An Automatic Ship Classification System for ISAR Imagery

Murali M. Menon, Eric R. Boudreau, and Paul J. Kolodzy

■ We have designed and developed an automatic ship classification (ASC) system for classifying potential naval targets from inverse synthetic-aperture radar (ISAR) imagery. Initially, the ASC system was developed for an over-the-horizon targeting application. In the application, an airborne platform with an on-board ISAR sensor transmitted imagery to a host ship carrying the ASC system. Our present focus is on placing the ASC system on board the sensor platform to assist the flight crew in classifying naval vessels.

The current ASC system uses both neural network and conventional processing techniques to determine the ship class of a target from ISAR imagery acquired during reconnaissance missions. An Adaptive Clustering Network (ACN) allows a single ship class to be distributed across several categories so that the system develops a degree of invariance to target motion. We have evaluated the ASC system on a limited set of actual ISAR imagery collected during operator training missions and on a larger database of imagery from IBM. Our preliminary results indicate that an operational ASC system with performance levels comparable to human operators can be achieved. From these results, we feel that the ASC system is now ready for a thorough field evaluation on board an ISAR sensor platform.

An automatic ship classification (ASC) system [1–3] for classifying potential naval targets from inverse synthetic-aperture radar (ISAR) imagery was initially developed at Lincoln Laboratory for an over-the-horizon targeting application. (For a description of the fundamentals of ISAR, see the box entitled "Inverse Synthetic-Aperture Radar [ISAR] Imagery.") In the application, an ISAR on board a Navy helicopter transmitted imagery to a host ship carrying the ASC system. After preliminary testing in June 1991, the system was installed on board a Navy destroyer prior to the vessel's deployment to Hawaii. The high-resolution data acquired during the Hawaii mission was used for subsequent laboratory evaluations and refinements of the ASC system. Although excellent performance was attained, the small size of the Hawaii database (approximately 500 frames) could provide only a rough estimate of the system's actual operational performance.

The present focus of the system's development is on placing the ASC system on board the sensor platform to assist the flight crew in classifying naval vessels. In support of these efforts, we have recently acquired from IBM Federal Systems a database containing imagery of ships of 12 different classes. This database is significantly larger than the Hawaii database, and the IBM targets have each been imaged at a variety of sensor-to-target orientations (in contrast to the Hawaii data, which were primarily acquired at a single target orientation). The IBM imagery, however, is significantly lower in video quality than the Hawaii database. Furthermore, the IBM imagery was acquired from a lower-resolution sensor. Thus, as expected, the performance of the ASC system on the IBM data was lower than that on the Hawaii data.

We have implemented several new techniques and algorithms both to improve the overall performance of the system on the IBM imagery and to reduce the

INVERSE SYNTHETIC-APERTURE RADAR (ISAR) IMAGERY

ALTHOUGH SIMILAR in concept, inverse synthetic-aperture radar (ISAR) differs from synthetic-aperture radar (SAR) in one fundamental way: SAR images are generated by the motion of the sensor platform with respect to the target, whereas ISAR images are generated by target rotation. For the case of ships, rotation around any of the three principal axes generates the image. Consequently, ISAR imaging requires that any relative motion between the radar and the ship be compensated, leaving only the rotation of the ship on the ocean surface. Hence, motion compensation is accomplished by tracking a point on the target that provides a consistent, strong radar return.

The reflected energy from a complex three-dimensional (3-D) target such as a ship is detected and processed into range-Doppler space. If the ship is broadside relative to the radar, then there will be very little range extent, and a mostly range-unresolved image will result. Such an image is referred to as a beam-orientation view. Bow-orientation and stern-orientation views occur when the ship is nearly head on (bow or stern, respectively) to the radar and highly range-resolved images are produced. The details of the imagery result from the rotational motion of the ship, which gives rise to different velocities for the various ship features.

Measurement Theory

The ISAR image is a range-Doppler representation of a rigid rotating object. The range-Doppler principle implies that an appropriate signal can be processed to determine the range (time delay) and radial velocity (Doppler frequency) of each scattering element of the object. By associating the time delay and Doppler frequency with each element on a rigid rotating object, we can use the radial velocity gradients across the object to generate the object's image. The image intensity is proportional to the returned signal strength, which depends on the radar cross section of the scattering element.

The range r of a particular scattering element on a rigid rotating object is given by the time delay Δt between the transmitted and received signals:

$$r = \frac{c\Delta t}{2},$$

where c is the speed of light. The rigid object rotates at an angular velocity of ω radians per second about a fixed axis. If the scattering element is located a distance d from the origin and is rotated α radians from the vertical axis, the corresponding Doppler frequency f_d is given by the following equation:

$$f_d = \frac{2\omega d \cos\alpha}{\lambda} = \frac{2\omega y}{\lambda},$$

where λ is wavelength and y is the distance of the element from the horizontal axis. Given a radar coherence time $T_c = 1/\Delta f_d$, the Doppler resolution Δy can be found with the following equation:

$$\begin{aligned}\Delta y &= \Delta f_d\left(\frac{\lambda}{2\omega}\right)\\ &= \frac{1}{T_c}\left(\frac{\lambda}{2(\Delta\theta / T_c)}\right)\\ &= \frac{\lambda}{2\Delta\theta},\end{aligned}$$

where $\Delta\theta$ is the angle of rotation during the radar coherence time T_c.

For further details of ISAR measurement theory, see References 1 and 2.

References

1. M.L. Skolnik, *Introduction to Radar Systems* (McGraw-Hill, New York, 1980).
2. S.A. Hovanessian, *Introduction to Synthetic Array and Imaging Radars* (Artech House, Dedham, MA, 1980).

number of misclassifications to an operationally acceptable level. In the preprocessing stage, for example, an automatic frame rejecter has been added to identify frames that have excessive noise as well as frames that have no targets. We have also incorporated a *class contrast* mechanism that enables the system to report ambiguous ship imagery as "unknown," thereby reducing the number of misclassifications. In addition, the system has been extended to take advantage of the multiple frames obtained during observation intervals of the target ships. The multiple frames are used to accumulate evidence for a particular decision, thereby reducing errors that arise when only a single frame of a target is used. And we have also incorporated target-orientation information into the system to reduce misclassifications.

This article is divided into three major sections: "Databases Used" describes both the data collection mission of August 1991 and the database derived from the IBM videotapes; "System Description" presents details of the ASC system; and the remainder of the article describes the system performance achieved with the Hawaii and IBM databases.

Databases Used

Hawaii Mission and Database

A preliminary version of the ASC system was initially demonstrated in June 1991 at the North Island Naval Air Station (NAS) in San Diego. The system, which was installed at the NAS site, processed imagery from flight videotapes that had been generated by an ISAR sensor platform on board a Navy helicopter. The classification performance of the system at the NAS site warranted an operational evaluation. Thus in August 1991 the system was installed on board a Navy destroyer. During the ship's subsequent transit to Hawaii, sensor evaluation missions were flown, and a database was compiled from videotapes that had been generated by the helicopter's on-board ISAR.

The sensor imaged the target ships primarily at near-bow and near-stern sensor-to-target orientations. For the majority of frames, the target was centered with little or no drift, fading, or truncation. A total of 502 frames were collected for six different classes of target ships. Table 1 shows the number of frames in each class.

Table 1. Hawaii Database

Class	*Class Type*	*Number of Frames*
H1	Frigate	7
H2	Oil supply ship	205
H3	Destroyer	35
H4	Commercial (assorted)	31
H5	Destroyer	126
H6	Destroyer	98
Total		502

IBM Database

The ISAR imagery from IBM Federal Systems was obtained in the form of four videotapes containing a total of roughly three hours of data. The visual quality of the data appeared to have suffered from the degradation that results from multigenerational videotape copying. Each tape contained different numbers of observations of each target class. A title screen indicating the class of the imaged ship preceded each observation, and a color bar that lasted for several seconds marked the beginning and end of the observation. This format greatly facilitated the formation of the database. The video imagery within each observation was digitized at a rate of about 1 frame/sec, and the resulting frames were stored on a separate disk file. As a result, the full range of target variations was captured in the database. The digitized data were then screened to remove frames that did not contain ISAR data and to eliminate any numerals that had been superimposed on the imagery by the sensor system.

Because the Hawaii data were collected from an experimental high-resolution ISAR sensor, only limited data were available from that system. Thus imagery containing only six ship classes at a single orientation could be obtained for processing. The IBM data, on the other hand, were collected by low-resolution ISAR sensors for which a significant amount of data was available. Thus imagery containing 12 ship classes of up to five target orientations each were obtained from IBM.

The ASC system preprocessed the IBM imagery with a frame-rejection algorithm to remove images that were either devoid of targets or of a low signal-to-noise ratio. Because the system was configured to process single targets in a field of view, observations that contained two targets were disqualified. Table 2 lists the total number of usable observations and frames for each of the 12 classes of ships. The table indicates that the numbers of observations and frames were not evenly distributed among the 12 ship classes. The ground truth and number of frames for each observation of each class in the final database are contained in Reference 3. The ground truth provided in that reference was taken from a document that accompanied the video tapes, which roughly described the sensor-to-target orientation as bow, oblique bow, stern, oblique stern, and beam (i.e., broadside) for each observation.

System Description

ISAR imagery is characterized by a high degree of variability. Two frames of the same ship seldom look exactly alike. The ASC system automatically addresses variations in ISAR imagery by employing the Adaptive Clustering Network (ACN) classifier, which generates separate categories, or clusters, for the various presentations of a target.

In the baseline ASC system, the processing consists of target segmentation, clutter rejection, centering, feature-vector extraction, and classification by the ACN, as illustrated in Figure 1. The target is segmented from the noise background by applying a threshold based on the image statistics. After clutter is eliminated, a feature vector is formed by summing the intensities in each range bin of the cleaned image. During training, the ACN gathers similar feature vec-

Table 2. IBM Database

Class	*Class Type*	*Number of Observations*	*Number of Frames*
1	Cruiser	16	1910
2	Cruiser	4	338
3	Frigate	6	662
4	Frigate	8	830
5	Battleship	18	1668
6	Ice breaker	6	357
7	Minesweeper	3	132
8	Destroyer	9	771
9	Frigate	36	1403
10	Cruiser	12	1217
11	Hydrofoil	5	244
12	Aircraft carrier	14	1403
Total		137	10,935

tors together to form clusters. During testing, the ACN reports the class name of the cluster that best matches a particular feature vector.

Segmentation

Figure 2 shows an artist's rendition of ISAR imagery after digitization, and Figure 3 contains the scaled histograms of the background pixels and the target pixels. Note that the application of an appropriate threshold to the image intensity values will remove most of the background noise while leaving the target mostly intact. The ASC system performs the operation automatically by

$$z_{ij} = \begin{cases} x_{ij} & \text{if } x_{ij} > m_{\text{global}} + N\sigma_{\text{global}} \\ 0 & \text{otherwise,} \end{cases}$$

where x_{ij} is the input image value, m_{global} is the global mode of the image, σ_{global} is the global standard deviation, and N is a user-selected parameter. The value

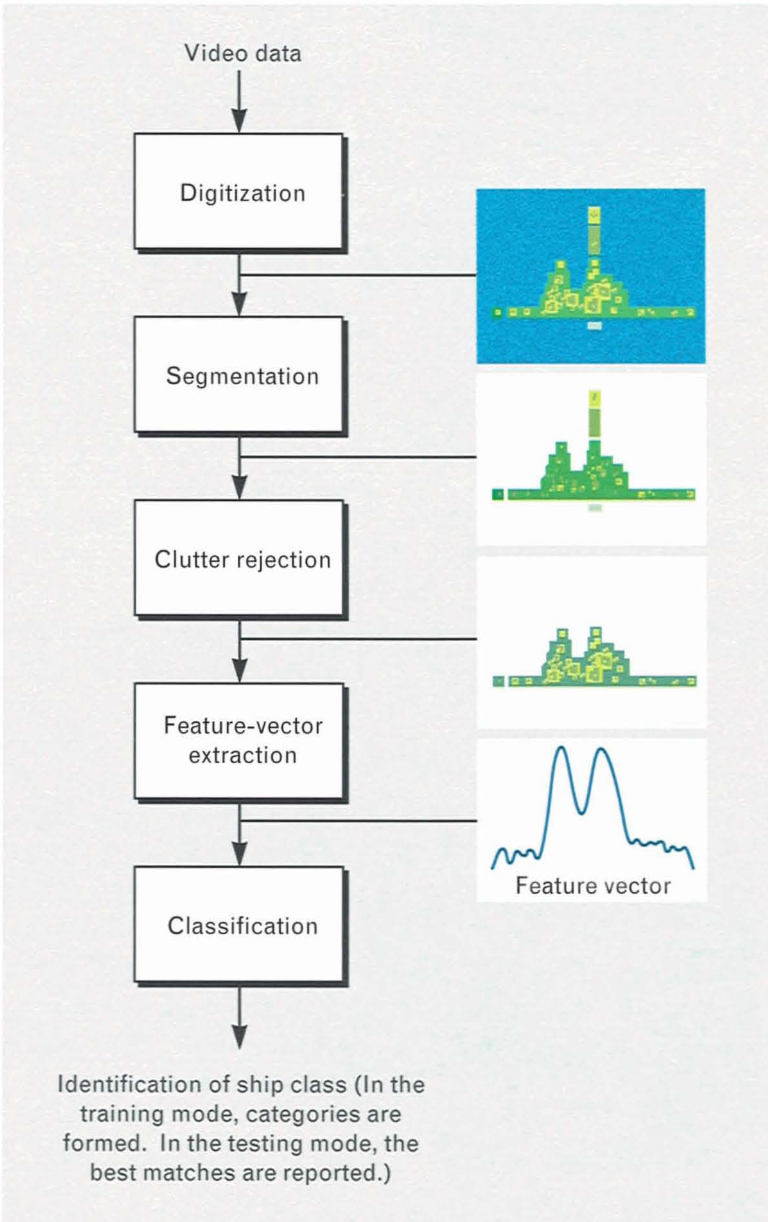


FIGURE 1. Automatic ship classification (ASC) system concept.

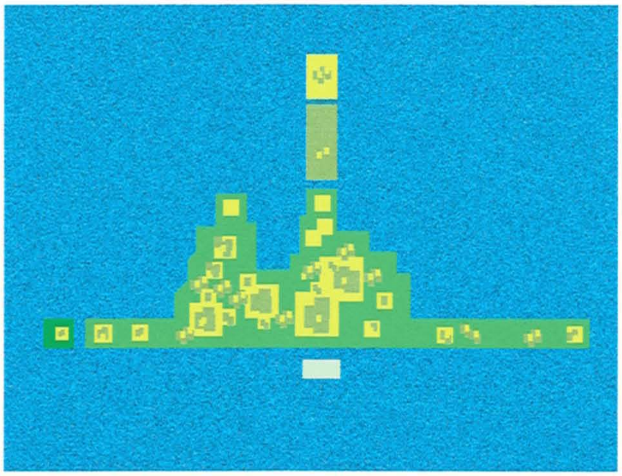


FIGURE 2. Artist's rendition of ISAR imagery after digitization.

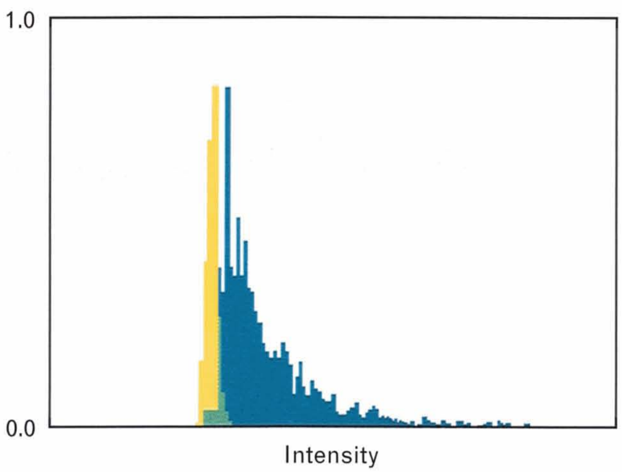


FIGURE 3. Scaled histogram of background pixels (yellow), target pixels (blue), and the overlap (green) for the imagery shown in Figure 2.

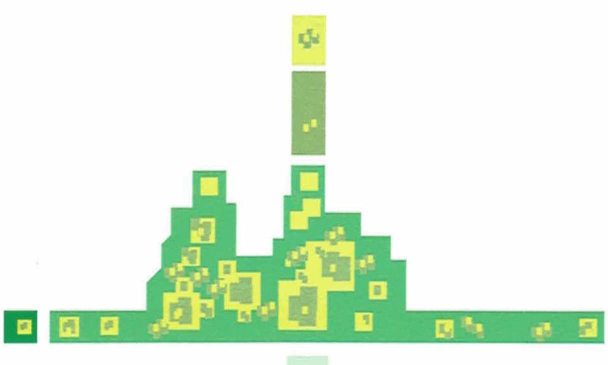


FIGURE 4. The result of target segmentation performed on the imagery in Figure 2.

of the parameter was chosen to provide the best classification performance and Figure 4 shows the resulting image.

Clutter Rejection

ISAR imagery often contains vertical streaks as seen in Figure 2. Figure 4 shows that the target-segmentation process does not remove this vertical streak. These streaks can have an adverse effect on the system's performance. During feature-vector extraction (to be discussed later), the intensities in each range bin are summed to form the profile of the ship. A result of this process is the formation of large peaks in the profile at the locations of the vertical streaks. When the profile is later normalized by scaling, the large peaks act to suppress details of the ship.

To eliminate vertical streaks and other outliers, we can group the nonzero pixels in the segmented image, find the major axis of the ship by using a weighted least-squares line fit, and eliminate those pixel groups whose centers are more than 10 pixels away from the least-squares line [4, 5]. Figure 5 illustrates this process. In addition to generating a profile that better represents the ship, this procedure also removes noise pixels that can adversely affect measurements performed on the segmented target.

Centering

The location of a target in the sensor field of view is controlled by the ISAR operator during data collection. Centering a target in the sensor field of view is difficult when there is relative motion between the target and the sensor platform—a target often shifts its location in the image frame during an observation, and a target may not be positioned in the same location between observations. Because the current version of ACN cannot account for such effects, the ASC preprocessor must center the segmented target. The centering can be accomplished by shifting the image so that its intensity-weighted center of mass is in the center of the frame.

Extraction of Feature Vector

ISAR imagery is three dimensional: each pixel is mapped according to its range-Doppler position (x, y), and the pixel intensity is proportional to the radar return. Because the summed intensity distribution over range remains consistent during target perturbations, the feature vector is formed by accumulating the intensities for each range bin. This process consistently provides a useful feature vector, even when there is little vertical extent in the imagery (Figure 6). The ASC system also accomplishes a type of smoothing, or coarse coding, by summing the data for a particular bin with the data for the two adjacent bins.

During an ISAR observation, the overall vertical extent of a target can fluctuate widely. Thus the feature vector must be normalized to allow the classifier to group the varying images correctly into the same class. The current system normalizes the feature vector by scaling to 8 bits the profile values that are between 0 and the maximum [6]:

$$f_i = 255\left(\frac{f_i - f_{\min}}{f_{\max} - f_{\min}}\right).$$

Because the ship length is incorporated into the scaled feature vector, the accurate extraction of this measurement is critical. The initial system found the length by summing the occurrences of nonzero range bins. With this procedure, the occurrence of a few outliers did not have a significant effect on the resulting length, but gaps in the target could reduce the measured length considerably. Another method of finding the length is to apply the clutter-rejection routines to the segmented image to eliminate most of the outliers. Then the length can be found by taking the difference between the minimum and maximum nonzero range bins. This length will be immune to dropouts in the target, but it will be very susceptible to any outliers that have not been removed by the application of thresholds and clutter-rejection techniques. The current system incorporates both of the above measurement methods to take advantage of their respective strengths. The two calculated lengths are averaged to produce a length that is used to scale the feature vector, and this average length measurement is then appended to the feature vector.

Adaptive Clustering Network (ACN)

After the feature vector has been extracted, the salient information in the vector needs to be isolated. One

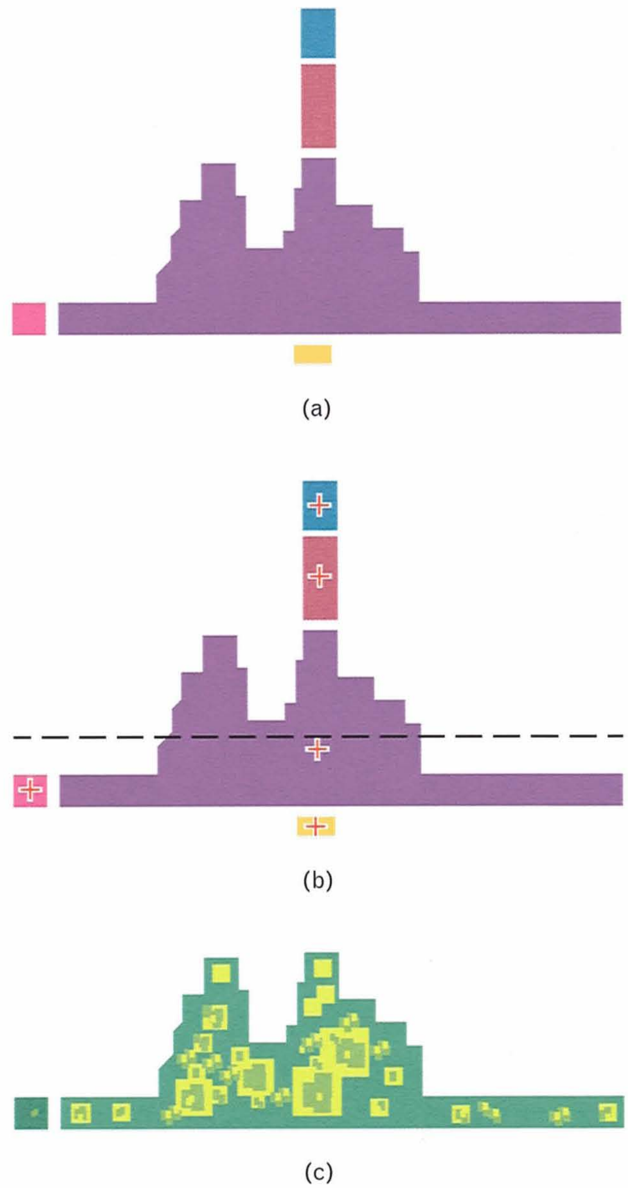


FIGURE 5. The result of clutter rejection performed on the imagery of Figure 4: (a) the nonzero pixels are grouped with a single morphological clustering algorithm (the resulting five different groupings are shown in five different colors), (b) the ship's major axis (denoted with a dashed black line) is found by using a weighted least-squares line fit that is weighted by pixel intensities, and (c) pixel groups whose centers are more than 10 pixels away from the major axis are eliminated. (In part *b*, the centers of the five different pixel groups are denoted with red "+" signs.) Note that the use of clutter rejection eliminates vertical streaks and outliers that may be present in the imagery. In addition to generating a profile that better represents the ship, the process also removes noise pixels that can adversely affect measurements on the segmented target.

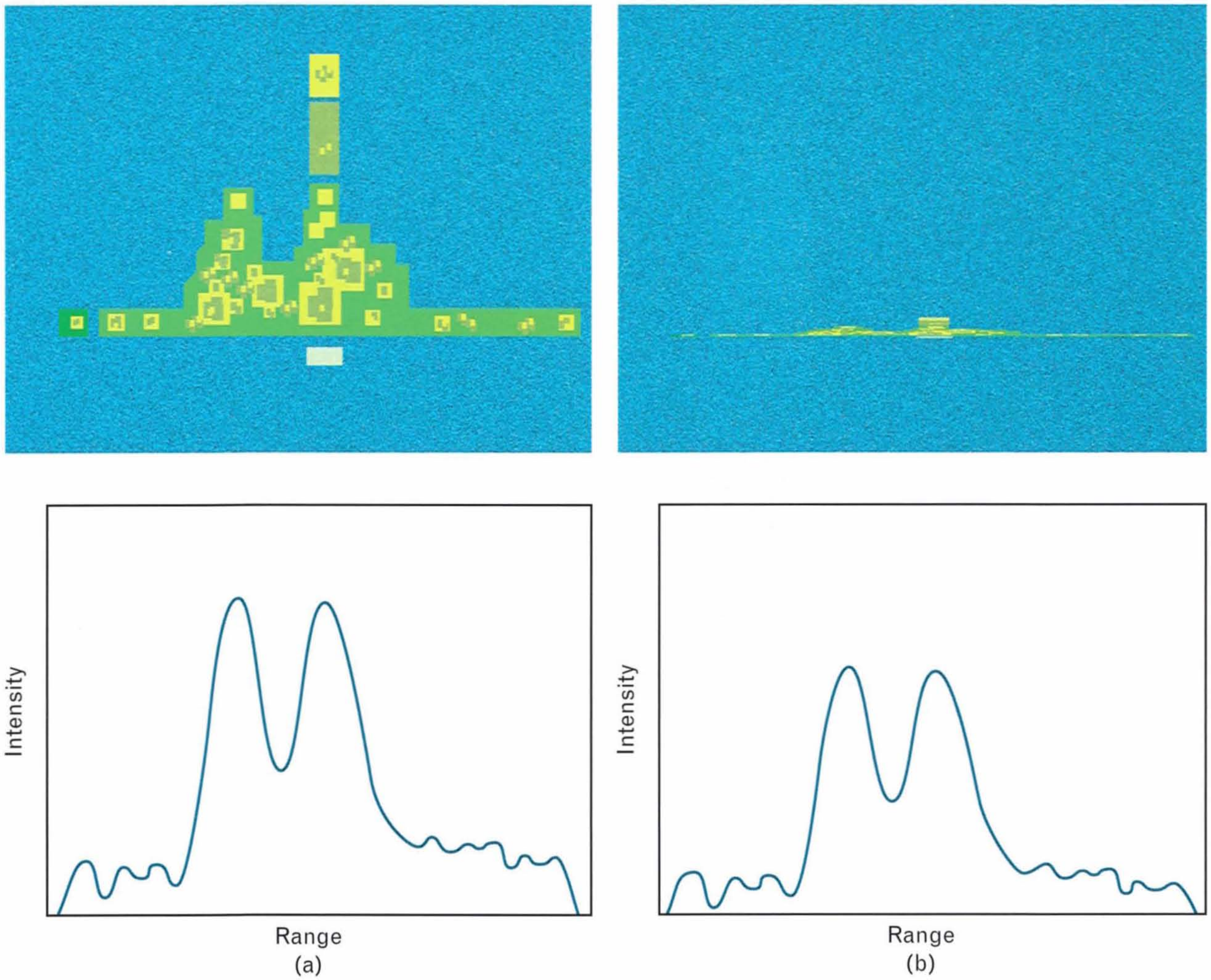


FIGURE 6. Artist's rendition of ISAR imagery and corresponding feature vector for (a) typical case and (b) transition state.

approach looks for a fixed set of relationships that defines the target classes, and classifies inputs based on this information. The difficulty with this approach is in the selection of a metric that provides separability across the wide range of inputs. The Adaptive Clustering Network (ACN) [7], an unsupervised neural-network-based associative classifier, avoids the need for such a metric by automatically determining the relevant information in the input through calculations of the correlations between different feature vectors. Essentially a clustering algorithm, the ACN automatically groups feature vectors together to form clusters, and a slow learning rule (described below) is used to adapt the cluster centers as new data are encountered. The ACN avoids the construction of a look-up table of the training data by allowing overlap between clusters (i.e., multiple classes are permitted in the same cluster). The overlap is resolved later (after the training process has been completed) by statistically assigning a class name to each cluster.

The ACN shares many common attributes with other classifiers described in the literature, in particular, the Adaptive Resonance Theory (ART) neural network [8]. The ACN, however, is unique because it combines a slow learning rule with statistical methods to assign class names to clusters for single-frame and multiframe classification. The ACN is similar to the Recursive Coulomb Energy (RCE) model [6, 9–12] in that new clusters are automatically formed, but the RCE does not adapt the cluster centers as new data are encountered. The *k*-means algorithm [13] updates the cluster centers by averaging all exemplars

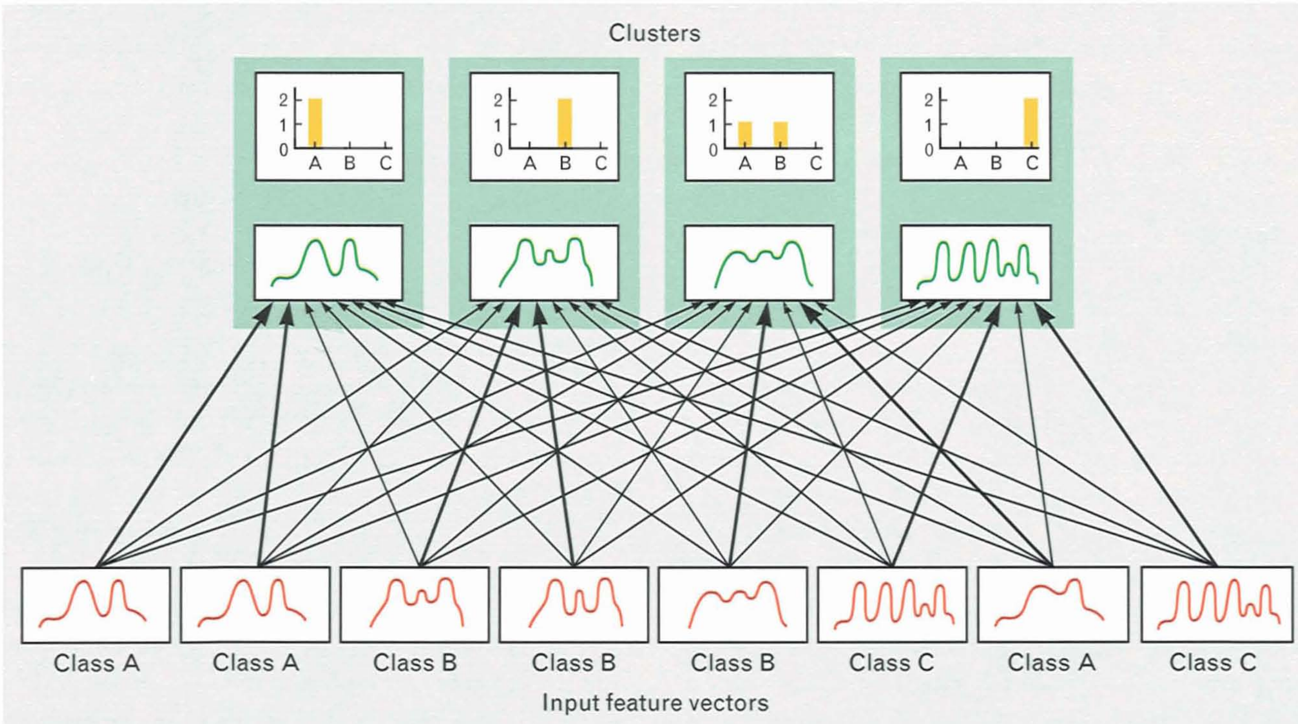


FIGURE 7. Adaptive Clustering Network (ACN). For a detailed description of the network, see the main text.

associated with a cluster, while the ACN uses slow learning for the selective reinforcement of localized regions. The standard *k*-Nearest Neighbor (*k*-NN) algorithm [14] uses the most frequently occurring cluster to assign class names but does not cluster the training data. The Learning Vector Quantizer (LVQ) [15] clusters the training data but requires a *supervised* scheme for adjusting the cluster centers to reduce classification errors.

Figure 7 shows a diagram of the ACN. In the network, inputs are grouped together into clusters by calculating the correlations between the inputs. For ship recognition, the training correlation c_{train} between a feature vector $\mathbf{F}_{\text{input}}$ and an existing cluster $\mathbf{F}_k$ is defined as

$$c_{\text{train}}^{k} = \frac{\mathbf{F}_{\text{input}} \cdot \mathbf{F}_k}{\left\|\mathbf{F}_{\text{input}}\right\| \left\|\mathbf{F}_k\right\|}.$$

If the best match from the above equation is less than the training correlation threshold λ_{train} (i.e., if $\max_k c^{k}_{\text{train}} < \lambda_{\text{train}}$), then a new cluster is added. Otherwise, the best-match cluster is updated according to a simple "learning" rule given by

$$\mathbf{F}_k^{\text{new}} = \mathbf{F}_k^{\text{old}} + \gamma\left(\mathbf{F}_{\text{input}} - \mathbf{F}_k^{\text{old}}\right),$$

where γ is the learning rate, which governs the rate of update of a cluster $\mathbf{F}_k$. For $\gamma = 1$, the input value replaces the cluster. For $\gamma = 0$, no update occurs. Typically, the learning rate is set to a value between 0.10 to 0.25 (called "slow" learning) to suppress small variations caused by noise and target shifts in the image frames, and to reinforce consistent features in the images. After the training database has been processed, each cluster is assigned a class name (described below).

The ACN is an unsupervised neural network in the sense that the feature vectors are allowed to group together into clusters regardless of what class they represent. Thus it is possible that a given cluster may contain several different input classes, as illustrated in Figure 8. The class with the highest frequency of occurrence in a cluster is the class name that is applied to that cluster. Ideally, one class would contain most of the entries in a given cluster. However, if there exist different classes with similar feature vectors, then a given cluster may not be dominated by a single class. Such clusters can result in incorrect classifications during the testing process. To reduce this possibility, the ACN reviews the clusters after training, and those clusters in which no single class dominates are labeled "unknown." Thus, during testing, confusable classes (i.e., classes with similar feature vectors) may be reported as unknown, thereby reducing the number of misclassifications.

To determine the degree to which the peak class dominates the other classes in a given cluster, we use a measure called the *cluster class contrast*. The cluster class contrast is calculated for each cluster by taking the number of entries for the peak class and subtracting the average number of entries for the remaining classes. The result is normalized by dividing by the total number of entries in the cluster. Thus the class contrast for a cluster formed from a single class will be unity, and the class contrast for a cluster in which no single class dominates will be close to zero because the difference between the peak and the mean will be very small. After training, clusters whose class-contrast measurements fail to meet a given threshold are labeled unknown. A potentially adverse effect of this technique is the labeling of all of the clusters that represent a given class as unknown by setting too high a threshold. This effect can occur particularly for low λ_{train} settings because such settings increase the likelihood that multiple classes will fall into the same cluster. These classes will always be reported as unknown during testing.

At the conclusion of training, the actual class names of the feature vectors versus the matching cluster class names are tabulated and reported in the form of a *confusion matrix* (to be discussed in the following section). The percentage of vectors that matched the correct cluster classes and the percentage that were matched incorrectly are also calculated. Finally, the number of clusters formed are reported to indicate the level of generalization that the system achieved.

The operation of the ACN during testing is similar to the operation during training. During testing, however, the clusters are not modified by the incoming

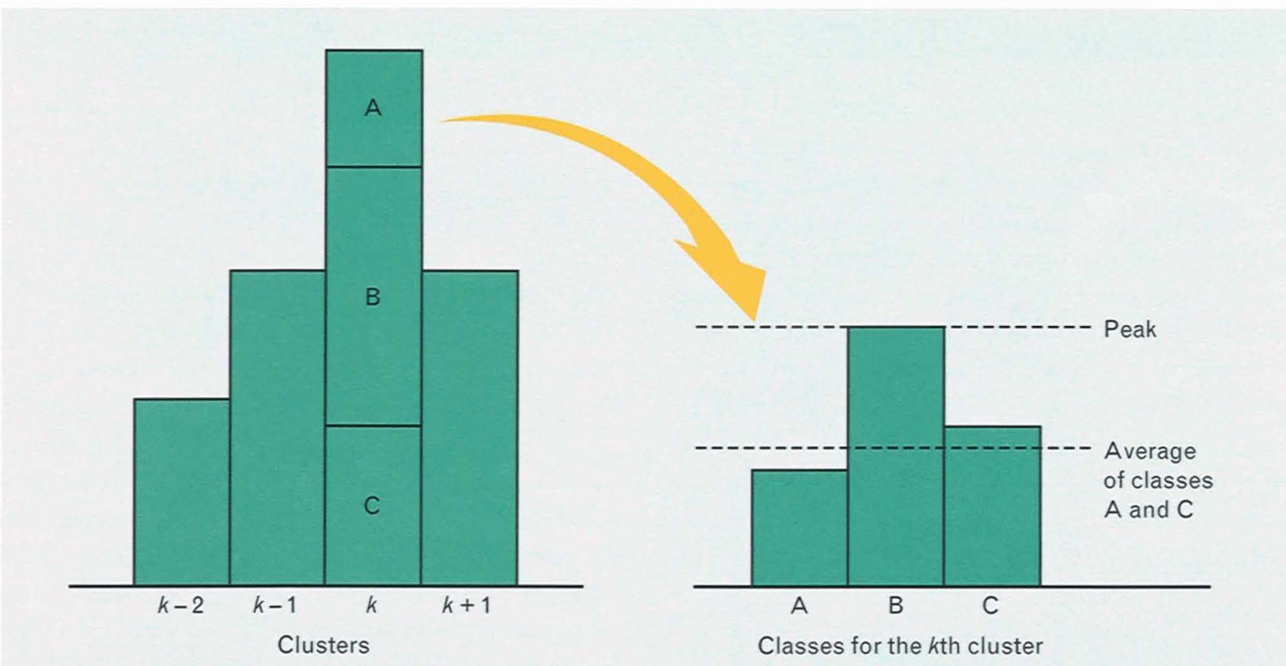


FIGURE 8. Cluster class contrast. The class with the highest frequency of occurrence in a cluster becomes the class name that is applied to that cluster. Thus for the *k*th cluster the class name B is assigned. To determine the degree to which the peak class dominates the other classes in a given cluster, we use a measure called the *cluster class contrast*, which is calculated for each cluster by taking the number of entries for the peak class and subtracting the average number of entries for the remaining classes. The result is then normalized by dividing by the total number of entries in the cluster.

feature vectors and no new clusters are formed. In the testing process, the ACN calculates the correlation between an input vector and each of the stored clusters. The clusters are then sorted by their correlation to the input, and a class histogram based on the top *k*-NN clusters is formed. The relative contribution of each cluster to the class histogram is determined by the class-contrast values that were calculated earlier during the final phase of the training process. A *k*-NN class contrast based on the peak in the class histogram is then calculated. If the calculated value exceeds a given threshold, then the input is identified as the class name associated with the peak in the histogram. Otherwise, the input is reported as unknown. Note that the case of $k = 1$ simply identifies the input as the class name of the best-match cluster. At the conclusion of testing, a confusion matrix is reported, along with the percentages correctly and incorrectly identified, and the percent classified as unknown.

Implementation

The ASC system, which has been implemented entirely in software, resides on a Sun Microsystems Sparcstation 2. A Sun VideoPix Sbus card performs the digitization of the ISAR video data.

Results Using the Hawaii Database

The performance of the ASC system on the Hawaii data has been presented in detail elsewhere [1, 2, 16]. For the purposes of this article, only a summary of the evaluation is provided. The classification performance is typically stated in terms of the percentage of frames classified correctly, the percentage classified incorrectly, and the percentage reported unknown.

Training

Using the Hawaii data, we trained the ASC system with 276 images, comprising roughly half the data. Consecutive subsets of the frames were chosen at random. The training correlation threshold λ_{train} and the learning rate γ for the ACN were set to 0.90 and 0.25, respectively. With those settings, the classifier grouped 88% of the frames correctly, and a total of 21 clusters were formed, representing a 13:1 reduction of the training data. The confusion matrix in Table 3 summarizes the results. (Note: The horizontal rows in the table represent the actual classes of the targets while the vertical columns contain the classes reported by the system. Thus, for the ideal case in which the system classifies every target correctly, all of the off-diagonal entries would be zero.) All four examples of Class 1 were grouped incorrectly with Class 5 because of the small number of frames for Class 1. Table 3 also shows that many Class 2 frames were grouped incorrectly with Class 6. These errors occurred because the target orientations at which these two classes were imaged caused the peaks in the feature vectors to coincide and the measured lengths to be roughly the same. A slightly higher training correlation threshold λ_{train} would have eliminated this confusion, but it would have also resulted in the creation of a significantly higher number of clusters.

Testing

The remaining 226 images were used to test the trained system. At a testing correlation threshold λ_{test} of 0.75, the system classified 95% of the images correctly and 5% incorrectly, and 0% were reported as unknown. The confusion matrix in Table 4 shows that five examples of Class 3 were misclassified as Class 6. The errors occurred because the two classes have similar characteristics that produced similar imagery under certain circumstances.

Results Using the IBM Database

In the IBM imagery, there were many instances in which the target faded into the noise background due to drift in the sensor's automatic gain control (AGC). There were also instances in which most of a target shifted off the edge of an image. Both of these situations resulted in unrecognizable targets. Because these unusable frames occurred frequently in the data, an automated frame-rejection capability was necessary to eliminate the labor-intensive need for editing the data manually, as well as to enable the ASC system to operate autonomously.

We incorporated a simple algorithm into the ASC preprocessor to reject the unusable frames. Ideally, an image should contain a region of high-intensity pixels that form the target, and a low-intensity background.

Table 3. Confusion Matrix for Hawaii Database: Training

	Class Reported by System					
Actual class	*1*	*2*	*3*	*4*	*5*	*6*
1	0	0	0	0	4	0
2	0	81	0	0	0	29
3	0	0	21	0	0	0
4	0	0	0	18	0	0
5	0	0	0	0	65	0
6	0	0	0	0	0	58

To find the target region, the algorithm first samples an image into overlapping windows and sums the pixel intensities contained in each window. The window with the highest sum is assumed to contain the target, and the average of the remaining windows is assumed to be indicative of the background level. Thus subtracting the average of the window sums from the highest window sum provides a measure of the target strength over the background noise level. If an image does not contain a target, then the difference between the highest sum and the average sum will be very small. The difference will also be small for images containing faint targets and high levels of background noise. By applying a threshold to the normalized value of this difference between the highest sum and the average sum, the frame-rejection algorithm was able to remove the majority of the unusable frames encountered in the IBM database, reducing the total number of resulting feature vectors from 12,814 to 10,935.

Training

The performance obtained with a neural-network-based automatic recognition system is highly dependent on the examples used to train the system. The training data should encompass as many variations of the targets as possible so that the system will be able to recognize the targets in most circumstances. On the other hand, the number of clusters formed should be kept to a minimum by including only those training examples which contain complete and clear representations of the targets.

To provide a controlled dataset for training, we selected the best frames in the IBM database according to the following criteria: good signal-to-background ratio, a complete target with bow and stern present, an image that characterizes the target class well, and frames that contain a single target. Applying these criteria to the IBM database yielded 4487 selected frames, i.e., 35% of the entire database of 12,814 frames. We refer to this dataset of 4487 frames as the selected-frame dataset.

Neural network ASC systems for ISAR imagery are commonly trained on every other frame in the data to accommodate the wide range of variations in the imagery [6, 10]. Thus roughly half of the selected-frame dataset was used for training, leaving the remaining frames for testing. The alternate-frame training set consisted of 2381 images. The training correlation threshold λ_{train} and learning rate γ were set such that each class was represented by at least one cluster (0.98 and 0.25, respectively), resulting in the formation of 1015 clusters. During training, 94% of the frames were grouped into clusters with the correct class name, and the remaining 6% matched incorrect clusters. Table 5 contains the confusion matrix.

The coarse target-orientation information available in the ground truth for the IBM database (mentioned earlier) allowed us to evaluate the effects of

Table 4. Confusion Matrix for Hawaii Database: Testing

	Class Reported by System					
Actual class	1	2	3	4	5	6
1	0	0	0	0	1	2
2	0	94	1	0	0	0
3	0	0	9	0	0	5
4	0	1	0	12	0	0
5	0	0	0	0	61	0
6	0	0	1	0	0	39

target-orientation cueing on system performance. When a target is imaged at an oblique orientation, a foreshortening of the target occurs and details of the target can become masked. Thus the imagery of a large ship at an oblique orientation can be similar to the imagery of a small ship at an end-on orientation. Because of such effects, we expected that the use of target-orientation cueing would lead to performance gains. When orientation information was manually incorporated into training (discussed in the section "Target-Orientation Cueing"), the number of frames that were grouped correctly increased to 98%, with 1062 clusters formed.

We also investigated the effects of applying a threshold to the cluster class contrast. By applying a threshold of 0.90 to the cluster class contrast, we found that the class names of 39 clusters were changed to "unknown," which helped to decrease the number of incorrect classifications made during testing. Only 39 clusters (3.8% of the total of 1015 clusters) were affected by the cluster class contrast because of the relatively high setting of the training correlation threshold λ_{train}. The majority of classes were spread across many clusters.

Testing

Testing of the various configurations of the ASC system was performed with two datasets, representing the two modes of system operation. For the operator-assistant mode, in which the operator chooses the frames to be classified by the ASC system, the remaining alternate frames from the selected-frame dataset were used to evaluate the system. Thus this dataset, which we refer to as the selected-frame testing dataset, consisted of 4487 − 2381 = 2106 frames. For the automatic mode, in which the system operates autonomously over all of the data, the dataset consisted of the all-frame database minus the training dataset. Thus this dataset, which we refer to as the all-frame testing dataset, consisted of 12,814 − 2381 = 10,433 frames. Note that we purposely did not include in either of the testing datasets any of the frames that were used for training. For all of the test cases, the correlation threshold λ_{test} was set to 0.90, unless otherwise noted.

Single-Frame Testing

When evaluated with the selected-frame testing dataset (2106 frames), the ASC system classified 87% of the frames correctly and 12% incorrectly, and 1% were reported as unknown. The confusion matrix in Table 6 shows that Class 9, which had the largest number of frames, was responsible for the majority of errors. This result was expected because the Class 9 ships were imaged at a greater range than the other vessels. Thus the Class 9 imagery contained more noise and less-well-defined targets. When the all-frame testing dataset (10,433 frames) was used, the ASC system classified 62% of the frames correctly and 31% incorrectly, and 7% were reported as unknown.

Table 5. Confusion Matrix for Selected-Frame Dataset from IBM Database: Training

	Class Reported by System											
Actual class	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
1	194	0	0	0	0	4	0	0	3	0	1	0
2	2	23	1	0	0	0	0	0	1	0	0	0
3	6	0	160	0	0	0	0	0	9	2	0	0
4	0	0	0	142	0	0	0	0	0	1	0	0
5	0	2	0	0	377	0	0	0	0	1	0	0
6	1	0	0	0	1	106	0	1	22	0	0	0
7	0	0	0	0	0	9	30	0	0	0	0	0
8	0	0	1	1	0	0	0	136	14	0	0	0
9	5	2	3	3	1	8	3	6	571	1	3	0
10	2	0	0	0	2	0	0	0	5	199	1	0
11	1	0	0	0	0	4	0	0	6	0	30	0
12	0	0	0	0	0	0	0	1	1	0	0	272

Table 6. Confusion Matrix for Selected-Frame Dataset from IBM Database: Testing

	Class Reported by System											
Actual class	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
1	135	0	0	1	6	5	2	2	10	5	1	1
2	3	19	2	1	0	0	0	1	1	0	0	0
3	3	0	143	1	1	0	1	2	4	1	1	0
4	0	0	0	113	0	0	0	1	0	0	0	0
5	2	3	4	1	323	0	0	2	12	2	0	0
6	9	0	1	0	2	93	1	1	8	1	2	0
7	0	0	0	0	0	8	22	0	2	0	0	0
8	6	1	2	0	1	0	0	111	10	2	2	0
9	6	7	8	4	3	20	2	1	461	6	3	0
10	1	0	1	0	7	0	0	1	8	162	3	2
11	1	0	0	0	0	3	2	0	8	0	19	1
12	0	0	0	1	0	0	0	1	3	6	0	232

If the correlation threshold λ_{test} had been set lower than the 0.90 value used, some of the frames that were reported as unknown might have instead been classified incorrectly, thus resulting in a higher percentage of incorrect classifications.

Multiple-Frame Processing

To date, ASC systems for ISAR imagery have treated each individual frame in the data as a separate event. Typically, however, an operator classifies the entire observation of a target, not a single snapshot.

Storage of the individual observations in the database into separate disk files allowed the ASC system to classify entire observations as well as the individual frames. Multiple-frame classification was accomplished by forming a histogram of the reported classes for each frame in an observation. Figure 9 shows the class histogram for Observation 2 of a Class 1 ship. The peak in the class histogram indicates the most frequently reported class during the observation. This multiple-frame technique leads to a significant performance increase because it reduces spurious single-frame errors caused by image variability.

Using multiple-frame processing to evaluate the selected-frame testing dataset, the system achieved a 95% correct classification rate, with 5% incorrect and 0% reported unknown. For the all-frame testing dataset, multiple-frame performance led to an 83% correct classification rate, with 14% incorrect and 3% reported unknown.

Cluster Class Contrast

We also investigated the effect of applying a threshold to the cluster class contrast. Figure 10 shows how different threshold settings affects the multiple-frame performance of the system for the all-frame testing dataset. Note that the cluster-class-contrast threshold removes some of the misclassifications at the expense of lowered correct-classification rates.

Observation Class Contrast

The class contrast, which we used to reduce misclassifications caused by confusable clusters, can also be applied to the multiple-frame class histograms to reduce misclassifications of confusable observations.

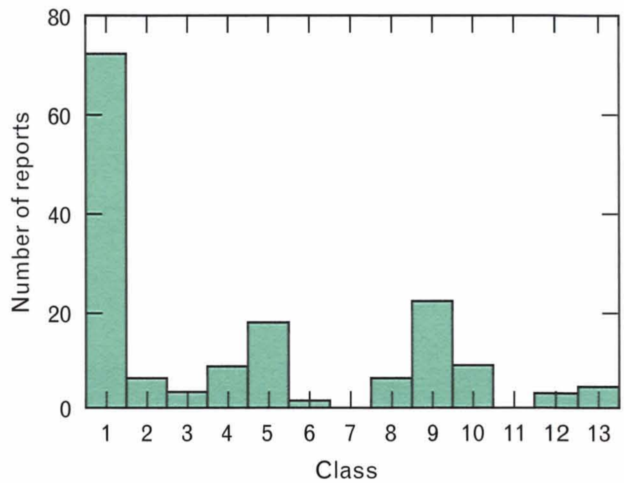


FIGURE 9. Example of class histogram using multiple-frame classification. The histogram is for Observation 2 (a Class 1 ship) from the IBM database. (Note: Class 13 represents the unknown class.)

In the process, observations in which no single class forms a distinct peak in the class histogram are labeled unknown.

Figure 11 shows the effect of applying different thresholds to the observation class contrast for the multiple-frame tests. For the selected-frame data, Figure 11(a) shows that this technique eliminates all misclassified observations at a threshold of 0.5, with a resulting performance of 81% correct and 19% unknown. For the all-frame data, Figure 11(b) shows that at the same threshold level the system achieves a performance level of 62% correct and

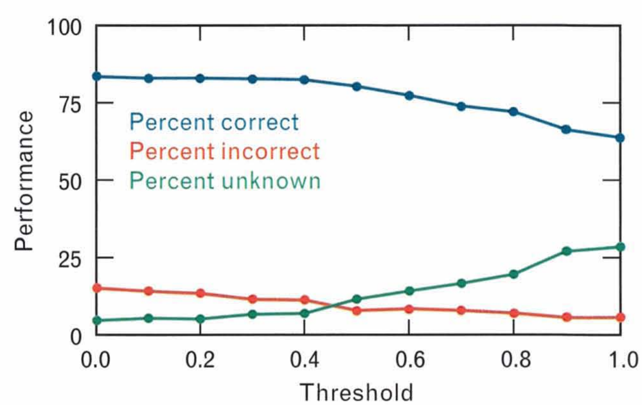


FIGURE 10. Effect of applying different thresholds to the cluster class contrast for multiple-frame performance on the all-frame testing dataset.

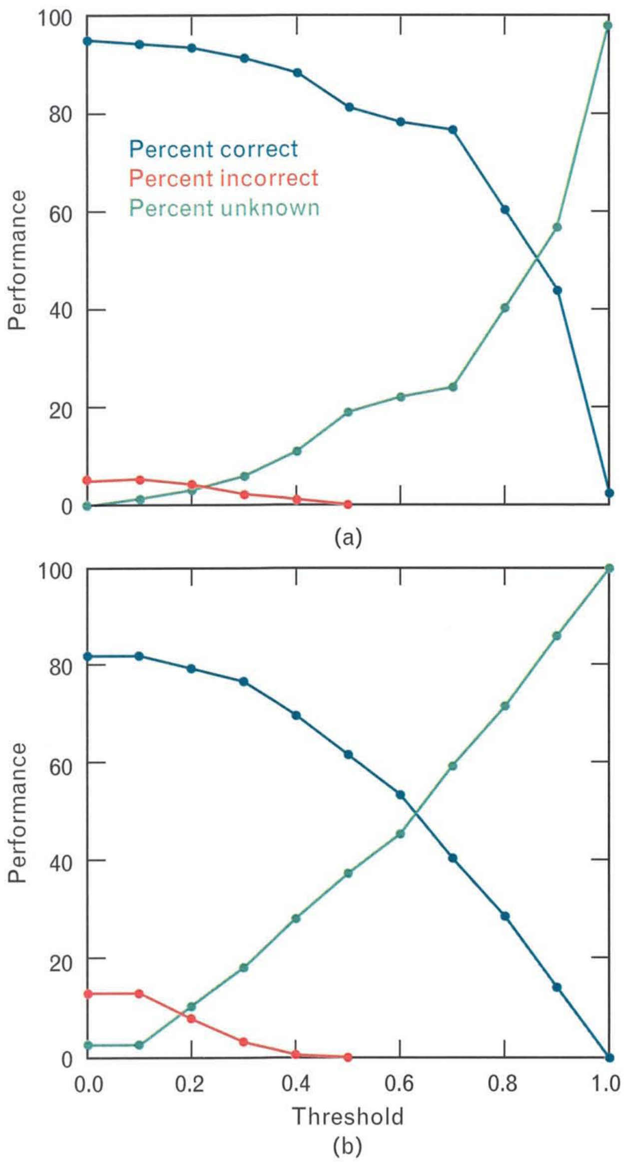


FIGURE 11. Effect of applying different thresholds to the observation class contrast for the (a) selected-frame testing dataset and the (b) all-frame testing dataset. The thresholds were applied to the multiple-frame class histograms to reduce misclassifications of confusable observations.

38% unknown with no misclassifications.

Target-Orientation Cueing

To investigate the effects of target-orientation cueing, we trained the system on all ships at given target orientations (bow, oblique bow, stern, oblique stern, and beam); thus the system formed separate ACN memories for each of the five different orientations. During testing the orientation of the target was utilized to select the appropriate ACN memory.

The use of target-orientation information for both training and testing with the selected-frame dataset improved the correct-classification rate to 100%. In comparison, without the orientation information the correct-classification rate was 95%, with 5% incorrect and 0% unknown.

The use of target-orientation information with the all-frame testing dataset resulted in an overall correct classification rate of 83%, with 11% incorrect and 6% unknown. For the stern observations, 100% correct classification was obtained because this orientation produced highly defined features. The performance for the bow orientation was lower (84% correct, 10% incorrect, and 6% unknown) because the bow observations contained a large number of frames with high noise, incomplete targets, and/or poor sensor stabilization. The lower performance for the oblique bow orientation (74% correct, 21% incorrect, and 5% unknown) and oblique stern orientation (84% correct, 8% incorrect, and 8% unknown) can be attributed to the coarse coding of orientation labels over a wide range of actual target orientations, which varied from near bow/stern to near beam. The lower performance for the beam orientation (82% correct, 9% incorrect, and 9% unknown) was expected because beam imagery is generally not well resolved in range. Application of a threshold to the class contrast for each observation eliminated all misclassifications and yielded an overall performance of 71% correct and 29% unknown.

Cluster Histogram

Another method of using multiple frames to identify an observation is to form a histogram of the best matching clusters (as opposed to the best matching classes) for each frame in the observation. The class name of the cluster with the peak number of entries for each observation is then reported. Applying this method to the all-frame testing dataset without the use of target-orientation information resulted in 80% correct classifications, with 20% incorrect and 0% unknown. As discussed earlier in the subsection "Multiple-Frame Processing," the class histogram method yielded 83% correct classifications, with 14% incorrect and 3% unknown. The performance of the cluster histogram was inferior to that of the class histogram because the correct class for a given observation may be distributed across several clusters, thus resulting in the lack of a distinct peak in the cluster histogram.

Correlation Threshold

During testing, the correlation value for the cluster that best matches the input feature vector is compared to the correlation threshold λ_{test}. When the correlation value exceeds the threshold, the class name of the cluster is reported; otherwise the input is labeled unknown. For the all-frame testing dataset, Figure 12 shows that the threshold has a very slight effect on the multiple-frame performance of the system except at very high values. For λ_{test} greater than 0.9, the correct and incorrect classification rates decrease sharply. It is important to note that the removal of all misclassifications by increasing the correlation threshold also drastically reduces the number of correct classifications. Thus the observation-based class-contrast technique is a more effective means of reducing or eliminating misclassifications than the correlation threshold.

Independent Observation

As discussed earlier, the alternate-frame method of selecting training and testing datasets is frequently used for the laboratory evaluation of ASC systems. All of the performance reported thus far has made use of this technique to provide results that could be compared with the efforts of other researchers. For a deployable ASC system, however, the training and testing methodology will be different. Presumably, data will be gathered for the specific purpose of initializing the ASC system so that the system can be trained on all of the frames from a fixed set of observations of each ship class to be recognized. In operation, the system will encounter new observations of previously seen ship classes, as well as unseen ship classes. Therefore, we tested the ASC system on observations that were independent of the training data to provide results that were more operationally relevant than the alternate-frame results.

To assess how the ASC system would perform on independent observations, we subgrouped the bow- and stern-orientation observations in the IBM database into new training and testing subsets. The observations at oblique bow and oblique stern orientations were not used because they contained too wide a range of actual target orientations. The observations at beam orientation were also omitted because there were too few of them for a meaningful analysis. For both the bow- and stern-orientation observations, we subgrouped the data such that some of the ship classes had entries for both training and testing (constituting the seen observations for testing), while others had

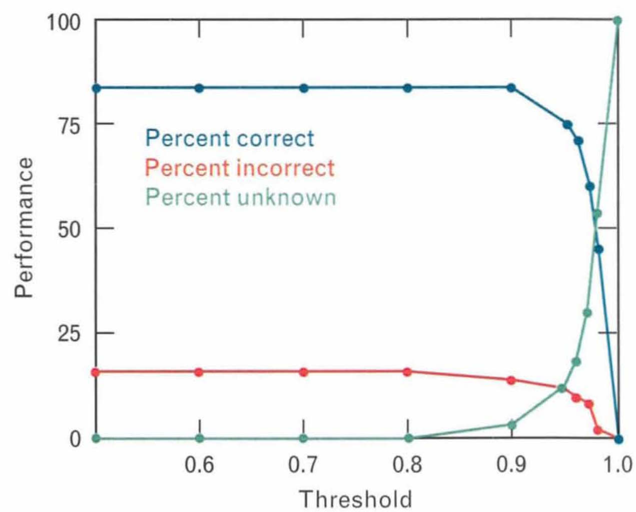


FIGURE 12. Effect of the correlation threshold λ_{test} on the multiple-frame performance of the system using the all-frame testing dataset.

only testing entries (constituting the unseen observations). We attempted to achieve the desired system response—that unseen observations be classified as unknown—by increasing the testing correlation threshold λ_{test} from 0.90 to 0.95.

For the bow-orientation case, the system was trained on a total of seven observations, which resulted in the formation of 176 clusters. Table 7 shows the results when the system was subsequently tested on the remaining 21 seen observations and three unseen ones. The increased correlation threshold (from 0.90 to 0.95) reduced both the number of correct and incorrect classifications from the previously reported levels.

Table 7. System Performance Using Independent Observations

	Seen Observations			Unseen Observations	
	Correct	*Incorrect*	*Unknown*	*Incorrect*	*Unknown*
Bow	13 (62%)	3 (14%)	5 (24%)	1 (33%)	2 (67%)
Stern	4 (67%)	0	2 (33%)	1 (17%)	5 (83%)
Overall	17 (63%)	3 (11%)	7 (26%)	2 (22%)	7 (78%)

Table 8. System Performance Using Independent Observations after the Elimination of Outlier Clusters

	Seen Observations			Unseen Observations	
	Correct	*Incorrect*	*Unknown*	*Incorrect*	*Unknown*
Bow	12 (57%)	1 (5%)	8 (38%)	0	3 (100%)
Stern	4 (67%)	0	2 (33%)	1 (17%)	5 (83%)
Overall	16 (59%)	1 (4%)	10 (37%)	1 (11%)	8 (89%)

For the unseen bow-orientation observations, there was one incorrect classification—a Class 6 ship was misclassified as Class 10. This error occurred because many of the frames from that one unseen observation matched an outlier cluster that had been formed from a single spurious frame from Class 10. The number of frames (from the one unseen Class 6 observation) that matched the outlier cluster was sufficient to produce a high class-contrast value for the observation, so that the observation-class-contrast threshold was not able to remove the misclassification without sacrificing most of the correct classifications. Table 8 shows that the elimination of clusters formed from single spurious frames removes that one misclassification of an unseen observation and reduces both the number of correctly and incorrectly classified seen observations. The removal of these single-entry outlier clusters resulted in 41 valid clusters for the bow orientation.

For the stern-orientation case, the system was trained on a total of five observations, which resulted in the formation of 106 clusters. In Table 7, the one incorrect classification of an unseen stern-orientation observation resulted from a Class 11 ship being misclassified as Class 6. Because of the very small sizes of the ships in both of these classes and because of the low resolution of the sensor used, the number of pixels for the ships in the two classes was small. As a result, the system had difficulty distinguishing between these two ship classes. We could not eliminate the misclassification by removing the single-entry outlier clusters because the one misclassified unseen observation matched clusters that were formed from large numbers of frames. All of the seen stern-orientation observations were classified correctly, and the system performance for these observations was also unaffected by the elimination of outlier clusters, as shown in Table 8. The removal of single-entry outlier clusters resulted in 30 valid clusters.

These preliminary results indicate that the system correctly reports as unknown most observations of classes that it has not seen before. But the system's performance in the independent-observation tests was lower than that in the alternate-frame test because, in the independent-observation experiment, the system was trained on only one observation per class per target orientation. (Note: In cases in which there was only one observation per class per a particular target orientation, the sole observation was used for testing.) The single training observation probably failed to provide examples of all the variations found in the testing imagery. Thus a database with a significantly higher number of observations per class is required to attain statistically significant independent-observation performance.

Summary and Conclusions

The Automatic Ship Classification (ASC) system has undergone substantial improvements and extensive evaluation. The latest testing was conducted on a database containing 12 classes of ships. The use of multiple frames of imagery obtained during observation intervals of the target ships significantly improved the system's performance. The incorporation of target-orientation information and the class-contrast technique also reduced misclassifications. When selected frames of the data were used in the training and testing, the improved system classified 100% of the targets correctly. When all of the frames were used, the system classified 71% of the targets correctly, with the remaining 29% reported as unknown; i.e., no targets were classified incorrectly.

The results using the selected-frame testing dataset show that the system would perform very well as an operator assistant. In this scenario, the operator would select the frames presented to the ASC system. The all-frame testing results show that several observations of a target would probably be required to obtain a confident identification during autonomous operation of the system.

From these results, we feel that the ASC system is now ready for a thorough field evaluation on board an ISAR sensor platform. For such an evaluation, digital data from the actual ISAR sensor should be provided for the initial training and subsequent fine tuning of the system.

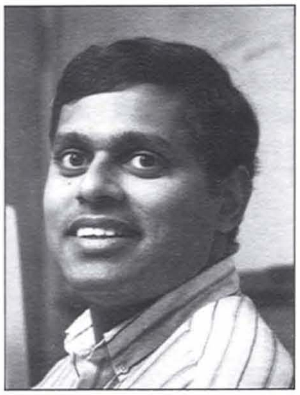
Acknowledgments

This research was performed with the valuable assistance of several staff members from the Opto-Radar Systems Group at Lincoln Laboratory. Lorraine Prior modified the digitization software to allow for continuous digitization of the video data. Vivian Titus digitized the video imagery, and she and Lionel Labrecque provided the data-editing expertise.

This work was sponsored by the Space and Naval Warfare Systems Command Office.

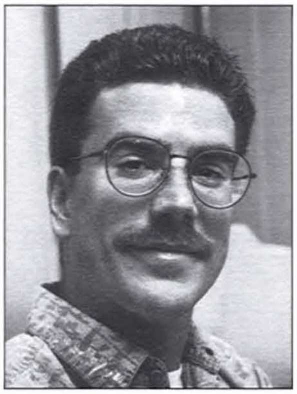
REFERENCES

1. M.M. Menon et al., private communication.
2. E.R. Boudreau, private communication.
3. E.R. Boudreau and M.M. Menon, private communication.
4. S.A. Musman, private communication.
5. C. Chu, private communication.
6. L.E. Berman and A. Schultz, private communication.
7. M.M. Menon and E.R. Boudreau, "Pattern Recognition System with Statistical Classification," U.S. Patent pending.
8. G.A. Carpenter and S. Grossberg, "Brain Structure, Learning, and Memory," *1985 AAS Symposium Series.*
9. D.L. Reilly, L.N. Cooper, C. Elbaum, "A Neural Model Category Learning," *Biol. Cybern.* **45**, 35 (1982).
10. T.M. McKannon, private communication.
11. J.R. Loyer, private communication.
12. J.M. Tierney, private communication.
13. R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis* (John Wiley, New York, 1973).
14. K. Fukunaga, *Introduction to Statistical Pattern Recognition* (Academic Press, Orlando, FL, 1972).
15. T. Kohonen, "An Introduction to Neural Computing," *Neural Networks* **1**, 3 (1988).
16. E.R. Boudreau and M.M. Menon, private communication.

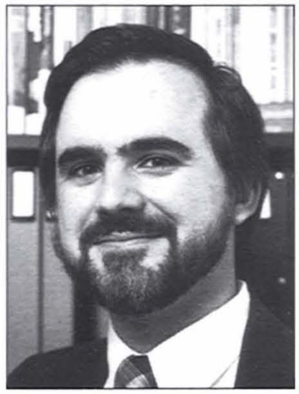


MURALI M. MENON
is currently a research staff member in the Opto-Radar Systems Group. He received a B.S., an M.S., and a Ph.D. degree in chemical engineering from Case Western Reserve University. His research interests include applied pattern recognition, signal processing, and image processing, with special interests in wavelets and artificial neural networks.

In 1987 Murali joined Lincoln Laboratory, where he has worked on applications of neural networks for processing sensor data, including the design of automatic target recognition (ATR) systems. He is currently involved with the transfer of technology to the commercial sector, and has established a cooperative research and development agreement with a company in the Boston area to develop an automated quality control system based on neural networks and advanced signal processing methods. In 1990 he chaired a workshop on the applications of neural networks to real-world machine-vision problems at the IEEE Neural Information Processing Systems (NIPS) conference. He is in the process of writing a book chapter entitled "The Neocognitron: Prospects for Scaling to Practical Applications," for *Progress in Neural Networks* **II**, to be published in 1994. Murali is a member of the IEEE.



ERIC R. BOUDREAU
received a B.S. degree in electrical engineering from Worcester Polytechnic Institute, and worked at COMPDATA Services Corp. before joining Lincoln Laboratory four years ago. He is currently an assistant staff member in the Opto-Radar Systems Group, where his focus of research has been on the development and evaluation of automatic target recognition (ATR) systems. He has worked on the detection of stationary targets in laser-radar and passive-infrared imagery, and recently participated in the implementation of a portable global surveillance demonstration system. Eric is a member of Eta Kappa Nu.



PAUL J. KOLODZY
is assistant leader of the Opto-Radar Systems Group. He received a B.S. degree from Purdue University, and M.S. and Ph.D. degrees from Case Western Reserve University, all in chemical engineering. His current areas of research include neuromorphic systems, automatic target recognition (ATR) systems, and advanced distributed simulation technology. In 1983, he was a visiting scientist working in the area of laser remote measurement systems at Risø National Laboratory in Roskilde, Denmark. He was cochairman of the Simulation/Emulation Tools & Techniques Panel of the ARPA Neural Network National Study, and he is a member of the Sensor Fusion program committee. Paul has been at Lincoln Laboratory since 1986.