LBP Features for Hand-Held Ground Penetrating Radar

Samuel Harris[a], Brenden Alvey[a], Dominic K. C. Ho[a], Alina Zare[b]
[a]Department of Electrical Engineering & Computer Science, University of Missouri, Columbia MO 65211, USA
[b]Department of Electrical & Computer Engineering, University of Florida, Gainesville, FL 32601, USA

ABSTRACT

Ground penetrating radar (GPR) has the ability to detect buried targets with little or no metal content. Achieving superior detection performance with a hand-held GPR can be very challenging due to the quality of the data, inconsistency of target signatures, variety of target types, and effects of a human operator. In this paper, we investigate the use of a local binary patterns (LBP) feature vector for target versus non-target discrimination from hand-held GPR data. First, a prescreener algorithm is applied to the GPR data. Then, a GPR B-scan is gathered at each prescreener alarm location and separated into several spatial and depth regions. LBP processing is applied to each spatial and depth cell individually and the LBP features from each cell are group together to form a feature vector. The resulting LBP features are invariant to amplitude scaling and represent the texture in the data. Using this feature vector, a classifier is trained to perform target versus non-target discrimination at each prescreener declaration. Experimental results illustrate the ability of the LBP features to improve detection of buried targets, especially low-metal and non-metal anti-tank and anti-personnel targets.

Keywords: Hand-held ground penetrating radar, local binary patterns, detection, classification

1. INTRODUCTION

Subsurface explosives, such as landmines, endanger military and civilian lives every day. These weapons are usually planted during wartime, but they can remain active for many years afterward. The hazard posed by these explosives has prompted a significant amount of research toward developing automatic subsurface object detection systems. These systems typically utilize one or more physical sensors to obtain a digital representation of an underground area. Traditional approaches have implemented electromagnetic induction (EMI) sensors, more commonly known as metal detectors. However, in recent years, ground penetrating radar (GPR) sensors have also become increasingly popular. This is largely because GPR is capable of detecting objects which contain little or no metal content.

Subsurface object detection systems can be implemented on a variety of platforms, such as vehicle mounted and hand-held systems. Vehicle mounted systems can scan large plots of land relatively quickly and typically focus on detecting anti-tank (AT) targets. However, since the sensors are attached to a rolling vehicle, these systems can only be used on terrain that can be traversed by the vehicle. Hand-held systems are more easily used on land that is difficult to access or traverse. However, hand-held systems require a human operator to sweep the detector back and forth over the ground, resulting in very slow rate of advance. Furthermore, it is often not possible for human operators to sweep the detector with a high level of precision. Even small variations in the speed or height of a sweep can significantly impact the quality of the data being collected, which can make it more difficult to achieve a high detection rate.

One common strategy is to first apply a prescreener algorithm to the hand-held GPR data. The goal of this algorithm is to detect potential anomalies in the incoming data and report any alarm locations that might contain a target. The reported alarm locations can then be analyzed by subsequent algorithms to lower the false alarm rate (FAR)[1-3]. If there is enough data available to train a classifier, reducing the FAR can be approached as a supervised learning task. Once a feature set is computed and a classifier is trained, new GPR data can be fed into the classifier to obtain a target or non-target prediction. The generalizability of such a model is usually gauged by testing its performance on data that was not used during training.

In this paper, we propose the use of a local binary patterns (LBP) feature set along with support vector machine (SVM) to improve the performance of an existing anomaly detection algorithm. Hand-held GPR B-scans are first split into several spatial and depth regions and LBP features are computed within each of the resulting cells. The feature set is reduced and used to train a linear kernel SVM to perform target versus non-target classification. Probability estimates

are obtained from the model and fused with the prescreener confidence values to achieve a lower FAR. The proposed algorithm is evaluated using two hand-held GPR datasets.

This paper is organized as follows. Section 2 describes the hand-held GPR datasets and prescreener alarm generation. Section 3 describes the LBP algorithm in general. Section 4 describes how the LBP feature vectors are extracted from each GPR B-scan, reduced, and used to train the SVM. Section 5 describes the experimental results observed on both datasets. Section 6 concludes the paper.

2. DATA DESCRIPTION

Two different GPRs are used in this study. The first is an impulse radar, which emits a wideband pulse that propagates into the soil. The second is a step-frequency radar that has a narrower bandwidth. Both GPRs contain three spatially separated receivers. A hand-held GPR is carried by a robotic arm, which sweeps the GPR back and forth while moving forward. This provides a sequence of GPR data, which is then regularized with a spatial sampling interval of 1 cm for processing.

The experimental hand-held demonstrator GPR data is first processed by an anomaly detection algorithm[4] in a causal manner. This algorithm acts as a prescreener to identify the locations that may contain targets of interest. It is intended to maintain 100% probability of detection while keeping the number of false alarms as less as possible. The prescreener used in this study relies on the Mahalanobis distance measure[5] to generate a real-valued confidence at each spatially sampled location at 1 cm apart. Locations that have confidence values above a certain threshold are labeled "alarm locations" for further analysis. The data for applying the proposed algorithm consist of the GPR B-scans at each of these alarm locations and the corresponding confidence values from the prescreener.

3. LOCAL BINARY PATTERNS

3.1 Basic LBP Algorithm

Although the two are not identical, GPR B-scans are similar to traditional image data in many ways. Thus, classifying GPR B-scans into target and non-target classes can be approached as an image classification problem. In image processing tasks, it is often helpful to extract features which capture local texture information about an image. One such feature extraction method is called the local binary patterns[6]. LBP first performs a nonlinear transformation on an image, or any similar 2-dimensional data sample, as follows. First, a 3x3 neighborhood of pixels is extracted from the data. Then, each of the 8 pixels located on the perimeter of the 3x3 neighborhood are individually compared to the pixel in the center of the neighborhood. Each perimeter pixel containing a value greater than or equal to that of the center pixel is assigned a binary value of 1. The perimeter pixels that contain a value less than that of the center pixel are assigned a binary value of 0. Each of the 8 perimeter pixels is numbered from 1 to 8. Then, each perimeter pixel is assigned a weight of $2^{(i-1)}$, where i is the pixel number. The binary values that were assigned to the perimeter pixels are then multiplied by the corresponding weights. Finally, the weighted values are summed to obtain a single, unsigned integer. This process is illustrated in Figure 1. The final integer value uniquely describes the local texture of the 3x3 neighborhood.

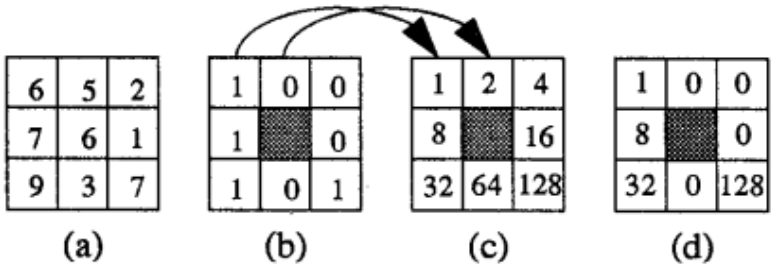

Figure 1: LBP example[6]. This figure illustrates how to perform LBP on a single, 3x3 neighborhood. (a) Original 3x3 neighborhood that has been extracted from some 2-dimensional data sample. (b) Binary values that are assigned to each perimeter pixel. (c) Weights of the perimeter pixels. (d) Result of applying the weights to the binary values. In this example, the perimeter pixels in (d) would be summed to obtain a final LBP value of 169 for this center pixel .

This process is then repeated for each 3x3 neighborhood in the image, resulting in a new LBP matrix of unsigned integer values. Each element of this matrix represents the local texture observed within the corresponding neighborhood in the original image. Since the pixels on the edges of the original data matrix cannot be center pixels of any 3x3 neighborhood, the LBP matrix is of size $(r-2)\times(c-2)$, where r and c are the number of rows and columns in the original data matrix, respectively. Once the LBP matrix has been computed, features are extracted by summing the number of occurrences of each integer in the LBP matrix. Since there are 8 perimeter pixels in each neighborhood, there are 256 possible LBP patterns. Thus, the resulting feature vector consists of 256 features.

3.2 Uniform Patterns

Although 256 different LBP values are possible, some patterns are more intuitively meaningful than others. For example, the patterns shown in Figure 2 all correspond to typical edge behaviors, such as horizontal walls, vertical walls, or diagonal edges. However, the patterns shown in Figure 3 do not clearly correspond to any intuitively meaningful structures. A distinction is drawn between these two pattern types, known as uniform and non-uniform patterns[7].

The distinction between uniform and non-uniform patterns is drawn more rigorously by observing the number of 0 to 1 and 1 to 0 transitions encountered while circularly traversing the 8 perimeter pixels in the local neighborhood. Uniform patterns have two or fewer transitions, while non-uniform patterns have 3 or more transitions. With a 3x3 neighborhood, there are only 58 possible uniform patterns. These uniform patterns often represent the majority of the meaningful texture information in an image, while the remaining non-uniform patterns frequently represent noise in the data[7].

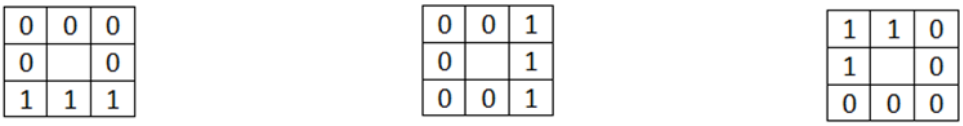

Figure 2: Uniform pattern examples. When circularly traversed, each of these patterns has only two 0 to 1 or 1 to 0 transitions. Thus, each of these patterns is uniform. Notice that each pattern can be intuitively understood as a horizontal, vertical, or diagonal edge texture within the data sample.

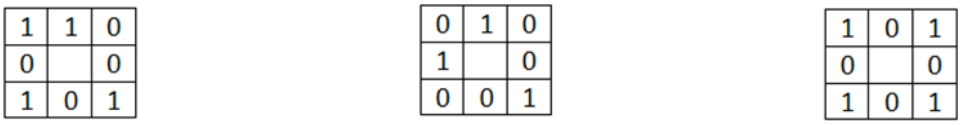

Figure 3: Non-uniform pattern examples. When circularly traversed, each of these patterns has greater than two 0 to 1 or 1 to 0 transitions. Thus, each of these patterns is non-uniform.

4. LBP FEATURE EXTRACTION AND CLASSIFICATION

4.1 Extracting LBP Features from GPR Data

At each prescreener alarm location, a GPR B-scan consisting of 60 successive A-scans was split into 3 overlapping spatial regions and 4 non-overlapping depth regions. The leftmost spatial region consisted of A-scans 1 through 30, the center spatial region consisted of A-scans 16 through 45, and the rightmost spatial region consisted of A-scans 31 through 60. The intention of these spatial regions was to capture the rising edge patterns of each target pattern in the leftmost spatial region and the falling edge patterns in the rightmost spatial region. The center region was included to capture the horizontal textures for wide target patterns and to potentially capture the entire pattern for smaller targets. The 4 depth regions were defined in order to limit the effect of background noise at one depth on the features computed at a different depth.

Due to the oscillation behavior of the signal received by the GPR, the GPR data incrementally shifts between positive and negative values as the depth increases. In order to reduce the effect of this oscillating between positive and negative on the LBP features, the absolute value was taken at each element in the GPR B-scan. Using the absolute value of each element, the LBP matrix for each of the 12 cells was then computed. Features were extracted by counting the number of

occurrences of each uniform pattern in each cell, resulting in 58 features per cell. The feature vectors for each of the 12 cells were concatenated, resulting in a vector of 696 LBP features.

4.2 Reducing the Feature Set

Once a feature vector was extracted at each alarm location, the set of alarm locations was then partitioned into 10 stratified subsets and 10-fold cross validation was carried out. At each iteration of the cross-validation process, the mean and standard deviation across each feature in the set of training targets were computed. This mean and standard deviation were used to perform z-score normalization on both the training and test set prior to training and testing. Then, a linear kernel SVM was trained on the training set and tested on the test set. This way, a test prediction was obtained for each alarm location in the dataset. Once the cross validation was complete, the resulting SVM weight vectors were averaged and the 696 LBP features were ranked from the highest to lowest weight[8].

Ranking the features in this way provided insight into which features were the most useful for performing target versus non-target discrimination. With the feature ranking in place, feature selection was performed by choosing the subset of the *k* highest-ranked features that provided the best detection performance. A script was written to perform stratified 10-fold cross validation using every possible subset of the *k* highest ranked features, with *k* incremented from 2 up to 696. Probability estimates were derived for each prescreen alarm location[9]. By using these probability estimates as confidence values and varying the classification threshold from 0 to 1, a receiver operator characteristic (ROC) curve was computed for each feature subset size. The feature subset that gave the largest cross-validated area under the curve (AUC) was selected. Figure 4 illustrates the effect of this feature selection on a particular AT target in one of the datasets. Figure 5 illustrates the same feature subset on an anti-personnel (AP) target in the same dataset.

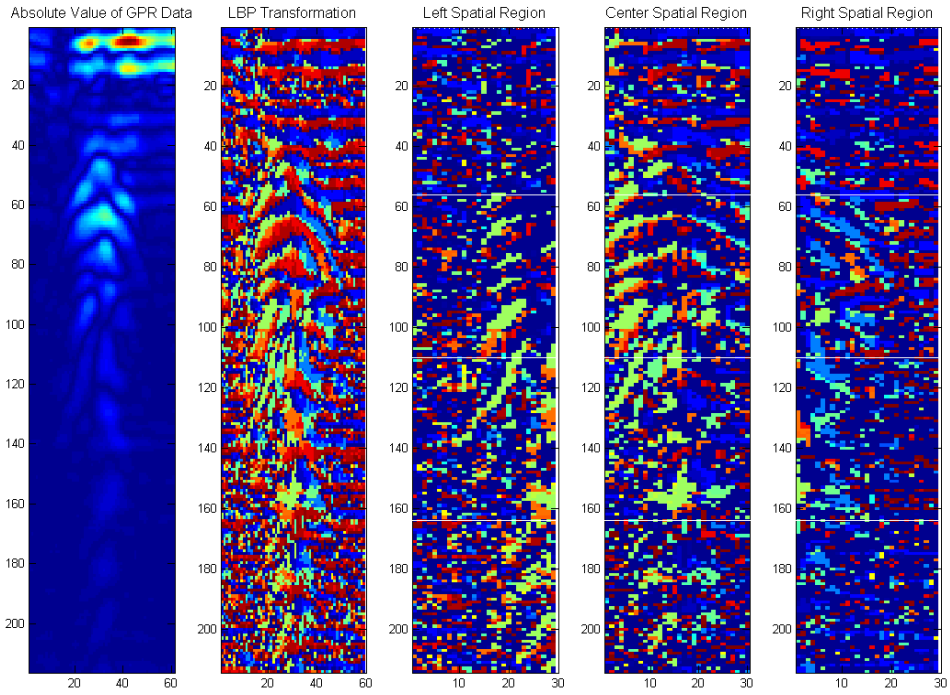

Figure 4: Selected LBP textures (AT target example). The leftmost image shows the absolute value of the GPR B-scan at a particular alarm location. The second image from the left is the LBP matrix computed for the same alarm location, where each color represents a different local texture. The center image is the LBP matrix for the leftmost spatial region, with all eliminated textures set to 0 (dark blue). The second image to the right is the LBP matrix for the center spatial region, with all eliminated textures set to 0. The rightmost image is the LBP matrix for the rightmost spatial region, with all eliminated textures set to 0. Each feature is computed by summing the number of occurrences of a particular texture in the LBP matrix at a certain depth within a particular spatial region. The white lines in the rightmost 3 images indicate the separation among the four depth regions within each spatial region. The rising edge, middle area, and falling edge of the target pattern are clearly visible in the feature-reduced LBP transformation. This target pattern is quite large and spans across multiple depth

regions. Therefore, this pattern will influence the LBP features computed across each of these depth regions. This target received a very high confidence from both the prescreener and the SVM.

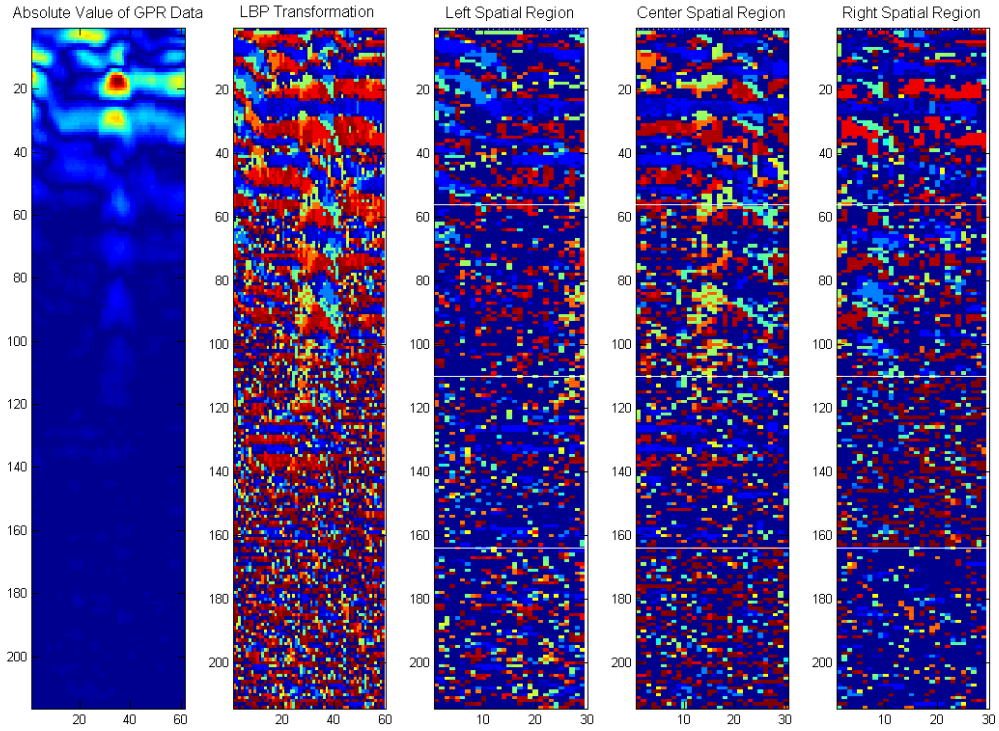

Figure 5: Selected LBP textures (AP target example). The leftmost image shows the absolute value of the GPR B-scan at a particular alarm location. The second image from the left is the LBP matrix computed for the same alarm location, where each color represents a different local texture. The center image is the LBP matrix for the leftmost spatial region, with all eliminated textures set to 0 (dark blue). The second image to the right is the LBP matrix for the center spatial region, with all eliminated textures set to 0. The rightmost image is the LBP matrix for the rightmost spatial region, with all eliminated textures set to 0. Each feature is computed by summing the number of occurrences of a particular texture in the LBP matrix at a certain depth within a particular spatial region. The white lines in the rightmost 3 images indicate the separation among the four depth regions within each spatial region. This target pattern is not as large or as clear as the AT target pattern in Figure 4. However, the same yellow, orange, green, and light blue textures that were observed in the AT target example are observed here as well. This target received a very low confidence from the prescreener. However, it received a probability estimate greater than 0.9 from the SVM.

4.3 Final Classification & Confidence

After the optimal subset of highest-ranked features was selected, a new linear SVM was trained on this reduced feature set. Once again, stratified 10-fold cross validation was performed. Probability estimates were derived for each alarm location in each test set and these probability estimates were used as confidence values. The confidence values produced from the trained SVM were then fused with the prescreener confidence values using the geometric mean. The result of this fusion, defined in Equation 1 below, was then used as the final confidence value for each alarm location.

$$fused\_conf = prescreen\_conf^{0.5} \times SVM\_conf^{0.5} \quad (1)$$

5. EXPERIMENTAL RESULTS

5.1 Impulse Radar Results

For the impulse radar dataset, observing the SVM weight vector allowed 334 features to be eliminated. The final 10-fold cross validation was performed using the remaining 362 features. The performance of the prescreener alone, the SVM

alone, and the fusion of the two by Equation 1 was compared by plotting the corresponding ROC curves. Figure 6 shows these results when all targets are included in scoring. Figure 7 illustrates the results when only AT targets are included in scoring. Figure 8 gives the results when scoring only AP targets. The SVM alone provided very good AP target detection. In all cases, the fused confidence values provided better detection performance than either the prescreener or SVM alone.

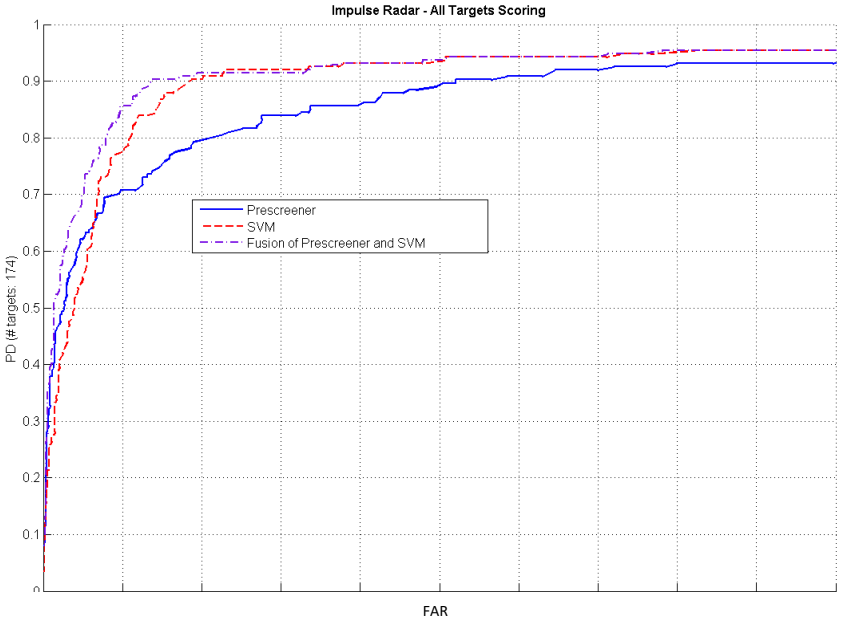

Figure 6: Impulse radar detection results of all targets. The fusion of the prescreener confidence values and SVM probability estimates provides the best overall detection performance.

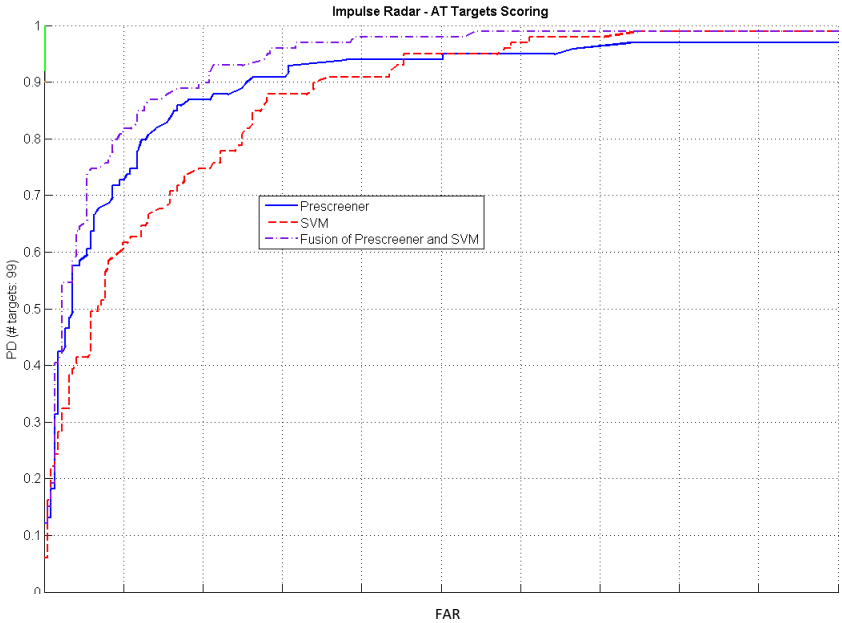

Figure 7: Impulse radar detection performance of AT targets. The SVM alone provides poorer AT target detection than the prescreener. However, the fusion of the SVM probability estimates with the prescreener confidence values provides the best AT target detection.

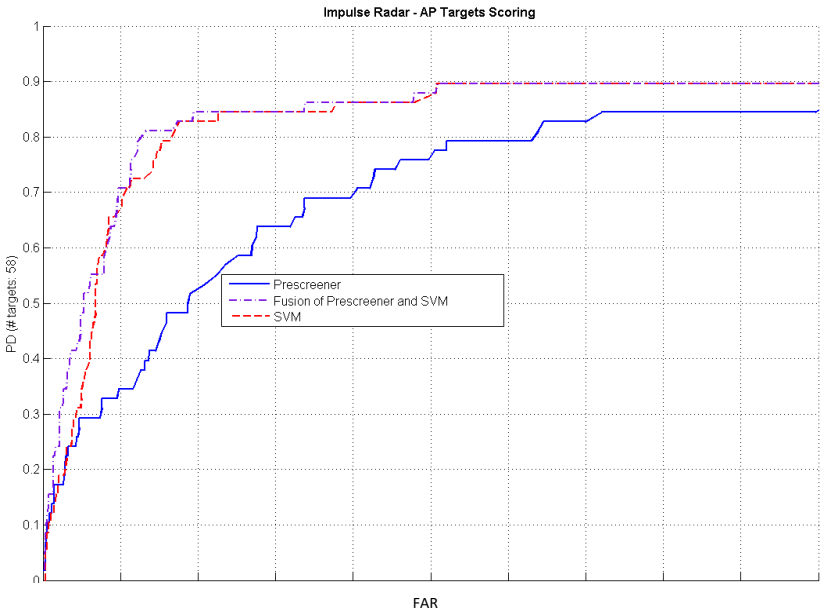

Figure 8: Impulse radar detection performance of AP targets. The SVM alone provides very good AP target detection. The fusion of the prescreener confidence values and SVM probability estimates yields slightly better detection than the SVM alone.

5.2 Step-Frequency Radar Results

For the step-frequency radar dataset, observing the SVM weight vector allowed 472 features to be removed. The final 10-fold cross validation was performed using the remaining 224 features. The performance of the prescreener alone, the SVM alone, and the fusion of the two by Equation 1 was compared by plotting the corresponding ROC curves. Figure 9 shows the results when all targets are included in scoring. Figure 10 are the results when only AT targets are included in scoring. Figure 11 gives the results when only AP targets are scored. As was observed in the impulse radar dataset, the SVM provided very good AP target detection.

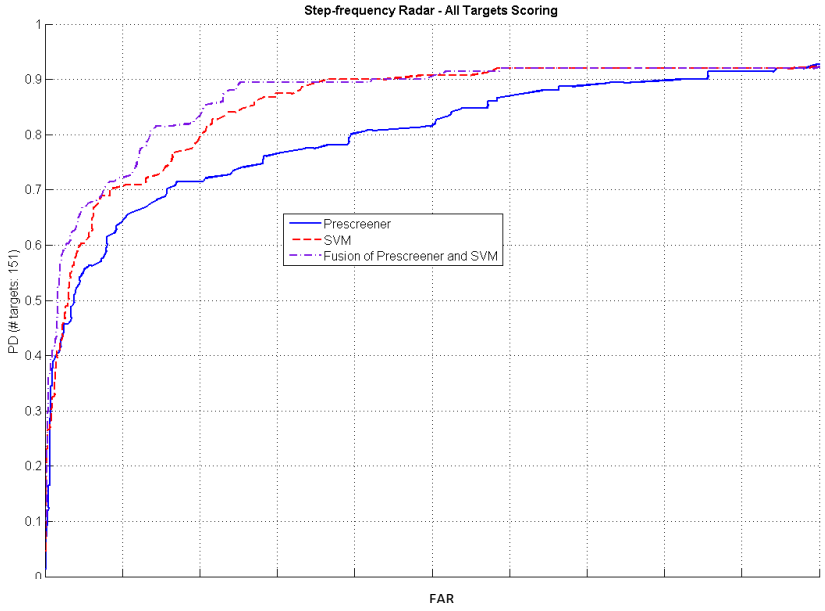

Figure 9: Step-frequency radar detection results of all targets. The fusion of the prescreener confidence values and SVM probability estimates provides the best overall detection performance.

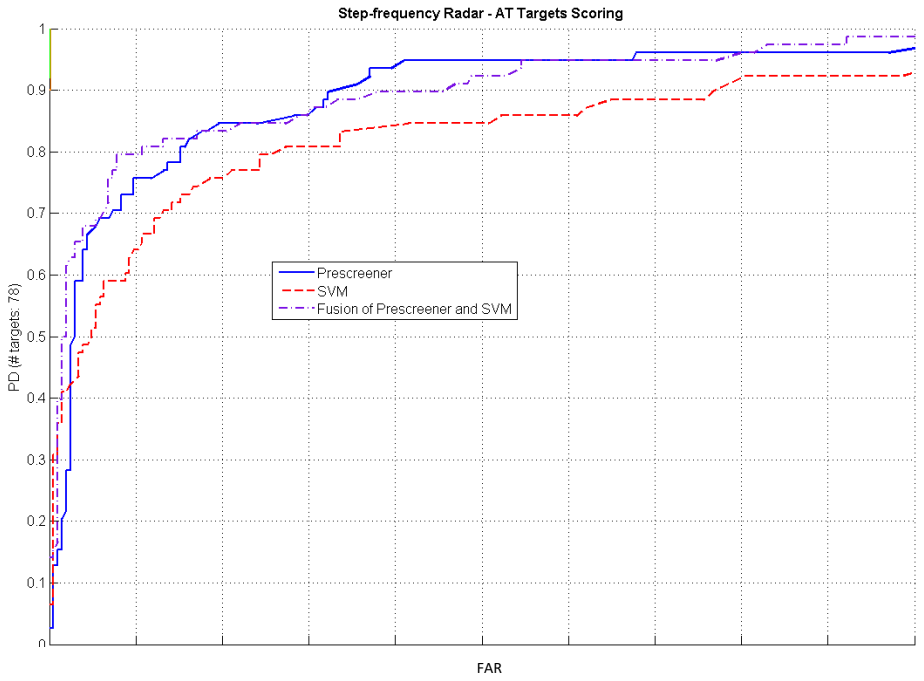

Figure 10: Step-frequency radar detection performance of AT targets. The SVM alone provides significantly poorer AT target detection than the prescreener. However, the fused confidence values perform comparably to the prescreener on AT targets.

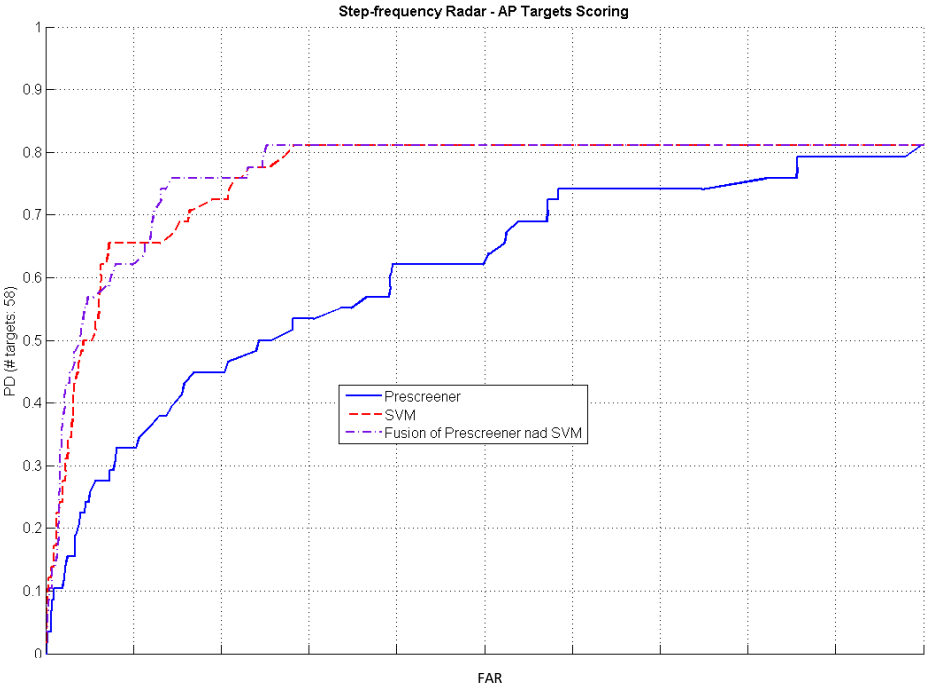

Figure 11: Step-frequency radar detection performance of AP targets. The SVM provides very good AP target detection. The fusion of the prescreener confidence values and SVM probability estimates performs comparably to the SVM alone.

6. CONCLUSION

We have proposed the use of LBP features to improve the detection performance of a hand-held GPR system. GPR B-scans were extracted from prescreener alarm locations. Each B-scan was split into 3 spatial regions and 4 depth regions, and LBP features were computed within each of the resulting 12 cells. The feature set was reduced using a linear-kernel SVM, and another SVM was trained to perform target versus non-target classification with the reduced feature set. Using 10-fold cross validation, probability estimates were derived for each alarm location. These values were then coupled with the prescreener confidence values to increase the detection performance. The proposed features was tested on two datasets collected from two different GPRs. For both datasets, the use of LBP features yielded improved performance over the prescreener alone. Although the composite result improved in both cases, the most drastic improvement in detection performance was observed among the AP targets.

REFERENCES

[1] Ho, K. C., Gader, P.D., and Devaney, J. B., "Locate mode processing for hand held land mine detector," Proc. SPIE Conf. Detection and Remediation Technologies for Mines and Minelike Targets VII, (2002).

[2] Wilson, J. N. Gader, P. D., Ho, K. C., Lee, W. H., Stanley, J. R., and Glenn, T. C., "Region processing of ground penetrating radar and electromagnetic induction for handheld landmine detection," Proc. SPIE Conf. Detection and Remediation Technologies for Mines and Minelike Targets IX, (2004).

[3] Ngan, P., Burke, S., Cresci, R., Wilson, J. N., Ho, K. C., and Gader, P. D., "Region processing algorithm for HSTAMIDS," Proc. SPIE Conf. Detection and Remediation Technologies for Mines and Minelike Targets XI, (2006).

[4] Ho, K. C., Harris, S., Zare, A., and Cook, M., "Anomaly detection of subsurface objects using handheld ground-penetrating radar," Proc. SPIE Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XX, 94541B (2015).

[5] Mahalanobis, P. C., "On the generalized distance in statistics," Proceedings of the National Institute of Science, 49-55 (1936).

[6] Ojala, T., Pietikainen, M., and Harwood, D., "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions," Proceedings of 12th International Conference on Pattern Recognition, Jerusalem, 582-585 (1994).

[7] Ojala, T., Pietikainen, M., and Maenpaa, T., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Transaction on Pattern Analysis and Machine Intelligence 24(7), 971-987 (2002).

[8] Brank, J., Grobelnik, M., Milic-Frayling, N., and Mladenic, D., "Feature selection using linear support vector machines," Proceedings of the third international conference on data mining methods and databases for engineering, finance and other fields, (2002).

[9] Chang, C. C., and Lin, C. J., "LIBSVM: a library for support vector machines," ACM Transactions on Intelligent Systems and Technology (TIST) (2011).