form a set of three-dimensional configuration spaces.

6.2. Experimental results

We have implemented the object recognition method using Knowledge Craft (Carnegie Group Inc.)[11]. The models are described by schemata (frames) and CRL-OPS (upwardly compatible with OPS5[12]) is used to control the recognition process. The models arc completely separated from the procedure. If a new model of an object is created according to the specified format, the method can be applied to the new object.

Two binary images are generated by thresholding a SAR image at high and low thresholds. Regions in the binary images are labeled, and the attributes of the regions are calculated. Schemata of the regions are created and put into working memory in CRLOPS'. The regions obtained at the high threshold are used as the reliable, high priority features in the bottom-up process. The other regions are used mainly in the topdown process. If a hypothesis cannot find a matched feature in working memory, the hypothesis creates a binary template of itself and examines the match of the template around the hypothesized location in the binary images. If it gets a high matching score. the hypothesis can be verified.

We performed experiments using images generated by the SAR simulator. Figure 11 shows an example which was synthesized by taking the maximum value at each point of two images for a jet airplane with orientation −30° and the same jet airplane with orientation 120° created by the SAR simulator. We call them jetplane-30 and jetplane+120 here. This image is synthesized for this hypothetical experiment and is not a real SAR image. For instance, the effect due to interaction between the two objects is not considered. Figures 12 (a) and (b) show extracted features obtained at the high and low thresholds. respectively.

Figure 13 shows a recognition result. Figures 13 (a) and (c) show the models of jetplane-30 and jetplane+120, respectively. They also display the status of the current bottom-up process. A symbol such as F15 under the circles indicating the parts

'We have used rule-based programming for rapid prototyping of the control mechanism of the method. The recognition process is determined by several cles in this program. In our implementation, all the features are extracted and stored in working memory in advance except features detected by template matching in the top-down process. The boaom-up process is represented by a rule; if a feature matches apart. then write possible configurations in the evidence accumulation OCS. The rule corresponding to the top-down process is that if there is a feature which satisfies a hypothesis. then the hypothesis is verified. The use of rule-based programming in this way has made it easier to test the recognition method.

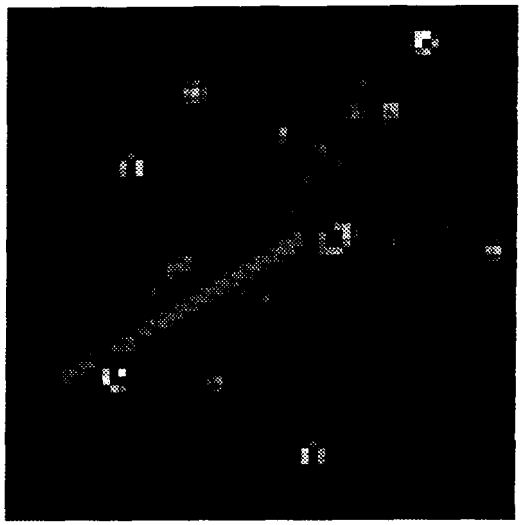
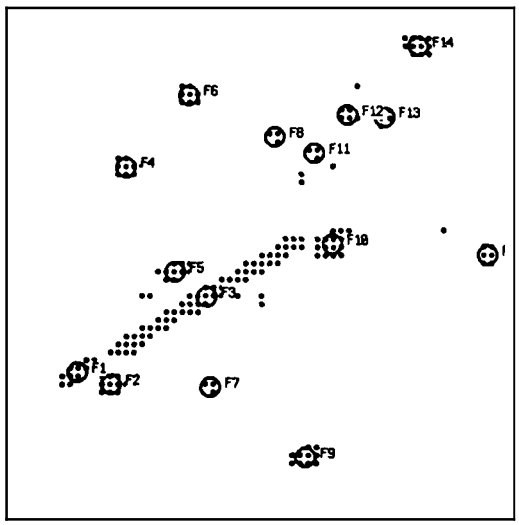
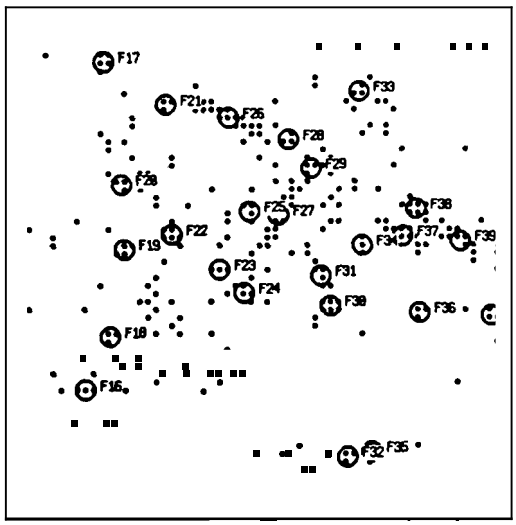


Figure 11: Synthesized SAR image used in the experiment.



(a) Features extracted at high threshold.



(b) Features extracted at low threshold.

Figure 12: Feature extraction result.

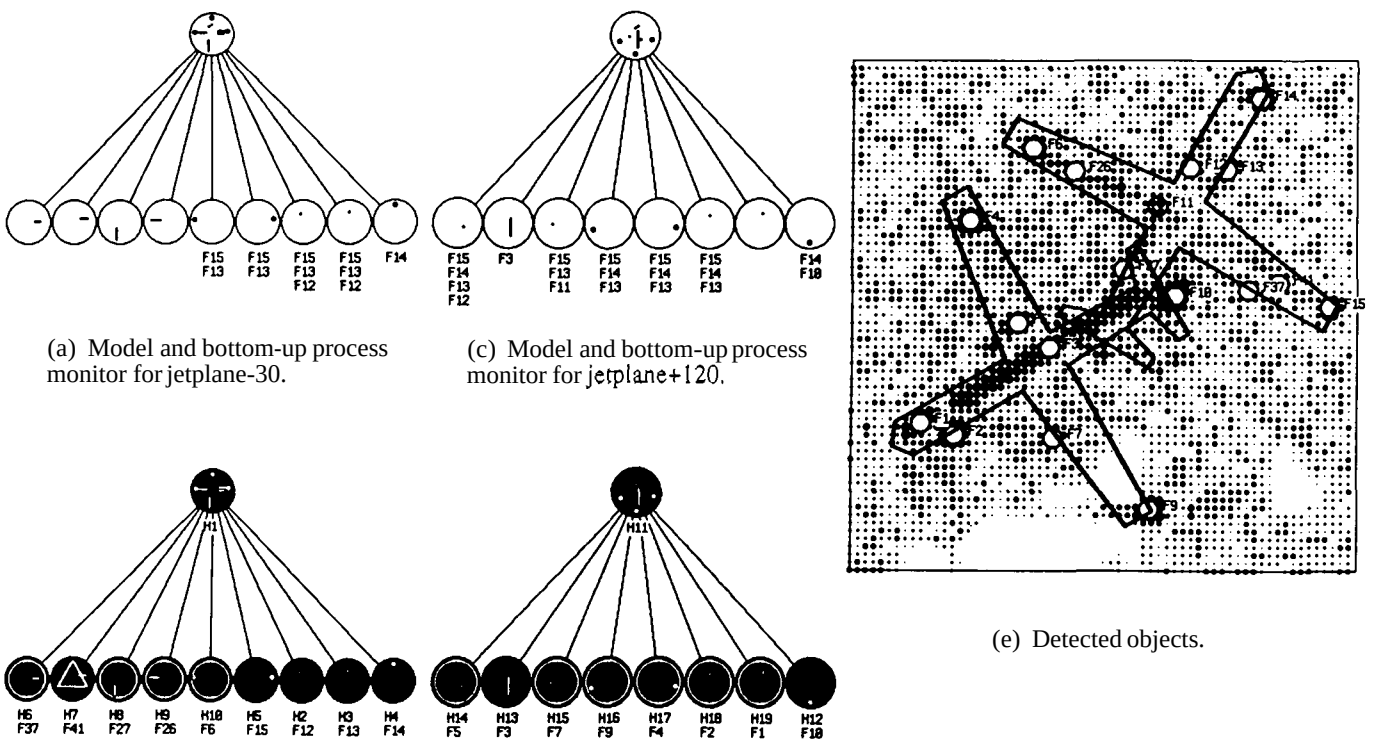
shows that the feature F15 matches the above part and that the evidence is accumulated in the evidence accumulation OCS. A feature may match multiple parts. In this example, F15 matches nine parts. However, the feature-part matches for a correct object form a small cloud with a high certainty of the existence of the object in the evidence accumulation OCS.

Figure 13 (b) and (d) show the status of the topdown process. If the cost of the top-down process computed from the information of the accumulated cloud is lower than that of the bottom-up process, the object's existence is hypothesized. The simple dark circles for the parts show that these feature-part matches formed small dense clouds indicating a lower top-down cost and that hypotheses are generated based on them. Four feature-part matches are used to generate the hypothesis for jetplane-30. while only two are required for jetplane+120.

In this implementation, the certainty of each point in the evidence accumulation OCS is decided only by the combination of the parts supporting the point. Higher weight is given to parts or combinations of parts specific to a model. Other parts have a constant weight. This information is contained in the models. The certainty is calculated by adding those weights. In this example, a higher weight is assigned to the fuselage part (the line part) of jetplane+120, which matches F3. That is why jetplane+120 needs only two feature-part matches to generate the hypothesis. In a real application, the certainty value and the cost comparison method can be adjusted through experiments.

The dark circles with a white ring indicate that a hypothesis for an undetected part, such as H19 in Figure 13 (d), is verified by feature F1 in the working memory. The dark circle with a mangle means that the hypothesis is verified by using template matching. In this example the feature which should match H7 is divided into F38 and F39 in Figure 12. Therefore, there is no feature satisfying H7 in working memory. Template matching finds F41 combining the two features. Figure 13 (e) shows the feature positions used in the recognition process. The features used to generate the hypotheses and to verify them are marked by white circles.

Since all the hypotheses for the undetected parts have been verified, the existence of jetplane-30 and that of jetplane+120 are verified. The rough outlines of the detected objects are superimposed in Figure 13 (e). The verification condition can be loosened if necessary. such that the existence can be verified even if some parts have been refuted.



(a) Model and bottom-up process monitor for jetplane-30.

(c) Model and bottom-up process monitor for jetplane+120.

(e) Detected objects.

(b) Topdown process monitor for jetplane-30.

(d) Topdown process monitor for jetplane+120.

Figure 13: Result of SAR object recognition.

The method has been tested with several simulated images with one object and with images with two objects synthesized from the images with one object. It shows promising performance.

7. Conclusion

We have proposed a basic framework of a model-based object recognition method. It combines a bottom-up process and a top-down process by using configuration spaces as a unified tool to represent object and part relations and to accumulate pieces of evidence from images. It decides when to change the process by comparing the costs of the two processes. We applied the method to SAR image recognition. A SAR simulator is used as a sensor model to determine the reliable features.

Acknowledgement

The authors thank Robert W. Pinto and G. Stephen Zabele of The Analytic Sciences Corporation for offering the SAR simulator and giving valuable comments on the properties of SAR images.

John Krumm proofread earlier drafts of this paper and provided many useful comments which have improved the readability of this paper. The authors also thank the members of VASC (Vision and Autonomous System Center) of Carnegie Mellon University for their valuable comments and discussions.

This research was sponsored by the Defense Advanced Research Projects Agency. DOD. through ARPA Order No. 4976, and monitored by the Air Force Avionics Laboratory under contract F33615-84-K-1520 and The Analytic Sciences Corporation subcontract 87123 modification No. 1. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied. of the Defense Advanced Research Projects Agency or of the U.S. Government.

References

1. Goad, C., "Special purpose automatic programming for 3D model-based vision". *Proc. of DARPA Image Understanding Workshop,* DARPA. 1983.

2. Bolles. R. C. and Horaud, P., ''3DPO: A three-dimensional part orientation system", in *Three-Dimensional Machine Vision,* Kanade. T.. ed.. Kluwer, Boston MA, 1987.

3. Grimson, W. E. L. and Lozano-Perez, T., ''Model-based recognition and localization from sparse range or tactile data", The *International Journal of Robotics Research,* Vol. 3, No. 3, 1984.

4. Ballard, D., "Generalizing the Hough transform to detecting arbitrary shapes", *Pattern Recognition,* Vol. 13, No. 2, 1981, pp. 111-122.

5. Thompson, D. W. and Mundy, J. L., "Three-dimensional model matching from an unconstrained viewpoint", *Proc. IEEE International Conference on Robotics and Automation,* IEEE Council on Robotics and Automation, Raleigh, North Carolina, March 1987.

6. Hwang, V., "Recognition of two dimensional objects using hypothesis integration technique", *Proc. IEEE Workshop on Computer Vision,* IEEE Computer Society. Miami Beach, Florida, November 1987.

7. Sato. Y.. Tamano, R. and Tamura, S., "Connectionist approach to 3-D object recognition: matching of parameterized models and monocular image", *Proc. IEEE Workshop on Computer Vision,* IEEE Computer Society, Miami Beach, Florida, November 1987.

8. Ikeuchi, K. and Kanade. T., "Modeling sensors and applying sensor model to automatic generation of object recognition program", *Proc. of DARPA Image Understanding Workshop,* Science Application Inc., April 1988.

9. Ikeuchi. K., "Generating an interpretation tree from a CAD model for 3-D object recognition in bin-picking tasks", *International Journal of Computer Vision,* Vol. 1, No. 2, 1987.

10. Thomas, S. and Zabele, G. S., "Synthetic aperture radar prediction tools user's manual". Tech. report TIM-5417-2, The Analytic Sciences Corporation, 1988.

11. Carnegie Group Inc., *Knowledge Craft Manual,* 1987.

12. Forgy. C. L.. ''OPS5 user's manual", Tech. report CMU-CS-81-135. Carnegie Mellon University, 1981.