

ScanCloud: Holistic GPR Data Analysis for Adaptive Subsurface Object Detection

Maxwell M. Omwenga¹, Dalei Wu^{1*}, Yu Liang¹, Dryver Huston², Tian Xia³

¹ Department of Computer Science and Engineering, University of Tennessee at Chattanooga, USA

² Department of Mechanical Engineering, University of Vermont, USA

³ Department of Electrical and Biomedical Engineering, University of Vermont, USA

Email: dgh179@mocs.utc.edu; {dalei-wu, yu-liang}@utc.edu; {dryver.huston, txia}@uvm.edu

Abstract—The conventional ground penetrating radar (GPR) data analysis methods, which use piecemeal approaches in processing the GPR data formulated in variant formats such as A-Scan, B-Scan, and C-Scan, fail to provide a global view of underground objects on the fly to adapt the operations of GPR systems in the field. To bridge the gap, in this paper, we propose a novel GPR data analysis approach termed "ScanCloud" which is focused on the whole in situ GPR dataset rather than on individual A-Scans, B-Scans or C-Scans. We also study the integration of ScanCloud and a deep reinforcement learning method called deep deterministic policy gradient (DDPG) to adapt the operation of GPR system. The proposed method is evaluated using GPR modeling software called GprMax. Simulation results show the efficacy of ScanCloud and the adaptive GPR system enabled by the integration of ScanCloud and DDPG.

Index Terms—Autonomous Cognitive GPR, deep reinforcement learning, subsurface sensing, object reconstruction, 3D ScanCloud

I. INTRODUCTION

Accurate 3D maps for underground infrastructure, like gas, water, and sewage pipes, are significant for governments, service organizations, and structural architects. Nonetheless, exact locations and conditions of underground infrastructure in old cities are generally unknown. A ground-penetrating radar (GPR) is a significant device for locating and identifying underground objects [1]. However, there are some limitations with conventional GPR techniques: piecemeal analysis of different formats of GPR data (A-Scan, B-Scan, and C-Scan) and offline interpretation of GPR data with the need of domain expert experience. These limitations result in inaccurate and incomplete knowledge of underground objects and time-consuming operation of GPR systems.

To address these limitations, we propose a new approach termed ScanCloud that considers all the GPR A-Scans in the field as a point cloud. The rationale behind ScanCloud is that holistically analyzing all the in-situ available GPR data may provide more knowledge than focusing on partial A-Scan data or individual B-Scan images. Specifically, we model the GPR sensing process by integrating the 1D A-Scan signals and analyzing them to extract (x, y, z) coordinates of the signal where the amplitude pulse from the object is

detected. This information is stored in a database to form a novel 3D ScanCloud dataset. This allows us to develop new data point each time an A-Scan is received.

We also integrate ScanCloud analysis and deep reinforcement learning (DRL) to adapt the operation of a GPR system. To that end, we first apply Rényi entropy and modified 3D OTSU methods to detect the region of interest (RoI) in the ScanCloud. The RoI is transformed into a signature that serves as input into an object classifier. All the previous procedures result in a reward function for the deep deterministic policy gradient (DDPG) [2], [3], a typical DRL method dealing with large and continuous state and/or action spaces. This paper is focused on the development of a DDPG framework that enables adaptive GPRs for subsurface object detection. To this end, a proper reward function is needed to effectively reflect the value of different actions of the GPR agent at different states. To the best of our knowledge, this is the first work on subsurface object detection based on ScanCloud and DDPG. The main contribution of this paper can be summarized as follows.

- 1) By formulating GPR-based subsurface object detection as a Markov decision problem, a 3D data modality and a deep reinforcement learning framework is established to resolve the problem.
- 2) A deep deterministic policy gradient (DDPG) algorithm with a novel reward function that combines rewards from amplitude analysis, Region of Interest (RoI) identification and object classification is proposed.
- 3) To show the efficacy of the proposed framework, simulation based validations are performed on real-time GPR data from GprMax simulator by combining DDPG with GPR operation modeling [4], [5].

In this paper, we investigate continuous subsurface object detection to generate object shape and orientation using DDPG [6]. The rest of this paper is organized as follows: the related work is presented in Section II. In Section III, an overview of the system model and architecture of the proposed ScanCloud is presented. The proposed DDPG approach is discussed in Section IV. Section V presents performance evaluation and discussion. Finally, Section VI concludes the paper.

II. RELATED WORK

Among different sensor modalities, GPRs have been widely used in subsurface target detection. However, mapping the

*Corresponding Author

This work was supported by the National Science Foundation under grant numbers 1647175 and 1924278.

underground targets from GPR signals is nontrivial, because different from a laser scanner, a GPR cannot provide 3D positions by any other means but C-Scans. Feng *et al.* [7] designed a GPR-based model reconstruction system for underground utilities using an auto-encoder. Yang *et al.* [8] also designed a GPR-based subsurface object detection and reconstruction using random motion and depthnet. Chen *et al.* [9] proposed an automatic concrete crack-detection method fusing point clouds and images based on improved Otsu's algorithm.

Recent research such as combining visual exploration of 2D ground data and 3D point cloud data for roads environment [10], [11], [12] is gaining attention. This method [10] is designed to inspect road surfaces, manholes covers and gullies. Weilin *et al.* [11] proposed a method that detects tree trunks with irregular contours, using LiDAR data and 2D images from GPR scanner. The authors in [12] designed a two-step process that involves a grid data creation, where through interpolation, the point cloud data is changed into grid data suitable for filling the empty data, and then enhancing the homogeneous points.

Point cloud has become one of the most significant data format for 3D representation. It's gaining increased popularity as a result of increased availability of acquisition devices, such as LiDAR, as well as increased application in areas such as robotics, autonomous driving, augmented and virtual reality. Jürgen describes the relevance of 3D point clouds for a large number of geospatial applications through machine learning and deep learning [13]. Other sophisticated applications of point clouds include but not limited to connected autonomous vehicles [14], ground points segmentation for challenging terrains for autonomous vehicles [15], and in modeling and predicting vehicle accident occurrence in smart cities [16].

III. THE PROPOSED SYSTEM MODEL AND ARCHITECTURE

A. MDP Formulation of Subsurface Detection

The cognitive control of the positioning and operational parameters of a GPR can be formulated as a sequential decision-making problem which can be further modeled as a finite-horizon Markov decision process (MDP) with finite state and action spaces.

Without loss of generality, we consider a discrete-time system in which time is divided into slots of unit length ΔT such that each slot t corresponds to the time duration $[(t-1) \cdot \Delta T, t \cdot \Delta T)$. The complete notation used in this paper is given in Table I.

The MDP model is described as follows:

- $\mathcal{S}$: a set of environment and system operational states. Let $s_t = (\kappa_t, \Psi_t) \in \mathcal{S}$ denote the state of the GPR sensing system and the environment in each discrete time slot t . κ_t is the newly updated observation about the environment, in the form of captured A-Scan signal. Ψ_t is the operating state vector of the GPR, such as the remaining battery energy of the mobile GPR platform and the agent's position $X_t \in \mathbb{C}$ (a complex number), i.e., $X_t = x_t + jy_t$, representing the GPR location with coordinates (x_t, y_t) .

TABLE I
NOMENCLATURE

ScanCloud	
x_t, y_t, z_t	The x, y , and z coordinates of the received pulse signal
φ	Amplitude
$ m_a(i) $	Signal Envelope
r_t^{amp}	Amplitude reward
r_t^{roi}	Region of interest (RoI) reward
r_t^{sc}	Object classification reward
DDPG	
θ^Q	Q network
θ^μ	Deterministic policy function
$\theta^{Q'}$	Target Q network
$\theta^{\mu'}$	Target policy network
$r_t \in \mathbb{R}$	Reward at time step t

- $\mathcal{A}$: a set of actions of the GPR. Let $a_t = (\xi_t, \vec{v}_t, \vec{p}_t) \in \mathcal{A}$ denote the action vector to be performed at time step t where $\vec{p}_t$ is the operational parameter values of the GPR; $(\xi_t, \vec{v}_t)$ denote the moving direction and velocity of the GPR platform, respectively. Thus, the position of the GPR at time step t can be derived as $X_t = X_{t-1} + \vec{v}_t \cdot \Delta T \cdot e^{j\xi_t}$.

In reinforcement learning for discrete action spaces, exploration is done via probabilistically selecting a random action such as ϵ -greedy. For continuous action spaces, exploration is done via adding noise to the action itself. In this paper we will investigate a DDPG framework where a GPR agent is reinforced to learn a Q-function and a policy.

B. The Architecture of the ScanCloud based Cognitive GPR

In this section, we present an overview of the proposed ScanCloud architecture, as shown in Figure 1. The architecture has an iterative operational process involving environment observation, reward identification, DDPG-based policy learning, and action execution. The observations (A-Scan signals) from the agent are feed into the amplitude analyzer for x, y, z coordinates extraction, which are stored in the ScanCloud database. From the accumulated ScanCloud points a region of interest (RoI) is detected in the object formation module where the RoI is made of point with high amplitude values signifying the present of a subsurface object. These RoI points are identified and extracted through the modified 3D OTSU technique and Rényi entropy [17], [18]. The pre-trained classifier receives the RoI signature representation as input for classification. The classification probability output is used to characterize the classification confidence. The output results from the amplitude analyze module, RoI module, and classification module are used to form the reward for the GPR agent, which will be described in Section IV-A.

The DDPG module takes a tuple of state, action, reward, and future state as experience, and guides the GPR agent to learn a Q-function and optimum policy that maximizes the future discounted reward for continuous action spaces. The algorithm of the DDPG module will be described in Section IV-B. DDPG uses off-policy data and the Bellman equation to learn the Q-function, and uses the Q-function to learn the policy.

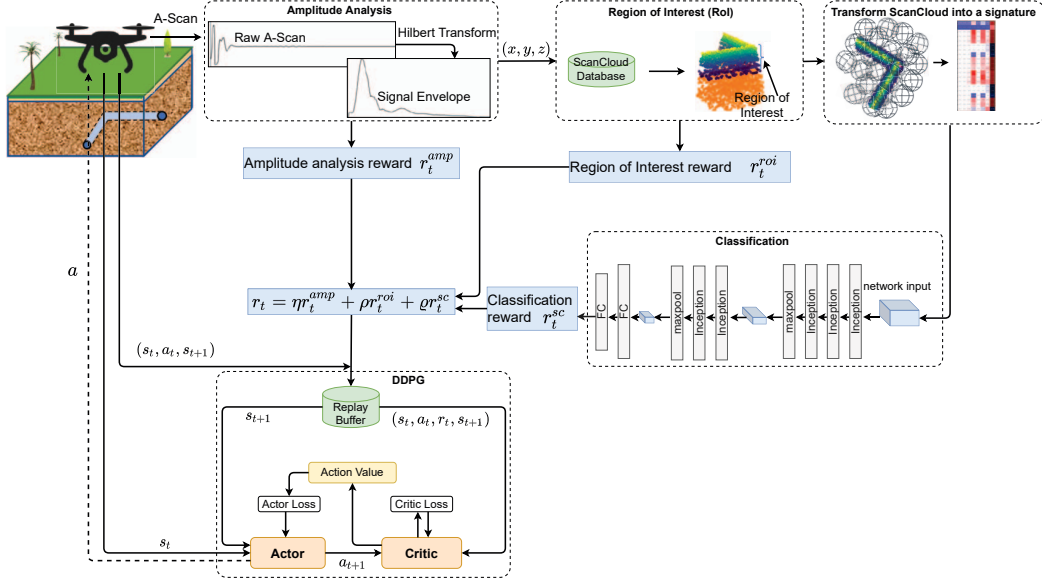


Fig. 1. The iterative operational process of the proposed 3D ScanCloud and DDPG enabled GPR.

IV. THE PROPOSED SCANCLOUD APPROACH

The region of interest (RoI) module in Figure 1 is a critical component of the proposed method. It produces intelligence to direct the GPR movement and its operational configurations based on the collected ScanCloud data and prior knowledge about GPR measurement. This section presents a DDPG approach to the implementation of the 3D subsurface object detection method with a novel rewarding mechanism.

A. Reward Function

The GPR agent is rewarded through the combined outcomes from amplitude analysis, RoI detection and object classification while interacting with the environment. The reward function $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is derived by combining three types of rewards that are computed based on Hilbert Transform, alpha shapes and Shannon entropy, respectively, effectively characterizing the GPR agent's newly acquired subsurface knowledge about the subsurface object from the sensory data.

The rationale behind this combination of the three sub-rewards is that the GPR agent would receive reward r_t^{amp} from analysing the A-Scan's amplitude, r_t^{roi} when it identifies an RoI in the A-Scan image and receive reward r_t^{sc} when it recognizes some object properties, such as the diameter and material of a subsurface pipeline, through GPR data classification.

Thus the overall reward function is

$$r_t(s_t, a_t) = \eta r_t^{amp} + \rho r_t^{roi} + \varrho r_t^{sc}, \quad (1)$$

where r_t^{amp} denotes the amplitude analysis reward, r_t^{roi} the region of interest (RoI) detection reward, r_t^{sc} object recognition reward and η, ρ, ϱ denote the weight coefficients whose values are determined based on the relative importance of the amplitude analysis reward, RoI detection reward and the object recognition reward.

In this subsection, we briefly present the concept of amplitude analysis reward in Subsection IV-A-1, RoI detection reward in Subsection IV-A-2, and then describe subsurface object classification reward in Subsection IV-A-3.

1) *Reward Based on Amplitude*: GPR works by sending a signal from a receiver into a surface. The signal is reflected off of any materials it encounters within the surface, and creates a reading. Each reflected A-Scan pulse from the reflected signal off the subsurface object is examined to produce a tuple (x, y, z) that forms the coordinates of the ScanCloud points, where x, y denote the x and y coordinates of the GPR scanner, and z denotes A-Scan's two way travel time. The z of the reflected electromagnetic pulse signal is analyzed for amplitude strength. These signals are pre-processed through the following steps: 1) We stack every β A-scan traces then calculate the average to boost the signal-to-noise ratio (SNR). The selection of β traces for calculation considers the balance between the obtainable signal resolution and noise reduction performance. 2) A signal pulse envelope is extracted through Hilbert Transform which measures the signal power. The Hilbert Transform of the i th A-Scan trace $m(i)$ can be considered as the convolution of $m(i)$ with the function $h(i) = \frac{1}{\pi i}$ which can be expressed as

$$\hat{m}(i) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{m(\tau)}{i - \tau} d\tau, \quad (2)$$

where $\hat{m}(i)$ is the direct output of the Hilbert Transform of $m(i)$. The magnitude of the analytical signal $m_a(i)$ equals

$$|m_a(i)| = \sqrt{m(i) + \hat{m}(i)}, \quad (3)$$

where $|m_a(i)|$ denotes the envelope of $m(i)$, which facilitates the signal power characterization. 3) Search for the amplitude pulse φ_t of the reflected A-Scan signal that corresponds to the

subsurface object in the envelope signal $|m_a(i)|$ and record the amplitude reward as

$$r_t^{amp} = |m_a(i)|[“Lookup”, \varphi_t]. \quad (4)$$

2) *Reward Based on ScanCloud*: Through the Rényi entropy method, separate computation was conducted with respect to the x, y, z coordinates of the ScanCloud points. Rényi entropy is preferred because of its high level of accuracy on signal processing tasks compared to Tsallis [19], [20], [21].

In this work Rényi entropy is calculated to recognize the singular region on a ScanCloud. In particular, a high Rényi entropy value demonstrates a high level of information similarity while a low Rényi entropy value features a high level of information peculiarity [18]. The Rényi entropy upon the ScanCloud is calculated as

$$E_\alpha(x) = \frac{1}{1-\alpha} \log_e \sum_{f=1}^F [z_f(x)]^\alpha, \quad (5)$$

where $E_\alpha(x)$ is the entropy quantification with respect to the x axis, $z_f(x)$ is the normalized signal, f is the coordinate index of pulse and F is the total number of x indexes; $\hat{x} = \delta t \cdot \vec{v}$ is the displacement along the trace x axis.

$$E_\alpha(y) = \frac{1}{1-\alpha} \log_e \sum_{g=1}^G [z_g(y)]^\alpha, \quad (6)$$

where $E_\alpha(y)$ is the entropy quantification with respect to the y axis, $z_g(y)$ is the normalized signal, g is the coordinate index of pulse and G is the total number of y indexes; $\hat{y} = \delta t \cdot \vec{v}$ is the displacement along the trace y axis.

$$E_\alpha(z) = \frac{1}{1-\alpha} \log_e \sum_{h=1}^H [z_h(z)]^\alpha, \quad (7)$$

where $E_\alpha(z)$ is the entropy quantification with respect to the z axis, $z_h(z)$ is the normalized signal, h is the trace index, H is the total number of traces included, and z is the time index of pulse data on each reflection A-Scan trace waveform.

The region of interest (RoI) signifying the area where the subsurface object is located, is computed through the 3D OTSU threshold method. The 3D OTSU is our modified version of OTSU, which is suitable for 3D ScanCloud data. With two selected entropy thresholds $\mathfrak{I}1$ and $\mathfrak{I}2$, the ScanCloud can be segmented into three classes of non-overlapping regions: singular region, stationary background region, and the transition region in-between. The singular region entropy values are lower than threshold $\mathfrak{I}1$, the stationary background region entropy values are higher than $\mathfrak{I}2$. While for the transitioning region, its entropy values are between these two thresholds. The 3D OTSU method was applied to all ScanCloud points to generate three Rényi entropy plots corresponding to the x, y, z axes, where respective optimum thresholds were computed $\mathfrak{I}1_x^*, \mathfrak{I}1_y^*, \mathfrak{I}1_z^*$. The Rényi entropy points which fall below intersection between the Rényi entropy and 3D OTSU $\mathfrak{I}1_x^*, \mathfrak{I}1_y^*, \mathfrak{I}1_z^*$ are extracted and the corresponding ScanCloud points are selected to form the intermediate first, second and third, parts of the global RoI ScanCloud points, which is calculated as

$$roi = \mathcal{X}^* \cap \mathcal{Y}^* \cap \mathcal{Z}^*, \quad (8)$$

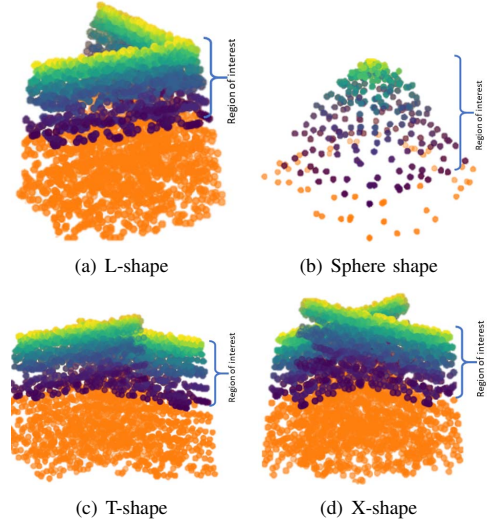


Fig. 2. Region of interest from various underground object shapes: L-shape, sphere shape, T-shape, and X-shape

$$\mathcal{X}^* = \{x \mid E_\alpha(x) < \mathfrak{I}1_x^*\}, \quad (9)$$

$$\mathcal{Y}^* = \{y \mid E_\alpha(y) < \mathfrak{I}1_y^*\}, \quad (10)$$

$$\mathcal{Z}^* = \{z \mid E_\alpha(z) < \mathfrak{I}1_z^*\}, \quad (11)$$

where $\mathcal{X}^*, \mathcal{Y}^*$, and $\mathcal{Z}^*$ are sets of intermediate RoI points of ScanCloud with respect to x, y , and z axis respectively. All the three intermediate RoIs are finally concatenated to form the global RoI denoted as *roi*. The global RoI samples are shown in Figures 2(a) - 2(d), where the top part of the ScanCloud (non-orange ScanCloud points) are detected as the RoI. This region is special because it explicitly indicates the region where the subsurface object is located.

Let a, b, c and d denote the sides of its tetrahedrons of the global RoI *roi*. We calculate the volume covered by the global RoI which is considered as the RoI reward, which can be expressed as

$$r_t^{roi} = \frac{\kappa}{1 + e^{-V^{roi}}} \quad (12)$$

where V^{roi} indicates the volume of the region-of-interest formulated by tetrahedral elements, namely

$$V^{roi} = \sum_{p=1}^P \frac{|(a-d) \cdot ((b-d) \times (c-d))|}{6}. \quad (13)$$

In Eq. (13), P is the total number of ScanCloud tetrahedral elements.

3) *Reward Based on Subsurface Object Classification*: The obtained RoI is transformed into a signature through Gaussian mixture model (GMM) method [22]. For the underlying density model, we use a mixture of Gaussians with Gaussian centers (μ_k) on a uniform 3D $m \times m \times m$ grid. Such Gaussians induce a Fisher vector [23] that preserves the point set structure: the presence of points in a specific 3D location would significantly influence only some, pre-known, Fisher components. The other GMM parameters, weight and covariance, are common to all Gaussians. Figure 3 depicts a

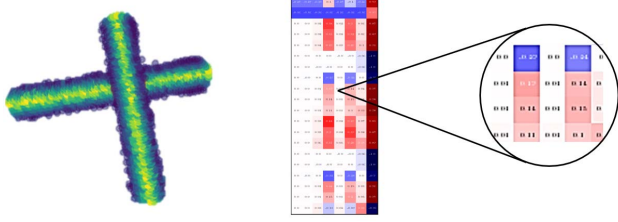


Fig. 3. X-shaped ScanCloud (left) and the corresponding signature (right)

ScanCloud (left) and its corresponding signature representation ($m = 8$) as a color coded image (right). Each column of the image represents a single Gaussian in a $8 \times 8 \times 8$ Gaussian grid. Zero values are white whereas positive and negative values correspond respectively to the red and blue gradients. Note that the representation lends itself to intuitive interpretation. For example, many columns are white, except for the first two top entries. These correspond to Gaussians that do not have model points near them.

A 4D matrix representing the derived signature serves as input to the pre-trained 3D convolutional neural network (CNN) classifier. Subsurface objects can be recognized through different GPR data classification tasks, for example, determining the shape, material type, burial depth, and diameter depending on specific applications. Our ScanCloud method is very descriptive in that it not only provides information on object shape, and depth as the work in GPRNet [7], GPRDepthNet [8] did, but also material type.

The use of CNN is motivated by the fact that CNNs outperform other artificial neural networks on conventional computer vision tasks such as object detection and recognition [24], [25], [26] through feature learning. Let $P = \{p(1), p(2), \dots, p(N)\}$ denote the classification probability output from the classifier where $p(n)$ is the class probability that the processed ScanCloud belongs to class n , and N is the total number of classes. The possible classes depend on the specific classification task. For example, if the classification task is to determine the material type of the subsurface object, the possible classes could be different object shapes, namely sphere, T shape, L shape, X shape. As entropy is a measure of uncertainty [27], [28], in this work Shannon entropy is considered to quantify the confidence in the classification. The Shannon entropy of the classification probability distribution P can be computed as

$$r_t^{sc} = - \sum_{n=1}^N p(n) \log(p(n)). \quad (14)$$

It is inferred from Eq. (14) that a balanced classification probability distribution results in high entropy indicating high uncertainty and low classification confidence while a skewed classification probability distribution has low entropy indicating low uncertainty and high classification confidence.

B. DDPG Algorithm

In our previous work [29], we leverage deep Q-Network (DQN) method for subsurface object detection. However,

DQNs are meant for problems with a few possible actions, and are therefore not appropriate for continuous action space such as the operational process of an adaptive GPR. Nevertheless, a recently proposed Deep RL algorithm referred to as Deep Deterministic Policy Gradient (DDPG) [6] can address this issue. It combines the actor-critic classical RL approach [30] with Deterministic Policy Gradient [31].

As used in Deep Q learning, DDPG also uses a replay buffer to sample experience to update neural network parameters. During each trajectory roll-out, we save all the experience tuples (state, action, reward, next_state) and store them in a finite-sized cache called “replay buffer.” Then, we sample random mini-batches B of experience from the replay buffer when we update the value and policy networks.

The value network is updated similarly as is done in Q-learning. The updated Q value is obtained by the Bellman equation as shown in Algorithm 1 (Line 27). However the next-state Q values are calculated with the target policy network and target value network. Then, we minimize the mean-squared loss between the updated Q value and the original Q value:

$$L = \frac{1}{N} \sum_j (y_j - Q(s_j, a_j | \theta^Q))^2. \quad (15)$$

To calculate the policy loss, we take the derivative of the objective function with respect to the policy parameter. It is noted that the actor (policy) function is differentiable, so we have to apply the chain rule. But since we are updating the policy in an off-policy way with batches of experience, we take the mean of the sum of gradients calculated from the mini-batch:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_j \nabla_a Q(s, a | \theta^Q)|_{s=s_j, a=\mu(s_j)} \nabla_{\theta^\mu} \mu(s | \theta^\mu)|_{s_j}. \quad (16)$$

The GPR agent is inclined to execute the action with the highest Q-value derived from each episode. The Q-value corresponding to the pair of state and action represents an expected discounted accumulated future reward.

During learning the Q-function and the policy, the agent needs to perform action selection and execution. The agent faces the well-known exploration-exploitation dilemma of whether to exploit the current knowledge by following the learned policy or to continue to explore the uncertain environment to acquire more knowledge. To resolve the dilemma, exploration is done by adding mean-zero Gaussian noise to the actions during training. During testing, we do not add noise to the actions to see how well the policy exploits what it has learned.

As described in Algorithm 1, at first (Lines 1-4), the network parameters are initialized. To generate a more descriptive and interpretive region of interest (RoI) and object classification, an iterative process is implemented (Lines 11 – 22) where periodical RoI computation and object classification are conducted after a substantial amount of ScanCloud points have been created, resulting in reduced computational overhead.

The DDPG employs the amplitude analyzer, RoI and object classifier modules to update the proposed reward function (Lines 16 and 20), by combining the amplitude reward, RoI

Algorithm 1: DDPG algorithm

```

1 Randomly initialize critic network  $Q(s, a|\theta^Q)$  and
  actor  $\mu(s|\theta^\mu)$  with weights  $\theta^Q$  and  $\theta^\mu$ 
2 Initialize target network  $Q'$  and  $\mu'$  with weights
   $\theta^{Q'} \leftarrow \theta^Q, \theta^{\mu'} \leftarrow \theta^\mu$ 
3 Initialize replay buffer  $D$ 
4 Initialize ScanCloud buffer  $Z$ 
5 for  $episode = 1, M$  do
6   Initialize a random process  $\mathcal{N}$  for action
   exploration
7   Receive initial observation state  $s_1$ 
8   for  $t=1, T$  do
9     Select action  $a_t = \mu(s|\theta^\mu) + \mathcal{N}_t$  according to
     the current policy and exploration noise
10    Execute action  $a_t$ , capture A-Scan signal
11    if  $i < I$  then
12      Compute amplitude value  $\varphi_t$  and amplitude
      reward  $r_t^{amp}$ 
13      if  $r_t^{amp} > h$  then
14        store scan point data  $(x_t, y_t, z_t)$  in  $Z$ 
15      end
16       $r_t = \eta r_t^{amp} + C$ 
17    else
18      Compute  $\varphi_t, r_t^{amp}$  in lines (12)
19      Compute object shape & RoI rewards
       $r_t^{sc}, r_t^{roi}$ 
20       $r_t = \eta r_t^{amp} + \alpha r_t^{sc} + \beta r_t^{roi}$ 
21       $C = r_t^{sc}$ 
22       $i = 0$ 
23    end
24    Set  $s_{t+1} = (x_{t+1}, y_{t+1})$ 
25    Store transition  $(s_t, a_t, r_t, s_{t+1})$  in  $D$ 
26    Sample random batch  $(s_j, a_j, r_j, s_{j+1})$  from  $D$ 
27    Set  $y_j = r_j + \gamma Q'(s_{j+1}, \mu'(s_{j+1}|\theta^{\mu'}))|\theta^{Q'}$ 
28    Update critic by minimizing the loss in Eq.
    (15)
29    Update the actor policy using the sampled
    policy gradient in (Eq. 16)
30    Update the target networks:
     $\theta^{Q'} \leftarrow \tau \theta^Q + (1 - \tau) \theta^{Q'}$ 
     $\theta^{\mu'} \leftarrow \tau \theta^\mu + (1 - \tau) \theta^{\mu'}$ 
31  end
32 end

```

detection reward and object classification reward. The object classification reward is intuitively associated with the amount of the information acquired about the subsurface object from the observed A-Scan pulses. Next (Lines 25 - 14), a defined amount of experiences are stored into the replay buffer in order for the algorithm to have a stable behavior. Then the minibatch method is used to randomly collect examples from the replay buffer. The weights and biases of the network are updated by training the DDPG according to the loss function Eq.(15). The training process will terminate once it reaches a predefined number of episodes. During each episode, the GPR

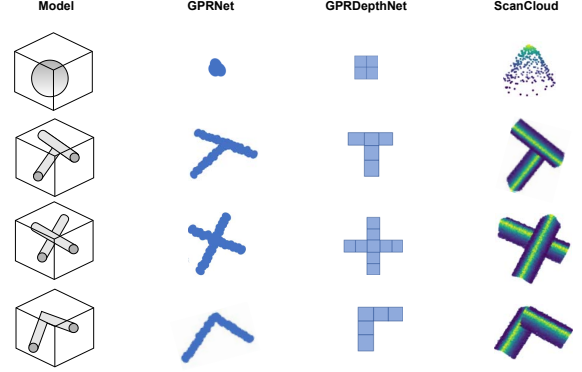


Fig. 4. The comparison of reconstruction results between GPRNet, GPRDepthNet, and our method ScanCloud.

agent stops performing actions after a predefined number of time steps, or could terminate the episode early if it detected the subsurface object. The computational complexity of the algorithm is expressed as $O(MT)$, where M denotes the total number of episodes and T the number of time steps.

V. PERFORMANCE EVALUATION AND DISCUSSION

In this section, we systematically evaluate the performance of the proposed method by using a GPR simulator called GprMax [4], [5] designed for modeling GPR operations. A-Scan signals are generated by GprMax on the fly in mimicing the GPR data acquisition in real environment. From these A-Scan signals a novel ScanCloud data format is formulated. ScanCloud is a 3D data format that incorporates derived information about the object, for example, object shape, signal amplitude, object depth, and object diameter.

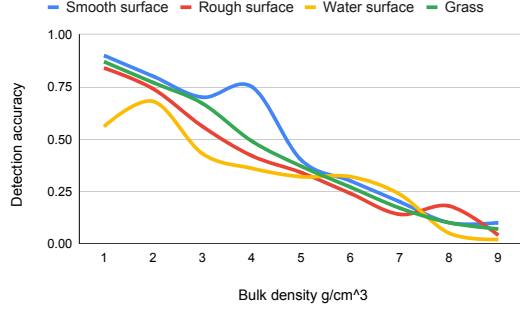
A. Experiment Settings

1) *GprMax Simulator*: The GprMax simulator used in this study solves two dimensional (2D) Maxwell equations using the Finite-Difference Time-Domain (FDTD) method [32]. The simulation in this work considers small diameter objects of various shapes namely sphere, T-shape, L-shape and X-shape as shown in Figure 2. GprMax characterizes the impact of common underground object shapes, which formulates the GPR ScanCloud based on the reflected A-Scan signals.

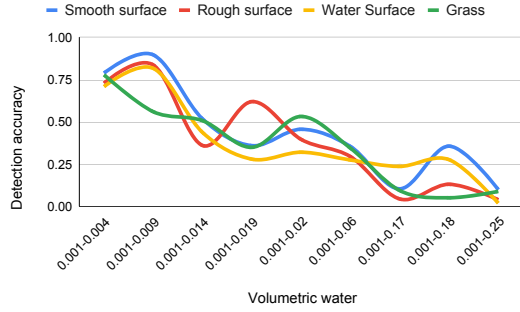
2) *DDPG*: Simulations are conducted on a core i7 computer with four cores, 2.2 GHz Intel Xeon CPU, and 16GB RAM. The training process is run with Python 3.6 and tensorflow 1.10.0. The size of the replay buffer is 5×10^4 , and the sample mini batch is $B = 32$. During the training process, the GPR agent interacts with the environment and receives tuples of state, action, reward and next state. A total of 5×10^4 such tuples are stored in the replay buffer as experiences which is then sampled and used during the learning. The agent starts by exploring the environment to build knowledge about transitions and action rewards.

B. Performance Results

1) *Object Reconstruction*: Based on the results shown in Figure 4, our method outperforms other methods on 3D object



(a)



(b)

Fig. 5. ScanCloud performance of varying bulk density and volumetric water

reconstruction. Depending on the view, other intuitive object feature like shape, depth, and diameter can be observed. Our method provides a much better visual intuition of special shapes like spheres, boxes, plates, and other complex underground infrastructure containing multiple objects.

Using DDPG (Algorithm 1) the GPR agent learns to take continuous sensing actions around the geographical area to acquire the global view of the underground infrastructure.

2) *Likelihood of Successful Object Detection vs. Heterogeneous Medium*: In this paper, we evaluate the performance of the proposed method under different levels of clutter noise caused by clutter from heterogeneous soil. We considered variations of bulk density and volumetric water of the medium. To make the clutter noise levels more distinct, different types of soil surfaces including smooth, rough, water, and grass surfaces, are added to the fractal-box (a box that houses Peplinski heterogeneous soil).

Figure 5(a) shows detection accuracy vs. bulk density and Figure 5(b) detection accuracy vs. volumetric water, indicating that a high object detection rate and reconstruction rate is obtained for setups with low bulk density and volumetric water. This is because higher the bulk density and volumetric water makes higher dielectric constant, resulting in higher GPR signal attenuation as it propagates through the medium.

3) *Classification Accuracy vs. ScanCloud Points*: As GPR data interpretation is affected by the density of the ScanCloud points, the impact of ScanCloud dimension on the classification accuracy of the classifier was also evaluated with various Gaussian grid sizes ($m = 11, m = 8, m = 5, m = 4$,

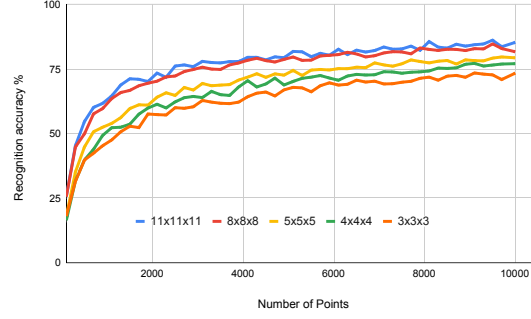


Fig. 6. ScanCloud accuracy based on $11 \times 11 \times 11$ grid, $8 \times 8 \times 8$ grid, $5 \times 5 \times 5$ grid, $4 \times 4 \times 4$ grid and $3 \times 3 \times 3$ grid

TABLE II
CLASSIFICATION ACCURACY OF REWARD FUNCTION CHOICES

Model	Dry sand	Wet sand	Pavement	Heterogeneous soil
ScanCloud- r^{amp}	85.1	83.5	84.2	82.8
ScanCloud- r^{sc}	90.1	87.9	89.1	83.0
ScanCloud- r^{roi}	90.7	88.4	89.5	87.4
ScanCloud	91.3	89.1	90.6	88.0

and $m = 3$). It is observed that the accuracy increases with the increase of the grid size (m) and the number of points representing the ScanCloud, as shown in Figure 6. However, careful considerations must be taken when selecting a large grid size which might introduce high computational overhead.

C. Convergence Analysis

1) *Comparison Between Reward Choices*: In order to evaluate the proposed reward function, we study the impact of different reward choices on performance convergence. In the evaluation, four types of rewards were considered, that is, the reward only from, amplitude analysis, RoI detection, object classification, and their combination. The rewards were computed from four types of mediums namely; dry sand, wet sand, pavement and heterogeneous soil. Table II displays the classification accuracy scores of the various reward choices.

2) *Time Steps to Reach Convergence*: The time needed for the proposed method to reach converged performance was also evaluated. The time steps to reach convergence for different ScanCloud architecture implementations adopting different grid sizes (m) are displayed in Table III. From the table the proposed method configured with a grid size of ($8 \times 8 \times 8$)

TABLE III
COMPARISON OF TIME STEPS CONVERGENCE WITH DIFFERENT GRID SIZES

Grid Size	Dry Sand	Wet Sand	Pavement	Heterogeneous Soil
$3 \times 3 \times 3$	3.25×10^3	4.32×10^3	4.12×10^3	4.25×10^3
$4 \times 4 \times 4$	3.02×10^3	4.25×10^3	4.05×10^3	4.21×10^3
$5 \times 5 \times 5$	3.11×10^3	4.16×10^3	3.84×10^3	4.03×10^3
$8 \times 8 \times 8$	2.61×10^3	3.64×10^3	3.21×10^3	3.56×10^3
$11 \times 11 \times 11$	2.95×10^3	3.98×10^3	3.56×10^3	3.95×10^3

outperforms the one with grid size ($11 \times 11 \times 11$), because larger grid size introduces higher computational overhead, despite yielding higher classification accuracy.

VI. CONCLUSION

In this paper, a GPR-based subsurface object detection based on deep deterministic policy gradient (DDPG) was proposed. A novel reward function was developed such that the GPR agent is rewarded from amplitude analysis, region of interest identification, and object classification. With the proposed reward function a DDPG-based model was developed to enable the GPR to learn to take optimal actions that maximize the long-term discounted reward, hence detecting and identifying subsurface objects from its experiences of interacting with the environment. Simulation results show the proposed ScanCloud-based GPR has superior performance in terms of object detection accuracy, object classification accuracy, and convergence. As part of the future work, the proposed methods will be tested in real-world environment.

REFERENCES

- [1] M. M. Omwenga, D. Wu, Y. Liang, L. Yang, D. Huston, and T. Xia, "Autonomous cognitive gpr based on edge computing and reinforcement learning," in *2019 IEEE International Conference on Industrial Internet (ICII)*, 2019, pp. 348–354.
- [2] S. Li, Y. Wu, X. Cui, H. Dong, F. Fang, and S. Russell, "Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 4213–4220, Jul. 2019.
- [3] C. Qiu, Y. Hu, Y. Chen, and B. Zeng, "Deep deterministic policy gradient (ddpg)-based energy harvesting wireless communications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8577–8588, 2019.
- [4] "Gprmax," <https://www.gprmax.com/about.shtml>, 2020, [Online; accessed 10-November-2020].
- [5] C. Warren, A. Giannopoulos, and I. Giannakis, "gprmax: Open source software to simulate electromagnetic wave propagation for ground penetrating radar," *Computer Physics Communications*, vol. 209, pp. 163 – 170, 2016.
- [6] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019.
- [7] J. Feng, L. Yang, E. Hoxha, S. Sotnikov, D. Sanakov, and J. Xiao, "Gpr-based model reconstruction system for underground utilities using gprnet," *CoRR*, vol. abs/2011.02635, 2020.
- [8] J. Feng, L. Yang, H. Wang, Y. Song, and J. Xiao, "Gpr-based subsurface object detection and reconstruction using random motion and depthnet," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 7035–7041.
- [9] X. Chen, J. Li, S. Huang, H. Cui, P. Liu, and Q. Sun, "An automatic concrete crack-detection method fusing point clouds and images based on improved otsu's algorithm," *Sensors*, vol. 21, no. 5, 2021.
- [10] J. Wolf, S. Discher, L. Masopust, S. Schulz, R. Richter, and J. Döllner, "Combined visual exploration of 2d ground radar and 3d point cloud data for road environments," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 231–236, 2018.
- [11] W. Li, J. Wen, Z. Xiao, and S. Xu, "Application of ground-penetrating radar for detecting internal anomalies in tree trunks with irregular contours," *Sensors*, vol. 18, no. 2, 2018.
- [12] Y. Ge, H. Tang, D. Xia, L. Wang, B. Zhao, J. W. Teaway, H. Chen, and T. Zhou, "Automated measurements of discontinuity geometric properties from a 3d-point cloud based on a modified region growing algorithm," *Engineering Geology*, vol. 242, pp. 44–54, 2018.
- [13] J. Döllner, "Geospatial artificial intelligence: Potentials of machine learning for 3d point clouds and geospatial digital twins," *Journal of Photogrammetry, Remote Sensing and Geoinformation Science* volume, p. 15–24, 2020.
- [14] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 2019, pp. 514–524.
- [15] P. Narksri, E. Takeuchi, Y. Ninomiya, Y. Morales, N. Akai, and N. Kawaguchi, "A slope-robust cascaded ground segmentation in 3d point cloud for autonomous vehicles," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 497–504.
- [16] J. Roland, P. D. Way, C. Firat, T.-N. Doan, and M. Sartipi, "Modeling and predicting vehicle accident occurrence in chattanooga, tennessee," *Accident Analysis & Prevention*, vol. 149, p. 105860, 2021.
- [17] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, Jan 1979.
- [18] Y. Zhang, P. Candra, G. Wang, and T. Xia, "2-D Entropy and Short-Time Fourier Transform to Leverage GPR Data Analysis Efficiency," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 1, pp. 103–111, 2015.
- [19] S. F. Solimene Raffaele, D'Alterio Antonietta, "Entropy-based clutter rejection for intrawall diagnostics," *International Journal of Geophysics*, vol. 2012, 2012.
- [20] J. Lerga, N. Saulig, and V. Mozetič, "Algorithm based on the short-term rényi entropy and if estimation for noisy eeg signals analysis," *Computers in Biology and Medicine*, vol. 80, pp. 1 – 13, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0010482516302888>
- [21] D. J. Cornforth, M. P. Tarvainen, and H. F. Jelinek, "Using renyi entropy to detect early cardiac autonomic neuropathy," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013, pp. 5562–5565.
- [22] J. Zhang, J. Yan, D. Infield, Y. Liu, and F. sang Lien, "Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and gaussian mixture model," *Applied Energy*, vol. 241, pp. 229–244, 2019.
- [23] H. Lin, H. Chen, K. Jin, L. Zeng, and J. Yang, "Ship detection with superpixel-level fisher vector in high-resolution sar images," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 2, pp. 247–251, 2020.
- [24] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [25] Y. Li, J. Zeng, S. Shan, and X. Chen, "Occlusion aware facial expression recognition using cnn with attention mechanism," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2439–2450, 2019.
- [26] M. Ghafari, J. Clark, H. B. Guo, R. Yu, Y. Sun, W. Dang, and H. Qin, "Complementary performances of convolutional and capsule neural networks on classifying microfluidic images of dividing yeast cells," *Plos One*, vol. 6(13), 2021.
- [27] C. E. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [28] A. Namdari and Z. Li, "A Review of Entropy Measures for Uncertainty Quantification of Stochastic Processes," *Advances in Mechanical Engineering*, vol. 11, no. 6, 2019.
- [29] M. M. Omwenga, D. Wu, Y. Liang, L. Yang, D. Huston, and T. Xia, "Cognitive gpr for subsurface object detection based on deep reinforcement learning," *IEEE Internet of Things Journal*, pp. 1–1, 2021.
- [30] S. P. S. Y. M. e. a. R. S. Sutton, D. A. McAllester, "Policy gradient methods for reinforcement learning with function approximation," p. 1057–1063, 1999.
- [31] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 1. Beijing, China: PMLR, 22–24 Jun 2014, pp. 387–395.
- [32] K. S. Kunz and R. J. Luebbers, *The finite difference time domain method for electromagnetics*. CRC press, 1993.