Automatic Target Detection using Ground Penetrating Radar (GPR) Data Based on a Machine Learning Approach

Gaurav R. Savade

Department of Electronics and Telecommunication, Babasaheb Naik College of Engineering Pusad, Karla Road, Pusad, Maharashtra, 445215, India.

---***---

Abstract - Automatic target classification in radar-based systems is critical for various applications, including defense, surveillance, and environmental monitoring. This research presents a robust approach for classifying targets in ground-penetrating radar (GPR) data by combining texture-based Haralick features and shape-oriented Histogram of Oriented Gradients (HOG) features. The integration of these complementary feature sets enhances target characterization by capturing both spatial textures and edge patterns in GPR images. Using the extracted features, we evaluate two prominent machine learning classifiers, Support Vector Machine (SVM) and Random Forest (RF), for their classification accuracy, precision, and computational efficiency. Experimental results on a benchmark GPR dataset show that the fusion of Haralick and HOG features significantly improves classification performance compared to individual feature sets. The SVM classifier achieves superior accuracy with 95.83% in predicting the target/non-target object as mine/non-mine. Further, mine target is detected with image annotation using image morphology. Our method highlights the potential of hybrid feature extraction and machine learning classifiers in achieving accurate and reliable automatic target classification in GPR data, paving the way for real-time applications in complex operational environments.

***Key Words*:** Machine Learning, Ground Penetrating Radar, Object Detection, Signal Processing.

1. INTRODUCTION

Ground Penetrating Radar (GPR) is a non-destructive electromagnetic (EM) subsurface probing technique used in fields as diverse as geophysics, archaeology, environmental science, and infrastructure inspection. By emitting high-frequency EM waves and capturing the reflected signals, GPR systems generate data that can reveal details about underground structures, anomalies, or targets. This capability has proven valuable in tasks ranging from detecting buried utilities and unexploded ordnance to mapping subsurface features in historical sites. However, traditional GPR data analysis is a labor-intensive process requiring significant expertise and interpretive skills, often yielding subjective results dependent on the analyst's experience. By identifying and learning patterns in complex GPR data, ML techniques can facilitate faster and more accurate detection processes, making GPR applications more reliable and accessible across a wider range of fields [1-2].

In recent years, machine learning (ML) has emerged as a powerful approach for processing GPR data, enabling the automation of target detection with reduced human intervention. Machine learning (ML), which has already transformed fields like computer vision, natural language processing, and speech recognition, holds significant potential for improving GPR target detection. Machine learning approaches allow for the extraction of meaningful patterns from complex GPR data, improving both speed and accuracy. The need for an efficient and objective approach to GPR data interpretation has led to growing interest in automated techniques for target detection [3]. This paper reviews existing literature on machine learning-based approaches for automatic target detection in GPR data, discussing the types of features, algorithms, and the main challenges faced in the field. The main objective of this paper is to develop a hybrid feature extraction and classification approach that integrates Haralick (texture) and HOG (shape) features for enhanced target classification using SVM and RF classifiers.

2. LITERATURE REVIEW

Ground Penetrating Radar (GPR) is widely utilized for subsurface exploration and target detection in fields like archaeology, construction, and defense. Advanced processing techniques have been developed to enhance the detection capabilities of GPR data, addressing issues such as noise reduction, resolution enhancement, and feature extraction. Here's a review of some advanced techniques in GPR data processing for improved target detection:

D. Chen et.al. [4] presented a detection method of DCAM-YOLOv5 for ground penetrating radar (GPR) to address the difficulty of identifying complex and multi-type defects in tunnel linings. Jiangkun Gong et.al. [5] presented a brief examination of Automatic Target Recognition (ATR) technology within ground-based radar systems which offers a lucid comprehension of the ATR concept, delves into its historical milestones, and categorizes ATR methods according to different scattering regions. Feifei Hou et.al. [6] developed an automatic method based on a deep instance segmentation framework to detect and segment object signatures from GPR scans which develops the Mask Scoring R-CNN (MS R-CNN) architecture. Zhongming Xiang et.al. [7] proposed a rebar can distinguish signals from different scales and directions. The results revealed that the proposed method can achieve F1 score of 0.9649 on the collected dataset; and it is a robust method that can discriminate strong noise, separate interlaced rebars, and remove cross rebar signals and direct wave. Donghwi Kim et.al. [8] investigated the effectiveness of data augmentation techniques in the automated analysis of B-scan images from ground-penetrating radar (GPR) using deep learning. A deep learning model for GPR data analysis was developed using three models (Faster R-CNN ResNet, SSD ResNet, and EfficientDet) based on transfer learning. Yu-Chen Zhang et.al. [9] proposed a method for the automatic corrosive environment detection of bridge decks from ground-penetrating radar (GPR) data based on the single-shot multibox detector (SSD) model. Leila Carolina Martoni Amaral et.al. [10] presented the application of the state-of-the-art You Only Look Once (YOLO) v5 algorithm to detect underground objects using GPR images.

Yi-Tao Dou et.al. [11] combined ground penetrating radar (GPR) and convolutional neural networks for the intelligent detection of underground road targets. The target location was realized using a gradient-class activation map (Grad-CAM). Hai Liu et.al. [12] proposed an automatic GPR method for recognition and localization of underground pipelines based on a deep learning model. Kehui Chen et.al. [13] proposed a novel recognition method based on time-frequency texture features and a support vector machine (SVM). The experimental results show that the proposed method can achieve higher recognition accuracies than the other two methods in a heavily noisy environment. Mohd Shuhanaz Zanar Azalan et.al. [14] proposed a framework to classify the size of underground metallic pipe by using Histogram of Oriented Gradient (HOG) as a feature extraction algorithm. Two machine learning algorithms, Support Vector Machines (SVM) and Backpropagation Neural Network, were proposed to classify the size of the underground metallic pipe. Man-Sung Kang et.al. [15] proposed a system based on deep convolutional neural networks, which is capable of autonomous underground cavity detection beneath urban roads using three-dimensional ground penetrating radar data. Daffa Dewantara et.al. [16] proposed a new method for automating hyperbola detection and apex extraction on radargram. The model consists of two modules that take radargram as input in a form of images. Changle Xin et.al. [17] proposed a method that can achieve fast goals in complex multi-goal situations. The simplified ResNet-50 network is used to detect dense intersecting hyperbolic curves and reduce the computation of the network. H. Harkat et.al. [18] proposed an alternative classification methodology in which classify the windows of GPR radargrams into two classes (with or without target) using a neural network radial basis function (RBF), designed via a multi-objective genetic algorithm (MOGA). Zhi Qiu et.al. [19] combined GPR and intelligent technology to research three aspects: acquiring real-time GPR images, using the YOLOv5 algorithm for real-time target detection and the coordinate positioning of GPR images, and the construction of a detection system based on ground-penetrating radar and the YOLOv5 algorithm that automatically detects target characteristic curves in ground-penetrating radar images. Zhimin Gong et.al. [20] proposed a deep-learning based Faster R-CNN algorithm for the automatic classification and recognition of GPR images. The results showed that Faster R-CNN could successfully identify GPR images and accurately classify them, with an average accuracy rate of 93.9%.

Guangyan Cui et.al. [21] proposed the accurate calculation of electromagnetic wave velocity (AC-EWV) by searching for the minimum image entropy of migrated radargrams. The average correct rate for GPR data of the double-layer rebars is 91.67%. Volodymyr Motyka et.al. [22] built a target recognition model in the form of AGM-86C (CALCM) and CR Taurus KEPD 350 cruise missiles. Yunjie Zhong et.al. [23] presented a technique in which the ground-penetrating radar echo images (B-scan) after processing are mean-filtered to eliminate the direct waves that interfere greatly with the echoes. The experimental results show that the algorithm models proposed in this paper have a good recognition effect in ground-penetrating radar echo image target detection. Wei Xue et.al. [24] proposed an efficient underground target detection method for urban roads based on neural networks. The experimental results show that the proposed method can effectively locate the horizontal ranges and vertical depths of underground targets for urban roads and has higher recognition accuracy and less processing time than the traditional segmentation recognition methods. Junyi Zou et.al. [25] proposed a real-time target detection system based on the multi-source fusion method.

3. PROPOSED METHODOLOGY

The entire working process of the proposed method is shown in Figure 1. It consists of a series of processes which are discussed further.

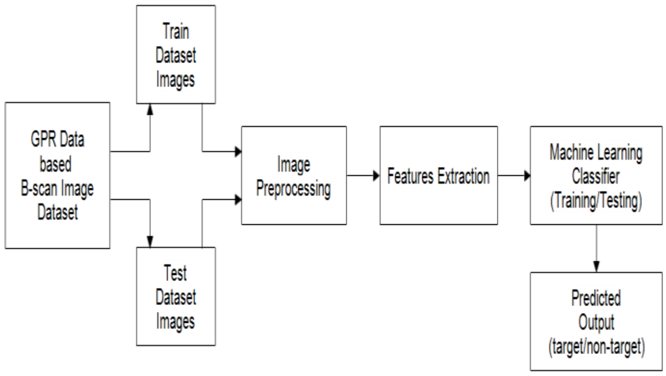

Figure 1: Target Classification Framework Using Machine Learning

Dataset

The dataset used in the proposed approach is from Scott Multi-static GPR experimentation [26][27]. Mine_clean.mat, Mine_rock.mat and Nothing_clean.mat files are used to classify the target data. Converting Ground-Penetrating Radar (GPR) data into B-scan images is a key part of interpreting and analyzing GPR experiments, such as those used in the Scott experimentation. B-scan images represent the reflected signal amplitude across the survey area, enabling the detection and classification of subsurface objects like mine targets. Each row in the GPR data matrix corresponds to a radar scan at a specific position, while each column corresponds to the time or depth axis. To create the B-scan, plot the matrix as an image where the x-axis represents the spatial dimension (distance), the y-axis represents time/depth, and the pixel values represent the amplitude of the radar reflections.

Preprocessing

Pre-processing is a common name for operations with images at the lowest level of abstraction -- both input and output are intensity images. The aim of pre-processing to improve the image data that suppresses unwanted distortions or enhances some image features which are important for further processing. Digital image processing is always an interesting field as it gives improved pictorial information for human interpretation and processing of image data for storage, transmission, and representation for machine perception. Some of the pre-processing techniques used in proposed approach are:

a. RGB to Gray Conversion

In image processing, converting an RGB (color) image to grayscale simplifies the data, reducing the color channels (typically red, green, and blue) into a single channel representing intensity. The grayscale intensity is computed by combining the three color channels in a weighted sum that reflects human perception of brightness.

b. Image Cropping

Image cropping to select a Region of Interest (ROI) is a common task in image processing, allowing to focus on a specific part of an image. MATLAB provides tools to manually and programmatically select and crop an ROI.

c. Image Resizing

Image resizing changes the dimensions of an image, either to scale it up or down, which is essential in tasks like image preprocessing for machine learning, display optimization, and file size reduction. MATLAB provides straightforward tools for resizing images. In proposed approach, 256x256 dimensions are used.

Feature Extraction

In the proposed model, primarily two features are extracted from Bscan Images to identify the characteristics of it. Haralick features and HOG Features.

a. Haralick Features

Haralick texture features are calculated from a Gray Level Co-occurrence Matrix, (GLCM), a matrix that counts the co-occurrence of neighboring gray levels in the image. The GLCM is a square matrix that has the dimension of the number of gray levels N in the region of interest (ROI). The GLCM functions characterize the texture of an image by calculating how often pairs of pixel with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix.

To illustrate, the following figure 2 shows how graycomatrix calculates the first three values in a GLCM. In the output GLCM, element (1,1) contains the value 1 because there is only one instance in the input image where two horizontally adjacent pixels have the values 1 and 1, respectively. glcm(1,2) contains the value 2 because there are two instances where two horizontally adjacent pixels have the values 1 and 2. Element (1,3) in the GLCM has the value 0 because there are no instances of two horizontally adjacent pixels with the values 1 and 3. graycomatrix continues processing the input image, scanning the image for other pixel pairs (i,j) and recording the sums in the corresponding elements of the GLCM. Process Used to create the GLCM

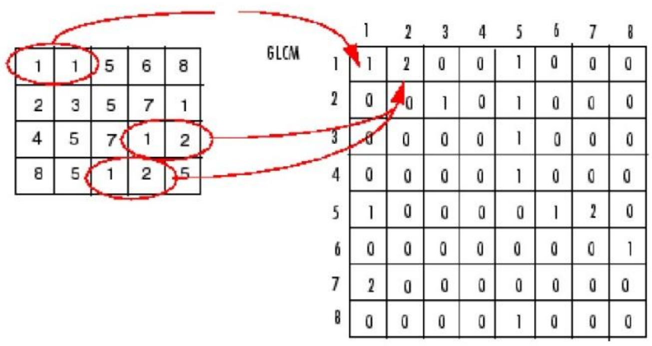

Figure 2: Example of GLCM computation

A GLCM is a matrix of rows and columns that are equal to number of grey levels in the image. Statistical features extraction is one of the primary methods in image processing. Extraction of texture features of images Most recommended statistical methods of extracting texture features from images is GLCM.

Here are some frequently used Haralick features, each computed from the GLCM:

Contrast: Measures the local variations in the GLCM. Higher contrast indicates larger intensity differences between neighboring pixels.

$$\text{Contrast} = \sum_{i,j} |i-j|^2 \, P(i,j)$$

Correlation: Measures the correlation between pixel pairs in the GLCM. A higher value indicates a predictable relationship between neighboring pixel intensities.

$$\text{Correlation} = \sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)P(i,j)}{\sigma_i \sigma_j}$$

Energy: Also called "Angular Second Moment" (ASM), it measures textural uniformity. Higher energy indicates less variation in pixel pairs.

$$\text{Energy} = \sum_{i,j} P(i,j)^2$$

Homogeneity: Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal. Higher homogeneity implies less variation in neighboring pixels.

$$\text{Homogeneity} = \sum_{i,j} \frac{P(i,j)}{1 + |i - j|}$$

Entropy: Measures the randomness in the image texture. Higher entropy indicates a more complex or heterogeneous texture.

$$\text{Entropy} = -\sum_{i=1}^{L} \sum_{j=1}^{L} P(i,j) \log(P(i,j) + \epsilon)$$

Mean Intensity: It gives insight into the overall brightness, which is valuable for separating foreground and background or identifying regions based on their illumination.

b. Histogram of Oriented Gradients (HOG) Feature

Histogram of Oriented Gradients (HOG) is a feature descriptor technique that represents the appearance and shape of an object by capturing the distribution of intensity gradients or edge directions. HOG is commonly used in image processing and computer vision, particularly for object detection tasks.

Key Concepts of HOG

1. *Gradient Computation:* The image is divided into small spatial regions (cells), and within each cell, gradients (changes in pixel intensity) are computed.

2. *Orientation Binning:* Each gradient in the cell is placed into orientation bins, creating a histogram based on the gradient direction. This histogram serves as a descriptor for the local region.

3. *Normalization:* Histograms from each cell are combined, and normalization is performed across larger regions called blocks, which helps in handling variations in illumination and contrast.

4. *HOG Descriptor:* The resulting concatenated histogram values across all cells form the HOG descriptor, a feature vector representing the image.

Steps for Computing HOG Features

Given an image, here's a step-by-step process to compute HOG features.

1. **Preprocessing**: Resize and convert the image to grayscale if needed.

2. **Compute Gradients**: Calculate horizontal and vertical gradients for each pixel.

3. **Orientation Binning**: Divide the image into cells, compute the gradient orientation and magnitude, and create orientation histograms for each cell.

4. **Normalization**: Group cells into blocks, and normalize the histogram across blocks to reduce sensitivity to lighting.

5. **Concatenate Histograms**: Flatten and concatenate all the histograms to create a single HOG descriptor for the image.

Machine Learning Classification

a. Random Forest Classifier

Random Forest is a popular ensemble learning method used for classification (and regression) tasks. It operates by constructing a multitude of decision trees during training and outputs the class that is the mode of the classes predicted by individual trees. It's robust to overfitting, effective for large datasets, and handles high-dimensional spaces well.

Key Concepts of Random Forest Classification

1. *Ensemble Learning:* Random Forest is an ensemble of many decision trees, each trained on a subset of the training data, with random subsets of features.

2. *Bagging (Bootstrap Aggregating):* Each tree is trained on a random subset of data (sampling with replacement) to ensure diversity among trees. This reduces variance and improves generalization.

3. *Random Feature Selection:* For each split in a decision tree, only a random subset of features is considered, further diversifying the trees and reducing overfitting.

4. *Majority Voting:* In classification tasks, each tree in the forest votes on the class, and the class with the most votes is chosen as the final prediction.

Steps in Random Forest Classification

1. **Train Multiple Decision Trees**: For a dataset with NNN samples, bootstrap samples (random subsets with replacement) are generated, and a decision tree is trained on each subset.

2. **Random Feature Selection for Each Split**: Instead of considering all features, only a random subset of features is used for splitting in each tree, promoting diversity among trees.

3. **Voting Mechanism**: Each tree makes a prediction for the test sample. The class with the most votes across trees is chosen as the final prediction.

b. Support Vector Machine Classifier

Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression tasks, although it is primarily applied to classification. SVM aims to find the optimal hyperplane that maximally separates different classes in the feature space.

Key Concepts of Support Vector Machine (SVM)

1. *Hyperplane*: In an SVM, the algorithm seeks to find the hyperplane that best separates the classes. In a two-dimensional space, this hyperplane is simply a line, while in higher dimensions, it becomes a plane or hyperplane.
2. *Margin*: The margin is the distance between the hyperplane and the closest data points from each class. SVM optimizes for the hyperplane that maximizes this margin, providing better generalization to new data.
3. *Support Vectors*: The data points that lie closest to the hyperplane are known as support vectors. These points are critical as they define the position and orientation of the hyperplane. Removing any other point would not change the decision boundary.
4. *Kernel Trick*: SVMs can be extended to classify non-linearly separable data using a kernel function. The kernel function transforms the data into a higher-dimensional space, making it possible to separate the classes linearly in that space.

Steps for SVM Classification

1. **Data Preprocessing**: Standardize the dataset by scaling the features, especially for kernels sensitive to feature scaling (e.g., RBF).
2. **Choose a Kernel**: Select an appropriate kernel based on the linearity or non-linearity of the data.
3. **Train the SVM**: Fit the model on training data to find the optimal hyperplane that maximally separates the classes.
4. **Hyperparameter Tuning**: Adjust hyperparameters such as C (regularization) and gamma (for RBF) to improve the model's performance.

4. RESULTS AND DISCUSSION

A. Experimental Setup

The proposed system is implemented and analyzed on Intel CORE processor i3, 8GB RAM Laptop configuration, and Windows 10 operating system. MATLAB R2018b Software is used to write the programming code in this we used Image processing and Statistics and Machine Learning toolbox. The images used to train and test are used from the benchmark Dataset [26][27] for experimentation analysis.

B. Performance Evaluation Parameters

Performance evaluation metrics are essential for assessing the effectiveness of machine learning models, particularly in classification, regression, and other predictive tasks. Different metrics suit different types of problems, so selecting the right ones helps interpret the model's accuracy, robustness, and generalizability. Below are commonly used metrics categorized by task type.

A confusion matrix contains information about actual and predicted classifications done by a classification system. The performance of such systems is commonly evaluated using the data in the matrix.

Accuracy: The proportion of correct predictions among the total predictions.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Precision: Measures the accuracy of positive predictions. High precision means fewer false positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall (Sensitivity): Measures the ability to find all positive samples. High recall means fewer false negatives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1 Score: The harmonic mean of precision and recall, useful when classes are imbalanced.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where TP, TN, FP, FN are True Positives, True Negatives, False Positives, and False Negatives, respectively.

C. Result Evaluation

In this experiment, we have two stages of implementation: training and testing. During the training phase, the train set of images is preprocessed. For the feature extraction procedure of the training images, we employed texture-based features are evaluated. After extracting the features, the next step is to train the model using two classifiers RF and SVM. The algorithm will be trained the data based on the input and output data, where the input is the dataset of bscan image texture features for training and the output is the corresponding labels. Once the training model was successfully validated, we proceeded to store the trained model. Then testing is performed.

Figure 3 shows the sample Bscan images from the experimentation which shows the mine_clean, mine_rock and nothing_clean targets. Here randomly display 15 images.

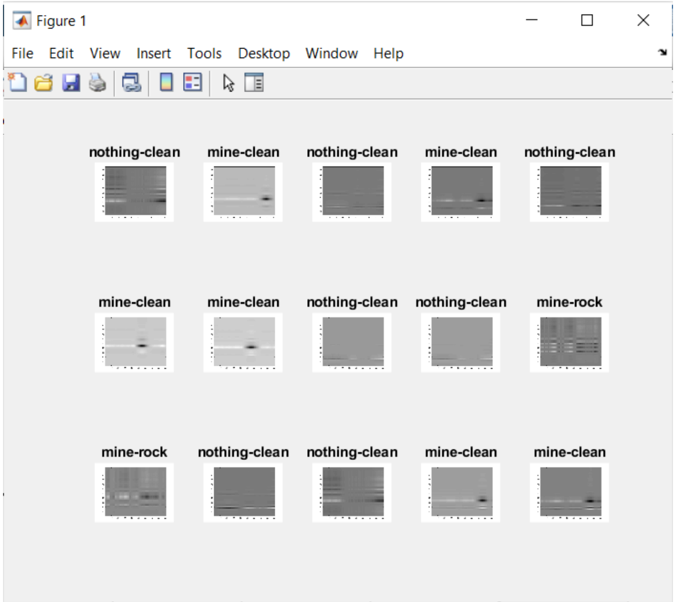

Figure 3: Sample Images from dataset

After learning all bscan image files of each output classes, ML models is trained and tested. After successful validation of training, confusion matrix is generated for testing phase as shown in figure 4 to 5 for RF and SVM. It shows the number of bscan images used for testing and its prediction for each class with correctly classified and misclassified data for three output classes, mine_clean, mine_rock, and nothing_clean.

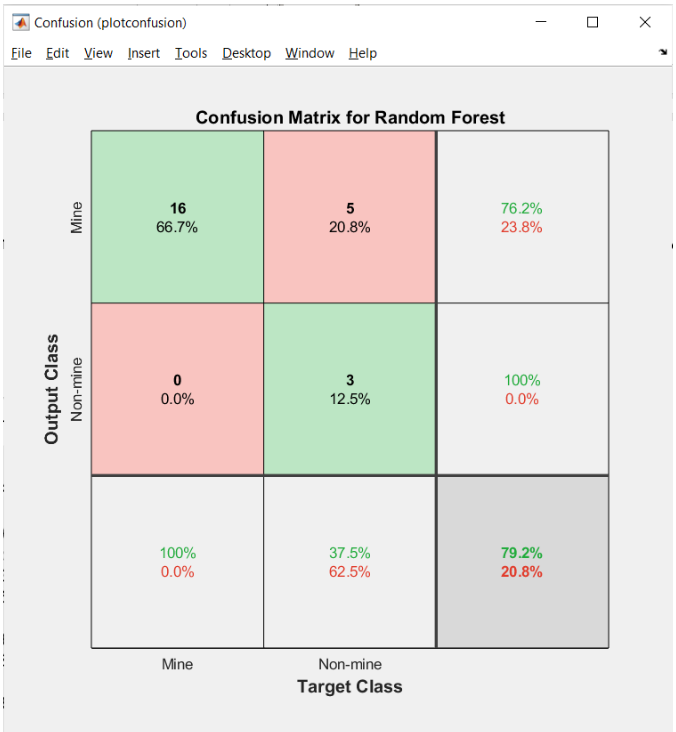

Figure 4: Confusion Matrix Result Using RF

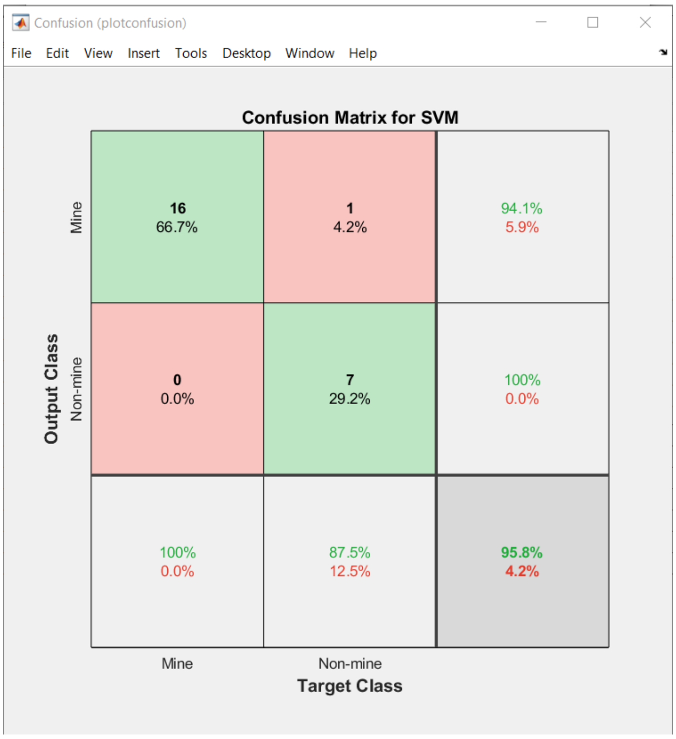

Figure 5: Confusion Matrix Result Using SVM

The performance of overall system as per the three output classes are calculated based on confusion matrix parameters and shown in figure 6 and figure 7 for RF and SVM in terms of precision, recall and f-score parameters for three output classes. It shows that mine_clean output class shows the best accuracy rate among all three classes with higher precision, recall and f-score rate as defined. Overall, SVM gives best accuracy with 95.83% as compared to RF which gives 79.16%.

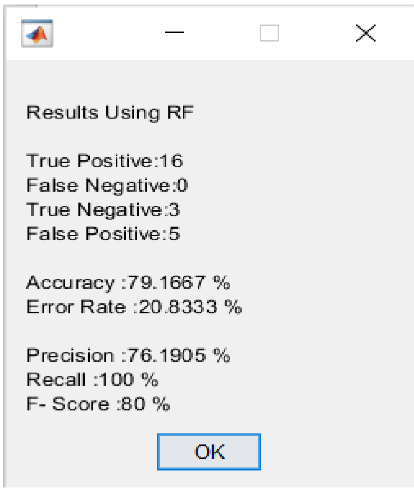

Figure 6: Result evaluation metrics using RF

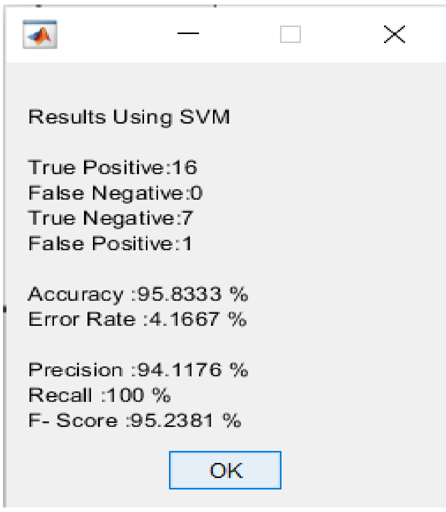

Figure 7: Result evaluation metrics using SVM

Figure 8, Figure 13, and Figure 18 show the sample test input bscan image T2R3_X24, T2R3_X25 and T2R3_X26 of each category mine_clean, mine_rock and nothing_clean. These images are pre-processed through image processing operations and provide preprocess images as shown in figure 9, figure 14 and figure 19 for each output class respectively. Figure 10, figure 15 and figure 20 shows the evaluation time for result prediction using RF classifiers and their respective output label are displayed in figure 11, figure 16 and figure 21. Figure 12 and Figure 17 represent the detected object in mine_clean and mine_rock images.

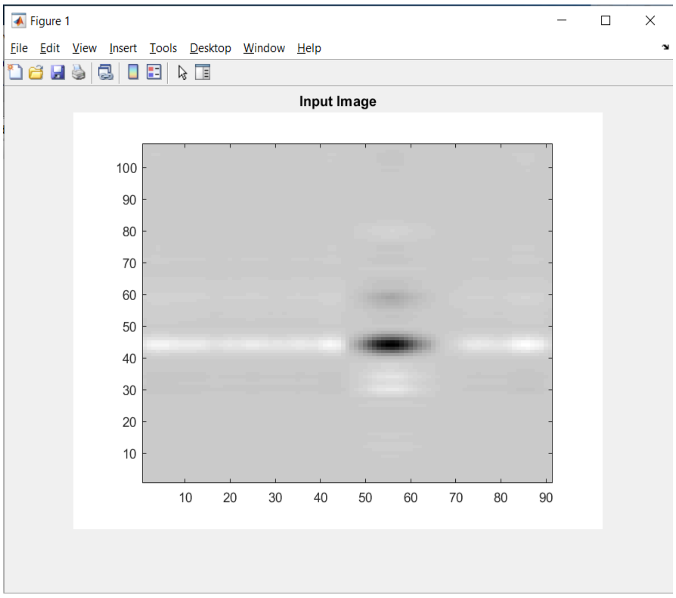

Figure 8: Input T2R3_X24 Mine_clean Bscan Test Image

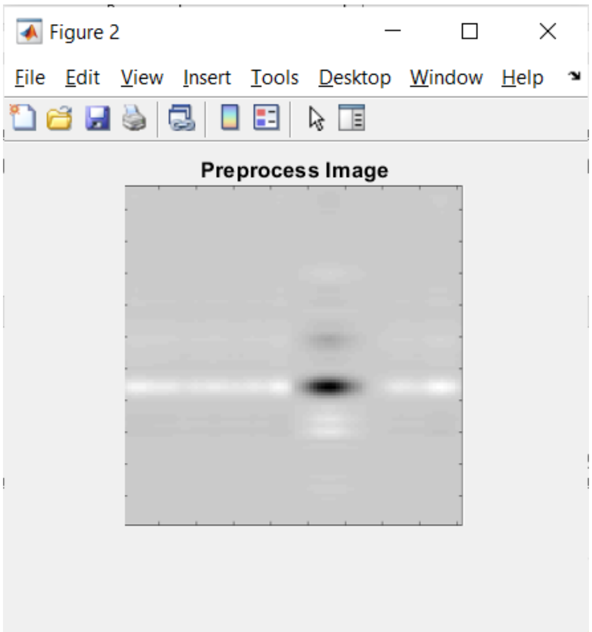

Figure 9: Preprocess T2R3_X24 Mine_clean Bscan Test Image

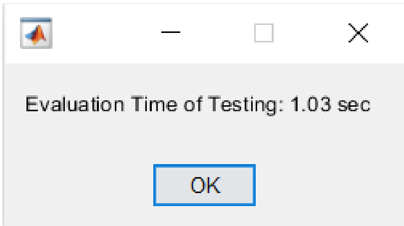

Figure 10: Evaluation Time for Testing Mine_clean Bscan Image

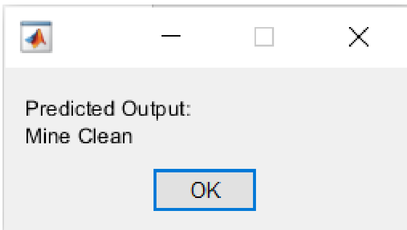

Figure 11: Predicted Output for Mine_clean Bscan Image

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 02 | Feb – 2025 SJIF Rating: 8.448 ISSN: 2582-3930

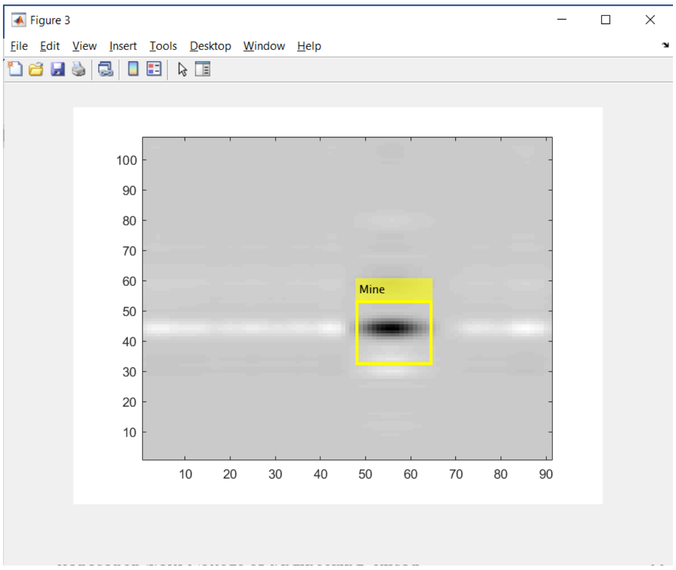

Figure 12: Mine Object Detection for Mine_clean Bscan Image

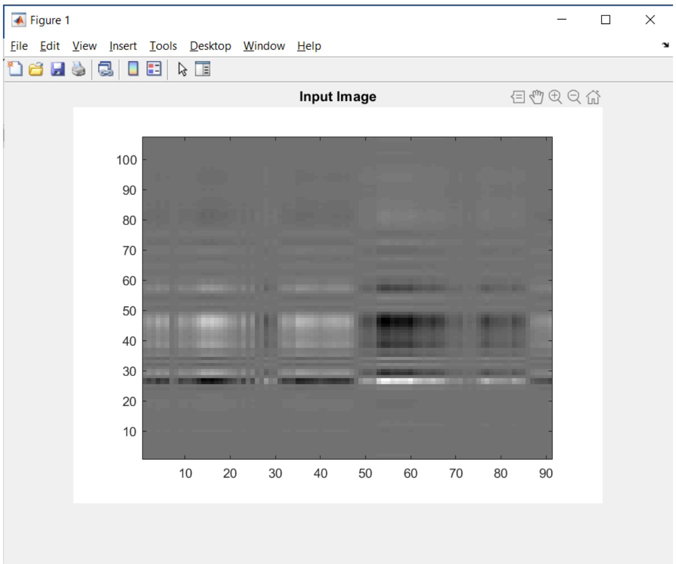

Figure 13: Input T2R3_X25 Mine_rock Bscan Test Image

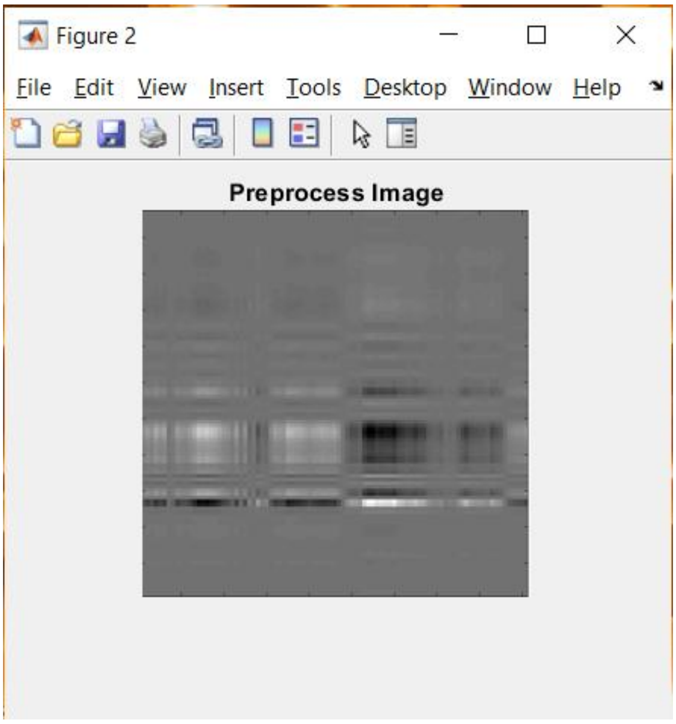

Figure 14: Preprocess T2R3_X25 Mine_rock Bscan Test Image

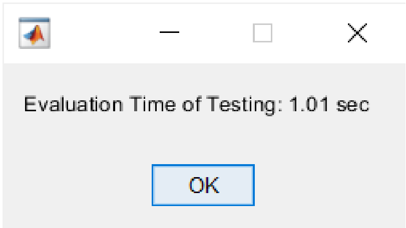

Figure 15: Evaluation Time for Testing Mine_rock Bscan Image

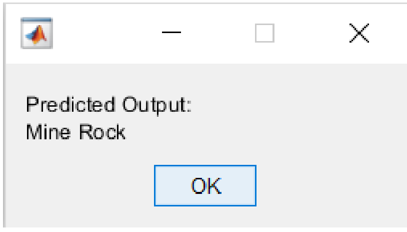

Figure 16: Predicted Output for Mine_rock Bscan Image

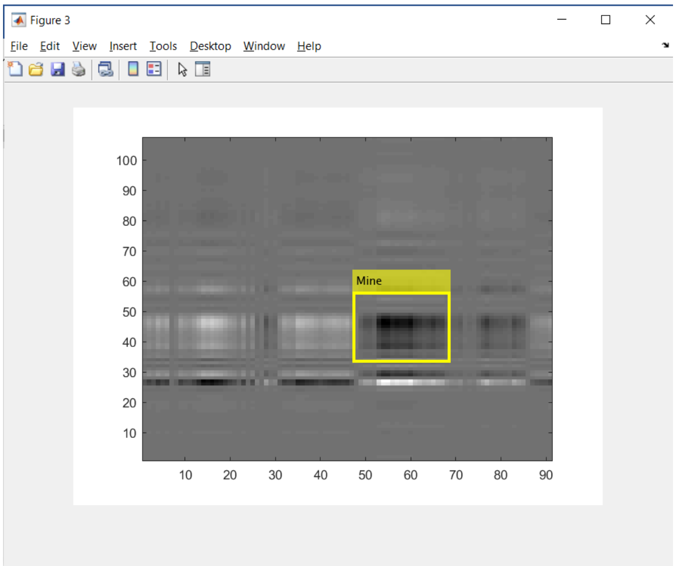

Figure 17: Mine Object Detection for Mine_rock Bscan Image

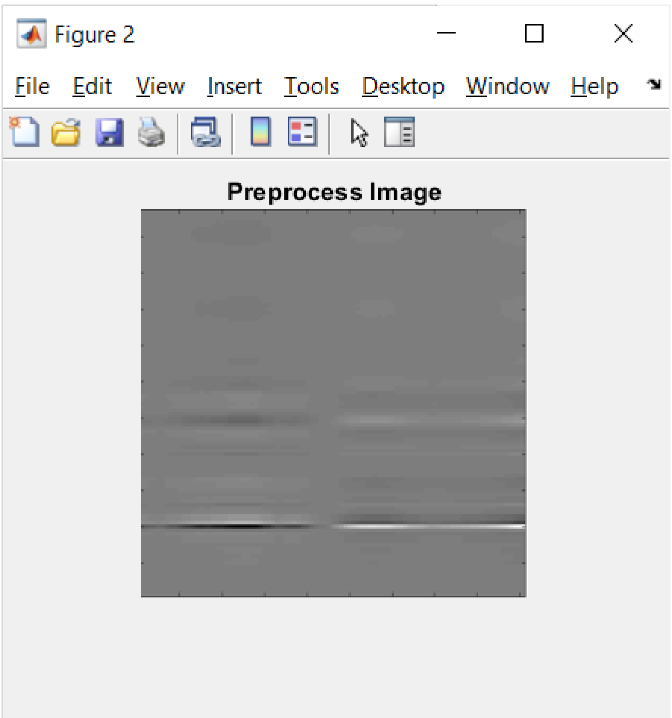

Figure 19: Preprocess T2R3_X26 Nothing_clean Bscan Test Image

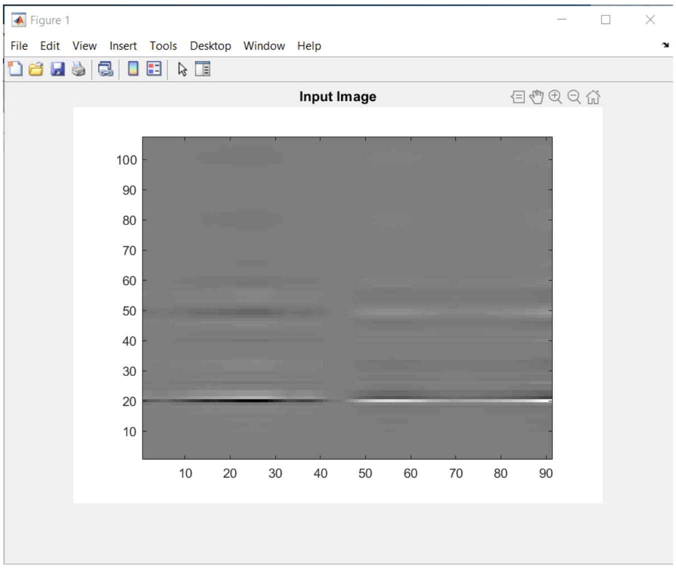

Figure 18: Input T2R3_X26 Nothing_clean Bscan Test Image

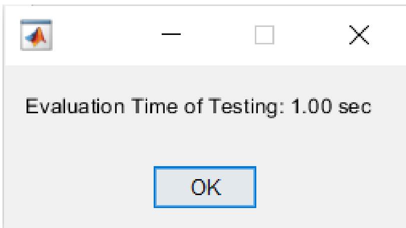

Figure 20: Evaluation Time for Testing Nothing_clean Bscan Image

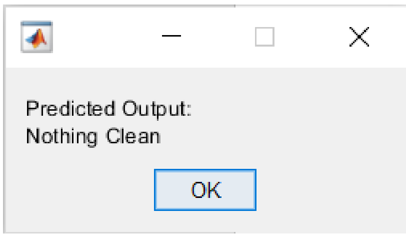

Figure 21: Predicted Output for Nothing_clean Bscan Image

5. CONCLUSIONS

This paper successfully demonstrated the application of machine learning classifiers, specifically Support Vector Machine (SVM) and Random Forest (RF), for automatic target classification based on Ground Penetrating Radar (GPR) data, using texture-based feature extraction techniques. Both classifiers demonstrated high accuracy and reliability in classifying targets when trained with appropriately selected GLCM and HOG features, which highlight the texture differences between target and non-target objects. The use of texture features, derived from GPR B-scan images, proved effective in capturing the unique characteristics of subsurface targets, such as landmines, and distinguishing them from non-target materials like rocks clutter. In proposed experimentation, SVM classifier provides better accuracy with 95.8% as compared to the RF classifier and detects the mine object correctly in bscan image. This approach has practical implications for landmine detection, archaeological exploration, and infrastructure assessment, providing a more reliable and automated solution to identifying hidden objects based on GPR signals.

Future research can expand upon this research by exploring deep learning architectures, incorporating additional signal processing techniques, and implementing real-time classification systems for field deployment.

ACKNOWLEDGEMENT

I would like to thanks **Dr. Waymond R. Scott,**(*Professor* School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta), for providing his research data in completion of this work.

REFERENCES

[1] A. Rangole, D. Sampurna, N. Kuchekar, and A A Bazil Raj, “A Comprehensive Review of Ground Penetrating Radar: Techniques, Applications and Future Directions,” International Journal of Engineering Research and Reviews, Vol. 12, Issue 3, pp. 30-53, July 2024 - September 2024, Doi: http://dx.doi.org/10.5281/zenodo.13842586.

[2] N. Li, H. Li, and R. Wu, “Ground Penetrating Radar Based Subsurface Object Detection: Mini Review,”, COJ Robotics & Artificial Intelligence, Vol. 1, Issue 3, Feb 2021, doi: http://dx.doi.org/10.31031/cojra.2021.01.000515.

[3] X. Sui, Z. Leng, and S. Wang, “Machine learning based detection of transportation infrastructure internal defects using ground penetrating radar: a state-of-the-art review,” Intelligent Transportation Infrastructure, Volume 2, 2023, June 2023, doi: http://dx.doi.org/10.1093/iti/liad004.

[4] D. Chen, S. Xiong, and L. Guo, “Research on Detection Method for Tunnel Lining Defects Based on DCAM-YOLOv5 in GPR B-Scan,” Radio Engineering. 2023 vol. 32, č. 3, s. pp. 299-311. ISSN 1210-2512, doi: http://dx.doi.org/10.13164/re.2023.0299.

[5] G. Jiangkun, et al. "An Introduction to Radar Automatic Target Recognition (ATR) Technology in Ground-based Radar Systems." ArXiv, 2023, doi: https://arxiv.org/abs/2310.14769.

[6] F. Hou, W. Lei, S. Li and J. Xi, "Deep Learning-Based Subsurface Target Detection From GPR Scans," in IEEE Sensors Journal, vol. 21, no. 6, pp. 8161-8171, 15 March15, 2021, doi: http://dx.doi.org/10.1109/JSEN.2021.3050262.

[7] Z. Xiang, G. Ou, A. Rashidi, and A. H. Mashhadi, “A Shearlet Transform-Based method for automated rebar recognition using GPR B-Scan image data,” Construction Research Congress 2022, vol. 112, pp. 30–39, Mar. 2022, doi: https://doi.org/10.1061/9780784483961.004.

[8] D. Kim, J. Kim and H. Youn, “Automated ground penetrating radar B-scan detection enhanced by data augmentation techniques”, Geomechanics and Engineering, Volume 38, Number 1, July 10 2024, pages 29-44. DOI: https://doi.org/10.12989/gae.2024.38.1.029.

[9] Y.-C. Zhang, T.-H. Yi, S. Lin, H.-N. Li, and S. Lv, “Automatic Corrosive Environment Detection of RC Bridge Decks from Ground-Penetrating Radar Data Based on Deep Learning,” Journal of Performance of Constructed Facilities, vol. 36, no. 2, Feb. 2022, doi: https://doi.org/10.1061/(ASCE)CF.1943-5509.0001712.

[10] L. C. M. Amaral, A. Roshan, and A. Bayat, “Automatic detection and classification of underground objects in ground penetrating radar images using machine learning,” Journal of Pipeline Systems Engineering and Practice, vol. 14, no. 4, Sep. 2023, doi: https://doi.org/10.1061/JPSEA2.PSENG-1444.

[11] Y.-T. Dou, G.-Q. Dong, and X. Li, “Automatic identification of GPR targets on roads based on CNN and Grad-CAM,” Applied Geophysics, Jun. 2024, doi: https://doi.org/10.1007/s11770-024-1105-8.

[12] L. Hai, et al. "Automatic Recognition and Localization of Underground Pipelines in GPR B-scans Using a Deep Learning Model." Tunnelling and Underground Space Technology, vol. 134, 2023, p. 104861, doi: https://doi.org/10.1016/j.tust.2022.104861.

[13] K. Chen, W. Xue, Z. Wang and S. Ma, "Automatic Target Recognition in GPR Images Based on Time-Frequency Texture Features and Support Vector Machine," 2021 China Automation Congress (CAC), Beijing, China, 2021, pp. 2876-2881, doi: https://doi.org/10.1109/CAC53003.2021.9727358.

[14] M. S. Z. Azalan, T. Esian, H. Ali, A. F. A. Zaidi, and T. S. T. Amran, “Classification Size of Underground Object from Ground Penetrating Radar Image using Machine Learning Technique,” Journal of Physics Conference Series, vol. 2550, no. 1, p. 012029, Aug. 2023, doi: https://doi.org/10.1088/1742-6596/2550/1/012029.

[15] M.-S. Kang, N. Kim, J. J. Lee, and Y.-K. An, “Deep learning-based automated underground cavity detection using three-dimensional ground penetrating radar,” Structural Health Monitoring, vol. 19, no. 1, pp. 173–185, Mar. 2019, doi: https://doi.org/10.1177/1475921719838081.

[16] D. Dewantara and W. W. Parnadi, “Automatic hyperbola detection and apex extraction using convolutional neural network on GPR data,” Journal of

Physics Conference Series, vol. 2243, no. 1, p. 012027, Jun. 2022, doi: https://doi.org/10.1088/1742-6596/2243/1/012027.
[16] C. Xin, W. Lei, C. Luo and W. Xue, "Ground penetrating radar target detection method based on optimized faster RCNN and k-means clustering," IET International Radar Conference (IRC 2023), Chongqing, China, 2023, pp. 70-73, doi: https://doi.org/10.1049/icp.2024.1053.
[17] H. Harkat, A. E. Ruano, M. G. Ruano, and S. D. Bennani, "GPR target detection using a neural network classifier designed by a multi-objective genetic algorithm," Applied Soft Computing, vol. 79, pp. 310–325, Apr. 2019, doi: https://doi.org/10.1016/j.asoc.2019.03.030.
[18] Z. Qiu, J. Zeng, W. Tang, H. Yang, J. Lu, and Z. Zhao, "Research on Real-Time Automatic Picking of Ground-Penetrating Radar image features by using Machine Learning," Horticulturae, vol. 8, no. 12, p. 1116, Nov. 2022, doi: https://doi.org/10.3390/horticulturae8121116.
[19] Z. Gong and H. Zhang, "Research on GPR image recognition based on deep learning," MATEC Web of Conferences, vol. 309, p. 03027, Jan. 2020, doi: https://doi.org/10.1051/matecconf/202030903027.
[20] G. Cui, J. Xu, Y. Wang and S. Zhao, "Multiple Objects Automatic Detection of GPR Data Based on the AC-EWV and Genetic Algorithm," in IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1-16, 2023, Art no. 5100516, doi: https://doi.org/10.1109/TGRS.2022.3228571.
[21] V. Motyka et.al., "Radar Target Recognition Based on Machine Learning", IntelITSIS'2023: 4th International Workshop on Intelligent Information Technologies and Systems of Information Security, March 22–24, 2023, Khmelnytskyi, Ukraine.
[22] Y. Zhong, K. Li, S. Mo, and X. Liu, "GPR Target Recognition Based on Improved YOLOv3-SPP," Preprints.org; 2023. DOI: https://doi.org/10.20944/preprints202305.1363.v1.
[23] W. Xue, K. Chen, T. Li, L. Liu, and J. Zhang, "Efficient underground target detection of urban roads in Ground-Penetrating radar images based on neural networks," Remote Sensing, vol. 15, no. 5, p. 1346, Feb. 2023, doi: https://doi.org/10.3390/rs15051346.
[24] J. Zou, H. Zheng, and F. Wang, "Real-Time target detection system for intelligent vehicles based on Multi-Source data fusion," Sensors, vol. 23, no. 4, p. 1823, Feb. 2023, doi: https://doi.org/10.3390/s23041823.
[25] T. Counts, A. C. Gurbuz, W. R. Scott, J. H. McClellan and K. Kim, "Multistatic Ground-Penetrating Radar Experiments," in IEEE Transactions on Geoscience and Remote Sensing, vol. 45, no. 8, pp. 2544-2553, Aug. 2007, doi: https://doi.org/10.1109/TGRS.2007.900677.
[26] https://waymond-scott.ece.gatech.edu/multistaticbeamformingdata/multistaticbeamformingdata/.

BIOGRAPHIES

Gaurav R. Savade, has received the B.E. Electronics and Tele-communication from Sant Gadge Baba Amravati university, Amravati 2008. He is currently doing M.E. in Digital Electronics from Babasaheb Naik College of Engg., Pusad from Sant Gadge Baba, Amravati University, Amravati. He worked as an Assistant Professor at Babasaheb Naik College Of Engineering, Pusad. His research interests include digital image processing, VLSI circuits and Embedded system design.