THESIS

UNDERWATER UXO CLASSIFICATION USING MATCHED SUBSPACE CLASSIFIER WITH SYNTHETIC SPARSE DICTIONARIES

Submitted by
John Joseph Hall
Department of Electrical and Computer Engineering

In partial fulfillment of the requirements
For the Degree of Master of Science
Colorado State University
Fort Collins, Colorado
Summer 2016

Master's Committee:

Advisor: Mahmood R. Azimi-Sadjadi

Jade (Yu) Morton
Michael Kirby

ABSTRACT

UNDERWATER UXO CLASSIFICATION USING MATCHED SUBSPACE CLASSIFIER WITH SYNTHETIC SPARSE DICTIONARIES

Classification of underwater objects such as unexploded ordnances (UXO) and mines from sonar datasets poses a difficult problem. Among factors that complicate classification of these objects are: variations in the operating and environmental conditions, presence of spatially varying clutter, variations in target shape, composition, orientation and burial conditions. Furthermore, collection of large quantities of *real* and representative data for training and testing in various background conditions is very difficult and impractical in many cases. In order to remedy the lack of data availability, physical models of varying computational complexity are often used to supplement training databases with synthetically created samples which predict the response of known target models.

In this thesis, we try to address several key questions for designing robust classifiers for UXO and munitions classification from low frequency sonar. These include: (1) "How can we form discriminative and highly separable features for describing UXO and non-UXO objects in a given dataset?", (2) "When do we reach a point of diminishing returns when utilizing synthetic models in a classifier's training?", and more importantly (3) "Which types of object variations cannot be modeled well by synthetic data?". Although, it may be somewhat ambitious to expect model data to capture *all* the essential features of proud or buried underwater objects for target characterization, these models can nevertheless provide us with clues on how to augment the training datasets to improve the robustness in different environmental conditions.

Using empirically validated scattering models developed by University of Washington's Applied Physics Laboratory (APL-UW), fast ray models were acquired to generate the required synthetic training dataset for various UXO and non-UXO objects. A comprehensive analysis is then carried out on the classification performance of two subspace matching classifiers, trained on the synthetic data generated from this physical model, and tested on three real underwater sonar datasets. Both single and multi-aspect classification were considered using a combination of linear subspace models. Our classification hypothesis is that the spectral content of sonar backscatter display unique signatures providing good discrimination between different classes of objects. To develop a robust target classification method that can be applied to discriminate munitions from non-hostile man-made objects and competing natural clutter, the Matched Subspace Classifier (MSC) framework was adopted in conjunction with multidimensional Acoustic Color (AC) feature data extracted from raw sonar returns.

Classification results of the MSC system constructed using two different signal subspace learning methods, namely K-SVD and locality preserving (LP) K-SVD are presented and benchmarked against each other. Additionally, a non-linear version of MSC using the kernel trick was developed and tested on the same datasets. The first two sonar datasets, PondEX09 and PondEX10, were collected for various underwater UXO and non-UXO objects using a rail system in a pond, under relatively controlled and clutter-free conditions. The third dataset, TREX13, was also collected using a similar rail system but in the bay area off of the Panama City coast where other factors such as schools of fish, water turbulence, seafloor roughness, and target range variations were more realistic. Classification results are presented using several standard performance metrics such as Receiver Operating Characteristic (ROC) curves, and Confusion Matrices. These results indicate promising performance for the MSC classifier using LP-KSVD dictionary learning method with the LP K-SVD trained system yielding correct classification rates of $P_{CC} = 93.1\%,$ and $P_{CC} = 82\%$ for UXO, vs. non-UXO classification on the PondEX and TREX datasets, respectively. The Kernel MSC when compared

with the linear MSC performed comparably for single range testing but degraded in performance when attempting to generalize across multi-range datasets demonstrating kneepoint probability of classification $P_{CC} = 88.0\%,$ and $P_{CC} = 76.7\%$ for PondEX and TREX all range datasets, respectively.

ACKNOWLEDGEMENTS

I would first like to thank my advisor, Dr. Mahmood Azimi-Sadjadi, without his support and guidance I may never have come to fully appreciate the product of thorough research or the importance of statistical signal processing. His time and effort are greatly appreciated. I would also like to thank my committee members, Dr. Jade Morton and Dr. Michael Kirby, for their time and assistance during my course of research.

I would like to thank the Strategic Environmental Research and Development Program (SERDP) for providing the funding used in this project under contract number W912HQ-14-C-0007. Thanks to my collaborators on this project, namely the researchers at APL-UW, Dr. Steve Kargl, and Dr. Kevin Williams. I would also like to thank Nick at ISTI, his guidance and expertise made this study possible.

I would like to thank my colleagues in the Signal and Image Processing Lab. They have provided a great environment for discussing work and providing help when most needed. Thanks to Jarrod, Vladimir, Puoya, and Pooria. Special thanks to Vladimir for his much needed advice and insight.

Finally, I would like to thank my family for their support and guidance throughout my entire education.

DEDICATION

To my parents, Renée and Scott, and my four brilliant siblings, Katherine, Jessica, Edward and Kenneth.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement and Motivations

The United States' Department of Defense (DoD) is currently responsible for clearing many sites which are potentially contaminated with munitions as a result of past training and weapons testing activities. In many cases, these activities occurred near or were performed in shallow water environments where munitions pose serious threats to public safety and the environment. In order to effectively search and retrieve munitions, automatic target recognition (ATR) systems that are robust across environmental and sonar system variations have long been pursued.

Classification systems employed to search for munitions typically consist of three types of sensory systems: magnetic metal detection, chemical sensing, and sonar. Metal detection and chemical sensors are typically restricted to short range sniffing ATR systems. Sonar technologies can also be limited in range or limited by poor penetration into sediments. Low frequency broadband sonar systems provide an alternative for the detection and classification of munitions in underwater environments. Low frequencies offer greater detection ranges which permits the surveying of wider areas while at the same time attaining greater penetration depths into sediment which permits detection of partially and completely buried munitions. Moreover, the use of transmitted signals with a wider bandwidth provides high range resolution for detailed surveys with greater localization capability as well as better ability to excite the discriminatory structural modes of the proud and/or buried targets for unambiguous classification.

The problem of underwater object classification in sonar imagery has recently attracted a substantial amount of attention [1–6]. This problem is rather complicated due to the numerous factors which inhibit repeatable and reliable ATR. These include: variations in operating and environmental conditions, presence of spatially varying clutter, as well as burial depth and variations in target shapes, compositions and orientation. Moreover, bottom features such as coral reefs, sand formations, and vegetation may totally obscure a target or confuse a classification system. Consequently, a robust ATR system should be able to quantify changes between the returns from the bottom features or structured clutter and any target activity in sonar data, while at the same time extract useful features for classification.

This work is motivated by several interrelated problems, the primary of which being the safe reclamation of dangerous expended munitions and remediation of the natural environment in areas affected by past military exercises. The ultimate goal of this work is to provide the DoD with improved classification and characterization techniques as it strives to find safer and more cost-effective technologies for underwater munitions remediation. Furthermore, this research attempts to enrich and expand the growing bodies of work concerned with the rapid and highly efficient detection and classification of underwater UXOs from sonar data. More specifically, a major motivation for this work is related to the fact that collecting real UXO data in realistic settings is a difficult, if not an impossible, task. Therefore, by using physical models to construct signal subspaces spanned by the acoustic response of a particular target over a wide range of aspect orientations, our hypothesis is that an effective classification system can be designed that remains robust to changes in target aspect in different environmental and operating conditions.

In this work, we plan to investigate the use of sparse dictionary learning for designing classification systems for underwater munitions through the subspace matching framework. Using empirically validated models of the frequency-dependent acoustic response from munitions and other objects with known geometrical and physical characteristics, signal subspaces will be constructed for each object type over a wide range of aspect orientations [7]. Our classification hypothesis is that these signal subspaces or "acoustic color" templates can effectively serve as a "fingerprint" to uniquely identify that object class. A matched subspace classifier with signal-dependent dictionaries will then be developed and used to decide whether or not the object in question is a munition. By keying in on specific target responses, the classifier will correspondingly exhibit low false alarm rates. Performance of the proposed algorithms will be demonstrated by applying them to synthesized as well as actual sonar data collected in realistic environments.

The classification algorithms developed in this thesis will allow for near real-time assessment of large underwater areas using data collected from low frequency, wideband sonar systems. These algorithms will not only give the user the ability to assess the degree to which a site is contaminated but will also exhibit sufficient discrimination capabilities to unambiguously characterize individual objects with high precision. The developed methods could be useful in a multitude of remote sensing applications in which sonar is used to search or survey underwater areas including environmental and oceanographic studies, undersea exploration, the search for wreckage on the sea floor, and mine-hunting.

1.2 Literature Review on Underwater Target Classification

Various methods have been developed for modeling the acoustic response of objects with geometries typically observed in mine and UXO-hunting applications and using this information for the purposes of classification. In [8], the authors considered Synthetic Aperture Sonar (SAS) imaging of simple targets by exploiting models for reverberation, acoustic penetration, and target scattering into a unified model. This is then used to generate pings suitable for SAS simulations over a wide range of environmental and experimental conditions. Experimentally measured target scattering from proud and buried targets are then used to validate the model through several simulations. In [9], the authors analyzed experimental results from a SAS data set collected in a fresh water pond. These measurements were conducted to investigate discrimination capabilities based on the acoustic response of targets for underwater UXO applications. Results from this study showed that it is possible to use the acoustic response template as a fingerprint to uniquely identify a given target.

The work in [10], showed that acoustic-color (AC) features are useful for discriminating similarly shaped targets and that through using Finite Element Methods (FEM), accurate simulations of some UXO targets can be used to test the performance of new sonar designs operating in realistic conditions. Despite the improvements in generating high fidelity simulations of some target models, it was found that reliable detection and classification of problematic UXO (e.g., those that are small and difficult to image) require additional processing tools beyond sonar imaging such as exploring other methods to extract discriminatory information from target responses. To deal with these cases, methods for predicting sonar configurations that extract more information from target responses and yield higher target SNR were considered. The concepts explored made increasing use of projections of target sonar responses onto non-imaging spaces that do not require high spatial resolution to determine the target type. In [4], the primary efforts were focused towards building a real UXO database providing logistical and technical support for the controlled sonar measurement experiments in Target Reverberation Experiment (TREX13), and BayEX14. Additionally, efforts to augment existing databases using FEM and T-matrix modeling were also carried out. Both datasets were processed to check sonar simulations against more realistic data for UXO applications and to further the evaluation of backscattering phenomena for extraction of classification features.

Considerable research have been devoted to the development of different detection and classification methodologies to identify underwater objects from sonar data. In [1] and [2], the authors developed new coherent-based feature extraction methods for SAS-like acoustic color-based detection and classification of underwater mines and UXO objects using the Canonical Coordinate Analysis (CCA) framework [11]. They demonstrated that the canonical correlation features are robust to changes in the bottom condition. Moreover, it was shown that under fixed operating conditions, canonical correlation features are relatively invariant to changes in aspect angle. New feature and decision-level fusion algorithms were also developed in [12] and [3] using a hidden Markov model (HMM) and a Collaborative Multi-Aspect Classifier (CMAC) to improve classification of targets while reducing the false alarms when multi-pings/aspects sonar data are available. The proposed CMAC system provided overall hard-limiting correct classification rates of 95% on a SAX '04 dataset, collected using the disk Buried Object Scanning Sonar (BOSS) system off the coast of Fort Walton Beach, FL in September of 2004 [13]. In [5], the authors developed a novel classification scheme based on wavelet packet [14] features extracted from backscattered sonar . The wavelet packets were utilized in conjunction with linear predictive coding (LPC), a feature selection scheme, and a backpropagation neural-network classifier [15]. The overall system offered excellent classification performance for discriminating mine-like vs. non mine-like for both single-aspect and multi-aspect cases when decision-level fusion was implemented using a separate neural network.

In [16], a new adaptive underwater target classification system that copes with environmental changes in acoustic backscattered data from targets and non-targets was developed. The core of the system had an adaptive feature mapping that minimizes the classification error rate of the classifier. The goal in this method was to map the feature vector in such a way that the mapped version remains invariant to the environmental changes. A K-nearest neighbors (KNN) system was used as a memory to provide the closest matches of an unknown pattern in the feature space. The classification decision was made by a backpropagation neural network (BPNN) trained on wavelet features of various object classes. In this approach, the authors defined two different cost functions for adaptation. A memory system was used to identify the closest matches of an unknown pattern in the feature space and provide their corresponding decisions (labels). Along with this memory system were feature mapping and classification subsystems. In this approach, based upon the collective evidence provided by the memory, the decision about the direction and magnitude of the feature mapping was made. Through this technique, a confidence measure based upon the probability of error was introduced that provided a means to determine when the feature mapping is needed, and how it should be directed. Balancing construction error with classification error, two cost functions were then combined together to improve the overall classification performance. Performance was seen to approach classification error rates as low as 7% on real sonar datasets.

1.3 Research Objectives

The main objective of this work is the development of efficient methods for the classification of military munitions in shallow underwater environments using data collected from SAS systems. Specifically, the technical question that is addressed in this work is whether or not the Matched Subspace Classifier (MSC) [17] can successfully be trained on model-based sonar data of various UXO-like and non-UXO like objects and then be applied to real sonar datasets to discriminate munitions from other natural or man-made competing objects with sufficient accuracy . The development of systems that can be trained on model-based data with guaranteed performance on real data could provide a significant contribution toward solving this difficult problem.

The comprehensive testing of the MSC and its application to the classification of munitions using low frequency sonar is another primary objective of this work. To this end, we test the hypothesis that the spectral features captured in the AC data, extracted from the sonar backscattered from various objects, display unique features providing excellent discrimination between different classes of detected objects. In this thesis we present new classification results of the MSC when trained via K-SVD [18] and LP-KSVD [19] dictionary learning methods on AC data that has been generated via a fast ray model (FRM). Additionally, to determine the effect of using a non-linear version of the MSC, a kernel MSC [20] was also trained and tested on the sonar datasets. These classifier systems are applied to the entire PondEX and TREX13 datasets to test the generalization ability of the classifiers trained exclusively on FRM-generated datasets.

The work presented here will develop a thorough analysis of the usage of linear and non-linear subspace matching for the problem of underwater UXO classification and some of the benefits and pitfalls of current dictionary learning methods when applied to this difficult problem. Test results on three different real low frequency sonar datasets are presented. The first dataset contains all objects from the PondEX09 exercise and the corresponding FRM generated data for this controlled low frequency (1-30 kHz) sonar experiment. The second dataset, from the PondEX10 features many of the same objects as the former and was collected in a similar fixed range and fixed motion controlled experiment. The third dataset, TREX13, features many objects from the previous two experiments, and many more, but at varying target ranges and with far more realistic environmental conditions. Classification results and analysis for each of the three experiments are provided in terms of performance metrics such as ROC curve and confusion matrices.

1.4 Organization of the Thesis

This thesis is organized as follows: In Chapter 2 we begin with an overview of the FRM, a physical model which characterizes an objects scattering behavior as well as the water sediment interactions of an underwater object. An overview of the Acoustic Color (AC) feature generation process is also given for the synthetic and real datasets. In this chapter, a list of the AC samples from objects used in testing and training is also given. Chapter 3 discusses two different methods of the sparsely constrained subspace representation methods known as K-SVD [18] and LP-KSVD [19] for subspace construction needed in MSC classifiers. Chapter 4 gives a review of the linear MSC method in [17] and its invariance properties. The results of this classifier using the two dictionary learning methods are also presented on all the testing datasets. Chapter 5 reviews the kernel version of the MSC [20] and benchmarks its classification performance against the linear MSC on all the listed datasets. Finally, in Chapter 6 we give concluding remarks on this work and suggestions for future developments.

CHAPTER 2

FAST RAY MODEL AND SONAR DATASETS

2.1 Introduction

For a classifier system to perform reliably, the system must be trained on samples that are representative of different object classes. In many practical situations, the acquisition of a complete and exhaustive, labeled, training sample library is nearly impossible to achieve. Furthermore, training biases that are introduced from specific properties of a sonar hydrophone element or the environment from which training samples were collected can often lead to poor generalization capabilities in sonar classification systems. As an alternative, several physical models seek to emulate the interactions of a solid object reflecting sound pressure underwater in order to predict what target responses we might expect in conditions we have yet to encounter. These models can be utilized in order to provide some indications of what response we can expect to receive from objects of a known class, with certain shape and composition, in different bottom conditions.

In this work, we utilize an empirically validated acoustic scattering model which can quickly and reliably generate synthetic, yet realistic, sonar datasets with desired environmental conditions and sonar path geometry. This chapter begins by reviewing the fast ray model (FRM) [7] utilized in this work in Section 2.2. The four-path model and its underlying parameters are also given. Section 2.2.1 discusses how the FRM data are used to generate Acoustic Color (AC) features for classification. An overview of the process used for creating AC plots for the real sonar datasets is also given. Section 2.3 provides descriptions of the real sonar datasets used for testing of the trained MSC classifier, namely the TREX13 and PondEx09-10 datasets, and their collection experiments. Section 2.4 gives the concluding remarks and the benefits and drawbacks of the FRM for data generation.

2.2 Fast Ray Modeling of the Acoustic Response from Munitions

In order to construct template signals needed to reliably represent the various UXOs in our classification system, the work of [7, 21–24] on fast ray modeling of scattering from objects at a water-sediment interface has been utilized. The scattering model allows for monostatic SAS data sets to be simulated via a fast ray model that combines an acoustic ray approximation for propagation in a fluid-filled halfspace with scattering from a target in a number of conditions and media. This fast ray modeling is beneficial for generating large data sets for dictionary construction. In this section, we will discuss this scattering model. Under typical operation for a short-range SAS platform, air-water scattering paths can be ignored, because paths that interact with the air-water interface are either removed by time-gating the received signals or are naturally suppressed by the directivity of the source and receiver. In addition, the separation distance between the actual source and receiver is much smaller than the distance between the interface and the target, so the source and receiver can be considered to be co-located. Under these conditions, only the four ray paths shown in Figure 2.1 contribute to the scattered pressure. The actual source, receiver, and target are denoted by S, R, and T, while $S1$ and $R1$ are image source and receivers. In this figure, the path 1 is a direct reflection path. The paths 2 and 3 interact with the sediment once and scatter from the target in a bistatic direction, and path 4 is a back-scattering path with two bottom interactions. In this model, the source, receiver, and target are located at $\mathbf{r}_s$, $\mathbf{r}_r$, and $\mathbf{r}_t$, and an image source is located at $\mathbf{r}_{s_i}$ with an image receiver at $\mathbf{r}_{r_i}$. To

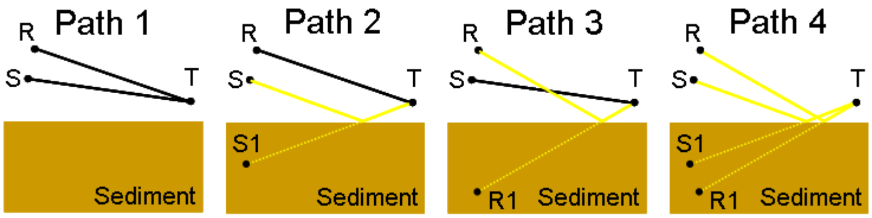


Figure 2.1: Four Ray Path Model.

distinguish path 2 and path 3, the source and receiver are shown at distinct locations; and with our assumption of co-located source and receiver, paths 2 and 3 are reciprocal and paths 1 and 4 are backscattered. With the specification of a source and receiver, the scattering from a target has been reduced to a superposition of 4 free-field scattering problems. Under operational conditions, the distance associated with each path satisfies $d \gg \lambda$ where λ is the wavelength of the pressure. The scattered steady-state pressure can then be written as

$$p_s = p_0 A\left(\mathbf{k}_s, \mathbf{k}_i, \omega\right) \frac{\exp(ikr)}{r} \tag{2.1}$$

where p_0 is the amplitude of the incident pressure, r is the range from a field point to the target, A is the scattering amplitude, $\exp(ikr)/r$ is a spherically diverging wave, $k = 2\pi/\lambda$ is the wavenumber, ω is the angular frequency, and $\mathbf{k}_i$ and $\mathbf{k}_s$ are the unit vectors associated with the direction of the incident and scattered fields, respectively. The scattering amplitude contains useful information concerning the material properties of the target and the directionality of the scattered field. The scattering amplitudes can be determined from analytic solutions to scattering problems (e.g., scattering from a spherical target), direct measurements from actual targets, or numerical simulations (e.g., a finite element (FE) model for a given target).

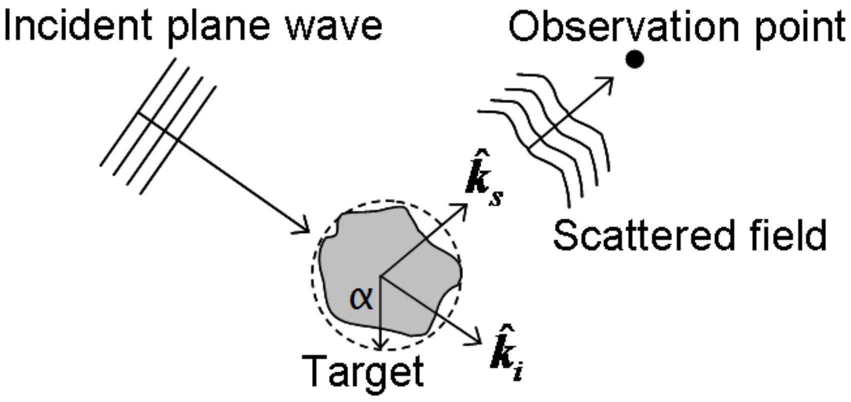


Figure 2.2: Free-field scattering assumes a portion of an incident wave is scattered to a distant target.

Combining the ray model paradigm with free-field scattering as given in (2.1), the spectrum of the total scattered pressure can be written as

$$P(\omega) = \left(\frac{A_1(\omega)}{d_1 d_2} e^{i\omega t_1} + \frac{V(\theta_g) A_2(\omega)}{d_2 d_3} e^{i\omega t_2} + \right. \\ \left. \frac{V(\theta_g) A_3(\omega)}{d_1 d_4} e^{i\omega t_3} + \frac{V^2(\theta_g) A_4(\omega)}{d_3 d_4} e^{i\omega t_4} \right) r_0 P_{src}(\omega) \tag{2.2}$$

with $d_1 = |\mathbf{r}_s - \mathbf{r}_t|$, $d_2 = |\mathbf{r}_t - \mathbf{r}_r|$, $d_3 = |\mathbf{r}_{s_i} - \mathbf{r}_t|$, and $d_4 = |\mathbf{r}_t - \mathbf{r}_{r_i}|$. The time delays are then $t_1 = (d_1 + d_2)/c_1$, $t_2 = (d_2 + d_3)/c_1$, $t_3 = (d_1 + d_4)/c_1$, and $t_4 = (d_3 + d_4)/c_1$; with c_1 being the speed of sound in water. The pressure spectrum $P_{src}(\omega)$ represents the frequency spectrum of the transmitted wave packet from the source, and $r_0 = 1$ m is a reference distance. The scattering amplitudes $A_k(\omega)$ in (2.2) also depend on the locations of the sources, receivers, and target. Note the indices of A_k correspond to the path enumeration depicted in Figure 2.1. The reflection coefficient at the water-sediment interface, $V(\theta_g)$, is a function of the grazing angle θ_g, and is defined as follows

$$V(\theta_g) = \frac{\rho \sin(\theta_g) - (\kappa^2 - \cos^2(\theta_g))^{1/2}}{\rho \sin(\theta_g) + (\kappa^2 - \cos^2(\theta_g))^{1/2}} \tag{2.3}$$

where $\rho = \rho_2/\rho_1$ and $\kappa = (1 + j\delta)/\nu$ with $\nu = c_2/c_1$. Here ρ_1 is the density of water. The density, sound speed, and loss parameter for the sediment are ρ_2 ,c_2 and δ. An inverse Fourier transform of $P(\omega)$ thus gives a generated sonar signal that includes the four primary acoustic propagations for a target near an interface. The four paths represented in this model take into account the interaction of the sonar with the target and the environment (seafloor). If one were to throw out the three paths that reflect from the seafloor and only keep the first path, then one would have "free-field scattering" from the target only. The term free-field scattering implies that the target is so far from any boundary that it can be thought of as suspended in an infinite volume of water.

2.2.1 Acoustic Color for Synthesized Data

In order to produce AC features for the synthetically generated data to train the MSC classifier, raw sonar returns generated by the fast ray model must first be pre-processed.

Generation of AC features amounts to forming the intensity magnitude of the returned spectral power over the entire range of aspect angles that are modeled in either linear path SAS (LSAS) or circular path SAS (CSAS) runs. This is accomplished by the following procedure: First, a FE model [24] is implemented to produce scattering amplitude information for the intended target. These scattering amplitudes are modeled for acoustic transmissions and returns in the low frequency range of 1-30 kHz. Next, the half-space model including the four described ray paths in (2.2) is utilized to generate a raw sonar return data set by generating the modeled returns of a target using the inverse FFT of (2.2) over a pre-generated coordinate set representing the various positions along a circular or linear path making soundings. As will be explained later in more details, in this simulation model, a copy of any transmitted pulse may be used. For experimentation, a copy of the transmitted pulse used during data collection was used, which provides spectral information of an object's backscatter in the desired frequency range. Next, these raw soundings were pulse-compressed with the original transmit signal. Finally, the magnitude value of the FFT of the pulse-compressed data is computed and the result is windowed to 0-30 kHz to remove the unused frequency portions and isolate the frequency range of interest. As an arbitrary trajectory of a SAS platform can be modeled with the FRM [7], the model was used to generate synthetic LSAS sonar data with a target located at the center of the LSAS path for a set of ranges and objects collected in all real datasets used in this research. Model parameters such as sonar interface elevation and water conditions were set to match those measured during data collection experiments. This was repeated for all objects present in the tested datasets. Lists of objects used in training and their characteristics are given in Table 2.1. An example of these AC features generated for a 21-m long simulated CSAS run at a range of 10 m and with an interface elevation of 3.8 m is given in Figure 2.3 (c) for an aluminum UXO object.

AC features generated for template signals via the described scattering model were then utilized as training data for our various MSC classifiers. AC feature vectors were generated for all proud objects and ranges that were encountered in the actual test datasets. The

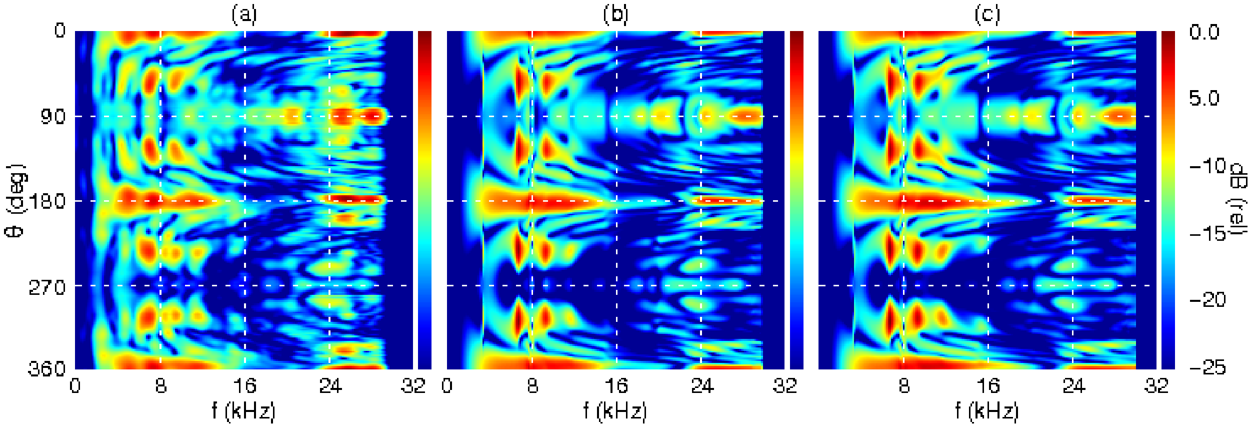


Figure 2.3: AC data of a bullet-shaped aluminum UXO replica at 10 m range generated via (a) Data collected during PondEX10 (SERDP MR-1665). (b) FEM and Kirchoff-Helmholtz integral. (c) FRM with scattering form function derived from the scattered pressure computed in (b).

major benefit of utilizing the ray model developed in [7,24] is that, after free-field scattering amplitudes for a desired object are collected or modeled via FE methods, the regeneration of the ray model simulating various aspects and orientations is far simpler and faster than re-running these variations with the FE method [24].

The validated performance of the FRM can be visualized in Figure 2.3. Figure 2.3(b) displays the AC template predicted for an Aluminum UXO via the FE method. Figure 2.3(a) displays the real AC data of a bullet shaped aluminum UXO collected from the TREX13 exercise, while Figure 2.3(c) displays the result of the FRM for the same objects response. Comparing AC plots in Figures 2.3(a)-(c), one can clearly observe that the AC plot in Figure 2.3(c), generated via the FRM [7], is not only preserving all the spectral information of that generated using the slow FE model in Figure 2.3(b), but also capturing the essential AC features of the real sonar data of Figure 2.3(a) for the same object collected in PondEX experiments. A great deal of spectral information for object discernment seems to be present in these AC plots. We shall present in Section 2.3.3 present the AC data generation for real sonar datasets.

Table 2.1: Fast Ray Model Combined (PondEX and TREX) Training Data Set.

No.	Class	Object Description	Ranges
1	non-UXO	2 ft Aluminum Cylinder	10 m
2	non-UXO	3 ft Aluminum Cylinder	$(10), 30, 35, 40$ m
3	non-UXO	2 ft Aluminum Pipe Section	$(10), 15, 25, 30$ m
4	UXO	100 mm Aluminum Rocket Round	$10, 15, 30$ m
5	UXO	100 mm Solid Steel Rocket Round	$10, 15, 25, 30$ m
6	UXO	105 mm Bullet (Air fill)	$15, 20, 25$ m
7	UXO	105 mm Bullet (H_2O fill)	$15, 20, 35$ m
8	UXO	155 mm Howitzer w/ Cap (Air fill)	$10^*, 15, 20, 35$ m
9	UXO	155 mm Howitzer w/ Cap (H_2O fill)	$15, 25, 30$ m
10	UXO	155 mm Howitzer no Cap	$25, 30, 40$ m

2.2.2 Synthesized Training Data Set

As stated before, the specific objective of this work is to determine if model-generated AC features, for a wide variety of UXO and non-UXO objects with known geometrical and physical characteristics at different sonar range and orientation, can be used to successfully construct dictionary matrices for the MSC-based classification system which provide good performance on real datasets. Similar to the real datasets described in Sections 2.3.1 and 2.3.2, object orientations were created that ranged from $-80°$ to $+80°$ in $20°$ increments, with $90°$ orientation corresponding to broadside view of the object and $0°$ corresponding to an end-on view of objects. For each object in the real sonar data collection experiments, the AC features generated were decimated along the frequency dimension to have $N = 301$ frequency bins spanning the 0-30 kHz frequency range corresponding to approximately 100 Hz separation of frequency bins and along the aspect dimension to have $0.5°$ aspect resolution (i.e. 721 aspect vectors $\propto 360°$). This database of AC features was created to match aspect resolution and frequency resolution of those generated for the real sonar data. Table 2.1 gives the list of all the objects and their range information for which synthesized data were generated to match those in PondEX and TREX experiments. This synthesized training set is given with range information in order to provide a reference for the FRM types used for the PondEX and three TREX experiments performed with the three developed classifiers.

2.3 Sonar Data Descriptions

In this section we overview the sonar datasets and the UXO and non-UXO objects that were used for testing the developed classifiers. The object characteristics and their ranges from the sonar platform are also provided.

2.3.1 PondEX Data Description

Figure 2.4 shows the layout of the PondEX10 experiment which also matches the setup that was used in the earlier PondEX09 experiment including the relative locations of the rail-mounted sonar system and the objects in the target field. The 21 m rail the sonar system is mounted on is fixed to eliminate platform motion as the sonar interface moves along its track. The sonar transmit signal is a 6 msec linear frequency modulated (LFM) pulse over 0.5-30 kHz with a 10% taper between the leading and trailing edges to minimize ringing in the transmitted signals. Sonar backscatter is received with $L = 6$ hydrophone elements that are arranged in a linear array approximately normal to the seafloor. For the formation of AC data, only the 3rd hydrophone element data was used. As can be seen from

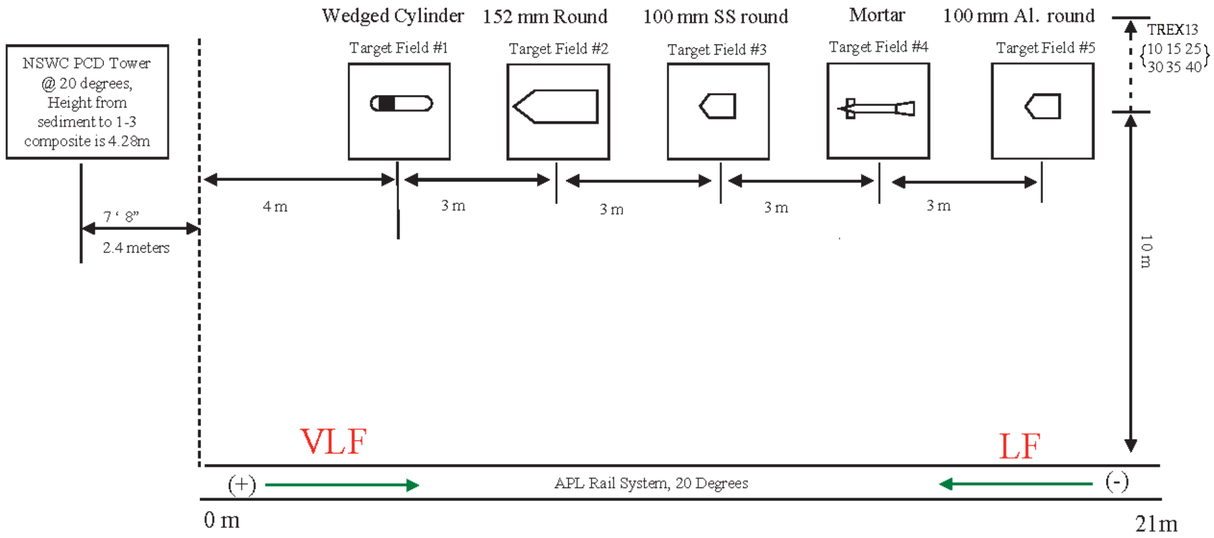


Figure 2.4: Layout of the Field for PondEx10 Data Collection.

Figure 2.4, for the PondEX experiments, the target field contained several objects at a time with varying shapes, sizes, and compositions, all of which were located approximately 10 m horizontally from the rail system and are proud on the sandy bottom. Tables 2.2 and 2.3 enumerate the tested objects for each of the experiments and briefly describes each. Since the only non-UXO types in PondEX10 were clutter/rock objects, this was considered to be the easier testing set. For the PondEX09 experiment, objects that we considered (those objects for which FRM scattering data exists) included hostile and non-hostile man-made objects for both non-UXO and UXO classes. This data set was expected to provide more of a challenge because there is the potential for the backscattered frequency structure of a man-made non-UXO object such as an aluminum pipe to be similar to that of aluminum UXO due to the regularity and similarity of the shapes and materials.

The sonar data sets used in this study were collected during ten runs through the target field with the target field for each run containing the listed objects. Each run differs in the orientation of all the objects, with each object having the same orientation for a given run. Ten total object orientations were used, ranging from -80° to $+80^\circ$ in 20° increments, with 90° orientation corresponding to broadside view of the object and 0° corresponding to an end-on view of objects. Each run consists of 769 pings in which the sonar platform moved along the fixed rail in increments of 0.025 m, transmitting and receiving once for each sonar position. The data was sampled at 1 MHz and the sonar platform was tilted at a fixed 20° grazing angle for all runs. Since the useful spectral information in the collected data has a Nyquist frequency well below the sampled rate, the 1 MHz data was down-sampled in time by a factor of 10 resulting in data sampled at 100 kHz. Then after AC data was generated, the frequency dimension has $N = 301$ bins spanning the 0-30 kHz frequency range corresponding to approximately 100 Hz in each frequency bin.

As the MSC classifier is to be implemented on AC data, testing was performed on AC's generated from filtered data from these experiments [23]. Using the method described in Section 2.3.3, AC Template testing data was formed by composing the useful aspects from multiple linear SAS runs with the field-centered target taking various rotational poses.

Table 2.2: Objects in the PondEX09 Experiment Testing Data.

Object No.	Class	Object Description	# Aspects
1	UXO	Aluminum UXO	721
2	UXO	Stainless Steel UXO	721
3	Non-UXO	3ft Alumnimum Cylinder	721
4	Non-UXO	Aluminum Pipe	721
5	Non-UXO	Rocks 1 & 2	722
Total # Tested:			3606

Table 2.3: Objects in the PondEX10 Experiment Testing Data.

Object No.	Class	Object Description	# Aspects
1	UXO	Small Aluminum Cylinder with Notch	721
2	UXO	De-militarized 152 mm TP-T Round	721
3	UXO	100 mm Solid Steel Rocket Round	721
4	UXO	Inert 81 mm Mortar	721
5	UXO	100 mm Aluminum Rocket Round	721
6	non-UXO	Rock 1	361
7	non-UXO	Rock 2	361
Total # Tested:			4327

2.3.2 TREX13 Data Descriptions

The experimental setup for TREX13 was similar to PondEx10 [25] described in the previous section. For TREX13, the target field contained several objects with varying shapes, sizes, and compositions, all of which were located between 10 to 40 m horizontally from the rail system and are proud on the sandy bottom. Table 2.4 gives the list and some properties of these objects, together with range information for each object. The rail the sonar system was mounted on was fixed to minimize platform motion as the sonar tower traversed along its track. The length of the rail during TREX13 was approximately 40 m. The sonar transmit signal was a 6 msec LFM pulse over 3-30 kHz with a 10% taper between the leading and trailing edges to minimize ringing effects in the transmitted signals. Sonar backscatter was received by a 6-element linear array that was approximately normal to the seafloor. For the formation of AC features, data from the 3rd hydrophone element data was used as before.

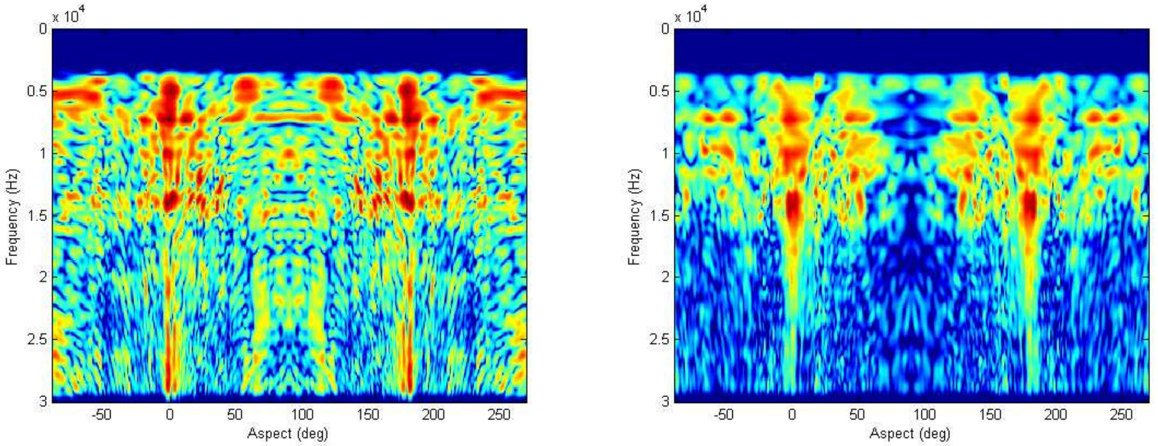
Table 2.4: TREX13 available proud targets used for testing

No.	Class	**TREX #**	**FRM Name**	Ranges
1	non-UXO	Target 17	alcyl2ft	10 m
2	non-UXO	Target 7	alcyl3ft	$30, 35, 40$ m
3	non-UXO	Target 16	alpipe	$15, 25, 30$ m
4	UXO	Target 20	aluxo	$10, 15, 30$ m
5	UXO	Target 21	ssuxo	$10, 15, 25, 30$ m
6	UXO	Target 25	bullet_105mm_air	$15, 20, 25$ m
7	UXO	Target 29	bullet_105mm_h2o	$15, 20, 35$ m
8	UXO	Target 9	howitzer_cap_air	$10, 15, 20, 35$ m
9	UXO	Target 28	howitzer_cap_h2o	$15, 25, 30$ m
10	UXO	Target 8	howitzer_nocap	$25, 30, 40$ m

The sonar data sets used in this study were collected during ten runs through the target field. Each run differed in the orientation of all the objects, with each object having the same orientation for a given run. Similar to the PondEX experiments, the object orientations varied from $-80°$ to $+80°$ in $20°$ increments. However, each run of data consisted of approximately 1600 pings in which the sonar platform moved along the fixed rail in increments of 0.025 m, transmitting and receiving once for each sonar position. The data was sampled at 100 kHz and the sonar platform was tilted at either a $10°$ or $20°$ grazing angle depending on range to the targets being observed.

2.3.3 AC for Real Sonar Data

The procedure for generating AC features for real raw sonar data experiments is similar to the procedure performed on the synthesized sonar data described in Section 2.2.1. Raw sonar data time series collected from the PondEX or TREX13 experiments are first pulse compressed with the transmit signal that was used in the data collection experiment and the result is further processed to remove returns from the neighboring objects to isolate the object of interest. This processing utilizes a reversible SAS imaging process, a spatial filtering process using a 2-D Tukey window, and a pseudo-inverse filtering [26]. This inverse filter maps the SAS image back to the pulse-compressed version that has less interface scattering



(a) Air-Filled Howitzer (object 8)

(b) Aluminum UXO Shell (object 4)

Figure 2.5: AC Features for two UXO objects in TREX13 dataset

noise. These filtered pulse-compressed signals are then transformed to the frequency domain via FFT and the magnitude of the FFT was utilized as the testing feature vectors. This process is repeated for all aspects of a given LSAS run and the amplitude spectrum is then generated and plotted to display AC for each object. Examples of these AC plots for two real UXO objects in the TREX13 test set, the Air-Filled Howitzer with a Cap and the Aluminum UXO, are shown in Figures 2.5(a) and 2.5(b). From these two figures, it is easy to verify that the frequency response of the air filled howitzer and aluminum UXO object types are distinctly different despite similarities in shape and object fill. Furthermore, when comparing to the Aluminum UXO plot of Figure 2.3 (a), some discrepancies and interference appears to be present in the TREX observation given in Figure 2.5(b) that was not present in the more controlled PondEX experiments.

2.4 Conclusion

In this chapter we first introduced the FRM [7] as a framework for modeling object scattering in a water sediment interface. Using the ray model, target scattering is reduced to a convolution of a free-field scattering amplitude and an incident acoustic field at the target location. A simulated or measured scattered free-field pressure from a complicated target can be reduced to a complex scattering, and this is then used within the ray model via interpolation. Using this model, we explained how the four-path wave guide model could be used along with pre-saved complex scattering coefficients, found via FEM methods, to rapidly generate a customizable and complete library of synthetic training samples. This section was then followed by an explanation of the AC data features used for sonar classification in this work and the procedure for generating AC features for synthesized and real sonar datasets. Examples of a patch of AC vectors generated from a synthesized LSAS run were also given in this chapter. Finally, overview of the testing and training datasets to be used by the various classifiers were given also. Using the model-generated data in conjunction with the subspace learning methods presented in the following chapter, we attempt to learn a robust signal model for different object classes.

CHAPTER 3

SUBSPACE LEARNING METHODS

3.1 Introduction

An overcomplete dictionary that leads to sparse representations [18, 19, 27] can either be chosen as a pre-specified set of functions or designed by adapting its content to fit a given set of signal examples. In this chapter, we overview two state of the art methods for learning sparse reprsentations for data from a training set. In the K-SVD dictionary learning method [18], the dictionary is designed specifically for a signal type in order to meet an imposed sparsity model. The goal is to find the dictionary that yields signal-dependent sparse representations for a given training set. Such dictionaries have the potential to outperform [28] commonly used predetermined dictionaries for the problem of sonar classification owing to the fact that for different object classes specifically trained dictionaries are utilized to perform classification. The locality preserving (LP) K-SVD dictionary learning method [19], on the other hand, extends to the sparse dictionary learning method by adding an additional constraint which attempts to make landmark points for dictionary atoms. This is done by enforcing atoms to be representative of the original neighborhoods from where they were sampled on the nonlinear ambient data manifold. The K-SVD 's strength lies in its adherence to strict minimization of squared-error reconstruction w.r.t. a specified sparse signal model. This can result in low error in sparse reconstructions but class ambiguity in the atoms learned. The LP K-SVD's strengths lie in its assumption of smoothness in the data domain, balancing representative-ness of atoms with reconstruction capabilities while enforcing a local sparse model.

The outline of this chapter is as follows. In Section 3.2, an overview of the K-SVD signal subspace construction method is given. Section 3.3 overviews the Locality Preserving K-SVD which slightly modifies the objective function in K-SVD to focus learning dictionary elements locally representative on the data manifold of observations in a selected class. Both of these methods of signal subspace representation are utilized in MSC classifier testing and their strengths and weaknesses are discussed in Chapter 4.

3.2 K-SVD Dictionary Learning Review

As will later be shown in Chapter 4, the linear signal model under both ‘UXO’ and ‘non-UXO’ hypotheses for our problem takes the form

$$\mathbf{Y}_m = \mathbf{H}_m \mathbf{X}_m + \mathbf{N}, \quad m = 0, 1 \tag{3.1}$$

where $\mathbf{Y}_m \in \mathbb{R}^{N \times Q}$ is the class m sample observation matrix, $\mathbf{N} \in \mathbb{R}^{N \times Q}$ is the noise or (inaccuracy modelling) matrix, $\mathbf{H}_m \in \mathbb{R}^{N \times K}$ is the subspace matrix, and $\mathbf{X}_m \in \mathbb{R}^{K \times Q}$ is the parameter matrix associated with data matrix $\mathbf{Y}_m$. The parameter matrix $\mathbf{X}_m$ specifies how to sample linear basis vectors in the m^{th} signal subspace represented by $\mathbf{H}_m$ for each of the Q observed samples $\mathbf{y}_q$ in $\mathbf{Y}_m$. Since with this classifier training amounts to generating suitable signal subspaces under both hypotheses, the discovery of discriminative basis vectors for constructing each subspace $\langle \mathbf{H}_m \rangle$ is of utmost importance. In the remainder of this chapter we review two methods which attempt to learn signal-dependent dictionaries $\mathbf{H}_m$ using the K-SVD [18] and LP-KSVD [19] frameworks.

The purpose of K-SVD is to create an optimal signal-dependent dictionary that reduces the dimension of a high dimensional signal vector by representing it as a sparse linear combination of relatively few atoms. More specifically, K-SVD aims to solve a constrained minimization problem to reduce the reconstruction error in a set of training vectors. Let $\mathbf{Y}_m \in \mathbb{R}^{N \times Q}$ be a matrix consisting of class m ($m = 0, 1$) training data vectors with $\mathbf{y}_q$ for $q \in [1 \cdots Q]$ as its columns, $\mathbf{H}_m \in \mathbb{R}^{N \times K}$ be the dictionary matrix to be found, and

$\mathbf{X}_m \in \mathbb{R}^{K\times Q}$ be the sparse representation of $\mathbf{Y}_m$ in terms of dictionary atoms in $\mathbf{H}_m$. Note it is desired that the number of non-zero elements of each $\mathbf{x}_q$ be substantially less than N as the dimension should be reduced in this process. The constrained optimization problem [18] is given by,

$$\min_{\mathbf{H}_m,\mathbf{X}_m} \{\|\mathbf{Y}_m - \mathbf{H}_m\mathbf{X}_m\|_F^2\} \qquad \text{subject to ,} \quad \|\mathbf{x}_q\|_0 \leq \tau, \quad \forall q \tag{3.2}$$

where $||\cdot||_F^2$ is the Frobenius norm of a matrix [29], and $||\cdot||_0$ is the ℓ_0 norm which counts the non-zero elements of a vector. Put simply, this optimization seeks $\mathbf{H}_m$ and $\mathbf{X}_m$ which simultaneously (1) minimizes squared error of reconstructing training samples $\mathbf{Y}_m$, and (2) enforces reconstructions that depend on at most $\tau \ll N$ basis elements.

During the training, the K-SVD algorithm is composed of two-phases. First, a sparse representation phase is applied where for each $\mathbf{y}_q$ the corresponding $\mathbf{x}_q$ is computed based on a given subspace matrix $\mathbf{H}_m$. These sparse codes are found using a pursuit method such as Basis Pursuit (BP) [30] or Orthogonal Matching Pursuit (OMP) [28]. Second, a dictionary update phase where each column $\mathbf{h}_k$ of matrix $\mathbf{H}_m$ is updated one at a time based on minimizing the reconstruction error using the Singular Value Decomposition (SVD) [29] of a restricted error matrix $\mathbf{E}_k^R$, a matrix representing the reconstruction error incurred when leaving out the k^{th} atom $\mathbf{h}_k$. These two coupled phases are repeated until convergence through monotonic MSE reduction [18]. These steps are briefly reviewed in the following subsections.

3.2.1 Sparse Coding Phase Using a Fast OMP Method

In the sparse representation phase, for each $\mathbf{y}_q$ in the training data matrix $\mathbf{Y}_m$, a corresponding $\mathbf{x}_q$ is computed based on a given $\mathbf{H}_m$ using a pursuit method, namely the OMP. An implementation of OMP that does not require matrix inversion for the sparse coding is briefly reviewed here [28].

To begin, let $\mathbf{H}_m = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_K] \in \mathbb{R}^{N\times K}$ be an overcomplete dictionary matrix representing the signal subspace for class m samples. Now, given an training vector $\mathbf{y}_k \in \mathbb{R}^N$, we would like to find its sparse representation $\mathbf{x}_k \in \mathbb{R}^K$, $K \ll N$ iteratively using the smallest possible number of basis vectors $\mathbf{h}_j$'s such that the norm of the reconstruction error is less than a pre-selected tolerance ϵ, i.e. $||\mathbf{e}_k|| \leq \epsilon$, where $\mathbf{e}_k = \mathbf{y}_k - \mathbf{H}_m\mathbf{x}_k$.

For a given observation vector $\mathbf{y}_k$, each iteration of standard OMP involves finding the inner product of the current residual error vector, calculated at iteration $t-1$, with each of the remaining (unselected) dictionary atoms and selecting that with the largest inner product. More specifically, at iteration t if the current residual error vector is $\mathbf{r}_{t-1}$, the next dictionary atom is chosen using,

$$k_t = \text{argmax}_j \, |\mathbf{r}_{t-1}^T \mathbf{h}_j| \tag{3.3}$$

Then, the augmented dictionary matrix (column-wise) and index set respectively become $\mathbf{H}_{m,t} = [\mathbf{H}_{m,t-1}\mathbf{h}_{k_t}]$ and $S_t = S_{t-1} \bigcup k_t$ with initial values $\mathbf{H}_{m_0}$ being a matrix with randomly selected training samples, and $S_0 = \emptyset$. Using this matrix $\hat{\mathbf{x}}_k(t)$ is estimated by the minimization,

$$\hat{\mathbf{x}}_k(t) = \text{argmin}_{\mathbf{x}_k} \, ||\mathbf{y}_k - \mathbf{H}_{m,t}\mathbf{x}_k|| \tag{3.4}$$

which leads to the least squares (LS) solution for $\hat{\mathbf{x}}_k(t)$,

$$\hat{\mathbf{x}}_k(t) = (\mathbf{H}_{m,t}^T \mathbf{H}_{m,t})^{-1} \mathbf{H}_{m,t}^T \mathbf{y}_k = \mathbf{Q}_{\mathbf{H}_{m,t}} \mathbf{y}_k \tag{3.5}$$

where $\mathbf{Q}_{\mathbf{H}_{m,t}} = (\mathbf{H}_{m,t}^T \mathbf{H}_{m,t})^{-1} \mathbf{H}_{m,t}^T$ represents the LS filter based upon the augmented matrix $\mathbf{H}_{m,t}$. The new residual error term then becomes

$$\mathbf{r}_t = \mathbf{y}_k - \mathbf{H}_{m,t}\hat{\mathbf{x}}_k(t) = \mathbf{P}_{\mathbf{H}_{m,t}}^{\perp} \mathbf{y}_k \tag{3.6}$$

where $\mathbf{P}_{\mathbf{H}_{m,t}}^{\perp} = \mathbf{I} - \mathbf{P}_{\mathbf{H}_{m,t}}$and $\mathbf{P}_{\mathbf{H}_{m,t}} = \mathbf{H}_{m,t}(\mathbf{H}_{m,t}^T \mathbf{H}_{m,t})^{-1} \mathbf{H}_{m,t}^T$ is the projection matrix onto subspace $\langle \mathbf{H}_{m,t} \rangle$ spanned by columns of $\mathbf{H}_{m,t}$. The process is repeated until the error tolerance is met.

As can be seen in (3.5) and (3.6), the computational cost of the original OMP algorithm grows very quickly as the number of chosen atoms increases. This is caused by the rapid increase in cost of matrix inversion as dimensionality increases, the relationship is demonstrated

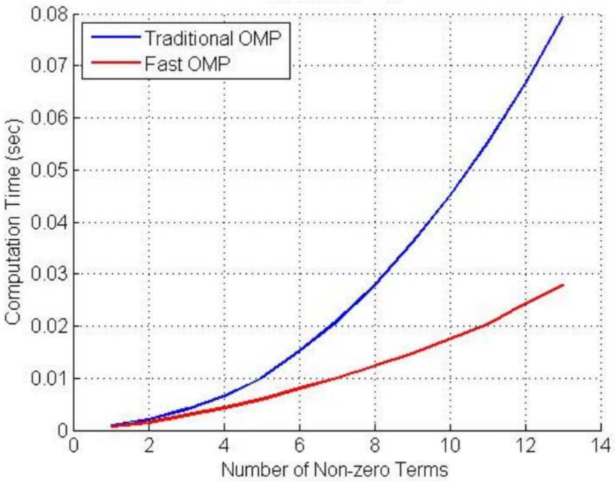


Figure 3.1: Timing Analysis of Pursuit Methods

in Figure 3.1. Additionally, to relate this sparse coding to iterative K-SVD an interface between these two processes is needed. To develop a fast OMP algorithm that avoids matrix inversion operation, the authors in [28] used the orthogonal projection updating [31, 32] for $\mathbf{P}_{\mathbf{H}_{m,t}}$ and $\mathbf{Q}_{\mathbf{H}_{m,t}}$ to get

$$\mathbf{P}^{\perp}_{\mathbf{H}_{m,t}} = \mathbf{P}^{\perp}_{\mathbf{H}_{m,t-1}} - \mathbf{P}_{\tilde{\mathbf{h}}_{k_t}} \tag{3.7}$$

and

$$\mathbf{Q}_{\mathbf{H}_{m,t}} = \begin{bmatrix} \mathbf{Q}_{\mathbf{H}_{m,t-1}} \\ 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{b}_{t-1} \\ 1 \end{bmatrix} \mathbf{q}_t^T \tag{3.8}$$

where $\mathbf{P}_{\tilde{\mathbf{h}}_{k_t}} = \frac{\tilde{\mathbf{h}}_{k_t}\tilde{\mathbf{h}}_{k_t}^T}{||\tilde{\mathbf{h}}_{k_t}||^2}$ is the projection matrix for $\tilde{\mathbf{h}}_{k_t} = \mathbf{P}^{\perp}_{\mathbf{H}_{m,t-1}}\mathbf{h}_{k_t}$ which is the projection of $\mathbf{h}_{k_t}$ onto the orthogonal subspace of $\langle \mathbf{H}_{m,t-1} \rangle$ (or innovation); $\mathbf{b}_{t-1} = \mathbf{Q}_{\mathbf{H}_{m,t-1}}\mathbf{h}_{k_t}$, the filtered version of $\mathbf{h}_{k_t}$ based upon LS filter $\mathbf{Q}_{\mathbf{H}_{m,t-1}}$; and $\mathbf{q}_t^T = \frac{\tilde{\mathbf{h}}_{k_t}^T}{||\tilde{\mathbf{h}}_{k_t}||^2}$ is the LS filter operator using $\tilde{\mathbf{h}}_{k_t}$. Premultiplying (3.7) and (3.8) by $\mathbf{y}_k$ yields recursive update equations for $\mathbf{r}_t$ and $\hat{\mathbf{x}}_k(t)$, respectively, i.e.

$$\mathbf{r}_t = \mathbf{r}_{t-1} - \alpha_t \tilde{\mathbf{h}}_{k_t} \tag{3.9}$$

and

$$\hat{\mathbf{x}}_k(t) = \begin{bmatrix} \hat{\mathbf{x}}_k(t-1) \\ 0 \end{bmatrix} + \alpha_t \begin{bmatrix} -\mathbf{b}_{t-1} \\ 1 \end{bmatrix} \tag{3.10}$$

where $\alpha_t = \mathbf{q}_t^T \mathbf{y}_k = \frac{\mathbf{h}_{k_t}^T \mathbf{r}_{t-1}}{||\mathbf{h}_{k_t}||^2}$ i.e. filtered version of $\mathbf{y}_k$ based upon LS filter operator $\mathbf{q}_t^T$. Thus, the adjustment term in (3.10) corresponding to the previous coefficients is equal to the products of two filtered outputs, namely $\mathbf{b}_{t-1}$ and α_t; whereas the coefficient associated with the newly added atom is α_t. These equations allow for "time-order" updates after adding a new dictionary atom. Note that in this algorithm, computing the projection matrix $\mathbf{P}_{\mathbf{H}_{m,t}}$ and LS filter operator $\mathbf{Q}_{\mathbf{H}_{m,t}}$ are completely avoided and hence no matrix inversion is required. To calculate the new part of $\mathbf{h}_{k_t}$ i.e. $\tilde{\mathbf{h}}_{k_t} = \mathbf{P}^{\perp}_{\mathbf{H}_{m,t-1}} \mathbf{h}_{k_t}$ we only need the filtered output $\mathbf{b}_{t-1}$ since, $\tilde{\mathbf{h}}_{k_t} = \mathbf{h}_{k_t} - \mathbf{H}_{m,t-1}\mathbf{b}_{t-1}$.

3.2.2 Dictionary Update Phase

The dictionary update phase involves writing the cost function in (3.2) in terms of columns of the dictionary matrix, $\mathbf{h}_j$ s and rows of matrix $\mathbf{X}_m$, $\mathbf{x}_T^j$'s. More specifically,

$$\left|\left|\mathbf{Y}_m - \sum_{j=1}^{K} \mathbf{h}_j \mathbf{x}_T^j\right|\right|_F^2 \tag{3.11}$$

Then, separating the effects of the k^{th} dictionary atom ($k \in [1, K]$) from the other terms yields,

$$\left|\left|\left(\mathbf{Y}_m - \sum_{j \neq k} \mathbf{h}_j \mathbf{x}_T^j\right) - \mathbf{h}_k \mathbf{x}_T^k\right|\right|_F^2 = ||\mathbf{E}_k - \mathbf{h}_k \mathbf{x}_T^k||_F^2 \tag{3.12}$$

where $\mathbf{E}_k$ is the error matrix that represents the reconstruction error when neglecting the k^{th} atom. Now, to consider the sparsity of $\mathbf{x}_T^k$, we define a subset of indices $\omega_k = \{i, |x_T^k(i) \neq 0\}$ which correspond to the training samples in $\mathbf{Y}_m$ that use column dictionary atom $\mathbf{h}_k$ for which $x_T^k(i) \neq 0$. Consequently, 3.12 is reduced to

$$||\mathbf{E}_k \Omega_k - \mathbf{h}_k \mathbf{x}_T^k \Omega_k||_F^2 = ||\mathbf{E}_k^R - \mathbf{h}_k \mathbf{x}_R^k||_F^2 \tag{3.13}$$

where Ω_k is a $Q \times P$ matrix whose entries $(\omega_k(i), i)$ are ones and the rest are zeros. Note that $P = |\omega_k| \leq K$ where $|\cdot|$ represents cardinality of set ω_k. When post-multiplying a vector by Ω_k, all the zero elements are discarded and the dimension of $\mathbf{x}_T^k$ is reduced to $\mathbf{x}_R^k = \mathbf{x}_T^k \Omega_k \in \mathbb{R}^P$ and also $\mathbf{E}_k^R = \mathbf{E}_k \Omega_k \in \mathbb{R}^{N \times P}$ is the restricted error matrix. Note that this is not needed if the proposed fast OMP method is used. When decomposing $\mathbf{E}_k^R = \mathbf{U}\Delta\mathbf{V}^T$ using SVD, the solution for the updated dictionary atom $\hat{\mathbf{h}}_k = \mathbf{u}_1$, or the first column of $\mathbf{U}$. Now, since $\hat{\mathbf{h}}_k$ has changed, $\mathbf{x}_R^k$ must also be updated. This restricted vector is updated using $\hat{\mathbf{x}}_R^k = \mathbf{v}_1\Delta_{1,1}$ where $\mathbf{v}_1$ is the first row of $\mathbf{V}^T$ and $\Delta_{1,1}$ is the first entry in the Δ matrix. After updating the k^{th} dictionary atom, the same procedure is followed for the $(k+1)^{th}$ atom until all atoms have been individually updated. Then, using this updated dictionary the sparse matrix $\mathbf{X}_m$ can be recomputed. The dictionary update and sparse coding steps are repeated until the stopping condition is met i.e. either maximum number of iterations or a reconstruction error below some threshold.

Table 3.1: K-SVD Algorithm

K-SVD Optimal Dictionary Construction Algorithm:
a) Initialization: Set the dictionary matrix $\mathbf{H}_{m,0} \in \mathbb{R}^{N \times K}$ with K randomly selected l_2 normalized columns of $\mathbf{Y}_m$. Set $t = 1$. Repeat following steps until a stopping rule is met.
b) Sparse Coding Stage: Generate $\mathbf{X}_m$ by computing the sparse representation $\mathbf{x}_{q,t}$ for each $\mathbf{y}_q$ based on $\mathbf{H}_{m,t-1}$ using the Fast-OMP method [28].
c) Dictionary Update Stage: Each column $\mathbf{h}_{k,t-1}$, $k \in [1,...,K]$, in $\mathbf{H}_{m,t-1}$ is updated separately by: 1. Compute $k-$exclusive error matrix $\mathbf{E}_k = \mathbf{Y}_m - \sum_{j \neq k} \mathbf{h}_j \mathbf{x}_T^j$, where $\mathbf{x}_T^j$ is the j^{th} row of $\mathbf{X}_m$. 2. Define column indices of training data $\mathbf{Y}_m$ that use the k^{th} atom in their reconstruction via $\mathbf{H}_{m,t-1}$: $\omega_k = \{i \mid x_T^k(i) \neq 0\}$. 3. Compute $\mathbf{E}_k^R$ and $\mathbf{x}_R^k$, the restricted error matrix and coefficient vector respectively, by selecting only columns of $\mathbf{E}_k$ corresponding to ω_k indices and likewise for entries of $\mathbf{x}_T^k$ (i.e. discard zero entries in the row vector). 4. Apply SVD: $\mathbf{E}_k^R = \mathbf{U}\Delta\mathbf{V}^T$. The updated dictionary column $\hat{\mathbf{h}}_{k,t} = \mathbf{u}_1$, the first column of $\mathbf{U}$ and the updated coefficient vector $\hat{\mathbf{x}}_R^k = \mathbf{v}_1\Delta_{1,1}$, where $\Delta_{1,1}$ is the first and largest singular value in the SVD of $\mathbf{E}_k^R$, and $\mathbf{v}_1$ is the first row of $\mathbf{V}^T$.
Set $t = t + 1$ and repeat until repeat b) and c) until stopping criterion is satisfied.

3.3 Locality Preserving KSVD Dictionary Learning Review

There are two main objectives behind LP-KSVD [19]: (1) establishing a compact dictionary $\mathbf{H}_m = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_K]$ such that linear combinations of $\mathbf{h}_k$'s can approximate the nonlinear ambient space manifold $\mathcal{M}$ of the original data, which is sampled based on $\mathbf{Y}_m$ or the training samples for class m; and (2) learning $\mathbf{h}_k$'s as landmark points capable of preserving the local information on $\mathcal{M}$. The dictionary learning problem of LP-KSVD is thus formalized as

$$\min_{\mathbf{H}_m, \mathbf{X}_m} \{||\mathbf{Y}_m - \mathbf{H}_m\mathbf{X}_m||_F^2\} \tag{3.14}$$

$$s.t. = \begin{cases} x_{k,q} = 0 & \text{if } \mathbf{h}_k \notin \Omega_\tau(\mathbf{y}_q) \ \forall k, q \\ \mathbf{1}^T\mathbf{x}_q = 1 & \forall q \end{cases}$$

where $x_{k,q}$ is the k^{th} element of sparse vector $\mathbf{x}_q$ associated with vector $\mathbf{y}_q$ in $\mathbf{Y}_m$ and $\Omega_\tau(\mathbf{y}_q)$ denotes the neighborhood containing τ nearest dictionary atoms of $\mathbf{y}_q$ in terms of Euclidean distance. The first constraint dictates that every training sample $\mathbf{y}_q$ can only be re-constructed by its τ nearest-neighbor dictionary atoms while the second constraint allows the reconstruction coefficients to be invariant to translation of the data. The main goal is to learn $\mathbf{h}_k$'s as landmark points, which requires that each $\mathbf{h}_k$ be locally representative of a small patch on the manifold $\mathcal{M}$.

The LP-KSVD solves minimization (3.14) iteratively by alternating between solving for $\mathbf{H}_m$ and $\mathbf{X}_m$. This is done by first fixing $\mathbf{H}_m$ and solving for the optimum coefficient matrix $\mathbf{X}_m$ and then, updating $\mathbf{H}_m$ once $\mathbf{X}_m$ is found. Iterations are terminated if either the objective function value reaches some prescribed threshold or a maximum number of iterations has been reached.

3.3.1 Solving for Local Reconstruction Codes

Fixing $\mathbf{H}_m$ to that of the previous iteration, the coefficient matrix $\mathbf{X}_m$ defined in (3.14) can be obtained by equivalently solving the following for each sample $\mathbf{y}_q$:

$$\min_{\mathbf{x}_q \in \mathbf{X}_m} ||\mathbf{y}_q - \sum_{k=1}^{K} x_{k,q}\mathbf{h}_k||_2^2 \tag{3.15}$$

$$s.t. = \begin{cases} x_{k,q} = 0 & \text{if } \mathbf{h}_k \notin \Omega_\tau(\mathbf{y}_q)\ \forall\ k, q \\ \mathbf{1}^T\mathbf{x}_q = 1 & \forall q \end{cases}$$

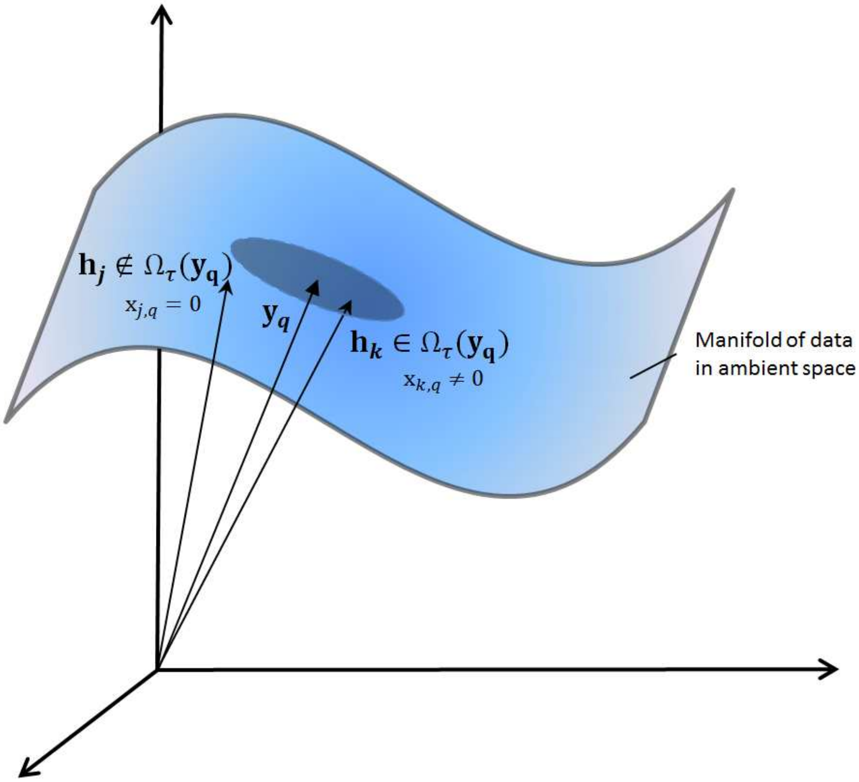

$\mathbf{h_j} \notin \Omega_\tau(\mathbf{y_q})$
$x_{j,q} = 0$
$\mathbf{y_q}$
$\mathbf{h_k} \in \Omega_\tau(\mathbf{y_q})$
$x_{k,q} \neq 0$
Manifold of data in ambient space

Figure 3.2: Local Representations used for LP K-SVD.

where $\mathbf{x}_q$ represents the q-th column in $\mathbf{X}_m$, containing linear representation coefficients for reconstructing $\mathbf{y}_q$, and $x_{k,q}$ is the k-th element in $\mathbf{x}_q$. Letting $\hat{\mathbf{x}}_q$ be a subvector containing only the non-zero elements in $\mathbf{x}_q$. The closed-form solution to (3.15) is given as:

$$\hat{\mathbf{x}}_q = \tilde{\mathbf{x}}_q / \sum \tilde{x}_{k,q} \tag{3.16}$$

where $\tilde{\mathbf{x}}_q = (\mathbf{G} + \eta\mathbf{I})^{-1}\mathbf{1}$, $\mathbf{1}$ is a vector of all ones, and $\mathbf{G} = (\Omega_\tau - \mathbf{y}_q\mathbf{1}^T)^T(\Omega_\tau - \mathbf{y}_q\mathbf{1}^T)$ is the local covariance matrix around sample $\mathbf{y}_q$. In this expression, η is a small constant used to secure numerical stability. This formulation for $\mathbf{G}$ was popularized by a method similar to the LP K-SVD method in a dimensionality reduction process known as Locally Linear Embedding [33]. Note that here, $\Omega_\tau = \Omega_\tau(\mathbf{y}_q)$ the neighborhood of τ dictionary atoms nearest to $\mathbf{y}_q$.

The concept of neighborhood reconstructions can be illustrated in Figure 3.2. In this figure, where a small patch around each new sample $\mathbf{y}_q$ is constructed to define the neighbors of $\mathbf{y}_q$, $\mathbf{h}_k \in \Omega_\tau(\mathbf{y}_q)$.

3.3.2 Local Dictionary Optimization

This step is similar to that in K-SVD algorithm. More specifically, we would like to minimize $||\mathbf{E}_k - \mathbf{h}_k\mathbf{x}_T^k||_F^2$ with respect to $\mathbf{h}_k$ and $\mathbf{x}_T^k$. As $||\mathbf{E}_k - \mathbf{h}_k\mathbf{x}_T^k||_F^2$ is only affected by the set $\omega_k = \{i, |x_T^k(i) \neq 0\}$ which corresponds to the training samples in $\mathbf{Y}_m$ that use column dictionary atom $\mathbf{h}_k$ for which $x_T^k(i) \neq 0$, the objective function is reduced to

$$\hat{\mathbf{h}}_k, \hat{\mathbf{x}}_R^k = \underset{\mathbf{h}_k, \mathbf{x}_R^k}{\text{argmin}} ||\mathbf{E}_k^R - \mathbf{h}_k\mathbf{x}_R^k||_F^2 \tag{3.17}$$

where $\mathbf{E}_k^R$ and $\mathbf{x}_R^k$ were defined before in Section 3.2. Using a similar approach taken in the K-SVD algorithm, the authors in [19] showed that the minimization problem in (3.14) leads to the following solutions for $\hat{\mathbf{h}}_k$, and $\hat{\mathbf{x}}_R^k$

$$\hat{\mathbf{h}}_{k,t} = s\mathbf{u}_1 \tag{3.18}$$

$$\hat{\mathbf{x}}_R^k = \frac{\mathbf{v}_1\Delta_{1,1}}{s} \tag{3.19}$$

where $\mathbf{u}_1$ and $\mathbf{v}_1$ are the first columns of $\mathbf{U}$ and $\mathbf{V}$ respectively; s is a scaling factor, and $\Delta_{1,1}$ is the largest eigenvalue in diagonal of matrix Δ defined before. The process is summarized in Table 3.2.

Table 3.2: LP-KSVD Algorithm

LP-KSVD Optimal Dictionary Construction Algorithm:

a) Initialization Set the initial dictionary matrix $\mathbf{H}_{m,0}$ with K randomly selected normalized columns of $\mathbf{Y}_m$. Set $t = 1$. Repeat following steps until a stopping criterion is met.

b) Local Reconstruction Coding Generate $\mathbf{X}_m$ by computing the local code representation $\mathbf{x}_{q,t}$ for each $\mathbf{y}_q$ based on $\mathbf{H}_{m,t-1}$ at the current iteration using the Local Codes $\hat{\mathbf{x}}_{q,t} = \tilde{\mathbf{x}}_q/(\mathbf{1}^T\tilde{\mathbf{x}}_q)$, where $\tilde{\mathbf{x}}_q = (\mathbf{G} + \eta\mathbf{I})^{-1}\mathbf{1}$ and $\mathbf{G} = (\Omega_{\tau,t-1}(\mathbf{y}_q) - \mathbf{y}_q\mathbf{1}^T)^T(\Omega_{\tau,t-1}(\mathbf{y}_q) - \mathbf{y}_q\mathbf{1}^T)$, the local covariance matrix of $\mathbf{y}_q$ where η is small constant and $\Omega_{\tau,t-1}(\mathbf{y}_q)$ denotes the neighborhood containing τ nearest dictionary atoms of $\mathbf{y}_q$ at time step $t-1$.

c) Local Dictionary Update Each column $\mathbf{h}_{k,t-1}$, $k \in [1, ..., K]$, in $\mathbf{H}_{m,t-1}$ is updated by:

1. Compute $k-$exclusive local error matrix $\mathbf{E}_k = \mathbf{Y}_m - \sum_{j\neq k}\mathbf{h}_j\mathbf{x}_T^j$ where $\mathbf{x}_T^j$ is the j^{th} row vector of $\mathbf{X}_m$ which contains the coefficients used by atom $\mathbf{h}_j$.

2. Define columns of training data $\mathbf{Y}_m$ that concurrently select the k^{th} atom in their reconstruction via $\mathbf{H}_m$ as the matrix Λ and $\mathbf{x}_R^k$ is the succinct row vector containing only the non-zero coefficients of the $\mathbf{x}_T^k$'s, which select the k^{th} atom.

3. Let $\mathbf{E}_k^R$ be the error matrix corresponding to samples Λ estimated by $\mathbf{x}_R^k$, i.e. $\mathbf{E}_k^R = \Lambda - \sum_{j\neq k}\mathbf{h}_j\mathbf{x}_R^j$.

4. For Local Representation Error (LRE) $\sum_{\mathbf{y}_i\in\Lambda}||\mathbf{h}_k - \mathbf{y}_i||_2^2$, the newly updated $\mathbf{h}_{k,t}$ and the $\mathbf{x}_R^k$ that minimize $||\mathbf{E}_k^R - \mathbf{h}_{k,t-1}\mathbf{x}_R^{k,t-1}||_F^2$ and yield the minimum LRE are given by: $\mathbf{U}\Delta\mathbf{V}^T = \mathbf{E}_k^R \Rightarrow$, $\hat{\mathbf{h}}_{k,t} = s\mathbf{u}_1$, $\mathbf{x}_R^k = \frac{\mathbf{v}_1\Delta_{1,1}}{s}$, where $s = \frac{1}{P}\sum_{\mathbf{y}_i\in\Lambda}\frac{\mathbf{u}_1}{||\mathbf{u}_1||_2}\mathbf{y}_i$ is a gain factor.

Set iteration index $t = t + 1$ and repeat b) and c) until stopping criterion is satisfied.

3.4 Conclusion

In this chapter, we reviewed two different methods for determining subspace matrices to be used in the MSC classifier. The K-SVD algorithm was reviewed first which generalizes k-means clustering [18] with flexibility as to what sparsity model is assumed. Both the sparse coding algorithm as well as the dictionary updating process were discussed for the K-SVD and a novel approach for sparse coding using fast OMP iterations was presented. Second, a recent extension of the K-SVD known as the LP K-SVD was reviewed which solves for the sparse codes by using information on the neighborhoods of dictionary atoms and training samples.

Both the K-SVD and LP-KSVD methods have their respective benefits and drawbacks. For the K-SVD method, the atoms are learned to be utilized in a greedy approach which selects the τ-best candidates to represent an observed sample. The LP-KSVD, on the other hand, learns atoms whose sparse linear combinations are representative of their neighborhood of origin on the original data manifold. The LP-KSVD has the added benefit of a closed form solution for Local code computation which does not involve matrix inversion. Instead a search for the $\tau-$nearest neighbors of every training sample is used which are then utilized to represent the sample. For the extreme sparsity case when $\tau = 1$, these algorithms essentially become clustering/learning vector quantization methods, their difference being that for K-SVD the learning goal is to minimize global errors, whereas for LP-KSVD, the focus is instead on minimizing local representations around the training samples. LP-KSVD manages to decrease global error manifold by learning it in a patch-by-patch manner, with each neighborhood increasing global reconstruction capabilities. In Chapter 4 we shall show how the algorithms are used in conjunction with the matched subspace classifier for UXO vs. non-UXO discrimination.

CHAPTER 4

MATCHED SUBSPACE CLASSIFICATION - LINEAR CASE

4.1 Introduction

Detection and Classification problems can be simply described as deciding which of a set of models has most likely generated a given set of observations. This is easily cast into the framework of statistical hypothesis testing, the most basic being the binary hypothesis (two-class) test, where we must decide among only two models. The binary hypothesis test is described as a decision between either the alternative hypothesis H_1 (target present), or the null hypothesis H_0 (background clutter). In signal processing applications, one typically encounters tests where observations are assumed to be a deterministic or stochastic signal additively corrupted by noise under the alternative hypothesis versus that of noise alone under H_0. The most fundamental building block for hypothesis testing is the Bayesian framework [15, 29, 34, 35] where the problem is to minimize the expected Bayesian risk involved with making a decision. Risk is quantified on a problem specific basis but essentially a high risk event corresponds to a highly undesirable event. Minimizing Bayesian risk leads to a solution involving the comparison of a likelihood ratio with a threshold that is dependent on the costs and the *a priori* conditional probability density functions (pdfs). When these costs and conditional pdfs are available, a Bayesian framework is optimal.

In developing a classification framework for sonar imagery, the *a priori* conditional probability densities are impossible to ascertain as this would require prior knowledge of the distribution of the background alone as well as when targets are included with the background. The background distribution in sonar data is highly variable. Studies on bottom return statistics reveal [36–38] that the distribution of the envelope of the matched filtered output is dependent on the frequency, grazing angle, range, and roughness properties of the sea-floor. Rough surface measurements made using high-resolution sonar have indicated that the amplitude distributions can often deviate from a Rayleigh distribution and be better modeled using log-normal, Weibull, or other more complex distributions [39]. This suggests that the underlying complex data are not Gaussian and that the first and second-order moments will not be sufficient for detection and classification purposes. Thus, the standard tests that rely on assumptions of normality can fail to capture the true properties of the background clutter, hence increasing the incident of false alarms.

The generalized likelihood ratio test (GLRT) is typically employed when unknown parameters are involved while the class of pdfs are known. The GLRT detector replaces the unknown parameters by their maximum likelihood estimates (MLE). However, if the pdf of the noise matrix is unknown, or it is not possible to derive the MLE's of the unknown parameters, as is the case in our problem, other methods should be employed.

In [17], the authors proposed the MSC framework specifically for non-Gaussian noise with unknown pdf. The MSC essentially assumes the observations lie in a linear subspace where the estimates of unknown model parameters are obtained via the Least Squares (LS) estimate, which happens to be the MLE estimate under Gaussian additive noise case. However, in reality, the distribution of the noise is unknown and certainly not Gaussian, and hence it is not possible to derive the MLE of the unknown parameters [17].

In this chapter, we begin by reviewing the classical MSC framework along with its invariance properties. This is followed by results of the MSC Classifier on PondEX09-10 and TREX13 datasets when using signal subspaces created from synthetic training datasets and the two dictionary learning methods described in Chapter 3.

4.2 Weighted Matched Subspace Classification Framework

In our application, the MSC classifier operates on AC feature vectors previously described in Chapter 2, with the assumption that each AC vector belongs to a given class, and can be formed using some linear combination of basis vectors associated with that particular class. Our classification problem is referred to as an $M-$ary classification problem in which the following hypothesis is to be tested: Our observation contains spectral features from M possible classes and one is uniquely most likely to be generating the data. For a simple consideration of classifying 'UXO' vs. 'non-UXO,' $M = 2$, i.e. binary hypothesis testing.

We will begin by defining the $M-$ary classification problem by considering $m = 1 \ldots M$ hypotheses each satisfying the signal model,

$$\mathbf{Y} = \mathbf{A}_m \boldsymbol{\Psi} + \mathbf{S}\boldsymbol{\zeta} + \mathbf{N} \quad \forall m \in [1, M] \tag{4.1}$$

In this general model, $\mathbf{Y} \in \mathbb{R}^{N \times Q}$, the observation matrix, consists of a sum of the subspace signal $\mathbf{A}_m \boldsymbol{\Psi}$, with $\mathbf{A}_m \in \mathbb{R}^{N \times (K-R)}$ a matrix whose columns are basis vectors that span the subspace associated with the m^{th} object class, and structured interference signal $\mathbf{S}\boldsymbol{\zeta}$ with subspace matrix $\mathbf{S} \in \mathbb{R}^{N \times R}$, and an additive zero-mean noise matrix $\mathbf{N} \in \mathbb{R}^{N \times Q}$. To account for the co-existence of structured interference in the same target data, we can combine the unknown parameters $\boldsymbol{\Psi} \in \mathbb{R}^{(K-R) \times Q}, \boldsymbol{\zeta} \in \mathbb{R}^{R \times Q}$ into an unknown parameter matrix $[\boldsymbol{\Psi}^T, \boldsymbol{\zeta}^T]^T = \mathbf{X} \in \mathbb{R}^{K \times Q}$, and the known subspace basis atoms into $[\mathbf{A}_m, \mathbf{S}] = \mathbf{H}_m$, allowing the expression in (4.1) to be simplified to the following form

$$\mathbf{Y} = \mathbf{H}_m \mathbf{X} + \mathbf{N}, \quad \forall m \in [1, M] \tag{4.2}$$

Here, our observation matrix (each column is an AC vector) is represented by $\mathbf{Y}$ a sum of the subspace $\mathbf{H}_m \mathbf{X}$ with $\mathbf{H}_m \in \mathbb{R}^{N \times K}$ is a matrix whose columns are basis vectors that span the subspace associated with the m^{th} object class together with background clutter, an unknown parameter matrix $\mathbf{X} \in \mathbb{R}^{K \times Q}$, and an additive zero-mean noise matrix $\mathbf{N} \in \mathbb{R}^{N \times Q}$.

In the ideal scenario where we do not consider atoms for interference on their own, we could consider our structured interference matrix $\mathbf{S} = [\cdot]$, 0×0 matrix. Alternatively, we can express (4.2) for a single observation vector $\mathbf{y}_q$ (or q^{th} column of $\mathbf{Y}$),

$$\mathbf{y}_q = \mathbf{H}_m \mathbf{x}_q + \mathbf{n}_q \tag{4.3}$$

where $\mathbf{n}_q$ is the q^{th} column of matrix $\mathbf{N}$ and $\mathbf{x}_q$ is the q^{th} column of $\mathbf{X}$. The model-based AC subspaces $\mathbf{H}_m$ can be constructed for the m^{th} object class utilizing different subspace reconstruction methods. As discussed in Chapter 3, two variations of the K-SVD method [18], which perform class specific subspace construction with sparse coding can be utilized to build $\mathbf{H}_m$'s.

The distribution of the noise matrix is unknown, and hence it is not possible to derive the MLE of the unknown parameters in $\mathbf{X}$ [17]. Instead, as mentioned earlier, we focus on minimizing the squared error between $\mathbf{H}_m\mathbf{X}$ and $\mathbf{Y}$ under the assumed linear model through LS estimation.

However, the core idea behind the weighting in the weighted MSC [17] is the implicit suppression of large amplitude residuals. This is done by weighting each of the row residual terms in the discriminant function

$$J_m(\mathbf{Y}) = \text{tr}\{(\mathbf{Y} - \mathbf{H}_m\mathbf{X})^T \mathbf{W}(\mathbf{Y} - \mathbf{H}_m\mathbf{X})\} = ||\mathbf{W}^{\frac{1}{2}}(\mathbf{Y} - \mathbf{H}_m\mathbf{X})||_F^2, \ \forall m \in [1, ..., M] \tag{4.4}$$

where $\mathbf{W}$ is a diagonal matrix with weights along the diagonal, corresponding to row-weighting, and $||\mathbf{A}||_F^2 = \text{tr}\{\mathbf{A}\mathbf{A}^T\}$ represents the squared Frobenius norm of matrix $\mathbf{A}$. For a given weight matrix $\mathbf{W}$, the weighted LS estimate of $\mathbf{X}$ under the m^{th} hypothesis is found using:

$$\hat{\mathbf{X}} = (\mathbf{H}_m^T \mathbf{W} \mathbf{H}_m)^{-1} \mathbf{H}_m^T \mathbf{W} \mathbf{Y} \tag{4.5}$$

Substituting $\hat{\mathbf{X}}$ in place of $\mathbf{X}$ in (4.5) into the discriminant function in (4.4) we get

$$J_m(\mathbf{Y}) = \text{tr}\{(\mathbf{Y}^T \mathbf{W}(\mathbf{I} - \mathbf{E}_m)\mathbf{Y})\} \tag{4.6}$$

where $\mathbf{E}_m$ is an oblique projection onto subspace $\langle \mathbf{H}_m \rangle$ (i.e. $\mathbf{E}_m = \mathbf{H}_m(\mathbf{H}_m^T\mathbf{W}\mathbf{H}_m)^{-1}\mathbf{H}_m^T\mathbf{W}$). This discriminant function can be reinterpreted as the orthogonal projection of the weighted data matrix $\mathbf{Z} = \mathbf{W}^{1/2}\mathbf{Y}$ onto subspace $\langle \mathbf{W}^{1/2}\mathbf{H}_m \rangle$.

$$J_m(\mathbf{Z}) = \mathrm{tr}\{(\mathbf{Z}^T(\mathbf{I} - \mathbf{P}_{\mathbf{W}^{1/2}\mathbf{H}_m})\mathbf{Z})\} \tag{4.7}$$

where $\mathbf{P}_{\mathbf{W}^{1/2}\mathbf{H}_m} = \mathbf{W}^{1/2}\mathbf{H}_m(\mathbf{H}_m^T\mathbf{W}\mathbf{H}_m)^{-1}(\mathbf{W}^{1/2}\mathbf{H}_m)^T$ is the projection matrix onto the subspace spanned by the columns of matrix $\mathbf{W}^{1/2}\mathbf{H}_m$.

For a single weighted observation $\mathbf{z}_q = \mathbf{W}^{1/2}\mathbf{y}_q$, this WMSC classifier assigns a class label to an observation vector $\mathbf{y}_q$ in 4.3 based on the following criterion,

$$m^* = \arg \min_{m\in[1,M]} J_m(\mathbf{z}_q) = \arg \min_{m\in[1,M]} \{\mathbf{z}_q^T \left(\mathbf{I} - \mathbf{P}_{\mathbf{W}^{\frac{1}{2}}\mathbf{H}_m}\right) \mathbf{z}_q\} \tag{4.8}$$

While for a sequence of observations (AC Vectors) forming matrix $\mathbf{Y}$, we make decisions using

$$m^* = \mathrm{argmin}_{m\in[1,M]}\, J_m(\mathbf{Z}) = \mathrm{argmin}_{m\in[1,M]}\, \mathrm{tr}\{\mathbf{Z}^T \left(\mathbf{I} - \mathbf{P}_{\mathbf{W}^{\frac{1}{2}}\mathbf{H}_m}\right) \mathbf{Z}\} \tag{4.9}$$

As Figure 4.1 and equation (4.8) indicate, this classifier measures the energy in each of the subspaces $\langle \mathbf{W}^{1/2}\mathbf{H}_m \rangle$ and selects the class label corresponding to the subspace that contains the largest amount of energy. If the noise vector $\mathbf{n}_q$ is assumed to be normal with covariance

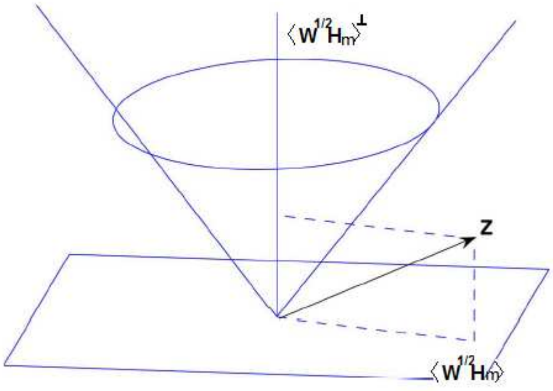

$\langle W^{1/2}H_m \rangle^{\perp}$
z
$\langle W^{1/2}H_m \rangle$

Figure 4.1: Weighted Matched Subspace Classifier Geometric Perspective.

matrix $E\left[\mathbf{n}_q\mathbf{n}_q^H\right] = \sigma^2\mathbf{I}$ then the criterion given in (4.8) also corresponds to the decision that minimizes the probability of assigning an erroneous class label. It is important to note that for $\mathbf{W} = \mathbf{I}$, this weighted MSC reduces to the standard MSC [11].

4.3 Invariance Properties

In this section we will discuss the natural invariances of the original MSC when $\mathbf{W} = \mathbf{I}$. At the end of the section, comments will be made on the invariance properties of the MSC when $\mathbf{W} \neq \mathbf{I}$.

We shall say that a disciminant function $J_m(\mathbf{y})$ is $\mathcal{G}$-invariant if it is invariant to transformation group $\mathcal{G} \in G$, i.e.

$$J_m(\mathcal{G}(\mathbf{y})) = J_m(\mathbf{y}) \tag{4.10}$$

We study invariance class $\mathcal{G}$ in order to gain geometrical insight into the mathematical structure of the classifier.

4.3.1 Invariance to Subspace Rotation

Given the discriminant statistic $J_m(\mathbf{y}) = \mathbf{y}^T(\mathbf{I} - \mathbf{P}_{\mathbf{H}_m})\mathbf{y}$ for a single observation vector $\mathbf{y}$ and under the assumption that $\mathbf{W} = \mathbf{I}$, we would like to demonstrate the invariance of the MSC test statistic w.r.t. rotation of data in the signal subspace $\langle\mathbf{H}_m\rangle$.

Letting group transformation G be $G = \{\mathcal{G}(\mathbf{y}) = \mathbf{Q}_{\mathbf{H}_m}\mathbf{y}\}$, where $\mathbf{Q}_{\mathbf{H}_m} = \mathbf{U}_{\mathbf{H}_m}\mathbf{Q}\mathbf{U}_{\mathbf{H}_m}^T + \mathbf{P}_B$ a matrix which rotates samples within subspace $\langle\mathbf{H}_m\rangle$. Here, $\mathbf{P}_B = \mathbf{P}_{\mathbf{H}_m^\perp}$ is an orthogonal projection into the perp space of signal subspace $\langle\mathbf{H}_m\rangle$ representing interference components not in the span of our signal subspace. To show that $J_m(\mathbf{y})$ is invariant under the above group transformation, we must first show the invariance property, i.e. that for an arbitrary sample $\mathbf{y}$, it holds that $J_m(\mathbf{y}) = J_m(\mathcal{G}(\mathbf{y}))$, $\forall\mathcal{G} \in G$.

First we will demonstrate that $\mathbf{Q}_{\mathbf{H}_m}^T\mathbf{P}_{\mathbf{H}_m}^\perp = \mathbf{P}_{\mathbf{H}_m}^\perp$, this can be done by using orthonormal decompositions of both the projection matrix $\mathbf{P}_{\mathbf{H}_m} = \mathbf{U}_{\mathbf{H}_m}\mathbf{U}_{\mathbf{H}_m}^T$ and using rotation matrix $\mathbf{Q}_{\mathbf{H}_m}$ for orthonormal basis $\mathbf{U}_{\mathbf{H}_m}$ whose columns span subspace $\langle\mathbf{H}_m\rangle$ and rank K rotation matrix $\mathbf{Q}$. Using these decompositions, basic properties of orthonormal basis $\mathbf{U}_{\mathbf{H}_m}$, and the fact that $\langle \mathbf{B} \rangle = \langle \mathbf{H}_m \rangle^{\perp}$ we see

$$\begin{aligned}
\mathbf{Q}_{\mathbf{H}_m}^T \mathbf{P}_{\mathbf{H}_m}^{\perp} &= (\mathbf{U}_{\mathbf{H}_m} \mathbf{Q}^T \mathbf{U}_{\mathbf{H}_m}^T + \mathbf{P}_B)(\mathbf{I} - \mathbf{U}_{\mathbf{H}_m} \mathbf{U}_{\mathbf{H}_m}^T) \\
&= (\mathbf{U}_{\mathbf{H}_m} \mathbf{Q}^T \mathbf{U}_{\mathbf{H}_m}^T - \mathbf{U}_{\mathbf{H}_m} \mathbf{Q}^T \mathbf{U}_{\mathbf{H}_m}^T \mathbf{U}_{\mathbf{H}_m} \mathbf{U}_{\mathbf{H}_m}^T + \mathbf{P}_B \mathbf{P}_{\mathbf{H}_m}^{\perp}) \\
&= (\mathbf{U}_{\mathbf{H}_m} \mathbf{Q}^T \mathbf{U}_{\mathbf{H}_m}^T - \mathbf{U}_{\mathbf{H}_m} \mathbf{Q}^T \mathbf{U}_{\mathbf{H}_m}^T + \mathbf{P}_{\mathbf{H}_m}^{\perp 2}) \\
&= \mathbf{P}_{\mathbf{H}_m}^{\perp}
\end{aligned} \tag{4.11}$$

Now, it is simple to show the invariance like so

$$\begin{aligned}
J_m(\mathcal{G}(\mathbf{y})) &= \mathbf{y}^T \mathbf{Q}_{\mathbf{H}_m}^T \mathbf{P}_{\mathbf{H}_m}^{\perp} \mathbf{Q}_{\mathbf{H}_m} \mathbf{y} \\
&= ||\mathbf{y}^T \mathbf{Q}_{\mathbf{H}_m}^T \mathbf{P}_{\mathbf{H}_m}^{\perp}||^2 \\
&= ||\mathbf{y}^T \mathbf{P}_{\mathbf{H}_m}^{\perp}||^2 \\
&= \mathbf{y}^T \mathbf{P}_{\mathbf{H}_m}^{\perp} \mathbf{y} = J_m(\mathbf{y})
\end{aligned} \tag{4.12}$$

i.e. $J_m(\mathbf{y})$ is $\mathcal{G}-$invariant.

4.3.2 Invariance to Scaling

Here we show scale invariance of (4.4) property that can be achieved with an slight modification of the discriminant function in (4.4). We would like to show that

$$J'_m(\mathbf{y}) = \frac{\mathbf{y}^T \mathbf{P}_{\mathbf{H}_m}^{\perp} \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \tag{4.13}$$

is invariant to scaling of data samples $\mathbf{y}$. That is, we would like to show $J'_m(\mathbf{y})$ is $\mathcal{S}-$invariant where $\mathcal{S} \in S = \{\mathcal{S}(\mathbf{y}) = \alpha \mathbf{y},\ \alpha \in \mathbb{R}\}$ is the family of all scaling transformations of the data. We can show that $J'_m(\mathcal{S}(\mathbf{y})) = J'_m(\mathbf{y})$ like so:

$$\begin{aligned}
J'_m(\mathcal{S}(\mathbf{y})) &= J'_m(\alpha \mathbf{y}) \\
&= \frac{\alpha^2 \mathbf{y}^T \mathbf{P}_{\mathbf{H}_m}^{\perp} \mathbf{y}}{\alpha^2 \mathbf{y}^T \mathbf{y}} \\
&= \frac{\mathbf{y}^T \mathbf{P}_{\mathbf{H}_m}^{\perp} \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \\
&= J'_m(\mathbf{y})
\end{aligned} \tag{4.14}$$

i.e. $J'_m(\mathbf{y})$ is invariant to scaling. It is important to note that since the denominator $\mathbf{y}^T\mathbf{y}$ is the power of the sample, the statistic in (4.13) is essentially giving the cosine of the angle between the sample and the signal subspace. It is also important to note that since $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$ for all rotation matrices $\mathbf{Q}$, it is easy to see that $\mathbf{y}^T\mathbf{Q}^T\mathbf{Q}\mathbf{y} = \mathbf{y}^T\mathbf{y}$ and thus the $\mathcal{G}-$invariance previously shown for $J_m(\mathbf{y})$ has not been lost in the modified $J'_m(\mathbf{y})$. For the testing of all MSC methods, a normalized form of the discriminant functions was utilized. Figure 4.2 demonstrates the geometric interpretation of the invariance properties of the MSC.

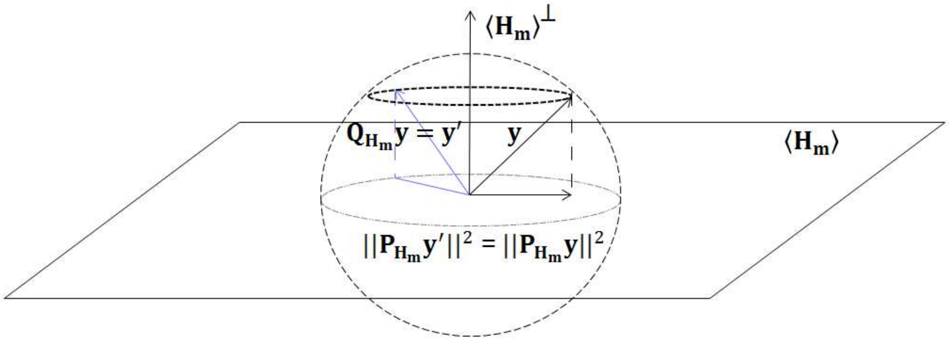

$\langle \mathbf{H_m} \rangle^\perp$
$\mathbf{Q_{H_m} y = y'}$
$\mathbf{y}$
$\langle \mathbf{H_m} \rangle$
$||\mathbf{P_{H_m} y'}||^2 = ||\mathbf{P_{H_m} y}||^2$

Figure 4.2: Invariance of Matched Subspace Classifier

4.3.3 Invariances for $\mathbf{W} \neq \mathbf{I}$

In the scenario where $\mathbf{W} \neq \mathbf{I}$, the story is quite similar. However, the properties can only be shown for a dataset and subspace that have simultaneously been weighted by the weighting matrix $\mathbf{W}$. As was shown in equation (4.7) the discriminant function can be reinterpreted as the orthogonal projection of the weighted data vector $\mathbf{z} = \mathbf{W}^{1/2}\mathbf{y}$ onto subspace $\langle \mathbf{W}^{1/2}\mathbf{H}_m \rangle$. Thus, the same invariance properties hold for this weighted MSC.

4.4 Computation of Weighting Matrix

A robust implementation of the weighted MSC is achieved by assigning a weighting matrix which can limit the influence of large error amplitudes for certain measurements or de-emphasize less important AC features by assigning small corresponding weights. Two estimators were suggested in [17] for creating non-identity weighting matrices. Here we briefly review the method of Least Median of Squares estimators (LMedS) which provides a robust estimate of these parameters. This method will be briefly outlined below.

The LMedS estimator seeks to solve the following minimization problem:

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \; \underset{i}{\operatorname{med}} ||\mathbf{r}_i||_2^2 \tag{4.15}$$

where $\mathbf{r}_i$ is the i^{th} row of the residual error matrix $\mathbf{R} = \mathbf{Y} - \mathbf{H}_m\mathbf{X}$.

The idea is to randomly select N_C subsets of $C < N$ rows from the measurement matrix $\mathbf{Y}$, and then fit a parameter matrix $\mathbf{X}_j$ to the data for each subset. This is done via a LS estimate formed of the C rows of $\mathbf{Y}$ and the corresponding rows of $\mathbf{H}_m$ denoted by $\mathbf{Y}^j$ and $\mathbf{H}_m^j$ respectively,

$$\mathbf{X}_j = (\mathbf{H}_m^{jT}\mathbf{H}_m^j)^{-1}\mathbf{H}_m^{jT}\mathbf{Y}^j \tag{4.16}$$

The dimension C of the subsets $\mathbf{Y}^j$ and $\mathbf{H}_m^j$ must clearly be larger than the number of parameters to be estimated (i.e. $C > K$ since $\mathbf{X} \in \mathbb{R}^{K \times Q}$ and we desire a full row rank estimation). The median, M_j, of each set of squared row residuals is then determined

$$M_j = \underset{i=1,...,N}{\operatorname{med}} ||\mathbf{r}_i(\mathbf{X}_j)||_2^2, \quad j = 1, ..., N_c \tag{4.17}$$

The LMedS solution is the estimator for which the corresponding M_j is minimum among all N_c different M_j's. The LMedS procedure is used to determine a set of binary weights (diagonal elements of $\mathbf{W}$), $w_i \in \{0, 1\}$, on which we can build our subspace classifier and which modifies the estimator of $\mathbf{X}$ to

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \sum_i w_i ||\mathbf{r}_i||_2^2, \tag{4.18}$$

with

$$w_i = \begin{cases} 1 & \text{for} \quad ||\mathbf{r}_i||_2^2 \leq (2.5\hat{\sigma})^2 \\ 0 & \text{else} \end{cases} \tag{4.19}$$

and $\hat{\sigma}$ is the robust standard deviation [40] given as

$$\hat{\sigma} = (1/\sqrt{D})[1 + 5/(N - C)]\sqrt{M_{min}} \tag{4.20}$$

where $M_{min} = \min_j\{M_j\}$ and D denotes median of a centralized Chi-Squared with Q degrees of freedom distribution (i.e. χ_Q^2). For fixed C and ξ, where ξ is the maximum fraction of 'large' amplitudes in an observation, the probability that at least one of the N_C subsets is good is $P = 1 - [1 - (1 - \xi)^{N_C}]^C$. We can then calculate N_C as

$$N_C = \frac{\log[1 - P]}{\log[1 - (1 - \xi)^C]} \tag{4.21}$$

Essentially, the medians of several subsets of row residuals are computed and the least median is used as a measure of deviation from $\hat{\sigma}$. This is then used to find the best estimators $\mathbf{X}$ for describing $\mathbf{Y}$ in $\mathbf{H}_m$ by removing highly deviant residual values completely. This method, as mentioned before, is intended to place appropriate weights in matrix $\mathbf{W}$ in order to attenuate large deviations in the representation. Preliminary testing of these methods did not yield very promising results and hence, the standard $\mathbf{W} = \mathbf{I}$ weighting was chosen for all experiments performed in this work.

4.5 Modified MSC

In order to use this MSC classifier successfully, it is crucial that class-dependent dictionary matrices $\mathbf{H}_m$ are accurately constructed from the training datasets in each class m. In this work, we used and tested two signal-specific dictionary learning methods, namely K-SVD [18] and LP-KSVD [19], using model-generated datasets as described in Chapter 3. These dictionary learning algorithms use sparse coding methods such as Orthogonal Matching Pursuit (OMP) or Basis Pursuit (BP) [30] to reduce the data into sparse vectors as columns of $\hat{\mathbf{X}}$ that contain only a few nonzero elements. Now, since these estimates differ from those of the LS solution [29] used to develop MSC decision rule in (4.4), a modification to this rule is needed. This simple modification involves the following decision rule instead of 4.8

$$m^* = \arg \min_{m \in [1,M]} J_m(\mathbf{Y}) = ||\mathbf{Y} - \mathbf{H}_m \hat{\mathbf{X}}_{\text{OMP}}||_F^2 \tag{4.22}$$

with $\hat{\mathbf{X}}_{\text{OMP}}$ being the estimate of $\mathbf{X}$ generated using the OMP algorithm when dictionary $\mathbf{H}_m$ is used.

4.6 Classification Results of Linear MSC

In this section we will give a review of the results obtained using the MSC framework with both the K-SVD and LP K-SVD subspace learning methods. Results are presented in the form of ROC curves and binary confusion matrices for 'UXO' vs. 'non-UXO' class objects. Results are first presented on the objects from the PondEX09 and PondEX10 experiments which were all captured at a fixed 10 m range from the sonar interface which was insonifying the target scene. This subsection briefly explains the experimental setup and tabulates the results of both methods. The Pond results are first presented. These are followed by a subsection on the same classifier's performance on the TREX13 dataset, a much larger dataset which was broken into 3 experiments, namely all range for objects at ranges $\{10 - 40\}$ m, short range for objects at ranges $\{10 - 25\}$ m, and long range object testing for objects at $\{30 - 40\}$ m range. Training samples for the linear MSC classifiers were chosen from those objects previously described in Table 2.1 which correspond to the appropriate ranges featured in the test dataset. For PondEX testing, trained and tested features were truncated to $N_{pond} = 295$ frequency bins corresponding to the $1 - 30$kHz freqeuncy response bins. For TREX testing, trained and tested features were truncated to $N_{pond} = 275$ frequency bins corresponding to the $3 - 30$kHz freqeuncy response bins.

4.6.1 PondEX Testing Results

To make a classification decision for a given sonar ping observation, the corresponding AC data vector of dimension $N_{Pond} = 295$ that contains the intended range ($1 - 30$ kHz) of spectral features of an underwater UXO or non-UXO object at a particular aspect is applied to the classifier. For the K-SVD and LP K-SVD MSCs, this decision-making is implemented using the modified classification rule given in (4.22). The same process can be carried out for all object aspects in a run in the PondEX09 and PondEX10 testing dataset. However, using the more general decision rule (4.22) for MSC, one can make decision on an augmented data matrix $\mathbf{Y}$ that contains multiple AC observation vectors at different aspects. This multi-aspect decision-making provides better opportunity to discriminate between UXO and non-UXO object classes. Moreover, multi-aspect classification is more amenable to actual operational situations where several views from an underwater objects are received. Here, we use three aspects of an object to perform classification decision. Although, in the PondEX experiments aspect separation is uniform due to the rail system, to account for platform instability in a realistic data collection scenario aspect separation is modeled by a uniformly distributed random variable $s \sim unif\ \{8, 16\}$ which shuffles data, beginning with the first aspect in a given run that meets the power threshold. Note that the procedure does not shuffle between rotations of objects, only the order in which the aspects of a single linear run are encountered is changed.

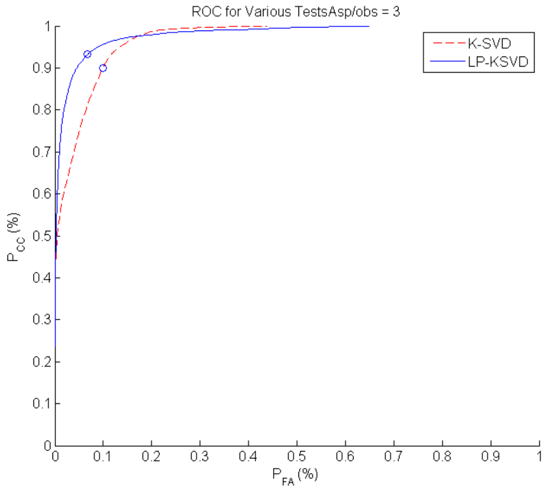


Figure 4.3: Pond ROC K-SVD vs. LP-K-SVD (3 Asp/dec.)

Table 4.1: Linear MSC Confusion Matrix PondEX

Truth\Dec.	‘UXO’	‘non-UXO’
‘UXO’	0.8800	0.1200
‘non-UXO’	0.0159	0.9840

Figure 4.3 gives the Receiver Operating Characteristic (ROC) curves of the MSC classifier using K-SVD, and LP-KSVD dictionaries. As can be observed by the 2 circles in Figure 4.3, the ROC curves for the K-SVD, and LP-KSVD exhibit knee-point (the point where $P_{CC} + P_{FA} = 1$) probability of correct classification of $P_{CC,KSVD} \approx 90.07\%$, $P_{CC,LP-KSVD} \approx 93.20\%$ and probability of false alarm $P_{FA,KSVD} \approx 9.93\%$, $P_{FA,LP-KSVD} \approx 6.80\%$, respectively. These results show that the MSC performs very well in discriminating 'UXO' vs. 'non-UXO' in the PondEX dataset in spite of the fact that there are indeed obvious discrepancies between the model data used for training and the AC sonar data of actual objects. Additionally, the best overall results for the PondEX datasets are obtained for a classifier trained and tested using data from LP-KSVD dictionary learning method.

Table 4.1 displays the confusion matrix for the MSC. It was found that the most common type I error (i.e. 'UXO' mis-classification) occurred for the Stainless Steel UXO object (Tables 2.1, object 2), which was most commonly mis-classified as 2 ft Aluminum Cylinder (non-UXO). All the observed type II errors (i.e. False Alarms) occurred for samples of Aluminum Pipe object, which were almost equally mis-classified as the Aluminum and Stainless Steel UXO classes. These results suggest that the inclusion of ambiguous or misleading aspects in the training of subspaces could potentially have a significant influence on the power of FRM data in representing different classes. Additionally, the results in Figure 4.3 and Table 4.1 reveal much improved performance of our approach when compared to the results of previously tested classifiers [23], including: kernel matching pursuit, support vector machine, and relevance vector machine, which at best demonstrated UXO vs. non-UXO discrimination with $P_{CC} \approx 90\%$ and $P_{FA} \approx 10\%$. However, in these previous cases the classifiers were trained on real sonar data whereas here training was exclusively done on model-generated datasets via the fast ray model [7] explained in Chapter 2.

4.6.2 TREX Testing Results

Using the discriminant rule given in (4.22) a class membership was decided for each observation matrix $\mathbf{Y}$ which contained three AC feature vectors (as its columns) extracted from data at three different aspects with certain separation. Identical processing and experiment setup to the PondEX testing was utilized for TREX testing, including simulated platform instability as described in Section 4.6.1. To prepare the data for the MSC classifier, the filtering method described in Section 2.3.3 was used to isolate the object response from those of the adjacent objects and to eliminate possible overlap in the sonar backscattered from the objects. For analyzing the performance of the MSC classifier, three experiments were conducted where the testing data sets varied depending on the range of the targets from the sonar rail. These experiments corresponded to: (a) complete testing data for all the object types, ranges, and orientations given in Table 2.4; (b) a subset of the entire test set corresponding to *short range* i.e. $\{10m, 15m, 20m, 25m\}$; and (c) a subset of the entire test set corresponding to *long range* i.e. $\{30m, 35m, 40m\}$. This was done to examine how the performance of the trained classifier varies depending on range in the actual TREX13 data sets. As was done with the PondEX dataset, AC features in the TREX13 experiments were truncated to $N_{TREX} = 275$ corresponding to the $3 - 30$ kHz frequency bins of the TREX chirp which had meaningful acoustic response data. The ROC curves of the MSC is presented in Figures 4.4 - 4.6 for Experiments 1-3, respectively. Tables 4.2 - 4.4 also display the corresponding confusion matrices for these three testing datasets. Note that the results in all of these tables are generated using a hard-limiting threshold which was determined from training dataset, typically this threshold was very near to $\gamma = 1$. The circles on the ROC curves in Figures 4.4 - 4.6, correspond to the knee-points of ROC curves at which $P_{CC} + P_{FA} = 1$ where P_{CC} and P_{FA} are the correct classification and false alarm rates, respectively. The knee-points of the MSC classifier exhibit $P_{CC} = 83\%$, $P_{CC} = 86\%$, and $P_{CC} = 91\%$ when tested on the data sets in Experiments 1-3, respectively. Comparison of these results with those presented in [23] on similar TREX13 data sets using a Relevance

Vector Machine classifier [41] indicates somewhat similar classification performance. However, as mentioned before, [23, 42] utilized real data for training of the classifiers while our system was trained entirely on model generated data.

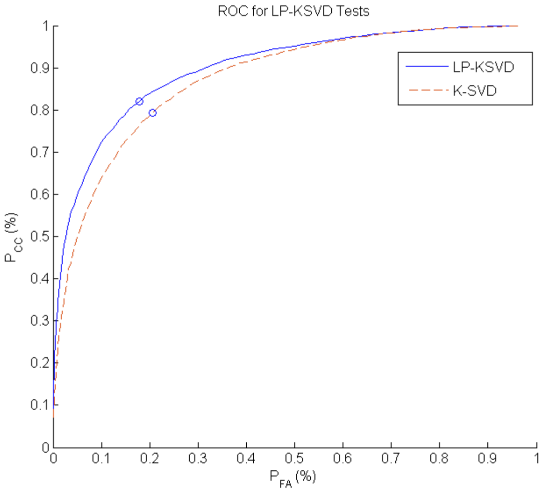


Figure 4.4: ROCs TREX Experiment 1.

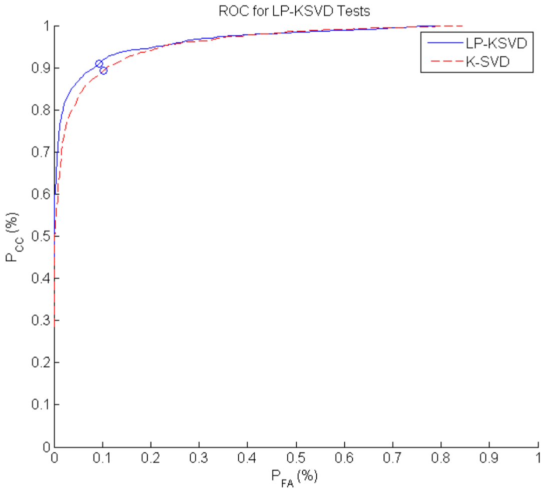


Figure 4.6: ROCs TREX Experiment 3.

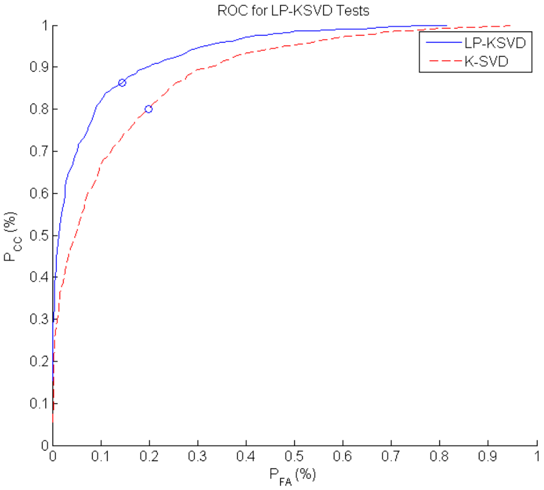


Figure 4.5: ROCs TREX Experiment 2.

Further investigation of the misclassified objects in the confusion matrices in Tables 4.2-4.4 revealed that the most common source of confusion (false alarm) was between 2 ft Aluminum Cylinder objects which were misclassified as the Aluminum UXO class. This

Table 4.2: Confusion Matrices for Experiment 1

	KSVD		LPKSVD	
Truth\Dec.	‘UXO’	‘non-UXO’	‘UXO’	‘non-UXO’
‘UXO’	0.8496	0.1504	0.9056	0.0944
‘non-UXO’	0.2162	0.7838	0.3748	0.6252

Table 4.3: Confusion Matrices for Experiment 2

	KSVD		LPKSVD	
Truth\Dec.	‘UXO’	‘non-UXO’	‘UXO’	‘non-UXO’
‘UXO’	0.8671	0.1329	0.8138	0.1862
‘non-UXO’	0.1470	0.8530	0.2069	0.7931

Table 4.4: Confusion Matrices for Experiment 3

	KSVD		LPKSVD	
Truth\Dec.	‘UXO’	‘non-UXO’	‘UXO’	‘non-UXO’
‘UXO’	0.8140	0.1860	0.7968	0.2032
‘non-UXO’	0.0209	0.9791	0.0356	0.9644

could be due to the fact that these objects have similar size and composition. The most common misclassification of UXO objects was Howitzer with a Cap (mostly the air-filled) which were commonly misclassified as aluminum pipe and cylinder samples. AC plots of objects 8 and 4 (Howitzer w/Cap (air fill) and Aluminum UXO) from Table 2.4 can be seen in Figures 2.5(b) and 2.5(a) which shows the variation that can occur between two UXO class objects. Additionally, it is interesting to note that misclassifications largely occurred in the 10 m and 15 m runs which warrants further investigation. For the long-range Experiment 3, for all of objects besides object 6, a 105 mm air-filled bullet UXO type from Table 2.4, much improved classification results were observed. Degraded performance in the all-range data set (Experiment 1) is another indication of the difficulty in generalizing across multiple ranges especially when the MSC classifier is trained exclusively on synthesized data sets. The author believes that enforcing sparse estimations in the classifier is a key difference between the classifiers used in this chapter and those used in Chapter 5, which uses a batch (all dictionary atoms at once) LS estimate from all representative class samples in its power projection terms.

While there could be a few reasons for the discrepancy in classification performance for differing ranges in the TREX datasets, the primary cause is believed to be the sensitivity of the acoustic color templates to grazing angle. Since the source/receiver are at a fixed height above the seafloor, as one goes out in range the grazing angle changes from 43° at 5 m, to 22° at 10 m, 11° at 20 m, and 5.4° at 40 m. Thus, going from 5 to 20 m the grazing angle sweeps through the critical grazing angle (i.e. the point where total internal reflection occurs) and the phase of the reflection coefficient goes through a phase shift of π. The AC templates used in testing are constructed from a fixed angular width of about 30° regardless of range to the target. In TREX13 and PondEX10, the targets were rotated by 20° increments. The AC templates for 20° and 40° orientations span $(5^\circ, 35^\circ)$ and $(25^\circ, 55^\circ)$. The overlapping region of $(25^\circ, 35^\circ)$ is used to help stitch the individual sections into the final plot.

Despite the variations within classes that occur due to range/grazing angle dependencies, the two linear systems using LP K-SVD and K-SVD managed to achieve low classification errors on the TREX datasets yielding $P_{CC} = 83\%$, $P_{FA} = 17\%$ and $P_{CC} = 80\%$, $P_{FA} = 20\%$, respectively, for UXO vs. non-UXO discrimination across all ranges. This suggests that enforcing a sparse model assumptions for observations made in AC feature space provides a solution to identifying class membership of samples that have components from more than one class subspace.

4.7 Conclusion

In this chapter we first reviewed classical linear MSC used for M-ary classification. The weighted MSC, and methods for computing the weight matrix were also discussed. The procedure of subspace matching was explained as well as the manner in which the weighting matrix augments the data and signal subspace to minimize errors from unreliable feature components. Invariance properties of the MSC under an identity as well as general weighting matrices were demonstrated. Classification results of the MSC system when used in conjunction with K-SVD and LP K-SVD subspace learning methods on three sonar sonar datasets introduced in Chapter 2 were also provided. These results indicated that the LP K-SVD trained MSC system performs best at UXO vs. non-UXO discrimination across all datasets. The LP K-SVD and K-SVD systems managed to acheive low classification errors on the PondEX datasets yielding $P_{CC} = 93\%$, $P_{FA} = 7\%$ and $P_{CC} = 90\%$, $P_{FA} = 10\%$ for UXO vs. non-UXO discrimination, respectively.

CHAPTER 5

MATCHED SUBSPACE CLASSIFICATION - KERNEL CASE

5.1 Introduction

It is well-known that certain non-linearly separable classification problems can be converted to linearly separable problems by mapping the data to a high dimensional feature space using kernel-producing nonlinear mapping functions [41]. Thus, to improve the performance of the linear MSC, in this chapter we investigate the kernel-based extension of the MSC classification framework. This method basically extends the hypothesis test for the linear subspace model described in (4.1) to a higher dimensional feature domain $\mathcal{F}$ associated with nonlinear mapping $\Phi(.)$. That is, the mapping $\Phi(.) : \mathcal{Y} \longrightarrow \mathcal{F}$ maps data in $\mathcal{Y}$ to points in $\mathcal{F}$, which are of much higher dimensionality when compared to the original signal domain, and in this mapped space, the mapped samples hopefully become linearly separable. The decision rules in the original observation space $\mathcal{Y}$ can completely be represented in the feature space $\mathcal{F}$ using only kernel eigenvector representations of kernel Gram matrices and their empirical maps [41].

In this chapter, we begin by reviewing the kernelized MSC framework proposed in [20]. This method essentially extends the framework of subspace matching to a higher dimensional feature space associated with a kernel function satisfying the Mercer theorem condtions [15]. This is followed by classification results of the kernel MSC on the PondEX09-10 and TREX13 datasets when trained exclusively on synthetic data.

5.2 Kernel MSC

In this section we briefly outline the steps required to convert the MSC likelihood function in 4.8 to a mapped feature domain.

Let $\Phi : \mathcal{Y} \longrightarrow \mathcal{F}$ be a mapping which maps data points in data domain $\mathcal{Y}$ to points in feature space $\mathcal{F}$, typically of much greater dimensionality than the original data domain. The MSC expressions in the feature space can completely be represented in the data space using only kernel eigenvector representations and empirical kernel maps. This method extends the hypothesis tests for the linear subspace model, described previously in the data domain, into the feature domain $\mathcal{F}$ associated with nonlinear mapping Φ. To begin, we will rewrite the linear subspace model in 4.1 as it would appear in the high dimensional feature space $\mathcal{F}$, i.e.

$$\begin{aligned}
\mathbf{H}_{0_\Phi} : \Phi(\mathbf{y}) &= \mathbf{S}_\Phi \boldsymbol{\zeta}_\Phi + \mathbf{n}_\Phi, && \text{Target Absent} \\
\mathbf{H}_{1_\Phi} : \Phi(\mathbf{y}) &= \mathbf{A}_\Phi \Psi_\Phi + \mathbf{S}_\Phi \boldsymbol{\zeta}_\Phi + \mathbf{n}_\Phi \\
&= [\mathbf{A}_\Phi \ \mathbf{S}_\Phi] \begin{bmatrix} \Psi_\Phi \\ \boldsymbol{\zeta}_\Phi \end{bmatrix} + \mathbf{n}_\Phi, && \text{Target Present} \\
&= \mathbf{H}_\Phi \mathbf{X}_\Phi + \mathbf{n}_\Phi
\end{aligned} \tag{5.1}$$

Here, $\mathbf{A}_\Phi$ and $\mathbf{S}_\Phi$ represent orthonormal matrices (though not necessarily mutually orthogonal), the columns of which span the signal and background subspaces in $\mathcal{F}$, respectively. As before in the linear case, we combine signal and interference components in $\mathbf{H}_\Phi = [\mathbf{A}_\Phi, \mathbf{S}_\Phi]$ and their respective parameter matrices in $\mathbf{X}_\Phi = [\Psi_\Phi^T, \boldsymbol{\zeta}_\Phi^T]^T$. Under the assumption of additive Gaussian noise in the $\mathcal{F}$-domain, a bit of a *wild* assumption since the noise was not likely to be Gaussian even in the original domain, the MSC is a form of GLRT [20] resulting in a test

$$J_\Phi(\mathbf{y}) = \frac{\Phi(\mathbf{y})^T(\mathbf{P}_{\mathbf{I}_\Phi} - \mathbf{P}_{\mathbf{S}_\Phi})\Phi(\mathbf{y})}{\Phi(\mathbf{y})^T(\mathbf{P}_{\mathbf{I}_\Phi} - \mathbf{P}_{\mathbf{H}_\Phi})\Phi(\mathbf{y})} \tag{5.2}$$

The basic operations described in (5.2) require taking a ratio of sums of the following three terms: $\Phi(\mathbf{y})^T \mathbf{P}_{\mathbf{I}_\Phi} \Phi(\mathbf{y})$, $\Phi(\mathbf{y})^T \mathbf{P}_{\mathbf{S}_\Phi} \Phi(\mathbf{y})$, and $\Phi(\mathbf{y})^T \mathbf{P}_{\mathbf{H}_\Phi} \Phi(\mathbf{y})$. The assumption behind the usage of $\mathbf{P}_{\mathbf{I}_\Phi}$ rather than $\mathbf{I}$ is that our training samples only occupy a small union of subspaces in possibly infinite dimensional feature space associated with the chosen kernel. Also, $\mathbf{P}_{\mathbf{H}_\Phi}$ and $\mathbf{P}_{\mathbf{S}_\Phi}$ are the projection matrices associated with signal and structured noise subspaces, respectively.

Clearly, the terms in (5.2) are not computable in their explicit form as $\Phi(\mathbf{y})$ is potentially infinite dimensional. However, we are only interested in computing the inner products represented in (5.2). In order to find expressions for these terms, we will utilize concepts of kernel Principal Component Analysis (PCA) [41].

5.2.1 Review of Kernel PCA

The terms in (5.2) are converted into kernel forms using kernel PCA which seeks to express the eigenvectors of the covariance matrix of the mapped data in terms of eigenvectors of the covariance matrix in the original data domain.

Consider the eigenvectors $\mathbf{e} \in \mathcal{F}$ with corresponding eigenvalue $\lambda \neq 0$ of the covariance matrix $\mathbf{C}_\Phi$ for centered training samples $\Phi(\mathbf{y}_i),\ i \in [1, ..., Q]$. Then, we have the eigenvalue problem

$$\lambda \mathbf{e} = \mathbf{C}_\Phi \mathbf{e} \ = \frac{1}{Q} \sum_{i=1}^{Q} \Phi(\mathbf{y}_i) \Phi(\mathbf{y}_i)^T \mathbf{e} = \frac{1}{Q} \sum_{i=1}^{Q} \langle \Phi(\mathbf{y}_i), \mathbf{e} \rangle \Phi(\mathbf{y}_i) \tag{5.3}$$

The previous equation shows that any of the eigenvectors $\mathbf{e}$ with $\lambda \neq 0$ can be spanned by the training samples $\Phi(\mathbf{y}_1), ..., \Phi(\mathbf{y}_Q)$ i.e.,

$$\mathbf{e} = \sum_{i=1}^{Q} \gamma_i \Phi(\mathbf{y}_i) = \Phi_{\mathbf{Y}} \boldsymbol{\gamma} \tag{5.4}$$

where $\Phi_{\mathbf{Y}} = [\Phi(\mathbf{y}_1), ..., \Phi(\mathbf{y}_Q)]$ and $\boldsymbol{\gamma} = [\gamma_1, ..., \gamma_Q]^T$ a vector of all the inner products $\langle \Phi(\mathbf{y}_i), \mathbf{e} \rangle$. Substituting (5.4) into (5.3) and multiplying both sides of (5.3) from the left by $\Phi(\mathbf{y}_n)^T$ for $n \in [1, ..., Q]$, yields

$$\begin{aligned}
&\lambda \sum_{i=1}^{Q} \gamma_i \langle \Phi(\mathbf{y}_n), \Phi(\mathbf{y}_i) \rangle \\
&\qquad = \frac{1}{Q} \sum_{i=1}^{Q} \gamma_i \Phi(\mathbf{y}_n)^T \Phi(\mathbf{y}_i) \Phi(\mathbf{y}_i)^T \sum_{j=1}^{Q} \Phi(\mathbf{y}_j) \qquad (5.5) \\
&\qquad = \frac{1}{Q} \sum_{i=1}^{Q} \gamma_i \langle \Phi(\mathbf{y}_n), \Phi(\mathbf{y}_i) \sum_{j=1}^{Q} \langle \Phi(\mathbf{y}_j), \Phi(\mathbf{y}_i) \rangle \rangle, \forall n \in [1, ..., Q]
\end{aligned}$$

If we denote $\mathbf{K} = \{K_{i,j} = \langle \Phi(\mathbf{y}_i), \Phi(\mathbf{y}_j) \rangle = k(\mathbf{y}_i, \mathbf{y}_j)\}$ i.e. the Gram matrix of inner products, then (5.5) can be written as

$$Q\lambda \mathbf{K} \boldsymbol{\gamma} = \mathbf{K}^2 \boldsymbol{\gamma} \qquad (5.6)$$

Assuming that $\mathbf{K}$ is full rank, the solution of the generalized eigenvalue problem in (5.6) are obtained by solving the following eigenvalue problem:

$$Q\lambda \boldsymbol{\gamma} = \mathbf{K} \boldsymbol{\gamma} \qquad (5.7)$$

where $\boldsymbol{\gamma}$ turn out to be the eigenvectors with nonzero eigenvalues $Q\lambda$ of the kernel matrix $\mathbf{K}$. Note that $\boldsymbol{\gamma}$ need to be normalized by the square root of their corresponding eigenvalues, and the kernel matrix $\mathbf{K}$ needs to be properly centered if the original data did not have zero mean. The centered kernel matrix is given by $\hat{\mathbf{K}} = (\mathbf{K} - \mathbf{1}_Q \mathbf{K} - \mathbf{K} \mathbf{1}_Q + \mathbf{1}_Q \mathbf{K} \mathbf{1}_Q)$ where the $Q \times Q$ matrix $(\mathbf{1}_Q)_{i,j} = \frac{1}{Q}$.

5.2.2 Kernel GLRT in $\mathcal{F}$-Domain

To kernelize the GLRT in (5.2), we will write the inner products in the projections in terms of kernel eigenvector representations which are computable. Using the result that eigenvectors of any data covariance matrix in mapped space $\mathcal{F}$ can be written as a vector in the span of the mapped training samples $\Phi(\mathbf{y})$, $\mathbf{S}_\Phi$ and $\mathbf{A}_\Phi$ can be written as

$$\mathbf{S}_\Phi = [\mathbf{e}_s^1 \mathbf{e}_s^2 ... \mathbf{e}_s^{Q_s}] = \Phi_{\mathbf{Y_S}} \mathcal{S} \qquad (5.8)$$

$$\mathbf{A}_\Phi = [\mathbf{e}_a^1 \mathbf{e}_a^2 ... \mathbf{e}_a^{Q_a}] = \Phi_{\mathbf{Y_A}} \mathcal{A} \qquad (5.9)$$

where $\mathbf{e}_s^i$ and $\mathbf{e}_a^j$ are the significant eigenvectors of $\mathbf{C}_{\mathbf{S}_\Phi}$ and $\mathbf{C}_{\mathbf{A}_\Phi}$, respectively, and

$$\Phi_{\mathbf{Y_S}} = [\Phi(\mathbf{y}_1)\Phi(\mathbf{y}_2)...\Phi(\mathbf{y}_{Q_s})],\ \mathbf{y}_i \in \mathbf{Y_S} \tag{5.10}$$

and

$$\Phi_{\mathbf{Y_A}} = [\Phi(\mathbf{y}_1)\Phi(\mathbf{y}_2)...\Phi(\mathbf{y}_{Q_a})],\ \mathbf{y}_i \in \mathbf{Y_A} \tag{5.11}$$

Here $\mathbf{Y_A}$ and $\mathbf{Y_S}$ represent the target signal and background training sets, respectively. The column vectors of matrices $\mathcal{S}$ and $\mathcal{A}$ represent only the significant normalized eigenvectors $(\varsigma_1, \varsigma_2, ..., \varsigma_{Q_s},)$ and $(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_{Q_a},)$ of the background centered kernel matrix $\mathbf{K}(\mathbf{Y_S}, \mathbf{Y_S}) = \{K_{i,j} = k(\mathbf{y}_i, \mathbf{y}_j), \mathbf{y}_i, \mathbf{y}_j \in \mathbf{Y_S}\}$ and the target centered kernel matrix $\mathbf{K}(\mathbf{Y_A}, \mathbf{Y_A}) = \{K_{i,j} = k(\mathbf{y}_i, \mathbf{y}_j), \mathbf{y}_i, \mathbf{y}_j \in \mathbf{Y_A}\}$, respectively. Now, using (5.8), the projection of $\Phi(\mathbf{y})$ onto $\mathbf{S}_\Phi$ becomes:

$$\mathbf{S}_\Phi^T \Phi(\mathbf{y}) = [\mathbf{e}_s^1 \mathbf{e}_s^2 ... \mathbf{e}_s^{Q_s}]^T \Phi(\mathbf{y}) = \begin{bmatrix} \varsigma_1^T \Phi_{\mathbf{Y_S}}^T \Phi(\mathbf{y}) \\ \varsigma_2^T \Phi_{\mathbf{Y_S}}^T \Phi(\mathbf{y}) \\ .. \\ \varsigma_{Q_s}^T \Phi_{\mathbf{Y_S}}^T \Phi(\mathbf{y}) \end{bmatrix} = \mathcal{S}^T \mathbf{K}(\mathbf{Y_S}, \mathbf{y}) \tag{5.12}$$

Using a similar approach, we can express the projection onto $\mathbf{A}_\Phi$ like so

$$\mathbf{A}_\Phi^T \Phi(\mathbf{y}) = [\mathbf{e}_a^1 \mathbf{e}_a^2 ... \mathbf{e}_a^{Q_a}]^T \Phi(\mathbf{y}) = \begin{bmatrix} \boldsymbol{\alpha}_1^T \Phi_{\mathbf{Y_A}}^T \Phi(\mathbf{y}) \\ \boldsymbol{\alpha}_2^T \Phi_{\mathbf{Y_A}}^T \Phi(\mathbf{y}) \\ .. \\ \boldsymbol{\alpha}_{Q_a}^T \Phi_{\mathbf{Y_A}}^T \Phi(\mathbf{y}) \end{bmatrix} = \mathcal{A}^T \mathbf{K}(\mathbf{Y_A}, \mathbf{y}) \tag{5.13}$$

The other two projections required to compute the likelihood function in (5.2) are formed in a similar manner resulting in the kernelized likelihood ratio $J_{\Phi K}(\mathbf{y})$ for the linear signal model in $\mathcal{F}$ i.e.

$$J_{\Phi K}(\mathbf{y}) = \frac{\mathbf{K}(\mathbf{Y_{AS}}, \mathbf{y})^T \Delta\Delta^T \mathbf{K}(\mathbf{Y_{AS}}, \mathbf{y}) - \mathbf{K}(\mathbf{Y_S}, \mathbf{y})^T \mathcal{S}\mathcal{S}^T \mathbf{K}(\mathbf{Y_S}, \mathbf{y})}{\mathbf{K}(\mathbf{Y_{AS}}, \mathbf{y})^T \Delta\Delta^T \mathbf{K}(\mathbf{Y_{AS}}, \mathbf{y}) - \big[\mathbf{K}(\mathbf{Y_A}, \mathbf{y})^T \mathcal{H}\ \mathbf{K}(\mathbf{Y_S}, \mathbf{y})^T \mathcal{S}\big] \Lambda^{-1} \begin{bmatrix} \mathcal{A}^T \mathbf{K}(\mathbf{Y_A}, \mathbf{y}) \\ \mathcal{S}^T \mathbf{K}(\mathbf{Y_S}, \mathbf{y}) \end{bmatrix}} \tag{5.14}$$

where

$$\Lambda = \begin{bmatrix} \mathcal{A}^T\mathbf{K}(\mathbf{Y_A}, \mathbf{Y_A})\mathcal{A} \; \mathcal{A}^T\mathbf{K}(\mathbf{Y_A}, \mathbf{Y_S})\mathcal{S} \\ \mathcal{S}^T\mathbf{K}(\mathbf{Y_S}, \mathbf{Y_A})\mathcal{A} \; \mathcal{S}^T\mathbf{K}(\mathbf{Y_S}, \mathbf{Y_S})\mathcal{S} \end{bmatrix} \tag{5.15}$$

and Δ is a matrix whose columns are the normalized (by the square root of their respective eigenvalues) eigenvectors of the combined background and target kernel matrix $\mathbf{K}(\mathbf{Y_{AS}}, \mathbf{Y_{AS}})$, with $\mathbf{K}(\mathbf{Y_{AS}}, \mathbf{Y_{AS}}) = (\mathbf{K})_{i,j} = k(\mathbf{y}_i, \mathbf{y}_j), \mathbf{y}_i, \mathbf{y}_j \in \mathbf{Y_S} \bigcup \mathbf{Y_A}$. For our $M-$ary hypothesis test for a linear signal model in space $\mathcal{F}$, the kernel MSC amounts to the following test.

$$\begin{aligned} m^* &= \underset{m=1,\ldots,M}{\operatorname{argmin}} \{J_m(\mathbf{y})\} \\ &= \underset{m=1,\ldots,M}{\operatorname{argmin}} \{\Phi(\mathbf{y})^T(\mathbf{P}_{\mathbf{I}_\Phi} - \mathbf{P}_{\mathbf{H}_{m\Phi}})\Phi(\mathbf{y})\} \\ &= \underset{m=1,\ldots,M}{\operatorname{argmin}} \{\mathbf{K}(\mathbf{Y}_{tot}, \mathbf{y})^T\Delta\Delta^T\mathbf{K}(\mathbf{Y}_{tot}, \mathbf{y}) - \mathbf{K}(\mathbf{Y}_m, \mathbf{y})^T\mathcal{H}_m\mathcal{H}_m^T\mathbf{K}(\mathbf{Y}_m, \mathbf{y})\} \\ &= \underset{m=1,\ldots,M}{\operatorname{argmin}} \{\mathbf{K}(\mathbf{Y}_{tot}, \mathbf{y})^T\Delta\Delta^T\mathbf{K}(\mathbf{Y}_{tot}, \mathbf{y}) \\ &- \left[\mathbf{K}(\mathbf{Y}_m, \mathbf{y})^T\mathcal{A}_m \; \mathbf{K}(\mathbf{Y_S}, \mathbf{y})^T\mathcal{S}\right]\Lambda_m^{-1} \begin{bmatrix} \mathcal{A}_m^T\mathbf{K}(\mathbf{Y}_m, \mathbf{y}) \\ \mathcal{B}^T\mathbf{K}(\mathbf{Y_S}, \mathbf{y}) \end{bmatrix} \} \end{aligned} \tag{5.16}$$

with

$$\Lambda_m = \begin{bmatrix} \mathcal{A}_m^T\mathbf{K}(\mathbf{Y}_m, \mathbf{Y}_m)\mathcal{A}_m & \mathcal{A}_m^T\mathbf{K}(\mathbf{Y}_m, \mathbf{Y_S})\mathcal{S} \\ \mathcal{S}^T\mathbf{K}(\mathbf{Y_S}, \mathbf{Y}_m)\mathcal{A}_m & \mathcal{S}^T\mathbf{K}(\mathbf{Y_S}, \mathbf{Y_S})\mathcal{S} \end{bmatrix} \tag{5.17}$$

Here $\mathbf{Y}_{tot}$ is the set of all training samples from all classes and background, Δ is a matrix whose columns are the normalized non-zero eigenvectors of the kernel matrix $\mathbf{K}(\mathbf{Y}_{tot}, \mathbf{Y}_{tot})$, and $\mathcal{A}_m, \mathcal{S}$ are the significant normalized eigenvectors of class m target and background kernel matrices $\mathbf{K}(\mathbf{Y}_{\mathbf{A}_m}, \mathbf{Y}_{\mathbf{A}_m})$ and $\mathbf{K}(\mathbf{Y_S}, \mathbf{Y_S})$, respectively.

5.3 Classification Results of Kernel MSC

In this section we present the results of the kernel MSC classifier on real sonar datasets described in Chapter 2. The datasets included in training and testing datasets are the same as those described in Chapter 4. However, training samples underwent an additional procedure of k-means clustering in order to reduce the number of training samples in $\mathbf{Y}_{\mathbf{A}_m}, \mathbf{Y}_\mathbf{S}$ used in the kernel PCA process for forming subspace projections in the mapped feature domain. Training samples for the kernel MSC classifiers were chosen from those objects previously described in Table 2.1 which correspond to the appropriate ranges featured in a given test dataset. For PondEX testing, trained and tested features were truncated to $N_{Pond} = 295$ frequency bins corresponding to the $1 - 30$kHz freqeuncy response bins. For TREX testing, trained and tested features were truncated to $N_{TREX} = 275$ frequency bins corresponding to the $3 - 30$kHz freqeuncy response bins. For testing of the KMSC system, the Radial Basis Function (RBF) $k(\mathbf{y}_1, \mathbf{y}_2) = \exp(-\mathbf{y}_1^T\mathbf{y}_2/\sigma_\Phi)$ was utilized with a spread parameter of $\sigma_\Phi = 0.1$.

5.3.1 PondEX Testing Results

To make a classification decision for given sonar observations, the corresponding AC data vector of dimension $N_{Pond} = 295$ that contains the spectral features of an underwater UXO or non-UXO object at a particular aspect is applied to the kernel MSC classifier. This decision-making is implemented using the classification rule given in (5.16). The same process is carried out for all object aspects in a run in the PondEX09 and PondEX10 testing dataset. As before, multi-aspect decision-making was utilized to provide a better opportunity to discriminate between UXO and non-UXO object classes. Here, we used three aspects of an object to perform classification decision. As in the previous experiments, to account for platform instability in a realistic data collection scenario aspect separation is modeled by a uniformly distributed random variable $s \sim unif\ \{8, 16\}$ which shuffles data, beginning with the first aspect in a given run that meets the threshold.

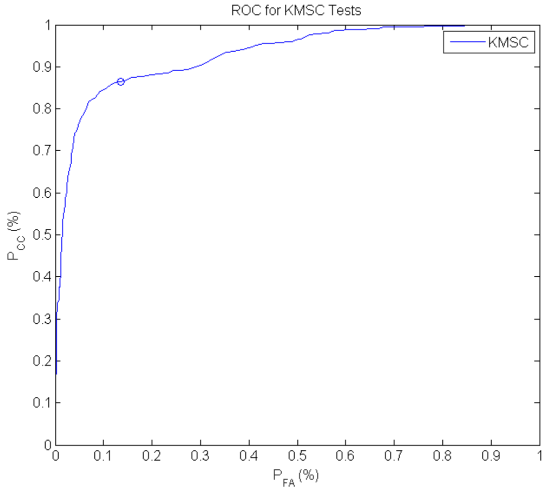


Figure 5.1: ROC for KMSC on Pond data (3 Asp./dec.)

Table 5.1: KMSC Confusion Matrix PondEX

Truth\Dec.	‘UXO’	‘non-UXO’
‘UXO’	0.8803	0.1197
‘non-UXO’	0.1149	0.8851

As can be observed by the circle in Figure 5.1, the ROC curves for the kernel MSC exhibits knee-point $P_{CC,KMSC} \approx 88.0\%$ and $P_{FA,KMSC} \approx 12.0\%$. These results show that the kernel MSC performs rather poorly in discriminating ‘UXO’ vs. ‘non-UXO’ in the PondEX dataset when compared to that of the results in Chapter 4 for linear MSC. The most common mis-classified ‘UXO’ type was again stainless steel UXO which was incorrectly labeled as aluminum pipe and cylinder about equally as often.

5.3.2 TREX Testing Results

A similar procedure was applied to the TREX13 dataset as the PondEX data, the difference being that for these experiments there were objects of varying ranges present and as a results, training datasets were chosen to include these variations. Training data was similar to that utilized in the linear MSC testing, for the all range test, Experiment 1, the training dataset featured FRM models of all 10 objects in Table 2.1 for ranges $\{10, 15, 20, 25, 30, 35, 40\}$ m. For the short range test, Experiment 2, all 10 objects were again used in training however only samples from ranges $\{10, 15, 20, 25\}$ m were used. Lastly the training set for the long range test, Experiment 3, featured all 10 FRM objects at ranges

$\{30, 35, 40\}$ m. To prepare the data for the kernel MSC classifier, the filtering method described in Section 2.3.3 was again used to isolate the object response from those of the adjacent objects and to eliminate possible overlap in the sonar backscattered from the objects. As was utilized before, AC features were truncated to $N_{TREX} = 275$ corresponding to the $3 - 30$ kHz frequency bins of the TREX chirp.

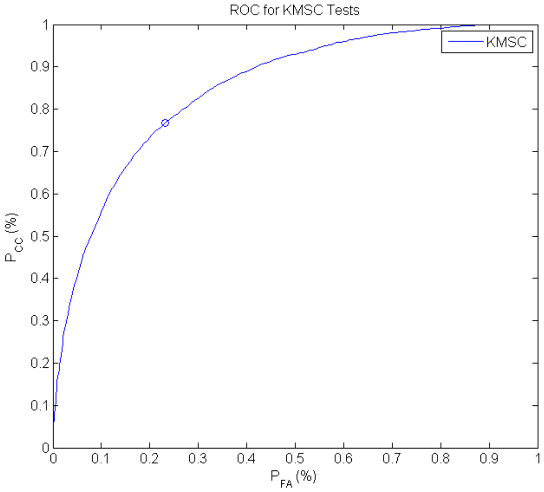


Figure 5.2: KMSC ROC TREX Ex. 1.

Table 5.2: KMSC Confusion Matrix TREX Ex. 1

Truth\Dec.	'UXO'	'non-UXO'
'UXO'	0.7672	0.2328
'non-UXO'	0.2315	0.7685

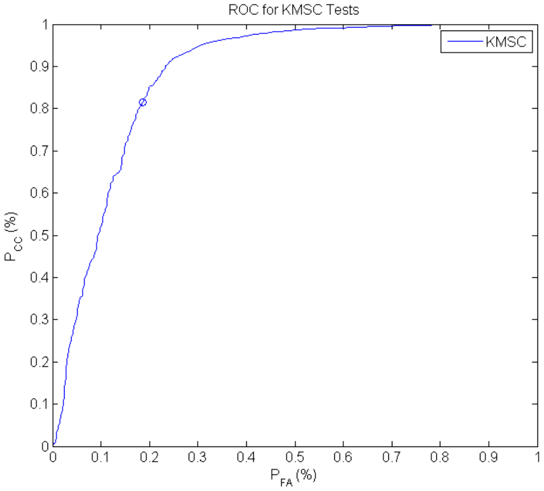


Figure 5.3: KMSC ROC TREX Ex. 2.

Table 5.3: KMSC Confusion Matrix TREX Ex. 2

Truth\Dec.	'UXO'	'non-UXO'
'UXO'	0.8146	0.1854
'non-UXO'	0.1862	0.8138

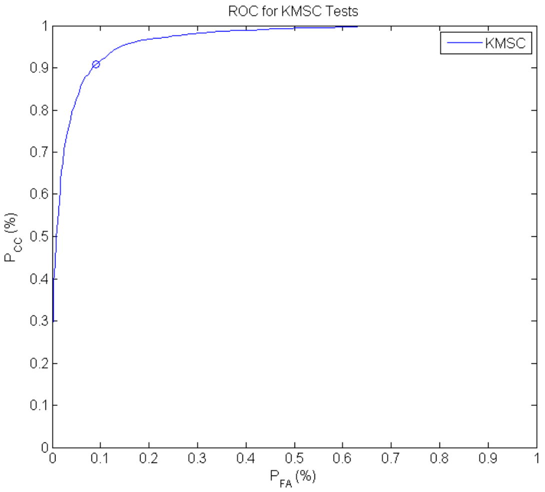


Figure 5.4: KMSC ROC TREX Ex. 3.

Table 5.4: KMSC Confusion Matrix TREX Ex. 3

Truth\Dec.	'UXO'	'non-UXO'
'UXO'	0.9080	0.0920
'non-UXO'	0.0909	0.9091

The ROC curves of the kernel MSC for TREX13 dataset is presented in Figures 5.2 - 5.4 for Experiments 1-3, respectively. Tables 5.2 - 5.4 also display the corresponding confusion matrices for these experiments. These results show that the kernel MSC performed quite poorly when compared with the linear MSC with knee-point probability of correct classification of only $P_{CC} = 76.7\%$ for all ranges (Experiment 1). For Experiments 2 and 3, the kernel MSC gave knee-point $P_{CC} = 81.5\%,$ and $P_{CC} = 90.8\%$ respectively. These results demonstrated inferior performance to the linear MSCs tested in Chapter 4 for all experiments.

Further investigation revealed that the kernel MSC struggled at correctly classifying the stainless steel UXO. However, as with the results in Chapter 4 for linear MSC, the best overall performance on the TREX13 dataset was obtained for long range cases. Figures 5.2 -5.3 indeed attest to this fact. In the short range experiment, similar to the observations in Chapter 4, the most common mis-classified 'UXO' type (Type I error) was the stainless steel UXO, commonly being identified at a cylinder or pipe. As with the PondEX dataset, the most common Type II error was from aluminum pipes being incorrectly labeled as Aluminum or Stainless Steel UXOs. As evident from both the long and short range testing results, the kernel MSC, in its current state (i.e. utilizing k-means for learning subspace samples), is not capable of generalizing well between different ranges of the same class. Moreover, the training of this system is very sensitive to the choice of model-generated data and kernel function spread parameter σ_Φ. A well selected parameter σ_Φ can yield excellent results for samples of any *fixed* range, but it was found that learning a single range very well necessarily meant some overfitting had occurred and thus resulted in poor generalization across multiple ranges.

5.4 Conclusion

In this chapter we presented a non-linear form of the MSC framework using the kernel trick which allows for hypothesis testing to utilize higher dimensional features. We began by presenting the GLRT form of this kernel detector as in [20], and then proceeded to kernelize it using the well-established results of kernel PCA. This kernelized detector was then converted to an M-ary classifier which we refer to as the kernel MSC. This kernelized MSC was then applied to the PondEX and TREX13 datasets and performance of the system was benchmarked against that of the linear MSC in Chapter 4. It was found that, comparing to the results of Chapter 4, the kernel MSC did not provide better results for either dataset.

These poor results of the kernel MSC classifier may be due to oversimplifications in the signal model that assumes Gaussianity in the infinite dimensional feature space which is certainly inappropriate. Indeed, the error between observations and the signal model is likely some sort of structured noise dependent on factors such as grazing angle, range, water turbulence, and density. Furthermore, the successful usage of a non-linear classifier depends largely on the choice of parameters used in a given kernel function. As is the case with many kernel-based approaches, parameters could be found which would fine tune the classifier to a specific classification task (e.g. UXO vs. non-UXO for only long range TREX samples) but it was found that these parameters would not generalize across all testing data when using only synthetic training data. As a result, it makes sense that the linear classifiers would generalize more appropriately because they find the best average boundary in the original data domain rather than attempting to separate the samples with a non-linear curve that may or may not describe the true class separation boundary but can be fine tuned to include non-linearly separable, outlier, data points in a given class of samples. Indeed, the flexibility of the kernel framework is both a blessing and a curse as it provides the tools for creation of arbitrary class boundaries but as a result can learn the most misleading portions of training data far too well, resulting in poor generalization performance.

These results also indicated that, while it is tempting to cast classification problems into the typical non-linear frameworks with expectations to achieve better results, it may be more tactful to rely on sparse representation methods which can provide better ability to discriminate different object classes. Obviously, further work is needed to format the signals into a range invariant form which allows for better generalization performance in classification. Indeed, the proposed kernel MSC appears to be somewhat flawed in its formulation and more careful consideration of the derivation and its intrinsic assumptions must be pursued.

CHAPTER 6

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

6.1 Conclusions and Discussions

The main objective of the work in this thesis was the development and testing of a MSC-based UXO vs. non-UXO classifier. In particular, the hypothesis that was tested involved building an MSC classifier exclusively using model-generated AC data sets while evaluating it using real PondEX and TREX13 low frequency sonar datasets. The motivation was that one cannot easily collect abundance of data for different UXO shapes and sizes in realistic settings. The scattering model that was developed allowed for monostatic sonar data sets to be simulated via a fast ray model (FRM). This large dataset was then used to build an over-complete dictionary for creation of signal subspaces used in MSC via the K-SVD and LP K-SVD signal-dependent dictionary learning methods. Once MSC was trained based on this model-generated data it was subsequently tested on real sonar datasets collected in PondEX and TREX13 exercises. The classification results revealed promising performance in UXO versus non-UXO discrimination when compared to the existing results on similar but arguably less challenging datasets [23]. For the K-SVD and LP K-SVD MSC systems, performance on PondEX was reported to be $P_{CC,KSVD} \approx 90.07\%$, $P_{CC,LP-KSVD} \approx 93.20\%$, respectively. For the more challenging TREX13 datasets, the K-SVD and LP K-SVD systems performed with $P_{CC,KSVD} \approx 80\%$, $P_{CC,LP-KSVD} \approx 83.0\%$ for a combined all range test. These results were found to be substantially better than those of the kernelized MSC in Chapter 5. The poor performance of the kernel MSC is mostly attributed to kernel parameter choice and over-fitting on synthetic data samples which led to poor generalization. What was particularly notable about the results presented is the utility of the synthetic FRM-generated data in representing real sonar data for classification purposes has been confirmed.

6.2 Future Work

The need to accomplish several new tasks has been revealed by this work. This research has suggested a path towards more consistent ‘UXO identification that is invariant to target range, aspect, and other important observation conditions for sonar systems. In future endeavors, we hope to accomplish the following two major tasks and several smaller tasks related to the problem of underwater UXO classification. These endeavors include incremental training of systems and multi-aspect fusion strategies.

Incremental Training for Better Robustness: Our results presented in this thesis clearly indicated the promise and effectiveness of the developed methods for classification of munitions from sonar data. However, it is obvious that the spectral features in the AC data for a specific target vary significantly depending on the object’s burial condition, sea floor properties and roughness, actual orientation of the object, range and grazing angles with respect to the sonar, etc. Although, it is unrealistic to expect model data will capture all such variations for target characterization, it can provide us with clues on how to augment the training datasets using perhaps a limited training samples from sonar returns of actual objects to improve the robustness in different environmental conditions. A preliminary study was carried out to show the importance of this data augmentation beyond the FRM-generated data and incremental training using real data for improving classification performance. Several PondEX object samples (only 2 % of training from real PondEX test data) that correspond to the same objects in the TREX13 dataset were added to the model-generated training set to form the dictionary training set and build the MSC classifier. The trained system was then tested on TREX 10m run datasets. It was found that the addition of real Pond samples from the relevant objects improved the overall results noticeably. Clearly, we didn't attempt to optimize this data selection process based upon the information content of the selected sample for classification purposes. Additionally, we had to retrain (in batch) all the dictionaries and consequently the MSC classifier, which is not an efficient way of doing this training. Future work should address these needs more formally.

Multi-Aspect Classification Fusion: In a real operating environment, the decision about the presence and type of an object is typically made based upon the observation of the properties of the sonar returns at several aspect angles. This is due to the fact that the multi-aspect processing yields substantial improvements in performance, resolution, and sensing of the properties of the object in a non-isotropic environment. Inspired by these desired properties, two different general frameworks were developed in [2, 5, 12, 43]. The framework in [2, 43] is based upon the idea of multi-aspect feature extraction that uses the two-channel canonical coordinate decomposition method [11] to extract robust features with maximum coherence (or mutual information) from pairs of sonar pings with certain separation. The idea is that the coherence pattern extracted from the UXO objects differ from those of the non-UXO objects, hence aiding the overall classification process. The theme of the other framework is multi-aspect classification using either (a) a decision-level multi-aspect fusion [44], which linearly or non-linearly combines the individual classification decisions, generated at several aspects, or (b) a feature-level multi-aspect fusion [12] using hidden Markov model (HMM) to generate one decision based upon observing a sequence of AC feature vectors at various aspects with certain separations, or (c) a collaborative decision-making process [43], which uses a combination of the feature-level and decision-level fusion methods. It would be prudent to study and test these different methods and compare their results with the multi-aspect MSC classifier in (4.4).

Other important areas and extensions that can be pursued in the future include, but are not limited to:

- Use real and synthesized sonar data to determine the effectiveness of the developed adaptive dictionary learning and sparse representation methods in different situations. One should investigate how the model and dictionaries can be designed to guarantee robustness to partial burial or occlusion of the targets, and separation of similar class types.

- The developed kernel MSC should be more thoroughly studied to overcome some of the theoretical shortcomings of the derivation in [20].

- The PC-SWAT sonar data generation program [45] developed by Naval System Warfare Center (NSWC), Panama City, FL should be used to generate UXO and non-UXO datasets and the classification results should be compared to those obtained from the FRM-generated datasets presented in this thesis.

- Additional and more challenging real datasets can be used to truly test the overall system for classification of buried or partially buried objects and should involve testing of the Buried Object Scanning Sonar (BOSS) datasets [46]. These datasets include sonar returns from various UXO, mine-like and non-target as well as plenty of difficult bottom clutter.

With these objectives in place, a clearer separation of 'UXO' and 'non-UXO' objects will be achieved by uncovering the relationships between range/grazing angle and the frequency response of objects of interest, and identifying a reliable means for mitigating these effects.

REFERENCES

[1] A. Pezeshki, M. Azimi-Sadjadi, and L. Scharf, “Undersea target classification using canonical correlation analysis,” *IEEE Journal of Oceanic Engineering*, vol. 32, pp. 948 – 955, October 2007.

[2] N. Wachowski and M. Azimi-Sadjadi, “A new synthetic aperture sonar processing method using coherence analysis,” *IEEE Journal of Oceanic Engineering*, vol. 34, pp. 665–678, October 2011.

[3] J. Cartmill, M. Azimi-Sadjadi, and N. Wachowski, “Buried underwater object classification using a collaborative multi-aspect classifier,” *IEEE Journal of Oceanic Engineering*, vol. 34, pp. 32–44, January 2009.

[4] R. Lim, “Data and processing tools for sonar classification of underwater UXO,” *Final Report, SERDP Project MR-2230*, pp. 1–64, August 2015.

[5] M. Azimi-Sadjadi, Q. Yao, D. Huang, and G. Dobeck, “Underwater target classification using wavelet packets and neural networks,” *IEEE Transactions on Neural Networks*, vol. 11, pp. 784–794, May 2000.

[6] G. J. Dobeck, J. Hyland, and L. Smedley, “Automated detection/classification of sea mines in sonar imagery,” *Proc. SPIE*, vol. 3079, pp. 90–110, April 1997.

[7] S. Kargl, A. España, 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,” *IEEE Journal of Oceanic Engineering*, vol. 40, no. 3, pp. 632–642, August 2015.

[8] S. Kargl, K. Williams, and E. Thorsos, “Synthetic aperture sonar imaging of simple finite targets,” *IEEE Journal of Oceanic Engineering*, vol. 37, pp. 516–532, July 2012.

[9] S. Kargl, K. Williams, T. Marston, and J. Kennedy, “Acoustic response of unexploded ordnance (UXO) and cylindrical targets,” *Proceedings of MTS/IEEE Oceans Conference 2010*, pp. 1–5, September 2010.

[10] R. Lim, “Sonar detection and classification of underwater UXO and environmental parameters,” *Interim Report, SERDP Project MR-1666*, pp. 1–19, October 2010.

[11] L. Scharf and B. Friedlander, “Matched subspace detectors,” *IEEE Transactions on Signal Processing*, vol. 42, pp. 2146–2157, August 1994.

[12] M. Robinson, M. Azimi-Sadjadi, and J. Salazar, “Multi-aspect classification using hidden markov models and neural networks,” *IEEE Transactions on Neural Networks*, vol. 16, pp. 447–459, March 2005.

[13] S. Schock, A. Tellier, J. Wulf, J. Sara, and M. Ericksen, “Buried object scanning sonar,” *IEEE Journal of Oceanic Engineering*, vol. 26, pp. 677–689, Oct 2001.

[14] S. Mallat, *A wavelet tour of signal processing.* Academic press, 1999.

[15] S. S. Haykin, S. S. Haykin, S. S. Haykin, and S. S. Haykin, *Neural networks and learning machines*, 3rd ed. Pearson Education Upper Saddle River, 2009.

[16] M. Azimi-Sadjadi, D. Yao, A. Jamshidi, and G. Dobeck, “Underwater target classification in changing environments using an adaptive feature mapping,” *IEEE Transactions on Neural Networks*, vol. 13, pp. 1099–1111, Sept 2002.

[17] A. Salberg, A. Hanssen, and L. Scharf, “Robust multidimensional matched subspace classifiers based on weighted least-squares,” *IEEE Transactions Signal Processing*, vol. 55, pp. 873–880, March 2007.

[18] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on Signal Processing*, vol. 54, pp. 4311–4322, November 2006.

[19] Y. Zhou, J. Gao, and K. Barner, "Locality preserving KSVD for nonlinear manifold learning," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3372–3376, May 2013.

[20] H. Kwon and N. M. Nasrabadi, "Matched subspace detectors for hyperspectral target detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 178–194, 2006.

[21] K. Williams, S. Kargl, E. Thorsos, D. Burnett, J. Lopes, M. Zampolli, and P. Marston, "Acoustic scattering from a solid aluminum cylinder in contact with a sand sediment: Measurements, modeling, and interpretation," *J. Acoust. Soc. Am.*, vol. 127, pp. 3356–3371, 2010.

[22] J. Yang, D. Tang, B. Hefner, K. Williams, and J. Preston, "Overview of reverberation measurements in target and reverberation experiment 2013," *TREX13 Workshop*, pp. 784–794, November 2014.

[23] 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.

[24] A. L. España, K. L. Williams, D. Plotnick, and P. Marston, "Acoustic scattering from a water-filled cylindrical shell: Measurements, modeling, and interpretation," *Acoustical Society of America*, vol. 136, pp. 109–121, July 2014.

[25] K. Williams, 2015, private communication on TREX13 Experimental Setup.

[26] T. Marston, P. Marston, and K. Williams, "Scattering resonances, filtering with reversible sas processing, and applications of quantitative ray theory," *Proc. 2010 Oceans MTS/IEEE Seattle*, pp. 1–9, September 2010.

[27] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *Information Theory, IEEE Transactions on*, vol. 49, no. 12, pp. 3320–3325, 2003.

[28] M. Azimi-Sadjadi, J. Kopacz, and N. Klausner, "K-SVD dictionary learning using a fast OMP with applications," *Proc. of IEEE Intern. Conference on Image Processing (ICIP)*, pp. 1599–1603, October 2014.

[29] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis.* Addison-Wesley, 1991.

[30] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[31] J. M. Cioffi and T. Kailath, "Fast, recursive-least-squares transversal filters for adaptive filtering," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 32, pp. 304–337, 1984.

[32] S. Alexander, *Adaptive Signal Processing: Theory and Applications.* Springer-Verlag, 1986.

[33] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[34] H. L. Van Trees, *Detection, Estimation, and Modulation Theory Part I.* John Wiley and Sons, 1968.

[35] R. O. Duda, P. E. Hart, and D. G. Sotrk, *Pattern Classification.* John Wiley and Sons, 2001.

[36] S. Stanic and E. Kennedy, "Reverberation fluctuation from a smooth seafloor," *IEEE Journal of Oceanic Engr.*, vol. 18, pp. 95–99, 1993.

[37] S. Stanic, R. Goodman, K. Briggs, N. P. Chotiros, and E. Kennedy, "Shallow-water bottom reverberation measurements," *IEEE Journal of Oceanic Engr.*, vol. 23, pp. 203–210, 1998.

[38] J. T. Cobb, K. C. Slatton, and G. J. Dobeck, "A parametric model for characterizing seabed textures in synthetic aperture sonar images," *To Appear in IEEE Journal of Oceanic Engineering*, 2010.

[39] N. Klausner and M. R. Azimi-Sadjadi, "Non-Gaussian target detection in sonar imagery using the multivariate Laplace distribution," *Oceanic Engineering, IEEE Journal of*, vol. 40, pp. 452–464, 2015.

[40] Z. Zhang, "Parameter estimation techniques: A tutorial with applications to conic fitting," *Image Vision Computing*, vol. 15, p. 5976, 1997.

[41] B. Schölkopf and A. J. Somola, *Learning with Kernels.* The MIT press, 2002.

[42] D. S. Kargl, "Acoustic response of underwater munitions near a sediment interface: Measurement-model comparisons and classification schemes," *Final Report, SERDP Project MR-2231*, pp. 1–104, April 2015.

[43] J. Cartmill, M. R. Azimi-Sadjadi, and N. Wachowski, "Buried underwater object classification using a collaborative multi-aspect classifier," *IJCNN 2007*, pp. 1807–1812, Aug. 2007.

[44] J. C. Hyland and G. J. Dobeck, "Sea mine detection and classification using side-looking sonar," *Proc. SPIE*, vol. 2496, pp. 442–453, April 1995.

[45] G. S. Sammelmann, "PC SWAT 7.0: Users manual," *CSS Draft Report, Coastal Systems Station Code*, vol. 22.

[46] S. Schock and J. Wulf, "Buried object scanning sonar for AUV's," *Proceedings of MTS/IEEE Oceans 2003*, pp. 494–499, Sept 2003.