Adaptive Learning Approach to Landmine Detection

Yijun Sun and Jian Li, *Fellow, IEEE*

***Abstract*— We consider landmine detection using forward-looking ground penetrating radar. The two main challenging tasks include extracting intricate structures of target signals and adapting a classifier to the surrounding environment through learning. Through the time-frequency analysis, we find that the most discriminant information is time-frequency localized. This observation motivates us to use the overcomplete wavelet packet transform to sparsely represent signals with the discriminant information encoded into several bases. Then the sequential floating forward selection method is used to extract these components and thereby a neural network classifier is designed. To further improve the classification performance and deal with the problem of detecting mines in an unconstraint environment, the AdaBoost algorithm is used. We integrate the feature selection process into the original AdaBoost algorithm. In each iteration, AdaBoost identifies the hard-to-learn examples and a new set of features which provide the specific discriminant information for these hard samples is extracted adaptively and a new classifier is trained. Experimental results based on measured data are presented, showing that a significant improvement on the classification performance can be achieved.**

***Index Terms*— Ground Penetrating Radar (GPR), landmine detection, time-frequency analysis, wavelet packet transform, pattern classification, boosting, feature selection, machine learning, neural network.**

I. INTRODUCTION

Landmines are causing enormous humanitarian and economic problems in many countries all over the world. Experts estimate that up to 110 million landmines need to be cleared and more than $20,000$ civilians are killed or maimed every year by landmines, with many of the victims being children [1]. However, landmine detection and clearance have turned out to be an extremely challenging problem. At the current clearance rate, it will take about $1,000$ years to remove all landmines that are already placed and for every landmine cleared, further 20 are being buried. Therefore it is urgent to develop a safe and cost efficient landmine detection system. In the past fifteen years, various techniques, including acoustic sensor, infrared technique, quadrupole resonance and down- and forward-looking ground penetrating radar (FLGPR), have been investigated. Among them, FLGPR has several advantages over others, including long standoff distances and fast interrogation of a large area, and thus is considered a viable technology for landmine detection [2], [3].

In this paper, we consider landmine detection using forward-looking ground penetrating radar. As shown in Figure 1, this system has GPR antennas mounted on the front of a vehicle and captures radar signals as the vehicle moves forward. Detailed descriptions of the system can be found in [2]. With the use of the receiver antenna array, a high resolution radar image can be formed from received signals. Our task is to detect the presence of landmines in radar images. It can in general be formulated as an object recognition problem. The conventional signal detection techniques such as the matched filter method may not be applicable here since it is very difficult, if not impossible, to estimate the target signatures as well as the clutter statistical properties due to the extremely complicated operating environments. One possible method to overcome this problem is to design a classifier through learning, which is often used in the problem of detecting faces or pedestrians in highly cluttered scenes [4], [5]. However, compared to the conventional object recognition problem, landmine detection has its own specific challenges. First, in contrast to face recognition and character recognition where intensity images provide us with an abundant amount of human recognizable information, radar images can only be fully understood through the analysis of radar scattering phenomena. Based on the object size, physical structures and composition materials, different objects react to incident radar signals differently [6],[7],[8]. Hence, how to quantify these differences and thereby design a classifier becomes the key to the success of landmine detection. In the context of pattern classification, this process is referred to as feature extraction. One may bypass this stage by using some special classifiers (for example, neural networks), which have a feature extractor implicitly embedded in the classifier structure. In our case of a limited number of training samples (in particular mine samples), however, this scheme may not be applicable. Though numerical simulations may help us identify these useful information, given the extremely complicated surveying scenarios, the usefulness of simulated information is very limited and we must resort to a training based method which is capable of automatically extracting intricate structures of target signals through learning. The second challenge in landmine detection is that, in contrast to the conventional pattern classification problem where we only need to decide between well-defined classes, a landmine detector is required to differentiate mines from the rest of the world. While the mine samples are well-defined, there are no typical examples for clutter. In other words, clutter can be anything other than mines. It requires us to design a classifier which has sufficient expression power to claim a region in the high dimensional feature space for mines. Note that this requirement is in conflict with the previous process of feature extraction in some ways: we want to reduce the data dimensionality to improve the classifier generalization capability over unseen samples; yet at the same time, the classifier should have sufficient expression power to attain a low training error.

Manuscript received in May 14, 2004; released for publication January 18, 2005. This work is in part supported by the U.S. Army under contract No. DAAB15-00-C-1024.

Y. Sun and J. Li are with the Department of Electrical and Computer Engineering, P. O. Box 116130, University of Florida, Gainesville, FL 32611, USA. Phone: (352) 392-2642. Fax: (352) 392-0044. E-mail: li@dsp.ufl.edu.

(a) (b) (c)

Fig. 1. (a) An illustration of the SRI FLGPR system. (b) A photograph of the SRI FLGPR system. (c) A typical radar image for mines. The term "M-3" denotes a metal mine buried at the depth of 3 inches.

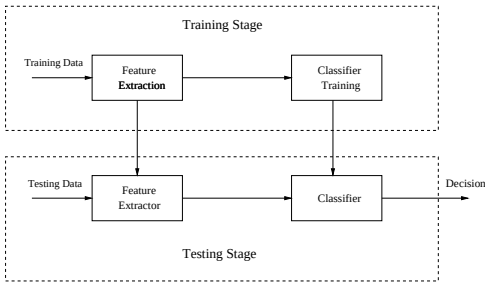


Fig. 2. A conventional pattern classifier design procedure.

A conventional classifier design procedure usually consists of two modules: feature extraction and classifier training (Figure 2). This paper covers every aspect of designing a classifier. To start the work, the first important thing we need to do is to find out what kinds of features we can extract. Toward this end, we have introduced the time-frequency analysis into the landmine detection [3]. Through the time-frequency analysis, we can obtain graphical understanding on how different objects react to the incident radar signals differently. We find that the most discriminant information between the two classes are time-frequency localized. This observation motivates us to propose a wavelet packet transform (WPT) based detector. Wavelet packet transform is used to encode the time-frequency localized information efficiently into several bases and then a feature selection method is used to find these components by optimizing a certain cost function. With these selected features, a neural network classifier with a simple structure is designed.

However, the conventional classifier design procedure cannot solve the aforementioned issues of detecting targets in unconstrained environments. To our knowledge, this problem is not fully addressed in the literature and even less in the radar community. For example, in feature selection, the features so-produced are aimed at optimizing a certain feature selection criterion with a predefined number of features based on the entire training data set and thus are not in favor of some particular training samples. For this reason, we refer to these features as the global features. Concerning over the problem of the unconstrained clutter samples, we need to utilize the local information in the data space, but not at the cost of decreasing the classifier generalization capability. In the paper, we show that this problem can be alleviated by using the boosting method. In particular, AdaBoost with soft decisions is used. The main idea of AdaBoost is to train an ensemble of classifiers sequentially with the subsequent classifiers focusing on the errors made by the previous ones. Hence the boosting method provides us with a way to identify the hard examples of separating mines from clutter. With these examples, a new set of features which provide the specific discriminant information for the misclassified samples can be extracted adaptively and a new classifier can be trained. The final decision is calculated as the weighted combination of the decisions of the member classifiers. The experimental results based on the measured data show that with this classification scheme, significant improvements on both the training and the testing performances can be achieved while at the same time no apparent overfitting is observed.

The remainder of the paper is organized as follows. In Section II, we give a brief description on the SRI (Stanford Research Institute) FLGPR system and present the time-frequency analysis for mine and clutter signals. Based on these discussions, in Section III, we propose a wavelet packet transform based landmine detector. We give a detailed discussion on how WPT can be used to extract localized information and how AdaBoost can be used to significantly improve the detection performance. The effectiveness of this landmine detector is demonstrated in Section IV based on the SRI measured data. Finally, we conclude our work in Section V.

II. System Description and Problem Formulation

A. System Description

A photograph of the SRI FLGPR system is shown in Figure 1(b). This system consists of 2 transmitter and 18 receiver quad-ridged horn antennas. The height of the transmitter antennas (two large horns) is about 3.3 m above the ground and their geometry center is 3.03 m away from each other. The 18 receiver antennas are horizontally equally spaced with 17 cm center to center and the height of the bottom row is about 2 m above the ground. The ultra wideband stepped frequency system operates at 1024 discrete frequencies evenly spaced over the frequency range from 442.5 to 3000 MHz with a step size of 2.5 MHz starting from the lowest frequency. The two transmitter antennas work sequentially and all the

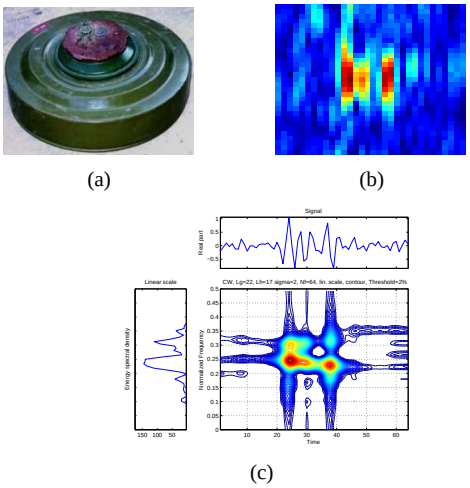


Fig. 3. Mine pictures, the corresponding radar image chips and the time-frequency representations. The mine is buried at the depth of 3 inches.

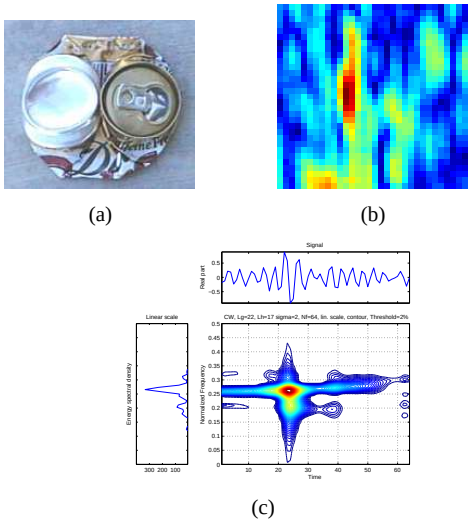


Fig. 4. Smashed coke can picture, the corresponding radar image chip and time-frequency representation.

receiver antennas work simultaneously. Hence there is a total of 36 channels of the received signal for each scan or vehicle location. Data are recorded while the vehicle moves forward and the distance between two adjacent scans is about 2 m. The GPS (Global Positioning System) is used to measure the location of the system for each scan. With the use of the delay-and-sum imaging algorithm, a high resolution radar image can be formed from received signals. At each scan location, the image region is 6 m (cross-range) by 30 m (down-range) with a 7 m standoff distance ahead of the vehicle (Figure 1(a)). A pixel spacing of 4 cm is used in both the down-range and cross-range dimensions. In the experiment, we only use the images ranging from 10 m to 20 m ahead of the vehicle. The system is optimized for this range.

In Figures 3 and 4, we present several radar image chips for clutter and two types of buried metal mines, denoted by M1 and M2, respectively. The image chip has 32×32 pixels. One may start the classifier design with the 2D image chips directly. In our case of a limited number of training samples (in particular mine samples, cf., Section IV), however, it will easily lead to overfitting. We observe that most of the target information is embedded along the down-range direction due to a much higher resolution in the down-range direction than that of the cross-range direction (Figure 5) which is determined by the aperture size and wavelength. Therefore, we decide to use the appearance based Fisherface method [9], [10] as a prescreener to exploit the global information and then further check the down-range profile through the center of each image chip for the final decision, the latter of which is the main focus of this paper.

B. Time-Frequency Analysis

To obtain a good understanding on the scattering phenomena of landmines, we have introduced the time-frequency (TF) technique into the landmine detection application [3]. In particular, the Choi-Williams distribution (CWD) was found to be appropriate for visualizing the time-frequency representation in this application. Mathematically, the CWD can be expressed

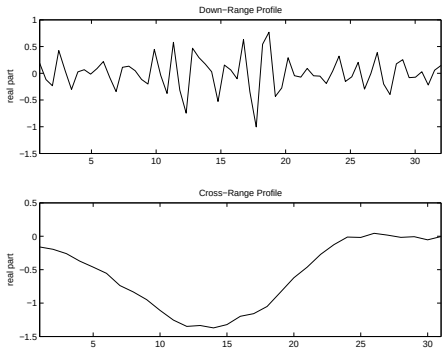


Fig. 5. The down-range and cross-range profiles of a metal mine. The down-range profile contains much more information than the cross-range profile.

as:

$$CWD(t,f) = \int\int\int \Phi(\nu,\tau)s(u+\tau/2)s^*(u-\tau/2)\cdot \cdot \exp^{j2\pi\nu(t-u)} \exp^{-j2\pi f\tau}\, d\nu d\tau du \quad (1)$$

where $s(t)$ is the signal of interest, $\Phi(\nu,\tau) = \exp^{-\alpha(2\pi\nu\tau)^2}$ is the kernel function to reduce highly oscillatory cross terms, and α is the parameter to control the spread of the kernel. (See, for example, [11], [12] for more detailed discussions.) From Eq. (1), we can see that after the CWD transform, a one-dimensional signal is transformed into a two-dimensional image over the time-frequency plane. In the radar applications, these images can greatly facilitate our understanding and interpretation of the different scattering phenomena of targets. In Figure 3 we present several TF representations for two types of mines, denoted as M1 and M2, both buried at the depth of 3 inches. The time domain signals, the power spectral densities and the metal mine pictures are also presented. These TF representations can be roughly interpreted as follows: the front and rear edges of the mines can be modelled as two scattering centers, with each edge serving as a discrete event in time. In the time-frequency domain, each scattering center shows up as a vertical line in the image since it occurs at a particular time instant but over all frequencies [6]. The edges thus can

be used as salient features for the discrimination of mines from clutter and even possibly for the discrimination among different types of mines. Another feature in the time-frequency domain is the discrete events in frequency, which is due to stronger responses of a target to certain stepped frequencies within the radar frequency band, and shows up in the time-frequency domain as horizontal lines. For comparison, we also present two clutter image chips and the corresponding TF representations in Figure 4. It should be noted that these two samples are by no means typical examples for clutter.

Through the time-frequency analysis, we can get a graphical understanding on how a mine reacts to incident radar signals. A question arises naturally: how can we extract these time-frequency localized information efficiently for signal classification? We may not be able to use the CWD directly since after the CWD transform, a TF image contains too much redundant information as well as cross terms. One widely used TF technique for signal classification purposes is the discrete wavelet transform (DWT). An apparent drawback associated with DWT is the poor frequency resolution at the high frequency range and the poor time resolution at the low frequency range. When used for classification applications, it may have difficulties in identifying desired discriminant features at the needed ranges. Wavelet packet transform, on the other hand, uses a rich library of redundant bases with arbitrary TF resolution [13] and therefore, compared with DWT, WPT is more suitable for extracting features from signals having nonstationary as well as stationary components. We will give detailed discussions below on how the wavelet packet transform can be used to extract the information, based on which a classifier can be designed.

III. Landmine Classifier Design

A. Wavelet Packet Transform

Wavelet packet transform, which is an important generalization of DWT, provides a much more flexible signal decomposition scheme than DWT. Like DWT, wavelet packet basis functions are also formed by scaling and translating a family of basis functions:

$$w_{j,b,k}(t) = 2^{-j/2} w_b(2^{-j}t - k), j, k \in \mathbb{Z} \tag{2}$$

where $\mathbb{Z}$ is the set of all integers. However, for WPT, in addition to the scaling parameter j and translation parameter k, there is also an oscillation parameter b, with a larger b corresponding to a higher frequency. A father wavelet $\phi(t)$ and a mother wavelet $\psi(t)$ correspond to w_b with b equal to 0 and 1, respectively: $w_0(t) = \phi(t), w_1(t) = \psi(t)$. The rest of the wavelet packet functions $w_b(t)$, $b = 2, 3, \cdots$, are defined as:

$$w_b(t) = \sqrt{2} \sum_k f^b(k) w_{\lfloor b/2 \rfloor}(2t - k) \tag{3}$$

where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. The filter $f^b(k)$ is either a lowpass or a highpass filter depending on the value of b [12]:

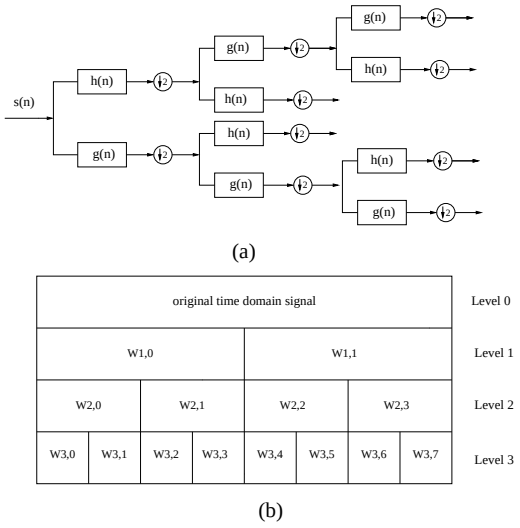
$$f^b(k) = \begin{cases} g(k) & \text{if } b \bmod 4 = 0 \text{ or } 3 \\ h(k) & \text{if } b \bmod 4 = 1 \text{ or } 2 \end{cases} \tag{4}$$

where $g(k)$ and $h(k)$ are the lowpass and highpass quadrature mirror filters (QMF) associated with the mother wavelet functions. Using the wavelet basis functions, the WPT coefficients can then be calculated as the inner product between a signal and the corresponding wavelet basis function:

$$\text{WPT}(j, b, k) = \int s(t) w_{j,b,k}(t) dt \tag{5}$$

The wavelet packet decomposition scheme may be better understood with the aid of the two-channel subband coding scheme which is used to implement DWT. Compared with DWT where at each level only the lower halfband signal is further decomposed, WPT decomposes the higher halfband signal as well (Figure 6(a)). If we retain all of the transform coefficients and stack them together in the order of the level, a wavelet packet table is constructed. Suppose we have a signal $s(n)$ of length of L, with L being a multiple of 2^J. The wavelet packet table then has $J + 1$ levels, where J is the maximum possible resolution level. At the resolution level j, the table has L coefficients, divided into 2^j coefficient blocks indexed by j and b, and usually named as a *node*: $\mathbf{w}_{j,b} = [\ w_{j,b,1} \quad w_{j,b,2} \quad \cdots \quad w_{j,b,L/2^j}\ ]$. Figure 6(b) shows a layout of a wavelet packet table with 3 resolution levels. The level 0 corresponds to the original signal. We see that after WPT, a signal of length L ends up with a maximum of $J \times L$ coefficients, indicating WPT is an overcomplete transform. Starting from this table, a particular set of coefficients can be selected to form a complete and orthogonal transformation, one of which is DWT by retaining the coefficients in the nodes of $\mathbf{w}_{1,1}, \mathbf{w}_{2,1}, \mathbf{w}_{3,0}$ and $\mathbf{w}_{3,1}$. In general, the selection of the bases is usually accompanied by the optimization of a certain cost function, known as the best basis method [13],[14],[15]. Two algorithms in particular for signal classification, refereed to as LDB (Local Discriminant Basis) [14] and the separability-based multiscale basis selection algorithm [15], were proposed aiming at selecting a complete and orthognal bases based on a certain cost function. However, for signal classification, we can argue that there is no need for the sought bases to be complete or orthogonal. We only need to determine the components that most efficiently encode the discriminant information among signal classes. In this way, the best basis selection process can be directly casted into a feature selection problem.

Before proceeding, we present a toy example to show how WPT can be used to extract localized features. We construct the full feature set from the wavelet table by squaring the WPT coefficients. This feature has an intuitive physical meaning. Since each wavelet basis function has its corresponding coverage in the TF plane, it can be viewed as a "window" through which we observe a signal. The value of a feature is then the energy of a signal through that "window". The way the wavelet basis functions interpret a signal is analogous to the way we humans do toward a signal's TF image. To be more clear, let us revisit the problem of extracting time-frequency localized information posed in the previous section. If we put some changeable "windows" on the image, we can easily describe the contents of the image with only a few WPT coefficients (Figure 7). Of course, the size of a window should comply



(a)

(b)

Fig. 6. (a) Wavelet packet transform with $h(n)$ and $g(n)$ being a pair of QMF. (b) Wavelet packet table. Each node is indexed by the corresponding wavelet packet coefficients.

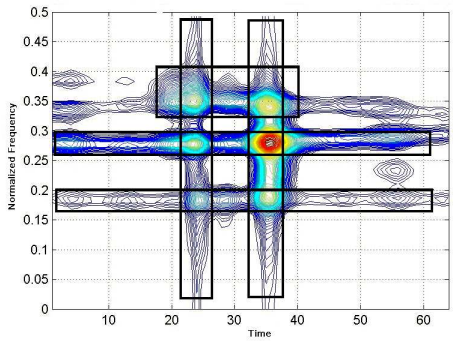


Fig. 7. An illustration of using the wavelet windows to extract localized information contained in a TF image. The size of a window should comply with the uncertainty principle.

with the uncertainty principle. Note that some windows are overlapped, which is impossible in the best basis method, but they render the information description more efficiently. The above example is only an illustration. The selection of the windows or bases should be determined by the signal classification problem itself.

For the convenience of the following discussions, we denote the training data set to be $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N \in \mathbb{R}^h \times \{\pm 1\}$ with a label $+1$ being a mine and a label -1 being a clutter sample. Whenever without confusion, we use a vector $\mathbf{x}$ to denote a mine sample as well as the corresponding WPT pattern.

B. Feature Selection

The full feature set constructed from the wavelet table usually has a high dimensionality relative to the training sample size. Since most of the signal classification systems learn the system parameters in a low dimensional space, reducing the data dimensionality while selecting the most salient features becomes critically important. Basically, there are two methods, feature selection and feature extraction [16], that can be employed to address the above issue. In this paper, we only focus on the methods of feature selection for reducing the data dimensionality.

The problem of feature selection is defined as follows: given a feature set $\mathcal{X}$ of size h, let $\mathcal{S} = \{\mathcal{P} : \mathcal{P} \subset \mathcal{X}, |\mathcal{P}| = l\}$ with $l < h$ (if possible, $l \ll h$) and denote by $\mathcal{D}(\mathcal{P}) = \{(\mathbf{x}_n^{\mathcal{P}}, y_n)\}_{n=1}^N \in \mathbb{R}^l \times \{\pm 1\}$ a training data set constructed from the feature subset $\mathcal{P}$, a feature selection algorithm finds a subset $\mathcal{P}^* \in \mathcal{S}$ such that a cost function $J(\mathcal{D}(\mathcal{P}^*))$ is optimized, i.e.,

$$\mathcal{P}^* = \arg\max_{\mathcal{P} \in \mathcal{S}} J(\mathcal{D}(\mathcal{P})) \tag{6}$$

Without loss of generality, we assume that the larger the cost function, the better the subset. For notational simplicity, in the following discussions we use $J(\mathcal{P})$ instead of $J(\mathcal{D}(\mathcal{P}))$. Suppose a suitable cost function has been chosen to evaluate the goodness of the candidate feature subset, the feature selection problem is reduced to a searching problem. Although an exhaustive search guarantees us to reach the optimal subset, it requires examining $\binom{h}{l}$ possible candidate subsets and consequently it is computationally prohibitive even for moderate values of h and l. The only optimal search strategy which avoids an exhaustive search is the Branch and Bound (BB) method [17]. However, it requires the cost function to be monotonic. Though the monotonicity condition is not particularly restrictive, the BB method does not work well for a large scale problem (when $d > 30$ as defined in [18]). Therefore, in our case we must resort to some computationally feasible strategies that avoid an exhaustive search but might lead to a suboptimal solution. One example is the Sequential Forward Selection (SFS) method. It starts from an empty set and sequentially adds one feature at a time, which, when combined with the already selected features, maximizes the cost function until a predefined feature number is attained. The main drawback of SFS is that it is unable to remove a feature once it is retained and becomes obsolete after the inclusion of other features. This is the so-called *nesting effect*. A more sophisticated search strategy is the Sequential Floating Forward Selection (SFFS) [19], which attempts to overcome the nesting problem of SFS through a flexible backtracking. The algorithm is summarized as follows:

Sequential Floating Forward Selection

Initialization: Full feature set $\mathcal{X}$, $\mathcal{Y}_0 = \{\varnothing\}$, predefined feature number l, $k = 0$

while $k \le l$

$x^+ = \arg\max_{x \in \mathcal{X} - \mathcal{Y}_k} J(\mathcal{Y}_k + \{x\})$

$\mathcal{Y}_{k+1} = \mathcal{Y}_k + \{x^+\}$; $k = k + 1$

if $k > 2$

$x^- = \arg\max_{x \in \mathcal{Y}_k} J(\mathcal{Y}_k - \{x\})$

while $J(\mathcal{Y}_k - \{x^-\}) > J(\mathcal{Y}_{k-1})$ and $k > 2$

$\mathcal{Y}_{k-1} = \mathcal{Y}_k - \{x^-\}$; $k = k - 1$

if $k > 2, x^- = \arg\max_{x \in \mathcal{Y}_k} J(\mathcal{Y}_k - \{x\})$, **end**

end

end

end

Although SFFS does not guarantee the optimality of the selected features, it is reported that, through the flotation, SFFS is able to provide a close to optimal solution [19].

Concerning the cost functions, we consider using a cost function based on the linear discriminant analysis (LDA). Suppose we have a data set $\mathcal{D}(\mathcal{P}) = \{(\mathbf{x}_n^{\mathcal{P}}, y_n)\}_{n=1}^N \in \mathbb{R}^l \times \{\pm 1\}$. Then the *within class* scatter matrix $\mathbf{S}_w$ and the *between class* scatter matrix $\mathbf{S}_b$ are defined as follows [20]:

$$\begin{aligned}\mathbf{S}_w &= \textstyle\sum_{\{n:y_n=+1\}}(\mathbf{x}_n^{\mathcal{P}} - \mathbf{m}_+)(\mathbf{x}_n^{\mathcal{P}} - \mathbf{m}_+)^T + \\ &\quad + \textstyle\sum_{\{n:y_n=-1\}}(\mathbf{x}_n^{\mathcal{P}} - \mathbf{m}_-)(\mathbf{x}_n^{\mathcal{P}} - \mathbf{m}_-)^T \end{aligned} \tag{7}$$

$$\mathbf{S}_b = (\mathbf{m}_+ - \mathbf{m}_-)(\mathbf{m}_+ - \mathbf{m}_-)^T \tag{8}$$

where $\mathbf{m}_+$ and $\mathbf{m}_-$ are the mean vectors of the mine samples and the clutter samples, respectively. To achieve a good class separability, we need to find a subset such that the within class scatter is small while the between class scatter is large. One possible scalar measure using the trace of a matrix is:

$$J = \mathrm{tr}(\mathbf{S}_w^{-1}\mathbf{S}_b) \tag{9}$$

Compared with the wrapper method which uses the classification performance of a certain classifier as the selection criterion, using the LDA cost function has a low computational complexity. However, it only exploits the second-order statistical information and attempts to select the features with unimodal distributions. Hence, even if the exhaustive search method is used, the so-generated feature set may still be suboptimal. Furthermore, the problem of how to choose the feature number is still open. These problems can be mitigated when the boosting algorithm is used to select features adaptively.

C. Neural Network

With the selected features, a neural network (NNW) classifier is designed. We train the network by minimizing the cross-entropy error function given as follows:

$$E = -\sum_{n=1}^{N}\left(\frac{y_n+1}{2}\ln z_n + \frac{1-y_n}{2}\ln(1-z_n)\right) \tag{10}$$

where $y_n \in \{\pm 1\}$ and $z_n \in [0,1]$ are the target value and the output of the network, respectively, corresponding to the input $\mathbf{x}_n$. With the output activation function being chosen to be the logistic function, it can be shown that the output z_n of the network is the estimate of the posterior probability of $\mathbf{x}_n$ belonging to the mine class , i.e., $z_n = \hat{P}(y=+1|\mathbf{x}_n)$ [21].

Although NNW is capable of providing a nonlinear mapping for the training data, it is prone to being stuck by local minima and thus reaching the global minima is not guaranteed. Moreover, the problem of finding the optimum NNW structure, i.e., specifying the numbers of hidden units and hidden layers, is not trivial. It is usually a trial-and-error process. A large amount of data is needed to support this process. Although we have a large number of clutter data, collecting mine samples is expensive. Therefore, in this paper, our strategy is to design a relatively simple classifier, in particular, a NNW classifier with a simple structure, and then use the boosting method to transform the weak learner into a strong learner.

D. Boosting

Boosting is a general method of producing a very accurate prediction rule by combining rough and moderately inaccurate "rules of thumb". It is considered as one of the most important developments in the classification methodology in recent years. The basic idea of the boosting is to linearly combine simple hypotheses, called *base learners* or *weak learners*, to form an ensemble so that the performance of each simple ensemble member is boosted. Given a classifier class $\mathcal{H}$, from which the base learners can be recalled, we are interested in forming an ensemble hypothesis:

$$F(\mathbf{x}) = \sum_{t=1}^{T}\alpha_t h_t(\mathbf{x}) \tag{11}$$

such that a cost function is optimized, where h_t is an element of $\mathcal{H}$ and both α_t and h_t are learned in the boosting procedure. (See, for example, a good tutorial paper [22] and the references therein for more detailed discussions.)

In the past several years, several ensemble methods have been developed. Among them, adaptive boosting (AdaBoost) is the most popular one. The basic theoretical property of AdaBoost is its ability to reduce the training error. The training error decreases exponentially with respect to the number of combined classifiers. Moreover, AdaBoost can also reduce the generalization error and in many cases, the generalization error continues to decrease even after training error becomes zeros.

The original AdaBoost [23] uses the binary-valued classification functions, i.e., $h_t(\mathbf{x}) : \mathbf{x} \to \{\pm 1\}$, as the base learners. Schapire et al. [24] extended it to a more general version, which used the real-valued functions as the base learners, that is, $h_t(\mathbf{x}) : \mathbf{x} \to [-1,1]$ with $\mathrm{sgn}(h_t(\mathbf{x}))$ being the class label and $|h_t(\mathbf{x})|$ the classification confidence. The pseudocode of AdaBoost using soft decisions is presented as follows:

AdaBoost

Initialization: $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N \in \mathbb{R}^l \times \{\pm 1\}$, Maximum iteration number T, $d_1(n) = 1/N, n = 1, \cdots, N$.

for $t = 1 : T$

1. Train weak learner with respect to distribution $\mathbf{d}_t$ and get hypothesis $h_t(\mathbf{x}) : \mathbf{x} \to [-1,1]$.
2. Calculate the weighted margin:

$$r_t = \sum_{n=1}^{N} d_t(n) y_n h_t(\mathbf{x}_n)$$

3. Set $\alpha_t = \frac{1}{2}\ln(\frac{1+r_t}{1-r_t})$
4. Update weights:

$$d_{t+1}(n) = d_t(n)\exp(-\alpha_t y_n h_t(\mathbf{x}_n))/z_t \tag{12}$$

where z_t is the normalization constant such that $\sum_{n=1}^N d_{t+1}(n) = 1$.

end

Output :

$$F(\mathbf{x}) = \sum_{t=1}^{T}\alpha_t h_t(\mathbf{x})$$

AdaBoost minimizes the following cost function:

$$C = \frac{1}{N}\sum_{n=1}^{N}\exp(-y_n F(\mathbf{x}_n)) \tag{13}$$

which is the upper bound of the empirical error $E_{\text{emp}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{I}\{y_n \neq \text{sgn}(F(\mathbf{x}_n))\}$ where $\mathbf{I}\{\cdot\}$ is the indicator function. Mason et al. [25] presented an interesting explanation for AdaBoost by viewing the optimization as a stage-wise gradient descent procedure in the functional space. That is, in the t-th iteration, AdaBoost first chooses $h_t(\mathbf{x})$ as the one to minimize the weighted error and then computes the combination coefficient α_t such that the intermediate cost function:

$$C_t = \frac{1}{N}\sum_{n=1}^{N}\exp\big(-y_n(\sum_{i=1}^{t-1}\alpha_i h_i(\mathbf{x}_n) + \alpha_t h_t(\mathbf{x}_n))\big) \tag{14}$$

is minimized.

In this paper, we use a neural network classifier as the weak learner to estimate the posterior probability of training samples. As mentioned before, one problem associated with the neural network is the possibility of the network being trapped by local minima. One method suggested in the literature is to re-train the network. Here, however, with the using of AdaBoost, as long as the weighted margin $r_t > 0$, the cost function always directs downhill. To see this, we first define the negative functional derivative of C at $F_{t-1}(\mathbf{x})$, if $\mathbf{x} = \mathbf{x}_n$, as [25]:

$$-\nabla C(F_{t-1})(\mathbf{x}) = \frac{1}{N}y_n\exp(-y_n F_{t-1}(\mathbf{x}_n)) \tag{15}$$

To reduce the cost function (Eq. (13)), we need to find a hypotheses h_t such that the inner product $\langle -\nabla C, h_t\rangle$ is positive, i.e.,

$$\begin{aligned}
&\langle -\nabla C, h_t\rangle \\
=\ & \frac{1}{N^2}\sum_{n=1}^{N}\exp(-y_n F_{t-1}(\mathbf{x}_n))y_n h_t(\mathbf{x}_n) \\
=\ & \frac{\sum_{n=1}^{N}\exp(-y_n F_{t-1}(\mathbf{x}_n))}{N^2}\cdot \\
& \cdot\sum_{n=1}^{N}\frac{\exp(-y_n F_{t-1}(\mathbf{x}_n))}{\sum_{i=1}^{N}\exp(-y_i F_{t-1}(\mathbf{x}_i))}y_n h_t(\mathbf{x}_n) > 0
\end{aligned} \tag{16}$$

By unravelling Eq. (12), we get:

$$\begin{aligned}
d_t(n) &= d_{t-1}(n)\exp(-\alpha_{t-1}y_n h_{t-1}(\mathbf{x}_n))/z_{t-1} \\
&= \frac{\exp(-y_n F_{t-1}(\mathbf{x}_n))}{\sum_{i=1}^{N}\exp(-y_i F_{t-1}(\mathbf{x}_i))}
\end{aligned} \tag{17}$$

It immediately follows that:

$$r_t = \sum_{n=1}^{N} d_t(n)y_n h_t(\mathbf{x}_n) > 0 \Rightarrow \langle -\nabla C, h_t\rangle > 0 \tag{18}$$

Moreover, with the classifier class $\mathcal{H}$ being negative closed ($h \in \mathcal{H}$ means that $-h \in \mathcal{H}$), even in the worse case where $r_t < 0$, the cost function can still be reduced by choosing the classifier as $-h_t(\mathbf{x})$ instead of $h_t(\mathbf{x})$. Therefore, due to the iterative nature of AdaBoost, the neural network being trapped by local minima is not a problem but only changes the convergence rate of the cost function.

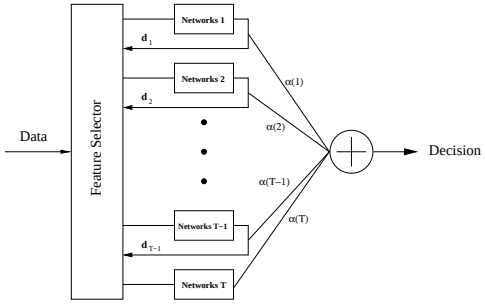


Fig. 8. The training process of the ensemble neural network classifier using AdaBoost.

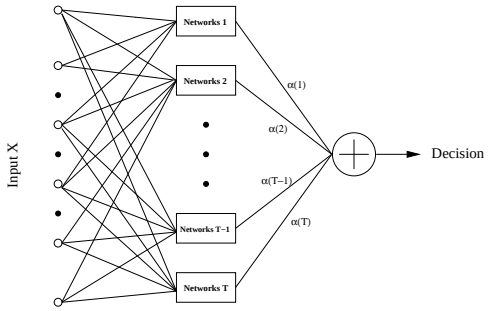


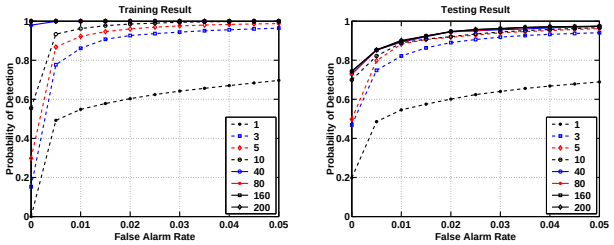
Fig. 9. The resulting ensemble neural network classifier. Due to the feature selection process, most of the connecting weights between the input vectors and each neural network are set to zero.

An intuitive idea of AdaBoost is that the examples which are misclassified get more weights in the following iterations (Eq. (12)); hence, the subsequent classifiers are forced to focus on those hard-to-classify cases, for instance, the samples near the decision boundary. In the original AdaBoost, the algorithm trains an ensemble of classifiers based on the training data with different distributions but with the same features. In our case, we reduce the data dimensionality through feature selection in order to avoid the curse of dimensionality. However, the features so-produced are aimed at optimizing the cost function based on the entire training data set other than being in favor of part of the data and thus may not be able to provide sufficient discriminant information for these harder samples. This observation motivates us to introduce the idea of re-extracting features adaptively based on the misclassified samples before entering the next iteration. This process is depicted in Figure 8. With the iterations, one may expect that the ensemble classifier will overfit the training data eventually. The regularized AdaBoost algorithms [26],[27] can be used to alleviate the problem. However, in our experiment, due to the good overfitting resistance of AdaBoost, the ensemble classifier does not show an apparent overfitting phenomena even after 200 iterations. Shown in Figure 9 is the resulting ensemble neural network classifier. Note that due to the feature selection process, most of the connecting weights between the input vectors and each neural network are set to zero. The resulting classifier can be viewed as an ensemble of experts making the decisions based on the different sets of features.

IV. Experimental Results

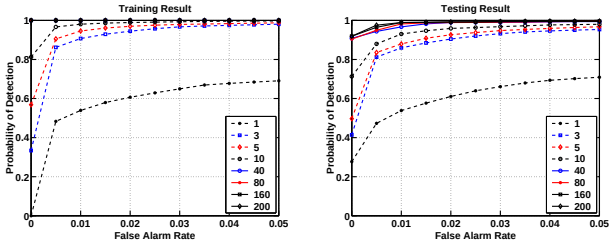
To demonstrate the effectiveness of the proposed classifier, the experimental results based on the measured FLGPR data are presented. We totally collect 133 mine chips and 3962 clutter chips. The mine samples are collected manually and the clutter samples are collected based on the outputs of an energy detector. The training samples are used to train both the appearance based detector and the WPT based detector. In the real system, we use the appearance based Fisherface method as a prescreener to exploit the global information and then further check the down-range profile through the center of each image chip for the final decision. We partition the dataset into 10 realizations. For each realization, 113 mines and 3462 clutter are randomly selected as the training data and the rest as the testing data. Note that the training data for the two classes are highly unbalanced. We hence take five down-range profile signals through the center of each mine chip to augment the mine data set, which leads to 565 and 100 samples, respectively, in the mine training and testing datasets. A Daublet 10 wavelet filter is used to decompose the signals into the WPT coefficients and then SFFS with LDA cost function finds 10 features from the WPT table. With the selected features, a multilayer neural network classifier is designed. The structure of the network is quite simple: it only has 10 input units, 5 hidden units and 1 output units. The sigmoid function and the logistic function are used in the hidden and output layers, respectively, as the activation functions. We train the network to minimize the cross-entropy error function. In term of the network outputs, a weighted margin is calculated and the training data distribution is updated. The above procedure is iterated until the maximum iteration number is reached. The final decision is calculated as the weighted combination of the decisions of the base learners. The 10 training and testing results are averaged and plotted in Figure 10. As we can expect, the training errors are continuously reduced with the increase of the iteration number and reach zero when about 50 classifiers are included in the ensemble classifier. In general, one may not be interested in making the empirical error zero due to the overfitting concerns. However, as we can see in Figure 10, the ensemble classifier demonstrates a very impressive generalization capability. With the inclusion of more classifiers, the receiver operation characteristic (ROC) curve of the testing results are continuously pushed toward the upperleft corner and is saturated when the number of ensemble members reaches 40. Interestingly, the ensemble classifier does not show an apparent overfitting phenomena even when 200 classifiers are combined. For comparison, an ensemble classifier which uses as the features the top 10 PCA (principal component analysis) components are also trained. All of the experimental settings are the same as described above. The classification performance of the PCA based classifier however is much worse than that of the WPT based classifier with the same number of features. This may be explained by the fact that PCA only exploits the global information while WPT can effectively extract intricate structures of target signals.

In the second experiment, we integrate the feature selection module in each iteration of AdaBoost. That is, during the learning procedure, AdaBoost identifies the hard examples of separating mines from clutter and a new set of features, which provide the specific discriminant information for the misclassified samples, is extracted adaptively and a new classifier is trained. The above procedure is iterated until the maximum iteration number is reached. The training and testing results are presented in Figure 11. Again, the training errors are reduced to zero with the increase of iteration number and the testing results show that the ensemble classifier has a good generalization capability. Compared to the classifier without adaptive feature selection, the classifier with adaptive feature selection can give a much better testing performance (Figure 12). It indicates that the AdaBoost algorithm with the feature selection being integrated can effectively extract the discriminant information and at the same time controls the side effect of overfitting.



(a) (b)

Fig. 10. The training and testing results for the ensemble classifier without the feature selection being integrated into each iteration. The numbers in the legend are the iteration numbers. The first iteration corresponds to the classifier without AdaBoost.



(a) (b)

Fig. 11. The training and testing results for the ensemble classifier with the feature selection being integrated into each iteration. The numbers in the legend are the iteration numbers. The first iteration corresponds to the classifier without AdaBoost.

V. Conclusions

In this paper, we have developed a landmine detector by using the wavelet packet transform and machine learning approaches. Through the time-frequency analysis, we have found that most of the discriminant information between signal classes is time-frequency localized. This observation has motivated us to use the wavelet packet transform to sparsely represent the signals with the discriminant information encoded into several bases. The SFFS with the LDA cost function has been used to extract these components. With the extracted features, a neural network classifier has been designed. In order to further improve the classification performance and deal with

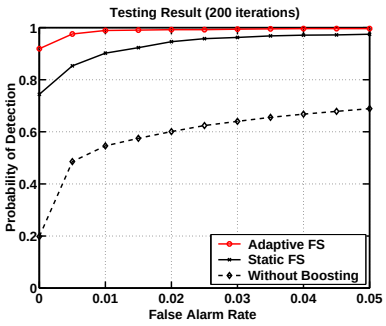


Fig. 12. The ensemble classifier with adaptive feature selection can improve the testing performance significantly over the classifier without adaptive feature selection.

the problem of unlimited possibilities of clutter, the AdaBoost algorithm has been used. We have introduced the idea of integrating the feature selection module into *each* iteration of the AdaBoost algorithm. The experimental results have shown that with the proposed classifier, significant improvement on both the training and testing performances can be achieved. The extensions of our classification scheme to general object recognition problems are possible.

REFERENCES

[1] "Adopt-a-minefield, http://www.landmine.org," 2000.
[2] J. Kositsky and C. Amazeen, "Result from a forward-looking GPR mine detection system," *Proceedings of SPIE on Detection and Remediation Technologies for Mine and Minelike Targets VI*, vol. 4394, pp. 700–711, April 2001.
[3] Y. Sun and J. Li, "Time-frequency analysis for plastic mine detection via forward-looking ground penetrating radar," *IEE Proceedings-Radar,Sonar and Navigation, Special Issue on Time-frequency Analysis for Synthetic Aperture Radar and Feature Extraction*, vol. 150, pp. 253–261, August 2003.
[4] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian detection using wavelet templates," *Conference on Computer Vision and Pattern Recognition, Puerto Rico*, June 17-19 1997.
[5] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," *Proceedings of International Conference on Computer Vision, Bombay, India*, January 1998.
[6] V. Chen and H. Ling, *Time-Frequency Transforms for Radar Imaging and Signal Analysis*. Boston, London: Artech House, 2002.
[7] V. Chen and H. Ling, "Joint time-frequency analysis for radar signal and image processing," *IEEE Signal Processing Magazine*, vol. 16, pp. 81–93, March 1999.
[8] H. Ling, J. Moore, D. Bouche, and V. Saavedra, "Time-frequency analysis of backscattered data from a coated strip with a gap," *IEEE Transactions on Antennas and Propagation*, vol. 41, pp. 1147–1150, August 1993.
[9] A. Martinez and A. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 228–233, February 2001.
[10] D. Swets and J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 831–836, August 1996.
[11] H.-I. Choi and W. Williams, "Improved time-frequency representation of multicomponent signals using exponential kernels," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 862–871, June 1989.
[12] S. Qian, *Introduction to Time-Frequency and Wavelet Transforms*. New Jersey: Prentice Hall PTR, 2002.
[13] R. Coifman and M. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, pp. 713–718, March 1992.
[14] N. Saito and R. Coifman, "Local discriminant bases," in *Wavelet Applications in Signal and Image Processing II, Proc. SPIE 2303* (A. F. Laine and M. A. Unser, eds.), pp. 2–14, 1994.
[15] K. Etemad and R. Chellappa, "Separability-based multiscale basis selection and feature extraction for signal and image classification," *IEEE Transactions on Image Processing*, vol. 7, October 1998.
[16] A. Jain and D. Zongker, "Feature selection: evaluation, application, and small sample performance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, February 1997.
[17] K. Fukunaga, *Statistical Pattern Recognition, 2nd ed.* New York: Academic, 1990.
[18] P. Pudil and J. Novovicova, "Novel methods for subset selection with respect to problem knowledge," *IEEE Intelligent Systems*, vol. 13, Mar/Apr 1998.
[19] P. Pudil, J. Novovicova, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognigion Letters*, pp. 1119–1125, November 1994.
[20] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. New York: J. Wiley, 2000.
[21] C. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford, 1995.
[22] R. Meir and G. Rätsch, "An introduction to boosting and leveraging," *In S. Mendelson and A. Smola, editors, Advanced Lectures on Machine Learning, LNCS, Springer*, pp. 119–184, 2003.
[23] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *European Conference on Computational Learning Theory*, pp. 23–37, 1995.
[24] R. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, pp. 297–336, December 1999.
[25] L. Mason, J. Bartlett, P. Baxter, and M. Frean, "Functional gradient techniques for combining hypotheses," *In B. Scholkopf, A. Smola, P. Bartlett and D. Schuurmans, editors, Advances in Large Margin Classifiers*, 2000.
[26] G. Rätsch, T. Onoda, and K.-R. Müller, "Soft margins for AdaBoost," *Machine Learning*, vol. 42, pp. 287–320, March 2001.
[27] A. Demiriz, K. P. Bennett, and J. Shawe-Taylor, "Linear programming boosting via column generation," *Machine Learning*, vol. 46, no. 1-3, pp. 225–254, 2002.