This CVPR2015 workshop paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Sonar Automatic Target Recognition for Underwater UXO Remediation

Jason C. Isaacs
Naval Surface Warfare Center, Panama City, FL USA
jason.c.isaacs1@navy.mil

Abstract

Automatic target recognition (ATR) for unexploded ordnance (UXO) detection and classification using sonar data of opportunity from open oceans survey sites is an open research area. The goal here is to develop ATR spanning real-aperture and synthetic aperture sonar imagery. The classical paradigm of anomaly detection in images breaks down in cluttered and noisy environments. In this work we present an upgraded and ultimately more robust approach to object detection and classification in image sensor data. In this approach, object detection is performed using an in-situ weighted highlight-shadow detector; features are generated on geometric moments in the imaging domain; and finally, classification is performed using an Ada-boosted decision tree classifier. These techniques are demonstrated on simulated real aperture sonar data with varying noise levels.

1. Introduction

The detection and classification of undersea objects is considerably more cost and risk effective and efficient if it can be performed by Autonomous Underwater Vehicles (AUVs) [16]. Therefore, the ability of an AUV to detect, classify, and identify the targets is of genuine interest to the Navy. Targets of interest in sonar and optical imagery vary in appearance, e.g., intensity and geometry. It is necessary to formulate a general definition for these objects which can be used to detect arbitrary target-like objects in imagery collected by various sensors. We can define these objects as man-made with some inorganic geometry, which has coherent structure, and whose intensity may be very close to that of the background given the potential time lag between deployment and inspection.

1.1. Objectives

This work presents methods for the automated detection and classification of targets in cluttered and noisy sensor data. Prior work in related areas is known in the mine-counter-measures imaging domain [3, 7, 13, 15] but not as well in the non-imaging domain [12, 9]. Some of these techniques which are used in the algorithm are well known in the literature[1, 18]; however, some of the features used to classify the most statistically significant targets for the UXO ATR problem are introduced here.

1.2. Outline

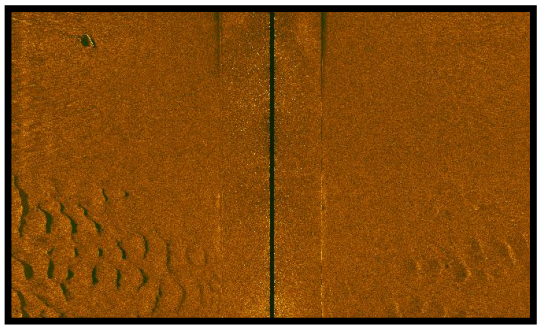
First, we will describe sonar imagery in general and the simulated sonar imagery used here. Then we will describe the detection of targets, continue with methods used to analyze objects, i.e. feature extraction, establish criteria for classifying these objects, and discuss a way forward.

1.3. Sonar Imagery

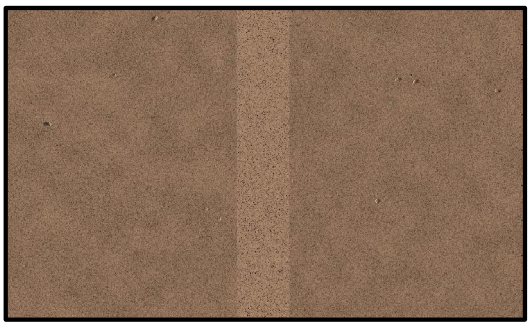
Sound navigation and ranging (SONAR) was developed in WWII to aid in the detection of submarines and sea-mines, prior sound ranging technology was used to detect icebergs in the early 1900s. Today sonar is still used for those purposes but now includes environmental mapping and fish-finding. Side-looking or side-scanning sonar is a category of sonar system that is used to efficiently create an image of large areas of the sea floor. It may be used to conduct surveys for environmental monitoring or underwater archeology. Side-scan sonar has been used to detect objects and map bottom characteristics for seabed segmentation [2] and provides size, shape, and texture features [8]. This information can allow for the determination of the length, width, and height of objects. The intensity of the sonar return is influenced by the the objects characteristics and by the slope and surface roughness of an object. Strong returns are brighter and darkness in a sonar image can represent either an area obscured by an object or the absorption of the sound energy, e.g. a bio-fouled object will scatter less sound. Sonar system performance can be measured in many ways, e.g. geometrical resolution, contrast resolution, and sensitivity, to name a few. An example of real aperture sonar images is shown in Figure 1 for an 850 kHz Edgetech sonar on the top and an 230 kHz simulated sonar image on the bottom.

Synthetic aperture sonar [5] (SAS), is similar to synthetic aperture radar in that the aperture is formed artificially from

received signals to give the appearance of a real aperture several times the size of the transmit/receive pair. SAS is performed by collecting a set of time domain signals and match filtering the signals to eliminate any coherence with the transmitted pulse. SAS images are generated by beamforming the time domain signals using various techniques, e.g. time-delay, chirp scaling, or ω-k beamforming [5]. Beamforming is the process of focusing the acoustic signal in a specific direction by performing spatio-temporal filtering. This allows us to take a received collection of sonar pings and transform the time series into images. The goal of



(a) Edgetech 850 kHz Image



(b) Simulated 230 kHz Image

Figure 1. Example real aperture sonar images.

ATR, here, is to classify specific UXO from within groups of objects that have been detected in sonar imagery, see Figure 2. As shown, one can see that the objects of interest exhibit strong highlights with varying shadows depths. These edges are not necessarily unique to objects of interest, however, a similar response is demonstrated by sea-floor ripples and background clutter.

1.4. Object Detection

The purpose of the detection stage is to investigate the entire image and identify candidate regions that will be more thoroughly analyzed by the subsequent feature extraction and classification stages. This is a computationally intensive stage because a target region surrounding each image pixel must be evaluated. Therefore, the goal is to keep the computations involved with each region small. The goal of detection is to screen out the background/clutter regions in the image and therefore reduce the amount of data that must be processed in the feature extraction and classification stage. The detector used here is one that inspects the image in two separate ways. First, the probability distribution function (pdf) of the normalized intensity image $I(x, y)$ is solved for in order to set the threshold levels for the shadow and highlight regions, T_S and T_H respectively. For more on the pdf (first-order histogram), see the next section on feature extraction.

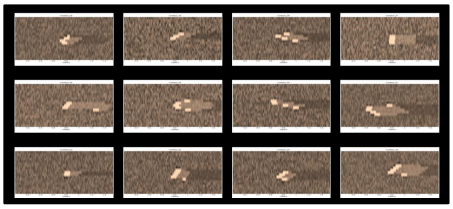


Figure 2. Example simulated target snippets.

Once these levels are set then then two separate images are thresholded at the two values. Anything that meets these limits is then analyzed further for regional continuity. This continuity is determined quickly by convolving the regions, with in some neighborhood, with a Gaussian mask for computational efficiency resulting in two separate matrices X_S and X_H representing the shadow highlight regions of interest respectively. The Gaussian mask size can be set based on expected object size. The mask acts to weight areas more highly that are similar, e.g. if two high intensity pixels are near each other the more likely it is that they correspond to the same object. After the masking operations a weighted combination of the two locality matrices X_S and X_H is evaluated for target criteria as follows

$$X_I = (X_S \wedge X_H) \vee \omega_s(X_S > T_{SL}) \vee \omega_h(X_H > T_{HL})), \quad (1)$$

where ω_s and ω_h are configurable weights on the importance of the shadow and highlight information, derived from *a priori* target information. The threshold values T_{SL} and T_{HL} are set dynamically based on *a priori* clutter and environmental information. Any location (x, y) that meets a global threshold T_I is then passed on to the feature extraction and classifier stages to be analyzed further. The detection algorithm is shown in Figure 3 and an example of the detection steps is demonstrated in Figure 4. This analysis involves extracting a ROI, Figure 4, about (x, y) that meets predetermined size criteria, e.g *a priori* target knowledge of 1.5m spheres would determine then a fixed ROI of 3m square based on training for that target. However, if no prior knowledge is provided then a general ROI is considered and

fixed at 2m square.

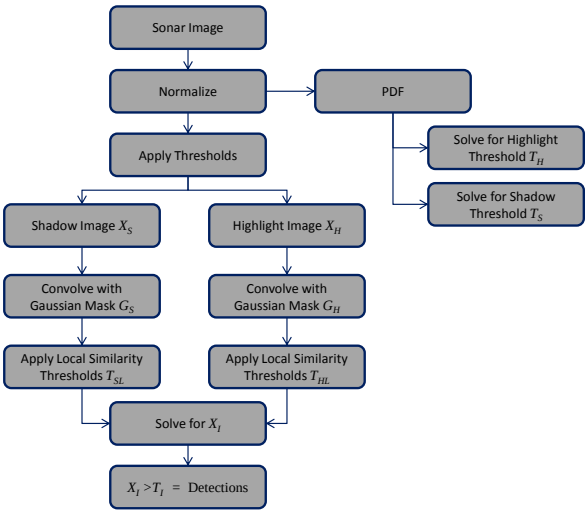


Figure 3. HLS detector process from a sonar image to a detection list.

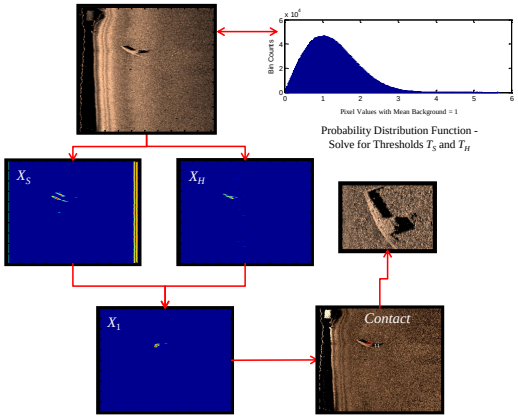


Figure 4. HLS detector process from a sonar image to a detection ROI.

2. Feature Generation

There are many different features to choose from when analysis or characterizing an image region of interest, see [1, 18] . In this work we will focus on generating two sets of features, one based on statistical models of pixel distributions and the other on the response of targets to spatial filter configurations. The statistical models are descriptors that attempt to represent the texture information utilizing the intensity distribution of the area. The spatial filters measure the response of an area of interest and how it is changed changed by a function of the intensities of pixels in a small neighborhoods within this area of interest.

2.1. Statistical Models of Pixel Distributions

Geometric distribution based moment and order statistic features have been in use for image analysis since the 1960s [6] and has been prominent in digital image analysis through the years [17]. There are many geometric moment generating methods [14], we will focus on the use of two types of geometric moments: Zernike moments [10] and Hu moments [6]. The order statistics [18] methods will be derived from the probability distribution function and co-occurrence matrix of the image. The moments are well known as feature descriptors for optical image processing. However, they have been employed in the sonar image processing domain in recent years []. To better understand these features we will begin with a description of the probability distribution of the intensities within a sonar region of interest. Whereby the image intensity I is the magnitude of the signal of a RAS sonar image. The distribution of pixels is represented as $P(I)$.

2.1.1 First-Order Statistics Features

Given a random variable I of pixel values in an image region, we define the first-order histogram $P(I)$ as

$$P(I) = \frac{\textit{number of pixels with gray level } I}{\textit{total number of pixels in the region}}. \quad (2)$$

That is, $P(I)$ is the fraction of pixels with gray level I. Let N_g be the number of possible gray levels. Based on 2 the following moment generating functions are defined.

Moments:

$$m_i = E[I^i] = \sum_{I=0}^{N_g-1} I^i P(I), \quad i = 1, 2, \ldots \quad (3)$$

where $m_0 = 1$ and $m_1 = E[I]$, the mean value of I.

Central moments:

$$\mu_i = E[(I - E[I])^i] = \sum_{I=0}^{N_g-1} (I - m_1)^i P(I). \quad (4)$$

The most frequently used moments are variance, skewness, and kurtosis, however higher order moments are also utilized. In addition to the moment features the entropy of the distribution can also provide some insight into I. Entropy, here, represents a measure of the uniformity of the histogram. The entropy H is calculated as follows

$$H = -E[\log_2 P(I)] = -\sum_{I=0}^{N_g-1} P(I)\log_2 P(I). \quad (5)$$

The closer I is to the uniform distribution, the higher the value of H.

Given $I(x, y)$ a continuous image function. Its geometric moment of order $p + q$ is defined as

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q I(x, y) dx dy \quad (6)$$

If we define the central moments as

$$\mu_{pq} = \int \int I(x, y)(x - \bar{x})^p (y - \bar{y})^q dx dy \quad (7)$$

where $\bar{x} = \frac{m_{10}}{m_{00}}$ and $\bar{y} = \frac{m_{01}}{m_{00}}$. We then define the normalized central moments as

$$\eta_{pq} = \frac{mu_{pq}}{mu_{00}^{\gamma}}, \gamma = \frac{p + q + 2}{2} \quad (8)$$

2.1.2 Hu Moments

The seven Hu moments, developed in 1962 by Hu [6], are rotational, translational, and scale invariant descriptors that represent information about the distribution of pixels residing within the image area of interest. Using 8 we can construct the Hu moments $\phi_i, i = 1, \cdots, 7$ as follows

For $p + q = 2$

$\phi_1 = \eta_{20} + \eta_{02}, \quad \phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2.$

For $p + q = 3$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2,$$
$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2,$$
$$\phi_5 = (\eta_{30} - 3\eta_{12}) + (\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (\eta_{03} - 3\eta_{21}) + (\eta_{03} + \eta_{21})[(\eta_{03} + \eta_{21})^2 - 3(\eta_{12} + \eta_{30})^2],$$
$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} - \eta_{12})(\eta_{03} + \eta_{21}),$$
$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (\eta_{30} - 3\eta_{12}) + (\eta_{21} + \eta_{03})[(\eta_{03} + \eta_{21})^2 - 3(\eta_{30} + \eta_{12})^2].$$

The first six moments are invariant under reflection while ϕ_7 changes sign. For feature calculations we will use the $log(|\phi_i|)$. We must note that these moments are only approximately invariant and can vary with sampling rates and dynamic range changes.

2.1.3 Zernike Moments

Zernike moments can represent the properties of an image with no redundancy or overlap of information between the moments [10]. Zernike moments are significantly dependent on the scaling and translation of the object in an ROI. Nevertheless, their magnitudes are independent of the rotation angle of the object [18]. Hence, we can utilize them to describe texture characteristics of the objects. The Zernike moments are based on alternative complex polynomial functions, known as Zernike polynomials [19]. These form a complete orthogonal set over the interior of the unit circle $x^2 + y^2 \leq 1$ and are defined as

$$V_{pq}(x, y) = V_{pq}(\rho, \theta) = R_{pq}(\rho) e^{(jq\theta)} \quad (9)$$

where $p \in Z^*$ and $q \in Z$ such that $p - |q|$ is even and $|q| \leq p$, $\rho = \sqrt{x^2 + y^2}$, $\theta = \tan^{-1} \frac{x}{y}$, and

$$R_{pq}(\rho) = \sum_{s=0}^{(p-|q|)/2} \frac{(-1)^s [(p - s)!] \rho^{p-2s}}{s!(\frac{p+|q|}{2} - s)!(\frac{p-|q|}{2} - s)!}.$$

The Zernike moments of an image region $I(x, y)$ are then computed as

$$A_{pq} = \frac{p + 1}{\pi} \sum_i I(x_i, y_i) V^*(\rho_i, \theta_i), \quad x_i^2 + y_i^2 \leq 1 \quad (10)$$

where i runs over all image pixels. Each moment A_{pq} is used as a feature descriptor for the region of interest $I(x, y)$.

In addition to the features above, the energy and entropy are calculated from ROI images that have been spatially filtered to reinforce the presence of some specific characteristic, e.g. vertical or horizontal edges [11]. Examples of the spatial filters that are used here are shown in Figure 5. These are representations of oriented Gabor and scaled Gaussian filters. Overall, this results in a feature vector of 384 features per training sample.

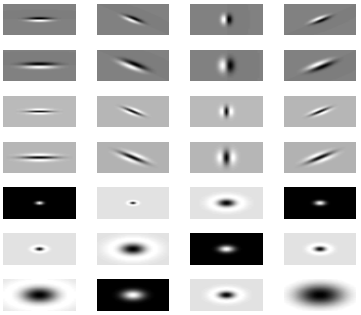
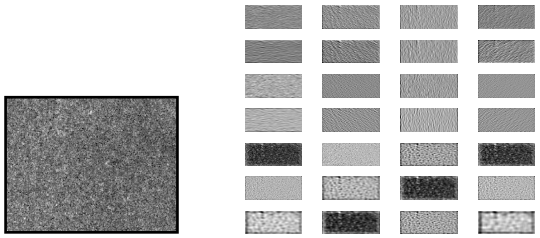
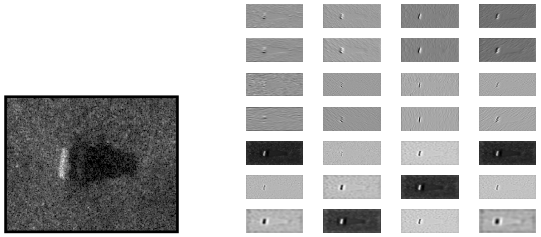


Figure 5. Example spatial filters for image characteristic enhancement.



(a) Muscle SAS background snippet. (b) Filtering results using the filters shown in Figure 5.

Figure 6. Example spatial filtering results for a background ROI.



(a) Muscle SAS target object snippet. (b) Filtering results using the filters shown in Figure 5.

Figure 7. Example spatial filtering results for a target ROI.

3. Feature Selection

Due to the large number of features $X = (x_1, \ldots, x_t)$ generated versus the number of training samples N we will down select the features that maximize the Kullback-Liebler (K-L) divergence for mutual information. The K-L divergence measures how much one probability distribution is different from another and is defined as

$$KL(p, q) = \sum_x p(x) \log \frac{p(x)}{q(x)}.$$

The goal is to reduce the burden on the classifier by removing confusing features from the training set. This should lead to more homogeneity amongst the classes. More precisely, we maximize the following

$$KL_t(S) = \frac{1}{N} \sum_{d_i \in S} KL(p(x_t|d_i), p(x_t|c(d_i))),$$

where $S = \{d_1, \ldots, d_N\}$ is the set of training samples and $c(d_i)$ is the class of d_i. This results in a feature reduction from 384 to 41 over the training data.

4. Classification

The next step in ATR after feature extraction and feature selection, which will not be discussed here, is classification. This work focuses primarily on binary target recognition. Classification of the targets will be done using Ada-boosted decision trees. Ada-boost is a machine learning algorithm, formulated by Yoav Freund and Robert Schapire[4]. It is a meta-algorithm, and can be used in conjunction with many other learning algorithms to improve their performance. Ada-boost is adaptive in the sense that subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. Ada-boost is somewhat sensitive to noisy data and outliers. Otherwise, it is less susceptible to the over-fitting problem than most learning algorithms. The classifier is trained as follows:

Given a training set $(x_1, y_1), \ldots, (x_m, y_m)$ where $y \in \{-1, 1\}$ are the correct labels of instances $x_i \in X$.

- For $t = 1, ..., T$:
- Construct a distribution D_t on $\{1, \ldots, m\}$.
- Find a weak classifier $h_t : X \to \{-1, 1\}$ with small error ϵ_t on D_t

For example, if $T = 100$ then we would have the following classifier model

$$H_{final}(x) = sign\left(\sum_{t=1}^{100} \alpha_t h_t(x)\right). \tag{11}$$

Thus, Ada-boost calls a weak classifier repeatedly in a series of rounds. For each call the distribution D_t is updated to indicate the importance of examples in the dataset for classification, i.e., the difficulty of each sample. For each round, the probability of being chosen in the next round of each incorrectly classified example are increased (or alternatively, the weights of each correctly classified example are decreased), so that the next classifier focuses more on those examples that prove more difficult to classify. The weak classifier used here in this work is a simple decision tree. A decision tree predicts the binary response to data based on checking feature values, or predictors. For example, the following tree, in Figure 8 predicts classifications based on six features, $x_1, x_2, \cdots, x_6$. The tree determines

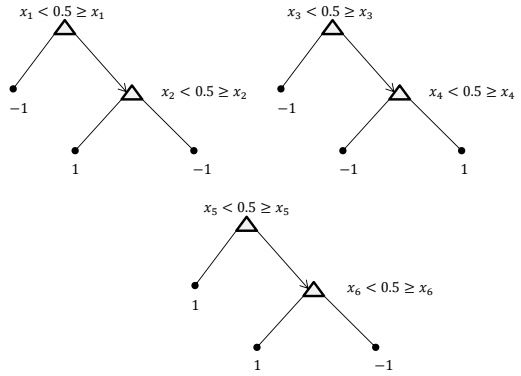


Figure 8. Simple binary decision trees for six features.

the class by starting at the top node, or root, in the tree and traversing down a branch until a leaf node is encountered. The leaf node contains the response and thus and a decision is made as to the class of the object. As shown above in eq. 11, the boosted tree result would then be the sign of the sum of traversing T binary trees. For this work we chose $T = 100$ and D_1 is 0.5 for all samples.

5. Experiments and Datasets

The experimental setup for verification of the ATR methodology is to perform detection, feature extraction, and

classification on six separate datasets containing differing levels of both noise and clutter with the same base target set. The goal then is to demonstrate reduced performance as environmental conditions deteriorate. All six datasets will contain 628 targets of varying scale (length, width and height) and rotation, examples can be seen in Figures 1 and 2. In addition, the background will include small 600 pieces of clutter, i.e. non-target-like objects with variable rotation and reflectance levels. This data was created over a one square nautical mile area, thus giving us a clutter density of 0.0185 per $10m^2$. However, considering our survey lane spacing is half of the sonar range we are guaranteed to see almost everything twice and this artificially increases the density to 0.037 per $10m^2$. Two types of temporal noise are added to the data to mimic degrading environmental conditions. The first type of noise is the sea-bottom temporal noise τ which can vary from 0 to 99.99% of the mean bottom spatial reflectance. The second type of noise is an ambient temporal noise γ that effects both the background and the targets and can vary from 0.0 to 9.99% of the mean background spatial reflectance. For this work, noise variation will range from 0 to 15.0 for τ and 0 to 2.0 for γ. The training of the classifiers was done using dataset 1 from Table 2 and testing was performed with the remaining sets.

Table 1. Fixed parameters for the dataset of Table 2 SLS simulation data experiments.

Target HL Range ($\times\mu(I_{BK})$)	[8, 20]
Clutter HL Range	[5, 10]
Target Size(m)	[.4, 3]
Clutter Size(m)	[.2, .6]

6. Results

The experiments were designed to test the robustness of the ATR algorithm against degraded data. The goal was to demonstrate gradual and predictable behavior from the ATR algorithm given the known environmental conditions. Results are evaluated on the probability of detection and classification P_{DC} and the area under the ROC curve (AUC). The results shown in Table 2 above and Figure 9 below give us a clear picture of the performance versus known temporal noise and clutter densities. The more noisy the data becomes the poorer the performance and thus the ability to distinguish between targets of interest and clutter diminishes. It is also shown that the detector struggles to find the targets and that even when they are found the temporal noise level is so high the classifier cannot determine the class.

Table 2. Dataset descriptions for the SLS simulation data experiments and the resultant P_{DC} and AUC for each set.

Dataset	τ	γ	$P_{DC}[0,1]$	$AUC[0,1]$
0	0.0	0.00	0.922	0.968
1	2.0	0.50	N/A	N/A
2	5.0	0.75	0.879	0.919
3	8.0	1.00	0.798	0.798
4	10.0	1.50	0.774	0.697
5	10.0	2.0	0.775	0.697
6	15.0	2.0	0.775	0.569

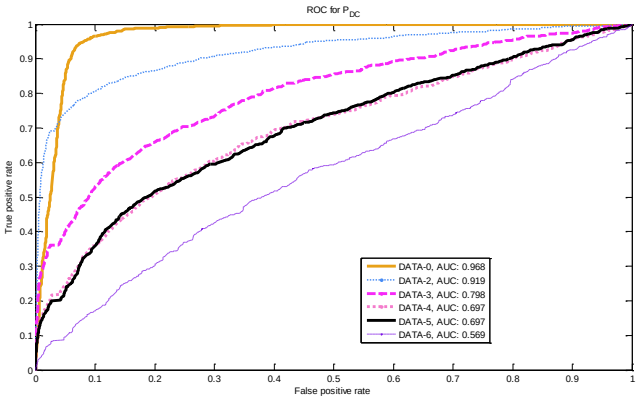


Figure 9. ROC performance curves for the data listed in Table 1.

7. Conclusions:

In this paper we have presented an approach for detecting and classifying target objects in sonar imagery with variable background noise levels and fixed clutter density. The experiments demonstrated a gradual degradation of the ATR with increasing sea-bed and ambient temporal noise levels. This predictable behavior then allows us the ability to utilize the noise information by designing a model for environmental characterization. This environmental characterization could then trigger the ATR to respond by utilizing different features, detector thresholds, or classifier parameters. We believe that this would allow for a more robust algorithm that can be applied to most sonar imagery where the objects exhibit some response above background levels.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

[2] J. Cobb, K. Slatton, and G. Dobeck. A parametric model for characterizing seabed textures in synthetic aperture sonar images. *IEEE Journal of Ocean Engineering*, (Apr.), 2010.

[3] G. J. Dobeck. Adaptive large-scale clutter removal from imagery with application to high-resolution sonar imagery. In

Proceedings SPIE 7664, Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XV, 2010.

[4] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *In Computational Learning Theory: Eurocolt 95*, page 2337, 1995.

[5] P. Gough and D. Hawkins. Imaging algorithms for a strip-map synthetic aperture sonar: minimizing the effects of aperture errors and aperture undersampling. *Oceanic Engineering, IEEE Journal of*, 22(1):27 –39, jan 1997.

[6] M. K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962.

[7] J. C. Hyland and G. J. Dobeck. Sea mine detection and classification using side-looking sonar. In *Proc. SPIE 2496, Detection Technologies for Mines and Minelike Targets, 442*, 1995.

[8] J. C. Isaacs. Laplace-beltrami eigenfunction metrics and geodesic shape distance features for shape matching in synthetic aperture sonar. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 14–20, 2011.

[9] J. C. Isaacs and J. D. Tucker. Diffusion features for target specific recognition with synthetic aperture sonar raw signals and acoustic color. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 27–32, 2011.

[10] A. Khotanzad and Y. H. Hong. Invariant image recognition by zernicke moments. *IRE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):489–497, 1990.

[11] P. D. Kovesi. A dimensionless measure of edge significance from phase congruency calculated via wavelets. In *First New Zealand Conf. on Image and Vision Comp.*, pages 87–94, 1993.

[12] B. Marchand and N. Saito. Earth mover's distance based local discriminant basis. *Multiscale Signal Analysis and Modeling, Lecture Notes in Electrical Engineering*, pages 275–294, 2013.

[13] A. Pezeshki, M. R. Azimi-Sadjadi, and L. L. Scharf. Classification of underwater mine-like and non-mine-like objects using canonical correlations. In *Proc. SPIE. 5415, Detection and Remediation Technologies for Mines and Minelike Targets IX :336.*

[14] R. J. Prokop and A. P. Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. 54(4).

[15] S. Reed, Y. Petillot, , and J. Bell. An automatic approach to the detection and extraction of mine features in sidescan sonar. *IEEE J. Ocean. Engineering*, 28(1):90105, 2003.

[16] J. R. Stack. Automation for underwater mine recognition: current trends and future strategy. In *Proceedings SPIE 8017, Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XVI*, 2011.

[17] M. R. Teague. Image analysis via the general theory of moments. 70.

[18] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Elsevier, 1999.

[19] F. Zernike. Beugungstheorie des schneidenverfahrens und seiner verbesserten form, der phasenkontrastmethode. *Physica*, 1:689–690, 1934.