

FINAL REPORT

Environmentally Adaptive UXO Detection and Classification Systems

SERDP Project MR-2417

APRIL 2016

Dr. Nick Klausner
Information System Technologies, Inc.

Distribution Statement A

This document has been cleared for public release



Page Intentionally Left Blank

This report was prepared under contract to the Department of Defense Strategic Environmental Research and Development Program (SERDP). The publication of this report does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official policy or position of the Department of Defense. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Department of Defense.

Page Intentionally Left Blank

REPORT DOCUMENTATION PAGE					<i>Form Approved</i> OMB No. 0704-0188	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.						
1. REPORT DATE (DD-MM-YYYY) 12/04/2016		2. REPORT TYPE Final Report			3. DATES COVERED (From - To) October 15, 2014 - March 15, 2016	
4. TITLE AND SUBTITLE Environmentally Adaptive UXO Detection and Classification Systems				5a. CONTRACT NUMBER W912HQ-14-P-0033		
				5b. GRANT NUMBER		
				5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Dr. Nick Klausner Information System Technologies, Inc.				5d. PROJECT NUMBER MR-2417		
				5e. TASK NUMBER		
				5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Information System Technologies, Inc. 425 West Mulberry St, Ste 108 Fort Collins, CO 80521					8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Strategic Environmental Research and Development Program 4800 Mark Center Drive Suite 17D03 Alexandria, VA 22350-3605					10. SPONSOR/MONITOR'S ACRONYM(S) SERDP	
					11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.						
13. SUPPLEMENTARY NOTES N/A						
14. ABSTRACT This Final Report addresses the problem of detecting and classifying underwater munitions using data collected from synthetic aperture sonar (SAS) systems.						
15. SUBJECT TERMS N/A						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Nick Klausner	
U	U	U	UU	39	19b. TELEPHONE NUMBER (Include area code) (970) 224-2556	

Page Intentionally Left Blank

Contents

1	Abstract	4
2	Objective	4
3	Background	5
4	Materials and Methods	6
4.1	A GLRT-Based Approach to Matched Filter Detection	6
4.1.1	Motivation	6
4.1.2	Development of the Detection Statistic	7
4.1.3	Invariance Properties of the Detector	9
4.1.4	Simulation	10
4.2	Manifold-Based Classification	12
4.2.1	Feature Extraction Using Manifold Learning	13
4.2.2	Manifold-Based Classification Using Local Linear Representations	15
5	Results and Discussion	17
5.1	Dataset Description	17
5.2	Detection Results	18
5.2.1	Formation of the Test Statistic	18
5.2.2	Model Validation	19
5.2.3	PondEx Detection Results	19
5.3	Classification Results	22
5.3.1	Feature Extraction and Classification Procedures	22
5.3.2	PondEx Classification Results	26
6	Conclusions and Implications for Future Research/Implementation	28
6.1	Conclusions and Discussions	28
6.2	Proposed Future Research and Development	29
6.2.1	Task 1: Detector's Performance Prediction and Optimization	29
6.2.2	Task 2: Adaptive Platform Motion Compensation Adaptive Matched Filter Detection	30
6.2.3	Task 3: Computationally Efficient Manifold-Based Classification using Sub- space Averaging	33
6.2.4	Task 4: Comprehensive Testing & Evaluation	35

List of Acronyms

Acronym	Meaning
APL	Applied Physics Laboratory
ATR	Automatic Target Recognition
BOSS	Buried Object Scanning Sonar
CGF	Cumulant Generating Function
CPA	Closest Point of Approach
DoD	Department of Defense
FFT	Fast Fourier Transform
iid	Independent and Identically Distributed
ISOMAP	Isometric Mapping (Algorithm)
GLRT	Generalized Likelihood Ratio Test
LE	Laplacian Eigenmaps
LFM	Linear Frequency Modulated
LLE	Local Linear Embedding
MC	Munitions Constituents
MDS	Multidimensional Scaling
MGF	Moment Generating Function
ML	Maximum Likelihood
MVU	Maximum Variance Unfolding
MVUB	Minimum Variance Unbiased
NSWC	Naval Surface Warfare Center
PC SWAT	Personal Computer Shallow Water Acoustic Toolset
PDF	Probability Density Function
ROC	Receiver Operating Characteristic
SAS	Synthetic Aperture Sonar
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
UXO	Unexploded Ordnance

Keywords

Manifold Learning, Laplacian Eigenmaps, Acoustic Color, Matched Filter, Generalized Likelihood Ratio Test, Synthetic Aperture Sonar, Unexploded Ordnance (UXO) Detection and Classification.

Acknowledgments

The investigators would like to thank APL-University of Washington as well as NSWC-Panama City for support and providing the data used in this study.

Environmentally Adaptive UXO Detection and Classification Systems

1 Abstract

This final report addresses the problem of detecting and classifying underwater munitions using data collected from synthetic aperture sonar (SAS) systems. The first portion of this report discusses an adaptive scheme for UXO detection. Our detection hypothesis is that the presence of munitions in the sonar backscatter collected from a hydrophone array will inherently lead to a low-rank component in the pulse compressed data from multiple pings. A statistical hypothesis test is developed to determine when this low-rank component is present using the Generalized Likelihood Ratio Test (GLRT).

The second portion of this report addresses the problem of discriminating UXO from non-UXO objects using manifold learning principles when applied to data collected from SAS systems. This effort addresses the second and third tasks in our SEED proposal which involve the development of feature extraction and classification strategies for this problem. Our classification hypothesis is that the sequence of measurements collected from an object in a linear SAS survey results in data that lies in some low-dimensional subspace which is locally linear but globally non-linear, i.e. the data is assumed to lie on a low-dimensional manifold embedded in a high-dimensional space. The coordinates on that low-dimensional manifold and their behavior can then be used to discriminate among various UXO and non-UXO objects that may be encountered in practice. With this goal in mind, techniques are developed to not only learn the low-dimensional manifold but to also provide an out-of-sample embedding for newly acquired training data. The manifold features from the training set are then used to construct local linear subspaces for representing each newly embedded testing feature. A statistically motivated technique is then used to select the most likely class label by finding the class which best represents the data.

Test results for both the detector and classifier are then presented using an experimental data set which was designed to collect sonar data from underwater objects in a relatively controlled and clutter-free environment. Additionally, various experiments are conducted to observe the proposed system's robustness to various forms of mismatch that may enter the data collection process. Results are presented using standard performance metrics such as probability of detection (P_d), probability of correct classification (P_{cc}), probability of false alarm (P_{fa}), as well as Receiver Operating Characteristic (ROC) curve and confusion matrix characteristics.

The results of these studies show that the detector, although applied in fairly ideal conditions, is capable of discriminating sonar returns of objects lying on the seafloor from the background with a probability of detection of $P_d = 98\%$ and an average of 2.4 false alarms per image (with each image covering approximately 20 m²). Images of the likelihood ratio produced by the detector also demonstrate its ability to localize each object on the seafloor. Classification results generated using the same experimental data set demonstrate the ability of the proposed classification technique to accurately discriminate the sonar returns of UXO objects from those of non-UXO objects with a probability of correct classification of $P_{cc} = 93\%$. Moreover, the proposed method is able to correctly classify nearly 70% of the testing data for the 'real' UXO which was not included during the training process, demonstrating the relative robustness of the proposed method.

2 Objective

Detection, classification, and remediation of military munitions and unexploded ordnance in shallow water is of utmost importance to many DoD agencies owing to the severity of threats they pose

to humans and the environment. The problem is technically challenging due to variability in environmental conditions as well as obscuration of the munitions. Thus, new methods are needed to rapidly and reliably assess large areas that are potentially contaminated with munitions and detect, localize, and identify each individual threat with a high degree of accuracy. To this end, this research addresses an important shortcoming of the existing Automatic Target Recognition (ATR) algorithms that use sonar data by developing new environmentally adaptive algorithms for the detection and classification of military munitions in shallow underwater environments using data collected from low frequency broadband SAS systems.

Specifically, one of the technical objectives addressed in this work concerns the development and preliminary testing of an adaptive detection technique using a GLRT and its application to the detection of munitions using low frequency sonar data. For this problem, the hypothesis is that the presence of an object in the sonar backscatter collected from a synthetic aperture array will lead to time delayed and scaled versions of a similar response which will manifest itself in the form of a strong low-rank component in the data. The presence of this low-rank representation can give one an indication of which areas of the field may possibly contain munitions and subsequent classification and further analysis may then be conducted in those areas. As the model proposed in this work contains a number of deterministic but unknown variables, we employ the use of the GLRT [1] which implements a likelihood ratio test by replacing these unknown quantities with their maximum likelihood (ML) estimates. This leads to a test statistic that remains invariant to certain linear transformations of the data. This means that the detector remains robust to transformations of the data that fall within this specific transformation group.

The second technical objective addressed in this work concerns the development and preliminary testing of a feature extraction and classification technique using manifold learning principles and their applications to the discrimination among various UXO and non-UXO objects using low frequency sonar data. The proposed methods are designed around the manifold learning framework [2] which assumes that the data lies in some unknown low-dimensional subspace which is locally linear. These low-dimensional features and their aspect-dependent behavior on the manifold can then be used to discriminate the data from one object type from another. With this objective in mind, manifold training using the Laplacian Eigenmaps (LE) algorithm [3] was adopted. This algorithm can be easily extended to include previously unseen testing data to yield an out-of-sample embedding technique. This features allows us to not only embed new points on the manifold but to also track changes in the manifold as data is collected. Once these manifold features have been extracted, the training features from each object type are then used to construct linear subspaces for locally representing each extracted feature. A minimum error criterion for classification is then proposed based on maximum likelihood (ML) principles which selects the class which best represents the data.

The performance of the proposed methods is then demonstrated using both simulation as well as by applying it to a data set (PondEx09 and PondEx10) collected in a freshwater pond consisting of a rail system collecting sonar backscatter from multiple munitions. Experiments are also conducted to test the proposed system’s robustness to various modes of mismatch such as discrepancies in the material properties of the object. Metrics such as probability of detection, probability of correct classification, false alarm rate, as well as ROC curve and confusion matrix characteristics will be used to evaluate the performance of the proposed method.

3 Background

Underwater Unexploded Ordnance (UXO) and Munitions Constituents (MC) pose serious risk of harm to humans, marine life and the environment. Automatic detection and classification of these underwater threats is, however, a very challenging problem due to many complicating factors including: (a) highly variable operating and environmental conditions (e.g., lakes, ponds, rivers,

gulf, or open ocean); (b) variations in target features as a function of range, grazing angle, and orientation with respect to the sensor platform as well as its size and composition; and (c) targets may be partially, or fully buried, or obscured by marine growth and vegetation. This particular research responds to SERDP SON MRSEED-14-01 in *Wide Area and Detailed Surveys* for rapid and highly efficient detection and classification of underwater UXO and MC from sonar returns by developing novel adaptive ATR algorithms that remain robust to a wide range of environmental conditions found in contaminated sites.

In many areas such as machine learning, ATR, and information retrieval/data mining, one is interested in extracting low-dimensional data that is truly representative of the properties of the data from a very high dimensional (ambient) space. Among the manifold learning methods are Isometric Feature Mapping (ISOMAP) [4],[5], Locally Linear Embedding (LLE) [6],[7], Maximum Variance Unfolding (MVU) or semi-definite embedding [8], and Laplacian Eigenmaps [3]. The general principle behind Isomap is to use geodesic distances (not Euclidean distance which obscures the intrinsic manifold structure) on a graph together with classical Multidimensional Scaling (MDS). Unlike ISOMAP [4] which is a global approach, i.e. preserves geometry at all scales, LLE and Laplacian Eigenmaps are local approaches in that they attempt to only preserve the local geometry of the data by mapping nearby points on the manifold to nearby points in the low-dimensional representation. Similar to ISOMAP and LLE, MVU [8] also belongs to the class of *spectral embedding*, however, it exploits different geometrical properties. ISOMAP is based upon geodesic distances, LLE on the coefficients of local linear reconstructions, and Laplacian eigenmaps on the discrete graph Laplacian, whereas MVU is based on estimating and preserving local distances and angles. In spite of their similarities, in [8] it was shown that, for cases where the sampled manifold is not isometric to a convex subset of Euclidean space, ISOMAP produces totally different results than that produced by MVU. Additionally, these methods exhibit other problems including inability to deal with highly curved manifolds and out-of-sample extension for nonisometric manifolds. The latter implies that they fail to provide a feature mapping (explicit or implicit) to map new data points that are not included in the original training set.

4 Materials and Methods

In this section, we give a brief review of the theory that motivates the detection, feature extraction, and classification algorithms studied in this project. More specifically, we will begin by reviewing the theory of the GLRT-based detection method and its application to low frequency SAS data. We then give a review of the manifold-based feature extraction and manifold domain classification techniques used to discriminate various UXO targets from other object types.

4.1 A GLRT-Based Approach to Matched Filter Detection

4.1.1 Motivation

In this section we give a review of the theory behind the techniques used to detect the presence of objects in sonar backscatter from the seafloor. For this application, we assume that sonar data is collected by translating one (or more) receiver elements in along-track as the transmitter ensonifies the target area as is depicted in Figure 1 (a). Let $x[n, m]$ denote the observed sonar backscatter at temporal sample (range) n and ping (along-track) m after matched filtering the received waveform with a replica of the transmit signal. If a target in the field is present at a range r and produces the response $h[n]$, then in the absence of other returns (such as direct bottom reflections and multipath effects) it is assumed that the matched filtered response at ping m can be written as $x[n, m] = \theta_m h[n - \tau_m] + w[n, m]$. That is, we assume that the response observed at ping m is simply a time-delayed, scaled version of the target response plus additive noise $w[n, m]$. The time delay τ_m is due to the increase in path length as the sensor moves in along-track. Specifically, if the sensor

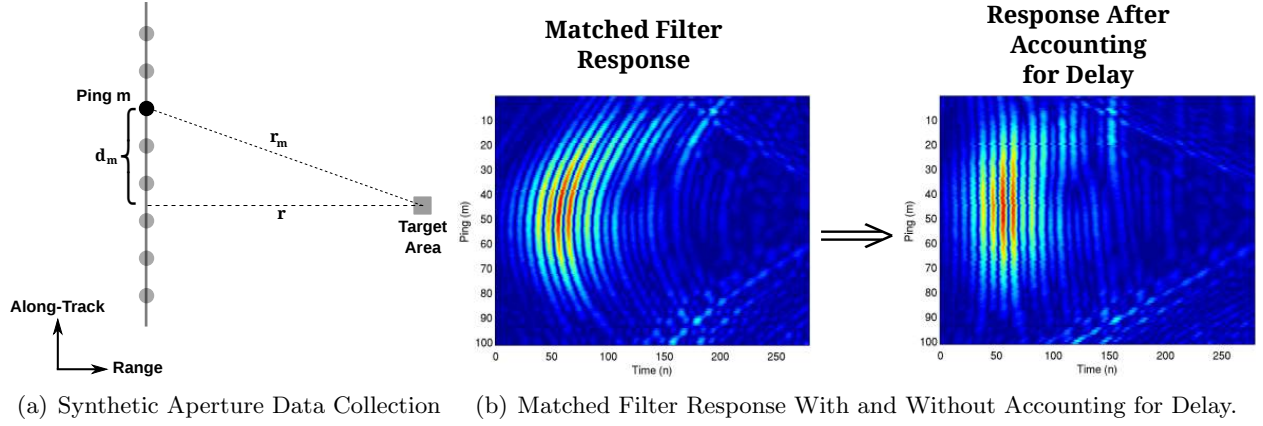


Figure 1: Collection of data using a synthetic aperture and the matched filter response with and without accounting for delays due to increasing path length.

has been displaced by a distance d_m in along-track and assuming no platform motion, the range at ping m can be related to the range of the target at the CPA as $r_m = \sqrt{r^2 + d_m^2}$. Since this time delay only depends on the range and geometry of the aperture, it can be accounted for and its effect removed from the matched filtered data by lagging the time series at each ping accordingly. Figure 1 (b) gives an example of the matched filter response of an object sitting proud on the seafloor as well as the response that results after accounting for and removing the effects of time delays. From this response one can see that, from ping to ping, one tends to observe scaled replicas of the same target response. Inherent in this assumption, however, is that the sonar observes the target over a relatively small range of aspects such as in linear SAS applications. One would expect this assumption to be less applicable in applications such as circular SAS where the sonar observes the target over all aspect angles.

Although this model of the data is fairly simplistic, there are certain quantities that remain unknown. One, the matched filter response of the target $h[n]$ is unknown as it can vary depending on many things including target type and composition, range, aspect angle, properties of the transmission medium, whether the target is lying proud on the seafloor or it is buried, etc. Moreover, the scaling of the target response is unknown as the intensity of the sonar return will vary from ping to ping. Finally, the noise process $w[n, m]$ is assumed to be zero-mean white Gaussian noise but its variance σ^2 is assumed to be unknown as its second-order statistics can vary depending on the environment in which data is being collected. To account for these unknown quantities, we employ the GLRT which replaces each unknown variable with its ML estimate.

4.1.2 Development of the Detection Statistic

Recalling the arguments given in Section 4.1.1, we begin by assuming that we've collected the set of vectors $\{\mathbf{x}[m]\}_{m=1}^M$ with the vector $\mathbf{x}[m] = [x[0, m] \ x[1, m] \ \cdots \ x[N-1, m]]^T \in \mathbb{R}^N$ representing the length- N matched filtered response collected at ping m . Predicting and removing the time delay τ_m from each ping, we assume that the response can be expressed as $x[n, m] = \theta_m h[n] + w[n, m]$ so that the sequence of vectors $\mathbf{x}[1], \dots, \mathbf{x}[M]$ follows the linear model

$$\mathbf{x}[m] = \theta_m \mathbf{h} + \mathbf{w}[m], \quad m = 1, \dots, M$$

In this equation, the vector $\mathbf{h} = [h[0] \ \cdots \ h[N-1]]^T \in \mathbb{R}^N$ contains the response of the target and $\mathbf{w}[m] = [w[0, m] \ \cdots \ w[N-1, m]]^T \in \mathbb{R}^N$ is a vector of random noise with distribution $\mathbf{w}[m] \stackrel{iid}{\sim}$

$\mathcal{N}(0, \sigma^2 \mathbf{I})$. Defining the data matrices

$$\mathbf{X} = [\mathbf{x}[1] \ \mathbf{x}[2] \ \cdots \ \mathbf{x}[M]] \in \mathbb{R}^{N \times M} \quad (1)$$

$$\mathbf{W} = [\mathbf{w}[1] \ \mathbf{w}[2] \ \cdots \ \mathbf{w}[M]] \in \mathbb{R}^{N \times M} \quad (2)$$

this representation can be alternatively expressed as $\mathbf{X} = \mathbf{h}\boldsymbol{\theta}^T + \mathbf{W}$ where $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \cdots \ \theta_M]^T \in \mathbb{R}^M$. Since there exists a scaling ambiguity in the outer product $\mathbf{h}\boldsymbol{\theta}^T$, there is no loss in generality to assume that the vectors $\mathbf{h}$ and $\boldsymbol{\theta}$ both have unit length ($\|\mathbf{h}\| = \|\boldsymbol{\theta}\| = 1$) and that there exists a scalar $\lambda \geq 0$ such that the model of our observation can be written $\mathbf{X} = \lambda \mathbf{h}\boldsymbol{\theta}^T + \mathbf{W}$. So to review, we assume that the data matrix containing our observations, $\mathbf{X}$, consists of the unknown rank-one matrix $\lambda \mathbf{h}\boldsymbol{\theta}^T$ plus a matrix containing *iid* realizations of zero-mean, Gaussian distributed noise with unknown variance σ^2 . Given this model, we are interested in testing the null hypothesis that our observations consist of noise alone by considering the hypothesis test

$$\begin{aligned} \mathcal{H}_0 &: \lambda = 0 \quad (\text{No Target}) \\ \mathcal{H}_1 &: \lambda > 0 \quad (\text{Potential Target Present}) \end{aligned} \quad (3)$$

Given the assumptions of this model, the data matrix $\mathbf{X}$ has the probability density function (PDF)

$$f(\mathbf{X}; \lambda, \mathbf{h}, \boldsymbol{\theta}, \sigma^2) = \prod_{i=1}^M f(\mathbf{x}[m]; \lambda, \mathbf{h}, \theta_m, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{NM/2}} \exp \left\{ -\frac{1}{2\sigma^2} \|\mathbf{X} - \lambda \mathbf{h}\boldsymbol{\theta}^T\|_F^2 \right\} \quad (4)$$

where $\|\mathbf{A}\|_F^2 = \text{tr}(\mathbf{A}^T \mathbf{A})$ denotes the Frobenius norm of matrix $\mathbf{A}$. The first step in the computation of any GLRT involves finding the ML estimates of the unknown parameters. Taking the negative logarithm of the expression given in (4) and removing data and parameter-independent constants, maximizing likelihood is equivalent to minimizing the expression

$$\ell(\lambda, \mathbf{h}, \boldsymbol{\theta}, \sigma^2) = NM \ln(\sigma^2) + \frac{1}{\sigma^2} \|\mathbf{X} - \lambda \mathbf{h}\boldsymbol{\theta}^T\|_F^2 \quad (5)$$

Looking at this expression, it is clear that the values of λ , $\mathbf{h}$, and $\boldsymbol{\theta}$ must be chosen such that the squared error $\|\mathbf{X} - \lambda \mathbf{h}\boldsymbol{\theta}^T\|_F^2$ is minimum. Let ν denote the largest singular value of matrix $\mathbf{X}$ with corresponding left and right singular vectors $\mathbf{u}$ and $\mathbf{v}$, respectively, found by taking the Singular Value Decomposition (SVD) of matrix $\mathbf{X}$. Then by the Eckart-Young theorem [9], the matrix $\nu \mathbf{u}\mathbf{v}^T$ is the rank-one matrix that minimizes squared error [10], i.e.

$$\nu \mathbf{u}\mathbf{v}^T = \arg \min_{\text{rank}(\tilde{\mathbf{X}})=1} \|\mathbf{X} - \tilde{\mathbf{X}}\|_F^2$$

Accordingly, it follows that $\hat{\lambda} = \nu$, $\hat{\mathbf{h}} = \mathbf{u}$, and $\hat{\boldsymbol{\theta}} = \mathbf{v}$ are the ML estimates of these parameters. Substituting these estimates into the expression given in (5) and minimizing over the parameter σ^2 , one obtains the expression

$$\hat{\sigma}_1^2 = \arg \min_{\sigma} \ell(\nu, \mathbf{u}, \mathbf{v}, \sigma^2) = \frac{1}{NM} \|\mathbf{X} - \nu \mathbf{u}\mathbf{v}^T\|_F^2$$

Using properties of the SVD, we note that $\nu \mathbf{v}^T = \mathbf{u}^T \mathbf{X}$ so that this expression can be written as

$$\hat{\sigma}_1^2 = \frac{1}{NM} \|\mathbf{X} - \nu \mathbf{u}\mathbf{v}^T\|_F^2 = \frac{1}{NM} \|\mathbf{X} - \mathbf{u}\mathbf{u}^T \mathbf{X}\|_F^2 = \frac{1}{NM} \|(I - \mathbf{P}_{\mathbf{u}}) \mathbf{X}\|_F^2 = \frac{1}{NM} \text{tr}(\mathbf{X}^T \mathbf{P}_{\mathbf{u}}^\perp \mathbf{X})$$

where $\mathbf{P}_{\mathbf{u}} = \mathbf{u}\mathbf{u}^T$ denotes the orthogonal projection matrix onto the one-dimensional subspace $\langle \mathbf{u} \rangle$ and $\mathbf{P}_{\mathbf{u}}^\perp = \mathbf{I} - \mathbf{P}_{\mathbf{u}}$ denotes the projection onto its orthogonal complement subspace $\langle \mathbf{u} \rangle^\perp$. Likewise, under the null hypothesis that $\lambda = 0$ one similarly obtains the estimate

$$\hat{\sigma}_0^2 = \arg \min_{\sigma} \ell(0, \mathbf{u}, \mathbf{v}, \sigma^2) = \frac{1}{NM} \text{tr}(\mathbf{X}^T \mathbf{X})$$

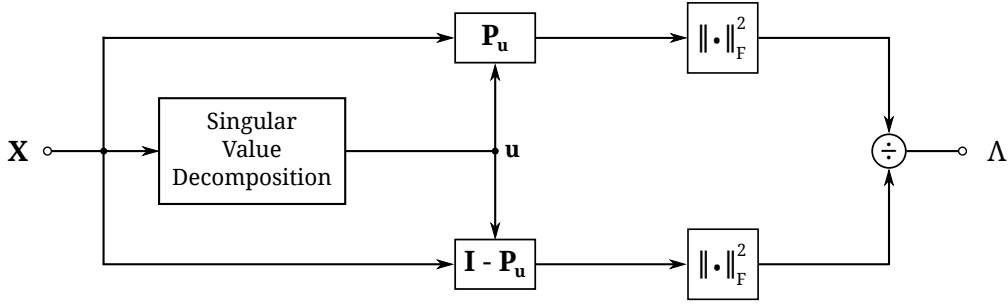


Figure 2: Block diagram of the implementation of the test statistic in (6)

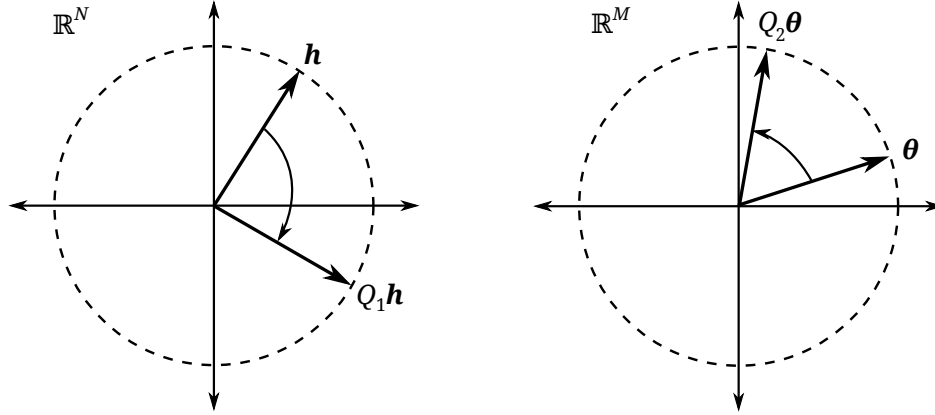


Figure 3: Multiplication by the orthogonal matrices $\mathbf{Q}_1$ and $\mathbf{Q}_2$ rotates $\mathbf{h}$ and $\boldsymbol{\theta}$ in $\mathbb{R}^N$ and $\mathbb{R}^M$, respectively.

Substituting these estimates into the expression given in (4), one obtains the GLRT

$$\Lambda = \frac{\max_{\lambda, \mathbf{h}, \boldsymbol{\theta}, \sigma^2} f(\mathbf{X}; \lambda, \mathbf{h}, \boldsymbol{\theta}, \sigma^2)}{\max_{\sigma^2} f(\mathbf{X}; 0, \mathbf{h}, \boldsymbol{\theta}, \sigma^2)} = \left(\frac{\hat{\sigma}_0^2}{\hat{\sigma}_1^2} \right)^{\frac{NM}{2}}$$

Taking a monotonically increasing function of this likelihood ratio, we can finally express the test statistic as

$$\tilde{\Lambda} = \Lambda^{\frac{2}{NM}} - 1 = \frac{\text{tr}(\mathbf{X}^T \mathbf{P}_{\mathbf{u}} \mathbf{X})}{\text{tr}(\mathbf{X}^T \mathbf{P}_{\mathbf{u}}^{\perp} \mathbf{X})} = \frac{\sum_{m=1}^M \mathbf{x}[m]^T \mathbf{P}_{\mathbf{u}} \mathbf{x}[m]}{\sum_{m=1}^M \mathbf{x}[m]^T \mathbf{P}_{\mathbf{u}}^{\perp} \mathbf{x}[m]} \quad (6)$$

Figure 2 gives a block diagram of the implementation of the test statistic given in (6). The SVD of the data matrix $\mathbf{X}$ is first computed. Its leading left singular vector $\mathbf{u}$ is then extracted and used to construct the projection matrices $\mathbf{P}_{\mathbf{u}}$ and $\mathbf{P}_{\mathbf{u}}^{\perp}$. The test statistic given in (6) is then measured by computing the ratio of total energy of $\mathbf{X}$ that lies in the subspace $\langle \mathbf{u} \rangle$ to the total energy that lies in the subspace $\langle \mathbf{u} \rangle^{\perp}$. The higher the percentage of energy that lies in $\langle \mathbf{u} \rangle$ relative to the total energy in $\mathbf{X}$, the more evidence in support of the conclusion that there is indeed a strong rank-one component present in the data.

4.1.3 Invariance Properties of the Detector

As is commonly the case in statistical hypothesis testing, there exists a natural group of transformations that leave both the hypothesis testing problem in (3) and test statistic in (6) invariant, i.e. both the test statistic and detection problem itself remain unchanged upon replacing $\mathbf{X}$ with

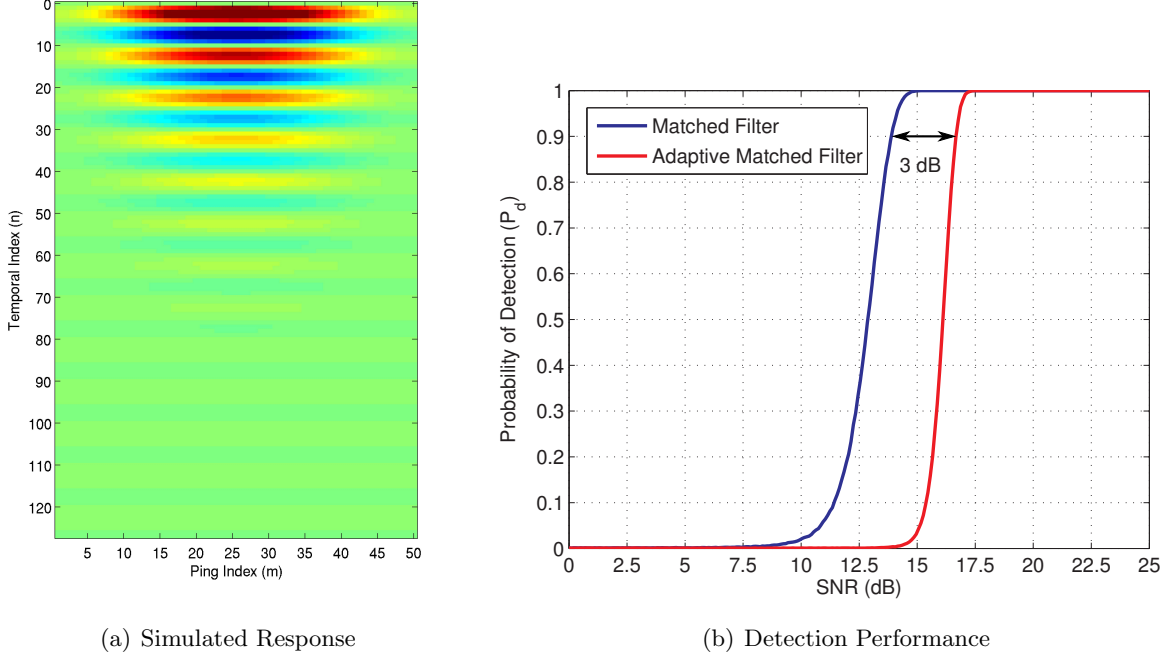


Figure 4: The rank-one matrix $\mathbf{h}\boldsymbol{\theta}^T$ used for simulation and the probability of detection versus SNR for both detection techniques at a false alarm probability of $P_{fa} = 1 \times 10^{-3}$.

$\tilde{\mathbf{X}} = g(\mathbf{X})$. In this case, the problem remains invariant to the group of transformations

$$G = \{g : g(\mathbf{X}) = c\mathbf{Q}_1\mathbf{X}\mathbf{Q}_2^T; c \in \mathbb{R}, \mathbf{Q}_1^T\mathbf{Q}_1 = \mathbf{Q}_1\mathbf{Q}_1^T = \mathbf{I}_N, \mathbf{Q}_2^T\mathbf{Q}_2 = \mathbf{Q}_2\mathbf{Q}_2^T = \mathbf{I}_M\}$$

That is, the detection problem itself is invariant to or unchanged by scaling as well as both pre and post-multiplication by any $N \times N$ and $M \times M$ orthogonal matrix $\mathbf{Q}_1$ and $\mathbf{Q}_2$, respectively. As depicted in Figure 3, pre and post multiplying the rank-one matrix $\mathbf{h}\boldsymbol{\theta}^T$ by these orthogonal matrices to produce the new rank-one matrix $\mathbf{Q}_1\mathbf{h}\boldsymbol{\theta}^T\mathbf{Q}_2^T = (\mathbf{Q}_1\mathbf{h})(\mathbf{Q}_2\boldsymbol{\theta})^T$ simply corresponds to a rotation of the vectors $\mathbf{h}$ and $\boldsymbol{\theta}$ in their corresponding spaces. This implies that it is not the particular shape of the vectors $\mathbf{h}$ and $\boldsymbol{\theta}$ that matters but only their energy $\|\mathbf{h}\|^2$ and $\|\boldsymbol{\theta}\|^2$. This is no doubt a consequence of the fact that both $\mathbf{h}$ and $\boldsymbol{\theta}$ were treated as unknown quantities in the derivation of the likelihood ratio. From a practical standpoint, this means that if a particular target produces the response and scaling vectors $\mathbf{h}$ and $\boldsymbol{\theta}$, respectively, and a transformation of that target such as rotation in aspect or translation in range produces the new vectors $\tilde{\mathbf{h}}$ and $\tilde{\boldsymbol{\theta}}$ but the energy remains conserved (i.e. each pair is related through multiplication by an orthogonal matrix) then the ability to detect that target will remain unchanged.

4.1.4 Simulation

In this section we provide some simulation results to demonstrate the usefulness of the test statistic given in (6). In this simulation we assume the presence of a target that produces the response $h[n]$ and ping-to-ping scaling θ_m given as

$$\begin{aligned} h[n] &= e^{-0.05n} \sin\left(\frac{2\pi n}{10}\right), \quad n = 0, \dots, N-1 \\ \theta_m &= \frac{1}{2} \left(1 - \cos\left(2\pi \frac{m-1}{M-1}\right)\right), \quad m = 1, \dots, M \end{aligned}$$

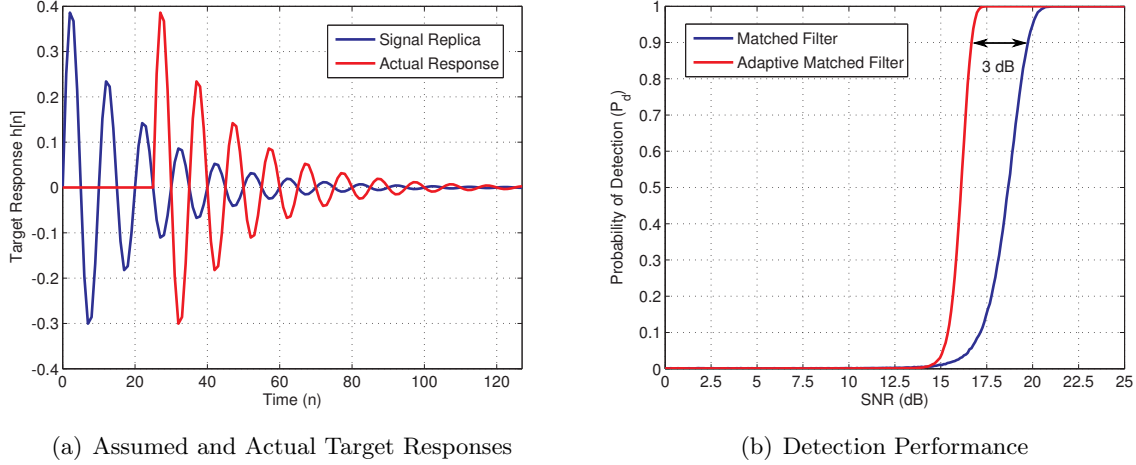


Figure 5: The assumed and observed target responses as well as the probability of detection versus SNR for both detection techniques at $P_{fa} = 1 \times 10^{-3}$.

with $N = 128$ and $M = 50$. Again, given the invariances of this problem, the particular choice in these two expressions is completely arbitrary. These two expressions are used to construct the vectors $\mathbf{h}$ and $\boldsymbol{\theta}$ which are both subsequently normalized such that $\|\mathbf{h}\| = \|\boldsymbol{\theta}\| = 1$. Data is then generated as $\mathbf{X} = \lambda \mathbf{h} \boldsymbol{\theta}^T + \mathbf{W}$ with matrix $\mathbf{W}$ containing *iid* realizations of a zero-mean normally distributed random variable with variance $\sigma^2 = 0.1$. Figure 4 (a) displays an image of the rank-one matrix $\mathbf{h} \boldsymbol{\theta}^T$ used in these simulations.

In addition to evaluating the likelihood ratio in (6), its performance is compared to that of the clairvoyant detector which has *a priori* knowledge of the response vector $\mathbf{h}$. Assuming that the scaling vector $\boldsymbol{\theta}$ is still unknown, employing the GLRT results in the following likelihood ratio [10] for the clairvoyant detector

$$\Lambda_{MF} = \frac{M(N-1)}{M} \frac{\text{tr}(\mathbf{X}^T P_{\mathbf{h}} \mathbf{X})}{\text{tr}(\mathbf{X}^T P_{\mathbf{h}}^{\perp} \mathbf{X})} \quad (7)$$

where $P_{\mathbf{h}} = \mathbf{h} \mathbf{h}^T$ and $P_{\mathbf{h}}^{\perp} = I - P_{\mathbf{h}}$ represent the orthogonal projection onto the one dimensional subspace $\langle \mathbf{h} \rangle$ and its orthogonal complement, respectively. Although the expressions in (6) and (7) look equivalent, the difference is that (7) uses the known signal $\mathbf{h}$ to build its projection matrices while (6) uses the singular vector $\mathbf{u}$ corresponding to the largest singular value. It is well known [10] that the test statistic given in (7) is distributed as $\Lambda_{MF} \sim F_{M, M(N-1)}(\lambda^2/\sigma^2)$ where $F_{\nu_1, \nu_2}(\delta)$ denotes a noncentral F -distribution with ν_1 and ν_2 degrees of freedom and noncentrality parameter δ .

Simulating both test statistics under the null hypothesis ($\lambda = 0$), a threshold was chosen for each detector to achieve a false alarm probability of $P_{fa} = 1 \times 10^{-3}$. Figure 4 (b) compares the probability of detection (P_d) for both the detector given in (7) (denoted “Matched Filter”) as well as that given in (6) (denoted “Adaptive Matched Filter”) as a function of the signal-to-noise ratio (SNR) defined to be $\text{SNR} = 10 \log_{10}(\lambda/\sigma^2)$. From the results of this simulation one can see that the Matched Filter given in (7) outperforms its adaptive counterpart given in (6). This is to be expected given the fact that the Matched Filter has *a priori* knowledge of the target response. To see what happens when this is not the case, a simulation was conducted in which there exists a mismatch between the assumed target response used to construct (7) and the actual target response that is used to generate the data. The plot shown in blue in Figure 5 (a) shows the target response $h[n]$ used to build the projection matrices used in the computation of (7). However, the plot shown in red in Figure 5 (a), which is simply a delayed version of the plot given in blue, is used to generate

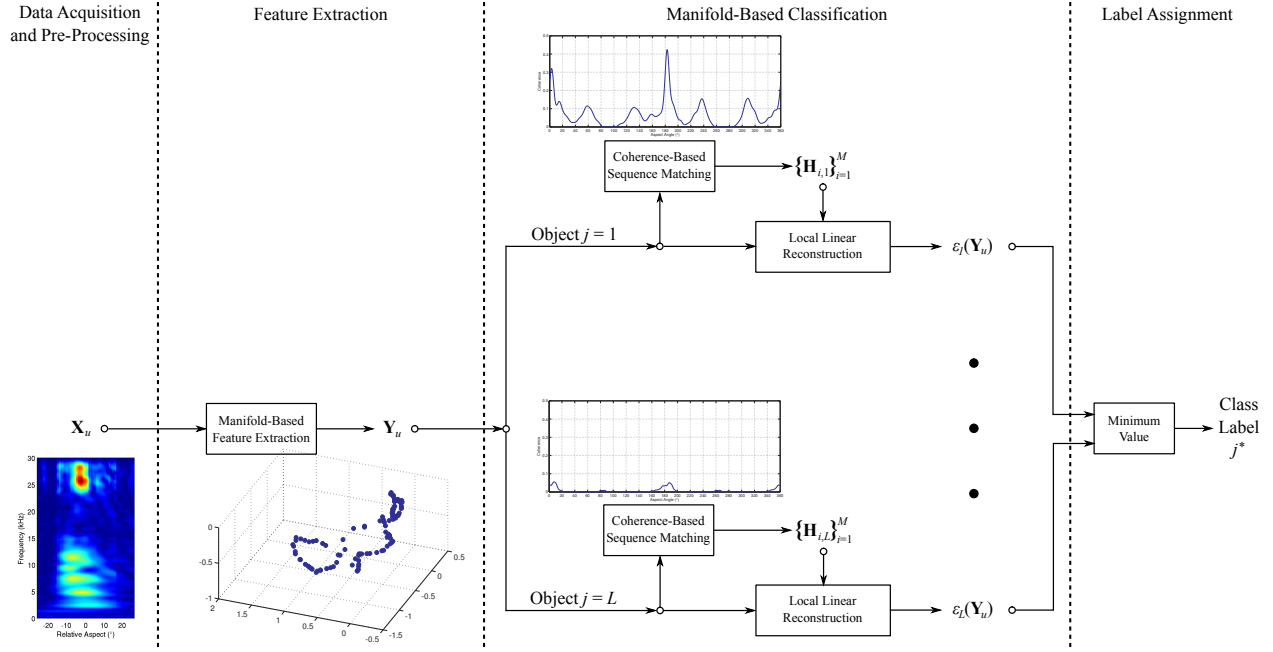


Figure 6: Block-diagram of the proposed feature extraction and classification system for UXO discrimination.

the data matrix $\mathbf{X}$, i.e. there is model mismatch in the target response. Figure 5 (b) once again compares the probability of detection for both methods at $P_{fa} = 1 \times 10^{-3}$ in the presence of this mismatch. From this figure one can see that the mismatch in target response does indeed cause a degradation in performance for the Matched Filter given in (7). However, the Adaptive Matched Filter in (6), which makes no *a priori* assumptions about the target response, exhibits the same performance as observed in Figure 4 (b) and is unaffected by this mismatch. Thus, from the results in Figures 4 (b) and 5 (b) one can conclude that, if one is able to predict the true target response with high confidence, then the Matched Filter given in (7) is clearly superior. In this application however, the target response can depend on many things including target type, composition, aspect angle, range, degree of burial, as well as different environmental factors and is therefore very difficult to predict *a priori*. For this reason, we choose to employ the Adaptive Matched Filter in (6) for the detection of underwater munitions given its robustness to the effects of model mismatch.

4.2 Manifold-Based Classification

This section gives a review of the manifold-based feature extraction and classification techniques used to classify low frequency sonar returns. Figure 6 gives a block-diagram of the entire process consisting of 4 intermediate stages. After acquiring and pre-processing the data to prepare it for the classification process, the first step involves extracting features by embedding the data on the manifold. The goal of the feature extraction step is to produce a set of low dimensional observations that are able to provide adequate discrimination between UXO and non-UXO objects. Extracting features and reducing the dimension of the original (e.g., frequency-aspect) data not only alleviates some of the computational burden of the algorithm but can also result in a reduced space where classification can be done more accurately. Once features have been extracted, the next stage involves classifying the data as one of L possible object types. Each branch of the classifier is constructed from each unique object type and its ability to represent the extracted features is measured using a discriminant function. Finally, a classification label is assigned to the data based on a minimum error criterion [11].

4.2.1 Feature Extraction Using Manifold Learning

As described above, the first step in the proposed classification algorithm, as shown in Figure 6, is to perform feature extraction and dimensionality reduction using manifold learning. More specifically, Laplacian Eigenmaps [3] is used to embed the high-dimensional data onto a low-dimensional manifold. For this problem, it is assumed that we are given a set of N labeled training patterns $\{\mathbf{x}_i\}_{i=1}^N$ with each $\mathbf{x}_i \in \mathbb{R}^D$ belonging to one of L different objects or classes. Given this set of data, we wish to define a mapping $\mathbf{f}: \mathbb{R}^D \rightarrow \mathbb{R}^d$ with $d \ll D$ such that the feature vector $\mathbf{y}_i = \mathbf{f}(\mathbf{x}_i) \in \mathbb{R}^d$ captures the general behavior of the data over a low-dimensional manifold. Laplacian Eigenmaps is an algorithm that attempts to achieve this by defining features such that if the data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are near one another in the original high-dimensional space then their corresponding feature representations $\mathbf{y}_i$ and $\mathbf{y}_j$ will be near one another in the lower dimensional space as well.

Given this overall goal of maintaining relationships among neighboring data points in both spaces, the Laplacian Eigenmaps algorithm achieves this by first defining a weighted graph. The edges or neighboring points on this graph can be set by selecting the K nearest neighbors to each point. Each edge relating the data point $\mathbf{x}_i$ to $\mathbf{x}_j$ is then weighted using

$$w_{ij} = \begin{cases} k(\mathbf{x}_i, \mathbf{x}_j) & \text{if nodes } i \text{ and } j \text{ are connected} \\ 0 & \text{otherwise} \end{cases}$$

for some symmetric continuous function $k(\cdot, \cdot)$, typically chosen to be the Gaussian $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left\{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^2}\right\}$. Given this graph connecting each point to its nearest neighbors, we then wish to find the set of coordinates $\mathbf{Y} = [\mathbf{y}_1 \cdots \mathbf{y}_N] \in \mathbb{R}^N$ that embed each data point on the real line $\mathbb{R}$ by minimizing the objective function

$$J(\mathbf{Y}) = \sum_{i=1}^N \sum_{j=1}^N w_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 \quad (8)$$

Once again, the motivation behind this cost function is to produce a set of coordinates that are near one another if their corresponding data points are near one another in the original high dimensional space. Enforcing additional constraints that remove several trivial solutions associated with minimizing (8) yields the optimization problem

$$\begin{aligned} \min_{\mathbf{Y} \in \mathbb{R}^{d \times N}} \quad & \mathbf{Y} \mathbf{L} \mathbf{Y}^T \\ \text{s.t.} \quad & \mathbf{Y} \mathbf{D} \mathbf{Y}^T = \mathbf{I} \\ & \mathbf{Y} \mathbf{D} \mathbf{1} = \mathbf{0} \end{aligned} \quad (9)$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the graph Laplacian, $\mathbf{W}$ is a symmetric matrix with elements $[\mathbf{W}]_{ij} = w_{ij}$, $\mathbf{D} = \text{diag}\left\{\sum_j w_{j1}, \dots, \sum_j w_{jN}\right\}$, and $\mathbf{1} = [1 \cdots 1]^T$ is an all-one vector. The first constraint in (9) removes an arbitrary scaling factor in the embedding while the second eliminates a trivial solution which maps all features to a value of 1. The solution to (9) can be found [3] by solving the generalized eigenvalue problem $\mathbf{L}\mathbf{y} = \lambda \mathbf{D}\mathbf{y}$ and setting the coordinates equal to the eigenvector corresponding to the smallest eigenvalue λ . This process can be extended to embed the data in a d -dimensional space by extracting the eigenvectors associated with the smallest d non-zero eigenvalues.

Although this technique gives one the ability to learn the structure of an underlying manifold in the training data, the theory described above does not directly allow one to extract features for novel testing data. That is, the method does not allow one to embed unseen data on the manifold. One could simply attempt to achieve this by adding new rows and columns to the weight matrix $\mathbf{W}$ corresponding to the new data points and solving (9) every time one extracts new features.

However, this would change the solution to (9) attained during training, i.e. adding new data would modify the structure of the manifold. Moreover, the algorithm is not directly capable of synthesizing data in the original high-dimensional space based on its corresponding location on the manifold. Therefore, we seek a method that is able to embed test data on the manifold without modifying the structure of the manifold and is also capable of synthesizing data from the manifold.

To embed novel testing data onto the manifold, we employ the use of a latent (unseen) variable approach [12] which combines the advantages of nonlinear dimensionality reduction methods such as Laplacian Eigenmaps with those of latent variable models. For this method we assume that there exists a pair of datasets $\mathbf{X}_s \in \mathbb{R}^{D \times N}$ and $\mathbf{X}_u \in \mathbb{R}^{D \times M}$ representing data previously seen during the training process and the unseen novel testing data, respectively. With these two datasets defined, we then wish to find an embedding $\mathbf{Y}_u \in \mathbb{R}^{d \times M}$ for the unseen data while leaving the embedding $\mathbf{Y}_s \in \mathbb{R}^{d \times N}$ for the training data unchanged. The most natural way to accomplish this is to extend the problem in (9) by modifying the objective function as

$$\begin{aligned} F(\mathbf{Y}_u) &= \text{tr} \left([\mathbf{Y}_s \ \mathbf{Y}_u] \begin{bmatrix} \mathbf{L}_{ss} & \mathbf{L}_{su} \\ \mathbf{L}_{us} & \mathbf{L}_{uu} \end{bmatrix} \begin{bmatrix} \mathbf{Y}_s^T \\ \mathbf{Y}_u^T \end{bmatrix} \right) \\ &= \text{tr}(\mathbf{Y}_s \mathbf{L}_{ss} \mathbf{Y}_s^T) + 2\text{tr}(\mathbf{Y}_s \mathbf{L}_{su} \mathbf{Y}_u^T) + \text{tr}(\mathbf{Y}_u \mathbf{L}_{uu} \mathbf{Y}_u^T) \end{aligned} \quad (10)$$

where $\mathbf{L}_{ss}$ and $\mathbf{L}_{uu}$ are the graph Laplacians for $\mathbf{X}_s$ and $\mathbf{X}_u$, respectively, and $\mathbf{L}_{su} = \mathbf{L}_{us}^T$ is the graph Laplacian shared between them. Using the two differentiation identities

$$\begin{aligned} \frac{d\mathbf{A}\mathbf{X}^T}{d\mathbf{X}} &= \mathbf{A} \\ \frac{d\mathbf{X}\mathbf{A}\mathbf{X}^T}{d\mathbf{X}} &= 2\mathbf{X}\mathbf{A} \end{aligned}$$

one may solve for the unknown feature matrix $\mathbf{Y}_u$ by taking the derivative of (10) and setting it equal to the zero matrix $\mathbf{O}$, i.e.

$$\frac{dF(\mathbf{Y}_u)}{d\mathbf{Y}_u} = 2\mathbf{Y}_s \mathbf{L}_{su} + 2\mathbf{Y}_u \mathbf{L}_{uu} = \mathbf{O}$$

Solving this expression under the constraint that $\mathbf{Y}_s$ remains fixed leads to the solution

$$\mathbf{Y}_u = -\mathbf{Y}_s \mathbf{L}_{su} \mathbf{L}_{uu}^{-1} \quad (11)$$

If we now consider a single novel testing sample (i.e. $M = 1$) so that $\mathbf{x} = \mathbf{X}_u \in \mathbb{R}^D$ and $\mathbf{y} = \mathbf{Y}_u \in \mathbb{R}^d$, then the two graph Laplacians used in embedding this data point on the manifold are simply given by $\mathbf{L}_{su} = -\mathbf{w}_{su} = -[k(\mathbf{x}, \mathbf{x}_1) \cdots k(\mathbf{x}, \mathbf{x}_N)]^T \in \mathbb{R}^N$ and $\mathbf{L}_{uu} = \ell_{uu} = \mathbf{1}^T \mathbf{w}_{su} = \sum_{i=1}^N k(\mathbf{x}, \mathbf{x}_i)$. Substituting these expressions into the solution given in (11) yields the feature vector

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) = -\frac{1}{\ell_{uu}} \mathbf{Y}_s \mathbf{L}_{su} = \frac{\mathbf{Y}_s \mathbf{w}_{su}}{\mathbf{1}^T \mathbf{w}_{su}} = \sum_{i=1}^N \frac{k(\mathbf{x}, \mathbf{x}_i)}{\sum_{j=1}^N k(\mathbf{x}, \mathbf{x}_j)} \mathbf{y}_i$$

Using these results for the purposes of both embedding and reconstructing data points on the manifold finally suggests the pair of analysis and synthesis relationships

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) = \sum_{i=1}^N \frac{k(\mathbf{x}, \mathbf{x}_i)}{\sum_{j=1}^N k(\mathbf{x}, \mathbf{x}_j)} \mathbf{y}_i \quad (12)$$

$$\mathbf{x} = \mathbf{g}(\mathbf{y}) = \sum_{i=1}^N \frac{k(\mathbf{y}, \mathbf{y}_i)}{\sum_{j=1}^N k(\mathbf{y}, \mathbf{y}_j)} \mathbf{x}_i \quad (13)$$

Thus, new data points are embedded onto the low-dimensional manifold (analysis) and reconstructed (synthesis) using a convex combination of the samples used in the training set.

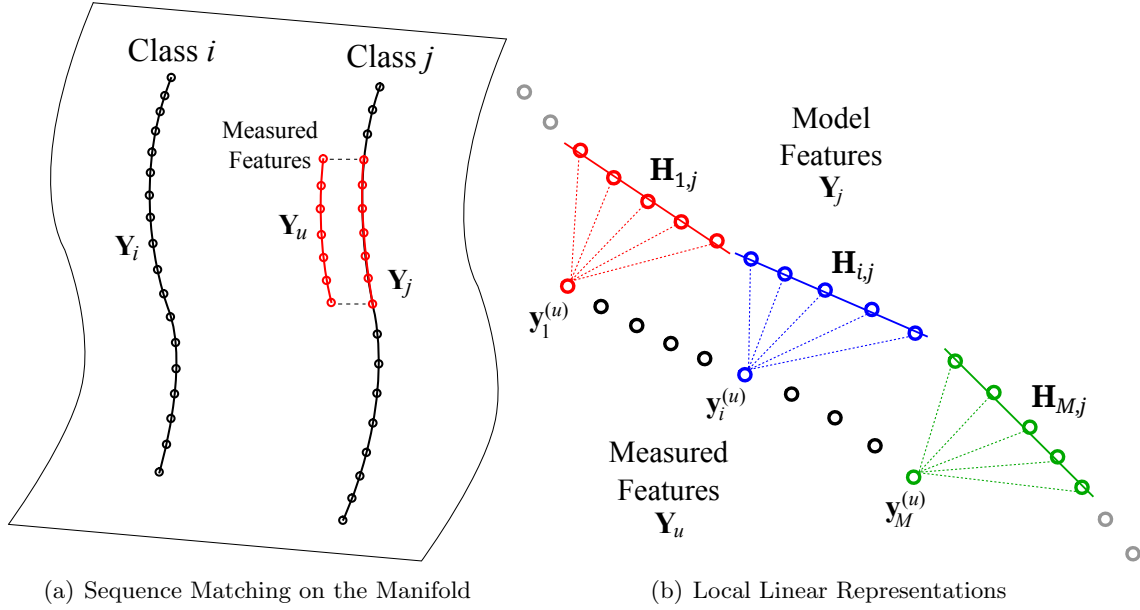


Figure 7: Performing classification on the manifold using the extracted sequence of features $\mathbf{Y}_u$.

4.2.2 Manifold-Based Classification Using Local Linear Representations

As discussed at the beginning of this section, the guiding principle behind many manifold learning algorithms is that the data lies on a low-dimensional manifold which parameterizes the inherently non-linear behavior of the data. However, it is often assumed that the manifold is locally linear so that, at least on small enough scales, one may measure distances between neighboring points using Euclidean means. In keeping with this same philosophy, we seek a classification method that relies on local measures when deciding the class label of the object. This is achieved by constructing local linear subspaces using the training data for each object type and selecting the one that is best capable of representing the data.

Given the features extracted from each object type in the training dataset, we wish to classify a set of newly acquired data to determine which of L different models the data belongs to. Let $\mathbf{Y}_j \in \mathbb{R}^{d \times N_j}$ denote the subset of low-dimensional training feature vectors associated with class or object type $j \in [1, L]$ and let $\mathbf{X}_u = [\mathbf{x}_1^{(u)} \cdots \mathbf{x}_M^{(u)}] \in \mathbb{R}^{d \times M}$ denote a sequence of newly-observed testing data vectors. It is assumed that the columns of matrix $\mathbf{X}_u$ form a naturally ordered sequence and that every column within the matrix corresponds to the same object. In this application, this sequence of vectors is formed by measuring an object's acoustic response in a linear synthetic aperture sonar (SAS) survey thus leading to a sequence of pings from the same object over a range of aspects. Using (9) to form the set of training feature vectors $\mathbf{Y} = [\mathbf{Y}_1 \cdots \mathbf{Y}_L]$ corresponding to all L classes, the first step in the classification algorithm is to extract features from the manifold corresponding to the newly acquired testing data. This is accomplished by individually applying each vector $\mathbf{x}_i^{(u)}$ in $\mathbf{X}_u$ to the analysis equation given in (12) using the training features in $\mathbf{Y}$ to produce the set of feature vectors $\mathbf{Y}_u$ in an unsupervised fashion.

As illustrated in Figure 6, once this set of feature vectors has been extracted from the manifold, the next step in the classification algorithm involves finding the subset of features in each data matrix $\mathbf{Y}_j$ that best match the sequence of extracted features corresponding to $\mathbf{X}_u$. This is accomplished by taking a length- M sliding window of the features in $\mathbf{Y}_j$ and computing the coherence between the features in that window and the extracted sequence of features in $\mathbf{Y}_u$. That is, if we let $\mathbf{Y}_j^{(m)} \in \mathbb{R}^{d \times M}$ denote the subset of features corresponding to the m^{th} window, the match between

these two sets of features is measured by finding the index m^* that solves the optimization problem

$$m^* = \arg \max_m \frac{\left| \langle \mathbf{Y}_j^{(m)}, \mathbf{Y}_u \rangle \right|^2}{\|\mathbf{Y}_j^{(m)}\|_F^2 \|\mathbf{Y}_u\|_F^2} \quad (14)$$

In this equation, the expression $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{B}^T \mathbf{A})$ represents the inner product between matrices $\mathbf{A}$ and $\mathbf{B}$ and $\|\mathbf{A}\|_F = \sqrt{\text{tr}(\mathbf{A}^T \mathbf{A})}$ is the Frobenius norm. Figure 7 (a) gives a depiction of this process for two sequences of features lying on the same manifold. For each class j , the coherence measure given in (14) is used to identify the subset of features in $\mathbf{Y}_j$ which provides the best match with the measured features in $\mathbf{Y}_u$. The main purpose behind using the coherence measure in (14) is to use the *a priori* knowledge that data is measured in an ordered sequence when defining each point's nearest neighbors. In this case, it makes sense to find where the data matrix $\mathbf{Y}_u$ as a whole best matches the training data rather than relying on point-wise estimates of proximity through Euclidean distance. Moreover, by doing so one is in essence able to capture local ping-by-ping dynamical behavior on the traces in the manifold domain.

Once it is determined where the sequence of features $\mathbf{Y}_u$ best fits with the training sequence from each class in $\mathbf{Y}_j$, the final stage of the classifier involves assigning a class label to the set of extracted features. This is accomplished by expressing each feature vector in $\mathbf{Y}_u$ as a linear combination of its P nearest neighbors in $\mathbf{Y}_j$ and finding the class which provides the best fit. Once again, the nearest neighbors in this case are determined based on the optimal index given in (14). Figure 7 (b) gives a depiction of this process where each point is connected to its $P = 5$ nearest neighbors using a dashed line. By finding local linear representations of each data point, the method is somewhat similar to making inference using other manifold-based techniques such as Local Linear Embedding (LLE) [6].

For each $\mathbf{y}_i^{(u)}$ in $\mathbf{Y}_u$ for $i = 1, \dots, M$, let $\mathbf{H}_{i,j} \in \mathbb{R}^{d \times P}$ denote the linear subspace formed from $\mathbf{y}_i^{(u)}$'s P nearest neighbors in $\mathbf{Y}_j$. That is, for each object type $j \in [1, L]$ and for every sample $i \in [1, M]$, a linear subspace is constructed using the training data that provides the best match with the extracted features. Note that there can and will be some overlap in the features used to construct each subspace for nearby samples, i.e. many of the feature vectors used to build $\mathbf{H}_{i,j}$ will also be used to build $\mathbf{H}_{i+1,j}$. Given these definitions, it is then assumed that each feature vector $\mathbf{y}_i^{(u)}$ obeys the following linear model

$$\mathbf{y}_i^{(u)} = \mathbf{H}_{i,j} \boldsymbol{\theta}_i + \mathbf{n}_i \quad (15)$$

where $\boldsymbol{\theta}_i \in \mathbb{R}^P$ is a vector of deterministic but unknown parameters and $\mathbf{n}_i \in \mathbb{R}^d$ is a vector containing as its elements *iid* realizations of a zero-mean normal random variable with unknown variance σ^2 . Thus, in this model the unknown vector $\boldsymbol{\theta}_i$ describes the coordinates in the local linear subspace $\mathbf{H}_{i,j}$ and the unknown variance σ^2 in some sense describes the scale in the error between measurement and model. Assuming that the sequence of features $\mathbf{y}_i^{(u)}$ for $i = 1, \dots, M$ are independently distributed according to the model in (15), the data matrix $\mathbf{Y}_u$ has the likelihood function

$$\ell_j(\mathbf{Y}_u) = \frac{1}{(2\pi\sigma^2)^{dM/2}} \exp \left\{ -\frac{1}{2\sigma^2} \sum_{i=1}^M \left\| \mathbf{y}_i^{(u)} - \mathbf{H}_{i,j} \boldsymbol{\theta}_i \right\|_2^2 \right\} \quad (16)$$

where in this case the notation $\|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^T \mathbf{x}}$ denotes the ℓ_2 norm of the vector $\mathbf{x}$. Replacing the unknown parameters $\boldsymbol{\theta}_i$'s and σ^2 in (16) with their maximum likelihood (ML) estimates

$$\hat{\boldsymbol{\theta}}_i = (\mathbf{H}_{i,j}^T \mathbf{H}_{i,j})^{-1} \mathbf{H}_{i,j}^T \mathbf{y}_i^{(u)} \quad (17)$$

$$\hat{\sigma}^2 = \frac{1}{dM} \sum_{i=1}^M \left\| \mathbf{y}_i^{(u)} - \mathbf{P}_{\mathbf{H}_{i,j}} \mathbf{y}_i^{(u)} \right\|_2^2 \quad (18)$$

results in the likelihood function

$$\begin{aligned}\ell_j(\mathbf{Y}_u) &= \frac{1}{(2\pi\hat{\sigma}^2)^{dM/2}} \exp \left\{ -\frac{1}{2\hat{\sigma}^2} \sum_{i=1}^M \left\| \mathbf{y}_i^{(u)} - \mathbf{H}_{i,j} \hat{\boldsymbol{\theta}}_i \right\|_2^2 \right\} \\ &= \left(\frac{2\pi}{dM} \sum_{i=1}^M \left\| \mathbf{y}_i^{(u)} - \mathbf{P}_{\mathbf{H}_{i,j}} \mathbf{y}_i^{(u)} \right\|_2^2 \right)^{-dM/2} \exp \left\{ -\frac{dM}{2} \right\}\end{aligned}\quad (19)$$

In expressions (18) and (19), the matrix $\mathbf{P}_{\mathbf{H}_{i,j}} = \mathbf{H}_{i,j} (\mathbf{H}_{i,j}^T \mathbf{H}_{i,j})^{-1} \mathbf{H}_{i,j}^T$ is the orthogonal projection matrix onto the P -dimensional subspace spanned by the training vectors in matrix $\mathbf{H}_{i,j}$. Ignoring all constants that aren't dependent on the data, one can see from (19) that finding the class index j that maximizes likelihood is equivalent to finding the class which solves the optimization problem

$$j^* = \arg \min_j \epsilon_j(\mathbf{Y}_u) \quad (20)$$

with the objective function

$$\epsilon_j(\mathbf{Y}_u) = \sum_{i=1}^M \left\| \mathbf{y}_i^{(u)} - \mathbf{P}_{\mathbf{H}_{i,j}} \mathbf{y}_i^{(u)} \right\|_2^2 \quad (21)$$

Thus, by projecting the observation onto the linear subspace $\mathbf{H}_{i,j}$ through the linear operator $\mathbf{P}_{\mathbf{H}_{i,j}}$, the minimization problem in (20) selects the class which best models the data in the sense of minimizing the error in representing the data. This process is illustrated in the last stage of the entire process shown in Figure 6.

5 Results and Discussion

5.1 Dataset Description

To test the ability of the proposed algorithms developed in Sections 4.1 and 4.2, both the detector and classifier were applied to the PondEx09 and PondEx10 datasets [13] collected at NSWC - Panama City, FL. The pond facility used in this experiment was designed to collect acoustical sonar data from underwater objects in a relatively controlled and clutter-free environment. Figures 8 (a) and (b) show the layout of the test setup for both experiments including the relative locations of the rail-mounted sonar system and the objects in the target field. As can be seen from Figure 8, both experiments consisted of a 21 m rail system collecting sonar returns from one or more targets located at a certain range from the rail. In most cases, the object was placed at a range of 10 m but there were several experiments where the object was located only 5 m from the rail. The 21m rail the sonar system is mounted on was fixed to eliminate platform motion as the sonar moves along its track, thereby increasing coherence between successive pings. The sonar transmit signal is a 6×10^{-3} s LFM pulse over 0.5-30 kHz with a 10% taper between the leading and trailing edges to minimize ringing in the transmitted signals. For the studies conducted in this report, 5 different object types were used. Table 1 gives a list of the object types used in this study which consists of three UXO objects of different material properties as well as two non-UXO objects, namely an aluminum cylinder and pipe.

As can be seen from Figure 8, both experiments involved collecting sonar backscatter from objects with varying shapes, sizes, and compositions, all of which are located approximately 10 m from the rail system. Nine total object orientations were used ranging from -80° to $+80^\circ$ in 20° increments where a 0° object orientation designates a configuration where the object's major axes are parallel to the rail system. Each run of data consists of 800 pings in which the sonar platform moved along the fixed rail in increments of 0.025 m, transmitting and receiving once for each fixed position. The data was sampled at 1 MHz and the sonar platform was tilted at a fixed 20° grazing angle for all runs (angle of the sonar main response axis with respect to the horizontal plane).

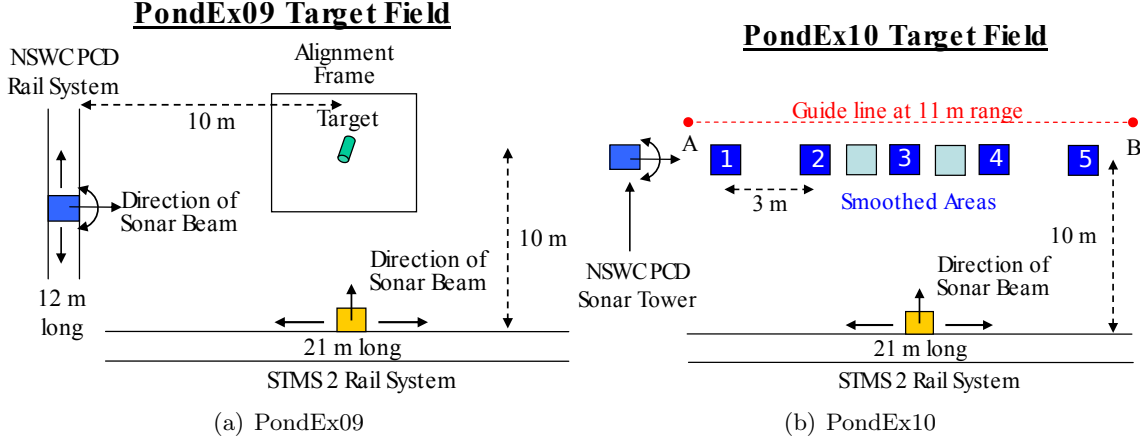


Figure 8: Layout of the target fields for the PondEx09 and PondEx10 datasets.

Table 1: Objects in the PondEx09-10 testing dataset

Object Type	Class
Aluminum UXO	UXO
Steel UXO	UXO
Real UXO	UXO
Aluminum Cylinder	Non-UXO
Aluminum Pipe	Non-UXO

5.2 Detection Results

5.2.1 Formation of the Test Statistic

To form detection decisions, each sonar return at a particular ping within the run was first applied to a pulse compression step by correlating the received waveform with the LFM transmit signal. Each matched filtered ping was then partitioned into overlapping windows of length $N = 281$ (cross-track) corresponding to a range resolution of approximately 0.25m. For every 0.25m in the direction of the rail system (along-track), the time series collected over a $M = 100$ ping window were appropriately lagged to account for time delays as discussed in Section 4.1.1. Recalling the diagram shown in Figure 1 (a), this is accomplished by accounting for the increase in path length due to the receiver's translation in along-track. Namely, if the area of the seafloor being tested is at range r and the received data at ping m has an along-track distance of d_m from the point of closest approach, then the range to the target at ping m is given by $r_m = \sqrt{r^2 + d_m^2}$. Since $r_m > r$ for $d_m > 0$, one must appropriately lag the extracted time series to account for the resulting time delay. After accounting for time delay, all $M = 100$ pings are stacked as columns to form the data matrix $\mathbf{X}$ given in (1) corresponding to each $0.25m \times 0.25m$ location within the target field. This was independently done for all $L = 5$ sensor elements in the linear array and the resulting data matrices are vertically stacked to form a composite data matrix. That is, if $\mathbf{X}_i$ represents the data matrix collected from sensor i then the composite data matrix $\mathbf{Z} = [\mathbf{X}_1^T \cdots \mathbf{X}_L^T]^T \in \mathbb{R}^{LN \times M}$ is formed. This composite data matrix is then applied to the test statistic given in (6) to determine whether or not that location within the field does indeed contain a target.

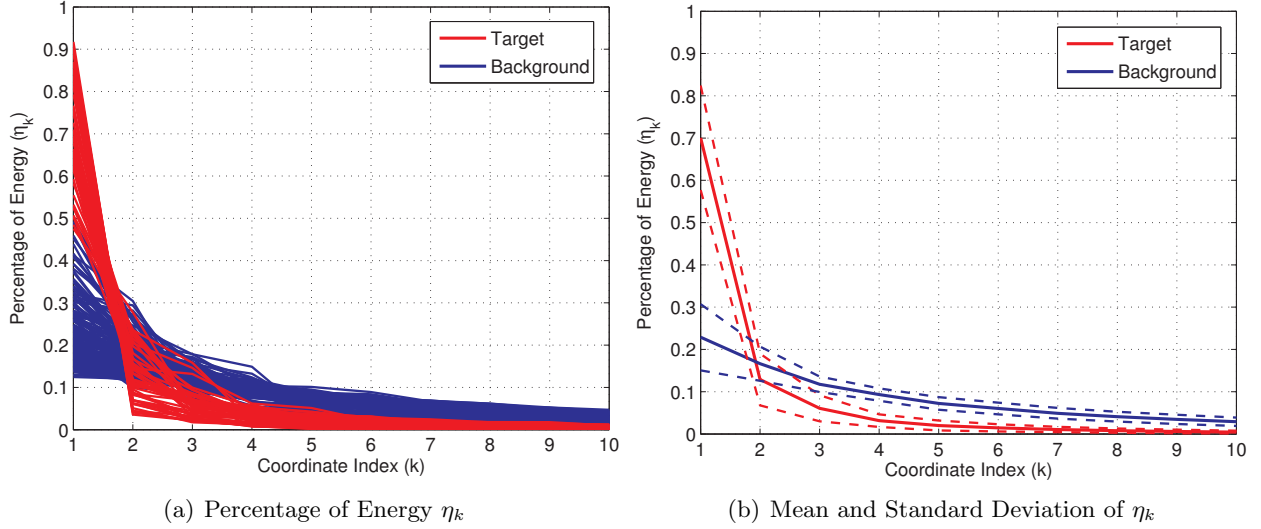


Figure 9: Percentage of energy in each singular component for both target (red) and background (blue).

5.2.2 Model Validation

Before applying the detection methodologies to this dataset, it is important to determine whether the data matches the fundamental assumptions made in Section 4.1 first. More specifically, we are interested in observing whether the measured response from munitions at multiple pings does in fact produce a data matrix $\mathbf{X}$ which exhibits a large rank-one component. To investigate this, a study was conducted where the data matrix $\mathbf{X}$ was found for all 50 observations of the targets (10 runs with 5 targets per run at a different orientation) in the PondEx10 dataset as well as for a randomly selected set of locations corresponding to background. For each instance of the data matrix $\mathbf{X}$, its thin SVD was computed such that $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ where $\mathbf{U} \in \mathbb{R}^{N \times N}$ and $\mathbf{V} \in \mathbb{R}^{M \times N}$ are orthogonal matrices and matrix $\mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_N)$ contains the singular values $\sigma_1 > \sigma_2 > \dots > \sigma_N$. The singular values of this matrix were then used to determine the percentage of energy in each component using

$$\eta_k = \frac{\sigma_k^2}{\sum_{i=1}^N \sigma_i^2}, \quad k = 1, \dots, N \quad (22)$$

That is, the value η_k computes the percentage of energy in the k^{th} component by normalizing the squared singular value σ_k^2 by the squared sum of them all. Figure 9 (a) plots η_k for $k = 1, \dots, 10$ for all 50 targets in the dataset (shown in red) and for all background locations (shown in blue) chosen for this test. Likewise, Figure 9 (b) plots the mean (using a solid line) plus or minus one standard deviation (using a dashed line) for the values of η_k plotted in Figure 9 (a). From the results shown in both these plots, one can clearly see that a majority of the energy for targets lies in the largest singular component ($k = 1$). However, for background the energy tends to be more uniformly distributed over all the coordinates. Hence, one can conclude that this rank-one assumption for targets is indeed useful for detection in this application.

5.2.3 PondEx Detection Results

The adaptive matched filter detector given in (6) was then applied to all 10 runs of the PondEx10 dataset and compared to the matched filter detector given in (7) where the target response vector $\mathbf{h}$ was trained using the matched filtered data of the targets from 3 different runs. More specifically,

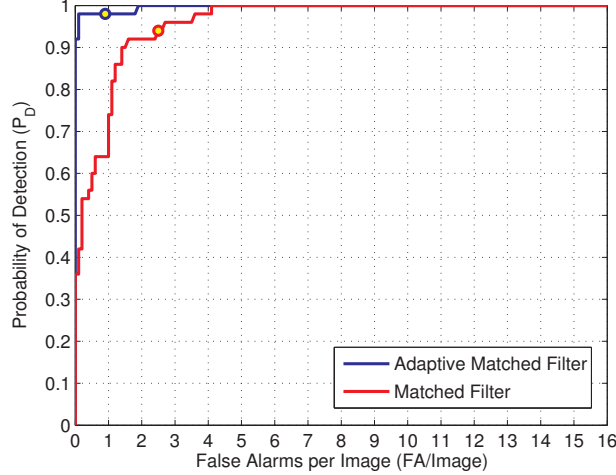
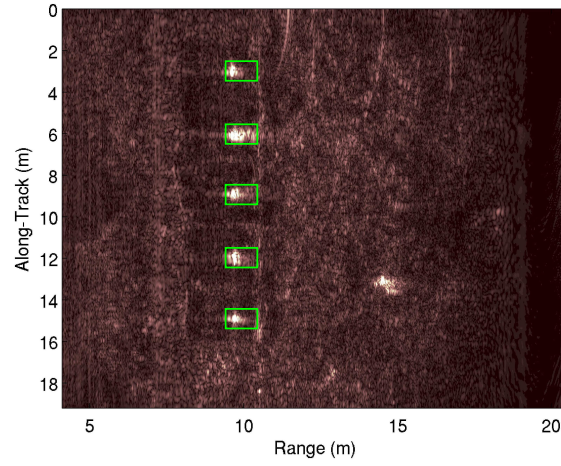


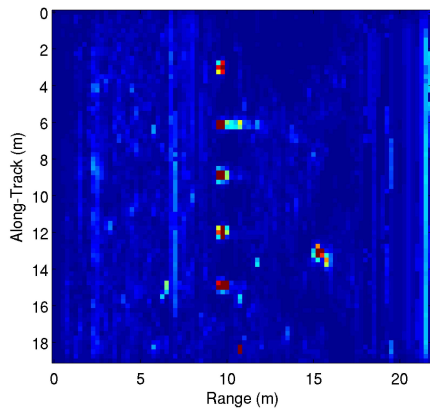
Figure 10: Receiver Operating Characteristic (ROC) curve for the PondEx10 dataset.

the training was accomplished by extracting the measured response from each of the 15 targets in the training set and using the principal left singular vector of the resulting data matrix as an estimate of the target response vector $\mathbf{h}$ in (7). Using all 50 targets in the dataset (10 runs $\times$ 5 objects per run) as well as a randomly selected set of locations corresponding to background, Figure 10 displays the Receiver Operating Characteristic (ROC) curve for both detectors. Also depicted in this figure using small circles is the knee-point of the ROC curve, i.e. the point where $P_d + P_{fa} = 1$, for each method. From the results of this figure, one can see that the adaptive matched filter given in (6) with a knee-point probability of $P_d = 98\%$ outperforms the trained matched filter given in (7) which achieves a lower knee-point probability of $P_d = 92\%$. Similar to the conclusions made using the results of Figure 5 (b), this is most likely due to the fact that, even when trained, it is very difficult to predict the target response from underwater munitions given the wide range of factors that can play a role.

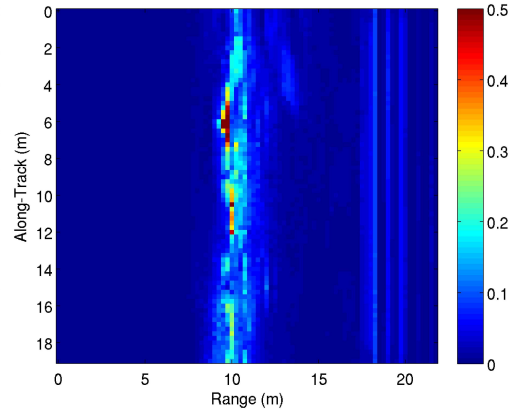
Both versions of the matched filter detector given in equations (6) and (7) were then applied to all 10 runs of the PondEx10 dataset with the thresholds for each method set to approximately achieve an average of two false alarms per image. The detectors were applied to each $0.25m \times 0.25m$ location in the target field of a given run. Overlapping areas that produced a likelihood ratio that exceeded the threshold for each respective detection method were then grouped into a single contact. If the location of that contact was within $1m$ of any of the known target locations in the dataset, that contact was labeled a target otherwise it was labeled a false alarm. Table 2 compares the detection and false alarm rates for both matched filter detectors. Although both methods achieve roughly the same false alarm rate, it is clear from the results of this table that the adaptive matched filter in (6) is much better at detecting the targets in this dataset. To see why this is so, Figure 11 (a) displays the beamformed SAS image corresponding to a fairly easy run where the objects, each of which is outlined with a green box in this image, have a 0° orientation. For every $0.25m \times 0.25m$ location, Figure 11 (b) displays the image of the likelihood ratio for the adaptive matched filter in (6) while Figure 11 (c) displays the same for the matched filter detector in (7). Likewise, Figures 12 (a)-(c) display the SAS and likelihood images for a more difficult run with an 80° object orientation. From these images of the likelihood ratio for each method, it is clear that the adaptive matched filter is not only capable of producing large likelihood ratio values for the targets in each of these two runs but also performs well at localizing the targets in each case. On the other hand, one can see that the matched filter also produces relatively high likelihood ratio values but does not do an adequate job of localizing the targets in the along-track dimension leading to poor performance. Thus, looking at the results given in Table 2 and Figures 10 – 12, one can see that the adaptive



(a) SAS Image



(b) Adaptive Matched Filter



(c) Matched Filter

Figure 11: Beamformed SAS image with 0° object orientation and images of the likelihood ratio for both versions of the matched filter detector.

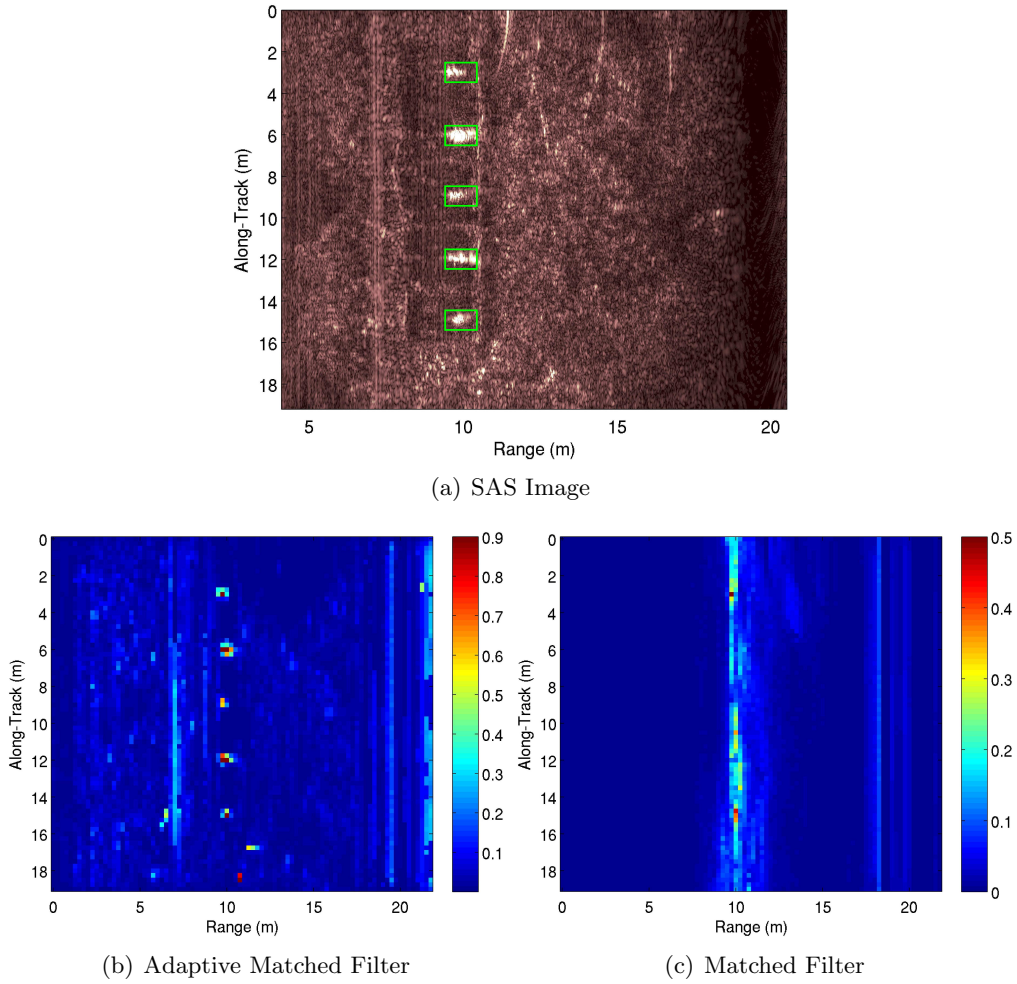


Figure 12: Beamformed SAS image with 80° object orientation and images of the likelihood ratio for both versions of the matched filter detector.

Table 2: Detection performance for both matched filter detectors.

	Targets Detected ($P_d\%$)	False Alarms per Image
Adaptive Matched Filter	49 (98%)	2.4
Matched Filter	28 (56%)	1.9

matched subspace detector performs very well at detecting the presence of UXO in the sonar returns collected in SAS applications. However, the high detection performance observed on this dataset is to be somewhat expected given the simplicity of the experiment, i.e. the lack of complicated seafloor clutter and the use of a rail to collect sonar returns.

5.3 Classification Results

5.3.1 Feature Extraction and Classification Procedures

(a) Manifold-Based Feature Extraction Process

Before reporting classification results for the PondEx datasets, we begin by giving a brief review

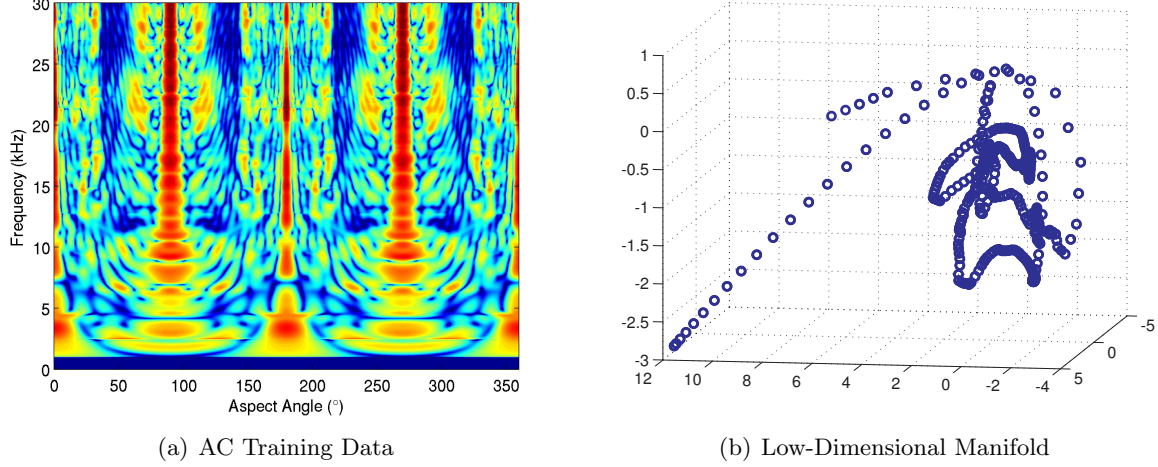


Figure 13: Training data and corresponding manifold for an aluminum cylinder object.

of the entire feature extraction and classification algorithm used to generate the results in this report. Since the objects in the pond experiments were placed relatively close to one another and the test facility can create competing backscatter interference, the measured data was first applied to a post-processing filtering algorithm [14] designed to retain the response of the object of interest while removing everything else. For this particular application, the original data vector $\mathbf{x}_i$ referred to throughout Section 4.2.1 represents the acoustic color response of a particular object at a given aspect angle (ping) extracted from the spatially filtered data. That is, each element of the vector $\mathbf{x}_i$ represents the magnitude of the frequency response over the 30 kHz bandwidth of the sonar system with a 50 Hz resolution resulting in a set of $D = 601$ dimensional data vectors in the ambient space. For all of the object types in the test set except for the real UXO, an acoustic color template containing the spectral information for that object over the entire 360° range in aspect were used to train the manifold mapping process. That is, the acoustic color templates from $L = 4$ objects at a 10 m range were used to construct the manifold: two UXO objects (aluminum and steel UXO) and two non-UXO objects (aluminum cylinder and pipe). Figures 13(a) and 14(a) give two examples of the training acoustic color data for an aluminum cylinder and UXO object, respectively. The training acoustic color data from all objects included in the training set are then used together to form the weight matrix $\mathbf{W}$ and graph Laplacian $\mathbf{L}$ used in (9). Here, the weighted graph was constructed by finding the $K = 64$ nearest neighbors to each training point and weighting them with the Gaussian $k(\mathbf{x}_i, \mathbf{x}_j) = \exp \left\{ -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\sigma^2} \right\}$ with smoothing parameter $\sigma^2 = 2.5$. Using the smallest $d = 48$ eigenvectors of the graph Laplacian $\mathbf{L}$, the set of coordinates $\mathbf{Y}$ associated with each object type are then used as training features to represent that particular object. Figures 13 (b) and 14 (b) give examples of the first three manifold coordinates for an aluminum cylinder and aluminum UXO object corresponding to their respective AC data given in Figures 13 (a) and 14 (a), respectively. From these two figures one can see that, although the manifold for each object does indeed form a definitive track as one moves from one aspect to another, that track is very complicated and somewhat chaotic.

When applying testing data to the classifier, an FFT is first applied to the filtered sonar returns and its magnitude taken to produce the unseen testing data $\mathbf{x}_i^{(u)}$ for $i = 1, \dots, M$ described in Section 4.2.2. That is, each element of the data vector $\mathbf{x}_i^{(u)}$ gives the magnitude response for the i^{th} ping at a particular frequency. Given the ping rate and beamwidth of the sonar system used here, a total of $M = 40$ sonar returns of the object are used to classify the object corresponding to a window spanning a range of approximately 20° in aspect angle. Figure 15 (a) and (b) give an

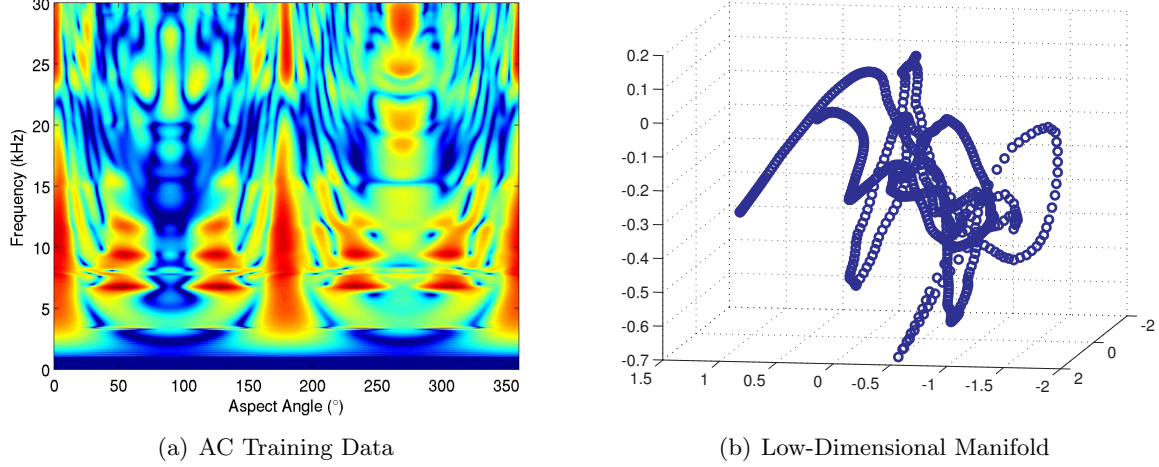


Figure 14: Training data and corresponding manifold for an aluminum UXO object.

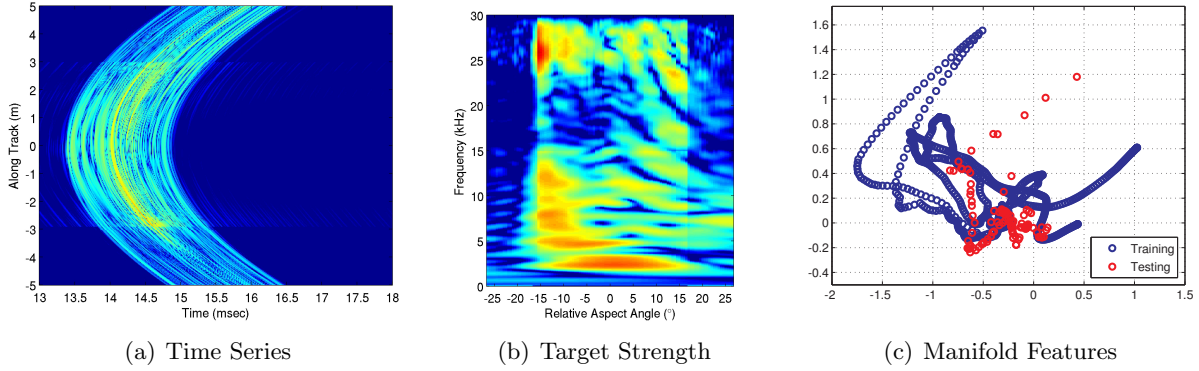


Figure 15: Filtered time series, target strength, and manifold features for an aluminum UXO.

example of the filtered time series and measured acoustic color for an aluminum UXO, respectively. This acoustic color data is then applied to the synthesis equation given in (13) to extract the first two manifold features and the resulting features for this UXO object are shown in Figure 15 (c).

(b) Manifold Domain Classification Process

Recalling the discussion given in Section 4.2.2, classification is achieved by first identifying the sequence of training features for each object type that best matches the extracted features using the statistical measure given in (14). In the contexts of this problem, this ordered sequence corresponds to the range of aspect angles observed for a given target so that finding the sequence that best matches the data in this case boils down to estimating the aspect angle of the target. Figure 16 gives a demonstration of this process for an aluminum cylinder observed at an 80° aspect angle. The top image in Figure 16 displays the measured acoustic color response for this target, the middle image displays the corresponding training data for an aluminum cylinder over the entire 360° range, and the bottom graph plots the coherence measure in (14) as one slides the measured response over the entire range of the training data. Note that although Figure 16 displays the raw acoustic color data for both the measured and training data, recall from Section 4.2.2 and also Figure 6 that the coherence measure is actually computed in the manifold feature domain. For this particular example, one can observe that the coherence statistic correctly estimates the true

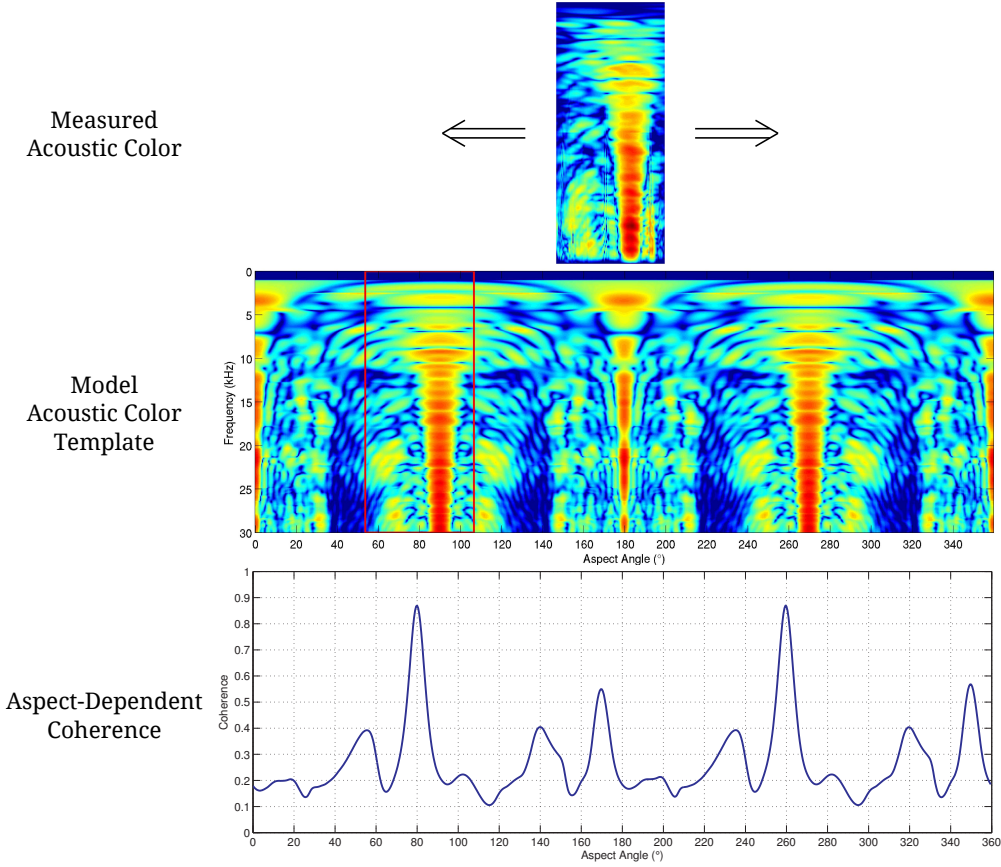


Figure 16: Estimating the aspect angle using coherence for a cylindrical object at with 80° aspect orientation.

aspect angle of the target as one observes distinct maxima at 80° and 260° aspect. The fact that there does exist two points where the statistic reaches a maximum is clearly due to the rotational symmetry exhibited by a cylindrical object.

Once the aspect has been estimated for each object type, the final step in the classification algorithm involves trying to represent each observed feature vector $\mathbf{y}_i^{(u)}$ for $i = 1, \dots, M$ using training features local to that observation. To accomplish this, the $P = 10$ training feature vectors corresponding to the j^{th} object that are nearest in aspect to $\mathbf{y}_i^{(u)}$ are used to construct the dictionary matrix $\mathbf{H}_{i,j}$ used in the linear model given in (15). More specifically, if we let $\{\mathbf{y}_k^{(i,j)}\}_{k=1}^P$ denote the P training features from object type j nearest to the extracted test feature $\mathbf{y}_i^{(u)}$, then this dictionary matrix is constructed as $\mathbf{H}_{i,j} = [\mathbf{y}_1^{(i,j)} \dots \mathbf{y}_P^{(i,j)}] \in \mathbb{C}^{d \times P}$. Using the same acoustic response given at the top of Figure 16 of an aluminum cylinder at an 80° aspect orientation, Figure 17 (a) and (b) plot the first two features of $\mathbf{y}_i^{(u)}$ with each aspect corresponding to a green dot. In both of these plots, the blue dots denote the subset of the trained manifold for two different objects that best matches the extracted features using (14): Figure 17 (a) shows the manifold for an aluminum cylinder while Figure 17 (b) shows that for an aluminum UXO. Given the extracted features as well as the trained manifold for each of these two objects, each red dot in both plots denotes the estimated feature vector $\hat{\mathbf{y}}_{i,j}^{(u)} = \mathbf{H}_{i,j} \hat{\boldsymbol{\theta}}_i = \mathbf{P}_{\mathbf{H}_{i,j}} \mathbf{y}_i^{(u)}$ where $\hat{\boldsymbol{\theta}}_i$ is the least-squares estimate of the unknown vector $\boldsymbol{\theta}_i$ given in (17). Physically speaking, the vector $\hat{\mathbf{y}}_{i,j}^{(u)}$ gives one the best estimate of the observation $\mathbf{y}_i^{(u)}$ in the linear subspace $\mathbf{H}_{i,j}$ by orthogonally projecting the observation into that subspace. Note that, given this definition, the optimization problem in (20) and (21) can be

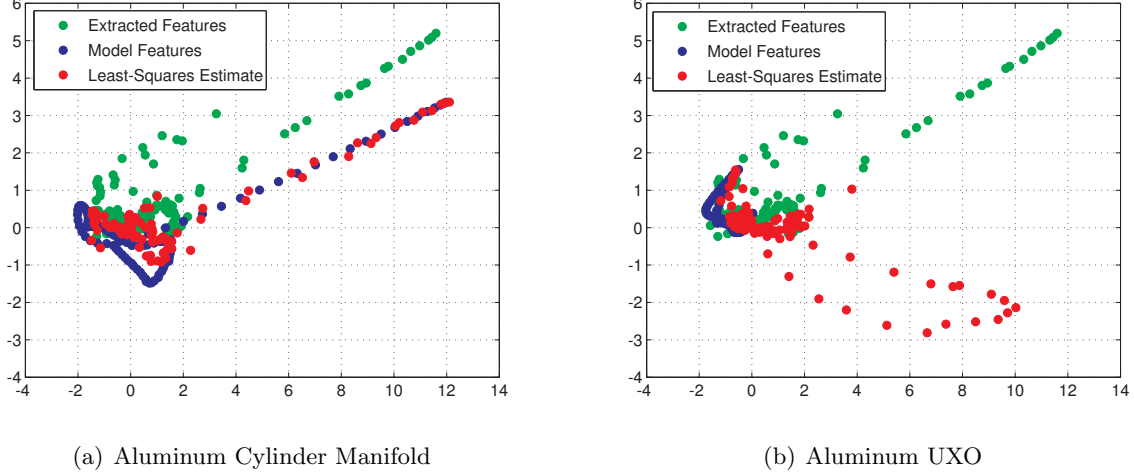


Figure 17: Comparing the extracted features to its least-squares estimate $\hat{\mathbf{y}}_{i,j}^{(u)} = \mathbf{H}_{i,j}\hat{\boldsymbol{\theta}}_i$ for two objects.

expressed the objective function in (21) can be rewritten in terms of these least-squares estimates as

$$j^* = \arg \min_j \epsilon_j(\mathbf{Y}_u)$$

$$\epsilon_j(\mathbf{Y}_u) = \sum_{i=1}^M \left\| \mathbf{y}_i^{(u)} - \hat{\mathbf{y}}_{i,j}^{(u)} \right\|_2^2$$

That is, the classifier selects the object that produces estimates that are the closest to the measured data. Looking at Figure 17, it is clear that the manifold corresponding to the aluminum cylinder does a much better job of producing estimates that reflect the extracted features than those produced by the manifold corresponding the aluminum UXO.

5.3.2 PondEx Classification Results

The proposed classification algorithm was then applied to the PondEx09 and PondEx10 datasets described in Section 5.1. For this study, training data from only four of the object types at a 10 m range and under proud conditions were used to train the manifold. Those objects include an aluminum cylinder, aluminum pipe, as well as an aluminum and steel UXO, i.e. no data for the 'real' UXO was used to train the manifold. This was done primarily to study whether or not replicas of different material properties can be used to adequately represent something previously unseen during training. The trained classifier was then applied to the filtered runs for the objects listed in Table 1 at a 10 m range and under proud conditions. Figure 18 displays the Receiver Operating Characteristic (ROC) curve for the classifier when applied to the Pond datasets. This figure plots the probability of correct classification P_{cc} (i.e. the percentage of UXO objects that are correctly classified as UXO) versus the probability of false alarm P_{fa} (i.e. the percentage of non-UXO objects that are incorrectly classified as UXO). The knee-point of the classifier (the point at which $P_{cc} + P_{fa} = 1$) is denoted in this plot using a blue circle in which case one can see that the classifier achieves a knee-point $P_{cc} = 90\%$.

Tables 3 and 4 give the classification and identification confusion matrices, respectively, for the Pond dataset. That is, for the entire set of 81 realizations of each object in the dataset, Table 3 gives the number of realizations classified as being either UXO or non-UXO while Table 4 gives the

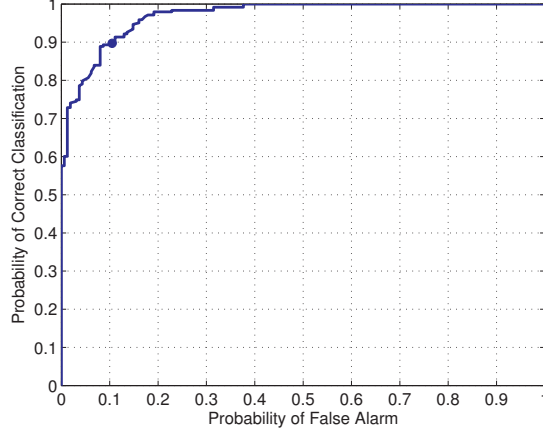


Figure 18: Receiver Operating Characteristic (ROC) Curve for the Pond Dataset.

Table 3: Classification Confusion Matrix

		UXO	Non-UXO
UXO	Aluminum UXO	76	5
	Steel UXO	72	9
	Real UXO	56	25
Non-UXO	Aluminum Cylinder	4	77
	Aluminum Pipe	6	75

Table 4: Identification Confusion Matrix

		Labeled ID			
		Aluminum UXO	Steel UXO	Aluminum Cylinder	Aluminum Pipe
True ID	Aluminum UXO	52	24	5	0
	Steel UXO	35	37	5	4
	Real UXO	30	26	11	14
	Aluminum Cylinder	0	4	72	5
	Aluminum Pipe	4	2	21	54

number of realizations assigned to each of the four object types modeled by the classifier. Overall, the classifier achieves a correct classification rate of $P_{cc} = 88\%$. From Table 3, one can see that the method performs well for the four objects specifically modeled by the classifier, namely the aluminum UXO, steel UXO, aluminum cylinder, and aluminum pipe, as the classifier achieves a correct classification rate of $P_{cc} \approx 93\%$ for those four objects. However, one can also see that the method doesn't achieve the same level of performance for the 'real' UXO with $P_{cc} \approx 69\%$. This is most likely due to the fact that this UXO object was the one object in this test that wasn't included in the training set leading to confusion at the classifier. Looking at the results displayed in Table 4, one can see that for the aluminum UXO, aluminum pipe, and especially the aluminum cylinder the classifier does a fairly good job of identifying which object type the data belongs to. However, one can also see that there is a fair amount of confusion for the steel and 'real' UXO objects when trying to decide the type of UXO object it corresponds to as approximately half of the test cases classified as UXO are labeled as being either an aluminum or steel UXO. This may

be due in part to the similarities among the training data for the aluminum and steel UXO objects making it difficult to discriminate between the two.

6 Conclusions and Implications for Future Research/Implementation

6.1 Conclusions and Discussions

The objectives addressed throughout the course of this project revolved around the development and testing of UXO detection and classification algorithms when applied to SAS data. The first task of this work specifically involved the development of a detection system capable of discriminating UXO-like objects from seafloor background. For this problem, our detection hypothesis is that the presence of munitions in the measured data will lead to the presence of a strong low-rank component in the data. Employing the use of the GLRT results in a matched filter detector that measures the percentage of energy that lies in the one dimensional subspace spanned by the principal left singular vector extracted from a data matrix containing the time aligned, pulse compressed data collected in a linear SAS survey. This test statistic remains invariant to scaling and both pre and post multiplication by any orthogonal matrix. This means that if changes in the characteristics of the sonar data due to things such as different environmental factors or different target conditions can be represented as transformations within this class, then the detector will remain robust. The performance of the detector is then demonstrated using both simulation as well as by applying it to data sets (PondEx 09 and PondEx10) collected in a freshwater pond consisting of a rail system collecting sonar backscatter from multiple munitions. Results of the simulation show that the use of a matched filter detector that relies on *a priori* knowledge of the target response will outperform its adaptive counterpart which computes an estimate of the target response using the singular value decomposition. However, the invariances of the adaptive matched filter make it more robust to model mismatch. This idea was explored further using the PondEx datasets by comparing the performance of the adaptive matched filter to a matched filter whose target response was trained using measured data of the targets from a few runs. Results of this study show that the adaptive version of the matched filter detector outperforms its trained counterpart. Once again, this is likely due to the fact it is very difficult to predict the response from various munitions in practice given the role various factors can play which in turn leads to high levels of model mismatch. The adaptive matched filter, on the other hand, attempts to estimate this response using measured data resulting in a more robust test statistic.

The next two tasks of this work involved the development and testing of a manifold-based feature extraction and classification strategy for discriminating among various UXO and non-UXO objects. The proposed feature extraction and classification system is designed based on the assumption that the data (in this case the AC data collected in a linear SAS survey) lies in some unknown low-dimensional subspace which is globally non-linear but locally linear. Based on this premise, a feature extraction technique using the Laplacian Eigenmaps [3] algorithm was proposed which produces a set of low-dimensional features that respect distances among training points in the high dimensional space by solving a generalized eigenvalue problem. By extending the algorithm to newly observed testing data yielded an out-of-sample embedding procedure for the purposes of feature extraction.

Given this set of low-dimensional features, the final task involved the development and preliminary testing of a multi-aspect classification technique used to discriminate among UXO and non-UXO objects. The first step in the algorithm involves identifying the subset of training features that best matches the extracted features using a coherence measure. In this particular application, this process corresponds to estimating the aspect orientation of the object. Once this sequence has been identified, the set of training features closest to each extracted feature are then used to form a local linear subspace to represent that feature vector. The most likely class label is then selected by finding the class that minimizes the error in representing the extracted features. The performance

of the classifier was then demonstrated on the PondEx datasets using the filtered sonar returns from several objects collected in a freshwater pond. For this study, the classifier was trained to discriminate two non-UXO objects (cylinder and pipe) from two UXO objects with different material properties. The method was then applied to testing data from these four objects as well as a 'real' and unseen (during training) UXO when each object was located 10 m from the rail and sitting proud on the seafloor. For the four objects modeled by the classifier, the method is able to correctly classify over 93% of the testing data in these datasets. Moreover, the method is able to correctly classify nearly 70% of the testing data for the 'real' UXO object.

6.2 Proposed Future Research and Development

6.2.1 Task 1: Detector's Performance Prediction and Optimization

Although the results in Section 5.2 show that the proposed test statistic in (6) is capable of detecting the presence of an object of interest in SAS data, one of the most difficult tasks in implementing any likelihood ratio test is defining thresholds that maintain a desired performance level. Most often, the detection threshold is chosen to maintain a predefined false alarm probability. However, to do so requires knowledge of the probabilistic behavior of the detector when applied to seafloor background. In this task, we propose to develop methods for accomplishing false alarm performance prediction and optimization for the adaptive matched filter detector. More specifically, performance prediction in this context involves developing theoretical models for how the detector behaves when applied to any particular background condition by analyzing the null distribution of the test statistic in (6). Performance optimization then involves developing methods for adapting this distributional model when encountering new environments.

Performance Prediction

As mentioned above, the first portion of this task involves developing a theoretical model of the distribution of the likelihood ratio for the purposes of predicting the false alarm rate of the detector. In the contexts of the statistical model described in Section 4.1.2, this corresponds to finding the distribution of the test statistic in (6) under the null hypothesis $\mathcal{H}_0$ that the scalar λ in (3) is zero. That is, we seek a univariate probability density $f(x)$ that describes the distribution of the likelihood ratio $\hat{\Lambda}$ under the assumption that the data matrix $\mathbf{X}$ in (1) consists of *iid* realizations of a zero-mean normal random variable.

Unfortunately, it is often times very difficult to derive an explicit expression for the null density $f(x)$ in many practical detection problems. However, one can often at the very least derive an expression for the moment generating function (MGF) $\phi(t)$ corresponding to the null distribution of the likelihood ratio. Given knowledge of a random variable's MGF, one may then employ methods such as saddlepoint approximations [15], [16] which approximates the density function $f(x)$ using the expression

$$\hat{f}(x) = \frac{1}{\sqrt{2\pi\psi''(\hat{t})}} \exp\{\psi(\hat{t}) - \hat{t}x\} \quad (23)$$

where $\psi(t) = \ln \phi(t)$ denotes the Cumulant Generating Function (CGF) [17] and the saddlepoint [16], $\hat{t}$, is the values of t such that $\psi'(t) = x$. Note that the notation ψ' and ψ'' denote the first and second-order derivatives of the cumulant generating function, respectively. Thus, for each value of the dependent variable x in (23), one may approximate the density function of the null distribution using the CGF $\psi(t)$ and its derivatives. Given a desired false alarm probability of $0 < \alpha < 1$, one may then find the threshold η that approximately achieves this false alarm probability by solving the equation

$$\int_{\eta}^{\infty} \hat{f}(x) dx = \alpha \quad (24)$$

where $\hat{f}(x)$ denotes the saddlepoint approximation in (23).

Performance Optimization

One of the possible pitfalls of the analysis given above is its reliance on the assumption that sonar backscatter from the seafloor background can be modeled as white Gaussian process. Any deviation from this behavior can result in a false alarm rate which is drastically higher than what was originally desired and can overwhelm the subsequent classification stage in the ATR algorithm. For this reason, the second portion of this task involves developing ways of adapting the approximation in (23) to better match the statistics of the likelihood ratio observed in a new environment. To accomplish this, let $\tilde{\psi}(t; \boldsymbol{\beta})$ denote a modification of the CGF used in (23) which is parameterized by the vector $\boldsymbol{\beta} \in \mathbb{R}^M$. Given the fact that derivatives of the CGF evaluated at $t = 0$ yield the cumulants of the distribution [17], i.e. $\psi^{(n)}(0) = \kappa_n$ with $\psi^{(n)}$ denoting the n^{th} order derivative of ψ and κ_n the n^{th} order cumulant, one may adapt the saddlepoint approximation by equating derivatives of the function $\tilde{\psi}$ with estimated sample cumulants $\hat{\kappa}_n$. These estimated sample cumulants can be computed using so-called k -statistics [18] which give one a minimum-variance, unbiased (MVUB) estimate of a random variable's cumulants. Given a set of likelihood ratio measurements $\{\Lambda_i\}_{i=1}^N$ collected from an operating environment and the n^{th} order sample moment

$$\hat{\mu}_n = \sum_{i=1}^N \Lambda_i^n,$$

the first few k -statistics are given by

$$\begin{aligned} \hat{\kappa}_1 &= \frac{1}{N} \hat{\mu}_1 \\ \hat{\kappa}_2 &= \frac{1}{N(N-1)} (N\hat{\mu}_2 - \hat{\mu}_1^2) \\ \hat{\kappa}_3 &= \frac{1}{N(N-1)(N-2)} (2\hat{\mu}_1^3 - 3N\hat{\mu}_1\hat{\mu}_2 + N^2\hat{\mu}_3) \\ &\vdots \end{aligned}$$

The modified CGF $\tilde{\psi}$ can then be made to match these estimated cumulants by finding the vector $\boldsymbol{\beta}$ that solves the constrained optimization problem

$$\begin{aligned} \min_{\boldsymbol{\beta} \in \mathbb{R}^M} \quad & \sum_{m=1}^M \left(\tilde{\psi}^{(m)}(0; \boldsymbol{\beta}) - \hat{\kappa}_m \right)^2 \\ \text{s.t.} \quad & \tilde{\psi}(0; \boldsymbol{\beta}) = 0 \end{aligned} \tag{25}$$

Upon finding the optimal parameter vector $\boldsymbol{\beta}^*$, one may then substitute the CGF $\tilde{\psi}(t; \boldsymbol{\beta}^*)$ into (23) to obtain a better estimate of the likelihood ratio's distribution observed in that particular environment. One may then adapt the threshold η by once again solving (24) but with the modified saddlepoint approximation.

6.2.2 Task 2: Adaptive Platform Motion Compensation Adaptive Matched Filter Detection

Although the results of Section 5.2 show that the detector performed very well when applied to the PondEx datasets, one of the concerns in its practical application is the effects of platform motion on the detection performance. Recall from Section 4.1.1 that in the development of the observation model we had assumed a receiver which collects sonar returns in regularly spaced intervals as it

travels linearly in along-track. In this ideal scenario, it becomes possible to reliably predict and account for relative time delays from ping to ping as depicted in Figure 1. This assumption was appropriate for the PondEx datasets as the hydrophone array was physically anchored to a rail system. When applied to systems deployed on Unmanned Underwater Vehicles (UUVs), however, deviations from this linear path due to platform motion will lead to unpredictable time delays, will invalidate the assumptions made in the development of the detector, and lead to sacrifices in detection performance as a result. Therefore, the objective of this task is to develop an adaptive matched filter detector that is capable of accounting for such delays to yield an algorithm that is more robust to the effects of platform motion and instability. *As opposed to alternative corrective approaches which typically rely on measurements taken from Inertial Measurement Units onboard the UUV, the proposed approach relies on the raw sonar data itself.*

To accomplish this task, we seek a statistically motivated technique for estimating and accounting for unknown time delays present in the data record collected from a SAS system. For this purpose, we propose to investigate the use of generalized coherence [19] as a means of finding time delays which maximize the linear dependence among the time series collected over a multiple ping window. Recall from Section 4.1.2 that in the development of the detector it was assumed that a SAS system collects multiple length N time series over M pings. Let $\mathbf{x}_n \in \mathbb{R}^N$ denote the length N pulse compressed response collected at the n^{th} ping. Assume that the data matrix $\mathbf{X}_n = [\mathbf{x}_0 \cdots \mathbf{x}_{n-1}] \in \mathbb{R}^{N \times n}$ contains the pulsed compressed responses from n pings which are previously coregistered in time and consider adding the vector $\mathbf{x}_n$ (not coregistered yet) to form the data matrix $\mathbf{X} = [\mathbf{X}_n \mathbf{x}_n]$. Note that by coregistered we mean that each ping is temporally aligned with one another such as that shown on the right hand side of Figure 1. Using the properties of 2×2 block matrices, the coherence between the data matrix $\mathbf{X}_n$ and the n^{th} ping $\mathbf{x}_n$ can be written as

$$\begin{aligned} C_n &= 1 - \frac{\det(\mathbf{X}^T \mathbf{X})}{\|\mathbf{x}_n\|^2 \det(\mathbf{X}_n^T \mathbf{X}_n)} \\ &= 1 - \frac{(\|\mathbf{x}_n\|^2 - \mathbf{x}_n^T \mathbf{X}_n (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{x}_n) \det(\mathbf{X}_n^T \mathbf{X}_n)}{\|\mathbf{x}_n\|^2 \det(\mathbf{X}_n^T \mathbf{X}_n)} \\ &= \frac{\mathbf{x}_n^T \mathbf{P}_{\mathbf{X}_n} \mathbf{x}_n}{\mathbf{x}_n^T \mathbf{P}_{\mathbf{X}_n} \mathbf{x}_n + \mathbf{x}_n^T \mathbf{P}_{\mathbf{X}_n}^\perp \mathbf{x}_n} \end{aligned} \quad (26)$$

where $\mathbf{P}_{\mathbf{X}_n} = \mathbf{X}_n (\mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T$ is the orthogonal projection onto the n dimensional subspace spanned $\langle \mathbf{X}_n \rangle$ spanned by the columns of matrix $\mathbf{X}_n$. By projecting onto this subspace, the ratio in (26) essentially represents the percentage of energy of $\mathbf{x}_n$ which lies in the subspace $\langle \mathbf{X}_n \rangle$. The higher the value of this statistic, the greater the linear dependence among $\mathbf{x}_n$ and the columns of $\mathbf{X}_n$.

Given this setup, we then wish to find the lagged version of vector $\mathbf{x}_n$ that maximizes the coherence in (26) in the hope that this yields the time delay where the vector $\mathbf{x}_n$ is most likely temporally coregistered with the columns of matrix $\mathbf{X}_n$. Consider the vector $\mathbf{y}_n[k] = \mathbf{S}_N^k \mathbf{x}_n$ with $\mathbf{S}_N$ an $N \times N$ cyclic shift matrix

$$\mathbf{S}_N = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

Note that the operation $\mathbf{S}_N \mathbf{x}$ cyclically shifts the elements of vector $\mathbf{x}$ down by one location while raising that matrix to the power k and computing $\mathbf{S}_N^k \mathbf{x}$ repeats this process k times over. Thus,

the vector $\mathbf{y}_n[k]$ is effectively a temporally lagged version of vector $\mathbf{x}_n$. Substituting this lagged vector into (26), one may temporally coregister the vector $\mathbf{x}_n$ with the columns of matrix $\mathbf{X}_n$ by finding the lag index k which maximizes the coherence statistic

$$C_n[k] = \frac{\mathbf{y}_n^T[k] \mathbf{P}_{\mathbf{X}_n} \mathbf{y}_n[k]}{\mathbf{y}_n^T[k] \mathbf{P}_{\mathbf{X}_n} \mathbf{y}_n[k] + \mathbf{y}_n^T[k] \mathbf{P}_{\mathbf{X}_n}^\perp \mathbf{y}_n[k]} \quad (27)$$

Upon finding the optimal lag k^* , one may then add the vector $\mathbf{x}_n^* = \mathbf{S}_N^{k^*} \mathbf{x}_n$ to the matrix $\mathbf{X}_n$ and repeat the process for the vector at the next ping $\mathbf{x}_{n+1}$.

The entire algorithm works as follows. Given an initial pulse compressed response $\mathbf{x}_0$, one begins by constructing the data matrix $\mathbf{X}_1 = \mathbf{x}_0 \in \mathbb{R}^N$. Given the next ping in the sequence, $\mathbf{x}_1$, one computes the statistic $C_1[k]$ in (27), selects the lag k that maximizes coherence, and adds the temporally coregistered vector $\mathbf{x}_1^*$ to matrix $\mathbf{X}_1$ to yield $\mathbf{X}_2 = [\mathbf{X}_1 \mathbf{x}_1^*] \in \mathbb{R}^{N \times 2}$. The new data matrix $\mathbf{X}_2$ along with the vector from the next ping $\mathbf{x}_2$ are used to compute the statistic $C_2[k]$ and its optimally lagged version $\mathbf{x}_2^*$ is used to form the data matrix $\mathbf{X}_3 = [\mathbf{X}_2 \mathbf{x}_2^*] \in \mathbb{R}^{N \times 3}$. This process is repeated until all M pings have been added to matrix $\mathbf{X}_n$ which is subsequently applied to the detector described in Section 4.1.2.

Although the coherence statistic in (27) would no doubt be able to correctly coregister sonar returns from munitions lying on the seafloor, one of the downsides of the proposed method is that it will also attempt to correlate data collected from the seafloor background as well. Although one would expect the improvement in detectability to be greater for target than background, coregistering background data may inadvertently lead to a strong low-rank component in the data and an increase in the false alarm rate when using the detector in (6). Assuming that the pulse compressed response $\mathbf{x}_n$ when collected from the background can be modeled as white Gaussian noise (note that the same assumption was made when developing the detector in Section 4.1.2), then it is well-known [19] that the coherence statistic in (26) is distributed as $C_n \sim \text{Beta}(\frac{1}{2}n, \frac{1}{2}(N-n))$ with $\text{Beta}(\alpha, \beta)$ denoting a beta distribution with parameters α and β . As this distribution is only dependent on the parameters N and n , it is worth noting that scale invariance properties of (26) yield a statistic whose distribution is independent of the white noise variance. Using the fact that the coherence statistic is beta distributed, one may then use the theory of order statistics [20] to show that the maximum coherence in (27), $C_n^* = \arg \max_k C_n[k]$, has the probability density function

$$f_{C_n^*}(x) = N \left[I \left(x; \frac{1}{2}n, \frac{1}{2}(N-n) \right) \right]^{N-1} f \left(x; \frac{1}{2}n, \frac{1}{2}(N-n) \right) \quad (28)$$

where $f(x; \alpha, \beta)$ and $I(x; \alpha, \beta)$ represent the density function for a beta distribution and the incomplete beta function, respectively,

$$\begin{aligned} f(x; \alpha, \beta) &= \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} \\ I(x; \alpha, \beta) &= \frac{1}{B(\alpha, \beta)} \int_0^x t^{\alpha-1} (1-t)^{\beta-1} dt \end{aligned}$$

Note that, in these two expressions, the term $B(\alpha, \beta)$ represents the beta function with parameters α and β

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1} (1-t)^{\beta-1} dt$$

With a time series length of $N = 256$, Figure 19 gives several examples of the density function in (28) for various values of n . One can observe from this plot that the larger the value of n , i.e. the more columns that are included in the data matrix $\mathbf{X}_n$, the higher the maximum coherence C_n^* tends to be. The whole point of this argument is that, by knowing how the statistic C_n^* behaves probabilistically when the data vector $\mathbf{x}_n$ contains only noise, one can derive a selective sampling

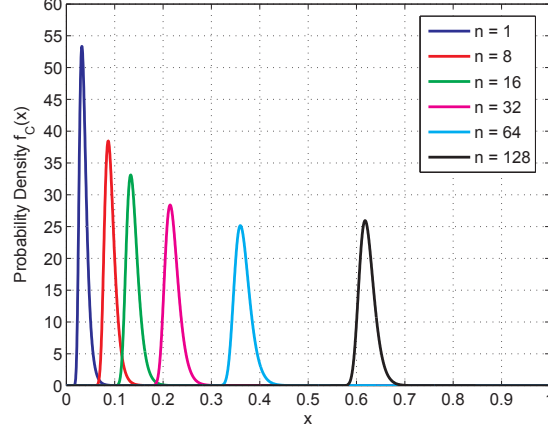


Figure 19: Probability densities $f_{C_n^*}(x)$ for the order statistic $C_n^* = \arg \max_k C_n[k]$ with $N = 256$ and for various values of n .

scheme which chooses if and when one should attempt to coregister the data. Those pings most likely to contain only noise as predicted by (28) can be removed from the process to avoid the inadvertent coregistration of background data.

6.2.3 Task 3: Computationally Efficient Manifold-Based Classification using Subspace Averaging

Recall from Section 4.2.2 that the manifold-based classifier employed in this work relies on having to build a local linear subspace $\langle \mathbf{H}_{i,j} \rangle$ consisting of P training vectors associated with class j which are nearest to each extracted feature $\mathbf{y}_i^{(u)}$. Although the results of Section 5.3 show that this process is indeed capable of adequately representing the features from different objects for classification purposes, one of the more prohibitive computational aspects of the algorithm is having to build the orthogonal projection matrix $\mathbf{P}_{\mathbf{H}_{i,j}}$ in (21) for all M feature vectors that are used to make a classification decision. Thus, the goal of this task involves the development and testing of techniques capable of producing a single projection operator $\mathbf{P}_{\tilde{\mathbf{H}}_j}$ which best approximates the entire collection of subspaces $\{\mathbf{H}_{i,j}\}_{i=1}^M$. By doing so, we will be able to build a more computationally efficient classification algorithm without having to sacrifice classification performance.

For notational convenience, we will drop the class index j in what follows with the understanding that this process must be applied to each class. Given the set of subspaces $\{\mathbf{H}_m\}_{m=1}^M$, let $\mathbf{V}_m \in \mathbb{R}^{d \times P}$ denote a matrix whose columns form a unitary basis for the subspace $\langle \mathbf{H}_m \rangle$ with $\mathbf{P}_m = \mathbf{V}_m \mathbf{V}_m^T$ its idempotent orthogonal projection matrix. Assuming that the dimension of the union of these subspaces has dimension $D = \dim(\cup_{m=1}^M \langle \mathbf{V}_m \rangle)$, we not only wish to find the subspace $\langle \mathbf{V}_s \rangle$ that best approximates this collection of subspaces but also wish to determine its optimal dimension s by solving the optimization problem

$$(s^*, \mathbf{V}_s^*) = \arg \min_{s \in [0, D], \mathbf{V}_s \in \mathbb{R}^{d \times s}} \frac{1}{M} \sum_{m=1}^M d(\langle \mathbf{V}_s \rangle, \langle \mathbf{V}_m \rangle)^2 \quad (29)$$

In [21] the authors suggest using the extrinsic distance metric

$$d(\langle \mathbf{V}_s \rangle, \langle \mathbf{V}_m \rangle) = \|\mathbf{P}_s - \mathbf{P}_m\|_F$$

with the projection matrix $\mathbf{P}_s = \mathbf{V}_s \mathbf{V}_s^H$. This distance metric gives one a measure of how close one subspace is to another and was shown in [21] to be closely related to the cosines of the principle

angles between the two subspaces. With this notion of distance, the optimization problem in (29) can be rewritten as

$$(s^*, \mathbf{P}_s^*) = \arg \min_{s \in [0, D], \mathbf{P} \in \mathcal{P}_s} \frac{1}{M} \sum_{m=1}^M \|\mathbf{P} - \mathbf{P}_m\|_F^2 \quad (30)$$

where $\mathcal{P}_s$ denotes the set of all projection matrices of rank s .

Given the optimization problem in (30), we can begin by considering the simpler problem: given $s \in [0, D]$, find the subspace $\langle \mathbf{V}_s \rangle$ that minimizes the cost function

$$\begin{aligned} E(s) &= \min_{\mathbf{P}_s \in \mathcal{P}_s} \frac{1}{M} \|\mathbf{P} - \mathbf{P}_m\|_F^2 \\ &= \min_{\mathbf{P}_s \in \mathcal{P}_s} \text{tr} \left[(\mathbf{P} - \bar{\mathbf{P}})^T (\mathbf{P} - \bar{\mathbf{P}}) + \bar{\mathbf{P}} - \bar{\mathbf{P}}^2 \right] \end{aligned} \quad (31)$$

where the symmetric matrix

$$\bar{\mathbf{P}} = \frac{1}{M} \sum_{m=1}^M \mathbf{P}_m$$

is the average of the M projection matrices. Writing the eigendecomposition of this average projection matrix as $\bar{\mathbf{P}} = \mathbf{F} \mathbf{K} \mathbf{F}^T$, where $\mathbf{K} = \text{diag}(k_1, \dots, k_D)$ with $1 \geq k_1 \geq k_2 \geq \dots \geq k_D$, one may ignore all constant terms and solve (31) by finding the unitary matrix $\mathbf{U}_s$ that satisfies

$$\max_{\mathbf{U}_s \in \mathbb{R}^{d \times s}} \mathbf{U}_s^T \mathbf{F} \mathbf{K} \mathbf{F}^T \mathbf{U}_s \quad (32)$$

In [21] it was shown that the solution to (32) is given by any unitary matrix whose column space is the same as the subspace spanned by the s principal eigenvectors of $\mathbf{F}$, i.e.

$$\mathbf{U}_s^* = [\mathbf{f}_1 \ \mathbf{f}_2 \ \dots \ \mathbf{f}_s] = \mathbf{F}_s \quad (33)$$

and the optimal projection matrix is given by $\mathbf{P}_s^* = \mathbf{U}_s^* (\mathbf{U}_s^*)^T$.

Given the optimal subspace $\mathbf{U}_s^*$ and its associated projection matrix, the next step involves finding the optimal subspace dimension s . Plugging this solution into (31), the minimum mean squared error (MSE) $E(s)$ can be expressed in terms of the eigenvalues of matrix $\bar{\mathbf{P}}$ as

$$E(s) = \sum_{i=1}^s (1 - k_i) + \sum_{i=s+1}^D k_i \quad (34)$$

As discussed in [21], a simple analysis shows that $E(s+1) \leq E(s)$ if $k_{s+1} \geq \frac{1}{2}$. Therefore, the optimal fitting rule amounts to selecting the largest s such that $k_s > \frac{1}{2}$. Interestingly, the MSE expression in (34) admits a bias-variance tradeoff interpretation [21] in which the first term is the variance due to the selected dimensions of $\bar{\mathbf{P}}$ whereas the second term is a squared-bias cost associated with the discarded dimensions.

A brief description of the algorithm is as follows. Given the collection of subspaces $\{\mathbf{H}_{i,j}\}_{i=1}^M$, one begins by finding the average subspace matrix $\bar{\mathbf{P}} = (1/M) \sum_{i=1}^M \mathbf{H}_{i,j}$. An eigendecomposition of this matrix is then taken such that $\bar{\mathbf{P}} = \mathbf{F} \mathbf{K} \mathbf{F}^T$ and the columns of $\mathbf{F}$ are sorted in a descending order based on their corresponding eigenvalues k_i for $i = 1, \dots, D$. The optimal subspace $\mathbf{U}_s^*$ in (33) is then found by retaining those eigenvectors whose eigenvalues are greater than $\frac{1}{2}$. One may then replace the projection matrix $\mathbf{P}_{\mathbf{H}_{i,j}}$ in (21) with the orthogonal projection $\mathbf{P}_s^*$. As can be seen, the proposed technique is very ubiquitous in that it non-parametrically determines a subspace which best represents a given set of measured subspaces. Moreover, the technique admits a natural rule for determining the optimal subspace dimension to minimize the MSE in the approximation.

6.2.4 Task 4: Comprehensive Testing & Evaluation

Throughout this two-year research, we shall test and evaluate the performance of the developed detection and classification methods developed on several real (e.g. TREX13, BayEX, and BOSS) and synthetic (e.g. PC SWAT) sonar datasets collected in different environmental and background conditions. The specific issues that will be thoroughly studied include:

1. Developing the theoretical null distribution of the test statistic in (6) for false alarm performance prediction. Once again, this involves deriving statistical properties of the likelihood ratio, such as the CGF $\psi(t)$, under a theoretical assumption that sonar data collected from the seafloor can be modeled as Gaussian white noise. This will also involve testing the ability of the saddlepoint approximation in (23) to produce a threshold capable of achieving a desired false alarm rate. The ability of the theoretical null distribution to achieve this desired false alarm rate will be tested by applying the technique to different environmental and operating conditions using the above mentioned datasets.
2. Based on the theoretical CGF $\psi(t)$ determined above, design a modified CGF $\tilde{\psi}(t; \theta)$ capable of adapting to the statistical properties of the likelihood ratio observed in a new environment for performance optimization. This will also involve testing the ability of the cumulant matching technique in (25) to adapt to changing environmental conditions. The ability of the proposed adaptive thresholding technique will be tested using the UXO datasets mentioned above and its ability to achieve a desired false alarm rate will be compared to that based on the theoretical null distribution.
3. Develop an adaptive temporal coregistration technique based on the principles of generalized coherence for autonomously accounting for the effects of platform motion to yield a more robust detection technique. The ability of the proposed technique to adequately correct for unknown time delays due to platform motion will be tested and the improvement in detection performance brought by including it as part of the detector will be compared to that where it is absent. Moreover, we plan to test the background coherence distribution given in (28) and its ability to selectively remove pings corresponding to background alone by measuring the improvement in detection performance brought by including it in the algorithm.
4. Apply the proposed subspace averaging and order fitting procedure to the manifold features used to represent various munitions objects. The main objective of this task will be to reduce the computational burden associated with defining a projection matrix $\mathbf{P}_{\mathbf{H}_{i,j}}$ in (21) for every single feature vector by replacing it with a single projection matrix $\mathbf{P}_s^*$ using the optimal subspace in (33) without adversely affecting the performance of the classifier. This will be investigated by testing gains in computational efficiency versus classification performance and comparing the results with those obtained using the original method.
5. Investigate the use of model-based simulation software for the purposes of building and studying the manifold structures corresponding to various UXO and non-UXO objects. This task will involve working with our counterparts (possibly through a subcontract) at NSWC-Panama City and/or APL-University of Washington to apply the manifold-based classification techniques developed here to synthetic physics-based data generated from software suites developed by each lab. The purpose of this work will be to study if synthetic data can indeed be used to train manifolds for classification purposes and to compare each algorithm to identify the pros and cons of each.
6. Prepare first-year interim and final reports to document all the developments and results of this two-year research. We plan to publish both the theoretical techniques developed

during this work as well as its application to various UXO datasets in IEEE Transactions and Journals in related fields.

7. As part of this two-year research, we plan to work very closely with our collaborators at NSWC-Panama City to transition the developed code for their evaluation and testing on the Navy's testbed systems such as the Modular Algorithmic Testbed Suite (MATS). Moreover, we plan on working with NSWC to apply the developed algorithms to datasets collected from platforms currently deployed as well as future platforms awaiting seatrials.

References

- [1] R. Muirhead, *Aspects of Multivariate Statistical Theory*. John Wiley and Sons, New Jersey, 2005.
- [2] J. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*. Springer, 2007.
- [3] M. Belkin and P. Niyogi, “Laplacian Eigenmaps and spectral techniques for embedding and clustering,” *Advances in Neural Information Processing Systems*, vol. 14, pp. 586–591, 2001.
- [4] J. Tenenbaum, V. de Silva, and J. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, pp. 2319–2323, December 2000.
- [5] —, “Global versus local methods for nonlinear dimensionality reduction,” *Advances in Neural Information Processing Systems*, vol. 15, June 2003.
- [6] S. Roweis and L. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, pp. 2323–2326, December 2000.
- [7] L. Saul and S. Roweis, “Think globally, fit locally: unsupervised learning of low dimensional manifolds,” *Journal of Machine Learning Research*, vol. 4, pp. 119–155, December 2003.
- [8] K. Weinberger and L. Saul, “Unsupervised learning of image manifolds by semidefinite programming,” *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 988–995, June 2004.
- [9] C. Eckart and G. Young, “The approximation of one matrix by another of lower rank,” *Psychometrika*, vol. 1, no. 3, pp. 211–218, September 1936.
- [10] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Addison-Wesley, 1991.
- [11] S. M. Kay, *Fundamentals of Statistical Signal Processing, Vol. II: Detection Theory*. Prentice Hall, 1998.
- [12] M. A. Carreira-Perpinan, *Continuous Latent Variable Models for Dimensionality Reduction and Sequential Data Reconstruction*. PhD Thesis, Dept. of Computer Science, University of Sheffield, UK, 2001.
- [13] S. Kargl, A. Espana, K. Williams, J. Kennedy, and J. Lopes, “Scattering from objects at a water-sediment interface: Experiment, high-speed and high-fidelity models, and physical insight,” *Oceanic Engineering, IEEE Journal of*, vol. 40, no. 3, pp. 632–642, July 2015.
- [14] S. Kargl and K. Williams, “Full scale measurement and modeling of the acoustic response of proud and buried munitions at frequencies from 1-30khz,” *Final Report, SERDP Project MR-1665*, pp. 18–19, May 2012.
- [15] H. Daniels, “Saddlepoint approximations in statistics,” *The Annals of Mathematical Statistics*, vol. 25, no. 4, pp. 631–650, Dec. 1954.
- [16] R. Butler, *Saddlepoint Approximations with Applications*. Cambridge University Press, New York, 2007.
- [17] M. G. Kendall and A. Stuart, *The Advanced Theory of Statistics, Volume I*, 3rd ed. Griffin, London, 1969.

- [18] C. Rose and M. D. Smith, *Mathematical Statistics with Mathematica*. Springer-Verlag, New York, 2002.
- [19] D. Cochran, H. Gish, and D. Sinno, “A geometric approach to multiple-channel signal detection,” *IEEE Transactions on Signal Processing*, vol. 43, no. 9, pp. 2049–2057, September 1995.
- [20] H. A. David and H. N. Nagaraja, *Order Statistics*, 3rd ed. Wiley Series in Probability and Statistics, 2003.
- [21] I. Santamaria, L. L. Scharf, C. Peterson, M. Kirby, and J. Francos, “An order fitting rule for optimal subspace averaging,” unpublished, 2015.