

# Robotic Inspection and Data Analytics to Localize and Visualize the Structural Defects of Concrete Infrastructure

Jinglun Feng, Bo Shang<sup>ID</sup>, Ejup Hoxha<sup>ID</sup>, César Hernández<sup>ID</sup>, Yang He,  
Weihan Wang, and Jizhong Xiao<sup>ID</sup>, *Senior Member, IEEE*

**Abstract**—This paper presents an innovative robotic inspection system designed to enhance the detection and analysis of structural defects in concrete infrastructure. The proposed inspection system is comprised of three modules: a robotic data collection module, a visual inspection module, and a subsurface mapping module. The robotic data collection module features an omnidirectional robotic platform, designed to move sideways without spinning. It is equipped with Ground Penetrating Radar (GPR) and RGB-D cameras, facilitating systematic data collection across construction sites. The visual inspection module employs a learning-based method, InspectionNet++, to analyze the frames for surface defects such as cracks, spalls, and stains, providing high accuracy and metric measurements of the defects. The subsurface mapping module processes the GPR data to detect and visualize hidden defects, creating a comprehensive map that correlates these with visible surface anomalies. Field tests demonstrate the system's ability to automate construction structural inspection with improved efficiency and precision. Additionally, the customized visualization software is introduced to enable intuitive and interactive exploration of the detected defects within a unified interface. By automating data collection and enhancing defect detection through learning algorithms, the system not only speeds up the inspection process but also increases the reliability of infrastructure evaluations, supporting more informed maintenance decisions.

**Note to Practitioners**—This paper introduces a robotic solution for inspection and condition assessment of concrete infrastructure. The system uses an omnidirectional robot equipped with GPR and RGB-D cameras to automatically collect data across construction sites. By harnessing vision-based positioning

Received 6 November 2024; accepted 8 January 2025. Date of publication 27 January 2025; date of current version 16 October 2025. This article was recommended for publication by Associate Editor J. H. Ryu and Editor J. Yi upon evaluation of the reviewers' comments. This work was supported in part by U.S. National Science Foundation (NSF) under Grant IIP-2112199; and in part by U.S. Department of Transportation through the Inspecting and Preserving Infrastructure through Robotic Exploration (INSPIRE) University Transportation Center, Missouri University of Science and Technology, under Grant 69A3551747126. The work of Jizhong Xiao was supported by InnovBot LLC, a company involved in research and development and commercialization of the technology. (*Jinglun Feng and Bo Shang contributed equally to this work.*) (*Corresponding author: Jizhong Xiao.*)

Jinglun Feng, Bo Shang, César Hernández, Yang He, and Jizhong Xiao are with the CCNY Robotics Laboratory, Department of Electrical Engineering, The City College of New York, New York, NY 10031 USA (e-mail: jxiao@ccny.cuny.edu).

Ejup Hoxha is with the CCNY Robotics Laboratory, Department of Electrical Engineering, The City College of New York, New York, NY 10031 USA, and also with Amazon AWS, New York, NY 10001 USA.

Weihan Wang is with the Stevens Institute of Technology, Hoboken, NJ 07030 USA.

Digital Object Identifier 10.1109/TASE.2025.3535227

technology, our system empowers the robot to scan the ground surface in free motion pattern. This eliminates the need for time-consuming grid line setup traditionally required for manual GPR data collection. Our approach combines multi-sensor data analytics with advanced software, which enables the detection and visualization of both surface defects (cracks, spalls, stains) and subsurface anomalies. For practitioners, this automated approach offers several key benefits over manual inspections: increased inspection speed and coverage, rapid data collection enabled by robotic free motion, higher detection accuracy, quantitative defect measurements, and unified visualization correlating surface/subsurface conditions. This system has the potential to revolutionize infrastructure assessment practices. By facilitating more frequent and reliable inspections with minimal human intervention, it paves the way for proactive maintenance and ensures the sustainability of critical infrastructure. A current limitation is the relatively small training dataset for visual inspection, which may affect generalizability across diverse concrete structures. Future work aims to expand the dataset, improve irrelevant feature filtering, and utilize other NDE sensors (e.g., impact echo) for infrastructure inspection.

**Index Terms**—Automation inspection, ground penetrating radar (GPR), visual inspection, robotic inspection, non-destructive evaluation (NDE).

## I. INTRODUCTION

THE infrastructure underpins nearly every aspect of life, and maintaining its quality and resilience is essential for supporting modern society. According to the 2021 Report Card for America's Infrastructure [23], the U.S. has made incremental progress towards restoring the nation's infrastructure, raising its GPA to a C- from a D+ in 2017. However, the total investment gap has increased from 2.1 trillion over ten years to nearly 2.59 trillion over the same period. To enhance the resilience and sustainability of civil infrastructure, it is essential to develop a strategic and comprehensive plan for inspection, evaluation, and modernization. In the field of infrastructure assessment and maintenance, both non-destructive evaluation (NDE) techniques and vision-based surface flaw detection systems are vital. NDE methods allow for condition assessment of concrete structures without causing damage, offering a reliable approach for ongoing evaluations [29]. Similarly, advancements in camera technology and computer vision algorithms have greatly improved the efficiency and accuracy of vision-based surface flaw detection systems, streamlining defect identification and assessment [40].

However, these systems and techniques present certain challenges. In current NDE practices, data collection relies heavily on human expertise, requiring inspectors with significant knowledge of the survey site for accurate defect identification. Furthermore, human inspectors are required to pre-draw a grid map on the ground especially for subsurface data collection, maneuver the GPR device along straight lines in the X and Y directions, utilize wheel encoders to trigger GPR sampling, and meticulously document these linear motion trajectories for 3D GPR imaging. At last, obtaining precise location and metric measurements, which are essential for NDE inspections, remains challenging, particularly in environments where GPS is unavailable or when there is low accuracy in the wheel encoders of GPR carts.

In vision-based surface flaw detection systems, the challenge lies in obtaining accurate measurements due to the limitations of camera sensors, integrating visual inspection results with other NDE data like GPR to create a more holistic understanding of structural conditions, and developing user-friendly software solutions to bridge the gap between research algorithms and the practical tools required by civil engineers on-site. These problems arise due to a myriad of factors, such as differences in data modalities, scales, and resolutions, as well as the need for data to be accessible and interpretable to engineers without extensive knowledge of the underlying research algorithms [8], [38]. Overcoming these challenges is key to further enhancing the efficiency and reliability of infrastructure assessment and maintenance processes.

To address these challenges, this paper presents the development of an omnidirectional robotic-based inspection system designed to automate NDE data collection for both subsurface and surface infrastructure elements, reducing the burden on civil engineers and providing metric measurement tracking of defects. Compared to current advancements in the NDE field, our main contributions are articulated through the integration of advanced robotics and multi-sensor data fusion technologies, as highlighted below:

- We develop a robotic inspection system that leverages vision-based SLAM to achieve precise positioning, allowing GPR measurements to be accurately tagged with the robot's real-time pose in a synchronized manner. This enables the robot to scan ground surfaces in a free-motion pattern and facilitates high-resolution 3D GPR imaging. Our previous publications [9], [11], [14] provide more technical details on 3D GPR imaging techniques.
- We introduce InspectionNet++, a deep learning model that achieves precise pixel-level segmentation of surface defects, categorizing them into cracks, spalls, and stains. Distinct from existing methods, InspectionNet++ not only detects surface flaws but also classifies them effectively through training on a large dataset, providing targeted defect categorization for more refined inspection insights compared to other crack detection approaches [29].
- We leverage the open-source WebODM framework and develop software plugins to produce data analytics tools capable of reconstructing 3D maps from 2D inspection

video images, automatically detecting and measuring defect lengths, and enabling users to annotate and visualize defects within a 3D space.

## II. RELATED WORKS

### A. Concrete Structure Inspection Methods

On the one hand, visual inspection is one of the main techniques that is frequently used for **surface inspection**, which primarily relies on learning-based algorithms. Savino and Tondolo [31] developed a pixel-wise segmentation approach based on deep learning for accurate defect assessment in civil infrastructures, significantly enhancing the precision in identifying and classifying defects. In addition, Ai et al. [2] provided a comprehensive review of computer vision frameworks for crack detection, offering insights into the evolution of these technologies and their impact on infrastructure maintenance. Furthermore, Manaf [27] explored the utilization of deep learning for automated inspection and damage assessment, showcasing the potential of these technologies to revolutionize the maintenance strategies of civil infrastructure systems. Adding to these advancements, Agyemang et al. [1] and Zhang et al. [45] introduced novel vision-based algorithms for automated structural health monitoring, with the latter employing an Attention-Enhanced Co-Interactive Fusion Network (AECIF-Net) that significantly improves the detection and classification of structural issues through deep learning.

Recent studies also expand the scope of visual inspection techniques through the integration of advanced robotic technologies and novel algorithmic approaches. Liang et al. [41], [43], [44] introduced a robotics system that pioneers automatic crack and spalls detection using deep neural network model from region proposal inputs. Lee et al. [22] discussed the broad application of robotics technologies in civil infrastructure inspection, emphasizing how these technologies enhance the accuracy and safety of inspections. Ivić et al. [20] demonstrated the potential of multi-UAV systems in conducting comprehensive 3D visual inspections of complex structures, highlighting the importance of optimized trajectory planning for effective coverage and data acquisition.

On the other hand, GPR is widely used in the NDE industry for civil engineers to locate and **subsurface concrete structures**, measure pavement thickness and properties, and characterize subsurface features [12]. Back-projection (BP) is the most commonly used algorithm for GPR imaging and subsurface inspection [11], [13]. The primary focus in subsurface structural detection has been on advancing GPR-based 3D imaging methods. Recent advancements include the development of GPRNet, a system for model reconstruction of underground utilities [10], which significantly enhances the accuracy and detail of subsurface imaging.

Further developments by Feng et al. [9] have shown how robotic systems equipped with GPR can dramatically improve the efficiency of underground utility inspections. Their work includes not only localization but also detailed characterization of subsurface objects, integrating these data into a comprehensive 3D model [14]. Additionally, Hoxha et al. [17], [18] have explored the synergistic use of impact-echo techniques

alongside GPR, broadening the scope of detectable subsurface defects, from voids and cracks to more complex structural anomalies.

### B. Robotic Inspection Systems and Approaches

Robotic inspection systems have shown significant benefits in delivering more consistent and cost-effective inspection outcomes through intelligent algorithms compared to manual inspections. Over the past decades, numerous efforts have been dedicated to creating robotic systems for construction inspection, encompassing mobile robotic inspection systems, aerial robotic inspection systems, and wall-climbing robots.

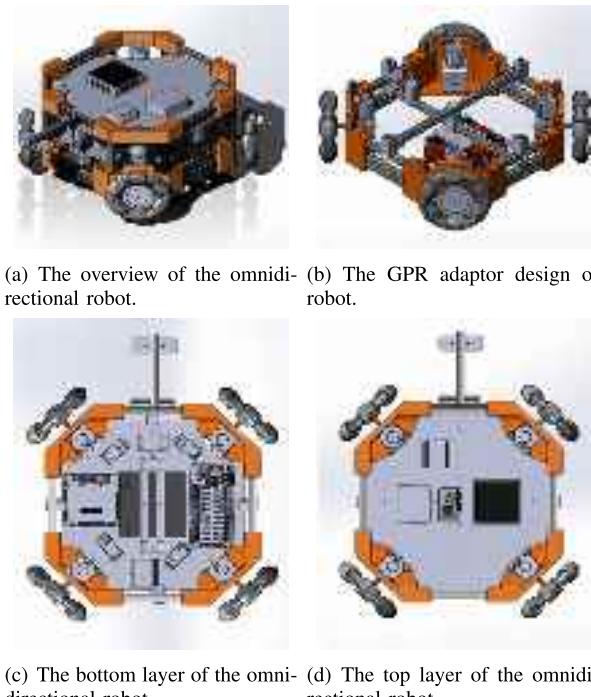
Ground mobile robots have demonstrated great potential for infrastructure inspection tasks, improving efficiency, safety, and data quality [16], [19], [33], [46]. Recent advancements include Shim et al.'s [33] remote robotic system for 3D measurement of concrete damage in tunnels, employing both ground vehicles and manipulators for enhanced damage assessment. Hu et al. [19] and Zhang et al. [46] have further expanded the use of unmanned ground vehicles, detailing applications, robotic platforms, sensors, and algorithms tailored for civil infrastructure management. Additionally, Xiao et al. [39] discuss the autonomous inspection and construction of civil infrastructures, presenting a comprehensive view of the integration of automation and robotics in the sector.

Besides ground mobile robots, aerial robotic inspection systems [4], [22], [43] and wall-climbing robots [42] have been utilized to examine hard-to-reach areas or obtain close-up images of different surfaces. These systems offer increased maneuverability and adaptability, making them suitable for various infrastructure inspection scenarios.

However, aerial and wall-climbing robotic systems face certain limitations. The primary limitation of aerial robots, particularly drones, is their restricted battery life and payload capacity, which confines inspection duration and scope to visual assessments only. This limitation prevents drones from performing contact-based NDT, which is often essential for comprehensive evaluations. [7]. In addition to capturing close-up visual inspection data in hard-to-access locations, wall-climbing robots offer vertical mobility and can perform contact-based inspection on vertical surfaces using NDT tools like GPR and impact echo devices. This capability provides a specialized solution for the challenging but critical niche of vertical surface inspections on infrastructure elements, including bridge piers, tunnels, dams, and building facades [28], [42]. On the other hand, ground mobile robots offer greater payload capacities, extended operational times, and more versatile inspection tool options. While they may lack the ability to access certain areas, their robustness and adaptability often make them more suitable for comprehensive, large-scale inspections [35]. Their efficiency and effectiveness can also be enhanced through integration with various sensors and advanced algorithms.

## III. ROBOTIC-BASED INSPECTION SYSTEM

The robotic system outlined in this paper features an omnidirectional design [9], [11] with mecanum wheels that



(a) The overview of the omnidi- (b) The GPR adaptor design of  
rectional robot. robot.

(c) The bottom layer of the omni- (d) The top layer of the omnidi-  
directional robot. directional robot.

Fig. 1. The overview of robot mechanism design. (a) shows the isometric view of the robot. (b) shows the cross-section view of the sensor adapter design. (c) shows the bottom view of the robot's top layer design and distribution. (c) shows the top view of the robot's top layer design and distribution.

enable multidirectional movement (forward, backward, and sideways) without the need for spinning. This mobility is essential for maneuvering across construction environments. The system is equipped with NDE tools including an RGB-D camera and a GPR antenna, enabling automated data collection on concrete infrastructures. Our study focuses on conducting visual inspections of surfaces using the RGB-D camera and detecting subsurface deterioration through the GPR sensor.

### A. Robot Mechanism Design

As illustrated in Fig. 1(a), the proposed robotic system features a 500mm square frame and is equipped with four omnidirectional wheels at each corner, as well as four independent motors, enabling efficient omnidirectional movement. The system is structured into two primary layers: a bottom layer and a top layer, with an adapter connecting the GPR sensor to the robot's mainframe.

As shown in Fig. 1(b), the adapter is designed to facilitate seamless integration of the GPR sensor with the robotic system. It employs a pair of bolts, which allows for adjusting the spacing between the GPR sensor and the ground. To maintain the alignment of the sensor, the adapter incorporates a cross-shaped arrangement of aluminum extrusions, preventing displacements along the X and Y axes.

The bottom layer, as demonstrated in Fig. 1(c), situated atop the cross-shaped arrangement, houses essential components, including the battery, power switch fuse, power distribution hub, roboRIO board, DC-DC converter, and Electronic Speed Controller (ESC) for the motors. This layer is

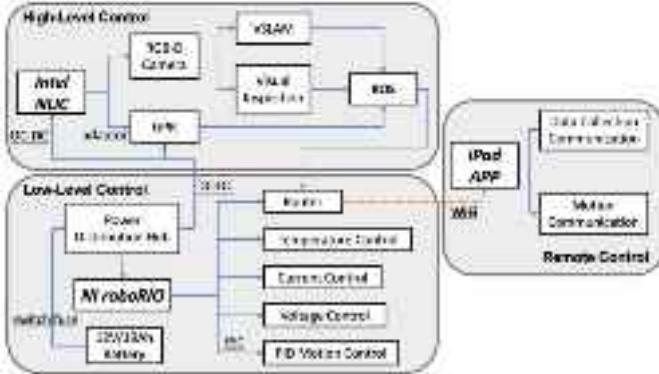


Fig. 2. Robot electronic system scheme.

responsible for low-level control, and the battery is strategically positioned in the center to preserve the robot's weight balance.

The top layer, as demonstrated in Fig. 1(d), secured to the robot using a cross-shaped arrangement of aluminum extrusions, accommodates high-level control hardware components such as the Intel NUC, 2.4GHz router for wireless operation, and a PoE adapter for the GPR sensor. Additionally, an Intel Realsense D435i camera is mounted on an aluminum extrusion connected to the top layer and is utilized for visual inspection of surface defects.

### B. Robot Electrical System

The robot's electrical system is presented in Fig. 2, emphasizing the three distinct levels of control: high-level control, low-level control, and remote control.

In the low-level control of our robot, a 12V/19Ah SLA battery is connected to a power distribution hub to deliver power to all the hardware components, including Intel NUC, GPR sensor, and roboRIO board. Furthermore, the roboRIO board functions as the motion controller, facilitating omnidirectional motion as illustrated in Fig. 3. The roboRIO board is wirelessly connected to a remote controller via a router. The remote controller, an iPad APP, is utilized for remotely controlling the robot's movement. Moreover, the roboRIO board can monitor the voltage, current, and temperature of each motor as well to provide safety monitoring.

In terms of high-level control, an Intel NUC board operates an RGB-D camera, the Intel Realsense D435i, and a GPR antenna (PaveScan RDM 2.0) from Geophysical Survey System Inc. (GSSI). The D435i, which is embedded with an IMU sensor, captures color and depth images of the surrounding environment. Then, using a time synchronizer function in the Robot Operating System (ROS), we can synchronize the GPR sampling and visual inspection images with positioning data, ensuring that multi-modal data collection is not limited to a straight line. Similar to the low-level control, ROS also wirelessly communicates with the remote controller, allowing the robot operator to remotely monitor the collection of GPR and vision data.

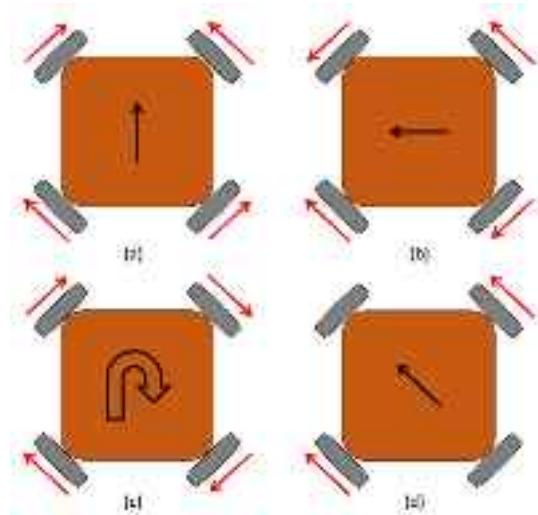


Fig. 3. Omnidirectional robot motion rules: (a) illustrates the robot moving straightforwardly; (b) depicts the robot's sideways movement; (c) demonstrates the robot's rotational motion; and (d) presents the robot's diagonal movement.

## IV. VISUAL INSPECTION FOR SURFACE STRUCTURAL DEFECTS

### A. Visual-Inertial SLAM Module

In this section, we utilize an Error-State Kalman Filter (ESKF) based visual-inertial SLAM (VI-SLAM) [36], [37] module to provide positioning information for the aforementioned robotic-based inspection system. This VI-SLAM module employs an online initialization method to estimate precise gyroscope bias and utilize the gyroscope bias value for rotation estimates from IMU measurements.

Giving an two consecutive keyframes  $KF_t$  and  $KF_{t+1}$  with the interval of time  $\Delta t$ , the nominal state is presented as  $\mathcal{X}_{KF_{t+1}} = [\mathbf{R}_{KF_{t+1}}^w, \mathbf{b}_{g,KF_{t+1}}]$ , where  $\mathbf{R}_{KF_{t+1}}^w$  represents the relative rotation from the keyframe to world coordinate, and  $\mathbf{b}_{g,KF_{t+1}}$  represents the gyroscope bias of the keyframe  $KF_{t+1}$ . Then, the error state can be expressed to  $\delta\mathcal{X}_{KF_{t+1}} = [\delta\theta_{\mathcal{X}_{KF_{t+1}}}, \delta\mathbf{b}_{g,KF_{t+1}}]$ , the error state of rotation and gyroscope bias could be integrated. We then use IMU measurements to estimate ESKF in Equ. 1:

$$\begin{aligned} \delta\theta_{KF_{t+1}} &= \text{Exp}(-(\omega_m - \mathbf{b}_g)\Delta t)\delta\theta_{KF_t} \\ &\quad - \delta\mathbf{b}_{g,KF_t}\Delta t + \eta_\theta, \delta\mathbf{b}_{g,KF_{t+1}} = \delta\mathbf{b}_{g,KF_t} + \eta_g, \end{aligned} \quad (1)$$

where  $\omega_m$  is the gyroscope measurement measured by IMU,  $\delta\theta_{KF_t}$  and  $\delta\mathbf{b}_{g,KF_t}$  are error state of keyframe  $KF_t$ . Furthermore,  $\eta_\theta$  and  $\eta_g$  are white Gaussian noise applied to rotation and gyroscope bias estimation respectively,  $\eta_\theta \sim \mathcal{N}(0, \sigma_{w_n}^2 \Delta t^2 \mathbf{I})$ ,  $\eta_g \sim \mathcal{N}(0, \sigma_{w_u}^2 \Delta t \mathbf{I})$ . Then, the covariance matrix of the prediction at  $KF_{t+1}$  can be updated by Equ. 2:

$$\mathbf{P}_{\text{pred}} = \mathbf{F} \hat{\mathbf{P}}_{KF_t} \mathbf{F}^\top + \mathbf{Q}, \quad (2)$$

where  $\mathbf{P}_{\text{pred}}$  represents the predicted covariance matrix for keyframe  $KF_{t+1}$ , while  $\hat{\mathbf{P}}_{KF_t}$  denotes the corrected covariance matrix from the previous keyframe  $KF_t$ . The matrix  $\mathbf{Q}$ , defined as  $(\mathbf{Q} = \text{diag}(\text{Cov}(\eta_\theta), \text{Cov}(\eta_g)))$ , encapsulates the

covariance of the perturbation impulses. Additionally,  $\mathbf{F}$  refers to the Jacobian matrix relative to the error state of keyframe  $KF_t$ , as shown in Equ. 1 and Equ. 2.

$$\mathbf{F} = \begin{bmatrix} \text{Exp}(-(\boldsymbol{\omega}_m - \mathbf{b}_g)\Delta t) & -I\Delta t \\ 0 & I \end{bmatrix} \quad (3)$$

In order to get the update of the error state, we need to calculate the Kalman gain as shown in Equ. 4:

$$\begin{aligned} \mathbf{K} &= \mathbf{P}_{\text{pred}} \mathbf{H}^\top (\mathbf{H} \mathbf{P}_{\text{pred}} \mathbf{H}^\top + \mathbf{V})^{-1}, \\ \delta \hat{\mathcal{X}}_{KF_{t+1}} &= \mathbf{K}(\log(\mathbf{e}_r)), \\ \hat{\mathbf{P}}_{KF_{t+1}} &= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_{\text{pred}}. \end{aligned} \quad (4)$$

where  $\mathbf{e}_r = \log(\mathbf{h}(\mathcal{X}_t)^\top \mathbf{r}_{KF_{t+1}}^w)$  represents the orientation difference between the predicted state and observed state. Specifically,  $\mathbf{r}_{KF_{t+1}}^w = \mathbf{h}(\mathcal{X}_t) + \mathbf{v}$  is an observation of orientation at keyframe  $KF_{t+1}$ , where  $\mathbf{h}()$  is the observation function of the system, and  $\mathbf{v}$  denotes white Gaussian noise with covariance  $\mathbf{V}$ , such that  $\mathbf{v} \sim \mathcal{N}(0, \mathbf{V})$ . Additionally, the Jacobian matrix  $\mathbf{H}$  is defined as the matrix of partial derivatives of the observation function with respect to the error state  $\delta \mathcal{X}$ .

Following the update via the ESKF, the posterior error state, denoted as  $\delta \hat{\mathcal{X}}_{KF_{t+1}}$ , is integrated into the nominal state. Subsequently,  $\delta \hat{\mathcal{X}}_{KF_{t+1}}$  is reset to zero, and its corresponding covariance matrix is adjusted to accurately represent this reset. The optimal estimate of the true state at keyframe  $KF_{t+1}$  is then derived using the necessary compositions as follows in Equ. 5:

$$\begin{aligned} \hat{\mathbf{R}}_{KF_{t+1}}^w &= \mathbf{R}_{KF_{t+1}}^w \text{Exp}(\delta \hat{\theta}_{KF_{t+1}}), \\ \hat{\mathbf{b}}_{g,KF_{t+1}} &= \mathbf{b}_{g,KF_{t+1}} + \delta \hat{\mathbf{b}}_{g,KF_{t+1}}. \end{aligned} \quad (5)$$

Initially, the gyroscope bias is presumed to be zero. Employing the ESKF over  $N$  keyframes, we determine that the gyroscope bias calculated at the final keyframe of the sequence is the most accurate representation of the true state. Consequently, this value is finalized as the true gyroscope bias. Utilizing this approach ensures precise estimation of rotation, enabling the use of a three-degree-of-freedom (3DoF) bundle adjustment (BA) to compute the translation  $\mathbf{t}_{KF_{t+1}}^w$  as outlined in Equ. 6:

$$\begin{aligned} \mathbf{t}_{KF_{t+1}}^w &* \\ &= \arg \min_{\mathbf{t}_w} \sum_{i \in \mathcal{M}} \rho \left( \|\mathbf{x}^i - \pi(\mathbf{R}_{KF_{t+1}}^w \mathbf{X}^i + \mathbf{t}_{KF_{t+1}}^w)\|^2 \right) \end{aligned} \quad (6)$$

where  $\mathbf{X}^i$  represents a 3D point in the world frame, and  $\mathbf{x}^i$  denotes its corresponding 2D feature. The function  $\rho$  represents the robust Huber cost function, while  $\pi$  is the reprojection function.

This ESKF-based pipeline facilitates the estimation of rotation and translation at each timestamp, thereby providing precise positioning information crucial for our robotics-based data collection system. As mentioned in Section. III-B, By synchronously tagging GPR measurements and visual inspection images with accurate poses on the fly, the system allows the robot to scan the ground surface in a free-motion pattern, enhancing the effectiveness of inspections for both surface and subsurface defects.

### B. Learning-Based Visual Inspection Module

In the process of automated defect detection within RGB images, we employ a learning-based algorithm following the robotics-based data collection module. The architecture of our learning-based method, referred to as InspectionNet++, is depicted in Fig. 4, tailored specifically to enhance the detection and segmentation of defects.

InspectionNet++ is architecturally engineered with a modified ResNet backbone [15], originally inspired by the renowned VGG-16 model [34]. This backbone is adapted to incorporate depth channels alongside the standard RGB input, enabling a more comprehensive analysis of the image data. The model's backbone is instrumental in extracting high-level features and is carefully calibrated to maintain the high accuracy characteristic of VGG-16 in large-scale image recognition tasks, while also accommodating the additional depth information. Further enhancing its capability, InspectionNet++ employs a Pyramid Pooling (PSP) module [47], which has been refined to perform depth-aware feature pooling. This module enhances the model's ability to handle spatial hierarchies and variations in scene composition. In addition, the feature pyramid network (FPN) [24] within InspectionNet++ is designed to fuse multi-scale features effectively, leveraging a hierarchical structure that allows for a detailed and nuanced understanding of both textural and depth cues within the RGB-D images. This fusion process is crucial for producing a segmented output that accurately localizes defects across varied image scales and depths. The final stages of InspectionNet++ consist of dedicated heads for segmentation and classification, designed to process the fused features to precisely localize and identify different types of defects within the RGB-D images. These heads use convolutional layers followed by interpolation to ensure that the output maps align perfectly with the input dimensions, maintaining spatial accuracy in the segmentation process.

*Model Training and Loss Function:* The InspectionNet++ carries out class-aware segmentation on RGB images, categorizing each pixel into one of three classes: **crack**, **spall**, and **stain**, with a probabilistic prediction for each class label. For each label,  $P(u, v) = p(u_i, v_i) | p(u_i, v_i, j)$ , where  $i$  denotes the pixel index in the image, and  $j$  denotes the class label. To aid civil engineers in recognizing the inspection results, we overlay the defect flaw areas on the original RGB image as follows: 1) background with original RGB color; 2) spall defects in blue; 3) crack defects in green; and 4) stain defects in red. Given that spall and stain defects typically cover larger areas and encompass more pixels than cracks, the proposed model is prone to overfitting. To address this, we employed a loss strategy as suggested in [30].

$$\text{loss} = \sum_{x_i \in X} w^j(x) \log(p(x_{i,j})) \quad (7)$$

where  $x_i$  represents a pixel in the image  $X$ ,  $p(x_{i,j})$  denotes the probabilistic prediction of pixel  $x_i$  for class  $j$ , and  $w^j$  signifies the weight assigned to each class. This paper explores the impact of class weights on segmentation outcomes, specifically examining how these weights influence the accuracy and effectiveness of the segmentation process.

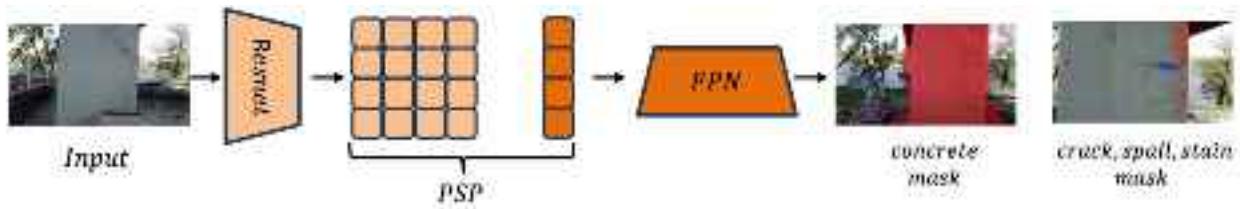


Fig. 4. The InspectionNet++ model consists of 3 modules: ResNet as the backbone for foundational feature extraction, PSPNet for multi-scale contextual information aggregation, and FPN for effective feature fusion across different scales. The output shows its ability to segment concrete, crack, and stain features.

Nevertheless, in our training process, it was noted that masks were frequently detected against the sky and tree branches, which raised a challenge for accurate defect segmentation. To refine the detection capabilities and specifically improve the isolation of relevant defects, we developed a dedicated network called ConcreteNet, described in Fig. 4. This network was trained using the Inspection++ architecture on a very small dataset comprising 50 labeled images. The primary motivation behind implementing ConcreteNet was to effectively filter out irrelevant background elements, thereby enhancing the detection of other critical defects within the images. The masks produced by ConcreteNet are crucial for distinguishing actual concrete areas from non-concrete components, thus providing a cleaner and more accurate base for subsequent defect detection processes.

### C. Concrete Defects Metric Measurements and Mapping

As depicted in Fig. 5, the original image undergoes processing through InspectionNet++ to produce a crack segmentation mask, displayed in the second column with cracks highlighted in red on a 3-channel image. This color mask is subsequently transformed into a single-channel binary image (0: black; 255: white) and a skeleton is extracted. Consequently, the cracks are represented as 1-pixel-width lines, as illustrated in the rightmost column. The final step involves calculating the dimensions of the 8-connected regions, which means from any pixel within a region, one can navigate to any other pixel via 8 possible directions: up, down, left, right, up-left, up-right, down-left, down-right. As illustrated in Fig. 6, the bottom left image showcases a binary skeleton with the cracks appearing as white lines, while the top right features a pixel-level zoom of the lower image. In this detailed view, 255 denotes the white line and 0 the black background. Utilizing the 8-connection extraction method from OpenCV, we were able to determine the number of 8-connected regions and the area of each, summing these areas provides the total length of the cracks in pixels.

Furthermore, to accurately convert pixel-based measurements into a metric scale, we utilize depth data along with known parameters, such as the fixed position of the camera on the robot and its distance from the ground, as introduced in Section III-A. By converting pixel length measurements into meters, we gain detailed and actionable insights into the structural integrity of the inspected surfaces, supporting effective intervention strategies.

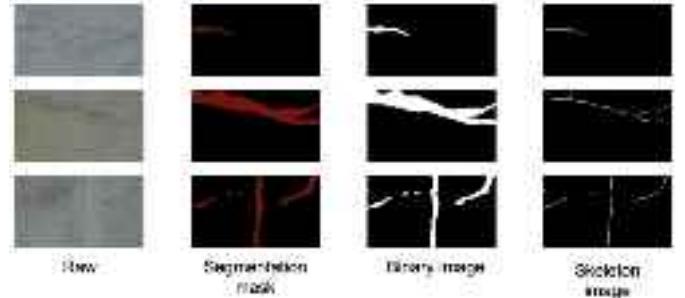


Fig. 5. Crack measurement for 2D images.

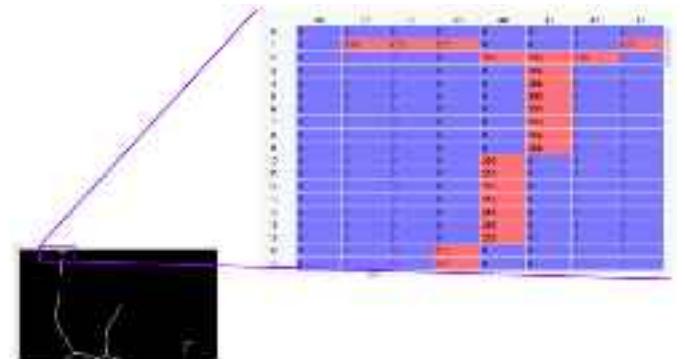


Fig. 6. Calculating crack length by 8-connection area size.

## V. SUBSURFACE DEFECTS INSPECTION AND MAPPING

In this section, we present an approach to inspect subsurface defects and generate subsurface condition maps using GPR. Our method focuses on detecting and characterizing subsurface defects, such as voids, cracks, and corrosion in structures. By utilizing GPR data, we create condition maps of the inspected area and subsequently use perspective projection to map the GPR condition map onto an RGB image. This process enables us to visualize the subsurface defects in the context of the inspected structure, providing an intuitive understanding of the subsurface condition.

### A. GPR Data Collection and Processing

The first step in subsurface defects inspection involves acquiring high-quality GPR data. As introduced in Section VI-B, our proposed robotic system is employed to collect GPR B-scan data.

After collecting the GPR data, we applied preprocessing techniques to enhance the quality of the data and remove any unwanted noise. Specifically, preprocessing steps including

filtering, gain control, and background removal are implemented in this paper. These techniques help improve the signal-to-noise ratio (SNR) of the GPR data, making it easier to detect and characterize subsurface defects.

### B. Generating Condition Maps

After preprocessing the GPR data, which involves converting reflection amplitudes to decibels and applying depth correction, we generate condition maps to illustrate subsurface defects. The conversion of amplitudes to decibels enhances data visualization, and depth correction improves the accuracy of subsurface feature localization.

Amplitude is converted to decibels using the Equ. 8:

$$A_{dB} = 20 \times \log_{10}(A) \quad (8)$$

Depth correction is achieved through Equ. 9:

$$D = \frac{V \times TWT}{2} \quad (9)$$

where  $V$  is the velocity of the signal in the material, and  $TWT$  is the two-way travel time.

Condition maps are then created by analyzing these adjusted values to detect and depict subsurface anomalies in a 2D or 3D format, representing defect severity or depth at specific locations.

### C. Mapping GPR Condition Maps Onto RGB Images

To facilitate the understanding of subsurface defects relative to visible structures, we employ perspective projection to overlay GPR condition maps onto RGB images. This process involves several key steps:

- Utilizing ROS synchronization functions, as detailed in [11], we synchronize GPR sampling with positioning data introduced in Section IV-A to ensure both the GPR condition map and RGB image share a common coordinate system.
- Apply perspective transformation using a homography matrix, which aligns the GPR condition map to correspond accurately with the RGB image. This matrix transforms coordinates from the GPR condition map to the RGB image, facilitating precise alignment and overlay.
- The transformed GPR map is then overlaid onto the RGB image using a blending formula,  $I_{composite} = \alpha I_{GPR} + (1 - \alpha)I_{RGB}$ , where  $\alpha$  adjusts the transparency. This composite image merges subsurface data with surface visuals, offering a comprehensive view that enhances defect assessment and maintenance planning.

This streamlined approach not only simplifies the visualization of subsurface defects in the context of structural surfaces but also aids in more effective structural assessments. The evaluation of this pipeline can be found in Section VI-E.

## VI. EXPERIMENTAL STUDIES

In this section, we first present the dataset used to train InspectionNet++, and then introduce an iPad App that was

developed for visual inspection online inference and robot motion control. Subsequently, we conduct a comprehensive evaluation of InspectionNet++ through both qualitative and quantitative analysis. Additionally, we discuss a visualization tool designed to enhance the comprehension of metric measurements. At last, we present the integration of surface and subsurface mapping results, synthesizing the findings from the multi-sensor inspection process.

### A. Data Preparation

To train the InspectionNet++, we utilized a robust dataset comprising 2250 raw images, which were collected from multiple field testing scenarios, including: 1) 24 GB of drone data from Stantec's inspection of the Bayonne Bridge in New Jersey, and 2) video imagery from a wall-climbing robot inspecting the bridge pier of Congress Street Bridge in Troy, NY, conducted under the oversight of the New York Department of Transportation (NYDOT). This dataset is supplemented by the data introduced in [43], enhancing our model's training resources. The images in our dataset are meticulously categorized and labeled according to the types of defects they display, with a total of 3070 labels. Specifically, the dataset includes 1630 instances of cracks, 1120 instances of spalls, and 320 instances of stains. Each image was carefully annotated to identify three primary types of structural defects—cracks, spalls, and stains—ensuring high-quality training data for InspectionNet++. The labeling process was conducted with rigorous attention to detail and subsequently subjected to a peer review process to minimize any errors and inconsistencies, thus maintaining high data integrity and reliability for effective model training.

Prior to feeding the data into InspectionNet++, each image underwent a series of preprocessing steps. These steps included normalization to standardize the intensity values and resizing images to meet the input size requirements of the network. The model was trained on a desktop PC equipped with a 3.2GHz Intel Core i9-9900K CPU with 32GB RAM and a GeForce RTX 2080 Ti GPU. In all experiments, the input resolution of each image is  $w, h = 640, 640$ , and our model is trained for 200 epochs with an 0.01 initial learning rate. We use Root Mean Squared Propagation (RMSprop) optimizer without the momentum to update the learning rate and the weight decay is set to 0.00006 for every 1000 iterations.

### B. iPad Application Development

Implementing a learning-based approach for visual inspection and using terminal commands to control the robot can be both time-consuming and impractical for real-world use by civil engineers. Thus, to better assist onsite construction engineering, we developed an iPad App interface that enables easy management of online visual inspection results, as well as robot remote motion control.

1) *InspectionNet++ Online Inference*: We first deploy our learning-based model with Amazon Web Services (AWS) to enable online inference for the collected defect images. This allows construction engineers to easily manage visual inspection results through AWS.



Fig. 7. An illustration of the AWS deployment for visual inspection.

As shown in Fig. 7, we provide an overview of the AWS deployment for visual inspection, which consists of four steps. First, we create a Docker image [3] for InspectionNet++, encapsulating the application and its dependencies in a portable container. Second, we set up input and output buckets using Amazon S3, a highly scalable and durable object storage service. These buckets facilitate the storage and retrieval of input data and processed results. Third, we utilize AWS Lambda, a serverless compute service, to run the Docker image, enabling the execution of InspectionNet++ without the need to manage any underlying infrastructure. Finally, we implement a user notification system to inform users about the completion of processing tasks and the availability of results.

### 2) Remote Control for Robotic-Based Data Collection:

We then utilize the developed iPad app to facilitate wireless communication with high-level and low-level controllers for remote management of the robot's data collection and movement. This functionality enables flexible, non-linear motion patterns during data collection.

As depicted in Fig. 9, there is a toggle button in the top right corner that activates the camera view, offering users a first-person perspective of the robot. Users can enable RGB data, depth data, and GPR data collection by toggling the buttons beneath the camera view. Additionally, once data is collected, information such as the size of the collected data and available storage space is displayed for the user.

On the left side of the app interface, users can choose between manual and automatic control modes. By toggling the manual area scan button, users can press arrow buttons to engage manual control, allowing the robot to move freely without adhering to a predetermined pattern. Conversely, by enabling automatic control, users can toggle the auto area scan button to enable an automatic zigzag motion pattern. To initiate automatic data collection, users must configure the scan area's length and width, as well as the line spacing. The app also allows users to adjust the robot's speed, with the default speed set to 0.5 m/s.

Moreover, Fig. 8 illustrates the file management interface of the developed iPad app. The left column presents the collected color images arranged in chronological order. When the iPad or the robot is connected to the internet, these raw color images are automatically uploaded to the AWS server for segmentation post-processing. Then, the right column displays the processed defect-highlighted segmentation images, which are automatically downloaded when the internet is available. Additionally, users can click the 'Clear' button at the bottom to remove any images.



Fig. 8. Developed iPad app - file management interface.



Fig. 9. Developed iPad app - robot motion control interface.

**3) Annotation Interface for Unseen Defects:** As illustrated in the user interface shown in Fig. 12, our system offers a robust mechanism for manual annotations to address cases where defects might not be automatically detected or are falsely identified. Specifically, the proposed InspectionNet++ detects cracks, spalling, and stains in Fig. 12 using green, blue, and red masks, respectively. To further enhance the quality of the visual inspection, we manually annotate any undetected defects with a yellow mask. This manual annotation is facilitated through an intuitive interface that allows users to directly annotate unseen defects on the image.

This manual annotation capability ensures that our system remains adaptable and accurate, even in complex scenarios where automatic detection may falter. By allowing for human oversight, we significantly enhance the reliability of defect detection and classification, thereby reducing false positives and improving the overall robustness of our inspection process.

### C. InspectionNet++ Performance Evaluation

In this section, the quantitative performance of InspectionNet++ are first compared against other state-of-the-art methods, including U-Net, FCN-8s, DeepSegmentor [25], [26], and crackseg9k [21], as illustrated in Table I. The evaluation encompasses three critical metrics: F1-score (F1),

TABLE I  
DETECTION PERFORMANCE COMPARISON

	InspectionNet			U-Net			FCN-8s			DeepSegmentor			crackseg9k		
	Crack	spalls	stain	Crack	spalls	stain	Crack	spalls	stain	Crack	spalls	stain	Crack	spalls	stain
F1	82.40	88.64	89.81	76.76	88.72	80.96	-	88.68	83.28	79.53	88.90	87.62	81.24	89.35	88.45
AP	83.59	91.69	90.32	80.96	91.71	85.18	-	91.54	88.37	80.12	90.88	91.28	82.41	92.10	90.31
AR	81.24	85.79	89.31	72.97	85.92	77.14	-	85.23	78.74	78.95	87.00	84.24	80.10	86.76	86.67

Fig. 10. Qualitative analysis of InspectionNet++. The first row illustrates results from crack segmentation, the second row displays results from spall segmentation, and the third row depicts results from stain segmentation.

Average Precision (AP), and Average Recall (AR), for crack, spall, and stain segmentation.

From the F1-score analysis, InspectionNet++ demonstrates robust performance with 82.40 for crack, 88.64 for spalls, and 89.91 for stain. These values indicate a favorable balance between precision and recall, outperforming U-Net and DeepSegmentor, and showing competitive results against the crackseg9k model.

The AP comparison study further accentuates InspectionNet++'s ability to provide a precise prediction, scoring 83.59, 91.69, 90.32 for crack, spalls, and stain respectively. While crackseg9k's performance is slightly inferior, the variations between U-Net and DeepSegmentor are worth noting. These discrepancies may reflect underlying differences in their configuration or training data.

For AR comparisons, InspectionNet++ posts values of 81.24, 85.79, and 89.31 for crack, spalls, and stain categories. These values underscore the system's capability to identify most of the relevant defects without substantial oversight, contrasting with lower scores from U-Net, which posts 72.97 for cracks, 85.92 for spalls, and 77.14 for stains. FCN-8s did not provide data for cracks, which may indicate limitations in segmenting more narrowly defined categories.

Additionally, we also present the qualitative results of InspectionNet++ in Fig. 10. This visualization underscores the segmentation results of the proposed method across diverse construction defects, including cracks, spalls, and stains. We use three different colors to overlay each of the categories:

green for crack, blue for spall, and red for stain. It is evident that InspectionNet++ successfully identifies the presence of each defect category within the structural components.

Another important observation is that the inherent properties of these defects make the task of crack segmentation particularly challenging. This conclusion is also supported by the data presented in Table I. To further validate the effectiveness of crack segmentation within our construction structure visual inspection system, we carried out a comprehensive set of experiments in real-world settings. The results of these tests are depicted in Fig. 11, where the cracks are distinctly segmented and highlighted with a green overlay, demonstrating the system's ability to accurately identify these structural flaws.

#### D. Visualization of Metric Measurements

Implementing a learning-based visual inspection approach, while beneficial for automating defect detection, can often be time-intensive and impractical for deployment in real-world scenarios that are routinely encountered by civil engineers. To mitigate these operational challenges and enhance the utility of the system for on-site engineers, we have deployed the proposed InspectionNet++ on a GPU server. Furthermore, we utilize the Potree Viewer [32] to facilitate the efficient visualization of results within a 3D environment after the data processing is complete.

The proposed method has a series of scripts to automate the processing from raw images to 3D point clouds with defect

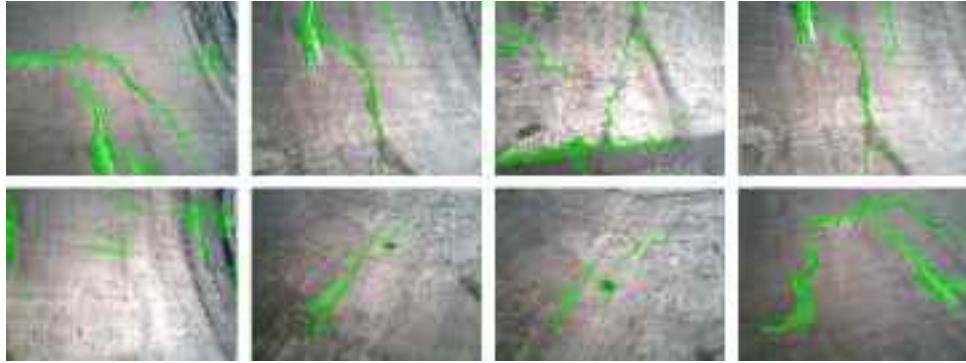


Fig. 11. More field test results of crack segmentation.



Fig. 12. A user-friendly interface to manually annotation undetected defects in an image.



Fig. 13. Potree viewer displays the 3D point cloud with highlighted cracks and showcases the metric measurement of the crack defects.

overlays. Initially, the raw data undergo defect detection and a concrete detector, where three types of defects are identified. The detected defect masks are filtered by the concrete mask

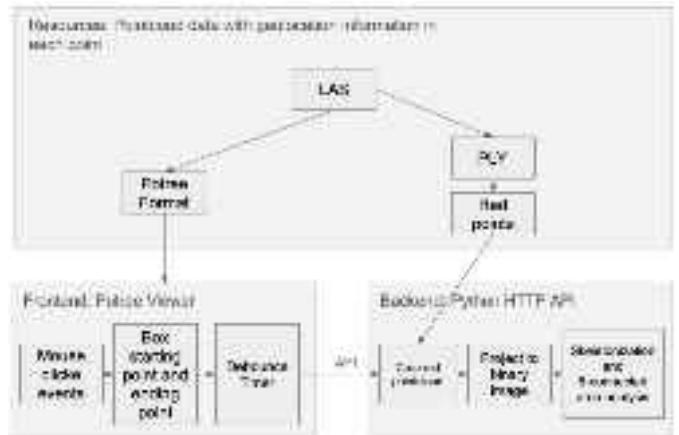


Fig. 14. Architecture for crack metric measurement for the selected area in Potree viewer.

and then overlaid to the raw images, and essential geospatial metadata such as GPS coordinates and velocity data are integrated into these images for 3D reconstruction purposes. The 3D reconstruction module would create 3D point clouds for display in the Potree Viewer. Accessibility to our GPU server is facilitated through a public-facing FRP reverse server,<sup>1</sup> enabling both development via SSH and, in later stages, user access to HTTP(s) services and APIs externally.

As shown in Fig. 13, the Potree Viewer displays the 3D model with cracks highlighted in red. It is noted that a sidebar button has been implemented, enabling users to select an interested area for crack metric measurements. Once users define an interested box area and the box remains stable for 500 ms, the frontend issues an HTTP request. This triggers our backend Python script to segment the selected point cloud, project it into a 2D binary image, and execute the 2D crack measurement procedure to determine the crack length. At last, the measured crack length will be displayed on the left side of the GUI.

The interplay between the frontend and backend processes is illustrated in Fig. 14. This deployment of the metric measurement interface ensures that operations can be conducted efficiently and reliably, thereby delivering a smooth user experience to civil engineers.

<sup>1</sup><https://github.com/fatedier/frp>

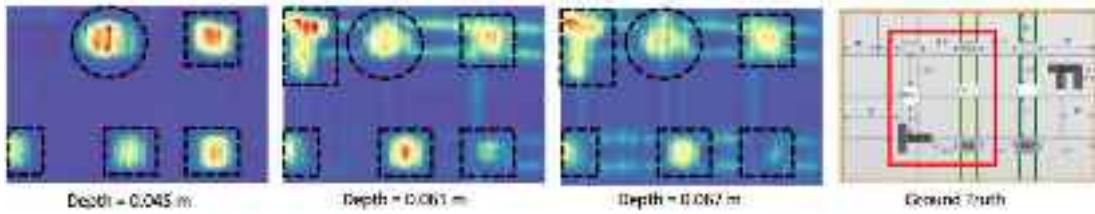


Fig. 15. GPR defect mapping: The GPR system scans the area within the red rectangle, effectively detecting various subsurface defects, including rebars and T-shaped metal elements.

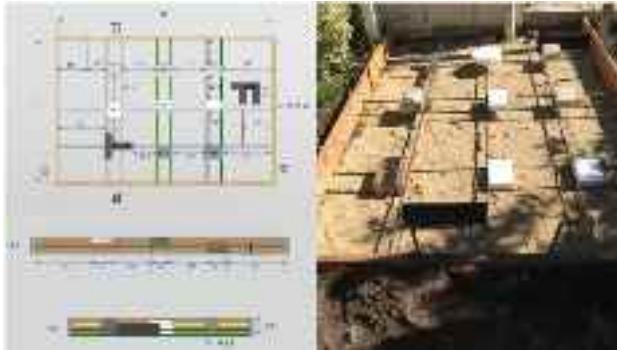


Fig. 16. GPR features overlay on field test results.

#### E. Comprehensive Map Visualization for Surface and Subsurface

As for the subsurface defect mapping, the CCNY Robotics Lab Testing Pit provided an ideal setting for intensive study. This testing pit, with its specially designed concrete slab embedded with artificially constructed defects, offered a unique vantage point for our tests. The designed blueprint of the slab against its pre-concrete pouring visual is captured in Fig. 16 (a), while the GPR imaging, derived using the BP method, precisely captured the intricate anomalies beneath the



Fig. 17. Surface defects and subsurface feature map visualization on the filed.

slab's surface, showcasing the embedded objects and other defects in Fig. 16 (b).

More specifically, GPR data were collected over the red rectangular area of slab as shown in Fig. 15. We show the different GPR defect detection results of the slab, which distinctly revealed the embedded rebars and T-shaped metal elements at different depths. Such precision in capturing subsurface features, coupled with visual inspections, offers a detailed map of both the seen and unseen.

At last, we integrated surface defect map and subsurface condition map to create a layered and comprehensive depiction of the construction's structural health. The experiments were carried out on the CCNY campus. As illustrated in Fig. 17, GPR condition image and visual inspection image are overlaid on the field, providing instant combined insights into the structural integrity from both surface and subsurface perspectives. This integration is enhanced by visual inspection images that compare surface-level anomalies with actual onsite images, offering exceptional clarity into the identification of surface defects and subsurface characteristics.

#### F. VI-SLAM Accuracy Evaluation

In this subsection, we evaluate the accuracy of our proposed monocular VI-SLAM method in comparison to ORB-SLAM3 [6]. The evaluation is conducted on a set of benchmark sequences from the EuRoC MAV dataset [5], which includes various indoor environments with different levels of complexity and difficulty.

Table II presents the Absolute Trajectory Error (ATE) and Relative Pose Error (RPE) in degrees for both our method and ORB-SLAM3, with and without Visual-Inertial Bundle Adjustment (VI-BA). The ATE measures the difference between the estimated trajectory and the ground truth, reflecting the global consistency of the estimated path. The RPE

TABLE II  
COMPARISON OF SLAM PERFORMANCE METRICS ACROSS DIFFERENT SEQUENCES OVER EUROC DATASET

Seq. Name	ATE (m)				RPE (deg)			
	Ours W/O VI-BA	ORB-SLAM3 W/O VI-BA	Ours W/ VI-BA	ORB-SLAM3 W/ VI-BA	Ours W/O VI-BA	ORB-SLAM3 W/O VI-BA	Ours W/ VI-BA	ORB-SLAM3 W/ VI-BA
MH01_easy	<b>0.145</b>	0.172	<b>0.145</b>	0.202	<b>0.123</b>	0.118	<b>0.123</b>	0.082
MH02_easy	<b>0.114</b>	0.164	<b>0.095</b>	0.164	<b>0.128</b>	0.115	<b>0.164</b>	0.115
MH03_medium	<b>0.310</b>	0.353	<b>0.246</b>	0.353	<b>0.254</b>	0.216	<b>0.124</b>	0.216
MH04_difficult	<b>0.312</b>	0.386	<b>0.216</b>	0.386	<b>0.210</b>	0.199	<b>0.134</b>	0.199
MH05_difficult	<b>0.239</b>	0.408	<b>0.141</b>	0.408	<b>0.143</b>	0.201	<b>0.081</b>	0.201
V1_01_easy	<b>0.069</b>	0.099	<b>0.055</b>	0.099	<b>0.234</b>	0.246	<b>0.155</b>	0.246
V1_02_medium	<b>0.187</b>	0.229	<b>0.120</b>	0.229	<b>0.367</b>	0.318	<b>0.137</b>	0.318
V1_03_difficult	<b>0.124</b>	0.153	<b>0.082</b>	0.153	<b>0.655</b>	0.472	<b>0.417</b>	0.472
V2_01_easy	<b>0.075</b>	0.088	<b>0.075</b>	0.080	<b>0.339</b>	0.488	<b>0.339</b>	0.436
V2_02_medium	<b>0.116</b>	0.155	<b>0.116</b>	0.128	<b>0.312</b>	0.407	<b>0.312</b>	0.157
V2_03_difficult	<b>0.133</b>	0.150	<b>0.109</b>	0.150	<b>0.633</b>	0.837	<b>0.493</b>	0.837
Avg	<b>0.166</b>	0.214	<b>0.127</b>	0.214	<b>0.309</b>	0.329	<b>0.225</b>	0.298

evaluates the local accuracy of the estimated trajectory over a fixed time interval, focusing on the relative motion between frames.

Our method consistently outperforms ORB-SLAM3 across all sequences in terms of both ATE and RPE. Without VI-BA, our method achieves an average ATE of 0.166(m), compared to 0.214(m) for ORB-SLAM3. With VI-BA enabled, the average ATE of our method further improves to 0.127(m), while ORB-SLAM3 remains at 0.214(m) due to its limited utilization of visual-inertial optimization in monocular configurations.

In terms of rotational accuracy, our method demonstrates a lower average RPE of 0.309° without VI-BA, compared to 0.329° for ORB-SLAM3. The inclusion of VI-BA reduces the average RPE of our method to 0.225°, indicating a significant improvement in rotational estimation. ORB-SLAM3 shows a marginal improvement with an average RPE of 0.298°, which still lags behind our method.

The performance gains of our method are particularly notable in challenging sequences such as *V103 difficult* and *V203 difficult*. For instance, in *V103 difficult*, our method achieves an ATE of 0.082(m) with VI-BA, substantially lower than the 0.153(m) achieved by ORB-SLAM3. Similarly, the RPE for our method is reduced to 0.417°, whereas ORB-SLAM3 records 0.472°.

The superior performance of our monocular VI-SLAM method can be attributed to the robotics system and provide the accurate georeference information to tag with the collected GPR data. This integration is critical for achieving reliable and high-quality mapping results.

## VII. CONCLUSION

As critical infrastructure systems around the world continue to age and deteriorate, effective assessment and maintenance have become imperative needs. However, traditional manual inspection processes are tedious, costly, subjective, and lack comprehensive analysis. This research aims to transform infrastructure inspection through an automated robotic platform and data-driven analytics.

The proposed method provides several improvements towards intelligent construction inspection. Firstly, the robotic

system enables flexible and efficient data collection over large areas without being restricted to linear paths. The integration of GPR and vision sensors within a single platform delivers both subsurface and surface sensing capabilities necessary for holistic inspection. Secondly, the learning-based method provides accurate pixel-level segmentation and categorization of concrete defects under real-world conditions. This represents a major advance over previous hand-crafted or traditional machine learning approaches. Thirdly, the integrated 3D visualization software generates detailed spatial localization and metric measurements of defects. This facilitates precise quantitative assessment rather than just visual detection. Finally, the intuitive integrated visualization correlated GPR-detected subsurface anomalies and surface defects mapped onto structural images delivers an easily understandable perspective for decision-makers.

In the future, we aim to improve the proposed network by integrating transformers as a replacement for CNNs. Additionally, we intend to develop automated techniques to filter out concrete joints based on the intermediate depth data generated during the 3D reconstruction phase.

## ACKNOWLEDGMENT

The views, opinions, findings and conclusions reflected in this publication are solely those of the authors and do not represent the official policy or position of the USDOT/OST-R, or any state or other entity.

## REFERENCES

- [1] I. O. Agyemang et al., "Automated vision-based structural health inspection and assessment for post-construction civil infrastructure," *Autom. Construct.*, vol. 156, Dec. 2023, Art. no. 105153.
- [2] D. Ai, G. Jiang, S.-K. Lam, P. He, and C. Li, "Computer vision framework for crack detection of civil infrastructure—A review," *Eng. Appl. Artif. Intell.*, vol. 117, Jan. 2023, Art. no. 105478.
- [3] C. Anderson, "Docker [software engineering]," *IEEE Softw.*, vol. 32, no. 3, pp. 10–2, May 2015.
- [4] T. Askarzadeh, R. Bridgelall, and D. D. Tolliver, "Systematic literature review of drone utility in railway condition monitoring," *J. Transp. Eng., A, Syst.*, vol. 149, no. 6, Jun. 2023, Art. no. 04023041.
- [5] M. Burri et al., "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, Sep. 2016.

- [6] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021.
- [7] D. Baek et al., "Battery-aware operation range estimation for terrestrial and aerial electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5471–5482, Jun. 2019.
- [8] T. H. Dinh et al., "Toward vision-based concrete crack detection: Automatic simulation of real-world cracks," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2023.
- [9] J. Feng, L. Yang, E. Hoxha, B. Jiang, and J. Xiao, "Robotic inspection of underground utilities for construction survey using a ground penetrating radar," *J. Comput. Civil Eng.*, vol. 37, no. 1, Jan. 2023, Art. no. 04022049.
- [10] J. Feng, L. Yang, E. Hoxha, D. Sanakov, S. Sotnikov, and J. Xiao, "GPR-based model reconstruction system for underground utilities using GPRNet," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 845–851.
- [11] J. Feng, L. Yang, E. Hoxha, and J. Xiao, "Improving 3D metric GPR imaging using automated data collection and learning-based processing," *IEEE Sensors J.*, vol. 23, no. 5, pp. 4527–4539, Mar. 2023.
- [12] J. Feng, L. Yang, H. Wang, Y. Song, and J. Xiao, "GPR-based subsurface object detection and reconstruction using random motion and DepthNet," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 7035–7041.
- [13] J. Feng, L. Yang, H. Wang, Y. Tian, and J. Xiao, "Subsurface pipes detection using DNN-based back projection on GPR data," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 266–275.
- [14] J. Feng, L. Yang, and J. Xiao, "Subsurface object 3D modeling based on ground penetration radar using deep neural network," *J. Comput. Civil Eng.*, vol. 37, no. 6, Nov. 2023, Art. no. 04023030.
- [15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [16] E. Hoxha, J. Feng, D. Sanakov, A. Gjinofci, and J. Xiao, "Robotic inspection and characterization of subsurface defects on concrete structures using impact sounding," 2022, *arXiv:2208.06305*.
- [17] E. Hoxha, J. Feng, D. Sanakov, and J. Xiao, "Robotic inspection and subsurface defect mapping using impact-echo and ground penetrating radar," *IEEE Robot. Autom. Lett.*, vol. 8, no. 8, pp. 4943–4950, Aug. 2023.
- [18] E. Hoxha et al., "Contrastive learning for robust defect mapping in concrete slabs using impact echo," *Construct. Building Mater.*, vol. 461, Jan. 2025, Art. no. 139829.
- [19] R. Hu, W. Pan, K. Iturralde, T. Linner, and T. Bock, "Construction automation and robotics for concrete construction: Case studies on research, development, and innovations," in *Proc. Int. Symp. Autom. Robot. Construct. (IAARC)*, vol. 40, Jul. 2023, pp. 683–690.
- [20] S. Ivić, B. Crnković, L. Grbčić, and L. Matleković, "Multi-UAV trajectory planning for 3D visual inspection of complex structures," *Autom. Construction*, vol. 147, Mar. 2023, Art. no. 104709.
- [21] S. Kulkarni, S. Singh, D. Balakrishnan, S. Sharma, S. Devunuri, and S. C. R. Korlapati, "CrackSeg9k: A collection and benchmark for crack segmentation datasets and frameworks," in *Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer*, Jan. 2023, pp. 179–195.
- [22] A. J. Lee, W. Song, B. Yu, D. Choi, C. Tirtawardhana, and H. Myung, "Survey of robotics technologies for civil infrastructure inspection," *J. Infrastructure Intell. Resilience*, vol. 2, no. 1, Mar. 2023, Art. no. 100018.
- [23] M. Lehman, "The American society of civil engineers' report card on america's infrastructure," in *Women in Infrastructure*. Cham, Switzerland: Springer, 2022, pp. 5–21.
- [24] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.
- [25] Y. Liu, J. Yao, X. Lu, M. Xia, X. Wang, and Y. Liu, "RoadNet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2043–2056, Apr. 2019.
- [26] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation," *Neurocomputing*, vol. 338, pp. 139–153, Apr. 2019.
- [27] S. F. B. A. Manaf, "Utilizing deep learning for automated inspection and damage assessment in civil infrastructure systems," *Appl. Res. Artif. Intell. Cloud Comput.*, vol. 7, no. 6, pp. 1–10, 2024.
- [28] S. Nansai and R. Mohan, "A survey of wall climbing robots: Recent advances and challenges," *Robotics*, vol. 5, no. 3, p. 14, Jul. 2016.
- [29] P. Prasanna et al., "Automated crack detection on concrete bridges," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 591–599, Apr. 2016.
- [30] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *IEEE Trans. Med. Imag.*, vol. 35, no. 1, pp. 234–241, Jan. 2015.
- [31] P. Savino and F. Tondolo, "Civil infrastructure defect assessment using pixel-wise segmentation based on deep learning," *J. Civil Struct. Health Monitor*, vol. 13, no. 1, pp. 35–48, Jan. 2023.
- [32] M. Schütz, S. Ohrhallinger, and M. Wimmer, "Fast out-of-core octree generation for massive point clouds," *Comput. Graph. Forum*, vol. 39, no. 7, pp. 155–167, Nov. 2020.
- [33] S. Shim, S. Lee, G. Cho, J. Kim, and S. Kang, "Remote robotic system for 3D measurement of concrete damage in tunnel with ground vehicle and manipulator," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 38, no. 15, pp. 2180–2201, Oct. 2023.
- [34] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–12.
- [35] B. Sutter et al., "A semi-autonomous mobile robot for bridge inspection," *Autom. Construct.*, vol. 91, pp. 111–119, Jul. 2018.
- [36] W. Wang et al., "Stereo-NEC: Enhancing stereo visual-inertial SLAM initialization with normal epipolar constraints," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 2691–2697.
- [37] W. Wang, J. Li, Y. Ming, and P. Mordohai, "EDI: ESKF-based disjoint initialization for visual-inertial SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 1466–1472.
- [38] L. Wu, S. Mokhtari, A. Nazef, B. Nam, and H.-B. Yun, "Improvement of crack-detection accuracy using a novel crack defragmentation technique in image-based road assessment," *J. Comput. Civil Eng.*, vol. 30, no. 1, Jan. 2016, Art. no. 04014118.
- [39] Y. Xiao et al., "Autonomous inspection and construction of civil infrastructure using robots," in *Automation in Construction Toward Resilience*. Boca Raton, FL, USA: CRC Press, 2024, pp. 1–26.
- [40] T. Yamaguchi, S. Nakamura, R. Saegusa, and S. Hashimoto, "Image-based crack detection for real concrete surfaces," *IEEJ Trans. Electr. Electron. Eng.*, vol. 3, no. 1, pp. 128–135, Jan. 2008.
- [41] L. Yang, I. Dryanovski, R. G. Valenti, G. Wolberg, and J. Xiao, "RGB-D camera calibration and trajectory estimation for indoor mapping," *Auto. Robots*, vol. 44, no. 8, pp. 1485–1503, Nov. 2020.
- [42] L. Yang et al., "Automated wall-climbing robot for concrete construction inspection," *J. Field Robot.*, vol. 40, no. 1, pp. 110–129, Jan. 2023.
- [43] L. Yang, B. Li, W. Li, H. Brand, B. Jiang, and J. Xiao, "Concrete defects inspection and 3D mapping using CityFlyer quadrotor robot," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 991–1002, Jul. 2020.
- [44] L. Yang, B. Li, W. Li, Z. Liu, G. Yang, and J. Xiao, "Deep concrete inspection using unmanned aerial vehicle towards cscs database," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Jun. 2017, pp. 1–8.
- [45] C. Zhang, Z. Yin, and R. Qin, "Attention-enhanced co-interactive fusion network (AECIF-Net) for automated structural condition assessment in visual inspection," *Autom. Construct.*, vol. 159, Mar. 2024, Art. no. 105292.
- [46] J. Zhang, X. Yang, W. Wang, J. Guan, L. Ding, and V. C. S. Lee, "Automated guided vehicles and autonomous mobile robots for recognition and tracking in civil engineering," *Autom. Construct.*, vol. 146, Feb. 2023, Art. no. 104699.
- [47] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.