



DOCUMENTATION PAGE

Form Approved OMB No 0704-0188

2

ation is estimated to average 1 hour per response including the time for reviewing instructions, searching existing data bases, completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

| | | | |
|---|---|---|--|
| 1. AGENCY USE ONLY (Leave blank) | 2. REPORT DATE 15 January 1993 | 3. REPORT TYPE AND DATES COVERED Final Report: 15 Sep 1989 - 14 Sep 1992 | |
| 4. TITLE AND SUBTITLE AUTOMATIC TARGET RECOGNITION FOR THE TOPCAT SYSTEM | | 5. FUNDING NUMBERS DAAL03-89-K-0166 | |
| 6. AUTHOR(S) A. C. Kak R. L. Cromwell | | 8. PERFORMING ORGANIZATION REPORT NUMBER | |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Robot Vision Lab Purdue University | | 10 SPONSORING MONITORING AGENCY REPORT NUMBER ARO 27464.1-EL | |
| 9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U. S. Army Research Office P. O. Box 12211 Research Triangle Park, NC 27709-2211 | | 11. SUPPLEMENTARY NOTES The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation. | |
| 12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited. | | 12b. DISTRIBUTION CODE | |
| 13. ABSTRACT (Maximum 200 words) We have developed a proposed sensor system for the TOPCAT anti-armor system. It analyzes cross-sectional measurements of the underlying terrain, detecting targets by their distinctive shape. Sparsely-sampled point from a single scan line are fed as input to a Parzen estimator, which makes the target/no-target decision. Morphological operators then remove spurious noise points, producing large blobs over the detected target region. Results are shown, using the GTVISIT simulation system to produce test data. | | | |
| 14. SUBJECT TERMS ATR, automatic target recognition, TOPCAT, statistical pattern recognition, Parzen classifier | | 15 NUMBER OF PAGES 13 | |
| 17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED | | 16. PRICE CODE | |
| 18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED | 19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED | 20. LIMITATION OF ABSTRACT UL | |

DTIC
EXTRACTED
FEB 19 1993
E D

AUTOMATIC TARGET RECOGNITION FOR THE TOPCAT SYSTEM

FINAL REPORT

A. C. Kak
R. L. Cromwell

15 September 1989 – 14 September 1992

U. S. Army Research Office

Contract DAAL03-89-K-0166

Robot Vision Lab
Purdue University

Approved for public release.
Distribution unlimited.

| | |
|--------------------|-------------------------------------|
| Accession For | |
| NTIS CRA&I | <input checked="" type="checkbox"/> |
| DTIC TAB | <input checked="" type="checkbox"/> |
| Unannounced | <input type="checkbox"/> |
| Justification | |
| By | |
| Distribution/ | |
| Availability Codes | |
| Dist | Avail and/or Special |
| A-1 | |

DTIC QUALITY INSPECTED 3

The view, opinions, and/or findings contained in this report are those of the authors, and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

93-03532



1408

OVERVIEW

We have developed a proposed sensor system for the TOPCAT anti-armor system. It analyzes cross-sectional measurements of the underlying terrain, detecting targets by their distinctive shape. Sparsely-sampled point from a single scan line are fed as input to a Parzen estimator, which makes the target/no-target decision. Morphological operators then remove spurious noise points, producing large blobs over the detected target region. Results are shown, using the GTVISIT simulation system to produce test data.

RESULTS

TOPCAT is a proposed smart anti-armor weapons system. It would be a missile which flew back and forth over the battlefield looking for targets of opportunity. A spinning mm-wave radar sensor (or LADAR sensor) in its nose would search for targets; when one was found, the missile would fire a submunition at it. We have developed a possible algorithm for automatic target recognition. This algorithm has been tested on synthetic data produced by the GTVISIT simulation system.

Figure 1 shows a schematic top view of a tank and a tree. A scene like this one was generated by GTVISIT, and processed into simulated TOPCAT data (this involves generating a new range image from the viewpoint of the center of each rotational scan, and then sampling points lying along an arc in this image). Figure 2 shows the range profiles associated with the scans one through six in Figure 1.

We know that armored vehicles will be roughly rectangular in cross-section. Natural clutter will have range profiles with a high variance, and are therefore unlikely to exhibit the characteristic rectangular shape (or, at most, only in very isolated cases). There are three general approaches to measuring this shape and making the associated decisions:

- A rule-based system that analyzes the presence or absence of step edges and straight line segments.
- A signal-processing based approach.
- A statistical pattern recognition type of classifier, where the feature vector is a sequence of range (or height) measurements. Since the amount of training data required for accurate classification is proportional to the length of the feature vector, this should be kept to a minimum, suggesting sparse sampling.

We elected to use the third approach; it yielded an accurate and computationally efficient method. A robust non-parametric classifier, the Parzen classifier, was implemented. Being non-parametric, this classifier does not entail any *a priori* assumptions regarding the probability distributions of the height data generated by the Topcat sensor. More specifically, unlike the more commonly used classifiers, this classifier does **NOT** assume that that the height values for, say,

the trees are Gaussian.

In our Parzen classifier for single scan line data, we use N adjacent height values as features. To provide further explanation regarding classification, we will denote a set of N adjacent height values by f_1, f_2, \dots, f_N . During the training phase, a large number of sets of N feature values are collected from the synthetic data, first from just the tanks and then from just the background. These sets of N values correspond to single points in an N dimensional feature space. During the training phase, let the vector \mathbf{f}_i^{tank} denote i -th such point. If we use, say, 1000 sets of these N values derived from just the tanks, then the following function constitutes a Parzen estimate of the probability density function corresponding to just the tanks:

$$p_{tank}(\mathbf{f}) = K \times \sum_1^{1000} h(\mathbf{f} - \mathbf{f}_i^{tank})$$

where p is the probability density function and \mathbf{f} any arbitrary point in the feature space. The function h , referred to as a kernel function, admits many different possibilities, such as rectangular, triangular, Gaussian, etc., each different choice affecting differently the rate of convergence of the estimated density function to its true value. We use Gaussian a kernel since it has been demonstrated to yield the best rates of convergence for a given number of samples for training. Note that just because we are using a Gaussian kernel for h , it does not imply that the classification technique is parametric; the shape of the overall density function can still be arbitrary. The factor K in the above formula is a normalization constant; its purpose is to make certain that the area under the probability density function is unity.

We can similarly collect, say, a 1000 sets of N adjacent heights for the background clutter such as trees. If we represent each set by \mathbf{f}_i^{trees} , we can form the following Parzen estimate of the probability density function corresponding to the background trees:

$$p_{trees}(\mathbf{f}) = K \times \sum_1^{1000} h(\mathbf{f} - \mathbf{f}_i^{trees})$$

After a classifier is trained by estimating p_{tank} and p_{trees} , a set of measurements on an unknown target can then be classified by invoking the Bayes rule. In its simplest implementation, which assumes that the prior probabilities associated with the tanks and the trees are identical, the set of measurements would correspond to, say, a tank if and only if the following condition is satisfied:

$$p_{tank}(\mathbf{f}) \geq p_{trees}(\mathbf{f})$$

Note that this approach to classification captures the correlations, at multiple orders, between the different height values. It seems intuitively plausible that the adjacent height values for a tank would be more highly correlated than the adjacent values for a bush or a tree

The algorithm processes the data as follows. First, each scan line is median filtered, an optional step that may be used to reduce the noise present in a real sensor. The data is then normalized, to make the decision insensitive to missile altitude. Sparsely-sampled feature vectors are then extracted, and a Parzen classifier is applied. Morphological operators then reduce the noise, leaving large contiguous regions labeled as "target" on the armored vehicles. Figure 3

shows an overview of our method.

To address some of these steps in more detail, the normalization can be as simple as subtracting the average range for a scan line from all points on it. The rectangular shape is produced by the derivative, not the absolute range. Therefore, the instantaneous missile altitude has no effect on the decision process (except, of course, for any effects it may have on sensor accuracy or noise characteristics).

Once the data has been median filtered and normalized, feature vectors must be extracted. Since the objects of interest are well known, the overall size of this feature vector is specified. For instance, a Russian-built T-72 measures approximately 8.1 meters corner-to-corner across the hull. Therefore, the feature vector must be large enough that, even in a worst-case scenario where a scan line hits both extreme corners, the outermost points on the feature vector will lie on the ground and the interior ones on the vehicle itself.

With the available information on the sampling rate, spin rate, and typical altitude of the proposed sensor system, it appeared that the following would be a useful feature vector. Note that r_i is the i 'th range measurement in a scan line, \hat{r} is the median-filtered version of r , and X_i is the feature vector for the i 'th pixel in the scan line.

$$X_i = \begin{bmatrix} \hat{r}_{i-10} - R_i \\ \hat{r}_{i-5} - R_i \\ \hat{r}_i - R_i \\ \hat{r}_{i+5} - R_i \\ \hat{r}_{i+10} - R_i \end{bmatrix}$$

where:

$$R_i = \frac{\hat{r}_{i-10} + \hat{r}_{i-5} + \hat{r}_i + \hat{r}_{i+5} + \hat{r}_{i+10}}{5}$$

Since R_i is a local average, then the characteristic increase in range at the end of the scan lines has a decreased effect. The feature vector X_i will still exhibit some slope if the sensor was viewing a roughly horizontal surface, but X_i will be centered around zero.

There will be some holes in the region labeled as "target", pixels where the Parzen classifier chose "non-target". Similarly, there will be isolated points in the background, or particularly likely, in the clutter regions, mistakenly labeled as "target". A morphological filter very quickly removes these single-point (or few-point) errors, yielding a binary image labeled as "target" in regions overlying points where the vehicles were originally located, and "non-target" elsewhere. Since the correct "target" regions are large blobs, and the false-positive points are isolated, it is a simple matter to simply ignore small regions.

Figure 4 shows simulated TOPCAT data for a cluttered scene, containing three T-72's and six trees. Range is shown as intensity; the darker splotches are the tall trees. Careful examination

of the target regions shows the turret protruding above the deck, thus slightly darker, and the barrel projecting out the front.

Figure 5 shows the points in this scene initially labeled as "target" by the Parzen estimator, and Figure 6 shows the detected target points output by the morphological operator. Notice how the "target" regions accurately correspond to the tanks in Figure 4.

The proposed TOPCAT system would be in a fast-moving missile with a slight look-ahead angle. Figures 7 and 8 show that only a few scan lines are needed to detect a target. In this way, TOPCAT could detect a target and still have adequate time to fire the submunition before passing over the vehicle.

CONCLUSIONS

We are very pleased with the promising results we have shown. This algorithm can detect armored vehicles with a very high probability of detection and a very low false alarm rate, even when the vehicles are in a cluttered scene. Even in the presence of clutter, only narrow strips of data corresponding to a few scan lines are needed for recognition, meaning that the algorithm would be amenable to real-time application. In all, we feel that this demonstrates a useful TOPCAT algorithm.

Sequence of Ladar Scan Lines

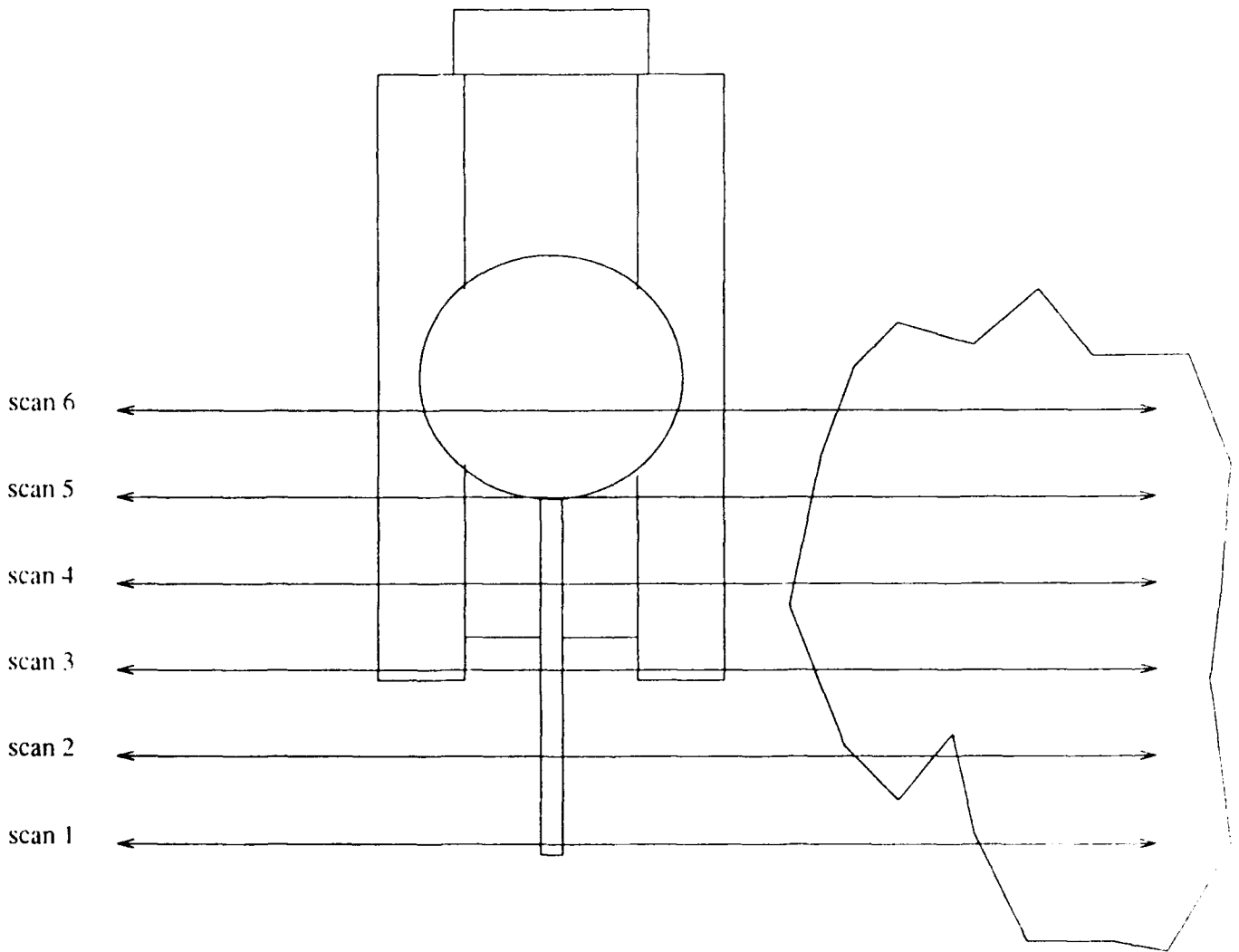


Figure 1

T-12

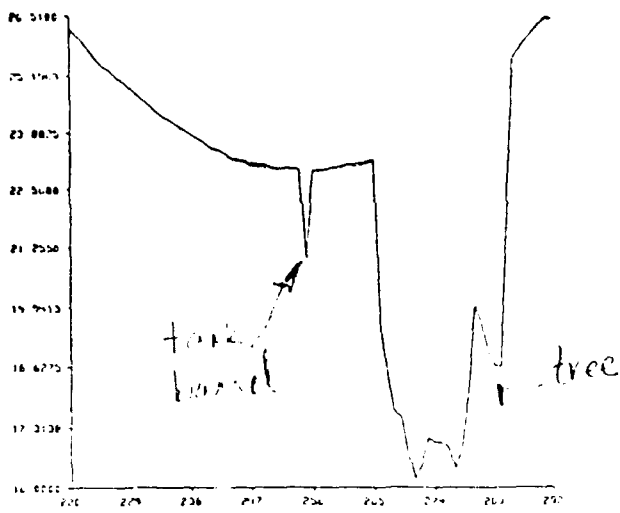


Image 1

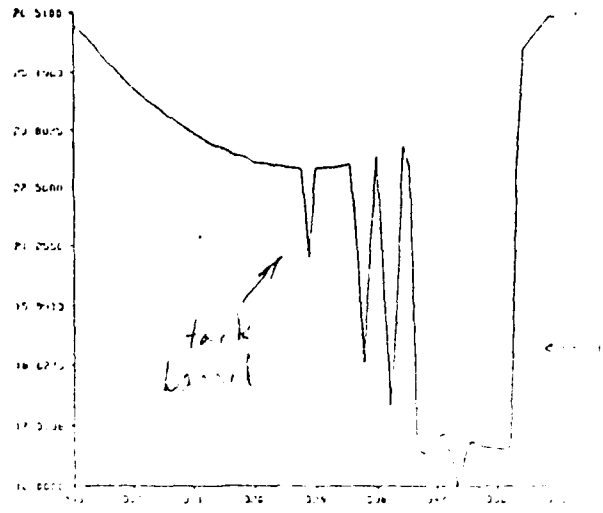


Image 2

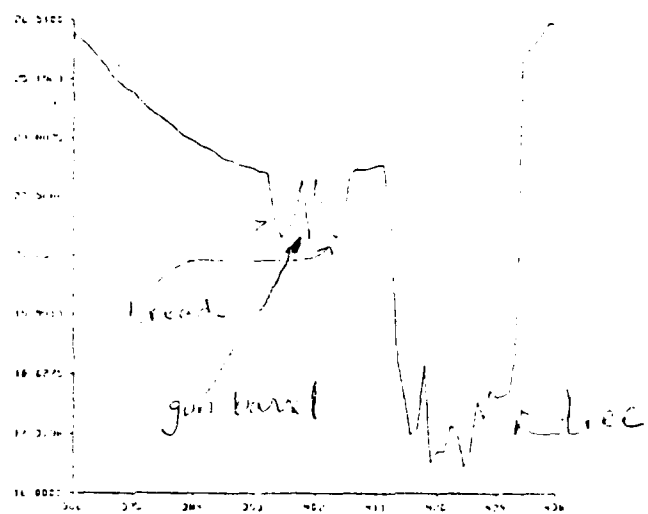


Image 3

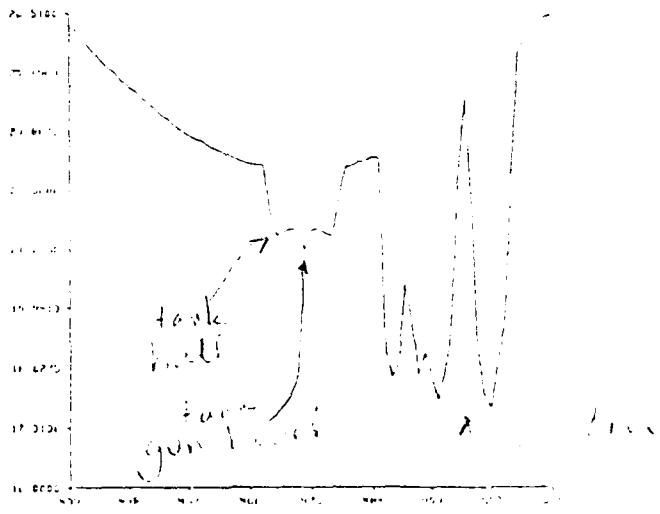


Image 4

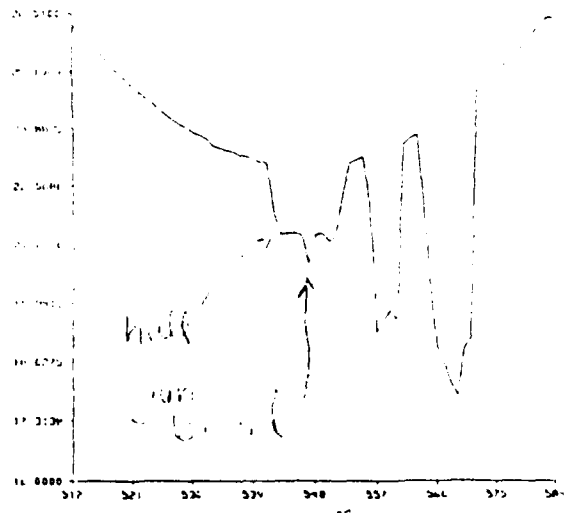


Image 5

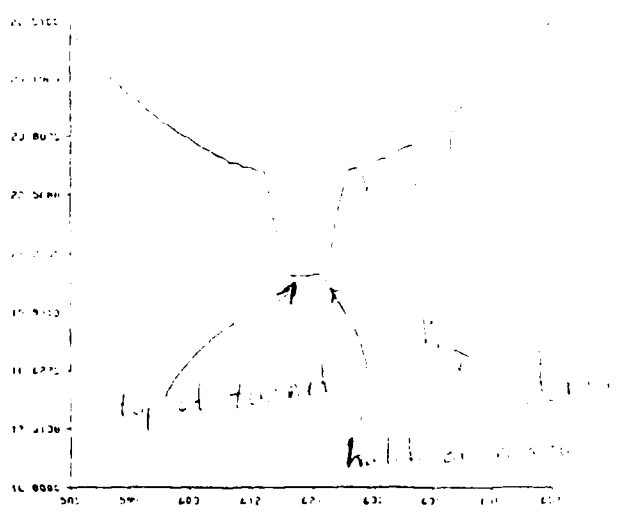


Image 6

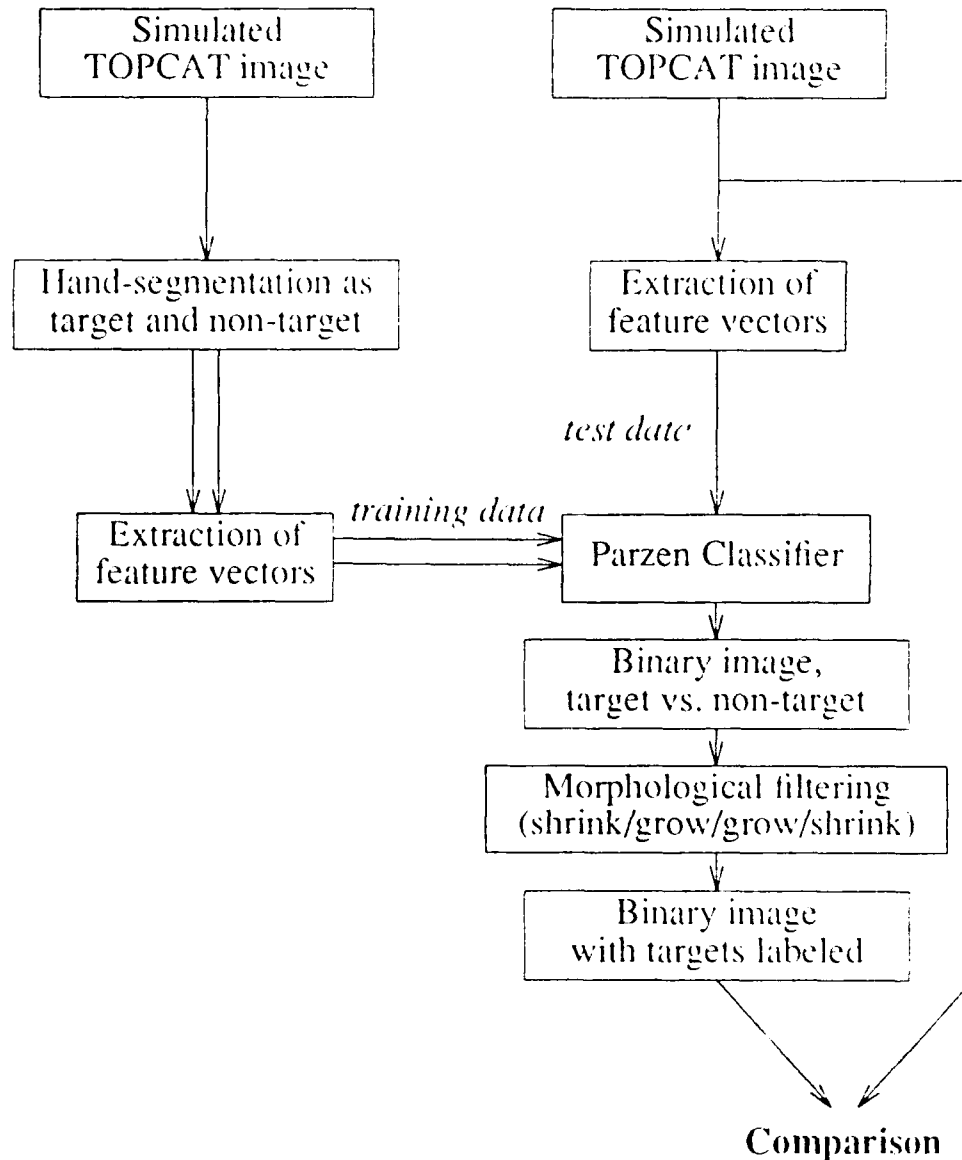
Simulated sensor data for topcat images

Figure 2

Figure 3

The feature vectors are used for statistical pattern classification. The Parzen method is used to estimate the probability density functions for target and non-target distributions.

Overview of method:



Scene 0

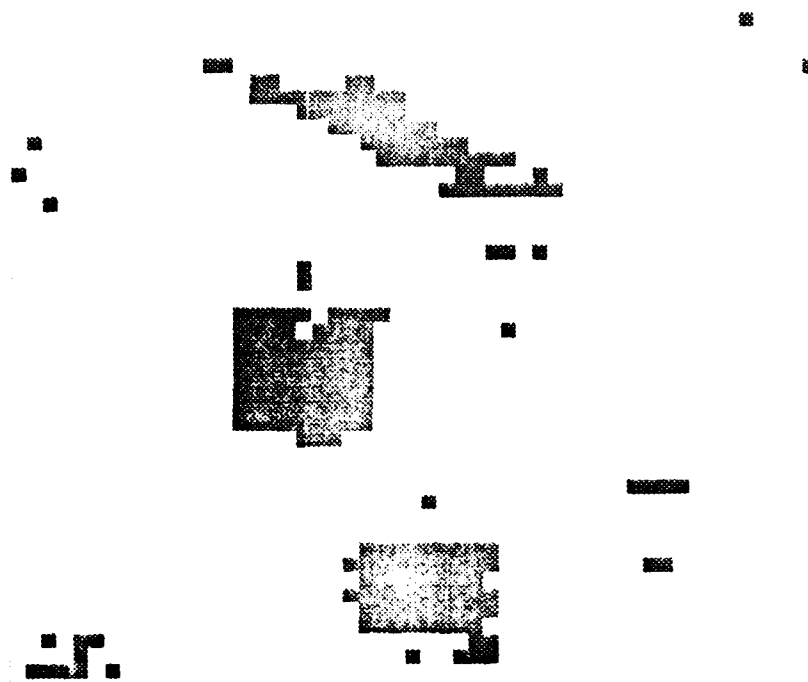


Simulated TOPCAT Image

| | |
|--------------------|---------------------|
| Missile Speed: | 274.3 m/sec |
| Altitude: | 22.5 m |
| Spin Rate: | 150 rev/sec |
| Sensor Pulse Rate: | 70 KHz |
| Laser Footprint: | 6.4 mrad x 1.0 mrad |
| Lookahead Angle: | 12° |
| Field of View: | 60° |

Figure 4

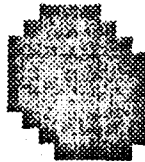
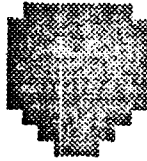
Scene 0



Initial Target Points

Figure 5

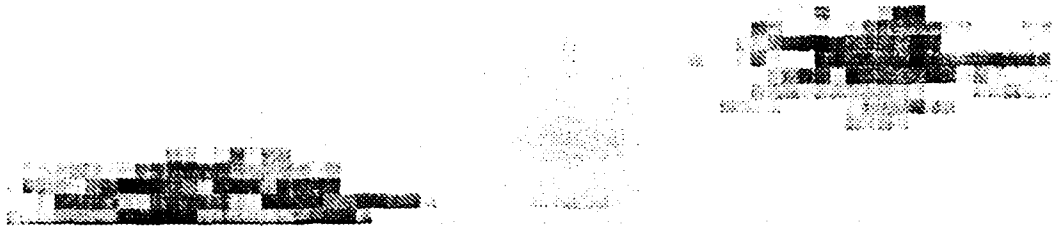
Scene 0



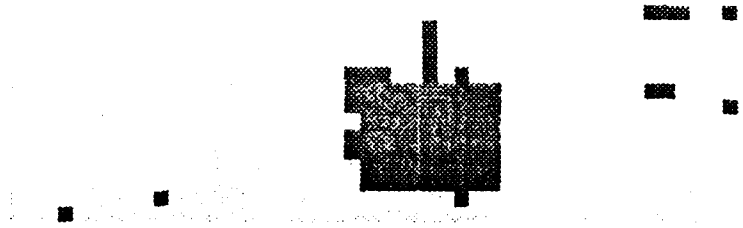
Detected Target Points

Figure 6

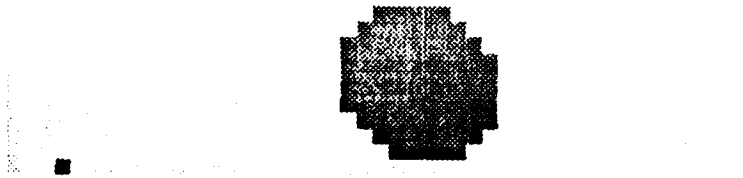
Scene 1



Simulated TOPCAT Image



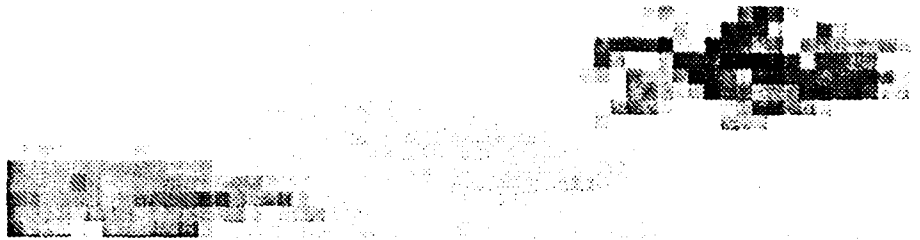
Initial Target Points



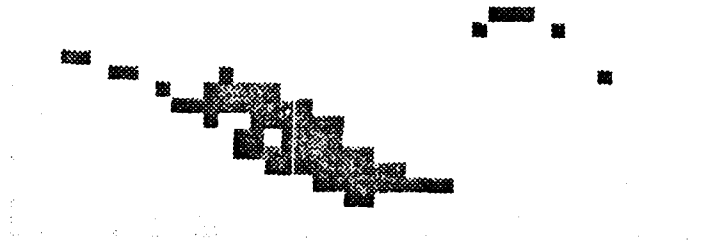
Detected Target Points

Figure 7

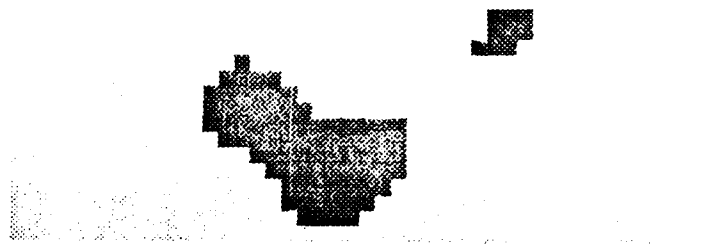
Scene 2



Simulated TOPCAT Image



Initial Target Points



Detected Target Points

Figure 8