Underground Infrastructure Detection and Localization Using Deep Learning Enabled Radargram Inversion and Vision Based Mapping

Mengjun Wang[1], Da Hu[2], Junjie Chen[3], Shuai Li[1]*

[1] Department of Civil and Environmental Engineering, The University of Tennessee, Knoxville, TN, USA
[2] Department of Civil and Environmental Engineering, Kennesaw State University, Marietta, GA, USA
[3] Department of Real Estate and Construction, The University of Hong Kong, Hong Kong, China

Abstract

Underground pipeline strikes, a pressing problem due to inaccurate subsurface data, are addressed in this paper with a pipeline detection and localization framework. First, abundant radargrams are generated to relieve radargram data shortage by simulating Ground Penetrating Radar (GPR) scans along the urban roadway and enhancing their realism with Generative Adversarial Network (GAN) technique. Second, a deep learning network is designed to directly reconstruct permittivity maps from radargrams for accurate pipeline detection and characterization, instead of detecting pipeline features within the radargram. Third, Simultaneous Localization and Mapping (SLAM) is employed for GPR position estimation, enabling precise georegistration of pipelines. The proposed method attains an R-squared (R^2) value of 0.957 in permittivity map reconstruction and 96.2% precision in pipeline detection. Additionally, it provides satisfactory performance with a deviation of 1.71% in depth and 20.44% in diameter for the detected pipelines. Real-world experiments validate the effectiveness of the proposed framework, highlighting its potential to prevent excavation accidents, reduce project delays, and offer significant benefits to utility companies, contractors, and urban planners.

Keywords: Underground Infrastructures; Detection and Mapping; Radargrams; Deep Learning; Ground Penetrating Radar.

1. Introduction

Utility strikes by excavation have been a persistent problem in the US [1]. The Pipeline and Hazardous Material Safety Administration (PHMSA) reported that there are 640 pipeline incidents, 14 fatalities, and 57 injuries every year on average from 2002 to 2021. These damages have a significant economic impact on the U.S., with an estimated $30 billion in annual societal costs [2]. Incomplete, inaccurate, and missing [3–6] subsurface pipeline information is one of the main reasons causing utility strike accidents during excavations. Therefore, it's crucial to construct a subsurface utility map with precise location and dimension information.

GPR has been utilized extensively in the field of locating underground utilities as a non-destructive technology (NDT) [7]. Pipelines and other cylindrical subsurface items are frequently identified in radargrams as hyperbola characteristics. As a result, by identifying and examining hyperbola signs, pipeline size and position information can be approximated [8,9]. Some handcrafted algorithms have been developed to detect and segment hyperbola features in the radargram to obtain pipeline information. For example, Yang et al. [10] proposed a hyperbola extracting method to analyze the pipeline information using the hyperbolic asymptote. Rohman and Nishimoto [11] presented a pattern-matching technique for pipelines with different shapes, such as triangles, squares, and circles, to find the location of pipelines. However, the underground conditions are generally complex and heterogeneous. Consequently, the aforementioned methods often struggle to produce reliable results in real-world scenarios and adapt to various subsurface variations. With the rapid development of computer vision techniques, many deep learning-based methods have been developed to detect and characterize underground utilities in more complex scenes. For example, Yamaguchi et al. [12] proposed a combination model of a 3D convolutional neural network with Kirchhoff migration to detect hyperbola box-by-box in the GPR radargram. Hou et al. [13] proposed a Mask Scoring R-CNN (MS R-CNN) model to detect, segment, and analyze the hyperbola features. However, two main drawbacks exist in these studies. First, their performance is significantly impacted by the quality of hyperbolic features. In urban areas, underground pipelines often appear clustered, causing radar signal interference. This can lead to inaccurate or fuzzy hyperbolas, substantially hindering these approaches. Second, they mainly focused on detecting the hyperbolic features while paid less attention to the geo-registration of underground pipelines. Given geo-registration's importance, some methods have been developed to register detected pipelines within a spatial referencing system. For example, Li et al. [7] proposed a hybrid method for underground utility detection and localization by fusing GPR and GPS data. However, GPS signals could be significantly attenuated in urban areas, resulting in inaccurate localization.

The overall goal of this study is to introduce an integrated framework for detecting and geo-registering subsurface pipelines by integrating GPR and camera data. First, a deep learning-driven GPR radargram inversion model is designed to reconstruct permittivity maps of road cross-sections, enabling the retrieval of information about road base layer as well as the location and dimensions of buried pipelines. Second, SLAM is employed to localize the GPR, allowing for the geo-registration of identified pipelines. The remainder of this paper is organized as follows. Section 2 reviews related works on subsurface object detection. Section 3 describes the proposed system's research methodology, which includes aboveground map reconstruction and underground pipeline detection. Section 4 provides an overview of the implementation, findings, and field case studies. Finally, Section 5 summarizes the research conclusions and discusses potential future research directions.

2. Literature Review

GPR has been extensively applied in subsurface objects detection fields, and it has been testified as an efficient technique for locating subsurface targets in many civilian applications, such as bridge deck inspection [14–16], void detection under rubbles [17–19] and subsurface utility detection [20–22]. For example, Porsani et al [23] utilized GPR to map existing infrastructures along the construction site to orient the subway tunnel. Metwaly [20] performed a survey along an urban asphalt road to detect all kinds of pipelines using GPR, highlighting the importance of carrying out surveys before excavation activities. Coster et al. [24] improved the GPR-based detection of pipelines and leaks in water distribution networks. All these studies demonstrated the great potential of GPR in detecting and reconstructing subsurface objects.

As computing power has advanced, many researchers have developed algorithms to automate the processing of GPR signals, which can be categorized into hyperbola detection and radargram inversion. Hyperbola detection methods analyze radargram features to determine the size and location of subsurface objects [25–28]. Numerous studies are progressing in tandem with the advancements in hyperbola detection methods. Wang et al. [29] proposed a template-based method to detect rebar apex and fit it in radargrams using partial differential equations. Terrasse et al. [30] detected the position of gas pipelines with GPR acquisitions using a dictionary of theoretical pipe signatures. Sagnard et al. [28] developed an algorithm that is not restrictive to the hyperbola pattern base on the template-matching and LS hyperbola fitting technique. Aided by computer vision techniques, many Deep Neural Network (DNN)-based hyperbola detection methods have also been developed. applied a column connection clustering (C3) algorithm to separate the regions of interest (ROI), and then identify hyperbola signatures from these ROIs. Yuan et al. [31] introduced a drop-flow algorithm that mimics the movement of raindrops to detect and decompose hyperbola signals from underground pipelines. Liu et al. [32] proposed an automatic detection and localization method via deep learning and migration, in which a Single Shot Multibox Detector (SSD) is utilized to extract regions of hyperbola in radargrams. Permittivity reconstruction-based methods focus on reconstructing the permittivity map of underground scenes to detect the targets. Some researchers have proposed AI-based methods for permittivity reconstruction to detect underground objects. For example, Qin et al. [33] presented a probabilistic inversion model based on Markov chain

Monte Carlo (MCMC) to estimate the grouting layer thickness with its relative permittivity and electric conductivity. Liu et al. [34] proposed an end-to-end DNN framework to invert the dielectric properties of tunnel linings. Ji et al. [35] proposed a Permittivity Inversion Network (PINet) to utilize the time compression operation so that the position, rough shape, and permittivity of targets can be reconstructed.

Based on the development of subsurface object detection methods, many recent studies have also developed methods to obtain the 3D position of subsurface pipelines. For instance, a mobile robot is combined with a GPR [36] to perform underground utility mapping. The robot localizes itself using GPS so that the buried pipelines could be further localized, while this process could be hugely affected by the GPS signal amplitude. To address this issue, other researchers proposed methods that are less restricted by environments. Li et al. [37] proposed a pipeline mapping method based on the J-linkage method and maximum likelihood estimation (MLE) and successfully reconstructed the 3D map of buried pipelines with great results. However, their methods are limited to detecting pipelines buried in a homogeneous medium with a priorly known radio wave propagation velocity. Feng et al. [38] presented a 3D imaging migration system that synchronizes the GPR pose with GPR scans to reconstruct and visualize the subsurface pipelines. They considered the outputs of visual inertial fusion (VIF) as the pose of the GPR device while not considering the calibration between GPR and the camera coordinate system. Addressing these limitations, our study harmonizes the coordinate systems of GPR and the camera, subsequently integrating the reconstructed permittivity and aboveground maps. This process facilitates the development of a comprehensive pipeline localization model, trained on diverse road cross-section data.

3. Methodology

Figure 1 illustrates the architecture of the proposed framework for underground pipeline detection and geo-registration. The framework consists of two modules: the Subsurface Pipeline Detection Module (SPDM) and the Localization and Aboveground Reconstruction Module (LARM). Addressing the shortage of labeled real GPR data in the SPDM, a substantial number of GPR radargrams are generated by simulating GPR scans on synthetic urban roadways. These simulations come with ground-truth permittivity maps, a topic explored in our previous research [39]. Additionally, GAN technique is employed to reduce discrepancies between simulated and real radargrams, thereby enhancing the simulated radargrams' realism. The paired radargrams and permittivity maps serve as the training dataset for the inversion network. For the LARM, SLAM is employed to reconstruct the aboveground map and determine the GPR position. The subsurface map, predicted by the trained GPR inversion model, can then integrate with the reconstructed aboveground map based on their relative positions. Consequently, the detected pipelines can be geo-registered. Each module is described in further detail in the subsequent sections.

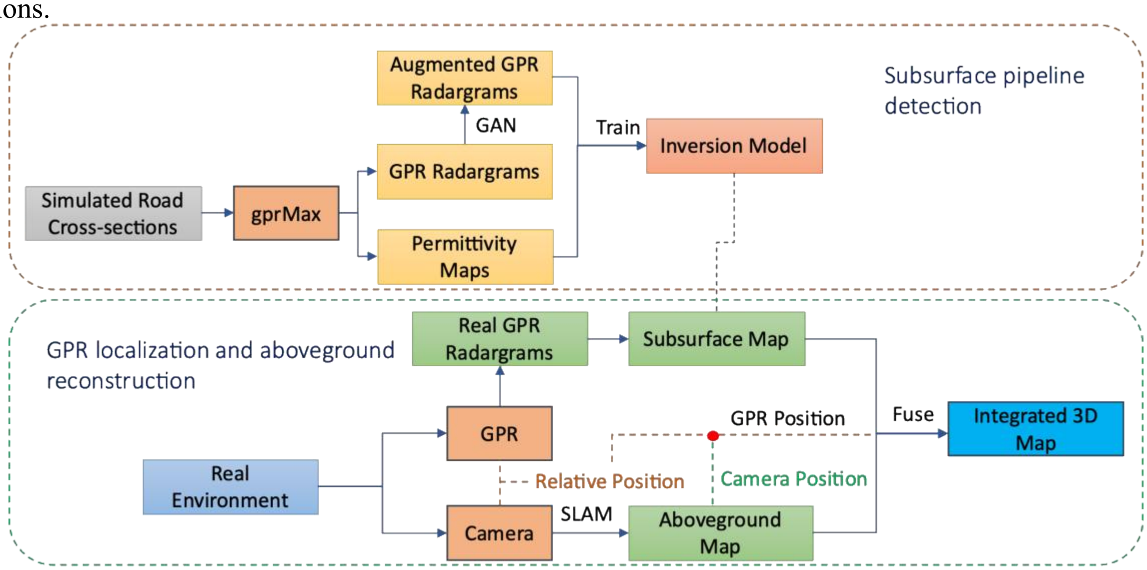

Figure 1: System architecture overview.

3.1 Pipeline Detection

Figure 2 presents an overview of the proposed subsurface pipeline detection framework, which consists of three stages. In the first stage, simulated road cross-sections containing a pipeline system are created to generate a large number of synthetic radargrams, accompanied by ground-truth permittivity labels. In the second stage, GAN is employed to augment simulated radargrams with realistic signal features. Finally, the inversion network is trained to directly invert GPR radargrams to produce permittivity maps using the augmented radargrams and associated permittivity labels. Each stage is detailed below.

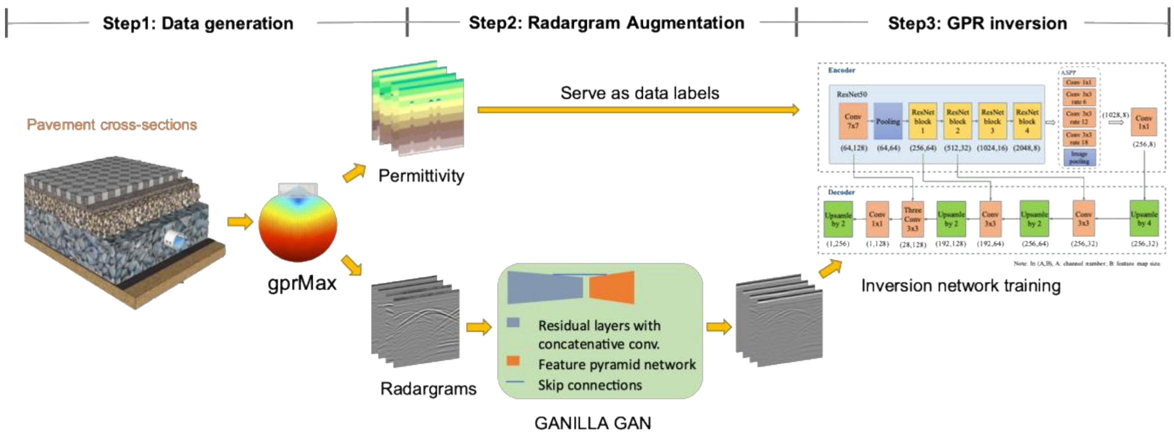

Figure 2: Subsurface mapping methodology overview.

3.1.1 Data Generation

Obtaining cross-section permittivity maps and GPR scans of underground pipelines in natural settings is challenging due to the unpredictability of permittivity values within soil layers. Factors such as density and humidity can significantly impact soil permittivity, making the creation of a precise permittivity map difficult in practice. To address this, our research generates abundant synthetic GPR radargrams alongside their corresponding permittivity maps, which are then used for training the inversion model. The gprMax [40] simulator is utilized to perform the synthetic radargram generation.

To simulate a more realistic underground cross-section with pipelines, this study investigated multiple design standards related to urban road design and subsurface pipe regulations. The common flexible urban pavement usually contains four layers: surface layer, base course, subbase course, and subgrade layer. Table 1 presents each layer's thickness of urban pavement.

Table 1: Variations in thickness across different pavement layers [41].

Road layer	Thickness (cm)
Surface	10-18
Base course	13-30
Subbase course	13-30
Subgrade layer	The rest

Urbanization has led to an increased demand for subsurface infrastructure installations, including gas pipelines, sewer lines, water pipes, electrical cables, and optical fibers. Varied pipeline types may have different criteria for material, size, depth, and installation. Furthermore, due to their various functionalities, many pipeline types require specified interval spacings. These factors were considered when generating simulated urban road cross-sections. Table 2 presents the settings for various buried pipelines.

Table 2: Spatial setting requirements of underground pipelines [42].

Utility lines	Inside radius (cm)	Outside radius (cm)	Depth (cm)	Remark
Gas lines	(1,14)	(1,15)	61 below	30.5 cm vertical clearance from water and sewer pipelines
Electrical wire	N/A	(2,5)	61 below	30.5 cm vertical clearance from water and sewer pipelines
Water lines	(1,28)	(3,30)	91.4 below	45.7 cm vertical clearance from sewage lines
Sewage lines	(3,19)	(5,20)	61-91.4 below	45.7 cm vertical clearance from water lines
Optical cable	N/A	(1,2)	30.5 below	30.5 cm vertical clearance from water and sewer pipelines

Road layers and pipelines can be made of various materials, and the same type of pipeline might be constructed from different materials. Therefore, when creating synthetic permittivity maps, considering a broad spectrum of materials is essential. Table 3 shows the permittivity value range for different materials.

Table 3: Permittivity values of different materials.

Object	Permittivity
PVC	3-5 [43]
Concrete	5-10
Clay	5-40
Asphalt	4.5[44]
Flexible road base	8-12 [44]
Gravel subgrade	8-15 [44]
Iron	1.4-1.6 [45]

Based on the above physical setting parameters range, this study constructed a large number of simulated cross-sections with buried pipelines. The size, depth, spatial setting, permittivity of the pipelines, and thickness of each road layer in these cross-sections are randomly set within the related value range in Tables 1 , 2 and 3. This study set the relative permittivity increases along with the depth increases considering the density and water content variation in the subgrade layer. Figure 3 shows three examples of the constructed urban road cross-sections with subsurface pipelines where different colors represent different relative permittivity values.

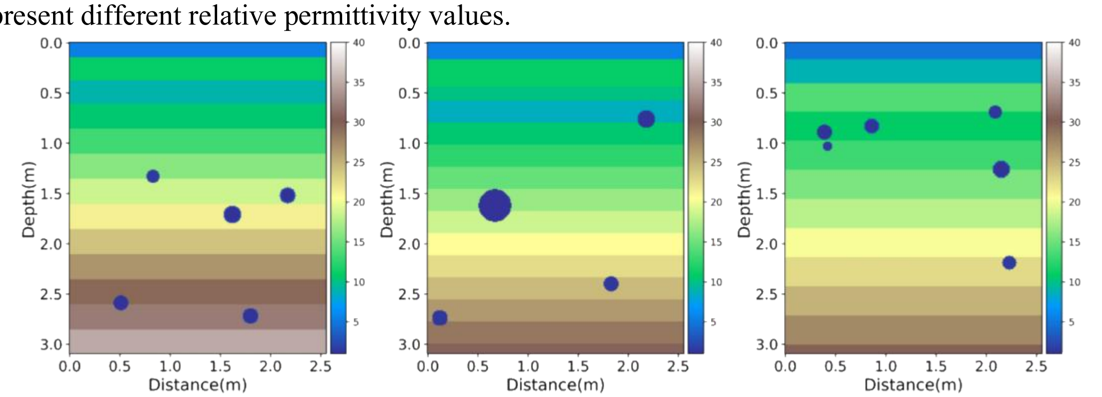

Figure 3: Permittivity map of pavement cross-section with subsurface pipelines.

The 350 MHz GPR antenna can penetrate up to 10 meters while still delivering enough resolution, which is sufficient for typical subsurface pipeline identification in urban areas. Due to this, 350 MHz was chosen as the gprMax frequency for this study. There are 256 traces in each simulated permittivity map with a length of 2.56m. The depth of the synthetic cross-section is set to 3.1m.

The simulated cross-sections with pipelines are fed into the gprMax simulator to simulate corresponding radargrams. Each permittivity map in the spatial domain needs to be transformed into the time domain in order to construct the necessary labels for these radargrams. Equation 1 elaborates on the conversion process.

$$t = \frac{2d}{C}\sqrt{\varepsilon} \tag{1}$$

Where t denotes the roundtrip time in ns, d means depth in meters, e is the permittivity value, and C is light speed which is $3 \times 10^8 \mathrm{m/s}$.

The depth is incremented by 0.01 m in the gprMax. The total iteration number is 3000 and the time window is set to 2.35865e-11s, in this way the roundtrip time equals 70.76 ns. Moreover, time-zero correction is utilized to obtain an accurate signal propagation time between transmitting and receiving antennas, which helps correct the detected depth of objects. In this study, the time-zero corrected iteration number is 2600 which equals 61.32 ns of travel time.

Inevitably, the GPR signal weakens as penetration depth increases. To counteract this attenuation, we incorporate an exponential gain during signal processing, thereby amplifying the diminished reflected signals from deeper substrates. This enhancement of subsurface object signals amplifies their visibility, thereby bolstering the efficacy of the predictive model. Figure 4 presents two cases of the generated radargram and related permittivity map. These paired images are further used to train the inversion network.

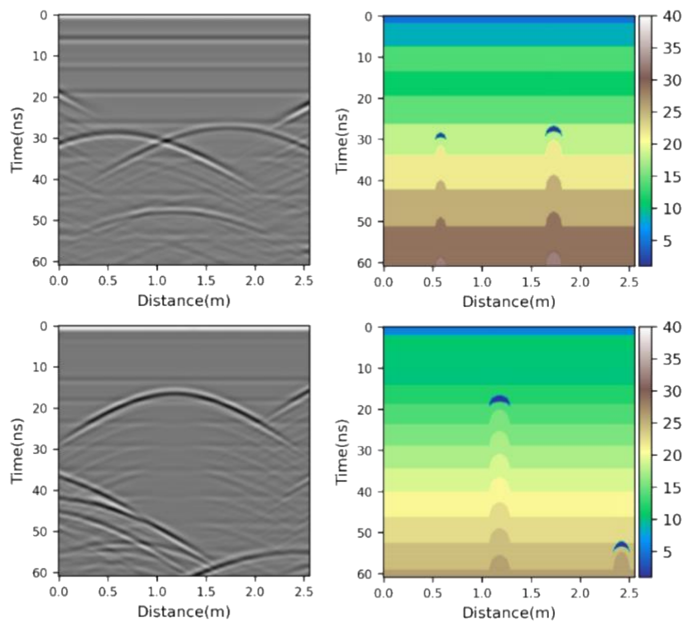

Figure 4: Examples of synthetic radargram and its corresponding permittivity map in time domain.

3.1.2 Data Augmentation

Although simulation could generate paired datasets for model training, a noticeable disparity exists between synthetic and real GPR scans. Training the reconstruction model only relying on the clear, low-noise, synthetic radargrams could hinder the model's practical applications. Thus, GANILLA GAN [46] is applied to enhance the realism of the generated GPR scans.

Figure 5 illustrates the architecture of the generator network and the generator-discriminator model. This network comprises two pairs of generator-discriminator models, wherein the generators and discriminators operate independently. The primary objective is to refine synthetic radargrams into realistic ones. In the composite model 1, generator G_{B2A} generates synthetic radargrams from real radargram input, while discriminator D_A aims to distinguish between real and fake radargrams. In composite model 2, generator G_{B2A} refines synthetic radargrams to enhance realism, and discriminator D_B contributes to increasing their realism. Upon applying these two composite models, synthetic radargrams are refined into realistic radargrams with high fidelity.

The discriminator network utilized in the GANILLA GAN is a 70×70 PatchGAN [47]. This network consists of four consecutive blocks, each containing one convolutional layer and one instance-normalization layer. The first block contains 64 filters, with the number doubling in each subsequent block. Additionally, there is a convolution layer with one filter followed by a sigmoid activation layer. The discriminator enables differentiation between real and fake radargrams generated by the network. The generator contains both a downsampling and an upsampling stage. The downsampling stage starts with a 7×7 convolutional layer, followed by an instance norm layer, ReLU, and max-pooling layer. Subsequently, four layers follow, each containing two residual blocks. Each residual block starts with a 3×3 convolutional layer followed by an instance norm layer. Then, another set of 3×3 convolutional layers and normalization layers are added. The output from these two sets of layers is concatenated with the residual block's input. Finally, the concatenated tensor is fed into a 3×3 convolutional layer. The upsampling stage contains one 1×1 convolutional layer followed by four consecutive upsample and summation layers. Moreover, low-level features extracted by the downsampling stage are concatenated to the summation layers before the upsampling through a long skip connection. Finally, two consecutive convolutional layers output the augmented radargram.

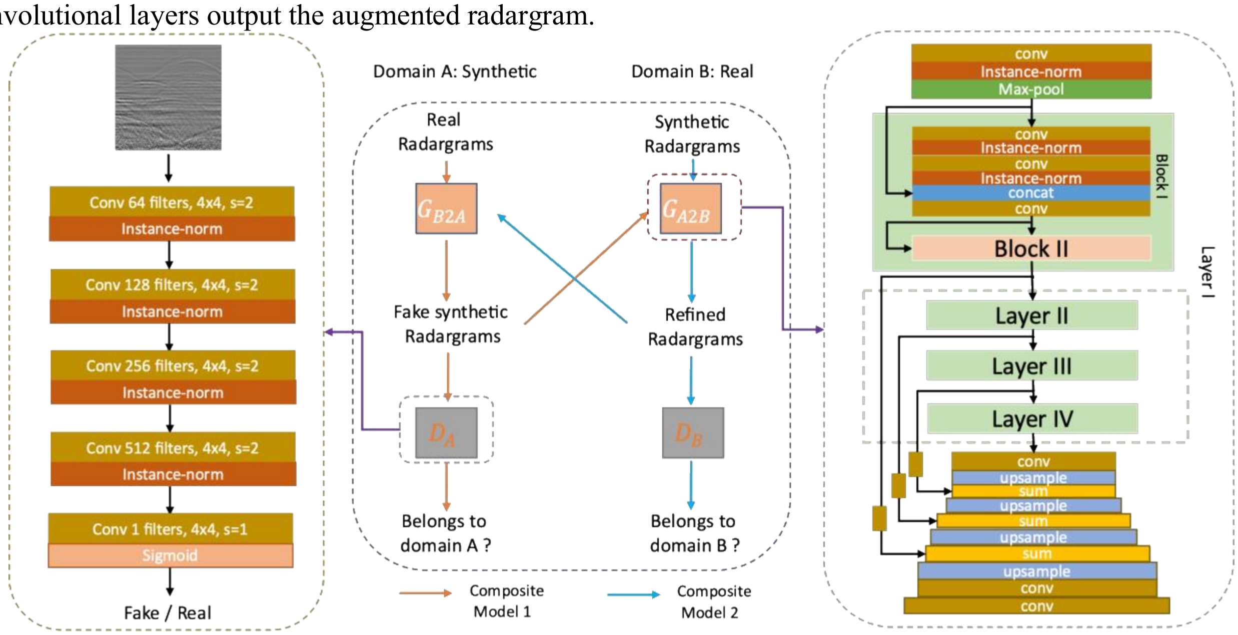

Figure 5: Structure of GANILLA GAN network applied in GPR radargram augmentation.

The employed GAN model incorporates two types of loss functions. Adversarial losses ensure that radargrams generated by the generator closely resemble the distribution of real radargrams. Cycle consistency losses serve to prevent generators (G) and discriminators (D) from producing conflicting outcomes; that is, images generated by the two generators can be transformed back to their original state. In the context of our research, minimizing the adversarial loss associated with transforming radargrams from the synthetic domain to the real domain is of paramount importance. The optimal generator can be identified with loss in Equation 2 through this iterative training process.

$$G^*_{A2B} = \arg\min_{G_{A2B}}\max_{D_B}\mathcal{L}_{GAN}(G_{A2B}, D_B, A, B) \tag{2}$$

Where $\mathcal{L}_{GAN}(G_{A2B}, D_B, A, B)$ is the objective function of mapping synthetic radargram to real radargram, which equals to $\mathbb{E}_{b\sim p_{data}(b)}[log D_B(b)] + \mathbb{E}_{a\sim p_{data}(a)}\left[\log\left(1 - D_B\left(G_{A2B}(a)\right)\right)\right]$

3.1.3 GPR Inversion Network

This research proposes a deep learning-based radargram inversion method for directly reconstructing the underground permittivity maps based on radargrams. The model structure is displayed in Figure 6. The presented network which is adapted from DeepLabv3+ architecture contains an encoder and a decoder. Integrating Atrous Spatial Pyramid Pooling (ASPP) with this encoder-decoder structure enables multi-scale contextual features in the GPR radargram to be detected. The augmented radargram data is passed to the encoder which contains a ResNet50-based backbone and an ASPP module. The spatial size of the original radargram data is reduced to 8×8 from 256×256 while the dimensions turn to 256 in the encoder part. These processed feature maps provide different scale object information for the decoder. At the decoder, each of the extracted low-level features is combined with the high-level features extracted by the decoder. The predicted permittivity maps with 256×256 spatial sizes are generated after a series of upsampling. The structure of the encoder and decoder are elaborated below.

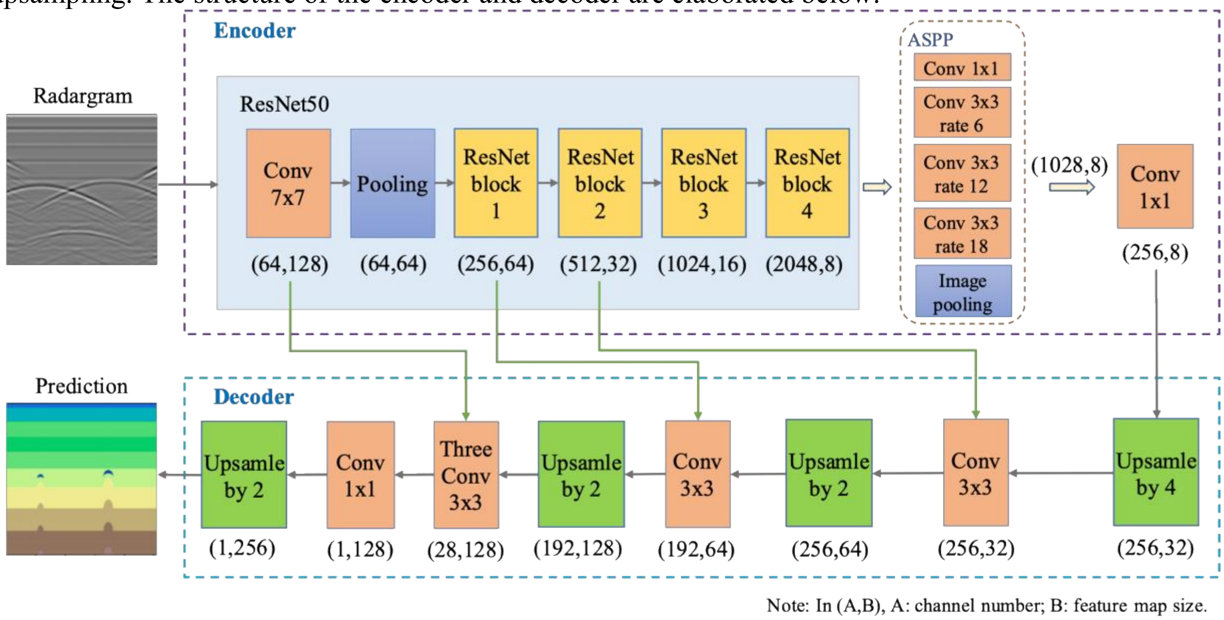

Figure 6: GPR inversion network architecture.

Encoder: The adapted ResNet50 backbone contains one 7×7 convolutional layer, one pooling layer, and four ResNet residual block groups with block numbers of 1, 2, 3, and 4 respectively. In each stage, the spatial size of feature maps would be reduced to half and the channel number would double. At the same time, the convolutional layer and the first two ResNet block groups extract low-level features and pass them into the decoder. The results of the whole backbone with a spatial size of 8×8 are fed into ASPP followed. There are five layers in the ASPP module, one 1×1 convolutional layer, three 3×3 convolutional layers, and one average pooling layer. The sampling rates of these four convolutional layers are 1, 6, 12, and 18 respectively. The results of these five layers are concatenated and output a feature map with 1028 channels. This outcome is then passed into a 1×1 convolutional layer.

Decoder: The decoder part is mainly used to recover the spatial size of feature maps and combine the low-level features with the high-level features. There are four upsample layers interpolated in this section with different scale factors. Moreover, there are three 3×3 convolutional layers utilized to extract high-level features integrated with low-level features. The outputs of these convolutional layers with 28 channels are then passed into one 1×1 convolutional layer. Finally, the predicted permittivity maps with 1×256×256 are generated after upsampling by 2. For better representation, we visualized all the predicted gray-scale permittivity maps to colormap using the matplotlib library of Python.

3.2 GPR Localization and Aboveground Reconstruction

To map the subsurface pipelines in the 3D environment, an aboveground autonomous localizing and mapping method is integrated into this mapping system. The above-ground 3D reconstruction map is combined with the subsurface pipelines map and adjusted for their relative positions. In this way, an integrated 3D pipeline library is generated.

3.2.1 Aboveground SLAM

For the SLAM part, the RTAB-Map SLAM [48] method is adopted in this research, which is a graph-based SLAM technique. The structure of the map consists of nodes and links. Odometry nodes publish odometry information to estimate robot poses. The short-time memory (STM) module is used to create nodes to memorize the odometry and RGB-D images and calculate other information. To limit the working memory (WM) size and reduce the time to update the graph, a weighting mechanism is used to determine which nodes in WM are transferred to long-term memory (LTM). Nodes in the LTM can be brought back to WM when a loop closure is detected. Links are used to store transformation information between two nodes. The neighbor and loop closure links are used as constraints for graph optimization and odometry drift reduction. The Bag of Words approach is used for loop closure detection. The visual features extracted from local feature descriptors such as Oriented FAST and rotated BRIEF (ORB) are quantized to a vocabulary for fast comparison. The outputs of the SLAM are the vehicle pose and the 3D reconstructed map.

3.2.2 GPR and Camera Coordinate System Calibration

Figure 7 shows the sensing suite design and calibration settings. A GPR and a binocular camera are mounted on a tricycle together, where the camera is used for obtaining the position of the GPR and reconstructing the aboveground map.

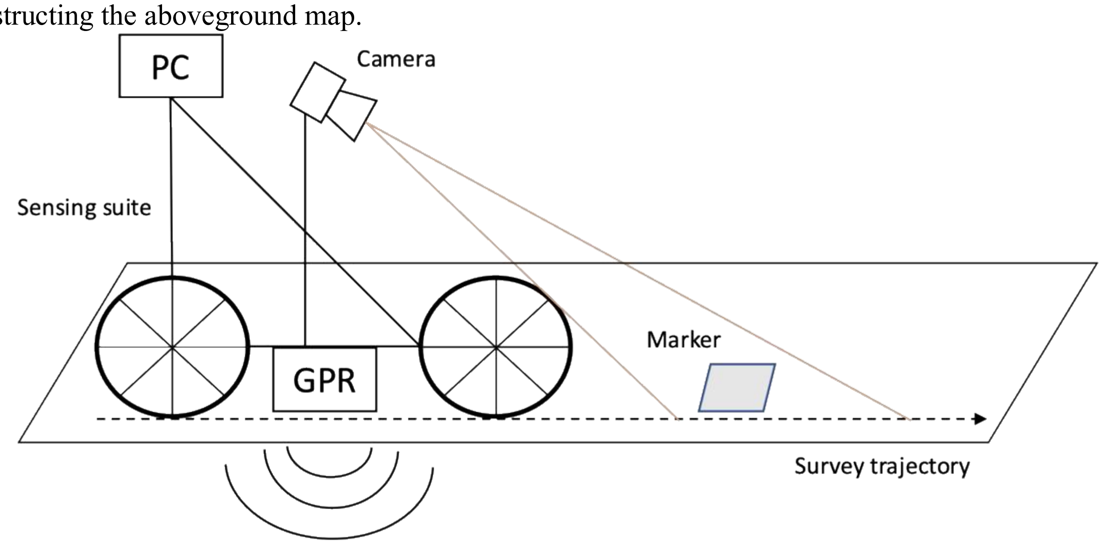

Figure 7: Sensing suite design and calibration setting.

To detect the relative position between GPR and camera origin, a checkerboard-based calibration procedure is performed before collecting data in real sites. As shown in Figure 8, in the experimental setup, a 4×4 Aruco checkerboard is put up on the ground, which is used to calculate the relative position of GPR origin and camera origin. Firstly, the center of the GPR is considered as its origin so that its coordinate in the checkerboard coordinate system is easy to measure. Secondly, a marker detection algorithm transforming the camera coordinate system to the marker coordinate system is applied to the image data collected by the camera. Equations (3)-(4) illustrate the transformation principle.

$$(\mathbf{T}_{\mathrm{cm}})^{-1}\mathbf{O} = \mathbf{M} \tag{3}$$

$$\mathbf{T}_{\mathrm{cm}} = \begin{bmatrix} \mathrm{r}_{11} & \mathrm{r}_{12} & \mathrm{r}_{13} & \mathrm{t}_{\mathrm{x}} \\ \mathrm{r}_{21} & \mathrm{r}_{22} & \mathrm{r}_{23} & \mathrm{t}_{\mathrm{y}} \\ \mathrm{r}_{31} & \mathrm{r}_{32} & \mathrm{r}_{33} & \mathrm{t}_{\mathrm{z}} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{4}$$

Where **O** is the camera coordinate system. **M** is the marker coordinate system which is coincidental with the world coordinate system. $\mathbf{T}_{\mathrm{cm}}$ is a transformation matrix from the camera coordinate system to the marker coordinate system. **R** is the rotation matrix while the **t** is the translation matrix.

This approach allows for the determination of camera poses and their origin coordinates in the marker coordinate system. By aligning the camera and GPR to the same coordinate system, their relative positions can be computed. This, in turn, aids in pinpointing the GPR location on the reconstructed aboveground map, enabling the geo-registration of detected subsurface pipelines.

4. Experiment and Results

4.1 GPR Radargram Augmentation with GAN

4.1.1 Implementation

The GAN network was trained on a Linux workstation equipped with one NVIDIA RTX A5000 GPU using PyTorch. The initial learning rate for the Adam optimizer is 0.00002. After 50 epochs, the learning rate starts to linearly decay to zero. The real GPR data are collected by GSSI 350 MHz GPR. To provide noise features for the training, 705 real radargram images which all contain 256 traces were used. In the meantime, 705 synthetic radargrams are randomly selected from 20867 images. The batch size of the training is set to 1. This training stops after 100 epochs.

4.1.2 Evaluation Metrics

To quantitatively evaluate the performance of the data augmentation process, three matrics including Frechet Inception Distance (FID), Structural Similarity Index Method (SSIM), and Mean Square Error (MSE) are compared between different sets of radargrams.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_x\sigma_y + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{5}$$

Where μ_x and μ_y are the local, σ_x and σ_y are the standard deviations for the radargram input and augmented radargram respectively. C_1 and C_2 are two constants.

The MSE measures the average of the square of the errors between two images.

$$MSE = \frac{1}{MN}\sum_{n=0}^{M}\sum_{m=1}^{N}[\hat{g}(n,m) - g(n,m)]^2 \tag{6}$$

Where $\hat{g}(n,m)$ and $g(n,m)$ denotes two images respectively.

FID compares the distribution of generated radargrams with the distribution of the real radargram set.

$$FID(K,R) = \|\mu_K - \mu_R\|_2^2 + Tr(\Sigma_K + \Sigma_R - 2(\Sigma_K\Sigma_R)^{\frac{1}{2}}) \tag{7}$$

Where K and R denote a different set of images, μ_K and μ_R are the mean feature vectors of K and R respectively, Σ_K and Σ_R are their corresponding covariance matrices. Note $\|\cdot\|_2^2$ is the Euclidean norm operator and Tr(·) is the trace operator here.

4.1.3 Augmented Results

Some examples of augmented radargrams using GAN are shown in Figure 8. As the training dataset contains real data from [49] and self-collected data on real site, the GAN model learned two different features, which increased its generalizability.

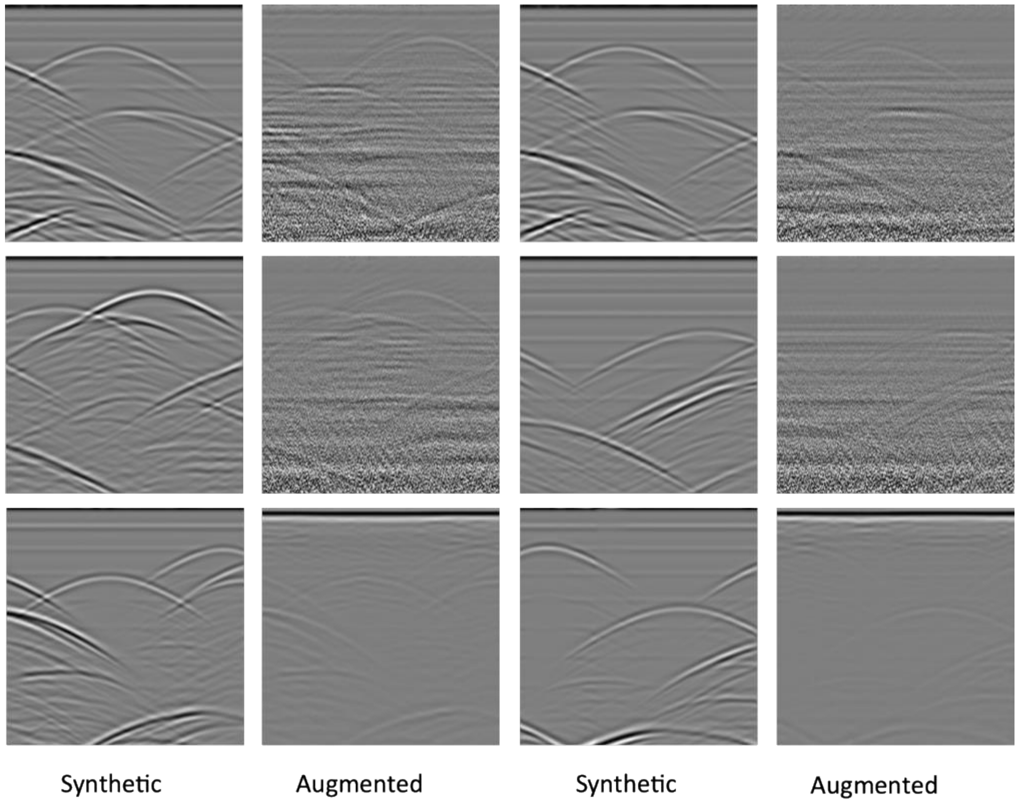

Figure 8: Synthetic and augmented GPR radargram.

Table 4 presented below demonstrates the performance of the proposed GAN model in enhancing synthetic radargrams. Three evaluation metrics are compared across different combinations of radargram sets. The Real set comprises 705 actual radargram images, while the Fake set consists of 705 randomly chosen synthetic radargrams. Corresponding augmented radargrams for these synthetic images are included in the Augmented set.

Table 4: Data augmentation performance.

Matrix	**Real and Fake**	**Real and Augmented**	**Fake and Augmented**
FID	365.098	65.952	352.540
SSIM	0.218	0.266	0.463
MSE	85.522	81.886	71.289

The FID scores represent the degree of similarity between different radargram distributions, where lower values imply greater resemblance. Among the sets, the Real and Augmented image set exhibited the lowest FID score (65.952), signifying its superior approximation to the real images compared to the other sets. The SSIM index quantifies the perceptual similarity between two images, with values ranging from -1 to 1. Higher SSIM values denote enhanced structural and visual similarity. The Fake and Augmented set recorded the highest SSIM value (0.463), which is expected given that images in these sets correspond to each other. Furthermore, compared to the Real and Fake set, the Real and Enhanced set showed a higher SSIM index, indicating that the augmented radargrams had more realistic qualities. The MSE metric calculates the average squared differences between the pixel intensities of the corresponding images, with lower values suggesting improved agreement between the compared images. The Real and Augmented set demonstrated a lower MSE value than the Real and Fake set, indicating that the average pixel intensity difference was smaller in the former set.

The findings indicate that by incorporating signal noise from actual radargrams, we can enhance the authenticity of augmented radargrams. As a result, these augmented radargrams mirror real ones more closely. Utilizing these realistic radargrams to train the inversion network bolsters its performance, providing greater robustness and generalizability in practical scenarios. This ensures the network's reliability and effectiveness.

4.2 GPR inversion

4.2.1 Implementation

The inversion model was developed on a Linux workstation featuring a 100 GB CPU and one NVIDIA RTX A5000 GPU, using the PyTorch liabrary. The dataset contains 20867 pairs of augmented GPR radargrams and corresponding permittivity maps, which contains two random sets, a training set (80%) and a validation set (20%). PyTorch's ReduceLROnPlateau function was utilized, starting with a learning rate of 0.02, subject to a decay factor of 0.5, and processed in batches of 32. To minimize overfitting, we implemented an early stopping mechanism during training. This involved using the training set to refine the model and the validation set to test it. If the validation loss failed to decrease for 10 consecutive epochs, the training ceased and the best-performing model was saved.

4.2.2 Evaluation Metrics

To quantitatively evaluate the performance of the trained inversion network, there are four evaluation matrices, i.e., R squared (R^2), SSIM, Mean Absolute Error (MAE), and confusion matrix were utilized in this study. R^2 which is also known as the coefficient of determination, measures the distance between the reconstructed results and its ground truth. It is detailed in Equation 8, where x_i means the predicted permittivity value, y_i represents the ground truth permittivity value, $\bar{y}$ is the mean of true values.

$$R^2 = \frac{SSR}{SST} = \frac{\sum(x_i - \bar{y})^2}{\sum(y_i - \bar{y})^2} \tag{8}$$

The SSIM measures the similarity between the predicted and true permittivity map [50]. In Equation (9), x is the predicted permittivity map while y is the original permittivity map.

$$SSIM(y, x|\omega) = \frac{(2\overline{\omega}_y\overline{\omega}_x + C_1)(2\sigma_{\omega_y\omega_x} + C_2)}{(\overline{\omega}_y^2 + \overline{\omega}_x^2 + C_1)(\sigma_{\omega_y}^2 + \sigma_{\omega_x}^2 + C_2)} \tag{9}$$

Where ω_y means a sliding window in the same original permittivity map, $\overline{\omega}_y$ is the average of ω_y, $\sigma_{\omega_y}^2$ is the variance of ω_y, $\sigma_{\omega_y\omega_x}$ means the covariance of ω_y and ω_x. C_1 and C_2 are two constants. ω_x, $\overline{\omega}_x$ andσ_{ω_x} mean the same in x .

The MAE represents the average vertical distance between the permittivity value of each pixel in the predicted and original permittivity map. The calculation is shown in Equation (10).

$$MAE = \frac{\sum_{i=1}^{n} abs(x_i - y_i)}{n} \tag{10}$$

Equations 11-13 defined three metrics to further measure the performance of the model. Recall means the ratio of correct positive detections to the total positive examples, precision denotes the ratio of correct positive detections to the total predicted positives, the accuracy is the ratio of correctly detected examples to the total examples.

$$Recall = \frac{TP}{(TP+FN)} \tag{11}$$

$$Precision = \frac{TP}{(TP+FP)} \tag{12}$$

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{13}$$

Where TP represents a true positive which is the number of correctly detected pipelines, FP means a false positive which is the number of incorrectly detected pipelines, TN is a true negative which is the number of detected negative pipelines, FN represents a false negative which is the number of fails detected negative pipelines.
Equation 14-15 shows the definition of predicted error and deviation, where ground truth means the original value, prediction means the predicted value.

$$error \;= abs(ground\ truth - prediction) \tag{14}$$

$$deviation = \frac{abs(ground\ truth - prediction)}{ground\ truth} \times 100\% \tag{15}$$

4.2.3 Inversion Results

This section detailed the results of GPR inversion. Figure 9 shows the variation of three evaluation matrixes along with the training epochs growing on both synthetic radargrams and augmented radargrams. It is easy to see the trend that the SSIM and R^2 grows while the MAE decreases as the epochs increase. The plots indicate that the inversion model quickly converges and becomes stable at around 70 epochs.

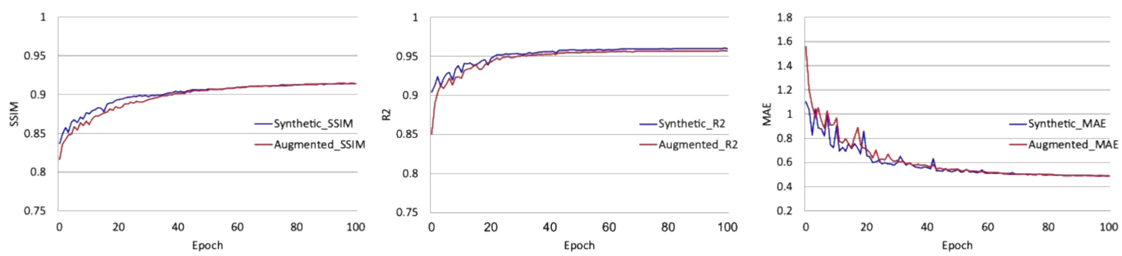

Figure 9: Variance of SSIM, R^2, MAE during the training on validation set.

Table 5 below shows the model performance on synthetic and augmented radargrams. Note that synthetic and augmented radargrams share the same permittivity map labels.

Table 5: Model performance on both synthetic and augmented data after 100 epochs training

Metric	SSIM	R^2	MAE
Synthetic	0.9153	0.96096	0.4825
Augmented	0.9150	0.95697	0.48852

The following analysis and testing would base on the model trained on the augmented dataset as it is expected to performs better in real scenes. To quantitatively evaluate the performance of the model trained on augmented radargrams in detecting and characterizing pipelines, 100 reconstructed permittivity maps which contain 388 pipelines are randomly selected from the validation dataset. Table 6 shows the detection results: 304 pipelines are successfully detected while 84 pipelines are miss detected among 388 pipelines. This indicates that the network achieves satisfactory results with a precision of 96.2% and an accuracy of 76%.

Table 6: Confusion matrix results of model trained on augmented data

Confusion Matrix	Observed Value
True Positive (TP)	304
False Negative (FN)	84
False Positive (FP)	12
Precision	0.962
Recall	0.784

Accuracy	0.76

Furthermore, We evaluated the model's performance by measuring the depth and diameter of 304 pipelines. Pipeline depth was determined from the vertex depth of the reconstructed pipeline, while diameter was assessed using the horizontal width of the reconstructed pipeline in the predicted permittivity map. As Table 7 details, the model excelled in depth prediction, with an average error of 1.77 cm and deviation of 1.71%. Diameter prediction showed an average error of 3.20 cm and deviation of 20.44%. Depth prediction was strongly influenced by the accuracy of the reconstructed soil permittivity, as this affects signal transmission time, hence reflecting depth. Meanwhile, the precision of diameter prediction relied on accurate pipeline permittivity value detection. Smaller pipelines proved more challenging for permittivity prediction, resulting in less accurate diameter predictions with around 20% deviation.

Table 7: Performance of pipeline diameter and depth prediction

	Average error (cm)	Average deviation (%)	SD of average error	SD of average deviation
Depth	1.77	1.71	2.31	2.35
Diameter	3.20	20.44	3.20	26.80

Note: SD is the standard deviation.

Figure 10 below presents examples of inversion results on the radargrams augmented by GANILLA GAN. Each row in Figure 10 shows one example of the prediction result using the trained model. The first column in the plot presents the augmented radargrams, and the rest columns show their related ground truth and predicted permittivity maps in the time and depth domain respectively. As indicated in the figure, the predicted permittivity map is in good agreement with the ground-truth permittivity map, which shows the robust applicability of the proposed inversion model.

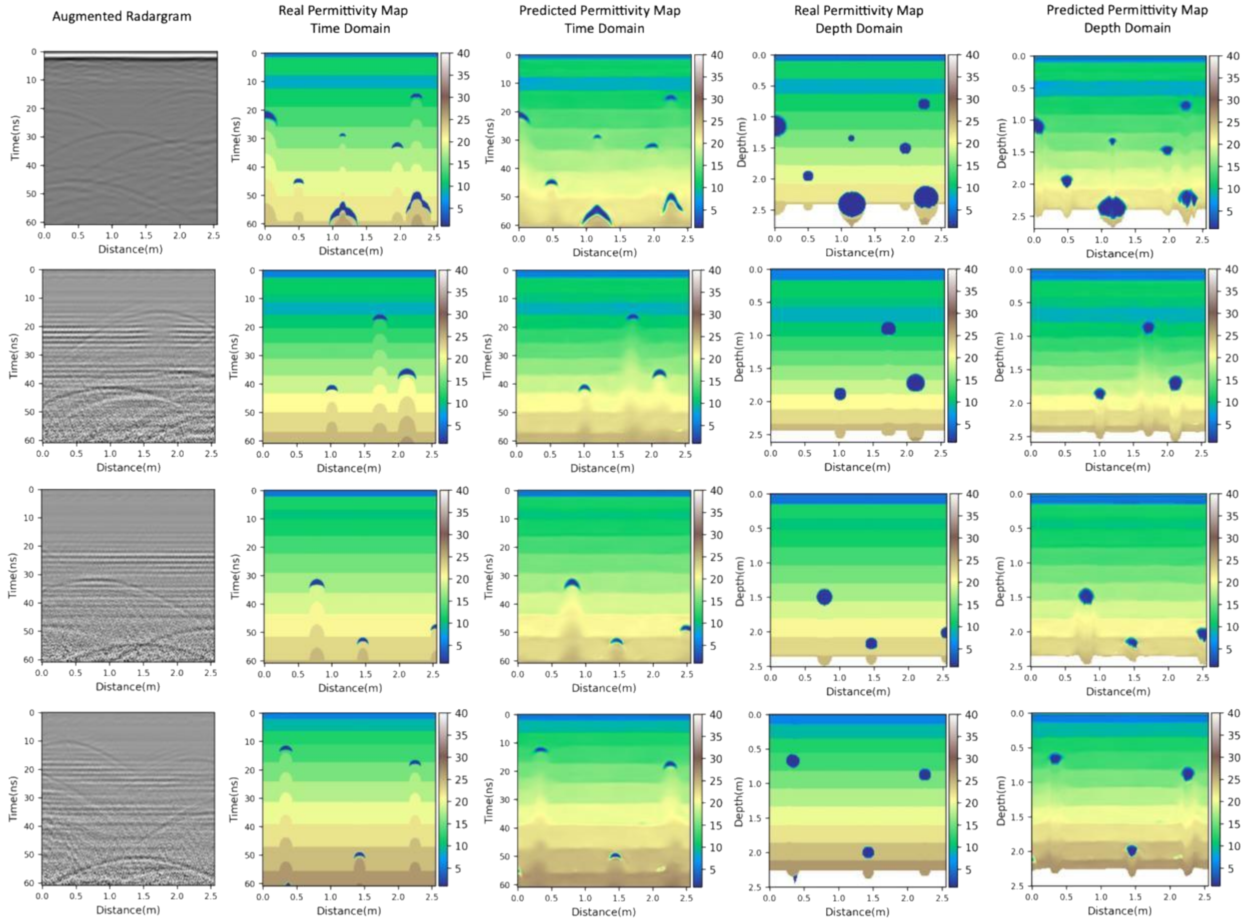

Figure 10: Examples of inversion model predictions results.

4.3 Subsurface Mapping Case study

To determine the performance of the proposed model in real scenes, the open-source radargram dataset created by [49] is adapted due to the availability of subsurface cross-sections. The experimental pit from the IFSTTAR geographical test site is a transversal trench filled with Gneiss 14/20 gravel. The overall length of this gravel region is 5 m and its density is around 1.8t/m3. The radargram data used for testing is collected by GSSI 350 MHz GPR. Figure 11 shows nine PVC tubes with a 0.1 m diameter buried in three depths separately, 0.8 m, 1.4 m, and 2.0 m. In addition, a concrete pipe with a diameter of 0.5m is buried 2m below the ground. The pink in the background means the Gneiss 14/20 gravels. Two cross-sections with pipelines are selected for evaluation as highlighted in the red bounding boxes in the figure.

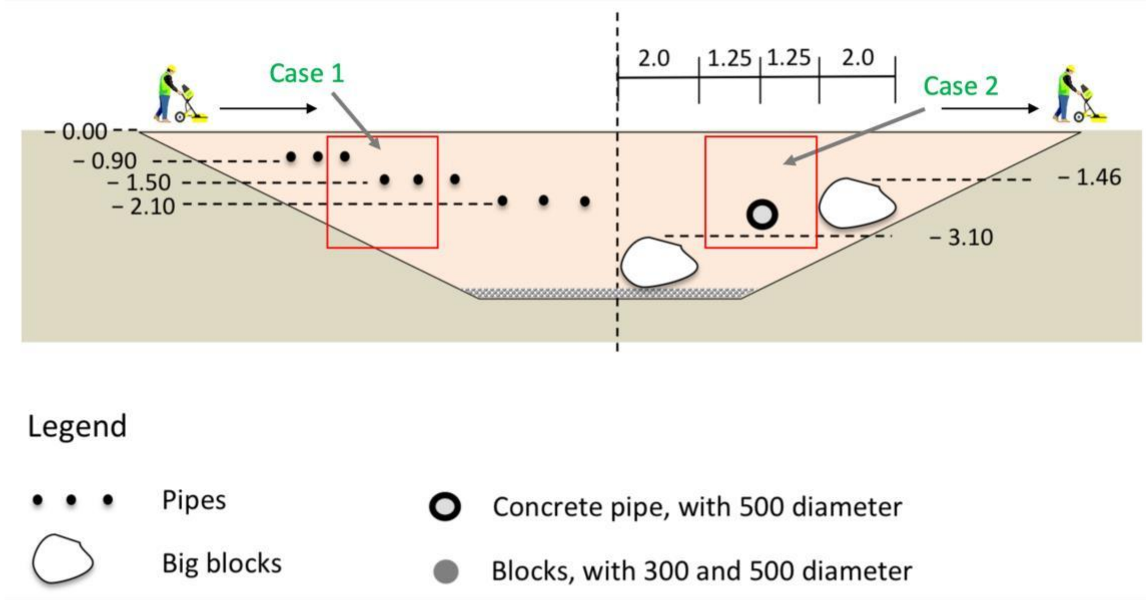

Figure 11: Cross-section of the real scene test case. (Adapted from [49])

The first case contains one PVC tube with a 0.1 m diameter buried in 0.9 m and another two same-size tubes buried in 1.5 m with an interval of around 1.0 m. The second case is the cross-section containing the concrete pipeline. Figure 12 shows the results of the two cases.

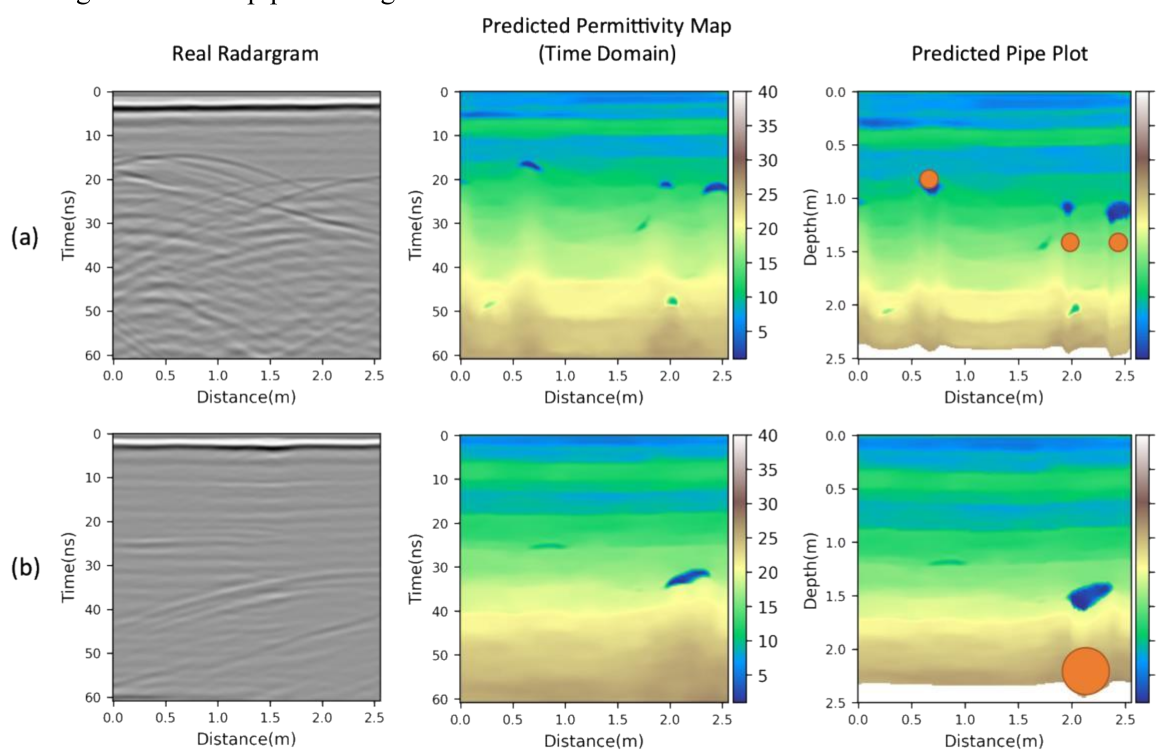

Figure 12: Prediction results of the real scene radargrams. Note: (a) denotes results related to the case 1, and (b) represents results related to case 2. The orange dots denote the real pipelines.

The result indicate that all the buried pipelines are successfully detected in both cases. However, there is approximately 0.4 m discrepancy in the depth of the detected pipeline in each case. The shape of the reconstructed pipelines appears deformed. These inaccuracies, including errors in depth prediction, radius prediction and shape deformation, can be attributed to imprecisions in the predicted permittivity values. Several factors contributed to this: First, although the proposed model possesses the ability to discern road base layers, the training data consists of simulated multi-layer data, whereas the case study is conducted on a single-layer site. Second, the model's training dataset lacks labeled real data, limiting the model's applicability in real-world scenarios. The limited real data used for GAN training also impacts the performance of the trained GAN model. These factors influence the accuracy of predicted permittivity values when the trained inversion model is directly applied to real sites, leading to a deviation between the reconstructed permittivity map in the depth domain and the original cross-section.

4.4 Integrated Mapping Case Study

The proposed system was field-deployed, with data collected near the Tickle College of Engineering at the University of Tennessee. Figure 13 shows the route of GPR survey.

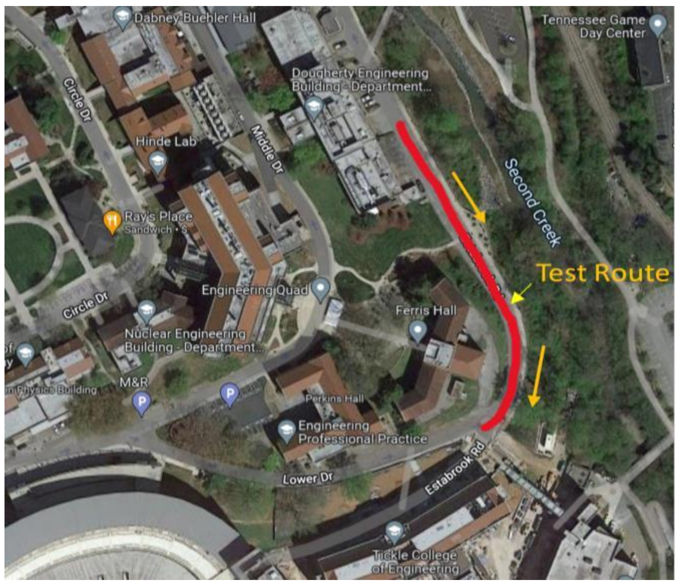

Figure 13: Test route displayed in Google Maps.

A GSSI GPR equipped with 350 MHz antennas was utilized in the field test. Concerning the collection parameters, the acquisition time window was set to 70 ns, and the wheel encoder was configured at 100 pulses per meter, resulting in a trace spacing of 0.01 m. Approximately 80 meters of GPR radargrams were collected along the test route. Additionally, a ZED camera was employed in the experiment to capture environmental data. After processing with the RTAB-Map SLAM model, the aboveground environment was successfully reconstructed, as demonstrated in Figure 14. The results accurately reconstructed the road and its surroundings, highlighting the proposed system's ability to generate timely outdoor aboveground maps. Concurrently, the collected GPR radargrams were input into the proposed inversion model to determine the location and size of buried pipelines at the site. Four sample detection results, including the real GPR radargram and its corresponding detected subsurface map, are also presented in Figure 14. These results showcase the proposed model's capability to detect and localize buried pipelines in real-world scenarios.

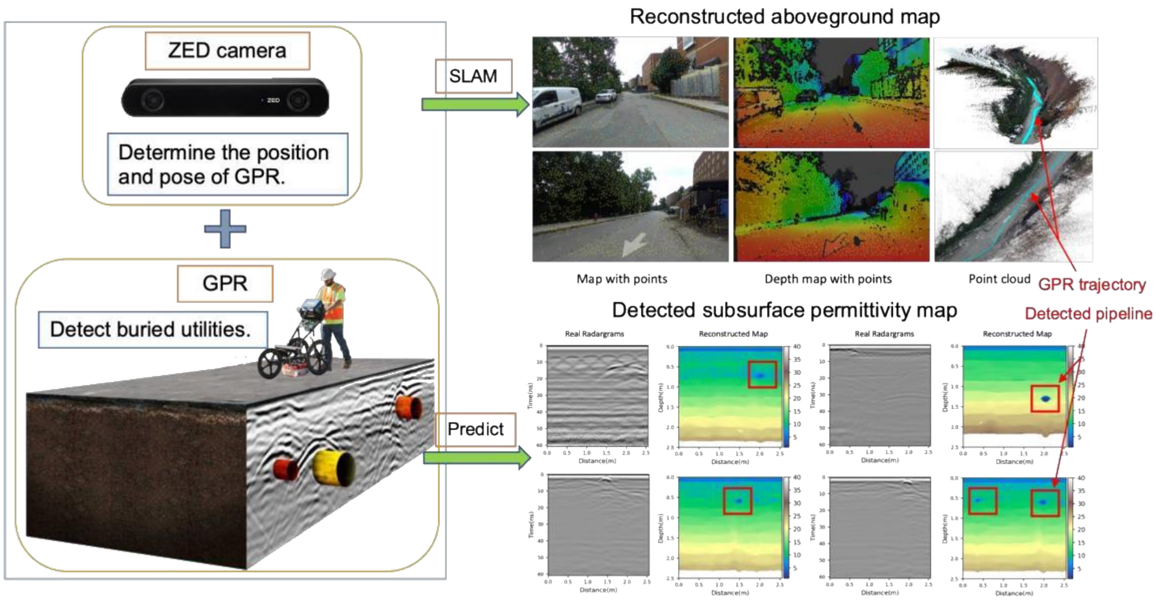

Figure 14: Pipeline geo-registration process.

5. Conclusions

The growing need to prevent excavation accidents, primarily caused by a lack of accurate pipeline information, has led to the development of subsurface pipeline mapping methods that geo-registers buried pipelines in a 3D environment. This study proposes a CNN-based GPR inversion model to directly reconstruct permittivity maps, enabling the extraction of size and location information of pipelines. Simultaneously, detected pipelines are geo-registered in the reconstructed aboveground map using vSLAM. A wealth of synthetic radargrams and corresponding permittivity map labels are generated for training the inversion network. Additionally, the GANILLA GAN network is employed to enhance the realism of synthetic GPR scans. The inversion model, based on an encoder-decoder structure, is trained on augmented radargrams and their corresponding labels, demonstrating high accuracy. In particular, training results on augmented data yield an SSIM of 0.915, R^2 of 0.957, and MAE of 0.488. The trained model is numerically evaluated on 100 randomly selected radargrams, revealing the proposed model's performance in detecting pipelines with a precision of 96.2%, recall of 78.4%, and accuracy of 76%. Furthermore, the reconstruction accuracy for size and depth is assessed, with an average buried depth reconstruction error of 1.77 cm (deviation of 1.71%) and an average diameter error of 3.2 cm (deviation of 20.44%). These results indicate excellent performance in predicting pipe location and size for simulated cases. In this study, the proposed GPR inversion model is evaluated in a field case, and the capacity of the proposed system to construct an integrated pipeline map is demonstrated in another field case.

While our GPR inversion model exhibits high accuracy when predicting pipeline size and depth in synthetic radargrams, it tends to present larger discrepancies in real-world cases. These discrepancies can be attributed to several factors. The first is the inherent simplicity of the current inversion model, which may not fully account for the complexity of real-world conditions. The second factor is the unpredictable noise in real-scenario radargrams, which can lead to less accurate predictions. Lastly, the limited number of real radargrams available for GAN training may affect the model's performance in diverse, real-world situations. To address these challenges, several avenues of future research are suggested. Improving the robustness of the training network can help the model better handle the noise and unpredictability of real-world data. Enlarging the dataset and collecting more real-world radargrams could enhance the model's generalization capability, improving its accuracy across various scenarios. Furthermore, the inversion network could be expanded to analyze additional subsurface conditions. For instance, it could be adapted to investigate pavement structures, allowing for a more comprehensive understanding of subsurface conditions. Lastly, improvements in the SLAM algorithm and location calibration technique could enhance the geo-registration process. By increasing the precision of the SLAM algorithm and refining the location calibration method, the system could provide more accurate localization of detected subsurface objects. These enhancements would result in a more robust and versatile tool for urban infrastructure management and planning.

Acknowledgement

This research was funded by the U.S. National Science Foundation (NSF) via Grant 2222810 and 2129003. The authors gratefully acknowledge NSF's support. Any opinions, findings, recommendations, and conclusions in this paper are those of the authors and do not necessarily reflect the views of NSF, the University of Tennessee, Knoxville, Kennesaw State University, and the University of Hong Kong.

References

[1] S. Talmaki, V.R. Kamat, H. Cai, Geometric modeling of geospatial data for visualization-assisted excavation, Advanced Engineering Informatics. 27 (2013) pp. 283–298. https://doi.org/10.1016/J.AEI.2013.01.004.

[2] Common Ground Alliance, 2020 Damage Information Reporting Tool (DIRT) Annual Report, 2021. https://commongroundalliance.com/Resources/2020-DIRT-Report (accessed September 29, 2022).

[3] S. Marvin, S. Slater, Urban infrastructure: the contemporary conflict between roads and utilities, Progress in Planning. 4 (1997) pp. 247–318. https://doi.org/10.1016/S0305-9006(97)90019-2.

[4] A. Beck, A.G. Cohn, J. Parker, N. Boukhelifa, G. Fu, Seeing the unseen: delivering integrated underground utility data in the UK, in: Proceedings of the GeoWeb Conference, Vancouver, 2009. https://www.isprs.org/proceedings/XXXVIII/3_4-C3/Paper_GeoW09/paper07_beck.pdf (accessed June 14, 2023).

[5] R.L. Sterling, J.H. Anspach, E.N. Allouche, J. Simicevic, C.D. Rogers, K.E. Weston, K. Hayes, Encouraging innovation in locating and characterizing underground utilities, SHRP2 Transportation Research Board, USA. (2009). https://doi.org/10.17226/22994.

[6] J.H. Anspach, Collecting and Converting 2-D Utility Mapping to 3-D, in: Pipelines 2010: Climbing New Peaks to Infrastructure Reliability: Renew, Rehab, and Reinvest, 2010: pp. 278–284. https://doi.org/10.1016/j.autcon.2020.103229.

[7] S. Li, H. Cai, M. Abraham Dulcy, P. Mao, Estimating Features of Underground Utilities: Hybrid GPR/GPS Approach, Journal of Computing in Civil Engineering. 30 (2016). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000443.

[8] H. Chen, A.G. Cohn, Probabilistic robust hyperbola mixture model for interpreting ground penetrating radar data, in: The 2010 International Joint Conference on Neural Networks (IJCNN), IEEE, 2010: pp. 1–8. https://doi.org/10.1109/IJCNN.2010.5596298.

[9] S. Li, H. Cai, V.R. Kamat, Uncertainty-aware geospatial system for mapping and visualizing underground utilities, Automation in Construction. 53 (2015) pp. 105–119. https://doi.org/10.1016/j.autcon.2015.03.011.

[10] F. Yang, X. Qiao, Y. Zhang, X. Xu, Prediction method of underground pipeline based on hyperbolic asymptote of GPR image, in: Proceedings of the 15th International Conference on Ground Penetrating Radar, 2014: pp. 674–678. https://doi.org/10.1109/ICGPR.2014.6970511.

[11] B.P.A. Rohman, M. Nishimoto, Basic Shape Classification of Buried Object Using Pattern Matching in Ultrawideband Radar Image, in: 2020 International Symposium on Antennas and Propagation (ISAP), 2021: pp. 739–740. https://doi.org/10.23919/ISAP47053.2021.9391241.

[12] T. Yamaguchi, T. Mizutani, T. Nagayama, Mapping Subsurface Utility Pipes by 3-D Convolutional Neural Network and Kirchhoff Migration Using GPR Images, IEEE Transactions on Geoscience and Remote Sensing. 59 (2021) pp. 6525–6536. https://doi.org/10.1109/TGRS.2020.3030079.

[13] F. Hou, W. Lei, S. Li, J. Xi, Deep Learning-Based Subsurface Target Detection From GPR Scans, IEEE Sensors Journal. 21 (2021) pp. 8161–8171. https://doi.org/10.1109/JSEN.2021.3050262.

[14] K. Dinh, N. Gucunski, T. Zayed, Automated visualization of concrete bridge deck condition from GPR data, NDT & E International. 102 (2019) pp. 120–128. https://doi.org/10.1016/j.ndteint.2018.11.015.

[15] D. Hu, S. Li, J. Cai, A Machine Learning-Based Framework for Automatic Bridge Deck Condition Assessment Using Ground Penetrating Radar, Computing in Civil Engineering 2021 - Selected Papers from the ASCE International Conference on Computing in Civil Engineering 2021. (2021) pp. 74–82. https://doi.org/10.1061/9780784483893.010.

[16] D. Hu, F. Hou, J. Blakely, S. Li, Augmented Reality Based Visualization for Concrete Bridge Deck Deterioration Characterized by Ground Penetrating Radar, Construction Research Congress 2020: Computer Applications - Selected Papers from the Construction Research Congress 2020. (2020) pp. 1156–1164. https://doi.org/10.1061/9780784482865.122.

[17] D. Hu, L. Chen, J. Du, J. Cai, S. Li, Seeing through Disaster Rubble in 3D with Ground-Penetrating Radar and Interactive Augmented Reality for Urban Search and Rescue, Journal of Computing in Civil Engineering. 36 (2022). https://doi.org/10.1061/(ASCE)CP.1943-5487.0001038/ASSET/07C27AC7-9A0D-42FF-ADD0-6806A2DFF2A9/ASSETS/IMAGES/LARGE/FIGURE20.JPG.

[18] D. Hu, J. Chen, S. Li, Reconstructing unseen spaces in collapsed structures for search and rescue via deep learning based radargram inversion, Automation in Construction. 140 (2022). https://doi.org/10.1016/J.AUTCON.2022.104380.

[19] D. Hu, S. Li, J. Chen, V.R. Kamat, Detecting, locating, and characterizing voids in disaster rubble for search and rescue, Advanced Engineering Informatics. 42 (2019). https://doi.org/10.1016/J.AEI.2019.100974.

[20] M. Metwaly, Application of GPR technique for subsurface utility mapping: A case study from urban area of Holy Mecca, Saudi Arabia, Measurement. 60 (2015) pp. 139–145. https://doi.org/10.1016/j.measurement.2014.09.064.

[21] G. Curioni, D.N. Chapman, N. Metje, Seasonal variations measured by TDR and GPR on an anthropogenic sandy soil and the implications for utility detection, Journal of Applied Geophysics. 141 (2017) pp. 34–46. https://doi.org/10.1016/j.jappgeo.2017.01.029.

[22] U. Boniger, J. Tronicke, Subsurface utility extraction and characterization: Combining GPR symmetry and polarization attributes, IEEE Transactions on Geoscience and Remote Sensing. 50 (2011) pp. 736–746. https://doi.org/10.1109/TGRS.2011.2163413.

[23] J.L. Porsani, Y.B. Ruy, F.P. Ramos, G.R.B. Yamanouth, GPR applied to mapping utilities along the route of the Line 4 (yellow) subway tunnel construction in São Paulo City, Brazil, Journal of Applied Geophysics. 80 (2012) pp. 25–31. https://doi.org/10.1016/j.jappgeo.2012.01.001.

[24] A. De Coster, J.L.P. Medina, M. Nottebaere, K. Alkhalifeh, X. Neyt, J. Vanderdonckt, S. Lambot, Towards an improvement of GPR-based detection of pipes and leaks in water distribution networks, Journal of Applied Geophysics. 162 (2019) pp. 138–151. https://doi.org/10.1016/j.jappgeo.2019.02.001.

[25] M.A. Rahman, T. Zayed, Viola-Jones Algorithm for Automatic Detection of Hyperbolic Regions in GPR Profiles of Bridge Decks, in: 2018 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI), IEEE, 2018: pp. 1–4. https://doi.org/10.1109/SSIAI.2018.8470374.

[26] C. Maas, J. Schmalzl, Using pattern recognition to automatically localize reflection hyperbolas in data from ground penetrating radar, Computers & Geosciences. 58 (2013) pp. 116–125. https://doi.org/10.1016/j.cageo.2013.04.012.

[27] S. Delbo, P. Gamba, D. Roccato, A fuzzy shell clustering approach to recognize hyperbolic signatures in subsurface radar images, IEEE Transactions on Geoscience and Remote Sensing. 38 (2000) pp. 1447–1451. https://doi.org/10.1109/36.843039.

[28] F. Sagnard, J.-P. Tarel, Template-matching based detection of hyperbolas in ground-penetrating radargrams for buried utilities, Journal of Geophysics and Engineering. 13 (2016) pp. 491–504. https://doi.org/10.1088/1742-2132/13/4/491.

[29] Z.W. Wang, M. Zhou, G.G. Slabaugh, J. Zhai, T. Fang, Automatic Detection of Bridge Deck Condition From Ground Penetrating Radar Images, IEEE Transactions on Automation Science and Engineering. 8 (2011) pp. 633–640. https://doi.org/10.1109/TASE.2010.2092428.

[30] G. Terrasse, J.-M. Nicolas, E. Trouvé, É. Drouet, Automatic localization of gas pipes from GPR imagery, in: 2016 24th European Signal Processing Conference (EUSIPCO), IEEE, 2016: pp. 2395–2399. https://doi.org/10.1109/EUSIPCO.2016.7760678.

[31] C. Yuan, S. Li, H. Cai, V.R. Kamat, GPR signature detection and decomposition for mapping buried utilities with complex spatial configuration, Journal of Computing in Civil Engineering. 32 (2018). https://doi.org/10.1061/(ASCE)CP.1943-5487.000076.

[32] H. Liu, C. Lin, J. Cui, L. Fan, X. Xie, B.F. Spencer, Detection and localization of rebar in concrete by deep learning using ground penetrating radar, Automation in Construction. 118 (2020). https://doi.org/10.1016/J.AUTCON.2020.103279.

[33] H. Qin, Y. Tang, Z. Wang, X. Xie, D. Zhang, Shield tunnel grouting layer estimation using sliding window probabilistic inversion of GPR data, Tunnelling and Underground Space Technology. 112 (2021). https://doi.org/https://doi.org/10.1016/j.tust.2021.103913.

[34] B. Liu, Y. Ren, H. Liu, H. Xu, Z. Wang, A.G. Cohn, P. Jiang, GPRInvNet: Deep Learning-Based Ground-Penetrating Radar Data Inversion for Tunnel Linings, IEEE Transactions on Geoscience and Remote Sensing. 59 (2021) pp. 8305–8325. https://doi.org/10.1109/TGRS.2020.3046454.

[35] Y. Ji, F. Zhang, J. Wang, Z. Wang, P. Jiang, H. Liu, Q. Sui, Deep Neural Network-Based Permittivity Inversions for Ground Penetrating Radar Data, IEEE Sensors Journal. 21 (2021) pp. 8172–8183. https://doi.org/10.1109/JSEN.2021.3050618.

[36] G. Kouros, I. Kotavelis, E. Skartados, D. Giakoumis, D. Tzovaras, A. Simi, G. Manacorda, 3D Underground Mapping with a Mobile Robot and a GPR Antenna, IEEE International Conference on Intelligent Robots and Systems. (2018) pp. 3218–3224. https://doi.org/10.1109/IROS.2018.8593848.

[37] H. Li, C. Chou, L. Fan, B. Li, D. Wang, D. Song, Toward Automatic Subsurface Pipeline Mapping by Fusing a Ground-Penetrating Radar and a Camera, IEEE Transactions on Automation Science and Engineering. 17 (2020) pp. 722–734. https://doi.org/10.1109/TASE.2019.2941848.

[38] J. Feng, L. Yang, H. Wang, Y. Song, 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. https://doi.org/10.1109/ICRA40945.2020.9197043.

[39] M. Wang, D. Hu, S. Li, J. Cai, Urban Subsurface Mapping via Deep Learning Based GPR Data Inversion, in: 2022 Winter Simulation Conference (WSC), IEEE, 2022: pp. 2440–2450. https://doi.org/10.1109/WSC57314.2022.10015357.

[40] C. Warren, A. Giannopoulos, I. Giannakis, gprMax: Open source software to simulate electromagnetic wave propagation for Ground Penetrating Radar, Computer Physics Communications. 209 (2016) pp. 163–170. https://doi.org/10.1016/J.CPC.2016.08.020.

[41] Tennessee Department of Transportation, Tennessee Department of Transportation Standard Specification for Road and Bridge Construction, 2021. https://www.tn.gov/content/dam/tn/tdot/construction/2021-standard-specifications/January_1_2021_Standard_Specifications.pdf (accessed April 21, 2022).

[42] ASME, Pipeline Transportation Systems for Liquids and Slurries ASME Code for Pressure Piping, 2016. https://www.academia.edu/44003373/Pipeline_Transportation_Systems_for_Liquids_and_Slurries_ASME_Code_for_Pressure_Piping_B31_A_N_A_M_E_R_I_C_A_N_N_A_T_I_O_N_A_L_S_T_A_N_D_A_R_D (accessed April 18, 2022).

[43] M.S. Hilario, B.W. Hoff, B. Jawdat, M.T. Lanagan, Z.W. Cohick, F.W. Dynys, J.A. Mackey, J.M. Gaone, W-Band Complex Permittivity Measurements at High Temperature Using Free-Space Methods, IEEE Transactions on Components, Packaging and Manufacturing Technology. 9 (2019) pp. 1011–1019. https://doi.org/10.1109/TCPMT.2019.2912837.

[44] S. Zhao, Y. Sun, Y. Zhou, X. Han, Experimental Study on Dielectric Properties of Pavement Structure Layer Based on Radar Image, Chemical Engineering Transactions. 66 (2018) pp. 883–888. https://doi.org/10.3303/CET1866148.

[45] G. Alsharahi, A. Driouach, A. Faize, Performance of GPR Influenced by Electrical Conductivity and Dielectric Constant, Procedia Technology. 22 (2016) pp. 570–575. https://doi.org/10.1016/J.PROTCY.2016.01.118.

[46] S. Hicsonmez, N. Samet, E. Akbas, P. Duygulu, GANILLA: Generative adversarial networks for image to illustration translation, Image and Vision Computing. 95 (2020). https://doi.org/10.1016/j.imavis.2020.103886.

[47] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 1125–1134. https://doi.org/10.48550/arXiv.1611.07004.

[48] M. Labbé, F. Michaud, RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation, Journal of Field Robotics. 36 (2019) pp. 416–446. https://doi.org/10.1002/ROB.21831.

[49] X. Dérobert, L. Pajewski, TU1208 open database of radargrams: The dataset of the IFSTTAR geophysical test site, Remote Sensing. 10 (2018). https://doi.org/10.3390/rs10040530.

[50] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Transactions on Image Processing. 13 (2004) pp. 600–612. https://doi.org/10.1109/TIP.2003.819861.