# Statistical Modeling to Improve Buried Target Detection with a Forward-Looking Ground-Penetrating Radar

by

Joseph A. Camilo

Department of Electrical and Computer Engineering
Duke University

Date:_____

Approved:

_____
Leslie M. Collins, Supervisor

_____
Henri P. Gavin

_____
Loren W. Nolte

_____
Galen Reeves

_____
Stacy L. Tantum

Dissertation submitted in partial fulfillment of
the requirements for the degree of Doctor
of Philosophy in the Department of
Electrical and Computer Engineering in the Graduate School
of Duke University

2017

ABSTRACT

Statistical Modeling to Improve Buried Target Detection with a Forward-Looking Ground-Penetrating Radar

by

Joseph A. Camilo

Department of Electrical and Computer Engineering
Duke University

Date:_____

Approved:

_____
Leslie M. Collins, Supervisor

_____
Henri P. Gavin

_____
Loren W. Nolte

_____
Galen Reeves

_____
Stacy L. Tantum

An abstract of a dissertation submitted in partial fulfillment of
the requirements for the degree of Doctor
of Philosophy in the Department of
Electrical and Computer Engineering in the Graduate School
of Duke University

2017

## Abstract

Forward-looking ground-penetrating radar (FLGPR) has recently been investigated as a remote sensing modality for buried target detection (e.g., landmines and improvised explosive devices (IEDs) ). In this context, raw FLGPR data is commonly beamformed into images and then computerized algorithms are applied to automatically detect subsurface buried targets. Most existing algorithms are supervised, meaning they are trained to discriminate between labeled target and non-target imagery, usually based on features extracted from the radar imagery. This thesis is composed of two FLGPR research areas: an analysis of image features for classification, and the application of machine learning techniques to the formation process of radar imagery.

A large number of image features and classifiers have been proposed for detecting landmines in the FLGPR imagery, but it has been unclear which were the most effective. The primary goal of this component of my research is to provide a comprehensive comparison of detection performance using existing features on a large collection of FLGPR data. Fusion of the decisions resulting from processing each feature is also considered. These comparisons have not previously been performed, and a novel 2DFFT feature was also developed for the FLGPR application. Another contribution of my research in the image feature investigation was the analysis of two modern *feature learning* approaches from the object recognition literature: the bag-of-visual-words and the Fisher vector for FLGPR processing. The results indicate that most image classification

algorithms perform similarly, though the newly designed 2DFFT-based feature consistently performs best for landmine detection with the FLGPR.

Based on the image feature results presented in this work, it appears that the current feature extractors are leveraging most of the information available in the radar images that are produced by the conventional beamforming process. The work presented in the second component of this thesis improves the beamforming process applied to the radar responses. By improving the radar images (i.e., increasing signal to noise ratio, or SNR), each feature extractor and classification algorithm is shown to subsequently increase in performance. These new methods are designed to incorporate multiple uncertainties in the physical world that are currently ignored during conventional beamforming. The two approaches to improving the underlying FLGPR image are a learned weighting applied to the antenna responses and a strategy for selecting the image creation depth. Both of these two new beamforming process approaches yield additional improvements to the imagery which are reflected in improved detection results.

# Contents

# List of Tables

# List of Figures

## Acknowledgements

The first person I have to thank for helping me with my PhD is my Clarkson University advisor Dr. Remus (known as J.J. to some). He not only directed me toward Grad school, he helped me find the Duke program and connected me with the SSPACISS/AMLL/Collins lab. When talking about Duke, I remember he asked if I liked basketball, and at that moment I became aware Duke had a basketball team. And little did I know how much the basketball culture would be a (helpful) distraction throughout my time here.

This of course led to the "snow-storm" Duke visit, where I was just happy to escape the -10 degree tundra of the New York North Country. This still somehow convinced me to move down to Durham NC in the fall. At the visit I met Kenny for the first time. And, thank you Kenny for getting me up and started in those first few years, and also showing me just how bad at MATLAB I truly was. Though, like Luke and his first mentor Obiwan Kenobi, Kenny was soon "cut-down" and joined "the force" known as CoVar.

Step in Jordan, the senior grad student soon to be research professor. Like Yoda riding the back of Luke skywalker in the degobah system, Jordan trained me in the sorcery of machine learning. I remember going into my master's defense a few months into working together, and asking him nervous and right before starting my presentation, "do you think this work is good?" He said "Yes". At that was what I needed to hear right then.

Beyond his many machine learning insights and constant pushing me to do more, Jordan continued to be an important barometer of what good research was, and that led us through a variety of cool projects. (And yes Jordan, I'm still going to get to a "fully convolutional beamforming network" one of these days). So thank you Jordan for the way you shaped my path here; not only do you continue to be a great mentor, you've also been a fantastic friend.

Of course, while an advisor can do a lot to help. I have to acknowledge the many other awesome people who came in and out of the lab while I was here. First, thank you Dima. You were the first friend I made here and you integrated me into your own friend group. You're pulling me to intermetals and to Duke sports helped me make great friends and enjoy my time here much more. Also, thank you Daniel, you were a great resource to bounce ideas off of and talk about the many many ways of building software frameworks. You too have been a good friend to spend time outside the lab, and your tidbits about Judaism never cease to amaze. But really, it's thanks to all the members of the Collins lab, who have always helped with exam prep, provided constant lab meeting feedback, cooked delicious holiday party food, and always gave the opportunity for great research discussions or just a place to vent about that particular day; I really enjoyed joining such a large lab, and I know it made my time here much more enjoyable. Lastly, to the older grad students that I had the privilege of overlapping with, specifically Ken and Patrick who became great friends and at times a great source of motivation to keep going.

This of course brings me to Leslie, or some would say the coach-K of PhD advisors. If there was a US National team of signal processing PhDs I have no doubt she would be called to coach. In all seriousness, there were countless moments in lab-meetings where I would be presenting some idea or result I had, and she almost always had one of two things to say, "keep going", or "that's enough". And that way we would know when we could start wrapping up that specific thread. While that feedback was invaluable, I think my most memorable feedback from Leslie will be on my "casual" writing. Idk, maybe I write in slang because I always see the reader as my boi that I'm just trying to explain some science to. But, thank you Leslie for continuing to challenge that notion, especially with comments like "If I see this again my head is going to explode". Well I'm glad your head is still intact and I know I'm a better writer for it.

Outside the lab, I've formed many great friendships around Durham. The free-time spent in and around Durham has been awesome, and always provided a great setting for such a formative time in my life. Specifically, thank you to my Sunday flag football crew for providing just enough physical activity for my heart not to explode while climbing the CIEMAS stairs.

Last but definitely not least, I have to thank my family. A number of which were in the triangle area and made me feel at home with delicious Sunday dinners. My sister, who found plenty of time to come visit and hang out. And my parents, who, when starting on my graduate school journey had no hesitation about my choice and continue to support

me in my future endeavors.

# 1 Introduction

Landmines continue to pose a significant problem in many regions around the world. Not only do landmines cause injuries and deaths to individuals currently involved in conflicts, but they also have devastating effects on areas after conflicts have ended and landmines are left behind [1]. At least 58 countries around the world are contaminated with landmines, or other buried threats. Between 1999 and 2014, there were nearly one hundred thousand recorded causalities from landmines; and in 2014, 31% of these deaths or injuries were not from factory-made landmines, but instead caused by "mine-like" buried threats, sometimes referred to as improvised explosive devices (IEDs) [1]. These non-traditional buried threats are generally harder to detect, thus motivating new development efforts in detection technologies [2], [3].

Historically, metal detectors, or electromagnetic induction (EMI) sensors, have been used for landmine detection, however EMI sensors suffer from a high false alarm rate and furthermore, they cannot detect low- or non-metallic threats, of which there are many. Recently, a downward-looking ground-penetrating radar (GPR) system has demonstrated real world success in the clearance of landmines, primarily as a result of its ability to reliably detect a diverse set of buried threats [3]–[6]. This downward-looking GPR system has been able to overcome the large number of false-alarms typically produced by traditional metal detectors and has the capability of sensing non-metal buried objects [7]. For vehicle-mounted buried threat detection systems, the GPR system

typically consists of an array of antennas that are mounted on the front of a vehicle, and the antennas collect data continuously as the vehicle moves down a road or path. The GPR senses radar returns from discontinuities in the dielectric properties of the subsurface being inspected [7]. Metal, non-metal, and other types of buried threats produce such dielectric discontinuities and, by analyzing the characteristics of the returned radar signals, the goal is to discriminate the responses from buried threats from the radar responses generated from sub-surface clutter (e.g. roots, rocks, holes etc.) [3], [5].

Currently, the most effective GPR configuration, in terms of high detection rates and low false alarm rates is the downward-looking GPR [8]. In this configuration, the GPR antennas are pointed toward the ground at a very close distance. Though this configuration is very capable, it can only declare suspicious locations that are physically close to the vehicle, and this effectively limits the speed at which the system can travel forward because of the short stopping distance when a threat is detected [9]. Forward-looking ground-penetrating radar (FLGPR) systems have been investigated as an alternative to the downward-looking GPR [10]–[13], with the advantage of such systems being that they can potentially detect threats far from the vehicle, thus permitting greater rates of advance.

Similar to the downward-looking GPR, the FLGPR systems consist of an array of radar antenna transmitters and receivers mounted on the front of a vehicle, but in this configuration the antennas are focused at a location further in front of the system. A

diagram of a typical FLGPR system is presented in Figure 1. As the vehicle travels forward along a road or path, radar pulses are emitted towards the ground, and the receivers measure energy reflected from the surface and subsurface. The raw data is typically used in a beamforming process to synthesize 2-dimensional spatial images of the ground [14]–[16]. The FLGPR, being further away from the area under inspection than the downward-looking GPR, benefits from the beamforming process by averaging multiple illuminations of the same ground area. Pixel intensities in the resulting images can be viewed as a measure of the energy reflected from the ground at that location [8].

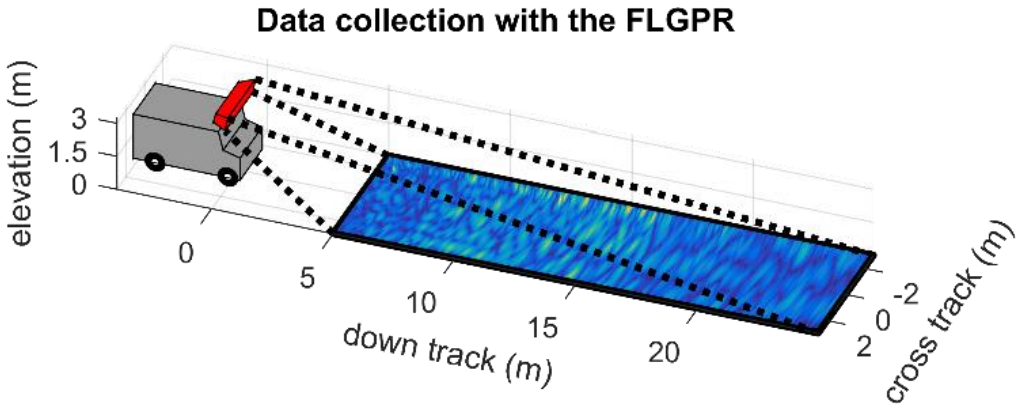



**Figure 1: A diagram of the FLGPR system.This system inspects the ground in front of the vehicle; the responses from the antenna array are then formed into images for detection. Cross track and down track labels are used to denote the axes perpendicular and parallel to the vehicle's direction of travel, respectively.**

Buried targets can be detected because they often reflect the radar energy differently than the surrounding soil and other subsurface clutter [17], and therefore appear differently from the background in the beamformed FLGPR imagery. Examples of beamformed FLGPR imagery over targets and non-targets, are shown in Figure 2.

Although the targets in Figure 2 are easily detectable, many targets exhibit much subtler patterns, and as such are confused with clutter or background. To aid in the detection of buried threats with the FLGPR, computer algorithms have been designed to automatically indicate the presence of targets in beamformed FLGPR imagery [8], [10], [13], [16], [18]–[23]. The work presented in this document focused on developing and analyzing algorithms that are used with the FLGPR data for improving detection of buried threats. This work is composed of two research areas: image feature analysis and comparison, and the application of machine learning techniques to the formation of FLGRP imagery.

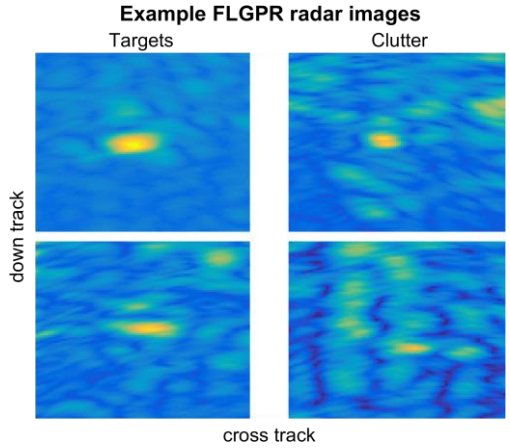



**Figure 2: Beamformed images at two different target locations (left column), and at two different clutter objects (right column).**

## 1.1 *FLGPR image feature investigation*

A large body of research has previously been conducted on the automatic detection of buried targets in FLGPR imagery [8], [10], [13], [16], [18]–[23]. Such algorithms usually employ supervised machine learning classifiers (e.g., a support vector machine

[24] or logistic regression [25]) to distinguish between target and non-target FLGPR imagery. For image recognition tasks such as the one considered here, classifiers typically operate on image features, statistics, or other measures that are computed based on the pixel intensities in the beamformed image. The performance of classifiers depends strongly on the features they are given, and as a result, a variety of features have been investigated for target detection with FLGPR data. Recent examples of features used for FLGPR processing include SIFT descriptors [21], the 2D FFT of the images [22], local image statistics [21], log-Gabor filtering statistics [23], and raw pixel intensities [13].

Although many features have been investigated, it is to date unclear which features are best for detection in the current FLGPR systems. A major complication in this interpretation of the literature for these systems is that the detection performance for each of the aforementioned features was examined under different conditions: using different data sets, classifiers, and experimental designs. These varying conditions can lead to large variations in reported system performance, and this makes it difficult to compare the relative merits of individual features and algorithms across different studies. Additionally, several important recent advances in feature extraction techniques that have been developed by the computer vision community have yet to be investigated with the FLGPR imagery. Specifically, *feature learning* has yet to be utilized in this context, even though it has been shown to be successful for many image problem domains [26]–[28].

To address the aforementioned limitations in the existing FLGPR image feature

research, this work makes two primary contributions. First, a comprehensive comparison of features that have previously been considered for supervised classification of FLGPR data is conducted on a large data set using several supervised classifiers. Second, two popular feature learning techniques are applied to FLGPR data: bag-of-visual-words (BOV), and Fisher vector (FV) [26]–[33]. These techniques may improve detection performance since they extract specific patterns learned from the data, leading to potentially improved data representation prior to classification. In addition to evaluating each feature's performance individually, decision-level fusion of features is also considered. Fusion among feature sets is investigated to leverage the independent information extracted by each feature and potentially achieve performance that exceeds that of any individual feature.

The results we have obtained to date indicate that the Fisher vector feature learning approach outperforms any individual feature set that has previously been considered, and further, that fusing the Fisher vector with other features yields an additional performance improvement. An analysis of the results provides insight about which image structures in FLGPR data are most indicative of the presence of a target. While there are improvements in performance with the feature learning techniques as well as through fusion, the feature learning analysis suggests that target responses mostly consist of simple structures. Further, it becomes apparent that more complicated feature extractors or classification algorithms are not likely to significantly increase performance

because the underlying images are not exhibiting any consistent sophisticated target signatures.

## 1.2 *Image creation investigation*

Based on results presented in this work, it appears that the current feature extractors are leveraging most of the information available in the radar images produced by the conventional beamforming process. The second set of work in this thesis improves the beamforming process applied to the radar responses. By improving the radar images (i.e. increasing signal to noise ratio, or SNR), each feature extractor and classification algorithm is shown to subsequently increase in performance. These new methods are designed to incorporate multiple uncertainties in the physical world that are currently ignored during conventional beamforming. The two approaches to improving the underlying FLGPR image are a learned weighting applied to the antenna responses and a strategy for selecting the image creation depth. Both approaches yield improvements to the imagery which are reflected in improved detection results.

The remainder of this document is organized as follows. Chapter 2 provides background information regarding the state of the art in detection of buried targets with the FLGPR. Chapter 3 contains the comparison of individual feature set performances. The results for fusion among feature sets are presented in Chapter 4. Also presented in Chapter 4 is an analysis of the feature set results, and the weaknesses in current FLGPR imagery classification systems are explored. Chapter 5 presents an improved 2DFFT

feature which results in the highest performing individual feature. The automatic antenna calibration methods and results are presented in Chapter 6. Chapters 7 and 8 investigate the 3D volume imaging and the automatic depth selection, respectively. Finally, Chapter 9 presents the conclusions from and summary of the work presented in this dissertation.

# 2 Background

In this chapter, a brief summary of the FLGPR system used in this work is presented along with an overview of the relevant research previously conducted in this area. First, this chapter outlines the standard detection process that is used in FLGPR buried threat detection. The standard processing description includes the current FLGPR image formation algorithm (i.e. beamforming), and the procedure for identifying suspicious locations in the sensed area. Second, a complete literature review of the feature sets (i.e., image descriptors) extracted over FLGPR imagery is presented. The features described here are those whose performance for classification is compared in Chapter 3. Third, descriptions of the machine learning methods used in this work for classification, analyzing algorithm performance, and fusion are provided. Lastly, we provide a description of how, and under what conditions, the FLGPR data used in this work was collected.

## 2.1 *Overview of detection processing for FLGPR*

The detection processing pipeline utilized in this work is shown in Figure 3. This process is representative of those considered in many studies [13], [22], [23]. The process begins with the data collection and beamforming described in Sections 2.1.1 and 2.1.2. Once FLGPR data from each data collection, or lane, is beamformed into radar images, a *prescreener* is run on the beamformed images and used to identify a subset of "suspicious" locations, termed alarms. This reduces the amount of imagery that must be considered for

more sophisticated processing (i.e., feature extraction and classification) [3], [4]. The result of prescreening is a list of alarm locations and a decision confidence from the prescreener processing that indicates how likely each alarm is to be a target.

The prescreening step is followed by feature extraction, in which features, or statistics, are extracted from imagery surrounding each of the prescreener alarm locations. These features are provided to the next stage of processing, classification, where a trained machine learning classifier is used to assign a new decision confidence to the alarm. The output of the classifier stage is a list of alarm locations (the same as the prescreener), but now with the assigned classifier decision confidence.

The next step of processing is the performance assessment. The score for a particular algorithm (i.e., a combination of features and a classifier) is computed using receiver operating characteristic (ROC) curves, which describe the tradeoff between target detection performance and false alarm rate. This is a performance metric that is commonly used for buried target detection algorithms in FLGPR [8], [10], [13], [16], [18]–[23].

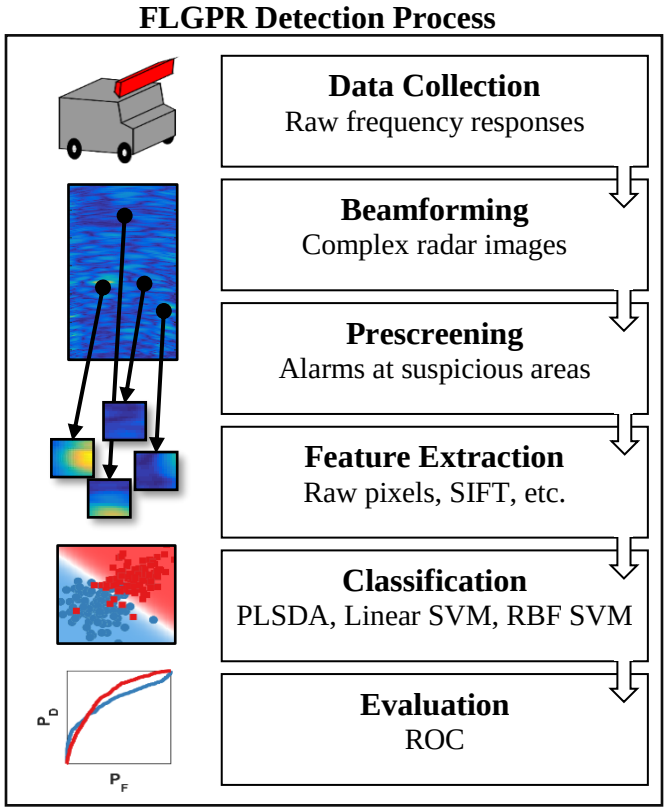



**Figure 3: Processing chain for the FLGPR system. Each block represents a major step in the processing pipeline. To the left of each stage is a small illustration outlining the process.**

### 2.1.1 FLGPR data collection

The data used in this work is collected by an FLGPR system that employs a bi-static antenna array and inspects the ground using a series of stepped frequency pulses [8], [15], [20]. Frequencies are emitted and collected one at a time for each transmit – receive antenna pair. The magnitude and phase change for each emitted frequency is measured and converted, with an inverse Fourier transform, into the time domain. This

processing yields a time-series for each transmit-receive pair. These time-series are the input to the beamforming process are used to synthesize FLGPR images [34].

In this FLGPR system, the ground is interrogated with L-Band frequencies (0.4 – 3.2 GHz). This frequency range was chosen, similar to reasoning for the downward-looking GPR, for its ability to penetrate the ground and generate responses from typical landmine-sized objects [15], [16], [18]. Using a stepped frequency sampling scheme, the L-Band is sampled in 2,702 frequency steps; i.e. sinusoidal pulses are emitted from each antenna between 0.4 – 3.2 GHz in 1.04 MHz steps [16].

The system uses three separate polarization schemes, HH, VV, and VH. The first letter in this notation corresponds to the transmitted polarization, and the second corresponds to the polarization measured by the receiving antenna. Polarization is a categorization for the orientation of the electrical component of the electromagnetic signal: the horizontal (H) or vertical (V) orientation. All the vertical antennas are evenly spaced, in one row, across the top of the vehicle. The horizontal antennas are split into three rows, evenly stacked vertically; and in each row antennas are evenly distributed across the vehicle.

### 2.1.2 Beamforming to create images

In this context, beamforming is used to describe the process of synthesizing images from the multiple time-series returns of the FLGPR transmit-receive antenna pairs. The beamforming process improves the signal-to-noise ratio (SNR) of the signals by averaging

over the returns from multiple antennas, this also provides potential information about the shape and size of objects [14]. Each beamformed image can be thought of (crudely) as a map of the radar energy reflected from the ground over some spatial extent in front of the vehicle [8]. Figure 4 demonstrates the complete imaging process beginning with raw complex frequency data and proceeding to imaging.

Conventionally, FLGPR imaging is produced using the delay-and-sum (DAS) method. DAS first computes the radio wave travel time between the individual antennas and the area under inspection (i.e., image pixel) and then sums the corresponding time-series values across the sensor array. Specifically, each antenna, $m$, has a time-series return, $x_m$; these time-series were estimated using the inverse Fourier transform of the magnitudes and phases from a range of stepped frequencies [8]. In accordance with the DAS beamforming method, the radio wave travel time, $R_{mp}$, from each antenna, $m$, to a desired pixel location, $p$, is then computed for use in indexing the estimated time-series signal. The summation in (1) shows the mathematical representation of the DAS beamforming.

$$I_p = \sum_{m=0}^{M-1} \alpha_m x_m(R_{mp}), \tag{1}$$

here $I_p$ is the image intensity at pixel location, $p$, and $a_m$ is the antenna window value for antenna $m$. Investigation into this antenna weighting is the primary consideration of the second part of this work and is discussed in more detail in Chapter

6.

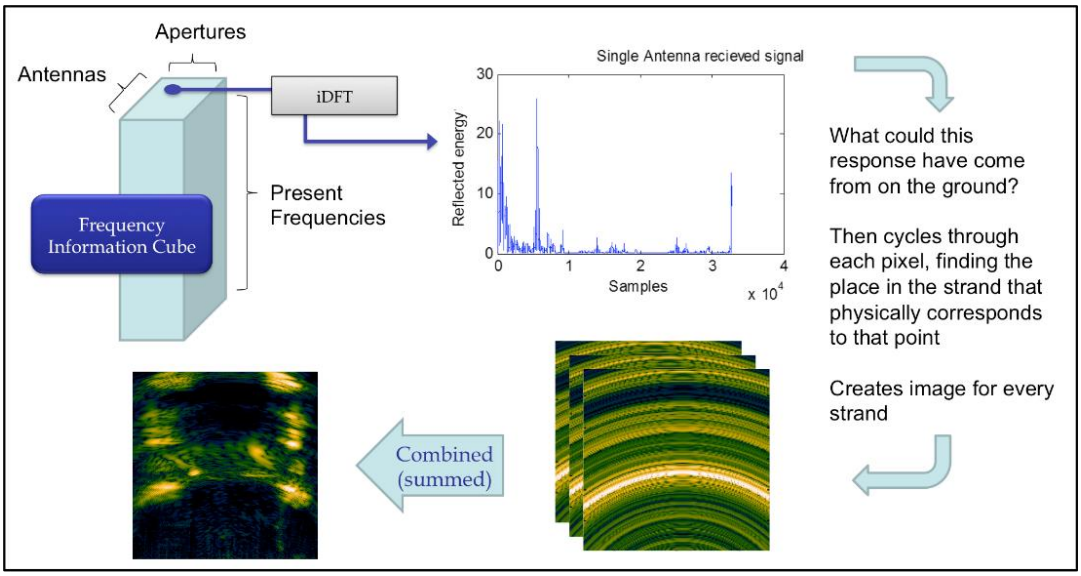



**Figure 4: This figure illustrates the process diagram of radar imagingfor a frequency cube, through time domain conversion, and then backprojection onto a desired area. The frequency information cube is the data over multiple antennas and apertures used to form one image. At each aperture and antenna, the received frequency information is converted to the time domain with an IDFT. This complex time domain signal is then backprojected across a 2D sampled area. These images, from every antenna, are coherently summed to form a radar image.**

In this work, each beamformed image corresponds to an area extending from 5 meters in front of the vehicle to 25 meters in front of the sensor. The images are roughly 6 meters wide (i.e., perpendicular to the travel direction of the vehicle). The images are created with a spatial resolution of 2.5$cm$/pixel. A separate image is beamformed every 2.5 meters, meaning that each location on the ground is included in multiple consecutive

images. Figure 5 illustrates three sequential frames of beamformed FLGPR data.

### 2.1.3 The prescreeners (A and B)

We developed two prescreeners, termed prescreener A and prescreener B, respectively. Prescreener A was developed first, and relied upon VV polarization L-band data. Prescreener B was developed later, and relied upon a combination of HH and VV polarization L-band data, and yielded substantial performance improvements.

Both prescreeners shared some common processing steps, beginning with an RX algorithm [35]. The RX algorithm, and other similar anomaly detectors, have been applied previously as prescreeners to this problem [13], [22], [23]. RX is a constant false alarm rate (CFAR) detector that identifies anomalous data by comparing the statistics of pixels in a foreground window to statistics in a background window at every pixel in the radar imagery [36]. A confidence image, $\lambda = (\mu_t - \mu_b)^2/\sigma_b^2$, is computed for every radar image; here $\mu_t$ is the mean of the foreground pixels, $\mu_b$ is the mean of the background pixels, and $\sigma_b^2$ is the variance of the background pixels. Alarms are declared by the prescreener at the local maxima of these confidence images.

Due to the multi-look nature of the beamformed images, each area of the ground is imaged in multiple frames of beamformed imagery as the vehicle moves down the lane. This means that the prescreener will often return multiple alarms for a single object (e.g. buried targets, clutter, etc.) imaged by the radar. To reduce this occurrence, and to combine the information in subsequent images, alarms are clustered [13], [37], [38]. For

this purpose, a clustering algorithm is used that enforces a limit on cluster radius, in order to maintain the locality of alarms. DP (Dirichlet Process) means is an extension of K-means that can enforce a limit on the cluster radius [39], and is implemented for our work. The alarm location after clustering becomes the cluster center, and alarm confidence is the $l_2$-norm of the cluster member alarms' RX confidences [13], [37].

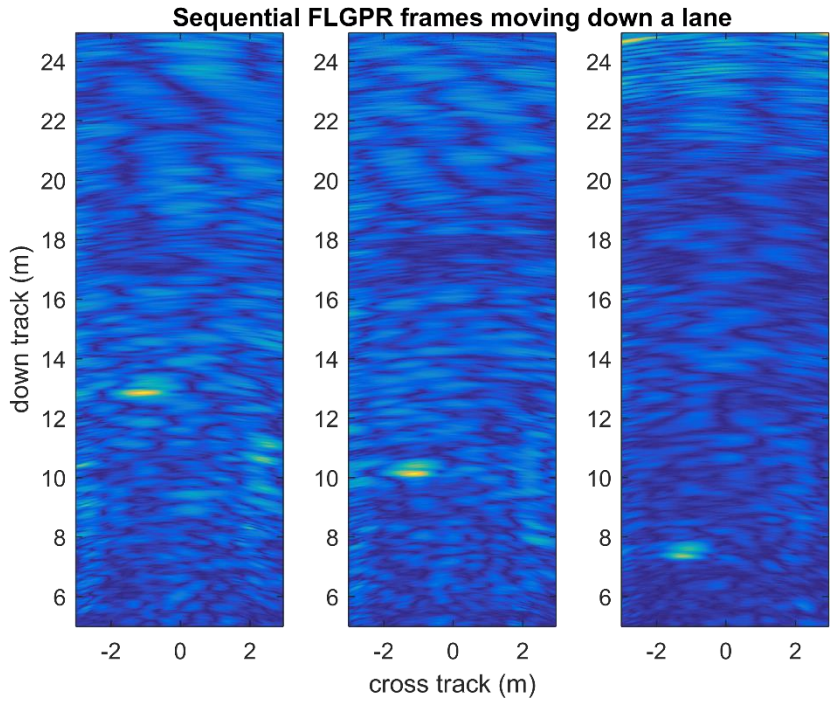



**Figure 5: Sequential frames in the HH polarization, formed as the sensor moves forward from the left most image to the right most image. Notice the target signature moves towards the bottom of the radar image (closer to the sensor array) as the sensor moves down the lane and towards the target.**

#### 2.1.3.1 Prescreener configuration A (VV images only)

The first version of our FLGPR prescreener utilized only on the VV polarization images. Alarms were gathered from each frame of the VV polarization images and clustered in UTM space as described above. By empirically optimizing for the best ROC performance, we found that target boxes of 40 x 40 pixels, background frames of 80 x 80 pixels and a cluster radius of 1 meter resulted in the best detection performance. These filter sizes are similar in size to those used in other studies [13], [22], [23]. For the experiments described Chapters 3 and 4 this prescreener is used. This prescreener was also used with limited target type population described with the experiment dataset in Section 2.4.1.

#### 2.1.3.2 Prescreener configuration B (HH and VV polarization images)

The second version of the prescreener used for this work was updated and improved based on the results from the large-scale feature comparison, the results of which are presented in Chapter 3. Prescreener B utilizes both the VV and HH polarizations, which we found to yield substantial detection performance improvements. Alarms are declared in each frame of the VV and HH images independently, using the local maxima of the RX filter output. Subsequently, all of the resulting alarms are clustered in UTM space using the method described above. By empirically optimizing for the best ROC performance, we again found that target boxes of 40 x 40 pixels, background frames of 80 x 80 pixels and a cluster radius of 1 meter resulted in the best performance

across both polarizations. These are the same filter sizes determined to be optimal for the first prescreener configuration. This prescreener is used for the experiments described in Chapters 5-8. Since prescreener B resulted in improved detection across all target types, the target population was expanded to include more traditionally difficult targets. This experimental dataset is described in Section 2.4.2.

### 2.1.4 Feature extraction

The work presented in Chapter 3 compares a variety of features extracted from patches beamformed at prescreener alarm locations [13], [21], [23], [40]. These features are those used by statistical classifiers to further discriminate buried targets from prescreener false alarms. For each prescreener alarm location, a $3m$ x $3m$ patch was beamformed at a centered down-track distance of 5 meters, and with a spatial resolution of $3cm$/pixel. These specifications are similar to those used in previous FLGPR studies [13], [21], [23].

Prior to implementing most feature extraction methods, images are normalized by their local background energy [23], [37], [41]. The background is defined here as anything outside of the 1.5m x 1.5m center of the patch. Here, $X_n$ is an alarm patch, or observation image, $\mu_{bg}$ is the mean of the background, and $\sigma_{bg}$ is the standard deviation of the background. This normalized observation, $X_n'$, is defined as follows:

$$X_n' = \frac{|X_n| - \mu_{bg}}{\sigma_{bg}} \tag{2}$$

### 2.1.5 Statistical classifiers

In this work we considered three classifiers: a linear support vector machine (SVM) [24], partial least squares discriminant analysis (PLSDA) [42], and a nonlinear SVM [24]. The two SVM classifiers were used because they have been implemented in previous studies using FLGPR data with a variety of feature sets, and as such provided a baseline. The non-linear SVM has been used with the 2D FFT and log-Gabor statistical feature [23]. This non-linear SVM uses a radial basis function with the default parameters settings C = 1 and $\gamma = \frac{1}{\#\ of\ features}$. The linear SVM is tested here because it is the conventional classifier used with the proposed feature learning approaches [26], [43]. It has also been previously used with these FLGPR image features: raw pixels [13], SIFT [21], and local image statistics [21]. The PLSDA classifier is considered because we found that it has previously been effective on FLGPR image features [37]; it also has the advantage of being computationally fast to train and run. PLSDA generally achieves similar, or better, performance on all of the features, as demonstrated by the results presented in Section 3.3.

### 2.1.6 Detection scoring metrics

The detection algorithms in this work (i.e., the various feature-classifier combinations) are compared using receiver operating characteristic (ROC) curves. ROC curves provide a measurement of the tradeoff between the target detection rate, or probability of detection, $P_d$, and the false alarm rate, FAR, as the sensitivity of the classifier is varied. Here $P_d$ refers to the proportion of observed targets that are detected by the

algorithm, and FAR refers to the number of false detections that the algorithm returns per square meter of observed lane area.

A related performance metric is also used in these comparisons, called the partial area under the ROC curve (pAUC) [44], [45]. The pAUC measure is the normalized area under the ROC curve from the origin of the x-axis to a specific FAR. For these experiments a pAUC is measured to a FAR of 0.02 FA/$m^2$, which corresponds to one false alarm every 50 $m^2$. The pAUC is normalized so that the minimum and maximum attainable values are 0 and 1, respectively. As the area under an ROC curve increases it reflects the ability of that algorithm to detect more targets within these ranges of sensitivity. This measurement is common to FLGPR work, and used as an alternative to directly comparing algorithms' ROC curves [21].

## 2.2 *Previously proposed FLGPR image features*

After introducing the detection pipeline used in most FLGPR processing, we will now review the radar image feature extractors previously proposed. The detection performance achieved with each of these feature methods will be compared in Chapter 3. Throughout this section, we will use $\psi$ to denote the feature vectors that are provided to a classifier for each method.

### 2.2.1 Raw pixels

This feature consists of rasterizing the pixels in $X'_n$, and treating them as a feature vector. More precisely, the raw pixel feature is given by $\psi_{Raw}(X'_n) = \text{vec}(X'_n)$, where the

vec(·) operator refers to the vectorization of a matrix. This type of feature is often used as a simple benchmark in image recognition tasks [27], [28] and has previously been applied to FLGPR [13], [18], [37].

### 2.2.2 Scale invariant feature transform (SIFT)

The SIFT feature aggregates the gradients over regions in an image into a histogram [46]. The first step in computing the SIFT descriptor involves calculating the gradient magnitude and orientation over the radar image, $X_n'$. Below, (3) and (4), show the magnitude, $M(i,j)$, and orientation, $\theta(i,j)$ , gradient calculations with $i$, and $j$ indexing the pixels of the image.

$$M(i,j) = \sqrt{\begin{matrix}\left(X_n'(i+1,j) - X_n'(i-1,j)\right)^2 + \\ \left(X_n'(i,j+1) - X_n'(i,j-1)\right)^2\end{matrix}} \tag{3}$$

$$\theta(i,j) = \tan^{-1}\frac{\left(X_n'(i,j+1) - X_n'(i,j-1)\right)}{\left(X_n'(i+1,j) - X_n'(i-1,j)\right)} \tag{4}$$

With these gradient calculations, the orientations are then aggregated into 4 by 4 non-overlapping cells. In each aggregation cell, the histogram of orientations is computed. This histogram separates angles into 8 bins between 0 and 360 degrees. The count for each angle bin is computed within each cell and is based on the magnitude of the gradients. There are 16 total cells and 8 angle bins resulting in a 128-dimentional descriptor vector, $f_{SIFT}(X_n')$. The final feature vector is this SIFT descriptor computed over

the whole alarm image.

$$\psi_{SIFT}(X_n') = f_{SIFT}(X_n'). \tag{5}$$

The SIFT operation is similar to the HOG (Histogram of Oriented Gradients) features used in previous FLGPR work [13], [21]. The SIFT feature, in this context, is performed over the image at one scale, and differs from 8-bin HOG in two ways [47]. First, HOG typically uses a 3 by 3 cell grid for aggregating gradients. Second, HOG generally normalizes within each cell, something that is not performed here. SIFT is specifically used here because it provides a more direct comparison to the feature learning with SIFT descriptors described further in Section 3.1.1.

### 2.2.3 Local statistics (LSTAT)

Local statistics over an alarm patch are often used in the natural object detection literature [48] and have previously been applied to FLGPR data [21]. To compute an LSTAT feature, the patch is divided into a 3 by 3 grid of non-overlapping regions. The feature vector consists of the mean variance of the pixel intensities in each of the grid regions. Mathematically this can be expressed by,

$$X_n' = \begin{bmatrix} x_1' & x_2' & x_3' \\ x_4' & x_5' & x_6' \\ x_7' & x_8' & x_9' \end{bmatrix}_n, \tag{6}$$

$$\psi_{LSTAT}(X_n') = \{[\mathbb{E}\{x_r'\}, Var\{x_r'\}];\ r = 1, \dots, 9\}_n. \tag{7}$$

### 2.2.4 2-Dimensional fast Fourier transform (2D FFT)

This feature consists of the of the 2D FFT of the FLGPR alarm patch [22], [23], [49]. In contrast to the other features, the 2D-FFT is computed on the complex patch, $X_n$. Before the 2D FFT operation, a Hamming window $(H)$ is applied to $X_n$, and the real component is taken. Equation (8) precisely defines the final feature.

$$\psi_{FFT}(X_n) = \text{vec}(\,|FFT_{2D}\{Re(H \circ X_n)\}|\,), \tag{8}$$

where $FFT_{2D}$ refers to the 2D FFT of an image, and vec(·) refers to the vectorization of a matrix. In accordance with the original implementation of this feature in FLGPR, we will only select the first quadrant of the 2D FFT as the feature vector [19], [20].

### 2.2.5 Log-Gabor statistical feature

The log-Gabor filter bank is intended to localize frequency information in an image [50]. These filters were applied to the FLGPR data to potentially improve upon other frequency based features such as the 2D FFT [23]. The implementation for FLGPR data uses statistics computed on the log-Gabor filter responses, this is described in detail in Shaw et. al. 2015, Section 2.3 [23], which we followed in these experiments. The log-Gabor filter bank that is applied here consists of six orientations and six scales, resulting in 36 different filters. The response of an observation to the $i^{\text{th}}$ log-Gabor filter is noted as $S^i(X_n)$. Statistics are extracted over 3 by 3 non-overlapping regions for each filter response. Mathematically we denote these grid regions as,

$$S^i(X_n) = \begin{bmatrix} s_1^i & s_2^i & s_3^i \\ s_4^i & s_5^i & s_6^i \\ s_7^i & s_8^i & s_9^i \end{bmatrix}_n \tag{9}$$

Statistics in each of the regions are then computed,

$$f_{LG}(s_r^i) = \left[\mathbb{E}\{s_r^i\}, \mathrm{Var}\{s_r^i\}, \mathrm{Kurt}\{s_r^i\}, \mathrm{Skew}\{s_r^i\}, \|s_r^i\|\right] \tag{10}$$

Here Kurt{·} and Skew{·} are the third and fourth standardized moments, respectively. Additionally, ‖·‖ represents the Euclidean norm for that region of the filtered image. The statistics from each region in the log-Gabor responses are concatenated to form the final log-Gabor statistical feature vector.

$$\psi_{LG}(X_n) = \left\{f_{LG}(s_r^i);\ r = 1, \ldots, 9\ ;\ i = 1 \ldots 36\right\}_n \tag{11}$$

## 2.3 *Sequential forward search algorithm*

We utilized decision level fusion to leverage independent information in the decision confidences delivered by each feature set after classification. However, instead of combining all available feature sets' decision confidences, it is usually more effective (equal performance with less number of feature sets) to find a subset of decisions to fuse, and so for finding this subset the sequential forward search (SFS) algorithm is employed [51]. SFS is a method of selecting a subset of features, based on a chosen performance metrics (i.e. pAUC), to then be used in classification. This is employed when finding a subset of individual feature sets' decision confidences in the decision fusion experiments.

For this decision-level fusion, the decision confidences of the individual feature sets become the feature vector used by the classifier during fusion.

An illustration of the SFS algorithm is shown in Figure 6. SFS starts by finding the detection performance associated with each of the individual features, and builds a feature vector in a greedy forward manner. For the current $N_f$ (the number of features in the subset), the base feature set is the highest performer from $N_f - 1$, and each of the next possible combinations was tested. This process continues until adding features leads to no meaningful gain in performance. In these experiments, the SFS performance metric for each combination is pAUC to a FAR of 0.02 FA/$m^2$.

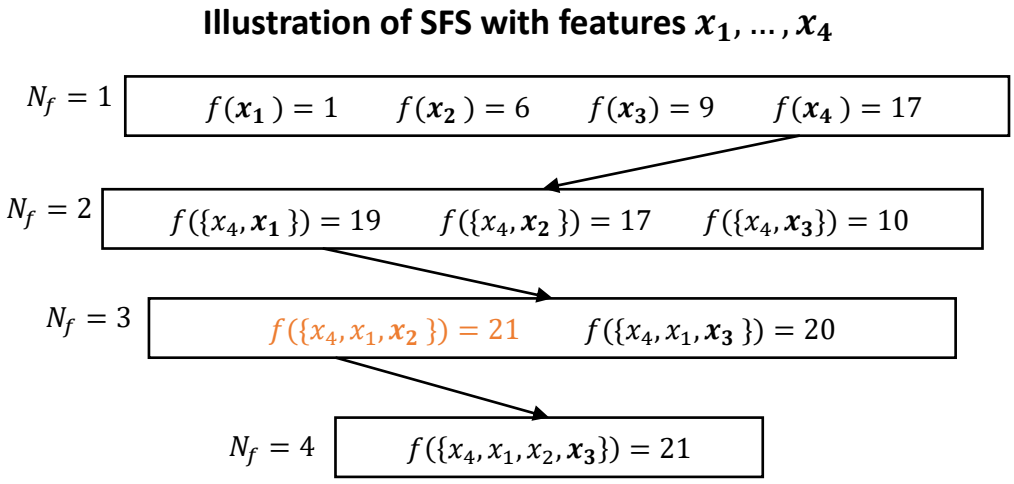



**Figure 6: This diagram illustrates a sequential forward search (SFS) algorithm.The base subset for any $N_f$ is the highest performing subset in $N_f - 1$. Performance for each subset is evaluated using $f$ for each number of features, $N_f$. In this illustrative example the highlighted entry would be the selected feature vector as it achieves the highest performance with lower $N_f$.**

## 2.4 *FLGPR experimental dataset*

All of the FLGPR collected data for the experiments detailed in this report was collected from a western test site in the U.S. during the spring of 2015. 10 data collection runs were conducted over 3 unique test lanes which totaled a scanned area of 36,000$m^2$. The precise location of each buried target was recorded so that the detection algorithms can be scored and trained with labeled examples. Each physical buried target is seen by the sensor multiple times in the collection, because some lanes are scanned multiple times in both directions. There are two datasets used in the feature sections of the work presented in this report, each corresponding to the output of a different prescreener (see Section 2.1.3 for prescreener details).

### 2.4.1 Prescreener configuration A and resulting dataset

Prescreener configuration A (see Section 2.1.3.1), which is used in the first part of this work had overall poorer performance. The lower performance led to modifying the target types considered for detection. For this dataset, the alarms which occurred over the deep (greater than 4 inches) and completely non-metal targets were removed. Accordingly, in scoring we did not penalize algorithms for missing any of the deep or non-metal targets. This resulted in a dataset with 245 target encounters over the 10 lane passes and 15,750 total alarm declarations. Of these 245 target encounters 202 were found by the prescreener, meaning that the highest probability of detection for a feature based algorithm is 0.824. The complete target type, depth, and metal content breakdown per lane is presented in Table 1. This dataset is used for the experiments in Chapters 3 and 4.

**Table 1: Target breakdown for the dataset that corresponds to the "prescreener configuration A". The total column is computed using all passes for each lane.**

| | **Lane A** | **Lane B** | **Lane C** | **Total – A** |
|---|---|---|---|---|
| **Lane passes** | 4 | 4 | 2 | 10 |
| **Lane area ($m^2$)** | 3,943 | 3,610 | 2,961 | 36,361 |
| **Unique targets** | 27 | 23 | 27 | 245 |
| **Metal** | 9 | 9 | 11 | 90 |
| **Low-metal** | 19 | 14 | 16 | 155 |
| **Non-metal** | 0 | 0 | 0 | 0 |

### 2.4.2 Prescreener and resulting dataset (B)

For the prescreener configuration B (see Section 2.1.3.2), alarms computed over the

HH and VV polarization images were included in the clustering for the final alarms. This prescreener improved overall performance and allowed for the inclusion of more traditionally difficult-to-detect target types. For the resulting dataset all the target types, depths, and metal contents were considered in scoring. Although there are more difficult-to-detect targets included in this dataset, with the improved prescreener the resulting performance still increased slightly. The dataset in this configuration has 546 total target encounters and the prescreener results in 19,128 total alarms; from those alarms 477 of the target encounters are found, which means the maximum performance for a feature based algorithm is a probability of detection of 0.874. The target type, depth, and metal content by lane is listed in Table 2. This dataset is used in the experiments described in Chapters 5 - 8.

**Table 2: Target breakdown for the dataset that corresponds to the "prescreener configuration B".The total column is computed using all passes for each lane.**

| | **Lane A** | **Lane B** | **Lane C** | **Total – B** |
|---|---|---|---|---|
| **Lane passes** | 4 | 4 | 2 | 10 |
| **Lane area ($m^2$)** | 3,943 | 3,610 | 2,961 | 36,361 |
| **Unique targets** | 52 | 53 | 77 | 546 |
| **Metal** | 18 | 30 | 51 | 284 |
| **Low-metal** | 34 | 30 | 38 | 314 |
| **Non-metal** | 4 | 4 | 19 | 68 |

# 3 Comparison of detection performance for different features using FLGPR images

This chapter compares a large set of image features and classification models that were previously proposed for buried threat detection in FLGPR imagery. We also present and describe two previously unexplored *feature learning* techniques for this application. These feature learning techniques are then compared, in terms of ROC performance, to the detection performance obtained with all of the previously proposed approaches. The experimental design is first described, followed by a presentation of the performance results. Each experiment in this chapter utilizes our first prescreener, prescreener A, described in Section 2.1.3.1 and the resulting dataset described in Section 2.4.1.

## 3.1 *Image based feature learning approaches*

This section presents the two modern feature learning approaches from the computer vision research community that we investigated for buried threat detection in FLGPR imagery: bag-of-visual-words (BOV) and Fisher vector (FV). In contrast to the previously proposed features, these methods automatically adapt to the data, and therefore necessitate being trained. Figure 7 provides a high-level overview of the training/testing process, which is similar for both of the two approaches that were investigated.

Details of the steps shown in Figure 7 are described further in the subsequent subsections. First, we will describe the (two) types of dense descriptors extracted in the

images. Next, we describe the clustering and encoding processes for each of the two methods, BOV and FV, respectively. Lastly, we provide some additional implementation details, explaining the specific design choices we made to adapt the BOV and FV approaches to work effectively for target detection on FLGPR data.

### 3.1.1 Local image descriptors for feature learning techniques

The feature learning techniques considered in this work are used with local descriptors that are densely sampled over the alarm patches. In this section, we define an observation (i.e., an FLGPR patch) as a set of local descriptors, $\mathrm{X} = \{\mathrm{x_t} \in \mathbb{R}^D;\ \mathrm{t} = 1, \ldots, \mathrm{T}\}$. Here, $t$ is the local-descriptor index and the image is comprised of $T$ total local descriptors; $D$ is the length of each local descriptor. The local descriptors for this work are sampled densely around the image in overlapping sub-patches, as shown in Figure 7. Two popular local descriptors, from the BOV and FV literature, are considered in this work: raw pixels ("Raw"), and SIFT ("SIFT") descriptors [26], [27], [43], [46], [52], [53]. The Raw descriptor is simply a vector containing the raw pixel intensities in each sub-patch. The SIFT descriptor was described in Section 2.2.2, and measures local gradient information. Specific details about the implementation of both local image descriptors are provided in Section 3.1.4.

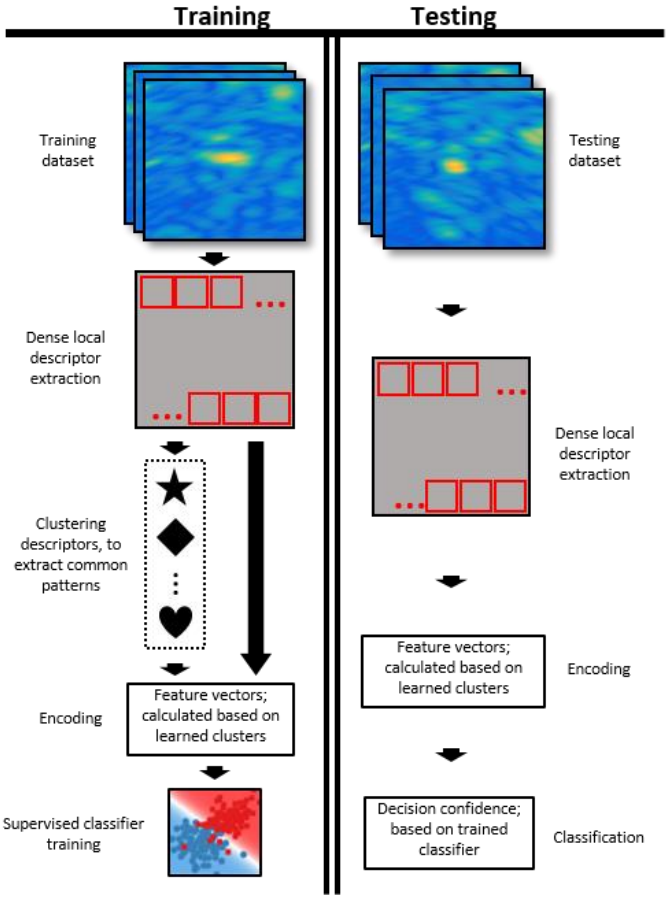



**Figure 7: A high-level diagram for the type of feature learning used in this work.First, in both training and testing, each approach requires that descriptors are densely extracted in small overlapping windows over each FLGPR image. During training, these descriptors are clustered to learn a codebook, or dictionary, of common patterns encountered in the data. The dictionary is then used to "encode" each individual FLGPR image, and this encoding acts as the feature vector for that image. This same procedure is repeated during testing to extract a feature vector, except that no clustering is needed.**

### 3.1.2 Bag-of-visual-words (BOV)

Our implementation of BOV is based on Coates and Ng, 2012 [26], but was modified for the specific application of target detection in FLGPR data. There are two

main components of the BOV implementation: the dictionary creation, and the encoding of observations.

To train the BOV algorithm, local descriptors from all training observations are clustered using Spherical K-means [54]. This yields a set of representative data signals, referred to as a dictionary, $\mathcal{D} \in \mathbb{R}^{K \times D}$. Following the suggestion of [26], the descriptors are whitened with zero component analysis (ZCA) before applying spherical K-means. ZCA whitens by projecting the descriptors onto an orthogonal basis. This tends to result in more independent dictionary elements, a desired trait in BOV [26].

The BOV encoding (i.e., feature vector) consists of a similarity measurement of $X$ to the learned clusters, or dictionary elements. The similarity measure is given by

$$\gamma_t(k) = \mathcal{D}_k x_t, \tag{12}$$

where $\mathcal{D}_k$ is the $k^{\text{th}}$ element of the dictionary. The feature vector for an FLGPR image is $K$ dimensional, and the $k^{th}$ feature value consists of the maximum inner product across all of the descriptors for the given FLGPR image, this is given by

$$\psi_{BOV}(X\,|\mathcal{D}) = \left\{ \max_t \{\, \gamma_t(k) \,\}; k = 1 \ldots K \right\}. \tag{13}$$

Notice that the encoding in (12) does not encode any information about the spatial location of descriptors, and therefore the BOV descriptor only encodes what is in the image, not where it is. Spatial information can be included in the encoding using spatial pooling, discussed in Section 3.1.4.

### 3.1.3 Fisher vector

Similar to BOV, the FV is designed to measure the occurrence of learned structures in a single observation. The FV, as implemented here, is based on Sanchez et al., 2013 [43]. Similar to BOV, during training, the FV begins with a clustering operation on the densely extracted local descriptors. Rather than K-means however, FV employs a K-component Gaussian Mixture Model (GMM). Inferring the parameters of the GMM requires finding the means and covariances for each cluster: $\mu_k \in \mathbb{R}^{1\times D}$, and $\Sigma_k \in \mathbb{R}^{D\times D}$. Here the subscript $k$ refers to the $k^{\text{th}}$ component in the GMM. Following common practice, we constrain the covariance matrix to be diagonal, implying that the elements of the local descriptor are independent. A single component of the GMM is given by

$$u_k \triangleq \mathcal{N}(\mu_k, \Sigma_k), \tag{14}$$

and the GMM is expressed as

$$u_\lambda \triangleq \sum_{k=1}^{K} w_k u_k, \tag{15}$$

where $w_k$ refers to the probability of each cluster (and therefore $\sum_k w_k = 1$), and $\lambda$ denotes the set of parameters learned for the GMM. In other words, $\lambda = \{w_k, \mu_k, \Sigma_k;\ k = 1, \ldots, K\}$.

Once the codebook (i.e., trained GMM) is available from training, it can be used to encode (i.e., compute a feature vector) for a new observation. The encoding (roughly) consists of computing first and second order differences between each of the $T$ descriptors, $x_t$, and the cluster centers. This computation is given by equations (16) and (17) below:

$$\mathcal{G}_{\mu_k}^X = \frac{1}{\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \left( \frac{x_t - \mu_k}{\sigma_k} \right), \tag{16}$$

$$\mathcal{G}_{\sigma_k}^X = \frac{1}{\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \frac{1}{\sqrt{2}} \left[ \frac{(x_t - \mu_k)^2}{\sigma_k^2} - 1 \right]. \tag{17}$$

The function $\gamma_t(k)$ in the preceding two equations is given by

$$\gamma_t(k) = \frac{w_k u_k(x_t)}{\sum_{j=1}^{K} w_j u_j(x_t)}. \tag{18}$$

In (15) and (16), and given a particular value of $k$, the terms $\mathcal{G}_{\mu_k}^X$ and $\mathcal{G}_{\sigma_k}^X$ are each vectors of length $D$, where $D$ is the dimensionality of the input descriptor (e.g., Raw or SIFT). The final FV feature is given by

$$\psi_{FV}(X|u_\lambda) = \{\mathcal{G}_{\mu_k}^X, \mathcal{G}_{\sigma_k}^X, k = 1..K\}, \tag{19}$$

and its dimensionality is 2DK.

It is important to notice that each encoding $\mathcal{G}_{\mu_k}^X$ and $\mathcal{G}_{\sigma_k}^X$ consist of a weighted sum, or "pooling", of contributions from individual descriptor differences. This implies that the spatial location of each descriptor is lost in the computation of the FV feature. Consequentially, and similar to the BOV encoding, the FV feature only encodes information about *what* is in each FLPGR image, but not *where* it exists in the image. As noted above, this is addressed through a technique called spatial pooling, which we describe next in Section 3.1.4.

### 3.1.4 Additional feature learning implementation details

This section presents some additional details about our implementation of BOV

and FV in order to adapt them to target detection in the FLGPR imagery. We first describe the details of the local descriptor extraction. Second, we describe the number of clusters, or components, used for these feature learning methods. Lastly, we describe our application of spatial pooling to enhance the performance of both the BOV and FV features.

In this work the BOV and FV methods are each applied using two types of local descriptors. First, raw radar image intensities of 11 x 11 pixel regions were densely sampled over the FLGPR images, with an overlap of 4 pixels. These sizes were chosen because in optimization experiments, they performed the best for both BOV and FV. This first descriptor will be referred to as BOV (Raw) and FV (Raw). The other local descriptor tested was SIFT, which was extracted every 8 pixels, over 8 x 8 pixel regions, for each background normalized magnitude alarm patch. This size and spacing were also chosen to maximize performance in cross-validation. Once again, the same settings yielded the best results for both the FV the BOV features. The SIFT-based BOV and FV features will be denoted as BOV (SIFT) and FV (SIFT), respectively.

For each encoding scheme 30 component clustering was used, and this was performed for both types of descriptors: SIFT and Raw. Again, in optimization experiments, we found that changing the number of components for both BOV and FV in the 10 – 50 component range yielded very similar overall detection performance.

Spatial pooling is also applied to BOV and FV for both types of local descriptors.

As described in Sections 3.1.2 and 3.1.3, the BOV and FV features primarily encode *what* is in the image, but not *where* it is. This is a well-known limitation of the BOV and FV approaches [27] and, similar to other recognition tasks [32], [55], we discovered that spatial information is important for identifying buried threats in FLGPR data. As a result, we adopted a technique called "spatial pooling" [55], which can be used to augment the feature encodings with spatial information. Figure 8 illustrates the concept of spatial pooling, as well as the way we applied it in this work. The radar image first has the dense local descriptors extracted; the simplest version of spatial pooling is to extract the BOV and FV encodings over the whole image area. Instead, spatial pooling can be applied in other sub-regions around the image; this is represented by the right-most encoding scheme in Figure 8. We found that spatial pooling using a 2 by 2 non-overlapping grid (as shown in Figure 8) resulted in substantial performance improvements, for both feature learning methods (BOV and FV), and for both local descriptors (Raw and SIFT).

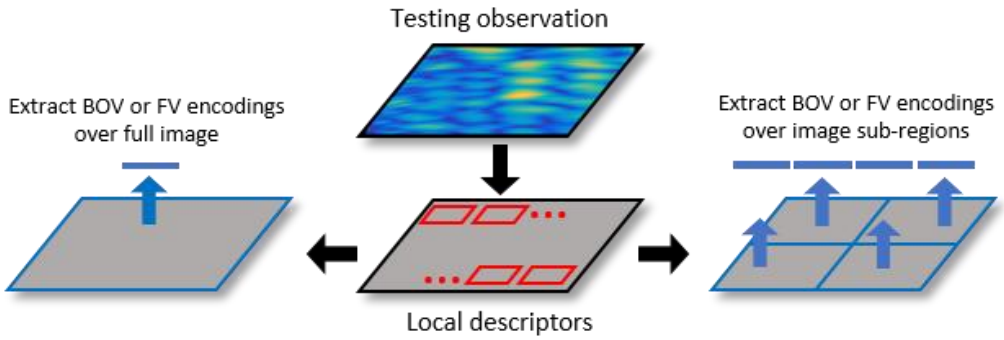



**Figure 8: Illustration of different spatial pooling techniques for BOV and FV.In all cases, local descriptors are densely extracted over the FLGPR image. In the original pooling scheme (left), one encoding is computed over the entire image. The encoding discards information about where objects exist in the image. In order to encode this information, we applied spatial pooling in which BOV and FV encodings can be computed on different sub-regions of the image. The resulting encodings (e.g., four in the illustration) are concatenated together to form one long feature vector, where each of the four segments encodes information about the image content in its respective spatial region. In this work we applied a 2 by 2 non-overlapping pooling scheme (as shown), which substantially improved performance.**

## 3.2 *Experimental Results*

In this section we present the experimental results of the large scale FLGPR image feature comparison. We begin by reporting results (in terms of pAUC) for each combination of polarization, feature set, and classification model. After this, we present results where we used a feature selection approach to select a good subset of feature sets for use in decision fusion. Finally, we present an analysis of the results, providing additional insight regarding the features.

All of the subsequent experimental results were conducted using patches extracted at alarm locations declared by the prescreener described in Section 2.1.3.1. This

prescreener yielded 15,750 alarms (i.e., patches) over the entire dataset. The classification algorithms were tested using a lane-based cross-validation procedure. There were three distinct lanes in our dataset and thus, three-fold cross-validation was used.

## 3.3 *The performance of individual feature sets*

In this section we present detection performance results, in terms of pAUC, for each possible combination of (i) radar polarization, (ii) feature set, and (iii) a classifier (e.g., 2D FFT feature of an HH image with the RBF SVM classifier). We refer to such 3-tuples as "algorithms" for ease of discussion. For each algorithm we report an average pAUC and a 95% confidence interval based on bootstrap aggregation [56]–[59]. Specifically, for each training fold, we created ten different datasets by taking bootstrap samples of the training data (i.e., sampling the original training data, with replacement, until a new equally large dataset is created). A classifier is trained on each of training folds for the ten datasets, and then applied to the corresponding test sets to obtain predictions. These ten trials permit us to measure the mean and variance of the performance for each algorithm, which help indicate the consistency and robustness of the algorithm performance. The results of these experiments are presented in three separate figures, where each figure corresponds to a polarization: HH (Figure 9), VV (Figure 10), and HV (Figure 11).

The results reveal a clear trend in the detection performance on each polarity. The HH polarity yields the best mean detection performance across all combinations of features and classifiers, without any exceptions. This is followed by the VV polarity

which, in turn, always outperforms the VH polarity.

The results also indicate that there are general differences in the performance of the classifiers. Among the linear classifiers, PLSDA almost always outperforms the linear SVM. The only exceptions to this occur for the SIFT features, and is less consequential to the overall result due to the poor performance of SIFT features. We believe PLSDA produces superior performance due to its ability to deal with collinearity (i.e., redundancy) in the features [42]. Many of the feature sets investigated here are very high dimensional, and occasionally the dimensionality is much greater than the number of observations (e.g., the FV features). This high dimensionality tends to increase the redundancy of the features, likely making PLSDA a more suitable classifier.

In contrast to the linear SVM, the (non-linear) RBF SVM performs similarly to PLSDA. The RBF SVM is nonlinear, and therefore it can model more complex relationships between the features than PLSDA. The performance similarity of the RBF SVM and PLSDA suggests that the greater complexity of the RBF SVM yields few benefits. This further suggests that the patterns in this data are relatively simple. We provide further qualitative support for this assertion in Section 4.3. Although PLSDA and the RBF SVM perform similarly, the RBF SVM is much more computationally expensive (i.e., slower) than PLSDA, during both training and testing, and thus we generally favor the utilization of PLSDA.

For the remaining discussion, we will only consider results with the RBF SVM and

PLSDA, since they both (almost) always outperform the linear SVM. When comparing detection performance results for individual features, several trends are revealed. First, many algorithms are outperformed by the raw pixel features for each of the polarization imagery. Secondly, overall the best performing features operating on HH polarization imagery *tend* to be the FV and BOV based features, Raw, and LSTAT. The best overall mean performance is achieved with FV (SIFT) for the RBF SVM.

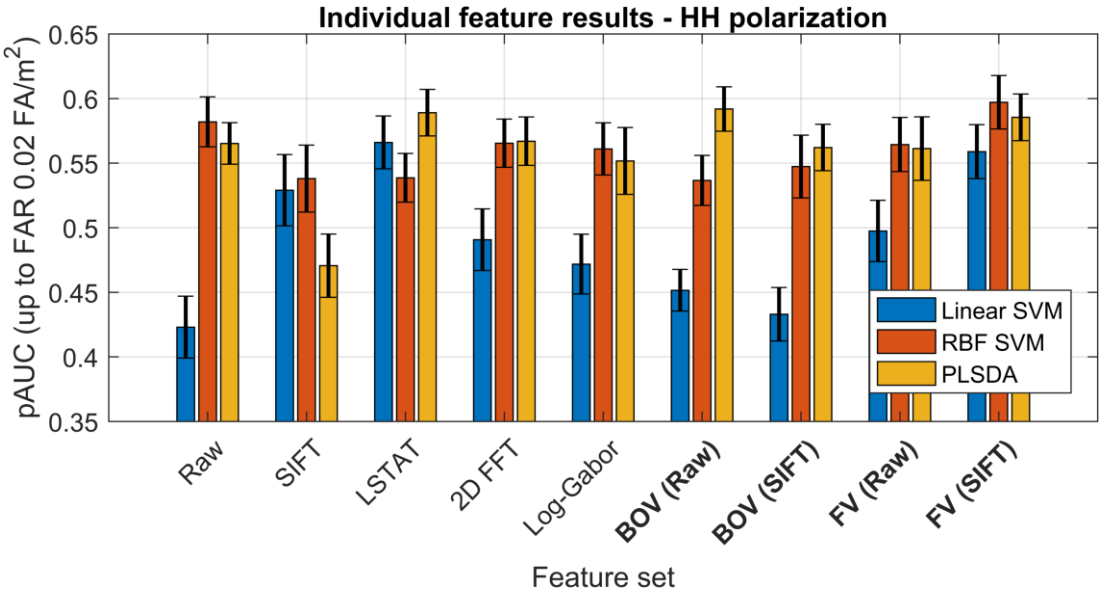



**Figure 9: pAUC, computed to a false alarm rate of 0.02 false alarms per $m^2$ for each individual feature set using the HH polarization data, for each considered classifier.Feature learning approaches are bolded. The mean pAUC and 95% confidence intervals for the 10 bootstrap trials are reported.**

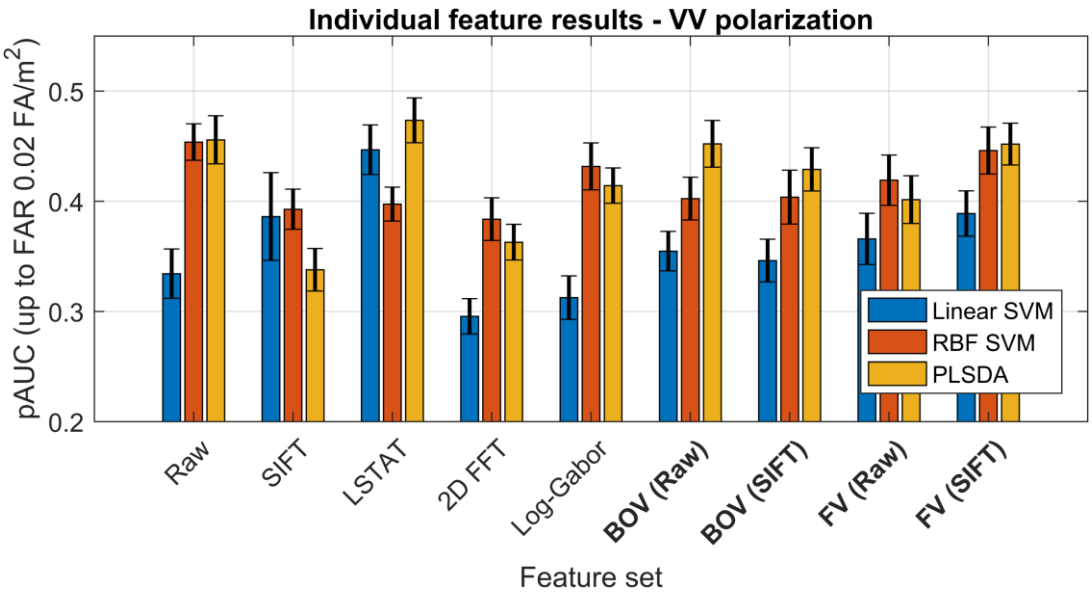



**Figure 10: pAUC, computed to a false alarm rate of 0.02 false alarms per $m^2$ for each individual feature set, using the VV polarization data, for each considered classifier.Feature learning approaches are bolded. The mean pAUC and 95% confidence intervals for the 10 bootstrap trials are reported.**

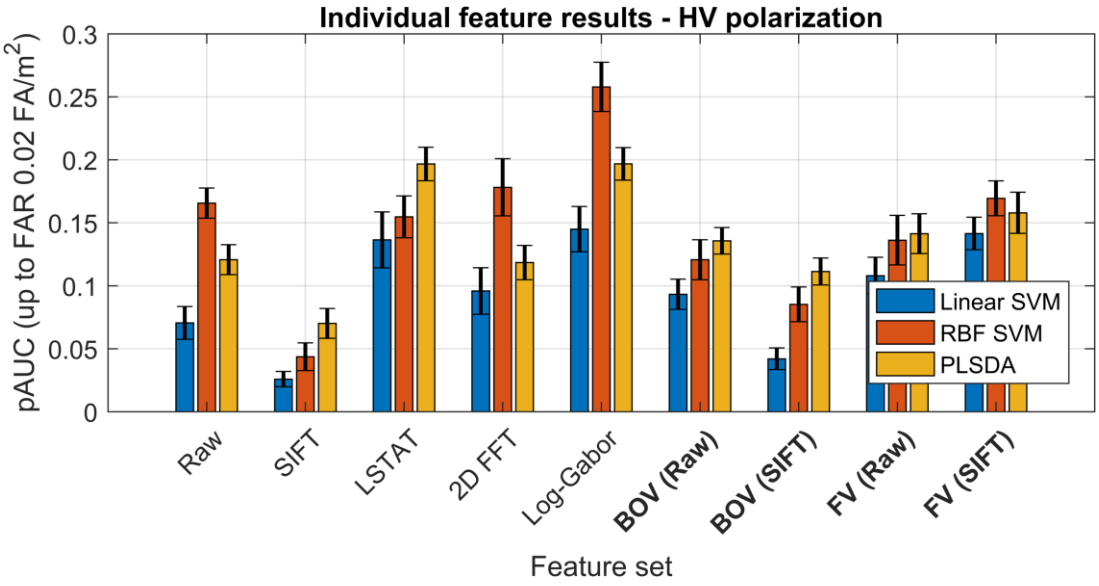



**Figure 11: pAUC, computed to a false alarm rate of 0.02 false alarms per $m^2$ for each individual feature set on the HV polarization data, for each considered classifier.Feature learning approaches are bolded. The mean pAUC and 95% confidence intervals for 10 bootstrap trials are reported.**

This section provides a table of the computational run-time for the algorithms compared here. Table 3 shows the computation run-time for each feature set – classifier combination. First, for each feature set the creation time is reported. Second, the training and testing time are each stated for every feature set – classifier combination. For the feature learning methods, the reported classifier run-times include the computational time for the clustering and encoding processes. Times are reported for a single bootstrap trial and are averaged across the three different polarizations. All experiments were performed on a system with a 3.40GHz Intel i7 processor and 16 GBs of RAM.

From Table 3 it is clear that multiple trends are present across both the feature set and classifier run-times. First, the PLSDA classifier is clearly the fastest classifier followed

by the linear SVM and then the RBF SVM. This speed difference is greatly expanded as the feature set dimensionality is increased, as evidenced by comparing the raw pixel run-times to those of the LSTAT feature. Second, the training and testing run-times for the feature learning techniques (BOV and FV) are greater than those of the previous feature sets and reveal the computational cost of implementing the more complicated models.

**Table 3: Computational run-time in seconds for each feature set creation, training, and testing for each classifier.**

| *Feature Set* | Creation | Training | | | Testing | | |
|---|---|---|---|---|---|---|---|
| | | PLSDA | Lin SVM | RBF SVM | PLSDA | Lin SVM | RBF SVM |
| **Raw** | **0** | **3.4** | **603** | **866** | **0.1** | **356** | **603** |
| **SIFT** | **37** | **0.3** | **6.0** | **4.5** | **0.4** | **2.9** | **3.3** |
| **LSTAT** | **1.6** | **0.1** | **2.4** | **2.1** | **0.4** | **0.7** | **1.4** |
| **2D FFT** | **6.6** | **0.9** | **157** | **299** | **0.4** | **99** | **198** |
| **Log-Gabor** | **1109** | **0.6** | **74** | **121** | **0.4** | **37** | **80** |
| **BOV (Raw)** | **201** | **168** | **181** | **180** | **138** | **140** | **144** |
| **BOV (SIFT)** | **225** | **215** | **269** | **331** | **121** | **966** | **1015** |
| **FV (Raw)** | **201** | **337** | **3430** | **9367** | **105** | **3656** | **6397** |
| **FV (SIFT)** | **225** | **196** | **2762** | **3594** | **103** | **2296** | **2764** |

## 3.4 *Summary and conclusions*

In this chapter, two contributions to the FLGPR research were presented; a comprehensive comparison of detection performance using existing features on a large consistent data collection, and the application of feature learning to the FLGPR radar data using the BOV and FV methods. Through these experiments a number of observations were made regarding the FLGPR processing system. First, that the best performance for any polarization and individual feature set results from the FV with SIFT descriptors on the HH polarization data. The feature learning approaches in general performed well, but did not outperform all previous feature sets for all polarizations. In summary, we find the following:

- The HH polarization contains more information for classifying buried threats, compared to both the VV and VH polarizations using the features and algorithms considered in this work.
- Feature learning approaches (FV with SIFT descriptors) yielded small improvement in detection performance when compared to the utilization of previously considered features, although detection performance between with all of the different feature sets is similar.
- A linear classifier, PLSDA, which was previously unused in this context, is able to perform roughly equal to and on some feature sets better than the RBF SVM.

Though there is an improvement from the feature learning technique, specifically the Fisher vector with SIFT, there is still a question pertaining to the utility of combining image features together in order to further improve performance. The next chapter applies a fusion method with the goal of taking advantage of the independent discriminability that is potentially embodied in each feature.

# 4 Decision fusion and feature analysis

This chapter contains decision fusion results along with an analysis of the individual image feature results. Decision fusion is a detection method that leverages performance from different feature sets once they each have been utilized by a classification model to generate a decision statistic. The decision statistics (or confidences) obtained by classifying each feature set are treated as feature vectors that are then the input into a second classifier, which attempts to combine them in order to obtain a single (better) decision statistic. The experiments in this chapter are designed to combine potentially independent information provided by each individual feature set. Also in this section, the individual feature set results are analyzed in an attempt to help explain what aspects of the FLGPR data are being captured by each feature. This analysis begins with a visualization of the learned filters from the feature learning techniques; providing insight into the shapes and patterns that are common among observations. Finally, a comparison of classification ability, for specific target examples, is presented to highlight the types of observations that the feature learning techniques classify well.

## 4.1 *Decision fusion experimental results*

In decision fusion we aim to combine the predictions (i.e., confidence values output by a classifier) from the different algorithms described in Section 3.3 in order to further improve detection performance [60]. Decision fusion was also investigated recently for FLGPR in [61], and we build on this work by using a larger collection of data

and a greater variety of image feature sets.

In order to fuse the algorithm predictions, we treat them as features that are input into a second PLSDA classifier, which is then trained to make a final prediction. For simplicity, in these experiments we only consider fusing the PLSDA classifier predictions, but we consider all polarities and feature sets. It is unlikely that all of these feature sets are useful for fusion, and so we attempt to select a good subset of them for fusion. To select this subset we used the sequential forward search algorithm (SFS) [51]. This algorithm begins with the predictors (i.e., the PLSDA classifier outputs) for the single best feature set, and then adds a new predictor one at a time, based on which one increases the performance the most. The SFS is only allowed to select new predictors based upon the training data in each fold, in order to avoid positive performance bias. Therefore, in order to obtain performance measures for each candidate fusion model, we perform a random five-fold cross-validation using only the training data. This yields pAUCs across the number of predictors that SFS can use to build the final fusion model.

In our experimental design, we control the total number of predictors that SFS is allowed to select for fusion, denoted $N_f$. Figure 12 presents the results of our feature fusion experiment as we vary $N_f$. Each point in Figure 12 is the mean of the pAUCs, and the error bars report the 95% confidence interval [59]. This interval is computed after repeating each experiment for the ten bootstrapped datasets from the individual feature comparison. The results show a sharp performance increase at $N_f = 3$ then a steady but slower rise until

$N_f \cong 15$.

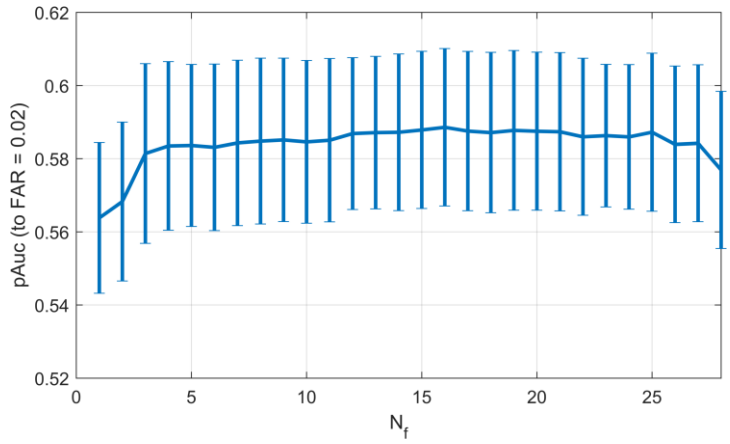



**Figure 12: pAUC obtained for decision fusion with the PLSDA algorithm as a function of the number of features used in the sequential forward search.Each point reports the mean performance for that number of features and the error bars are the 95% confidence intervals over multiple runs of the same experiment; for each $N_f$ we ran the experiment for the 10 bootstrapped datasets. Notice the leveling of performance after $N_f = 3$; adding more features beyond $N_f = 3$ provides only slight increases in overall performance.**

To estimate overall performance for the SFS algorithm we conducted an experiment using an auto-stopping criterion to select $N_f$. The algorithm would stop increasing $N_f$ when the pAUC within that fold started to decrease. The pAUC is determined from the random five-fold cross-validation performed within each training set, thus this experiment was repeated for the 10 bootstrapped datasets from the individual feature comparison to better estimate the performance. Within each 10 bootstrapped datasets lane-based cross-validation is performance; and within each of the lane-based training folds, five-fold cross-validation is used to determine which decision statistics to include in the SFS. In Figure 13 the vertically averaged ROC [62] and 95%

confidence intervals are reported for the SFS auto-stopping algorithm and the FV (SIFT) with the RBF SVM, along with the performance of the prescreener (PS). The decision fusion and best individual feature set yield similar performance at any given point on the ROC curve, with the SFS average being slightly lower.

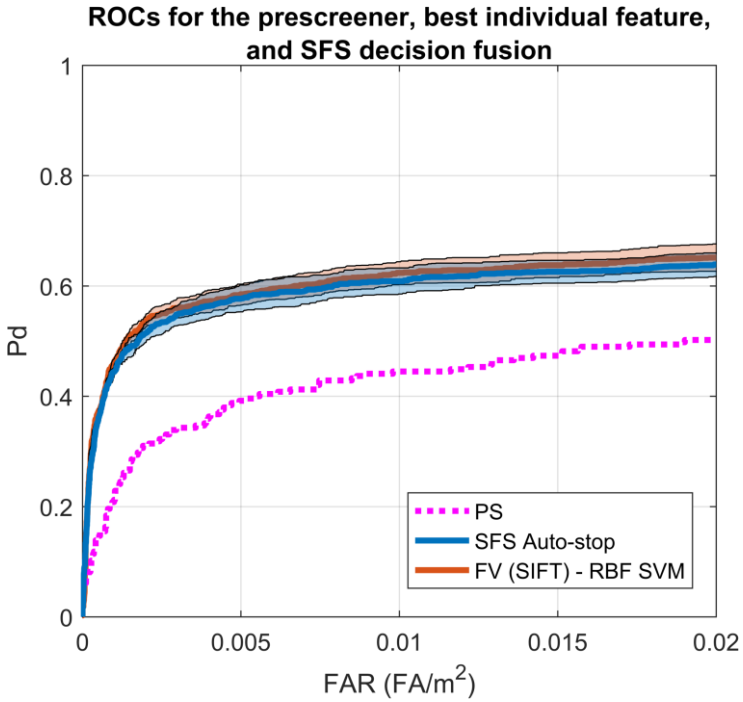



**Figure 13: ROC curve (probability of detection vs. false alarm rate) for prescreener (PS), the single best feature (FV SIFT – RBF SVM), and the sequential forward search (SFS) decision level fusion results.The FV SIFT and SFS fusion ROC curves have the 95% confidence areas shaded in their respective colors.**

## 4.2 *Learned filters*

Feature learning algorithms can be analyzed by visualizing the resulting filters from the clustering, and the results of such a visualization are shown in Figure 14 for the BOV (Raw) algorithm on the HH polarization. We analyze BOV (Raw) because (i) it was

nearly the best performing algorithm and (ii) it uses the raw pixel descriptor which results in more interpretable visualizations. During feature extraction the resulting BOV encodings are based on the similarity between each observation's descriptors and these cluster centers. We can surmise that these cluster centers are useful in discrimination because of the relatively good BOV (Raw) classification performance; that is to say, the space derived with these cluster centers produces a fairly discriminative representation. Notice the "blob-like" learned cluster center, and how many of the clusters are shifted versions of the same shape. This, along with knowing that performance did not increase when introducing more available clusters implies that there are not many cues beyond concentrations of high energy that indicate the presence of a target in FLGPR data.

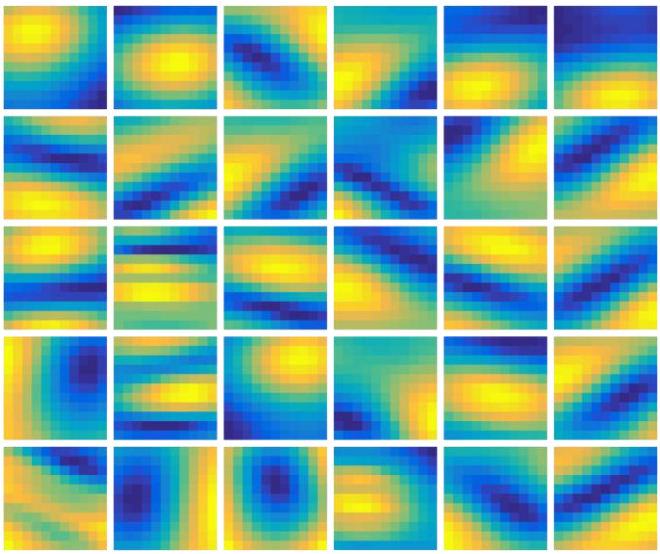


**Figure 14: Learned K-means cluster centers for the BOV on raw pixel local descriptors in the HH polarization.With this visualization, it becomes evident that the learned filters are measuring the different "blob-like" characteristics of the data.**

## 4.3 *What are the characteristics of FLGPR images that indicate the presence of a target?*

For many computer vision algorithms, including feature learning, it can often be difficult to understand which components of the input imagery are most indicative of the presence of the target object (e.g., buried threats). In an effort to analyze this we developed a "confidence map" visualization based on the feature learning encodings and trained classifier. The subsequent visualizations show the magnitude image and a BOV (Raw) "confidence map" of four target examples over a range of classification confidences. Again, BOV (Raw) on the HH polarization is used here due to its overall good classification performance and the visual interpretability of using raw pixel descriptors.

The "confidence maps" illustrate where, spatially, the BOV (Raw) features indicated target-like characteristics in the image. Figure 15 shows the process for obtaining this visualization. As the process shows, the dense descriptors (i.e., raw patches) in the image are encoded using a sliding window, where four descriptors are encoded at a time. This corresponds to spatial pooling of the BOV encodings (see Section 3.1.4) over very small image regions. Each encoding is then classified (i.e., assigned a confidence) using a trained BOV(Raw) PLSDA classifier. This process results in a map of confidence values across the image, indicating how much the local descriptors indicate the presence of a target.

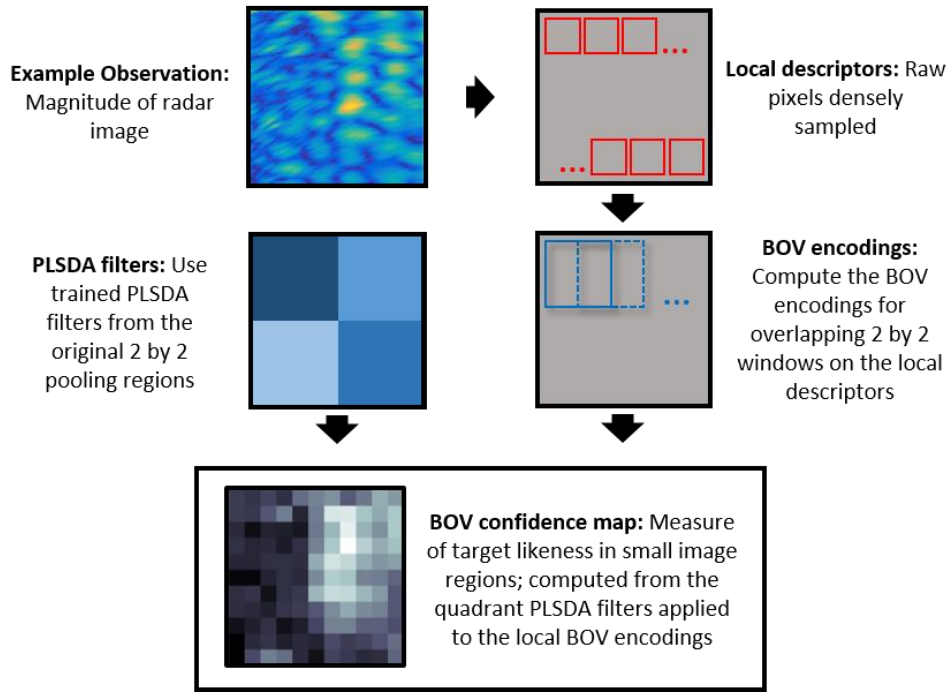



**Figure 15: Description of how the BOV (Raw) confidence map is computed for each example observation.Using the learned quadrant PLSDA filters and BOV encodings for 2 by 2 overlapping windows of the local descriptor, a measure of target-likeness is calculated for local regions around the observation. In order to encode the extracted descriptors, we used the same K-means clusters that were trained as described in Section 3.3. Obtaining an appropriate classifier for the local BOV encodings was more difficult. For this purpose, we used the PLSDA classifier that was trained as in Section 3.3, however, that classifier was trained using 2 by 2 non-overlapping pooling. Recall that this pooling scheme yielded one BOV encoding for each quadrant of the image. If the total BOV (Raw) dimensionality is given by $D$, then the PLSDA classifier consisted of $4D$ weights: one set of $D$ weights for each of the four quadrants. Therefore, for our discriminability map, if a local BOV encoding was located in the top right quadrant, then the $D$ PLSDA filters weights from that quadrant were applied to the encoding in order to assign a confidence. In the confidence maps, the brighter areas correspond to very target like locations.**

For each example presented in Figure 16, the magnitude images, BOV (Raw) confidence maps, and confidence percentile are given. The confidence percentile denotes the confidence a target was given by the classifier but normalized in relation to the other

observations to fit between 0% and 100% (100% confidence percentile being the highest confidence observation). The visualizations in Figure 16 reveal several interesting characteristics about the FLGPR images. First, it appears that the BOV (Raw) feature is largely cuing on high energy "blobs", of varying shape and size, to identify the presence of a target. As expected with the spatial invariance introduced by "bagging" in BOV, the algorithm is able to assign a high confidence to an off-center target response, like that for the 95.6% confidence percentile example. In the example in the last column (1.3% confidence percentile) there are high confidence locations around a target response but not enough to outweigh the large amount of background present in the image. The example with a 14.0% confidence percentile illustrates a very weak target response, and while some of the response appears target like it is not strong enough for the BOV (Raw) algorithm to classify well.

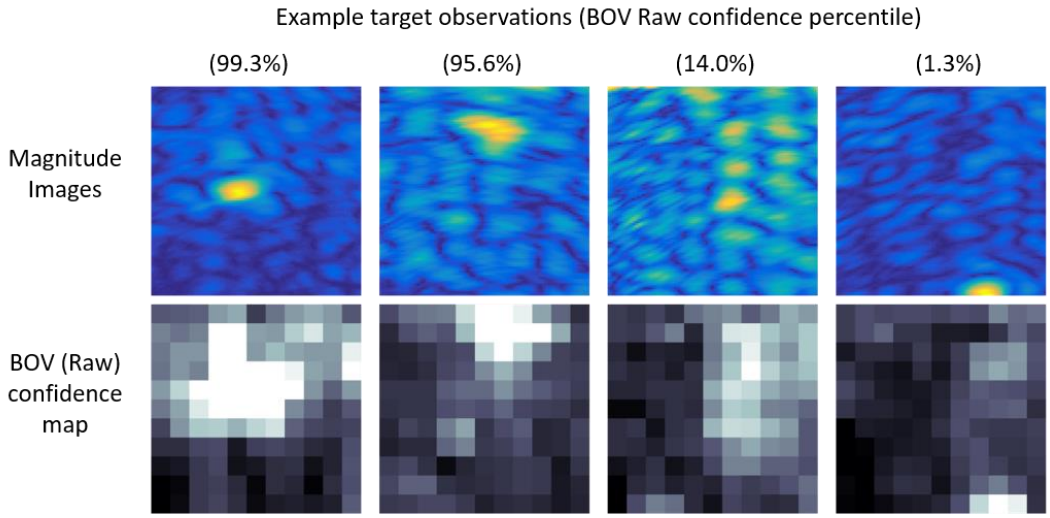



**Figure 16: Four target examples with various BOV (Raw) confidence percentiles.Along with magnitude image patch for each, a BOV (Raw) confidence map is included to show where in the image the algorithm found target-like characteristics**

## 4.4 *Summary of decision fusion approach*

This chapter used the individual classification results in fusion to combine potential independent information from the previously considered and newly feature proposed extractor algorithms. This greedy feature selection method ultimately provided no performance increase over the best single feature set, FV (SIFT). Also in this chapter, the feature learning approaches were examined to show what common shapes and structures were found in the radar images.

# 5 Proposed modifications to the 2DFFT feature

The large-scale comparison of FLGPR threat detection algorithms discussed in Chapter 3 (also described in [37], [63]) revealed that detection performance by classification algorithms when applied to most feature is similar. In this chapter, we propose enhancements to a previously considered frequency-domain feature [22], [23] that was based upon the two-dimensional Fourier transform (2D FFT) of the FLGPR imagery. This chapter presents a modified version of the 2D FFT feature, termed 2DFFT+, that yields substantial improvements to detection performance achieved when performing detection using the previously proposed 2DFFT feature, as well as substantially outperforming existing FLGPR algorithms. Further, we show that using partial least-squares (PLS) discriminative dimensionality reduction [64], it is possible to dramatically lower the dimensionality of the 2DFFT+ feature from 2652 dimensions to twenty dimensions (on average), while simultaneously improving its performance. In this chapter, and the remainder of this thesis, we use prescreener configuration B (see Section 2.1.3.2,) and the corresponding dataset (see Section 2.4.2).

## 5.1 *Description of the original 2DFFT feature*

In order to keep this chapter self-contained, we recap the original 2DFFT feature as it has been applied in numerous recent FLGPR threat detection algorithm publications [22], [23], [61], [65], and it is also described in more detail in Section 2.2.4. The first step in the calculation of this feature is to obtain the real component of the complex-valued

FLGPR alarm patches, and then apply a windowing function (e.g., a Hamming window). The 2DFFT feature then simply consists of rasterizing (i.e., vectorizing) the values in the first quadrant of the resulting 2DFFT. The first quadrant of the 2DFFT contains the positive frequencies in both the cross track and down track directions (see Figure 17).

## 5.2 *Modification 1: adding additional spectral quadrants to the feature*

The original 2D FFT feature used only one quadrant of the 2D FFT as the feature vector. Unfortunately, no motivation for the single quadrant approach has been provided when examined in previous FLGPR studies [22], [23], [61], [65]. We hypothesized that a 2DFFT feature including all of the unique frequency domain information (i.e., two quadrants) can improve threat and non-threat classification. By using the 1st and 2nd quadrants all of the unique information in the magnitude of the 2D FFT is being captured and presented to a classifier. The 3rd and 4th quadrants of the 2D FFT computed on a real-valued signal are complex conjugates of the 1st and 2nd quadrants and do not contain any unique information about the underlying signal [66]. Figure 17 provides an illustration of the difference in 2D FFT extractions over an example threat observation.

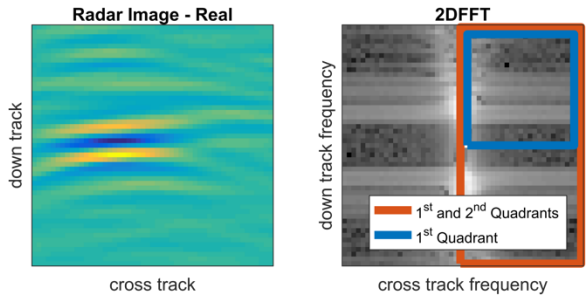



**Figure 17: 2D FFT estimation over a threat example.The left image is the real component of the complex radar image for a known threat location. The right image is the full 2D FFT of the real radar image with the two quadrant 2D FFT outlined in red and the one quadrant 2D FFT outlined in blue.**

To examine our hypothesis, we compare the performance achieved using the dual quadrant FFT feature to that obtained using the single quadrant. Both features are processed with the PLSDA classifier and resulting ROC curves are shown in Figure 18. The results clearly show that using the information contained in the additional quadrant leads to improvements in classification performance.

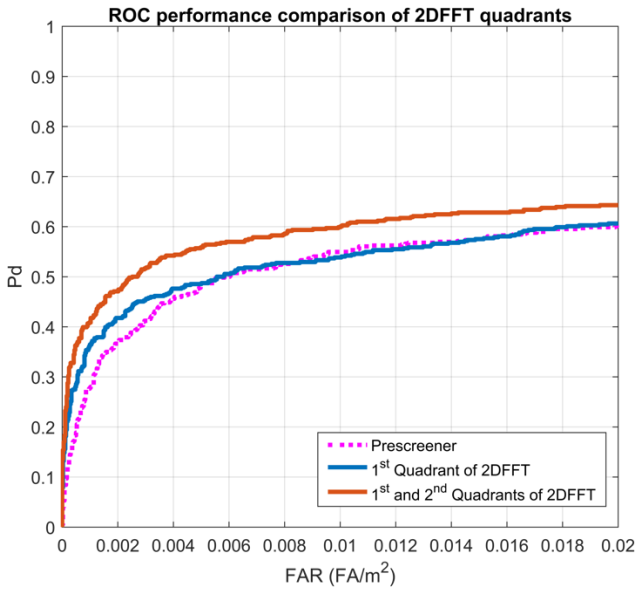



**Figure 18: ROC performance of the prescreener (pink dotted), one quadrant 2D FFT feature (blue) and two quadrant 2D FFT feature (red).**

## 5.3 *Modification 2: feature extraction on the real and magnitude FLGPR imagery*

The previous 2D FFT feature, described in section 5.1, is calculated from the real component of the complex radar image. In this section, we consider extracting features from the 2D Fourier transform of two different candidate components of the complex FLGPR images: the real component, and the magnitude of the complex image (or magnitude component). We also consider some combinations of the Fourier features extracted from these two images. The magnitude component is considered here because this is the image component used in many other non-2D FFT based features (e.g., HOG) and provides the shape of the energy response received at a given location. Figure 19

illustrates the extraction of the 2D FFT for the real component and the magnitude component images. Note that, based on the results in section 5.2, each feature set considered here includes two quadrants of the 2D Fourier transform. For many buried threats there is a characteristic oscillating response seen in the real component of the complex radar image; this effect is represented well in the 2D frequency domain [66]. The response over buried threats in the magnitude images is an energy measure across that spatial extent. A 2D FFT feature for the magnitude image represents the shape of the energy response while allowing for large amounts of spatial variance.

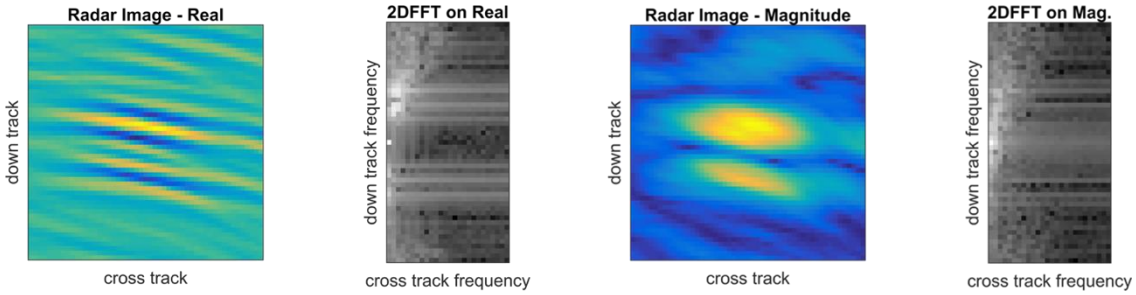



**Figure 19: Example 2D FFT estimation over the real and magnitude components of the complex radar image.The left two images are the real component of the radar image and the positive cross track spectrum from the 2D FFT computed from the real component. The right two images are the magnitude radar image and the positive cross track spectrum from the 2D FFT computed from the magnitude image.**

Detection performance results comparing the 2D FFT feature extracted over different image components are presented in Figure 20. The ROC curve for the real component 2D FFT feature is overall lower than the ROC curve for the magnitude of the 2D FFT and thus results in a lower pAUC. Further, the performance obtained when

concatenating the two 2D FFTs together results in an additional improvement in pAUC, although this difference is less pronounced when considering the ROC curves.

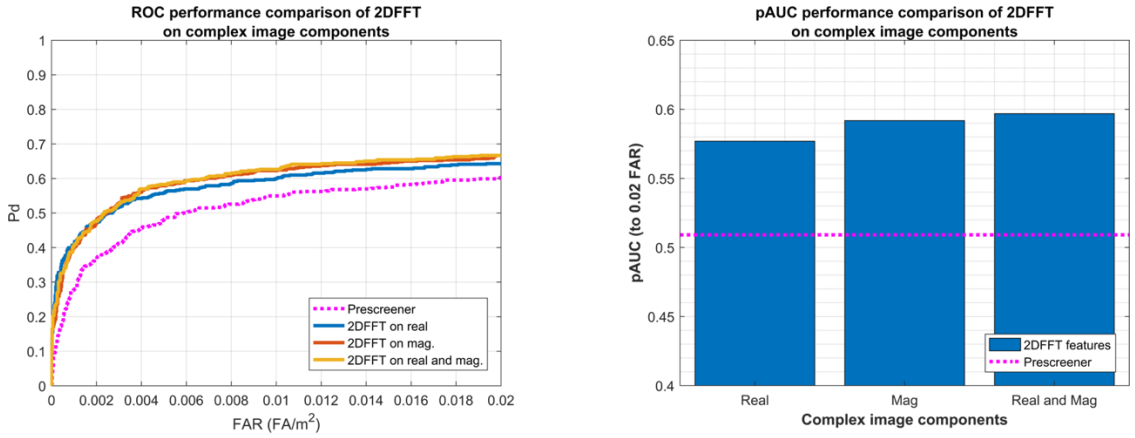



**Figure 20: Detection performance results for the 2D FFT feature computed using different complex image components.In the left plot the ROC curves for each complex component tested and prescreener are reported. In the right plot, the pAUC is displayed for each of the ROCs to show the slight classification advantage real and magnitude concatenated together achieve over just the magnitude 2D FFT feature.**

## 5.4 *The 2DFFT+ feature*

From the previous sections, there were two modifications to the 2D FFT feature that led to increased detection performance. The performance results from the first modification experiment suggested that including the 1st and 2nd quadrants from the computed 2D FFT is a better approach than only including the 1st quadrant in the feature vector. The second modification made use of the complex magnitude image, the component used in many other previous approaches, and is a feature that provides the

overall shape of the energy response. Including the 2D FFT feature applied to both the real and magnitude components resulted in the best detection performance as measured by pAUC. For the remainder of this thesis the concatenated 1st and 2nd quadrant 2D FFT feature computed for both the real and magnitude images will be used, and be referred to as the *2DFFT+ feature*.

## 5.5 *Reducing the dimensionality of the 2DFFT+ feature*

In this section, we present two data-driven methods for dimensionality reduction: principal components analysis (PCA), and the partial least-squares (PLS) algorithm. After describing each method, classification results for various levels of dimensionality reduction are presented. Dimensionality reduction can help in a variety of aspects, specifically: reducing the complexity of the function learned by a classifier, saving computational time and storage space. These improvements help with an ultimate goal of providing a small number of meaningful features to a system which can fuse across remote sensing modalities.

### 5.5.1 Principal component analysis (PCA) and Partial least-squares (PLS) for dimensionality reduction

Some experiments detailed below utilize feature dimensionality reduction as part of an algorithm for detecting buried threats. The two data-driven approaches that are applied are described in this section. Principal components analysis [67] (PCA) and partial least-squares [64] (PLS) are both methods that can be used for representing data in a lower

dimensional subspace. For both methods, the new representation is a linear subspace, which means that each new dimension in the new representation consists of a weighted linear combination of all of the dimensions in the original signal representation.

PCA is a commonly used tool for dimensionality reduction and lossy data compression. PCA is the orthogonal projection to a lower dimensional linear space which contains the maximum amount of dataset variance [25]. This projection results in linearly uncorrelated (or orthogonal) features known as *principal components*. These principal components are defined such that the first contains the largest possible feature variance in a linear dimension and the subsequent components contain the largest possible variance under the orthogonality constraint. The dimensionality reduction is introduced by retaining a smaller number of these principal components compared to the original feature dimension. A key assumption made when applying PCA to a dataset is that there is a high signal to noise ratio and that the major dimensions of variance are related to the dimensions useful in classification [67]. In the simulated example shown in Figure 21 the PCA 1$^{st}$ component is shown to not be sufficient to capture the primary dimension of class separation.

PLS is similar to PCA, but its projections are computed to retain the most possible covariance between the labels and original features in each orthogonal component. In practice the PLS components often create a better space for classification in fewer number of components compared to PCA [64], [68], though requires a labeled training data set. In

Figure 21 the first component for both PCA and PLS are computed over a simulated dataset. Here, the first PCA component is not able to resolve the two classes as it is "corrupted" by the large dataset variance in the orthogonal direction. In contrast, the first PLS component takes advantage of the classification labels to project to a first component meaningful for discriminating the two classes.

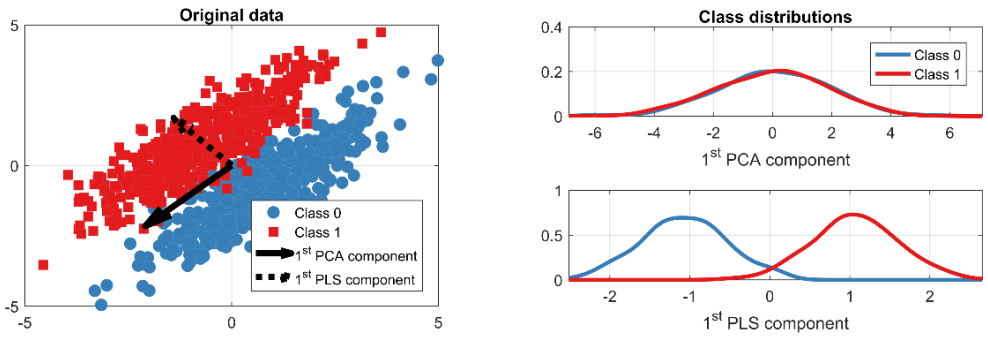



**Figure 21: Example with simulated data of the 1st component projections of PCA and PLS.The left plot shows simulated data sampled from two underlying classes in their original space. The arrows on this plot are the computed 1st components for PCA (solid line) and PLS (dotted line). The right two plots show the class distributions of the data after being projected onto those 1st components of PCA (top right) and PLS (bottom right). The PCA projections, which are only computed based on the auto-covariance of the data, and not the lables, does not create a one-dimensional space well suited for classification. However, the 1st PLS component produces a space in the most prominent data to label covariance direction, a space more useful in classification.**

### 5.5.2 Classification results with dimensionality reduction

The effectiveness of the PCA and PLS reduction techniques in terms of detection performance were tested over a range of the number of components retained. The resulting feature vectors from each method were used with the PLSDA classifier in lane-

based cross-validation to evaluate discriminability of the reduced feature sets. In Figure 22 the pAUC results are shown for each method using the 2DFFT+ feature. The PLS algorithm produces a feature space that preserves class separation using only two PLS components; in contrast, more than six PCA components are needed to achieve similar detection performance. As noted in other application settings, when reducing dimensionality for increasing detection performance it is often advantageous to employ PLS over PCA[64], which is corroborated by these results.

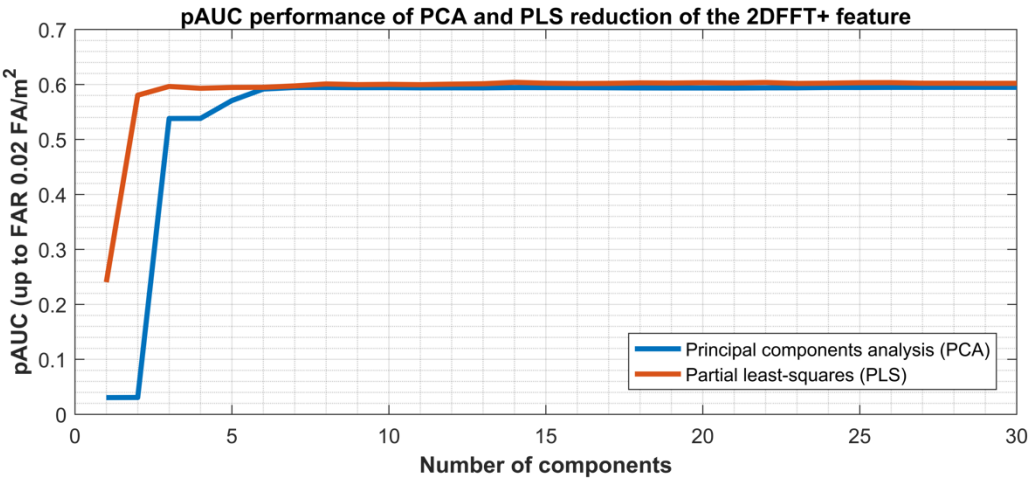



**Figure 22: pAUC performance as a function of the number of components retained for detection with PLS, in red, and PCA, in blue, when applied to the 2DFFT+ feature.**

## 5.6 *2DFFT+ performance comparison to other feature sets*

In this section we compare the detection performance for the 2DFFT+ feature with a variety of previous feature sets from the FLGPR buried threat detection literature. The

features used here to compare against were all included in the comparison described in Chapter 3. Each of these features are extracted on a consistent set of radar images created at each of the prescreener alarm locations, and used with the PLSDA classifier in the same lane-based cross-validation. Also included in this comparison is a reduced 2DFFT+ feature. For the dimensionality reduction, the number of retained PLS components was chosen with a random 5-fold cross-validation trial inside each training fold.

The pAUC results of the feature comparison are presented in Figure 23. The previous feature set results with the PLSDA classifier are shown in the blue bars in the same order presented in the original comparison. The two methods from the work described in this chapter, the 2DFFT+ feature and the reduced 2DFFT+ feature, are shown as dashed lines instead of bars to aid comparison to the performance achieved with the other feature sets. The detection performance using the 2DFFT+ feature outperforms the detection performance achieved with each of the previously considered feature sets based on detection results measured by pAUC. A further performance increase is reported for the 2DFFT+ feature when applying the PLS feature reduction technique. Included in each feature set label is the dimensionality of the feature vector delivered to the classifier. For the 2DFFT+ feature reduced by PLS, the dimensionality is the average of the three lane-based cross-validation folds since the number of retained PLS components is chosen within each training fold. With an average of 20 retained components, this shows a dramatic reduction in feature dimensionality compared to the 2DFFT+ feature which has

a length of 2,652. This dimensionality reduction can be potentially helpful for fusing the FLGPR system with other remote sensing modalities for detecting buried targets.

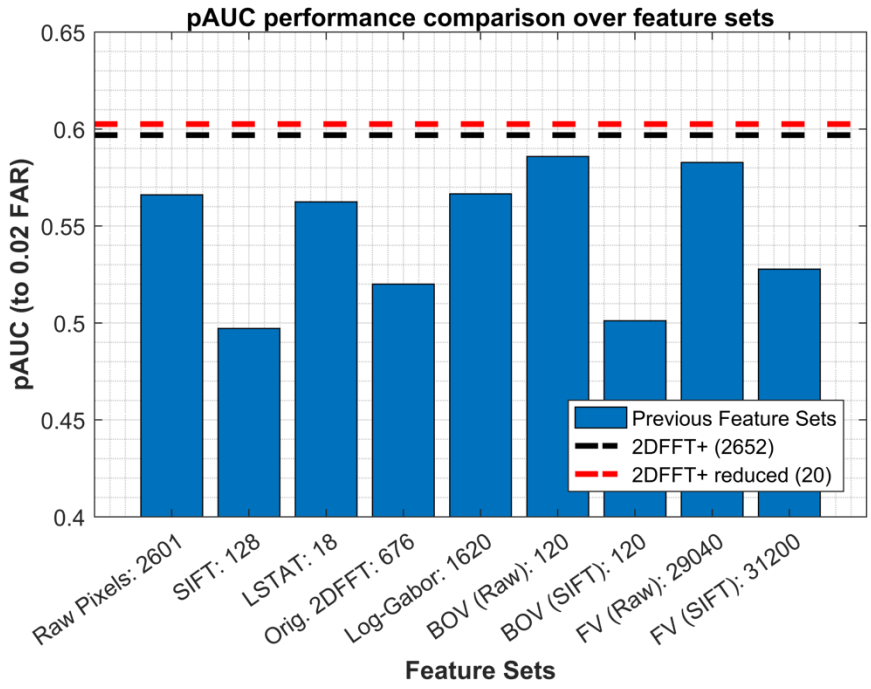



**Figure 23: pAUC performance comparison across a variety of existing FLPGR image features.In each feature label the dimensionality of that feature is reported. The 2DFFT+ and 2DFFT+ reduced features are reported as lines instead of bars to aid visual comparison to each previous feature set.**

## 5.7 *2DFFT+ feature conclusions*

This chapter considered the impact of several modifications to an existing 2DFFT feature, resulting in a new feature that we call the 2DFFT+ feature. Detection performance using the 2DFFT+ feature significantly increased over detection performance obtained by simply utilizing the original 2DFFT feature. The utilization of this feature resulted in a substantial increase in detection performance compared to performance achieved using a

large variety of existing FLGPR image features from the literature. Lastly, two data-dependent dimensionality reduction techniques were compared. The 2DFFT+ feature resulted in higher detection performance, measured in terms of pAUC, than detection performance achieved using any of the compared feature sets. Additional detection performance was obtained using PLS dimensionality reduction prior to applying the detection algorithm using the 2DFFT+ feature, and this dramatically reduced the number of dimensions that must be processed from 2652 to approximately 20.

The following specific conclusions can be drawn:

- Using two quadrants of the 2DFFT over both the real and magnitude radar images results in a detection performance improvement when compared to performance achieved with the original 2DFFT feature.
- The 2DFFT+ feature is the best single feature set compared to a large selection of existing FLGPR image features.
- Reducing the 2DFFT+ feature using PLS results in a further increase in performance.

# 6 Automatic FLGPR antenna calibration

This chapter contains the development and performance evaluation of an automatic FLGPR antenna calibration technique. The investigation into the radar imaging function was motivated by the overall simplicity of the features and classifiers that ultimately proved most useful for landmine detection in the feature comparison work described earlier. The radar imaging modifications introduced in this chapter are related to the weighting of the individual antenna returns prior to the summation that is then performed at given beamformed locations. It will be shown that learning an antenna weighting (sometimes referred to as an antenna window) using machine learning techniques instead of using standard beamforming weights results in images that can be utilized to improve the detection performance of the FLGPR system.

## 6.1 *Antenna weighting with the DAS method*

This section contains details about the FLGPR antenna configuration and an overview of the antenna weighting applied to the summation during the delay-and-sum (DAS) process. The primary focus of this work is a proposed method of automatically calibrating the antenna weightings to be applied to a previously unseen set of data. This section alters the conventional beamforming equation, (1), in order to apply a learned weighting on different selections of the antennas.

The antenna elements in the FLGPR system consist of a two-dimensional array; a cartoon diagram of the antenna array can be found in Figure 24. The 2D array is generally

separated into the two components [8], one across the sensor platform and one along the direction of platform motion, these are referred to as the Tx-Rx pairs and the apertures, respectively. In the synthetic aperture contexts, such as the one considered here, the apertures are not distinct sets of sensors but are the row of Tx-Rx pairs which have been moved forward as the platform advances down the roadway. This motion over time is what creates the full two-dimensional radar array used to beamform images.

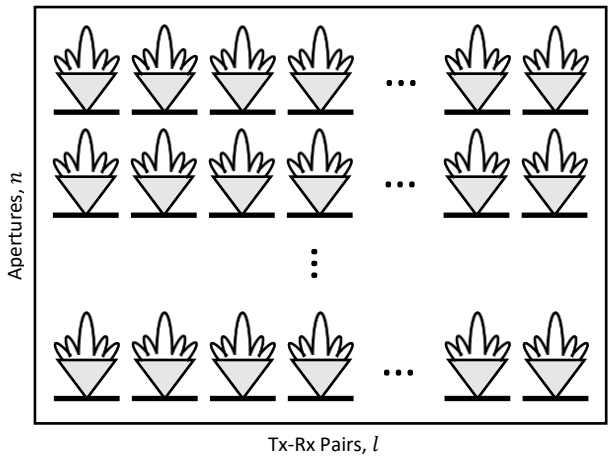



**Figure 24: Illustration of the FLGPR antenna array.The 2D array is decomposed into two primary components, the apertures, $n$, and the Tx-Rx pairs, $l$.**

In (20), expanded from (1) in Section 2.1.2, is the formula for estimating the radar reflection (i.e., image pixel value) at various places in a spatial extent in front of the sensor platform. Instead of containing all antenna elements in one summation, this created one for each dimension of the array. The imaging spatial extent is defined by the locations of the pixels, $p$, in a given image, $I$. Estimates at any location are the coherent sum of the corresponding radar returns, $x_{ln}$, from each antenna in the 2D array. The radar returns

are indexed by the distance, $R_{lnp}$, which is the radio wave travel time between the current antenna and pixel locations. The double summation is then performed across all apertures, $n$, and Tx-Rx pairs, $l$.

$$I_p = \sum_{n=0}^{N-1} \left[ \sum_{l=0}^{L-1} \alpha_{ln} x_{ln}\left(R_{lnp}\right) \right], \tag{20}$$

This equation can be updated to include a weighting on each sum, $\alpha_n$ for the apertures and $\alpha_l$ for the Tx-Rx pairs as shown in (21). This is a modification predicated on the assumption that the array element dimensions are independent, thus allowing antenna weighting functions which are not conditioned on the other. This assumption creates an easier learning task for both the aperture and Tx-Rx pair weightings; these are the weighting vectors which are investigated in the following sections.

$$I_p = \sum_{n=0}^{N-1} \alpha_n \left[ \sum_{l=0}^{L-1} \alpha_l x_{ln}\left(R_{lnp}\right) \right], \tag{21}$$

### 6.1.1 Conventional antenna weighting

When performing conventional beamforming for the FLGPR system the aperture weighting is uniform (i.e., $\alpha_n = 1^{1\times N}$). For the Tx-Rx pairs, although not noted in the FLGPR literature, it is common practice to apply a "power window" to this dimension of antennas. The "power window" definition is found in [69] and restated in (22). For this application, the window, $\alpha_{\text{power}}$, is designed to be $-3$dB at $0.8 * D$ and $-10dB$ at $D$, where $D$ is the width of the FLGPR sensor platform (i.e., length in meters along the Tx-Rx pair

direction). This window specification was originally created by the sensor manufacture and used without mention as a part of the conventional process in many FLGPR studies [13], [37], [63], [70]. Those specifications correspond to values of $\eta = 1.15$ and $\zeta = 5.4$. Below, $l_y$ denotes the cross track location of the Tx-Rx pair.

$$\alpha_{\text{power}} = \exp\left(\eta |l_y|^{\zeta}\right), \tag{22}$$

## 6.2 *Aperture weighting*

This section introduces and contains experimental results for a learned antenna weighting on the aperture dimension in the 2D radar array. As a way to isolate the conditions of the learning task, the Tx-Rx pairs are summed according to the conventional DAS procedure, while the individual aperture components are used to learn a window for that dimension of the array. This learned *aperture weighting* is then applied to each observation. Lastly, this procedure is tested in cross-validation with image detection algorithms to evaluate the effect on FLGPR target classification.

### 6.2.1 Computation and application of an aperture weighting

Starting with (22), the weighting on the Tx-Rx pairs is chosen to be the conventional weighting (i.e., $\alpha_l = \alpha_{\text{power}}$) and rewritten in terms of linear algebra in (23). Here $\boldsymbol{I}$ is a $1 \times P$ vector of the rasterized radar image with $P$ total pixels. The radar returns, summed across the windowed Tx-Rx pairs, for each aperture and pixel is denoted by $\boldsymbol{X_n}$

and has the shape $N \times P$, with $N$ being the total number of apertures. Rows in $\boldsymbol{X_n}$ can be conceptualized as individual aperture images prior to the summation. The desired weighting, $\boldsymbol{\alpha_n}$, is thus a $1 \times N$ vector and defines the weighted sum across apertures.

$$\boldsymbol{I} = \boldsymbol{\alpha_n X_n} \tag{23}$$

Without knowing the "true" image, $\boldsymbol{I}$, for any location being inspected, to learn $\boldsymbol{\alpha_n}$, an $\boldsymbol{I'}$ is substituted which promotes energy in the center of the image. By using a desired substitution, $\boldsymbol{I'}$, the learned $\boldsymbol{\alpha_n}$ will tend to put more weight on the aperture images which have greater centered energy to background ratios. This assumption of how a "true" target is expected to appear is informed by the practitioners' knowledge of the conditions of the current sensor deployment context. The target image, $\boldsymbol{I'}$, in all experiments described below is a 2D Gaussian function with the standard deviation set to a constant to match the average size of a target response. This 2D Gaussian is a smooth approximation of what is expected to be seen from the size and type of targets present in the current FLGPR dataset.

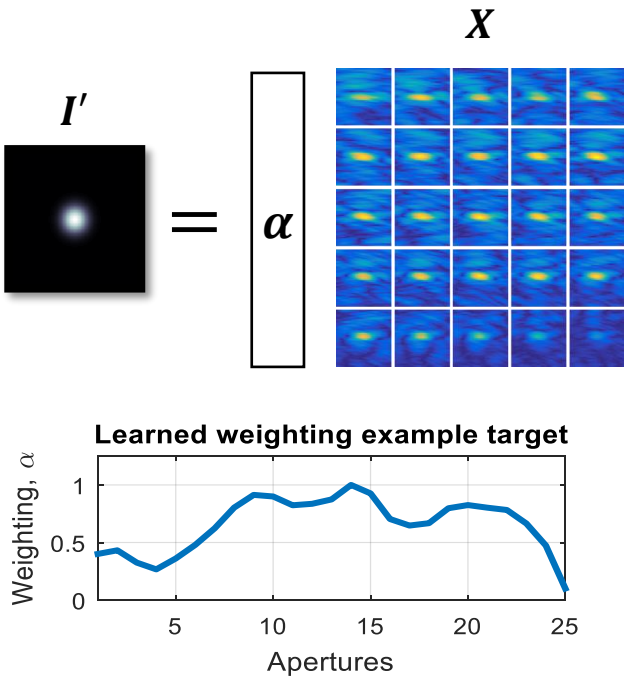



**Figure 25: (a) Illustration of the learning task for finding an aperture weighting, $\alpha_n$, to maximize the energy in the center of the resulting image.The left most image is an approximation of the idealized radar response which we set equal to a weighted combination of the individual aperture images. The weight vector is represented by the box labeled $\alpha$ and the individual aperture images are shown on the right in a 5 by 5 grid for the 25 individual apertures. (b) Learned aperture weighting using PLSR on the example presented above. This is the weighting value across the number of FLGPR apertures. Notice the higher weighting on the center few apertures, this corresponds to the brighter individual aperture images from above.**

To further illustrate this learning task, a diagram is provided in Figure 25(a). From this diagram, the desired image, $\boldsymbol{I'}$, is the left most image and represents the 2D Gaussian function. The right most set of images are the individual aperture images which are to be combined in the weighed summation according to the values in the $\boldsymbol{\alpha_n}$ vector. As indicated by this diagram, the center to background energy ratios change throughout the various apertures and further suggest how a non-uniform weighting can potentially take

advantage of this phenomenon.

The learning of an $\boldsymbol{\alpha_n}$ for an individual image was completed using partial least squares regression (PLSR) [42]. PLSR is a common linear regression technique that can generally operate well with multicollinear features in the data, a trait present between the pixels in the radar images. For the PLSR model, the data, $\boldsymbol{X_n}$, and target, $\boldsymbol{I'}$ spaces are set up as described in (24), (25), and (26). For $\boldsymbol{X_n}$, $\boldsymbol{T}$ is a projection of the space $\boldsymbol{P}$; this is similar for $\boldsymbol{I'}$ with $\boldsymbol{U}$ and $\boldsymbol{Q}$, respectively. The column spaces $\boldsymbol{P}$ and $\boldsymbol{Q}$ are restricted to have $c$ components, a value chosen during implementation, and selected to be $c = 1$ for all experiments in this work because the individual features in $\boldsymbol{X_n}$ are expected to be colinearly correlated in one dimension. Further, the error terms $\boldsymbol{E}$ and $\boldsymbol{F}$ are assumed to be independent and identically distributed.

$$\boldsymbol{X_n} = \boldsymbol{T}\boldsymbol{P}^{\mathrm{T}} + \boldsymbol{E} \tag{24}$$

$$\boldsymbol{I'} = \boldsymbol{U}\boldsymbol{Q}^{\mathrm{T}} + \boldsymbol{F} \tag{25}$$

$$\boldsymbol{U} = \boldsymbol{T}\boldsymbol{D} \tag{26}$$

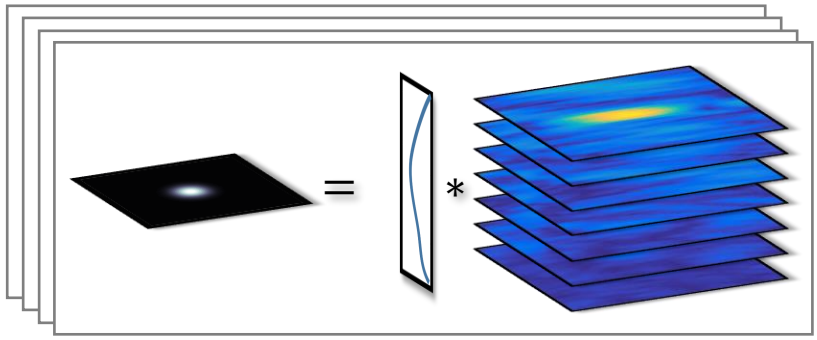
$$\boldsymbol{\alpha_n} = \boldsymbol{T}\boldsymbol{D}\boldsymbol{Q}^{\mathrm{T}} \tag{27}$$

The objective of PLSR is to maximize the covariance of the data and target projections, in other words maximize $\mathrm{cov}(\boldsymbol{U}, \boldsymbol{T})$; a complete solution for this is presented in [71]. To provide an intuitive impression for the current application, a situation with one column for the target variable $\boldsymbol{I'}$ and constrained to have a single component, $c = 1$, the

result of the weight vector, $\boldsymbol{\alpha_n}$, is directly proportional to the first eigenvector of $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{I}'$.

This regression procedure is completed for each observation. Figure 25(b) shows the learned weightings from the example discussed above. The center weights correspond to the individual aperture images which have the most prominent target responses. This example is representative of the learned weightings for many strong response areas found in this data. It is not necessary for these strong responses to correspond to labeled target data. For instance, clutter objects which result in large radar responses can be used in this calibration scheme to learn the aperture weightings. Many of the objects, target or non-target are generally the highest confidence alarms returned by the prescreener. A collection of the learned aperture weightings across a selection of observations is shown in Figure 26. While the individual regression weights are somewhat noisy, an average of the learned weights, shown as the solid black line, highlights the overall trend. This trend matches with the intuition that the targeted area comes in and then out of "view" as the platform advances down the roadway. Using the average aperture weightings, radar images are then created; an example is shown in Figure 27. The image from this figure illustrates the dampening of background energy compared to the centered response when the learned aperture weighting is applied.

Antenna weighting computation repeated across a set of training observations.



The average learned antenna weighting is then applied to the unseen observations during testing.

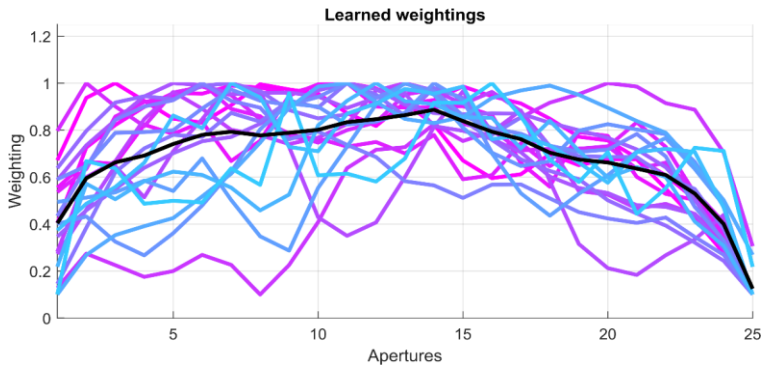



**Figure 26: A collection of individually learned aperture weightingsshown in color, with the average shown in black. The x-axis here is the aperture index, 1 being closest to the area under inspection and 25 being furthest.**

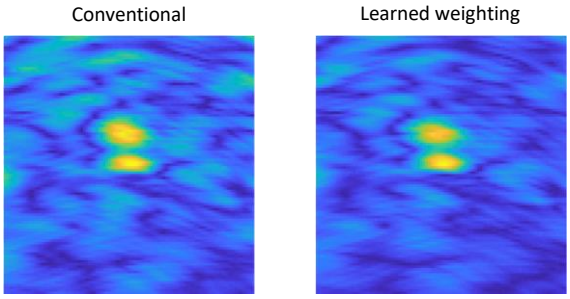



**Figure 27: Example of beamformed images from a target location for comparing the conventional imaging techniqueon the left and the learned aperture weighting on the right.**

### 6.2.2 Aperture weighting classification results

To test the effectiveness of the learned aperture weighting, the resulting images

are used with the most effective feature and classifier combination found in previous chapter on feature comparison (see Chapter 3 and 5), and this is the 2DFFT+ feature with the PLSDA classifier. For the first set of tests, the type of observations and quality of observations used to learn the new aperture window is varied. For each observation, a learned aperture weighting is computed using the method described in the previous section. With a collection of learned weightings from multiple observations, an average learned weighting can be computed which is then applied to the yet unseen set of observations as the aperture weighting. The collection observations used to compute this average was varied in type and quality to test what dataset characteristics are needed to successfully implement this automatic calibration method. Three types of observations were tested: targets, non-targets, and both types together. Within each type, the quality of observation was varied as sorted by its prescreener confidence; the top 20%, top 50%, and all available alarms were tested as different methods for learning the aperture window. Figure 28 shows the learned weighting across each combination of the type and quality of the observations averaged together.

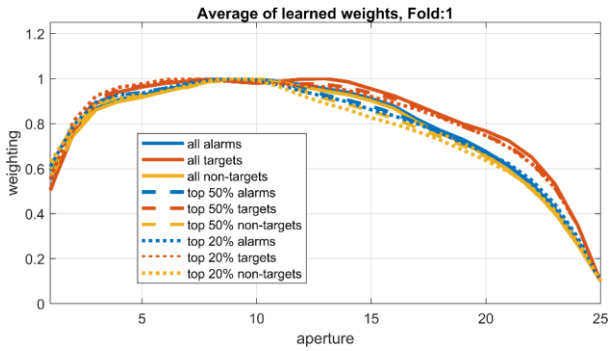



**Figure 28: Learned average aperture window across type and quality of observation.The consistent shape of the learned weighting across all combinations evaluated suggests a robust learning method.**

The pAUCs resulting from the 2DFFT+ feature and PLSDA classifier on each of the image sets generated by each set of weights, including the baseline, is presented in Figure 29. These results are completed using lane-based cross-validation; the weightings are learned using the training set only and have no dependence on the particular test set. As expected from visualizing the different averaged windows, the performance is stable over all types and qualities of observations, with a slight performance increase when including known target observations only. These results suggest that the learning method does not require labeled or even selected "high quality" responses to learn an effective aperture window. The robustness of the results coupled with an increase over the baseline performance is clearly illustrated. A final ROC is shown in Figure 30 that compares the conventional beamforming to the aperture window averaged from all observations. The results indicate that the learned aperture weighting is effective in increasing buried-target detection performance for the FLGPR system.

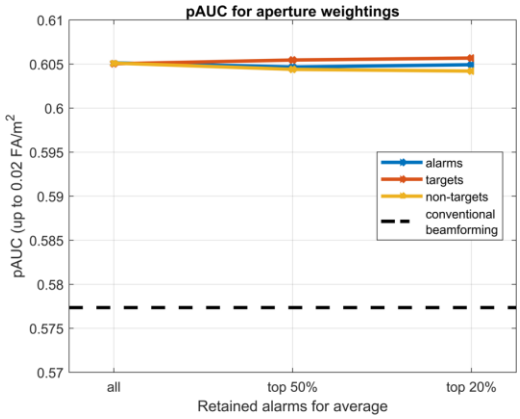



**Figure 29: pAUC for the 2DFFT+ with PLSDA on the images sets created with different aperture weighting averages.The aperture weighting average was computed with various quality of observations as measured by the prescreener confidence and different observations' label categories. The stability of the performance suggests the robustness of the learned aperture method and its improvement over the conventional beamforming window.**

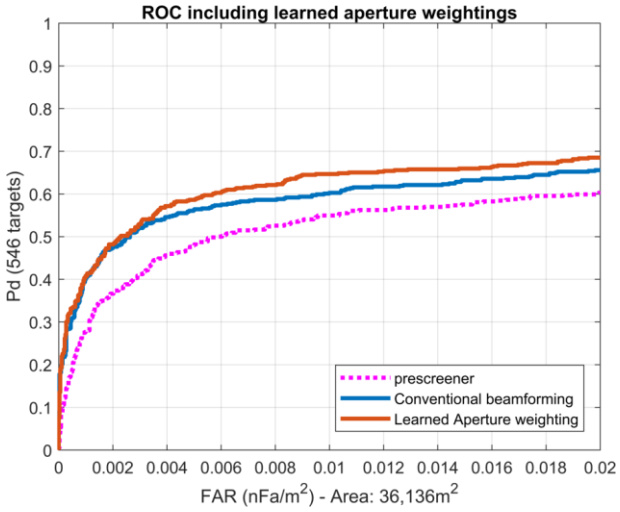



**Figure 30: ROC for the learned aperture weightings trained with all observations and processed in lane-based cross-validation.This figure indicates that there is an increase**

**in detection performance using the learned aperture weighting.**

## 6.3 *Tx-Rx pair weighting*

This section introduces and evaluates the performance of learned windows on the Tx-Rx pairs. Conventionally, there is a predefined static window applied across the Tx-Rx pairs (see Section 6); the following work produces a learned Tx-Rx pair weighting which is instead applied during radar image creation. The method for learning and applying the Tx-Rx pair weighting follows the method developed in the previous section for the learned aperture weighting.

Starting with (22), due to the assumed independence condition of the antenna dimensions the summation order can be switched so the apertures are summed inside each Tx-Rx pair. This simplifies the Tx-Rx pair weighting learning task and creates a problem which mirrors the above aperture weighting learning procedure. The summation reordering is reflected in (28) and formulated into a linear algebra context in (29). The aperture weighting is fixed as $\alpha_n = 1^{1\times N}$ to reflect the conventional beamforming method and allows for independent evaluation of the various antenna weightings.

$$I_p = \sum_{l=0}^{L-1} \alpha_l \left[ \sum_{n=0}^{N-1} \alpha_n x_{ln}\left(R_{lnp}\right) \right], \tag{28}$$

$$\boldsymbol{I} = \boldsymbol{\alpha_l X_l} \tag{29}$$

With an identical method as the aperture weightings, the Tx-Rx pair weightings are learned with PLSR to fit the same 2D Gaussian function. Example learned Tx-Rx pair

weightings are shown in Figure 31, along with their average learned response shown in black. There are consistent patterns which emerge from the learned weightings, and certain Tx-Rx pairs have consistently low weightings from most observations. This implies that there are Tx-Rx pairs which are consistently not useful in promoting the center energy in the resulting radar images. Also, note the overall trend of higher weightings in the center of the array with less weight being learned towards the outside of the array.

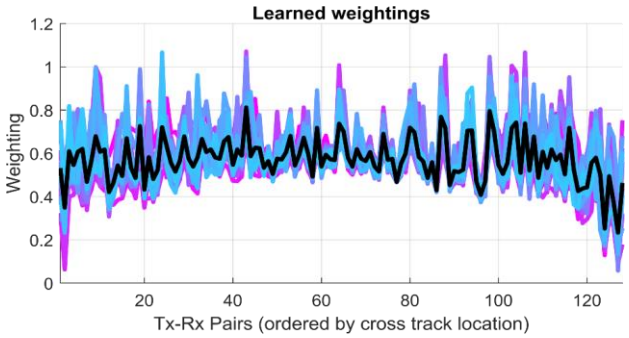



**Figure 31: Multiple examples of learned Tx-Rx pair weightings.The average is also plotted as the solid black line. Notice the consistent pattern over many of the learned responses, this implies that some antennas are not as useful in accentuating the target-like radar responses. There is also an overall trend of higher weighting in the center of the array and less weight towards the outside elements of the array.**

### 6.3.1 Weighting as a function of pixel cross track location

A secondary trend in Tx-Rx pair learned weightings becomes apparent when ordering by observation cross track location. The weightings learned on a particular side of the Tx-Rx pair array are greater when the observation is on that cross track side of the lane. This is intuitively satisfying in that a target on the left side of the vehicle will return

greater responses to the left side of the Tx-Rx pairs (i.e., the left side receives a better "look"). Figure 32(a) shows a number of high quality observation's Tx-Rx pair learned weightings sorted by observation cross track location and by Tx-Rx pair cross track location for the horizontal and vertical axis, respectively.

A modification of the Tx-Rx pair summation equation is shown in (30), this allows the antenna weightings to also depend on the current pixel cross track location, $p_y$. This equation reflects the phenomenon observed in the sorted visualization of learned Tx-Rx pair weightings from Figure 32(a).

$$I_p = \sum_{l=0}^{L-1} \alpha_{lp_y} \left[ \sum_{n=0}^{N-1} x_{ln}\left(R_{lnp}\right) \right], \tag{30}$$

To learn the Tx-Rx pair weighting as a function of pixel cross track location, PLSR is once again employed. First, the overall Tx-Rx pair average found in the previous section is used to normalize the individual learned weightings. Since the 2D function being estimated appears to be non-linear, a kernel is applied to the antenna weight values before use with PLSR. This kernel computes the second order interaction terms of each feature in the data matrix, which allows the surface to be non-linear in the original feature space. This is formalized in (31). $S\left(p_y, l_y\right)$ here is the surface function which depends on the cross track, $y$, locations of the current observation and the Tx-Rx pairs. This regression is performed on $\hat{S}$ which is compiled from the $\alpha_l$ learned in the previous section.

$$S(p_y, l_y) = \beta \left[ l_y, p_y, p_y l_y, l_y^2, p_y^2 \right] \tag{31}$$

With the learned $\beta$ a new observation is given a unique weighting depending on its cross track location. See Figure 32(b) for an example of the learned surface. For the creation of the images with this cross track dependent weighting, the Tx-Rx pairs are also weighted with the overall Tx-Rx pair weighting from the previous section, which was normalized out prior to this process.

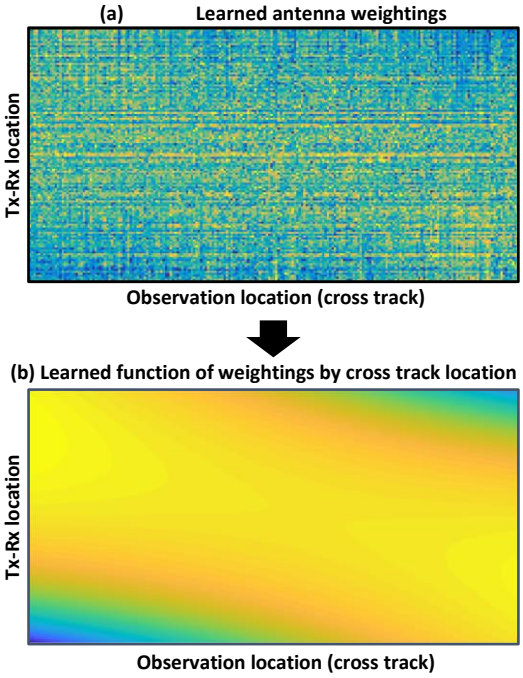



**Figure 32: This figure illustrates the learning of the Tx-Rx pair weighting as a function of the observation location.(a) shows the learned individual observation weights sorted by the observation cross track location (x-axis) and Tx-Rx pair cross track location (y-axis). (b) shows the learned surface of the underlying antenna weightings as a function of observation and Tx-Rx pair cross track location.**

## 6.4 *Tx-Rx pair weighting, classification results*

The following results illustrate the performance of the different Tx-Rx pair

weighting methods. For the ROC curves in Figure 33, each beamformed set of images is used with the 2DFFT+ feature and PLSDA classifier. The learned antenna weightings are computed in lane-based cross-validation. The overall Tx-Rx pair antenna weighting improves over the conventional window and additional performance is added when applying a cross track dependent weighting.

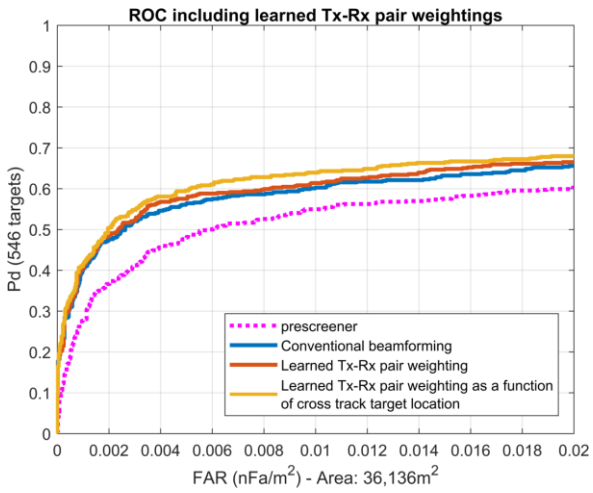



**Figure 33: This set of ROC curves shows the performance for three antenna weighting strategies: conventional, a learned global Tx-Rx pair weighing, and a learned Tx-Rx pair weighting as a function of cross track image location. The feature extracted on these three different image sets is the 2DFFT+ and is used with the PLSDA classifier. The global Tx-Rx pair weighting improves the detection performance compared to the conventional weighting. Further, when using a Tx-Rx pair weighting which is dependent on the image view cross track location, the performance is again ncreased.**

## 6.5 *Combined antenna weighting*

The experiments in this section are designed to compare two methods from combining the learned antenna weightings from the aperture and the Tx-Rx pairs. The first method relies on the independently learned aperture and Tx-Rx pair weighting. For

the second method, the antenna weightings are learned together in a single operation (i.e., mutually) by combining the double summation of the 2D array.

### 6.5.1 Independently learned antenna weightings

The independence assumption for the apertures and Tx-Rx pairs results in two separate antenna windows over each array dimension. For the experiments in this section, the independently learned antenna weightings are multiplied together (i.e., substituted in the double summation from (22)) to generate a unique weight for each antenna array element. This is completed for both the overall Tx-Rx pair weighting and the Tx-Rx pair weighting which is dependent on the pixel cross track location.

### 6.5.2 Mutually learned antenna weightings

For the mutually learned antenna weighting the double summation across the two array dimensions is reduce to a single summation. Recall (1) from above, where $\alpha_m$ is the weighting on the $m^{\text{th}}$ antenna and corresponds to one Tx-Rx pair at a particular aperture. This learning task is more difficult than the individual aperture and Tx-Rx pair weightings as it attempts to infer $N \times L$ parameters as opposed to just $N + L$, but could potentially reveal more complicated antenna relationships. The weighting here is learned in the same manner as the above methods with PLSR.

### 6.5.3 Results using all antenna weightings

Results from each combined antenna learning method are presented in Figure 34. This shows the ROC curve for the conventional weightings, the independently learned

aperture and Tx-RX pair weightings ($\alpha_n * \alpha_l$), the cross track dependent independently learned weightings ($\alpha_n * \alpha_{lp_y}$), and the mutually learned weightings $\alpha_m$. The feature used in this comparison is the 2DFFT+ with the PLSDA classifier. Here the best performing method is the independently learned antenna weightings with the Tx-Rx pair dependence pixel cross track location. The mutually learned antenna weightings, while improving upon the conventional beamforming do not match the detection performance of either independently learned weightings. Also, presented in Figure 34 is a comparison of the combined antenna weightings and the conventional beamforming with the application of the previously discussed feature sets. These are all used with the PLSDA classifier and the combined antenna weighting represented here is the independently learned weightings which use the Tx-Rx pair window depending on the observation cross track.

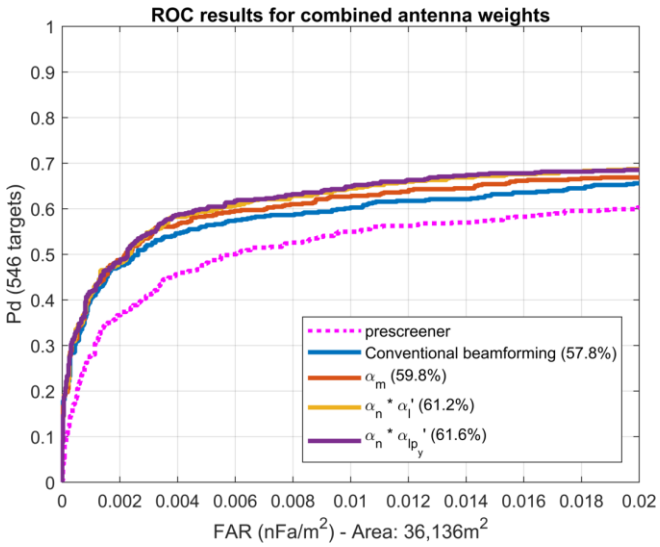



**Figure 34: ROCs (and pAUC) for the combined antenna weighting methods.Presented here are the conventional weightings, the independently learned aperture and Tx-Rx pair weightings ($\alpha_n * \alpha_l$), the cross track dependent independently learned weightings ($\alpha_n * \alpha_{lp_y}$), and the mutually learned weightings $\alpha_m$. The best performance was achieved when combining the independently learned aperture weights and cross-track dependent Tx-Rx pair weightings.**

### 6.5.4 Robustness across detection algorithms

As a way of testing the robustness of the automatic calibration, another experiment was performed in which many detection algorithms were compared. By varying the features extracted on top of the resulting imagery the stability of the patterns in the images is tested. Figure 35 shows the pAUC results from a lane-based cross-validation experiment with a variety of common FLGPR image features [63]. For the pAUC performance to consistently increase for each of the features considered, the learned antenna weightings were mostly likely resulting in a truly better radar image formation for classification.

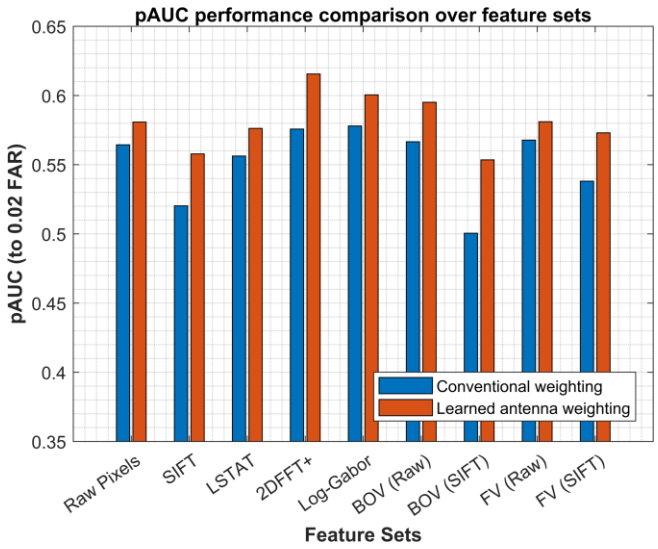



**Figure 35: pAUC comparison across all previously compared image features processed with the PLSDA classifier.The two underlying image formation techniques compared here are the conventional weighting and the combined aperture and Tx-Rx pair. The Tx-Rx pair weighting used here is the method which is learned based on the observation cross track location. For every feature considered, the learned antenna weighting increases the overall detection performance, which suggests that underlying imagery is being robustly improved by the new antenna learning method.**

## 6.6 *Conclusions*

This chapter introduces and provides detection performance results for methods of automatic calibration for the radar antenna array. The summation from the conventional beamforming equations were modified to include a learnable weighting for both the aperture and Tx-Rx pair array dimensions. Each individual learned weighting improved upon the conventional beamforming method and when combined showed even better detection performance. The Tx-Rx pair weighting was further improved by the inclusion of the cross track "look" direction and allowed for the antenna weighting to change depending on the side of the sensor array where ground was being inspected. It

is suggested by the results of the mutually learned antenna weightings that by splitting the regression problem into multiple smaller problems a better weighting for target discrimination is learned. Additionally, it was found that the method for learning and averaging the antenna weightings did not depend on the use of labeled target observations, a desirable trait in other applications for where labeled data may not be common.

In summary, the following broader conclusions can be drawn from this work:

- Using the combined antenna weightings which were learned independently across the 2D array leads to state-of-the-art detection performance.
- Learned antenna weightings benefit from being dependent on the cross track observation location
- Automatic calibration can be inferred well without ground labels.

# 7 FLGPR volume based detection

The 2-dimensional beamformed images represent a rough estimate of the radar signal reflected from the subsurface (i) across some spatial region, and (ii) at a particular depth under the ground. The images therefore account for variations in the spatial location of buried threats, but they assume that all threats exist at a single burial depth because images are beamformed at a single depth. This neglects the likely variations in the burial depth of threats, and also the likely variations in the contours of the ground that exist with respect to the sensors. We therefore hypothesize that this "single depth" assumption causes the beamforming to distort, or otherwise discard, important signal information, creating an information bottleneck that cannot be addressed by better classifiers or features.

In this chapter, we examine this hypothesis by beamforming multiple FLGPR images, each at a separate depth, and stack them to form 3-dimensional (3D) volumes. Figure 36 shows an example of this stacking process. We apply several popular FLGPR detection algorithms to both the 3D volumes, and conventional 2D images, and compare their performance on a large collection of FLGPR data. The results indicate that the 3D detectors consistently outperform their 2D counterparts. This suggests that the use of a single depth for beamformed images is in fact a potential performance bottleneck for detection in FLGPR data. Our use of 3D information is also relatively crude, and therefore presents only a conservative estimate of the performance improvements that may be

possible by accounting for multiple depths. In this chapter, we use prescreener configuration B (see Section 2.1.3.2,) and the corresponding dataset (see Section 2.4.2).

This chapter is organized as follows: Section 7.1 introduces the procedure for creating images over a range of depths. Section 7.2 describes the extensions of the image detectionfeatures from the 2D images to the 3D radar volumes for use in the classification experiment. In Section 7.3, the classification results and analysis are provided. In Section 7.4, conclusions are drawn for this FLGPR volume based detection and potential future work is outlined.

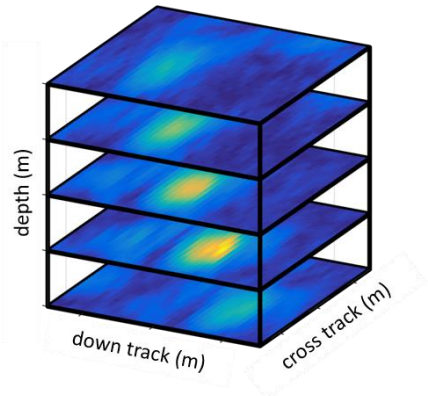



**Figure 36: Radar images over a known threat location, created by varying the focus depth when beamforming.**

## 7.1 *Beamforming in three-dimensions*

This section introduces the concept of beamforming images over multiple depths to create a radar volume. Recall that beamforming is the process of estimating the radar response for a spatial extent in front of the sensor. Traditionally, for the FLGPR system

has been performed in flat, 2D images at a single elevation (i.e., focus depth). For these experiments we consider back-projecting the radar responses at multiple depths which are organized as a *3D radar volume,* a concept introduced in [72] but not utilized for threat detection or classification. These radar volumes are created by beamforming 2D radar images at multiple focus depths and concatenating across the depth dimension. Examples are shown in Figure 37, where there are two known threat locations showing a 3D radar volume (left column) and a traditional 2D radar image (right column). For the visualizations, slices of this 3D volume are shown, as it is impossible to visualize the whole volume in this way. The traditional 2D image is shown as a flat plane in a 3D view for comparison to the 3D volume. For each threat example, the 3D volume is able to capture a higher reflected energy location as well as more of the reflection structure. In this work, at each alarm location we extract 3D volume and 2D images over the *down track* and *cross track* area consistent with many previous studies [13], [21], [23]. For the 3D volumes, the depth dimension was determined to cover the height of an average buried threat response with some background included for added context.

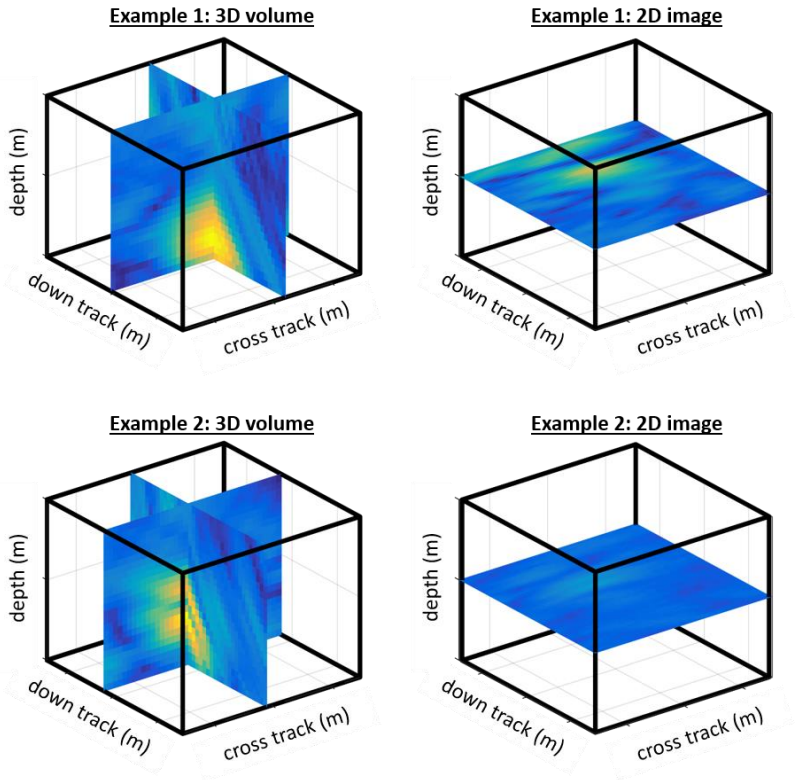



**Figure 37: For two known threat locations, 3D radar volumes (left column) and 2D images (right column) are beamformed.For each example the color scale is consistent showing the increased energy reflection when focusing at multiple depths during beamforming.**

Like in previous chapters, before features are extracted from an image patch, $X$, each patch is normalized by the observations' statistics [13], [23], [37], [73], [74]. A normalized patch, denoted $X'$, is computed by,

$$X' = \frac{X - \mu}{\sigma}. \tag{32}$$

Here $\mu$ is the mean, and $\sigma$ is the standard deviation of the image patch. All features in this work are extracted on the normalized image, $X'$. For the 3D radar volumes, this same normalization procedure is applied. The radar volume has its mean removed and is

normalized by the standard deviation on an individual observation basis.

## 7.2 *Volume Features*

Features extracted over the 2D radar image and 3D volumes are described in this section. Each is a standard feature that has been used previously in FLGPR data processing with the 2D radar images. The extensions from the 2D radar image features to those that are appropriate for a 3D radar volume are also explained.

### 7.2.1 Raw pixels

This feature uses the vectorization of the pixels in $X'$ as a feature vector. The direct extension of this concept is also used with the multiple beamformed slices, that is, a vectorization of the 3D radar volume. This increases the overall feature length by a multiple of the number of depth slices.

### 7.2.2 Local statistics (LSTAT)

LSTAT was used previously in the work presented in prior chapters, and can be extended to the 3D radar volumes. To compute an LSTAT feature, the radar image is divided into a grid of non-overlapping regions in the *cross track* and *down track* dimensions. The feature vector consists of the mean and variance of the pixel intensities in each of the grid regions. For the 3D LSTAT feature, statistics are extracted from a volume split into non-overlapping regions in *down track, cross track*, and *depth*. The statistics (i.e., mean and variance) are concatenated from each region and included in the

feature vector. This increases the dimensionality for the 3D case by a multiple of the number of regions in the depth dimension.

### 7.2.3 2-Dimensional FFT (2DFFT+ feature)

This feature, for a single 2D radar image, consists of two quadrants from the 2DFFT performed over the real and magnitude images, it is referred to as the 2DFFT+ feature [75]. When extended to the 3D radar volume, 2DFFT+ features are extracted at multiple depths, one at each depth slice. These slice 2DFFT+ features are concatenated together to form the feature vector. This "multi-2DFFT+ feature" increases in overall feature length by a multiple of the number of depth slices.

## 7.3 *Experimental Results*

This section contains a description of the experimental setup and performance results comparing features from the 2D radar image and the 3D radar volumes. First, the experimental methods are described and is followed by the classification results and analysis.

### 7.3.1 Experimental design

The experiments that were performed were aimed at comparing features extracted from a 2D image versus those extracted from a 3D volume. With a set of labeled threat and non-threat observations, classifiers were trained and tested, in a lane-based cross validation paradigm (e.g., [13], [21], [23]) to predict how likely an observation is to be a

threat. Each beamforming method, feature extractor, and classifier combination is evaluated in lane-based cross-validation. The classifier used for these experiments was the linear 3-component partial least squares discriminant analysis (PLSDA). The results are presented in the next subsection.

### 7.3.2 Three-dimensional feature classification results

Below, the classification results from the beamforming method experiments for each feature extractor and classifier are presented. In Figure 38, the classification results are shown in pAUC for the PLSDA classifier. Detection performance improves for each feature set when extended from the 2D images to the 3D volumes. There is a much more substantive performance increase for the raw pixel and LSTAT features than for the 2D FFT features, which may suggest that the spatial location of the radar reflection is increasingly important when adding the depth context.

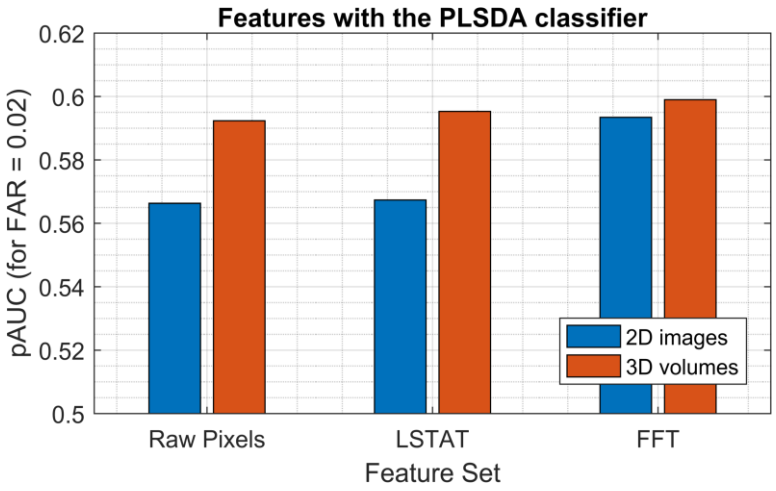



**Figure 38: Classification results for the 2D images and 3D volumes, presented by partial area under the ROC curve (pAUC) for each feature set with the PLSDA classifier. Each extension of the feature for the 3D radar volume of data out performs the 2D image feature counterpart.**

### 7.3.3 Three-dimensional feature analysis

Figure 39 presents a visualization of the PLSDA filter for the 3D volume raw pixel feature. The decision confidence computed for the PLSDA classifier is the inner product of this filter and the individual observations. In training, PLSDA tends to learn the areas in the feature vector that correlate best with the target label. In this figure the areas within the volume that are indicative of a buried threat are the more yellow areas. This figure illustrates the large amount of variation in depth over which threat responses are located. This further supports the hypothesis that images formed at a single depth are insufficient for identifying all threats effectively.

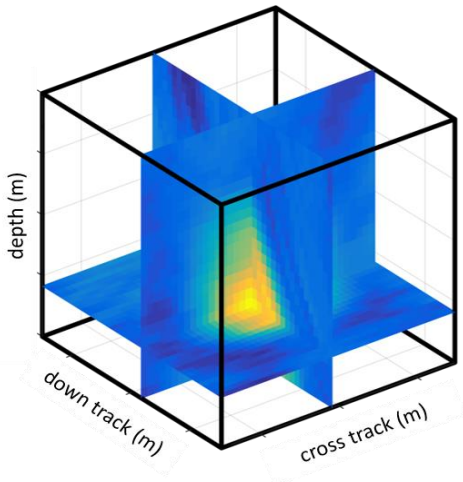



**Figure 39: PLSDA filter for the 3D volume raw pixel feature.Slices of the 3D filter volume in order to reveal the structure in the middle. Yellow indicates the more target-like areas of the volume.**

## 7.4 *Conclusion*

In these experiments, we explored the detection performance effects of beamforming responses of FLGPR data at multiple depths, and using this 3D volume for detection as opposed to the more traditional 2D beamformed image. Image features previously used with the FLGPR imagery were extended to be consistent with the radar volume data. With these features, experiments were conducted to compare the classification ability of the radar volume with that of the single depth radar image. The results show that by introducing the multiple depth images classification performance improves for each feature and classifier combination. Additionally, the results imply that the 3D radar volumes contain some useful classification information likely not present in the 2D image slices.

# 8 Automatic calibration of beamforming depth

This chapter introduces a method for selecting an appropriate beamforming depth for the 2D radar images. Before introducing the depth selection method, an experiment is conducted the results of which suggest that the previously used 2D images were being beamformed at a depth which detracts from overall detection performance. By beamforming at the correct depth, we learn that the performance gap between 3D and 2D features disappears. While it is always possible to search in a large 3D space for each observation (i.e., beamform large volumes for each prescreener alarm), the method presented here to find the correct beamforming depth reduces that need and increases performance with little additional computational time.

## 8.1 *Performance at individual depths*

This section presents an experiment for further examining the usefulness of the 3D radar volumes. At the conclusion of the previous chapter, the results indicated that using the 3D volumes proved to provide greater detection performance than using the single 2D slices. This is not a sufficient condition to conclude that the whole 3D volume is necessary for that increase in performance. Another hypothesis is that the single 2D image was being formed in a location in space not as useful for landmine detection. The results presented in Figure 40 plot the pAUC for 2D slices as the beamforming focus depth is varied. The feature used to generate the results shown here is the 2DFFT+ coupled with the PLSDA classifier. The dotted black line is the 3D volume performance from the

previous chapter using the same feature and classifier. Notice that the 3D volume performance is practically identical to the performance of a single 2D radar image at the $0.5m$ depth. This suggests that while the 3D volume was improving the overall detection performance of the FLGPR system, a way to achieve the same performance is to form the images at a more useful depth.

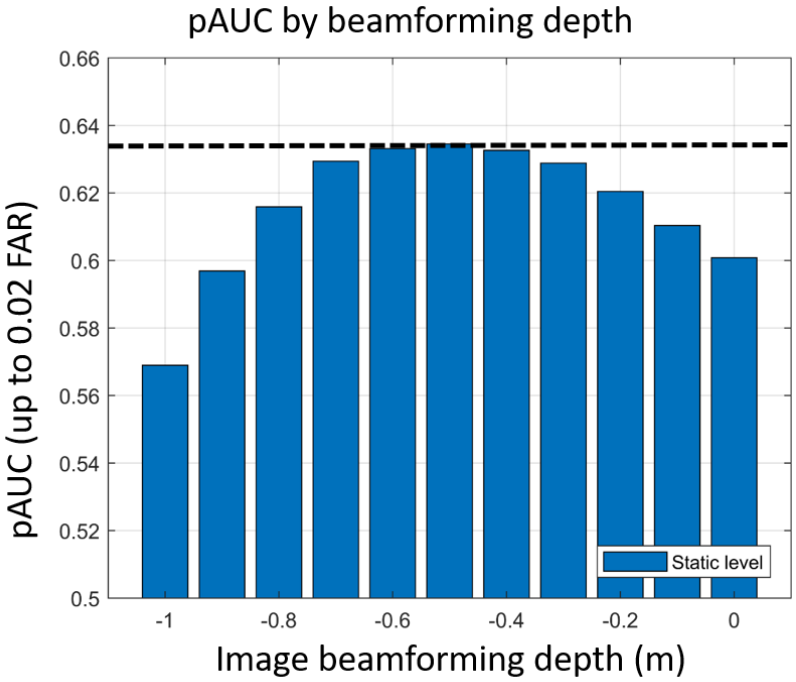



**Figure 40: Performance in pAUC for the 2DFFT+ feature with PLSDA on images formed as the search depth is varied from -1 meter to 0 meters beneath the ground.The black dotted line is the performance of the 3D volume with the equivalent feature set and classifier.**

## 8.2 *Automatic location of the beamforming depth*

As evident from the results shown above, the 3D volumes may not have been

capturing the shape of the response across depth but were instead exposed to better single depths for detecting landmines. Conventionally, the height for beamforming the 2D images for the FLGPR system has been at the ground surface [8]. This focus depth selection, as shown above, is not optimal for the detection of landmines. While using a *post hoc analysis* for beamforming the images at varied focus depths it becomes clear which depth should be used, this depth to use for each beamforming image formation is not known *apriori* to the system. The work in this section explores the development of a method for choosing a focus depth for beamforming without creating multiple slices of the radar images.

The strategy for finding the selecting a desirable image focus depth is inspired by the depth location of the high-energy region seen in the example target 3D radar volumes and the 3D Raw PLSDA classifier filter (see Figure 37 and Figure 39, respectively). Without computing the full volume of radar response, it is possible to see the depth location of the highest energy reflection of the target by only beamforming along a single cross track location and over multiple depth and down track locations. The average of all alarms, only target, and only non-targets is shown in Figure 41 over this altered view (i.e., down track by depth/elevation). The average depth response location is clearly below $0m$ (the value conventionally chosen for beamforming). In all three cases is it evident that the $0.5m$ depth is where the most energy is being returned by the average observation.

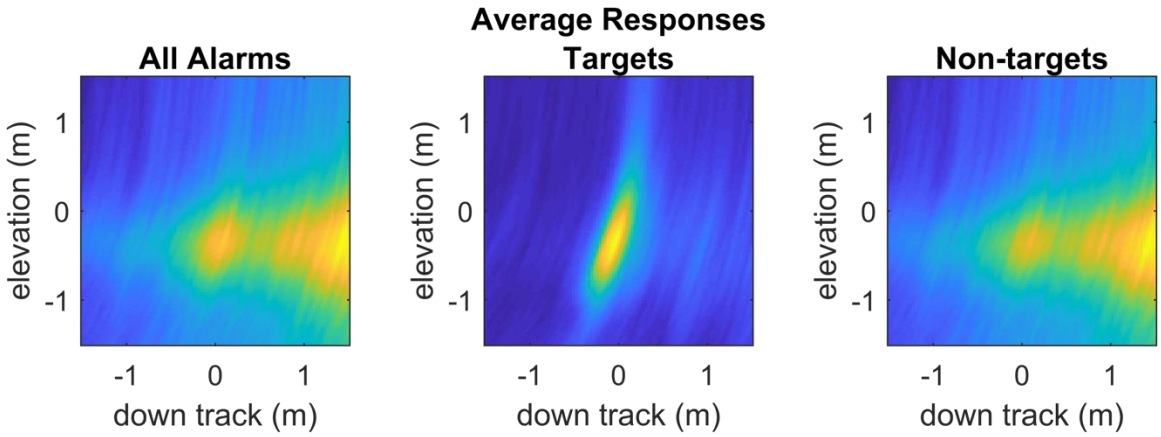



**Figure 41: Average radar images in the depth and down track dimensions for three observations sets: all alarms (left), only target (middle), and only non-targets (right). These results show where, on average, the reflection energy is concentrated in depth.**

This method, while taking less computational time than beamforming the full 3D radar volumes, does still require the beamforming of each observation. To reduce this constraint, the average max energy location is computed for the observations as the system advances down a single lane. Figure 42 shows the cumulative average of the max depth/elevation location of the observations' responses. The cumulative average converges to the $0.5m$ level for all alarms within the first half of the lane. This is the depth level which was found to be most useful in detection in the classification test above. Other lanes converged within similar distances and at nearly identical levels. With this method, a single lane at previously unknown site can be used to calibrate the image depth level and improve upon traditionally depth measuring approaches.

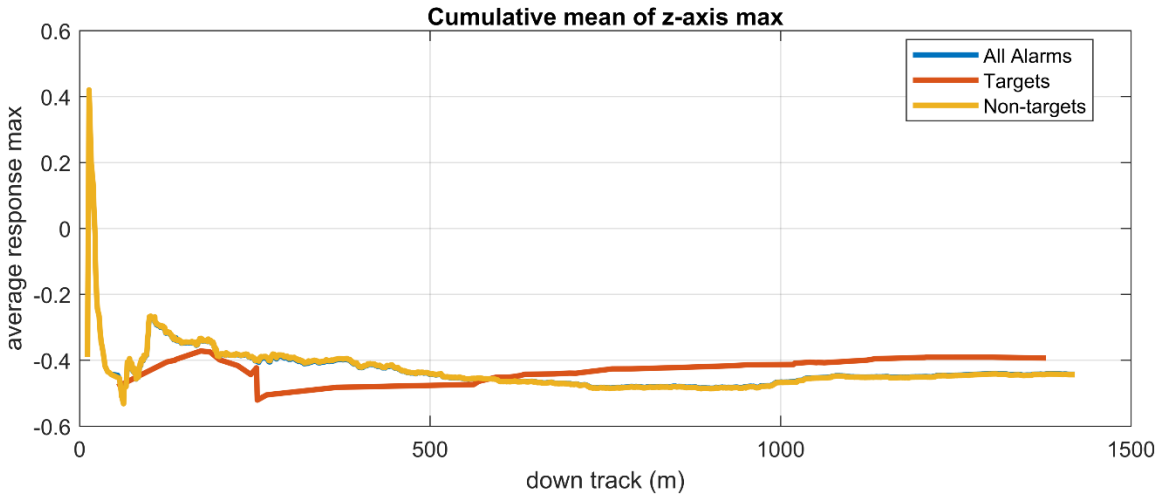



**Figure 42: cumulative average of the maximum depth location of the energy response across observations as the system advances down the lane.The average converges within the first half of the lane and implies that only that amount of the lane will need to be beamformed into images for the correct focus depth to be predicted. (note that the "All alarms" (blue) line is completely covered by the "Non-targets" (yellow) line)**

With this method, the depth is selected to be $0.5m$ and is used in the final performance comparison shown in Figure 43. The performance comparison shown here is between the conventional beamforming at both the $0m$ and $0.5m$ depth levels. Additionally, the learned antenna weightings are applied to both depth levels to show the effectiveness at either depth. The learned antenna weightings used here is the combined weighting from the independently learned aperture and Tx-Rx pair (with cross track location). There is a clear trend of performance increase by adding both the depth correction and the learned antenna weighting method.

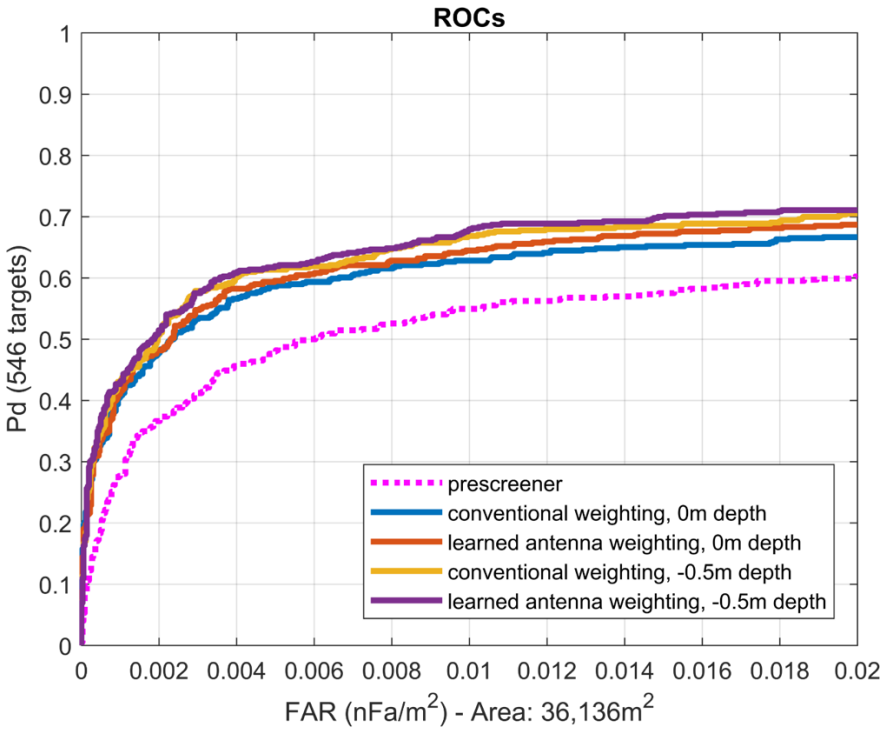



**Figure 43: Final ROC performance comparison with the depth selection and automatically learned antenna calibrations.Both methods provide performance improvements when applied separately and together.**

## 8.3 *Conclusions*

In this chapter the depth at which the 2D radar images are beamformed was investigated. It was found that by choosing a depth which aligns with the max energy locations of the response dramatically improves detection performance. This performance matches found from the 3D radar volume testing. The depth selection method presented in this chapter shows that the correct depth for beamforming can be found without the need to beamform full 3D volumes, saving on the large computational cost associated with created the large 3D radar volumes. Further, it is shown that by only beamforming

across depth and down track for less than half of a single lane, the maximum response energy average stabilizes and indicates the appropriate depth at which to beamform.

# 9 Conclusions

The following section is a summary of the work presented in this dissertation and the major conclusions which can be drawn.

The goal of the first part of this work was to provide a fair comparison of FLGPR image features. To achieve this goal, each of the existing features were extracted on a large dataset of FLGPR imagery along with several different supervised classification models: PLSDA, a linear SVM, and a nonlinear SVM. Also, two recent feature learning techniques were adapted for the application of detection of buried threats in FLGPR imagery, to compare existing features with recent state-of-the-art approaches from the computer vision community. The effectiveness of the features was also shown to vary depending on the polarity of the radar, so each feature-classifier pair was evaluated on each of three radar polarization schemes: HH, VH, and VV. The experiments to evaluate these different polarity-feature-classifier combinations employed a large dataset of FLGPR imagery and used lane-based cross-validation to provide an unbiased and reliable performance estimates. The results indicated that the features extracted from the HH polarization images are individually most effective for discrimination. Results also indicate that the linear PLSDA classifier, in this context, yields similar performance to the more complicated non-linear RBF SVM, but requires much less computation during both training and testing. The results also indicated that, while the disparity in performance between different features was not large, the 2DFFT+ feature with the PLSDA classifier on

HH polarization performed best among individual feature sets.

The second part of this FLGPR investigation focused on the image formation process. Two separate approaches for improving the image formulation process in order to improve detection performance were investigated. The first strategy modified the antenna summation to include a weighting, or window, to better combine the individual antenna responses. Ultimately this led to the learned weightings across both dimensions of the sensor array (apertures and Tx-Rx pairs). For the aperture weighting it was shown that unlabeled observations led to a weighting which improved landmine detection performance. The Tx-Rx pair weightings improved upon the conventional window in multiple ways; first by creating a weighting for the whole Tx-Rx pair, and second by adapting that weighting to depend on the specific look direction. By using the current pixel cross track location, the Tx-Rx pairs on the side of the vehicle toward the investigated location were systematically weighted higher; this lead to subsequent improvement in performance. A second strategy for improving the underlying radar images was motivated by the success of the 3D radar beamforming investigation. However, further experiments showed the performance gap between the 3D and 2D features entirely disappeared when the correct beamforming depth for the 2D image was selected. The investigation then turned to producing a method for finding the appropriate beamforming depth for 2D images without the need to beamform full 3D radar volumes. By examining the max energy location in depth for the first section of a single lane the best

beamforming depth becomes evident. Lastly, by combining the depth selection strategy and the learned antenna weightings state-of-the-art performance is achieved.

In summary, the following broader conclusions can be drawn from this work:

- Most feature sets resulted in similar performance, though the newly developed 2DFFT+ feature on the HH polarization performed best.
- Using the combined antenna weightings which were learned independently across the 2D array leads to state-of-the-art detection performance.
- Learned antenna weightings benefit from being dependent on the cross track observation location and can be inferred well without ground labels.
- A simple unsupervised method based off the depth of the radar response's max energy location correctly predicts the best burial depth for landmine classification.

# Bibliography


[1] iCBL-CMC Governance Board, "Landmine Monitor 2015," *Int. Campaign to Ban Landmines*, vol. 1, 2015.

[2] C. R. Ratto, P. A. Torrione, and L. M. Collins, "Exploiting ground-penetrating radar phenomenology in a context-dependent framework for landmine detection and discrimination," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, pp. 1689–1700, 2011.

[3] P. A. Torrione and L. M. Collins, "Texture features for antitank landmine detection using ground penetrating radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 7, pp. 2374–2382, 2007.

[4] J. N. Wilson, P. Gader, S. Member, W.-H. Lee, H. Frigui, and K. C. Ho, "A Large-Scale Systematic Evaluation of Algorithms Using Ground-Penetrating Radar for Landmine Detection and Discrimination," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 8, pp. 2560–2572, 2007.

[5] P. A. Torrione, K. D. Morton, R. Sakaguchi, and L. M. Collins, "Histograms of Oriented Gradients for Landmine Detection in Ground-Penetrating Radar Data," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1539–1550, Mar. 2014.

[6] NIITEK inc., "Ground penetrating radar systems for landmine detection." [Online]. Available: http://www.niitek.com/.

[7] Jacqueline MacDonald and J. R. Lockwood, "Alternatives for Landmine Detection," 2003.

[8] Y. Wang, X. Li, Y. Sun, J. Li, and P. Stoica, "Adaptive imaging for forward-looking ground penetrating radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 922–936, 2005.

[9] J. M. Malof, K. D. Morton, P. A. Torrione, and L. M. Collins, "Fusion of forward-looking infrared and ground penetrating radar for improved stopping distances in landmine detection," in *Proceedings of SPIE*, 2014, vol. 9072, pp. 1–11.

[10] T. C. Havens, C. J. Spain, K. C. Ho, J. M. Keller, T. T. Ton, D. C. Wong, and M. Soumekh, "Improved Detection and False Alarm Rejection Using FLGPR and Color Imagery in a Forward-Looking System," vol. 7664, p. 76641U–76641U–12, Apr. 2010.

[11] J. Kositsky, R. Cosgrove, C. Amazeen, and P. Milanfar, "Results from a forward-looking GPR mine detection system," in *Proceedings of SPIE*, 2002, vol. 4742, pp. 206–217.

[12] G. Liu, Y. Wang, J. Li, and M. R. Bradley, "SAR imaging for a forward-looking GPR system," *Proc. SPIE*, vol. 5089. pp. 322–333, 2003.

[13] J. A. Camilo, J. M. Malof, P. A. Torrione, L. M. Collins, and K. D. Morton, "Clutter and target discrimination in forward-looking ground penetrating radar using sparse structured basis pursuits," vol. 9454, p. 94540V, 2015.

[14] M. Cheney, "Fundamentals of Radar Imaging," 2008.

[15] I. S. Merrill and others, *Introduction to radar systems*. Mc Grow-Hill, 2001.

[16] T. Ton, D. C. Wong, M. Soumekh, I. I. Alaric, and F. System, "ALARIC Forward - Looking Ground Penetrating Radar System with Standoff Capability," *Wirel. Inf. Technol. Syst. (ICWITS), 2010 IEEE Int. Conf.*, pp. 1–4, 2010.

[17] Y. Sun and J. Li, "Plastic landmine detection using time-frequency analysis for forward-looking ground-penetrating radar," in *Proceedings of SPIE*, 2003, vol. 5089.

[18] T. Wang, J. M. Keller, P. D. Gader, and S. Member, "Frequency Subband Processing and Feature Analysis of Forward-Looking Ground-Penetrating Radar Signals for Land-Mine Detection," vol. 45, no. 3, pp. 718–729, 2007.

[19] T. Jin, J. Lou, and Z. Zhou, "Extraction of landmine features using a forward-looking ground-penetrating radar with MIMO array," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10 PART2, pp. 4135–4144, 2012.

[20] B. P. Burns, "Forward looking GPR sidelobe reduction using L1-norm minimization," *Proc. SPIE*, vol. 8357, p. 835713, May 2012.

[21] T. C. Havens, J. Becker, A. Pinar, and T. J. Schulz, "Multi-band sensor-fused explosive hazards detection in forward-looking ground-penetrating radar," *Proc. SPIE*, vol. 9072, p. 90720T, May 2014.

[22] J. Farrell, T. C. Havens, K. C. Ho, J. M. Keller, T. T. Ton, D. C. Wong, and M. Soumekh, "Evaluation and improvement of spectral features for the detection of buried explosive hazards using forward-looking ground-penetrating radar," in *Proc. SPIE*, 2012, vol. 8357, p. 8357C.

[23] D. Shaw, K. C. Ho, K. Stone, J. M. Keller, M. Popescu, D. T. Anderson, R. H. Luke, and B. P. Burns, "Explosive hazard detection using MIMO forward-looking ground penetrating radar," vol. 9454, p. 94540Z, 2015.

[24] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[25] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. 2007.

[26] A. Coates and A. Y. Ng, "@incollection{coates2012learning, title={Learning feature representations with k-means}, author={Coates, Adam and Ng, Andrew Y}, booktitle={Neural networks: Tricks of the trade}, pages={561--580}, year={2012}, publisher={Springer} }," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7700 LECTU, pp. 561–580, 2012.

[27] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: an evaluation of recent feature encoding methods," *Procedings Br. Mach. Vis. Conf. 2011*, no. 1, p. 76.1-76.12, 2011.

[28] A. Coates, A. Arbor, and A. Y. Ng, "An Analysis of Single-Layer Networks in Unsupervised Feature Learning," *Aistats 2011*, pp. 215–223, 2011.

[29] F. Perronnin, "Universal and adapted vocabularies for generic visual categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1243–1256, 2008.

[30] L. J. Zhao, P. Tang, and L. Z. Huo, "Land-use scene classification using a concentric circle-structured multiscale bag-of-visual-words model," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 7, no. 12, pp. 4620–4631, 2014.

[31] S. Xu, T. Fang, D. Li, and S. Wang, "Object classification of aerial images with bag-of-visual words," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 2, pp. 366–370, 2010.

[32] J. Feng, L. C. Jiao, X. Zhang, and D. Yang, "Bag-of-visual-words based on clonal selection algorithm for SAR image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 691–695, 2011.

[33] K. Simonyan, O. Parkhi, A. Vedaldi, and A. Zisserman, "Fisher Vector Faces in the Wild," *Procedings Br. Mach. Vis. Conf. 2013*, p. 8.1-8.11, 2013.

[34] C. J. Oliver, "Synthetic-aperture radar imaging," *J. Phys. D Appl. Phys.*, vol. 22, 1989.

[35] I. S. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *Acoust. Speech Signal Process. IEEE Trans.*, vol. 38, no. 10, pp. 1760–1770, 1990.

[36] I. S. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE Trans. Acoust.*, vol. 38, no. October, pp. 1760–1770, 1990.

[37] J. A. Camilo, J. M. Malof, and L. M. Collins, "A feature learning approach for classifying buried threats in forward looking ground penetrating radar data," *Proc. SPIE*, vol. 9823, p. 98231I, 2016.

[38] J. M. Malof, K. D. Morton, L. M. Collins, and P. A. Torrione, "Processing forward-looking data for anomaly detection: single-look, multi-look, and spatial classification," in *Proceedings of SPIE*, 2012, vol. 8357, p. 83571O–83571O–10.

[39] B. Kulis and M. I. Jordan, "Revisiting k-means: New Algorithms via Bayesian Nonparametrics," *Proc. 29 th Int. Conf. Mach. Learn.*, p. 14, 2011.

[40] A. Webb, T. C. Havens, and T. J. Schulz, "An apodization approach for processing forward-looking GPR for buried target detection," vol. 9454, p. 94540X, 2015.

[41] K. C. Ho, "Improving Landmine Detection Using Frequency Domain Features from Ground Penetrating Radar," vol. 0, no. C, pp. 1617–1620, 2004.

[42] S. Wold, M. Sjöström, and L. Eriksson, "PLS-regression: A basic tool of chemometrics," *Chemom. Intell. Lab. Syst.*, vol. 58, no. 2, pp. 109–130, 2001.

[43] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image Classification with the Fisher Vector: Theory and Practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, 2013.

[44] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognit.*, vol. 30, no. 7, pp. 1145–1159, 1997.

[45] S. D. Walter, "The partial area under the summary ROC curve," *Stat. Med.*, vol. 24, no. 13, pp. 2025–2040, 2005.

[46] D. G. Lowe, "Object recognition from local scale-invariant features," *Proc. Seventh IEEE Int. Conf. Comput. Vis.*, vol. 2, no. [8, pp. 1150–1157, 1999.

[47] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *CVPR '05 Proc. 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Vol. 1*, pp. 886–893, 2005.

[48] S. Clinchant, G. Csurka, F. Perronnin, J. Renders, C. Europe, and D. Maupertuis, "XRCE ' s participation to ImagEval."

[49] K. C. Ho, L. Carin, P. D. Gader, and J. N. Wilson, "An Investigation of Using the Spectral Characteristics From Ground Penetrating Radar for Landmine / Clutter Discrimination," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1177–1191, 2008.

[50] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells.," *J. Opt. Soc. Am. A.*, vol. 4, no. 12, pp. 2379–2394, 1987.

[51] P. Somol, P. Pudil, and J. Novovi, "Adaptive foating search methods in feature selection," *Pattern Recognit.*, vol. 20, no. November, pp. 1157–1163, 1999.

[52] G. Csurka and F. Perronnin, "Fisher vectors: Beyond bag-of-visual-words image representations," *Commun. Comput. Inf. Sci.*, vol. 229 CCIS, pp. 28–42, 2011.

[53] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher Kernel for Large-Scale Image Classification," *Comput. Vis. – ECCV 2010*, vol. 6314, pp. 143–156, 2010.

[54] I. S. Dhillon and D. S. Modha, "Concept decompositions for large sparse text data using clustering," *Mach. Learn.*, vol. 42, no. 1–2, pp. 143–175, 2001.

[55] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, pp. 2169–2178, 2006.

[56] L. Breiman, "Bagging Predictors," *Mach. Learn.*, vol. 24, no. 421, pp. 123–140, 1996.

[57] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*. SIAM, 1982.

[58] B. Efron, "Bootstrap Methods: Another Look at the Jackknife," *Ann. Stat.*, vol. 7, no. 1, pp. 1–26, 1979.

[59] A. Papoulis, "Probability, Random Variables, and Stochastic Processes," *McGraw-Hill*, pp. 149–151, 1984.

[60] P. K. Varshney, *Distributed detection and data fusion*. Springer Science & Business Media, 2012.

[61] D. Shaw, K. Stone, K. C. Ho, J. M. Keller, R. H. Luke, and B. P. Burns, "Sequential feature selection for detecting buried objects using forward looking ground penetrating radar," *SPIE Vol. 9831*, vol. 9823, p. 98231L, 2016.

[62] T. Fawcett, "An Introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 6, pp. 299–309, 2006.

[63] J. A. Camilo, L. M. Collins, and J. M. Malof, "A large comparison of feature-based approaches for buried target classification in forward-looking ground-penetrating radar," *arXiv: 1702.03000*, Feb. 2017.

[64] Y. Liu and W. Rayens, "PLS and dimension reduction for classification," *Comput. Stat.*, vol. 22, no. 2, pp. 189–208, 2007.

[65] J. Farrell, T. C. Havens, K. C. Ho, J. M. Keller, T. T. Ton, D. C. Wong, and M. Soumekh, "Detection of explosive hazards using spectrum features from forward-looking ground penetrating radar imagery," *Proc. SPIE*, vol. 8017, p. 80171F, 2011.

[66] J. Ponce and D. Forsyth, *Computer vision: a modern approach*. 2012.

[67] J. Shlens, "A Tutorial on Principal Component Analysis," 2014.

[68] M. Barker and W. Rayens, "Partial least squares for discrimination," *J. Chemom.*, vol. 17, no. 3, pp. 166–173, 2003.

[69] M.Soumekh, *Synthetic Aperture Radar Signal Processing with MATLAB Algorithms*. 1999.

[70] J. A. Camilo, M. Crosskey, K. D. M. Jr., L. M. Collins, and J. M. Malof, "An improved frequency domain feature with partial least-squares dimensionality reduction for classifying buried threats in forward-looking ground-penetrating radar data," *SPIE Defense, Secur. Sens.*, 2017.

[71] H. Abdi and L. J. Williams, "Partial least squares methods: Partial least squares correlation and partial least square regression," *Methods Mol. Biol.*, vol. 930, pp. 549–579, 2013.

[72] G. Liu, Y. Wang, J. Li, and M. R. Bradley, "SAR imaging for a forward-looking GPR system," *Proc. SPIE*, vol. 5089. pp. 322–333, 2003.

[73] J. Becker, T. C. Havens, A. Pinar, and T. J. Schulz, "Deep belief networks for false

alarm rejection in forward-looking ground-penetrating radar," vol. 9454, p. 94540W, 2015.

[74] A. Webb, T. C. Havens, and T. J. Schulz, "Spectral diversity for ground clutter mitigation in forward-looking GPR," *SPIE Vol. 9831*, vol. 9823, p. 98231M, 2016.

[75] J. A. Camilo, M. Crosskey, K. D. M. Jr., L. M. Collins, and J. M. Malof, "An improved frequency domain feature with partial least-squares dimensionality reduction for classifying buried threats in forward-looking ground-penetrating radar data," in *SPIE Defense, Security, and Sensing*, 2017.

# Biography

Joseph A. Camilo was born on January, 22nd, 1992 in Johnson City NY. He received his B.S. from Clarkson University, Potsdam, NY, in 2013. He completed his M.S. in electrical engineering at Duke University, Durham, NC, in 2015. He is currently working towards his Ph.D. in electrical engineering at Duke University in the Applied Machine Learning Lab (AMLL). His current research interests are machine learning, radar imaging, and computer vision applications.

**List of publications:**


Camilo, Joseph A, Leslie M Collins, and Jordan M Malof. 2017. "A Large Comparison of Feature-Based Approaches for Buried Target Classification in Forward-Looking Ground-Penetrating Radar." *arXiv: 1702.03000*, February. http://arxiv.org/abs/1702.03000.

Camilo, Joseph A, Miles Crosskey, Kenneth D. Morton Jr., Leslie M. Collins, and Jordan M. Malof. 2017. "An Improved Frequency Domain Feature with Partial Least-Squares Dimensionality Reduction for Classifying Buried Threats in Forward-Looking Ground-Penetrating Radar Data." *SPIE Defense, Security, and Sensing*. doi:10.1117/12.2263034.

Camilo, Joseph A, Leslie M Collins, and Jordan M Malof. 2017. "Three-Dimensional Features, Based on Beamforming at Multiple Depths, Improves Landmine Detection with a Forward-Looking Ground-Penetrating Radar." In *2017 9th International Workshop on Advanced Ground Penetrating Radar (IWAGPR)*, 1–6. doi:10.1109/IWAGPR.2017.7996107.

Camilo, Joseph A, Jordan M Malof, and Leslie M Collins. 2016. "A Feature Learning Approach for Classifying Buried Threats in Forward Looking Ground Penetrating Radar Data." *Proc. SPIE* 9823: 98231I. doi:10.1117/12.2223117.

Camilo, Joseph A, Jordan M Malof, Peter A Torrione, Leslie M Collins, and Kenneth D Morton. 2015. "Clutter and Target Discrimination in Forward-Looking Ground

Penetrating Radar Using Sparse Structured Basis Pursuits" 9454: 94540V. doi:10.1117/12.2176491.