Multi-Band Sensor-Fused Explosive Hazards Detection in Forward-Looking Ground-Penetrating Radar

Timothy C. Havens[a,b], John Becker[a], Anthony Pinar[a], and Timothy J. Schulz[a]

[a]Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI USA

[b]Department of Computer Science, Michigan Technological University, Houghton, MI USA

ABSTRACT

Explosive hazard detection and remediation is a pertinent area of interest for the U.S. Army. There are many types of detection methods that the Army has or is currently investigating, including ground-penetrating radar, thermal and visible spectrum cameras, acoustic arrays, laser vibrometers, etc. Since standoff range is an important characteristic for sensor performance, forward-looking ground-penetrating radar has been investigated for some time. Recently, the Army has begun testing a forward-looking system that combines L-band and X-band radar arrays. Our work focuses on developing imaging and detection methods for this sensor-fused system. In this paper, we investigate approaches that fuse L-band radar and X-band radar for explosive hazard detection and false alarm rejection. We use multiple kernel learning with support vector machines as the classification method and histogram of gradients (HOG) and local statistics as the main feature descriptors. We also perform preliminary testing on a context aware approach for detection. Results on government furnished data show that our false alarm rejection method improves area-under-ROC by up to 158%.

Keywords: buried explosive hazards, forward-looking ground-penetrating radar, multiple kernel learning, machine learning, context recognition, target detection, false alarm reduction

1. INTRODUCTION

An important goal for the U.S. Army is remediating the threats of explosive hazards as these devices cause uncountable deaths and injuries to both civilians and soldiers throughout the world. Since 2008, explosive hazard attacks in Afghanistan have wounded or killed nearly 10,000 U.S. Soldiers; worldwide, explosive devices on average cause 310 deaths and 833 wounded per month.[1] Systems that detect these threats have included *ground-penetrating-radar* (GPR), *infrared* (IR) and visible-spectrum cameras, and acoustic technologies.[2–4] Recent research has examined both handheld and vehicle-mounted GPR-based systems and much progress has been made in increasing detection capabilities.[5,6] *Forward-looking* synthetic aperture GPR (FLGPR) is an especially attractive technology because of its ability to detect hazards before they are encountered; standoff distances can range from a few to tens of meters. FLGPR has been applied to the detection of side-attack, surface, and buried devices.[7–9] A drawback of FLGPR systems is that they are not only sensitive to explosive devices, UXO, and landmines, but also to other objects, both above and below the ground. Because FLGPR is a standoff sensor, the area being examined for targets is much larger than with downward-looking systems. Thus, clutter is a serious concern. Furthermore, the explosive hazard threat is very diverse—they are made from many different materials, including wood, plastic, and metal, and come in many different shapes and sizes. And this threat continues to evolve. Thus, it is nearly impossible to detect explosive hazards solely by a modeling-based approach. We have shown in previous work that if *forward-looking infrared* (FLIR) or visible-spectrum imagery is combined with L-band FLGPR, false alarm rates can be reduced significantly.[10–14] We also demonstrated that fusing multiple sub-bands and spectral features in L-band FLGPR improves detection performance.[15–17]

Since 2005, the U.S. Army *Night Vision and Electronic Sensors Directorate* (NVESD) has been integrating sensing technologies to aid the soldier in detecting explosive hazards while maintaining *Situational Awareness* (SA) in route clearance and convoy escort vehicles.[18] The U.S. Army NVESD fielded a system called ALARIC. This system combined an

Further author information: (Send correspondence to T.C.H.)
T.C.H..: E-mail: thavens@mtu.edu, Telephone: 1 906 487 3115
J.B.: E-mail: jtbecker@mtu.edu; A.P.: E-mail: ajpinar@mtu.edu; T.J.S.: E-mail: schulz@mtu.edu

Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XIX,
edited by Steven S. Bishop, Jason C. Isaacs, Proc. of SPIE Vol. 9072, 90720T
 · doi: 10.1117/12.2051034

Table 1. FLGPR Parameters

	L-band	**X-band**
Waveform	Stepped frequency	Stepped frequency
Transmitters	8	32
Receivers	8	4
Bandwidth	0.5–3.4 GHz	8.4–10.4 GHz
# Frequencies	2702	1024
Pulse rate	12 Hz	50 Hz
Polarizations*	HH, VV, HV, VH	VV

*Note that we only have HH and VV polarizations for the L-band data sets used in this paper.

Table 2. Symbols Used in This Paper

Symbol	**Description**
$\mathbf{x}(t_{GPS})$	Position of vehicle in UTM at time t_{GPS}
$\mathbf{v}(t_{GPS})$	Velocity (m/s) of vehicle at time t_{GPS}
$\mathbf{x}_j(t_{GPS})$	Position of jth antenna element at time t_{GPS}
$\mathbf{w}_{jk}(f, t_{GPS})$	I/Q signal of jkth T/R pair at time t_{GPS} and frequency f
$a_w(f)$	frequency-domain window (Hamming)
$a_r(j,k)$	aperture window
c	speed of light, 2.998×10^8 (m/s)

L-band FLGPR array with a high resolution visible-spectrum camera. Most recently, the U.S. Army has contracted to develop an integrated L-band and X-band FLGPR system. This paper will focus on developing computer-aided classification for this dual-band system.

Past work on detecting explosive hazards with FLGPR focused on intensity-based features in L-band images, but these features are mostly ineffective in discriminating clutter from targets. We have shown in our previous research that fusing other sensors with FLGPR can significantly reduce the *false alarm* (FA) rate due to clutter, while maintaining a 90-100% *probability of detection* (p_d).[10–17,19] However, data and sensor fusion methods are only as good as the data and sensors to which they are applied. We aim to build upon our previous fusion work by proposing a method which uses both L-band and X-band FLGPR to improve detection statistics for standoff-detection of explosive hazards. The method we describe here uses *multiple kernel learning* (MKL) to fuse multiple types of features measured in both the radar bands. Furthermore, we present results for a method that uses context, i.e., the properties of the media surrounding each detection, to inform context-aware classifier design.

2. RADAR IMAGING

The X- and L-band FLGPRs operate as stepped-frequency arrays—each transmitter individually illuminates the scene and all receivers then measure the complex return at each frequency and polarization, repeating for each transmitter. The parameters of the radars are shown in Table 1. The *government-furnished data* (GFD) are represented as I/Q values at each frequency for each *transmit/receive* (T/R) pair, GPS location, and pitch, roll, and yaw of the array. From these data we are able to localize each T/R pair in 3D UTM coordinates, allowing for fully motion-compensated imaging (within the error of the platform motion estimation).

The radar images, denoted as $I_p(u,v)$, where p is the polarization and (u,v) are the horizontal and vertical UTM coordinates of the image are formed by a backpropagation procedure as illustrated in Fig. 1. The steps of this process are as follows:

1. Remove self-interference by subtracting a windowed time-average of $\mathbf{w}_{jk}(f, t_{GPS})$ over the variable t_{GPS} at each frequency f and for each T/R pair.
2. For each frame (as indicated by t_{GPS}) and polarization:
   (a) Inverse Fourier transform the zero-padded (up-sampled), windowed signals $a_w(f)\mathbf{w}_{jk}(f, t_{GPS})$, where $a(f)$ is a Hamming window, producing the range/time signals $\mathbf{r}_{jk}(t, t_{GPS})$.

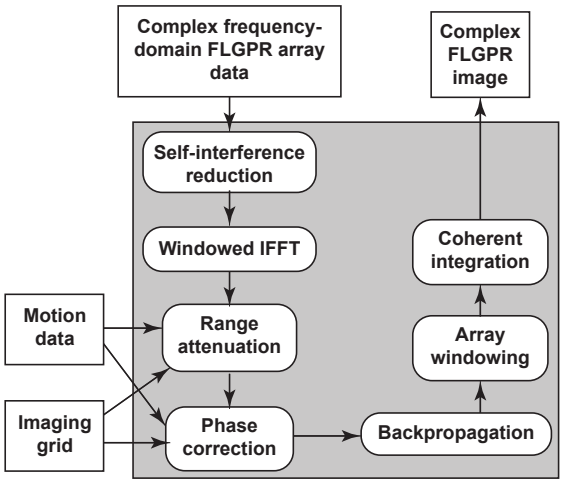


Figure 1. Block diagram of FLGPR backpropagation-based imaging algorithm.

(b) Interpolate and coherently integrate the windowed range signals $a_r(j,k)r_{jk}(t,t_{GPS})$ onto the predetermined grid (u,v), and apply the amplitude and phase correction $r^2\exp\{-i4\pi f_1 t\}$, where f_1 is the lowest frequency in the stepped frequency transmission. Note that the windowed range signals are only interpolated and integrated onto grid points that are between r_{min} and r_{max} range in front of the array (we use 5 and 25 meters, respectively).

We used a grid spacing of 2.5cm in both cross-range and down-range and upsample $\mathbf{w}$ by a factor of 16 (to the nearest power of 2). Figure 2 shows examples of images produced in two different lanes in both the X- and L-band systems. This form of backpropagation is the most basic synthetic aperture radar imaging method. There are other imaging techniques, e.g., recursive sidelobe minimization[20] and deblurring,[21,22] that we plan to investigate in the future to improve overall image results. However, the main focus of this paper is to examine how context-aware multiple kernel learning can improve detection results.

3. CFAR PRESCREENER

The result of the radar imaging procedure above is a coherently integrated image I at pre-determined UTM coordinates (u,v), one for each polarization of the L-band FLGPR and one image (the VV polarization) of the X-band FLGPR. It is well known that penetration depth increases with wavelength; hence, the L-band will have a deeper penetration than the X-band radar. Thus, we use the L-band radar as the detection radar for the method proposed here (although we will show results for X-band detection too).

The prescreening detector is the first algorithm that indicates candidate detection locations—a block diagram is shown in Fig. 3. We employed two methods to indicate the presence of a target, both of which could be considered to be *constant false alarm rate* (CFAR) detectors. Consider an FLGPR image $I(u,v)$, where u is the cross-range coordinate, and v the down-range. We then calculate four images from $I(u,v)$, denoted as $I_{\mu_c}(u,v)$, $I_{\mu_h}(u,v)$, $I_{\sigma_c^2}(u,v)$, $I_{\sigma_h^2}(u,v)$, calculated as

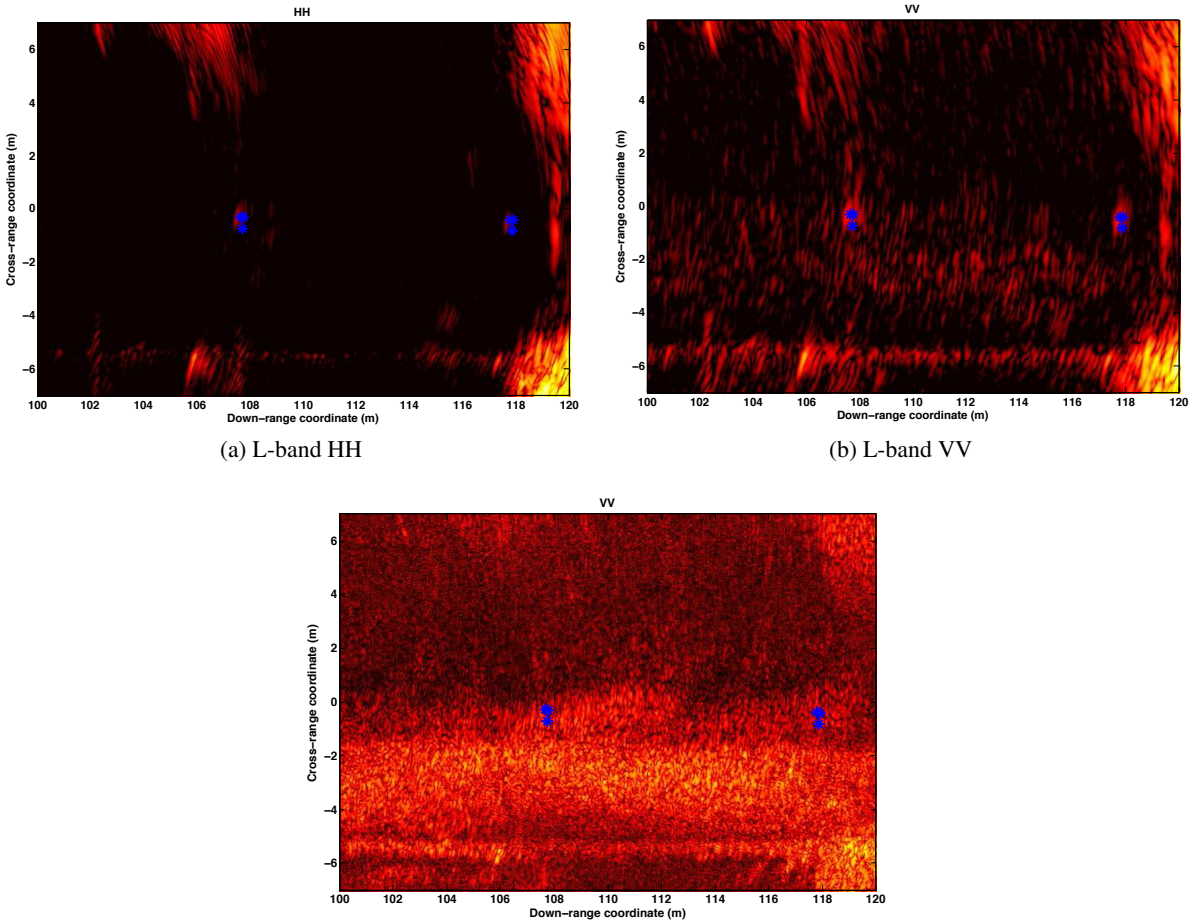
$$I_{\mu_c}(u,v) = \frac{\{I * H_c\}(u,v)}{\sum H_c}; \tag{1a}$$

$$I_{\mu_h}(u,v) = \frac{\{I * H_h\}(u,v)}{\sum H_h}; \tag{1b}$$

$$I_{\sigma_c^2}(u,v) = \{I^2 * H_c\}(u,v) - \{I * H_c\}^2(u,v); \tag{1c}$$

$$I_{\sigma_h^2}(u,v) = \{I^2 * H_h\}(u,v) - \{I * H_h\}^2(u,v); \tag{1d}$$

where I^2 indicates the image with each element squared, $*$ indicates convolution, and H_c and H_h are elliptical convolution kernels as shown in Fig. 4. In essence, I_{μ_c} and I_{μ_h} are the mean values of the pixels in the center and halo surrounding



(a) L-band HH

(b) L-band VV

(c) X-band VV

Figure 2. Example FLGPR images for X- and L-band radars—blue markers indicate ground-truth locations.

each pixel and $I_{\sigma_c^2}$ and $I_{\sigma_h^2}$ are the corresponding variances. Detections can now be indicated by either of the difference in the means (or a size-contrast filter) or the modified Bhattacharya distance:

$$I_{sc}(u,v) = I_{\mu_c}(u,v) - I_{\mu_h}(u,v); \tag{2a}$$

$$I_B(u,v) = \text{sgn}\{I_{sc}(u,v)\} \cdot \left[\log\left(\frac{1}{4}\left[\frac{I_{\sigma_c^2}(u,v)}{I_{\sigma_h^2}(u,v)} + \frac{I_{\sigma_h^2}(u,v)}{I_{\sigma_c^2}(u,v)} + 2\right]\right) + \frac{\left(I_{\mu_c}(u,v) - I_{\mu_h}(u,v)\right)^2}{I_{\sigma_c^2}(u,v) + I_{\sigma_h^2}(u,v)}\right]; \tag{2b}$$

where the Bhattacharya distance is modified so that it is signed such that positive distance indicates that the mean of the center is greater than the mean of the outer. In our experiments, we have determined the following prescreened parameters to be good choices: down-range radius $= 0.25$m; cross-range radius $= 0.5$m; and halo width $= 0.75$m.

One could simply threshold I_{sc} or I_B to indicate a detection; however, this can result in many detections in one local region. We wish to have one prototype detection location for each candidate target; hence, we first calculate a maximum order-filtered image, denoted $I_o(u,v)$, with a 3m (cross-range) by 1m (down-range) rectangular kernel. Detection locations are indicated by

$$A = \arg_{(u,v)}\{I_*(u,v) = I_o(u,v)\}, \tag{3}$$

where I_* is either I_{sc} or I_B and A is the set of cross-range and down-range locations of detections. At each detection location, we also extract a set of shape- or texture-based features, which we now describe.

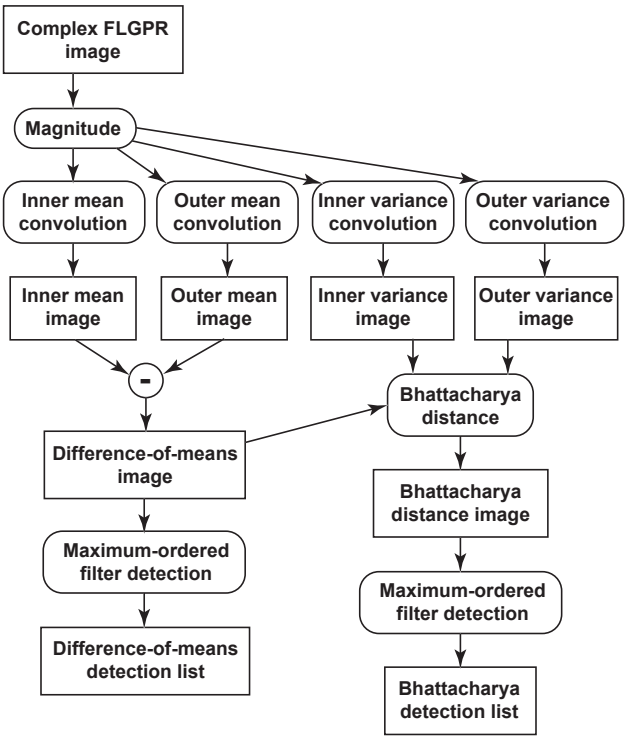


Figure 3. Block diagram of prescreener detection algorithm.

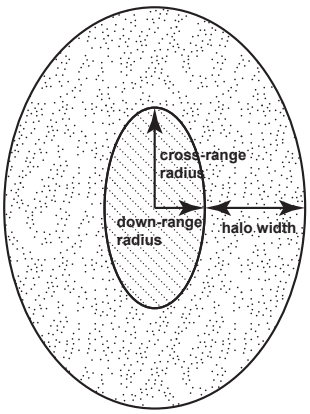
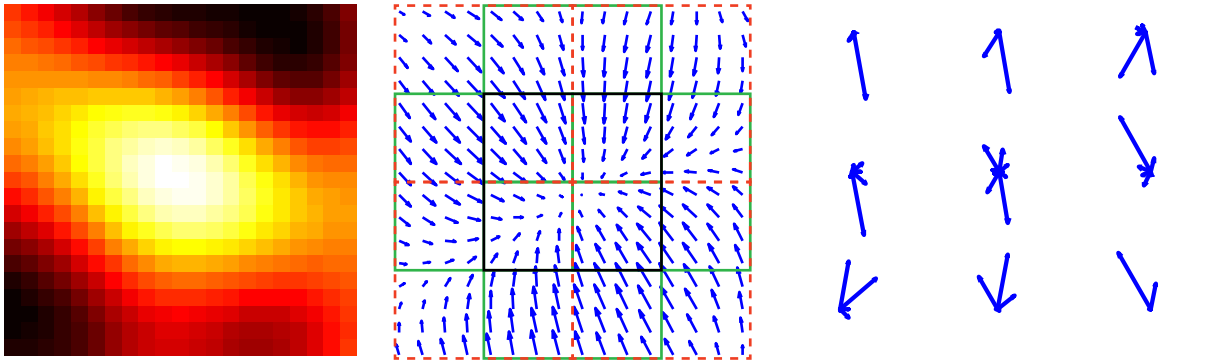


Figure 4. Elliptical convolution kernels used in prescreener. Detection is indicated by comparing distribution of pixel intensities in inner ellipse to distribution of pixel intensities in outer halo.



(a) Sub-image at hit location

(b) Gradient calculation and 3×3 cell arrangement with 50% overlap

(c) Cell-based 9-bin histogram of gradients feature

Figure 5. Example of histogram of ordered gradients (HOG) with 3×3 cell arrangement, 50% overlap of cells, 8×8 pixels per cell. Feature is a $3 \times 3 \times 9 = 81$-length vector of histogram components.

4. FEATURE EXTRACTION

At each detection location, we calculate an image-based texture feature called the *histogram of ordered gradients* (HOG).[23] This feature represents the texture by calculating local gradients and then compiling these gradients into a histogram descriptor. A histogram is computed for each cell of pixels surrounding the detection location. Each cell is composed of 8×8 pixels, and there are 3×3 cells. The center cell is centered on the detection location. The surrounding cells are organized such that there is 50% overlap between neighboring cells. The histogram in each cell has 9 bin centers. Thus, there are a total of 9 feature values per cell, for a total of 81 feature values that describe each detection. Figure 5 illustrates the HOG calculation for an example detection.

The second feature we calculate is a cell-based *local statistics* (LSTAT) vector. In each cell, as shown in Figure5(b), we compute the mean, median, and standard deviation of the pixels. These values are normalized over all hits by subtracting the mean and dividing by the standard deviation of the feature values.

5. MULTIPLE KERNEL LEARNING (MKL) CLASSIFIER

For this problem, a classifier is searching for a separating decision boundary in some hyperspace such that *explosive hazards* are on one side of the boundary and *false alarms* on the other. We denote the hazard class as $y = +1$ and the FA class as $y = -1$, where y is the class label. There are many methods that produce classifiers of this nature, including neural networks, k-nearest neighbor, and Bayesian classifiers.[24]

Most classifiers delineate the classes by finding some "best" decision boundary in the feature space. Neural networks and linear *support vector machines* (SVMs) find hyperplanes. These classifiers are easy to train, often can be effective, and are computationally very efficient (the operational decision is just a single dot-product in the feature space). However, they are ineffective against classes that are not linearly separable, i.e., by a hyperplane. Hence, we will use kernel classifiers to non-linearly project the features into a high-dimensional space, where hyperplanes may be more easily found that serve as good decision boundaries.

To do this projection, consider some nonlinear mapping $\phi : \mathbf{x} \rightarrow \phi(\mathbf{x}) \in \mathbb{R}^{d_K}$, where d_K is the dimensionality of the transformed feature vector $\phi(\mathbf{x})$. $\phi(\mathbf{x})$, being a high-dimensional version of $\mathbf{x}$, can have many dimensions; some kernel spaces have infinite dimensions. Thus it is inappropriate to explicitly operate in this new space. Hence, the classifiers only operate on the dot-product space of the higher-dimensional space, by using a kernel function $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$. This kernel function κ can take many forms, including the popularly used *radial basis function* (RBF) $\kappa_{RBF}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right)$. Most kernel algorithms work on what is called the *kernel matrix*, which is formed by storing all the pair-wise kernel functions, $K = [K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)] \in \mathbb{R}^{n \times n}$. This kernel matrix completely describes the transformed space, and this matrix (or collections of them) is how we will represent the features for this problem. We now describe the classifiers we will use in this paper.

5.1 Single Kernel SVM Classifier

The SVM algorithm is generally defined as the optimization problem

$$\min_{\mathbf{w},b} \frac{\|\mathbf{w}\|^2}{2}, \tag{4}$$

subject to

$$y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1,\ i = 1, \ldots, n, \tag{5}$$

where $y_i \in \{-1, +1\}$ are the class labels and $(\mathbf{w}^T\mathbf{x}_i - b)$ is the equation of the class-separating hyperplane. This form of the SVM, however, does not support overlapping classes,viz., errors in the training data. Hence, the soft-margin SVM was introduced as

$$\min_{\mathbf{w},\xi,b} \left\{ \frac{\|\mathbf{w}\|^2}{2} + C\sum_{i=1}^{n} \xi_i \right\}, \tag{6a}$$

subject to

$$y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 - \xi_i,\ \xi_i > 0,\ i = 1, \ldots, n, \tag{6b}$$

where C determines how many errors are allowed in the training.[25] Notice that this is the linear soft-margin SVM. The kernel soft-margin SVM is easily defined in the dual form of the SVM, where the optimization problem is solved using Lagrange multipliers. The *single-kernel* SVM (SKSVM) is defined as

$$\max_{\alpha} \left\{ \mathbf{1}^T\alpha - \frac{1}{2}(\alpha \circ \mathbf{y})^T K(\alpha \circ \mathbf{y}) \right\}, \tag{7a}$$

subject to

$$0 \geq \alpha_i \geq C,\ i = 1, \ldots, n,\ \alpha^T\mathbf{y} = 0, \tag{7b}$$

where $\mathbf{1}$ is the n-length vectors of 1s, $K = [\kappa(\mathbf{x}_i, \mathbf{x}_j)] \in \mathbb{R}^{n\times n}$ is the kernel matrix, and $\circ$ indicates the Hadamard product.[26] Note that SKSVM reduces to the linear SVM for the kernel $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$ (which is simply the Euclidean dot product).

One of the drawbacks of using the above SVM formulation is that it treats each datum equally; hence, when there is an imbalance between the number of datum in each class, then the SVM decision boundary is driven primarily by the data from the class with more data points. This is a problem in explosive hazards detection as there are typically many more false alarm detections than there are true targets—the true targets only comprise a small overall area of the lane. To attack this issue, we use a formulation of the SVM for imbalanced data which uses a different error cost for positive (C^+) and negative (C^-) classes. Specifically, we change the constraints of the kernel SVM formulation at (7) to

$$0 \geq \alpha_i \geq C^+, \forall i | y_i = +1;\ 0 \geq \alpha_i \geq C^-, \forall i | y_i = -1;\ \alpha^T\mathbf{y} = 0; \tag{8}$$

where C^+ is the error constant applied to the positive class and C^- is the error constant applied to the negative class. In our application, the positive class is true positives and the negative class is false alarms. We set $C^+ = n^-/n^+$ and $C^- = 1$, where n^- is the number of objects in the negative class and n^+ is the number of objects in the positive class. This essentially allows for fewer errors in the true positives class.

We use LIBSVM to efficiently solve the SKSVM problem.[27] The output of LIBSVM is a classifier model that contains the vector α and the bias b. A measured feature vector $\mathbf{x}$ can be classified by computing

$$y = \operatorname{sgn}\left[\sum_{i=1}^{n} \alpha_i y_i \kappa(\mathbf{x}_i, \mathbf{x}) - b \right], \tag{9}$$

where sgn is the signum function.

5.2 Multiple Kernel Learning Group Lasso Classifier

MKL extends the idea of kernel classification by allowing the use of combinations of multiple kernels. The kernel combination can be computed in many ways, as long as the combination results in a Mercer kernel.[28] In this paper we assume that the kernel K is composed of a weighted combination of pre-computed kernel matrices, i.e.,

$$K = \sum_{k=1}^{m} \sigma_k K_k, \tag{10}$$

where there are m kernels and σ_k is the weight applied to the kth kernel. The composite kernel can then be used in the chosen classifier model. For this paper, we will use the SVM. Thus, MKL SVM extends the SKSVM optimization at (7) by also optimizing over the weights σ_k,

$$\min_{\sigma\in\Delta} \max_{\alpha} \left\{ \mathbf{1}^T\alpha - \frac{1}{2}(\alpha\circ\mathbf{y})^T \left(\sum_{k=1}^{m} \sigma_k K_k \right) \alpha\circ\mathbf{y}) \right\}, \tag{11a}$$

subject to (typically)

$$0 \le \alpha_i \le C,\ i = 1,\ldots,n,\ \alpha^T\mathbf{y} = 0, \tag{11b}$$

where Δ is the domain of σ. Note that this is the *same* problem as SKSVM if the kernel weights are assumed constant.[29] This property has been used by many researchers to propose *alternating optimization* (AO) procedures for solving the min-max optimization problem. That is, solve the inner maximization for a constant kernel K, and then update the weights σ_k to solve the outer minimization, and repeat until convergence.

The domain of σ is very important and many MKL implementations only work for a single domain. For example, $\Delta = \{\sigma \in \mathbb{R}^m : \|\sigma\|_2 < 1, \sigma_k > 0\}$ is the ℓ_2-norm MKL.[30,31] We use a generalized MKL instantiation that allows for an ℓ_p-norm domain $\Delta = \{\sigma \in \mathbb{R}^m : \|\sigma\|_p < 1, \sigma_k > 0\}$.[32] We use the optimization procedure proposed by Xu et al. called MKL *group lasso* (MKLGL).[32] This method is efficient as it uses a closed form solution for solving the outer minimization in (11), i.e.,

$$\sigma_k = \frac{f_k^{2/(1+p)}}{\left(\sum_{k=1}^{m} f_k^{2p/(1+p)}\right)^{1/p}},\ k = 1,\ldots,m; \tag{12a}$$

$$f_k = \sigma_k^2(\alpha\cdot\mathbf{y})^T K_k(\alpha\cdot\mathbf{y}). \tag{12b}$$

We use a modified MKLGL algorithm which uses the SKSVM for unbalanced classes—i.e., we apply the constraints with C^+ and C^- as shown at (8). The MKLGL training algorithm is outlined in Alg. 1. The MKLGL is simple to implement and is efficient as the update equations for σ_k is closed-form. MKL can be thought of as a classifier fusion algorithm. It can find the optimal kernel among a set of candidates by tuning the weights on each kernel. The individual kernels can be computed in many ways—see our previous paper on this topic for more discussion on the formation of the kernel matrices.[19]

Algorithm 1: MKLGL Classifier Training[32]

Data: $(\mathbf{x}_i, y_i)$ - feature vector and label pairs; K_k - kernel matrices
Result: α, σ_k - MKLGL classifier solution
Initialize $\sigma_k = 1/m$, $k = 1,\ldots,m$ (equal kernel weights)
while *not converged* **do**
  Solve unbalanced SKSVM for kernel matrix $K = \sum_{k=1}^{m} \sigma_k K_k$
  Update kernel weights by eqs. (12)

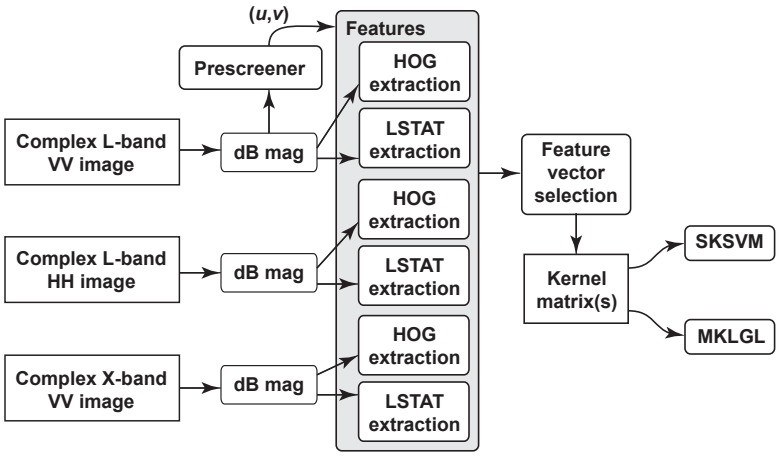


Figure 6. FA rejection classifier block diagram

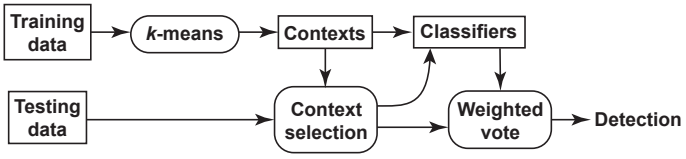


Figure 7. Context aware FA rejection classifier block diagram

5.3 FA Rejection

Figure 6 illustrates the FA rejection process. The FA rejection is accomplished by computing the HOG and LSTAT features in the two L-band and one X-band images for each detection location (where one of the sensors is chosen as the cueing sensor). We will use the L-band VV channel as the cueing sensor for the results in this paper. After the feature vectors are computed, we then train a FA rejecting classifier on the vector. The performance of the classifier is measured by applying the classifier to a testing lane that is geographically separated from the training lane. In this paper, we have two lanes: Lane A and Lane B. Results presented for Lane A are trained on Lane B and then tested on Lane A (and vice-versa for Lane B).

For the SKSVM classifier, the feature vector is created by appending all the features into one (long) vector and computing one kernel matrix. For the MKLGL classifier, the features can be combined in many different ways. One can take the one SKSVM feature vector and compute several kernel matrices as input, compute kernel matrices for each set of feature vectors (from each sensor or polarization, etc.), or some combination thereof.

5.3.1 Context aware FA rejection

Context-dependent classifier training has been proposed for many applications, including FA rejection in GPR.[33] The main idea is that contexts are regions or subspaces in the feature space that contain densely clustered feature vectors which are separated from other groups of feature vectors. We accomplish context aware FA rejection by the process illustrated in Fig. 7. First, features from the training lane are clustered into c contexts using the k-means clustering algorithm. A separate classifier is then trained on the features in each cluster, resulting in c classifiers, one for each context. The testing data are then classified by a weighted vote of the classifiers. First, the Euclidean distances to the c cluster centers are calculated for each feature vector. For the hard context classifier, the classifier is applied to the testing data that corresponds to the closest cluster center. For the soft context classifier, each testing feature vector is passed through all c classifiers and a weighted vote is applied, where the weights are the normalized distances to the cluster centers. In essence, a testing feature vector could be a mix of trained contexts; hence, the weighted vote allows the classifier from each context to be applied according to how well each feature matches with the contexts. We now turn to results of the described methods on two testing lanes.

6. RESULTS

The *area under ROC* (AUR) is used to show the relative performance of the different detection methods and classifiers we employ. This metric is calculated by normalizing the area under the *receiver operating characteristic* (ROC) curve for a

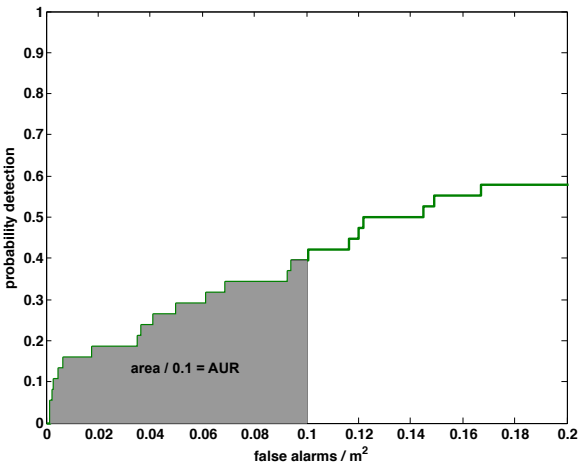


Figure 8. Illustration of AUR calculation

given method. Figure 8 illustrates the calculation of this metric for an example ROC curve. We chose a maximum FA per squared-meter rate (FAR) of 0.1 at which to limit the AUR calculation (this value was chosen because a FAR > 0.1 results in digging about every 1 meter down a 10 meter wide lane). The AUR equation is

$$AUR = \frac{1}{0.1}\int_0^{0.1} p_d(FAR)d(FAR), \tag{13}$$

where $p_d(FAR)$ is the probability of detection at a given FAR. The minimum AUR is 0, which indicates that no targets were detected, and the maximum AUR is 1, which indicates that all targets were detected with 0 FAR. For each ROC, we will also show the ROC curve and corresponding AUR of a uniform random detector, which is a detector such that one indicates hits at a predetermined uniform random spatial rate—the uniform random detector will be shown by a red dotted line.

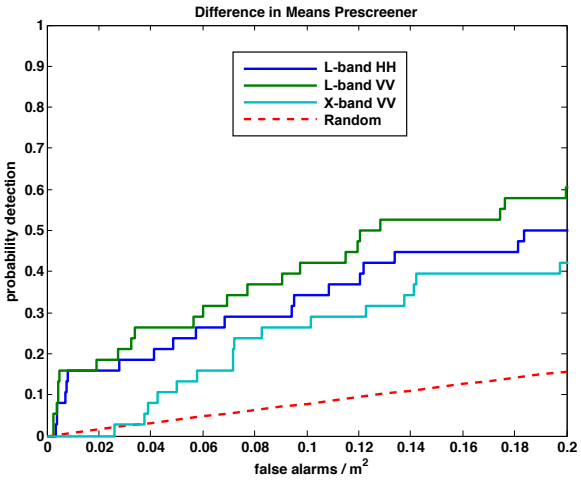
The miss-distance halo size used for these results is a 0.5 meter radius circle. In other words, if an indicated hit is within 0.5 meters of a ground-truth target location, then it is indicated as a hit. While this halo is larger than that used for downward-looking GPR systems—which tend to use < 0.25 meter halos—we believe that forward-looking systems and their inherent challenges because of the sensing geometry, ground-plane interaction, and lower SNR, require a larger halo. In essence, we are proposing a trade-off of lower accuracy of the hit location and a higher standoff distance to the detection.

6.1 Prescreener Results

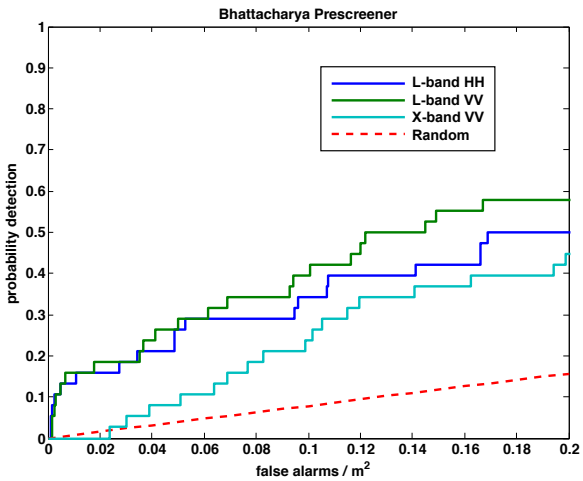
Figures 9 and 10 show the ROC curves for the prescreener algorithms on each lane for each polarization, and Table 3 provides the AUR results. View 9(a) shows the difference-of-means detector, while (b) shows the Bhattacharya detector. On Lane A, the best detector in terms of AUR was the difference-of-means detector using the L-band VV sensor. On Lane B, the best detector was the difference-in-means detector using the X-band VV sensor. This is actually a surprising result, as the X-band radar has a theoretically shallower penetration depth as compared to the L-band sensor. We believe it is possible that the X-band sensor on Lane B is detecting some other feature that indicates the presence of the target, perhaps a disturbed earth feature. Even so, we chose to the L-band VV sensor with the difference-in-means detector as the cueing sensor for the remaining results in this paper. Hence, the L-band VV difference-in-means ROC curve is the baseline performance to which the FA rejecting classifiers are compared.

6.2 Classifier Results

Table 4 provides the results of the FA rejection classifiers. The cueing detector used in these experiments was the L-band VV difference-in-means detector. For the Lane A results, the classifier was trained on Lane B, and vice-versa for Lane B results. The classifiers used here were SKSVM and MKLGL classifiers, modified for unbalanced training data as outlined

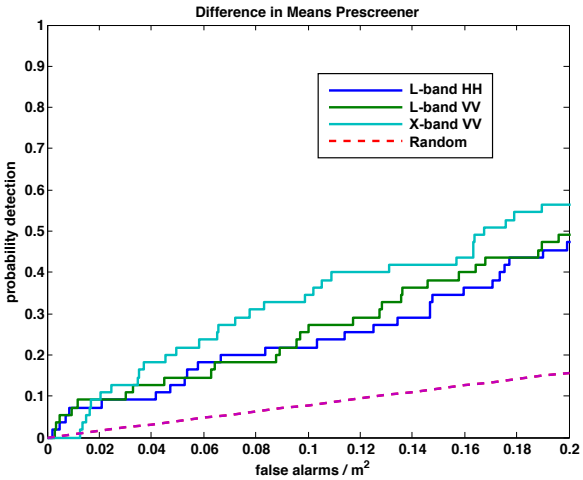


(a) Difference-of-means detector

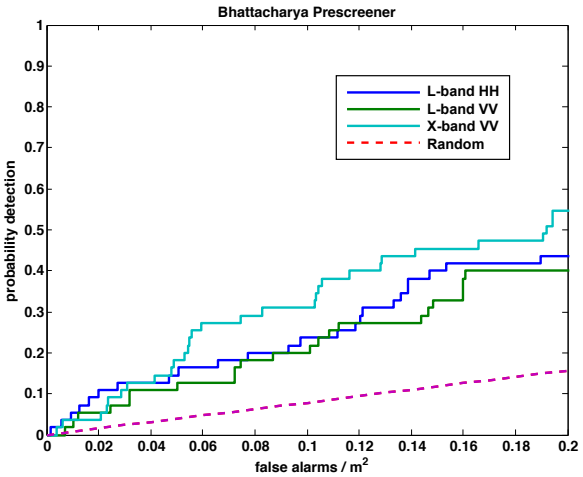


(b) Bhattacharya detector

Figure 9. ROC curves of prescreener algorithms on Lane A



(a) Difference-of-means detector



(b) Bhattacharya detector

Figure 10. ROC curves of prescreener algorithms on Lane B

Table 3. AUR values for prescreener algorithms

Lane A	**Difference-in-means AUR**	**Bhattacharya AUR**
L-band HH	0.22	0.23
L-band VV	0.27	0.26
X-band VV	0.12	0.10
Random	0.04	0.04
Lane B		
L-band HH	0.14	0.14
L-band VV	0.14	0.11
X-band VV	0.19	0.18
Random	0.04	0.04

Table 4. Percent AUR improvement for FA rejection classifiers.

			AUR improvement		
Classifier	**Kernel(s)**	**Features**	**Sensor**	**Lane A**	**Lane B**
SKSVM	linear	HOG	L	27%	110%
		HOG	LX	31%	114%
		HOG, LSTAT	L	3%	129%
		HOG, LSTAT	LX	24%	128%
SKSVM	RBF(1)	HOG	L	30%	137%
		HOG	X	0%	125%
		HOG	LX	64%	143%
		HOG, LSTAT	LX	-69%	-100%
SKSVM	poly(2)	HOG	LX	42%	116%
	poly(3)			46%	117%
	poly(4)			50%	120%
MKLGL	$\mathcal{K}_1$	HOG	LX	**67%**	140%
	$\mathcal{K}_1$	HOG, LSTAT		-45%	-12%
	$\mathcal{K}_2$	HOG		57%	129%
	$\mathcal{K}_1 \cup \mathcal{K}_2$	HOG		57%	129%
	$\mathcal{K}_1 \cup$ linear	HOG		34%	150%
	$\mathcal{K}_1 \cup$ linear	HOG, LSTAT		29%	148%
MKLGL*	HOG: $\mathcal{K}_1 \cup$ linear	HOG, LSTAT	LX	44%	157%
	LSTAT: $\mathcal{K}_1 \cup$ linear				
MKLGL*	HOG: $\mathcal{K}_1 \cup$ linear	HOG, LSTAT	LX	44%	**158%**
	LSTAT: linear				
MKLGL*	HOG: $\mathcal{K}_1 \cup$ linear	HOG, LSTAT	LX	-73%	-12%
	LSTAT: $\mathcal{K}_2 \cup$ linear				

*HOG and LSTAT kernels are separate.

previously. The kernels used were a linear kernel, $K_{ij} = \mathbf{x}_i^T \mathbf{x}_j$, an RBF($\gamma$) kernel, $K_{ij} = \exp\{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\}$, and a polynomial($d$) (poly) kernel, $K_{ij} = (\mathbf{x}_i^T \mathbf{x}_j + 1)^d$. The kernel sets used for MKLGL were

$$\mathcal{K}_1 = \{\text{RBF(0.001), RBF(0.01), RBF(0.1), RBF(1), RBF(10)}\};$$
$$\mathcal{K}_2 = \{\text{poly(2), poly(3), poly(4)}\}.$$

The results shown are in terms of AUR improvement over the prescreening detector, which is calculated as:

$$\text{AUR improvement} = 100\% \cdot \left(\frac{\text{AUR}_{classifier}}{\text{AUR}_{prescreener}} - 1 \right), \tag{14}$$

where an AUR improvement of 0% indicates no change in detection statistics.

The best overall classifiers for each lane are bolded in Table 4, which for Lane A was the MKLGL classifier that uses the $\mathcal{K}_1$ set of kernels with the HOG feature, and for Lane B was the MKLGL that uses separate kernels for the HOG and LSTAT features. Interestingly, the MKLGL classifier using the separate kernels for HOG and LSTAT provides a 158% AUR improvement for Lane B; however, this classifier only provides a 44% improvement for Lane A. The overall best classifiers seem to be the MKLGL using the HOG:$\mathcal{K}_1$ set of kernels and the SKSVM RBF(1) HOG. These classifiers are the overall best on Lane A and still provide very good AUR improvement on Lane B. The MKLGL using separate RBF kernels for the HOG and LSTAT features also provides very good results.

When examining the features, the LSTAT feature is clearly inferior to the HOG feature for the higher-dimensional kernel classifiers. However, the addition of the LSTAT feature to the SKSVM linear kernel classifier gives a better AUR improvement for Lane B. However, this is the *only* case where AUR improvement is better than just using the HOG feature. Another thing to notice is that the MKLGL is able to handle the addition of the LSTAT feature and still produce some decent results, as long as the linear kernel is selected as a member of the kernel set. This is a strength of the MKLGL is that it is able to lessen the impact of the less effective features, in essence, performing in situ feature selection.

Table 5. Percent AUR improvement for FA rejection classifiers. Context is detected from feature vectors.

				AUR improvement			
				Lane A		Lane B	
Classifier	**Kernel(s)**	**Features**	**Contexts**	**Hard**	**Soft**	**Hard**	**Soft**
SKSVM	linear	HOG	3	29%	19%	83%	103%
SKSVM	RBF(1)	HOG	2	39%	**68%**	139%	123%
			3	52%	26%	125%	**152%**
			4	-100%	-46%	-99%	11%
			5	-99%	-48%	-86%	66%
MKLGL	HOG: linear LSTAT: linear	HOG, LSTAT	3*	15%	29%	66%	83%
MKLGL	HOG: linear LSTAT: linear	HOG, LSTAT	3**	-10%	12%	59%	124%
MKLGL	HOG: $\mathcal{K}_1$ LSTAT: linear	HOG, LSTAT	3*	39%	11%	79%	70%
MKLGL	HOG: $\mathcal{K}_1$ LSTAT: linear	HOG, LSTAT	3**	-21%	7.5%	57%	126%

*Indicates that context was detected on HOG feature. **Indicates that context was detected on combination of HOG and LSTAT.

One of the challenges in training a classifier for these data is that nearly all the classifiers listed in Table 4 produced training AURs of 1, meaning that the classifier had perfect FA rejection on the training lane. This can result in an uninformative training data set which can lead to overtraining and a poor testing performance. This is especially seen in the classifiers that produced negative AUR improvements. These classifiers simply did not generalize well to the test data set.

6.3 Context Aware Classifier Results

Table 5 contains the resulting AUR improvement values for the context aware classifiers on testing Lanes A and B (Lane B is trained with Lane A and vice-versa). Again, the cueing detector is the L-band VV image. The number of contexts (clusters) used were 2-5, with 3 showing the overall best performance. Overall, the context aware approach does not give a drastic improvement. It is able to produce the best AUR improvement in the Lane A test, but only by a slight margin. The results on Lane B for the RBF(1) kernel, 3 contexts, and soft vote at 152% is near to the best seen in the MKLGL classifier in Table 4, but the improvement seen in the corresponding Lane A data is far from the mark.

We believe that more data are needed to properly train the context-aware classifier. One problem we encountered is that each of the clusters in the training data may not contain data with both class labels, which eliminates the chance to train an effective SKSVM or MKLGL classifier. Furthermore, the trained classifiers were over specified by the context aware clustering process. Because there were so few data in each cluster, the SVM classifiers were too easily able to find a separating hyperplane. However, this does not generalize well to the training data, which results in the negative AUR improvement scores seen in the table. While the results in this table do not support the idea of using the context aware classifier, we believe that this conclusion is premature and more data need to be considered before a proper justification for the idea can be made.

7. CONCLUSION AND FUTURE WORK

In this paper, we showed how L-band and X-band FLGPR can be fused to improve the detection statistics of individual detectors. We tested several types of classifier designs and showed that that both SKSVM and MKLGL in concert with the HOG feature have promise for significantly reducing FAs. The weakest element of our detection process thus far is the FLGPR images themselves. The prescreener is only able to achieve very modest performance, which is clearly lower than acceptable for operational capability. To combat this weakness, we are investigating methods for improving the standard backpropagation code and coherent integration process. We aim to improve by operating directly on the FLGPR frames, before coherent integration occurs. In essence, the "stack" of data frames at each location can be considered a feature vector itself and this idea will be considered in the next installment of this work.

7.1 Future Work

It is our conjecture that there is information that is not currently being used by standard magnitude-based feature descriptors. We believe that there is information contained in the coherence over multiple scans of the FLGPR array. Hence, we are going to investigate how feature representation can be learned for the step before coherent integration in the FLGPR imaging process. While this will negate the improvement of SNR and resolution gained by coherently integrating multiple scans over the forward motion of the vehicle, we believe that there may be target characteristics contained in the spatial coherence of multiple scans that will enable a better detection rate than standard magnitude-only detection. Furthermore, the scan-based detection could be fused with the magnitude-based detection to find those targets that are better found by coherent integration of multiple scans.

Second, we will continue to investigate better methods for producing the radar images, examining multiple methods for combining multiple scans from the forward motion of the vehicle, such as *recursive sidelobe minimization* (RSM) and other statistical methods (other than the coherent average that we are currently using).[20] Another avenue for improved localization of the scatterers is the $\ell 1$-minimization idea popular in compressive sensing.[34] The idea here is that one assumes that the signal producing the radar return is a sparse signal, i.e., the scene contains a small number of scatterers. Hence, we are investigating $\ell 1$ methods which are both effective and computationally efficient.

Third, we plan to examine other methods for multiple kernel learning, including our recent algorithm which uses a non-linear aggregation to produce the overall kernel.[35] It was shown in our work on this topic that our Choquet integral approach to MKL was more effective than MKLGL on several publicly-available benchmark data sets. Hence, we believe that we can achieve even better performance with our non-linear aggregation method. We will also investigate the addition of other shape and texture features, such as ensembles of HOGs, local-binary-patterns, and shearlets.[19,36]

Last, we will extend the analyses presented in this paper, integrate our future work ideas, and process more passes of the FLGPR arrays to get a more comprehensive understanding of the overall performance in different scenarios and against different types of targets. We also anticipate that using more training data will improve overall detection statistics.

ACKNOWLEDGMENTS

This work has been funded in part by U.S. Army grants W909MY-13-C-0013 and W909MY-13-C-0029. We would like to acknowledge U.S. Army RDECOM CERDEC NVESD for vital support in this project.

REFERENCES

[1] JIEDDO COIC MID, "Global IED monthly summary report," (August 2012).

[2] Cremer, F., Chavemaker, J. G., deJong, W., and Schutte, K., "Comparison of vehicle-mounted forward-looking polarimetric infrared and downward-looking infrared sensors for landmine detection," in [*Proc. SPIE*], **5089**, 517–526 (2003).

[3] Playle, N., Port, D. M., Rutherford, R., Burch, I. A., and Almond, R., "Infrared polarization sensor for forward-looking mine detection," in [*Proc. SPIE*], **4742**, 11–18 (2002).

[4] Costley, R. D., Sabatier, J. M., and Xiang, N., "Forward-looking acoustic mine detection system," in [*Proc. SPIE*], **4394**, 617–626 (2001).

[5] Collins, L. M., Torrione, P. A., Throckmorton, C. S., Liao, X., Zhu, Q. E., Liu, Q., Carin, L., Clodfelter, F., and Frasier, S., "Algorithms for landmine discrimination using the NIITEK ground penetrating radar," in [*Proc. SPIE*], **4742**, 709–718 (2002).

[6] Gader, P. D., Grandhi, R., Lee, W. H., Wilson, J. N., and Ho, K. C., "Feature analysis for the NIITEK ground penetrating radar using order weighted averaging operators for landmine detection," in [*Proc. SPIE*], **5415**, 953–962 (2004).

[7] Bradley, M. R., Witten, T. R., Duncan, M., and McCummins, R., "Anti-tank and side-attach mine detection with forward-looking GPR," in [*Proc. SPIE*], **5415**, 421–432 (2004).

[8] Cosgrove, R. B., Milanfar, P., and Kositsky, J., "Trained detection of buried mines in SAR images via the deflection-optimal criterion," *IEEE Trans. Geoscience and Remote Sensing* **42**(11), 2569–2575 (2004).

[9] Sun, Y. and Li, J., "Plastic landmine detection using time-frequency analysis for forward-looking ground-penetrating radar," in [*Proc. SPIE*], **5089**, 851–862 (2003).

[10] Stone, K., Keller, J. M., Ho, K. C., and Gader, P. D., "On the registration of FLGPR and IR data for the forward-looking landmine detection system and its use in eliminating FLGPR false alarms," in [*Proc. SPIE*], **6953** (2008).
[11] Stone, K., Keller, J. M., Popescu, M., Havens, T. C., and Ho, K. C., "Forward-looking anomaly detection via fusion of infrared and color imagery," in [*Proc. SPIE*], **7664**, 766425 (2010).
[12] Havens, T. C., Stone, K., Keller, J. M., and Ho, K. C., "Sensor-fused detection of explosive hazards," in [*Proc. SPIE*], **7303**, 73032A (2009).
[13] Havens, T. C., Ho, K. C., Farrell, J., Keller, J. M., Popescu, M., Ton, T. T., Wong, D. C., and Soumekh, M., "Locally adaptive detection algorithm for forward-looking ground-penetrating radar," in [*Proc. SPIE*], **7664**, 76442E (2010).
[14] Havens, T. C., Spain, C. J., Ho, K. C., Keller, J. M., Ton, T. T., Wong, D. C., and Soumekh, M., "Improved detection and false alarm rejection using ground-penetrating radar and color imagery in a forward-looking system," in [*Proc. SPIE*], **7664**, 76441U (2010).
[15] Havens, T. C., Keller, J. M., Ho, K. C., Ton, T. T., Wong, D. C., and Soumekh, M., "Narrow band processing and fusion approach for explosive hazard detection in FLGPR," in [*Proc. SPIE*], **8017**, 80171F (2011).
[16] Farrell, J., Havens, T. C., Ho, K. C., Keller, J. M., Ton, T. T., Wong, D. C., and Soumekh, M., "Detection of explosive hazards using spectrum features from forward-looking ground penetrating radar imagery," in [*Proc. SPIE*], **8017**, 80171F (2011).
[17] Farrell, J., Havens, T. C., Ho, K. C., Keller, J. M., Ton, T. T., Wong, D. C., and Soumekh, M., "Evaluation and improvement of spectral features for the detection of buried explosive hazards using forward-looking ground-penetrating radar," in [*Proc. SPIE*], **8357**, 8357C (2012).
[18] Perconti, P., Hodapp, J., Jellish, S., Marshall, C., and Hilger, J., "Counter IED graphical user interface," tech. rep., U.S. Army RDECOM CERDEC Night Vision and Electronic Sensors Directorate (2007).
[19] Havens, T. C., Keller, J. M., Stone, K., Ho, K. C., Ton, T. T., Wong, D. C., and Soumekh, M., "Multiple kernel learning for explosive hazards detection in FLGPR," in [*Proc. SPIE*], **8357**, 83571D (2012).
[20] Mnguyen, L. and Innocenti, R., "Suppression of sidelobes and noise in airborne SAR imagery using recursive sidelobe minimization technique," in [*Proc. IEEE Radar Conference*], 522–525 (May 2010).
[21] Richardson, W. H., "Bayesian-based iterative method of image restoration," *J. Optical Society of America* **62**(1), 55–59 (1972).
[22] Lucy, L. B., "An iterative technique for the rectification of observed distributions," *Astronomical J.* **79**(6), 745–754 (1974).
[23] Dalal, N. and Triggs, B., "Histogram of oriented gradients for human detection," in [*IEEE Conf. Computer Vision and Pattern Recognition II*], 886–893 (2005).
[24] Duda, R., Hart, P., and Stork, D., [*Pattern Classification*], Wiley-Interscience, second ed. (October 2000).
[25] Cortes, C. and Vapnik, V. N., "Support-vector networks," *Machine Learning* **20**(3), 273–297 (1995).
[26] Boser, B. E., Guyon, I. M., and Vapnik, V. N., "A training algorithm for optimal margin classifiers," in [*ACM Workshop on COLT*], 144–152 (1992).
[27] Chang, C. C. and Lin, C. J., "LIBSVM: a library for support vector machines," *ACM Trans. Intell. Sys. Tech.* **2**(3), 1–27 (2011).
[28] Mercer, J., "Functions of positive and negative type and their connection with the theory of integral equations," *Philosophical Trans. Royal Society A* **209**, 441–458 (1909).
[29] Lanckriet, G. R. G., Cristianini, N., Bartlett, P., Ghaoui, L. E., and Jordan, M. I., "Learning the kernel matrix with semidefinite programming," *J. Machine Learning Research* **5**, 27–72 (2004).
[30] Kloft, M., Brefeld, U., Laskov, P., and Sonnenburg, S., "Non-sparse multiple kernel learning," in [*NIPS Workshop on Kernel Learning: Automatic Selection of Optimal Kernels*], (2008).
[31] Cortes, C., Mohri, M., and Rostamizadeh, A., "ℓ_2 regularization for learning kernels," in [*Proc. Conf. Uncertainty in Artificial Intelligence*], 187–196 (2009).
[32] Xu, Z., Jin, R., Yang, H., King, I., and Lyu, M. R., "Simple and efficient multiple kernel learning by group lasso," in [*Proc. Int. Conf. Machine Learning*], 1175–1182 (2010).
[33] Ratto, C. R., Torrione, P. A., and Collins, L. M., "Exploiting ground-penetrating radar phenomenology in a context-dependent framework for landmine detection and discrimination," *IEEE Trans. Geoscience and Remote Sensing* **49**(5), 1689–1700 (2011).

[34] Yang, J., Jin, T., Huang, X., Thompson, J., and Zhou, Z., "Sparse MIMO array forward-looking GPR imaging based on compressive sensing in clutter environment," *IEEE Trans. Geoscience and Remote Sensing* **52**(7), 4480–4494 (2014).

[35] Hu, L., Anderson, D. T., and Havens, T. C., "Multiple kernel aggregation using fuzzy integrals," in [*Proc. IEEE Int. Conf. Fuzzy Systems*], 1–7 (July 2013).

[36] Tuomanen, B., Stone, K., Madison, T., Popescu, M., and Keller, J. M., "Buried target detection in FLIR images using shearlet features," in [*Proc. SPIE*], **8709**, 870919 (2013).