Encoder-Camera-Ground Penetrating Radar Tri-Sensor Mapping for Surface and Subsurface Transportation Infrastructure Inspection

Chieh Chou, Aaron Kingery, Di Wang, Haifeng Li, and Dezhen Song

Abstract— We report system and algorithmic development for a sensing suite comprising multiple sensors for both surface and subsurface transportation infrastructure inspection focusing on multi-modal mapping for inspection. The sensing suite contains a camera, a ground penetrating radar (GPR), and a wheel encoder. We design the sensing suite and propose a data collection scheme using customized artificial landmarks (ALs). We use ALs to synchronize two types data streams: camera images that are temporally evenly-spaced and GPR/encoder data that are spatially evenly-spaced. We also employ pose graph optimization with synchronization as penalty functions to further refine synchronization and perform data fusion for 3D reconstruction. We have implemented the system and tested it in physical experiments. The results show that our system successfully fuses three sensory data and product metric 3D reconstruction. The sensor fusion approach reduces the end-to-end distance error from 7.45cm to 3.10cm.

I. INTRODUCTION

Transportation infrastructure such as bridge decks, freeways, and airport runways requires periodic inspections for maintenance purposes. Manual inspections are labor-intensive and costly. A more viable approach is to employ a robot. The inspection tasks often include both surface and subsurface mapping to assist searching for cracks, voids, or other damages. The ability to combine surface images with subsurface scans is important for further inspections or future repairs. Therefore, it is necessary to combine multiple inspection sensors such as a regular camera, a light Detection and Ranging (LIDAR) device, and a ground penetrating radar (GPR) together along with navigational sensors such as a wheel encoder and/or a global position system (GPS) receiver. Combining data from heterogeneous sensors is challenging because there are challenges in system design, synchronization, correspondence, and data fusion.

We report our recent development on systems and algorithms that enable encoder-camera-GPR tri-sensor fusion for transportation infrastructure inspection which can be viewed as a multi-modal mapping process, a classic problem in robotics with a new sensor combination. We build a sensing suite consisting of the aforementioned sensors (see Fig. 1).

C. Chou, A. Kingery, D. Wang, and D. Song are with CSE Department, Texas A&M University, College Station, TX 77843, USA, Email: dzsong@cs.tamu.edu.

H. Li is with CS Department, Civil Aviation University of China, Tianjin, 300300, China, and also with Fujian Provincial Key Laboratory of Informaton Processing and Intelligent Control, Minjiang University, Fuzhou, 350108, China. Email: hfli@cauc.edu.cn.
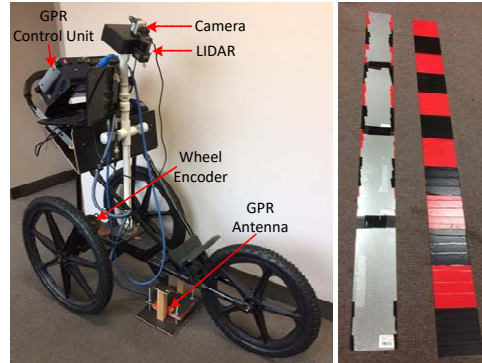
Fig. 1. Left: our sensing suite comprises a camera and a GPR; a GPR includes GPR control unit, wheel encoder, and GPR antenna. Right: artificial Landmarks. Black and red colored side is the upper side while the metal side is downside (best viewed in color).

We also design visualization software and a data collection scheme using artificial landmarks (ALs) to simplify the scanning process. Moreover, ALs allow us to synchronize two types of data streams: camera images that are temporally evenly-spaced and GPR/encoder readings that are spatially evenly-spaced. Our algorithm takes advantage of ALs to identify moments when the camera center, the GPR frame origin, and the intersection point between AL edges and the GPR trajectory are vertically collinear. The moments allow us to align data streams and then refine them in pose graph optimization with synchronization constraints.

We have implemented the algorithm and tested it under physical experiments using real data from bridge deck inspection to validate our system. The results show that our system and algorithm successfully achieve our design goal and the overall system is able to synchronize the two types of data streams with improved accuracy. In the bridge dataset, our algorithm reduces the end-to-end distance error from 7.45cm to 3.10cm.

II. RELATED WORK

Our proposed system relates to the areas of bridge deck or road surface inspection, GPR applications, visual simultaneous localization and mapping (SLAM), and sensor fusion.

Transportation infrastructure inspection is indispensable for safety. To assess infrastructure condition, nondestructive evaluation (NDE) is a highly recommended approach nowadays because it provides reliable diagnosis [1]–[3] without structure destruction. NDE techniques are usually achieved by using sensors such as GPRs, cameras, chain drags, etc. Moreover, the sensors can be further integrated on an autonomous robotic system for inspection, e.g. robotic bridge inspection tool [4]–[9]. Inspired by existing works, our

system comprise a GPR, a camera, and a wheel encoder; and our sensor fusion algorithm can utilize their complementary nature to achieve better 3D reconstruction results.

A GPR can measure the time between echoes of electromagnetic signals to survey the objects and layers underneath the ground surface [10]–[12]. Hence there are a lot of important applications, e.g. archeology [13], [14], mine detection and removal [15], [16], bridge deck inspection and evaluation [4]–[9], and planetary exploration [17], [18]. As an ideal sensor for scanning subsurface structure, it is often limited to flat surface due to its inability to obtain pose information from its wheel encoder alone. Fusing it with a camera with visual SLAM capability can address this problem and significantly increase GPR application scope.

As a variation of the SLAM problem [19], the visual SLAM problem is to simultaneously estimate robot pose and landmark positions using one or more cameras. To solve the visual SLAM problem, there are two popular approaches: filtering (e.g. [20]) and bundle adjustment (e.g. [21]). We employ ORB-SLAM2 [22] as a preprocessing step since it is one of the state-of-the-art visual SLAM algorithms for a monocular camera. In fact, our algorithm is not limited by one particular type of visual SLAM algorithm.

Taking advantage of the complementary nature of different sensory modalities, sensor fusion can improve sensing accuracy, increase robustness, and reduce noises for a variety of applications. Existing sensor fusion combination includes camera-LIDAR [23], [24], WiFi-IMU [25], [26], and camera-IMU [27], [28]. However, camera-GPR sensor fusion has not been well-studied yet. Our group focuses on robotic infrastructure inspection algorithm and system development. In [29], we address extrinsic calibration of for GPRs. Built on the result, we combine a GPR with a camera and develop a mirror-based calibration process to estimate the relative pose between the camera and the GPR [30]. With calibration problems solved, this paper focuses on how to fuse the surface scanning results from a camera with subsurface data from a GPR to provide a comprehensive solution for transportation infrastructure inspection.

III. SYSTEM DESIGN

To enable both surface and subsurface inspection, we design a sensing suite comprising a camera, a LIDAR, a GPR, and a laptop computer detailed in the attached video.

A. Hardware

1) Sensor Choices: The camera (DS-CFMT1000-H) is used not only for surface inspection but also for visual SLAM because it can provide more accurate pose and trajectory estimation than that from a GPS receiver in local region. In addition, GPS may not always be available due to terrain or high-rise buildings. The LIDAR (Hokuyo UST-20LX) can also be used for surface crack detection when cracks cannot be distinguished from background images. The GPR (GSSI SIR-3000) is used for substructure inspection and it is installed a wheel encoder which is also an important sensor

for data fusion. The wheel encoder data is pre-synchronized with GPR data by a hardware trigger.

2) Configuration: Fig. 1 illustrates the physical configuration of the hardware. Due to the coverage requirement, the camera needs to be mounted at least 1 meter above the ground to inspect surface cracks. The LIDAR is mounted as close to the camera as possible to allow the co-centered data registration between the two sensors. To scan the subsurface cracks, the GPR antenna needs to be installed close to the ground surface to ensure good radar signal penetration of the underneath concrete structure. The penetration depth of GPR is inversely proportional to the radio signal frequency, but the resolution of GPR image is proportional to the frequency. Therefore, we choose a 1.6-GHz GPR transceiver antenna because it ensures a 2-meter penetration depth for concrete decks with a resolution of less than 4.7cm. All sensors are mounted on a standard survey tricycle (GSSI model 623).

3) ALs: To facilitate the synchronization of the camera and the GPR, we design ALs that are made of colored patterned metal plates and clearly visible to both sensors as shown in Fig. 1. ALs enable us to align surface and subsurface structure from two different sensor modalities.

B. Software and User Interface

Our customized software visualizes all sensory data (see the attached video). It is developed using C++ based cross-platform graphics sdk QT version 5.9 to ensure it is easy to be ported to different platforms. It has three main components: a camera view, a GPR view, and a world coordinate view. The top right window is the camera view. The bottom right GPR view component renders the full GPR radargram and shows the current GPR scan index. The world coordinate view renders the camera poses, GPR poses, the GPR radargrams, and 3D landmarks in 3D space at the left side. Users can navigate in the world coordinate view to examine the data from all sensors. All three views allow the user to interact with the synchronized data frame by frame. If users choose a particular frame in any of the three views, the other views automatically switch to the corresponding data. "Play," "Next," and "Previous" buttons are located at the bottom of the interface to assist the user to navigate.

C. Data Collection Procedure

Since camera images are taken according to a fixed interval determined by camera internal clock and GPR scans are triggered by its wheel encoder based on a preset fixed distance traveled, there is no inherent synchronization between the two sensors. However, this would lead to significant issues when fusing the data streams. Therefore, we design a data collection procedure (see Fig. 2) to deal with the synchronization problem by using our ALs.

First, we place ALs on the survey area and make sure that those ALs are evenly spaced out and remain fixed on the ground. The number of ALs is adjustable in our setting; but we suggest to use at least two to cover either end of the survey area. Next, we push the sensing suite to traverse ALs by following the preplanned survey trajectory.

During the procedure, we ensure that the sensing suite traverses the edges of ALs on both sides along the trajectory. Fig. 2 shows an example of the setup. There are 4 lines $\{\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_3, \mathbf{L}_4\}$ indicating 4 edges generated by both sides of 2 ALs. Therefore, the GPR passes each edge 3 times and traverses 12 edges in the trajectory. Those edges will be used as references for the synchronization problem in Section IV.
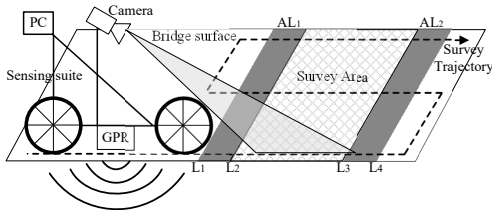

Fig. 2.   Data collection procedure.

Employing ALs makes the data collection procedure more efficient and flexible if comparing to a conventional approach which relies on manually painted lines or grids on the survey area in advance. The conventional approach also requires the GPR to painstakingly move along a preset linear trajectory which is slow and labor intensive. In our design, we only need to place ALs on the ground without precise positioning or strict linear scan motion of the sensing suite.

IV. ALGORITHMS

Recall that camera images and GPR scans are triggered by time and by position intervals, respectively. We propose an algorithm to solve the synchronization problem between the two data streams. The resulting sensor fusion algorithm can use GPR scans to solve the scale ambiguity and drifting issues in monocular visual SLAM and generate a 3D metric reconstruction for both surface and subsurface inspection. The visual SLAM capability enabled by the camera also addresses the inability to estimate poses by GPR alone. The fused metric reconstruction includes camera/GPR poses, 3D landmarks, and GPR scans. We ignore the LIDAR-camera sensor fusion here because it is a well-studied area.

Fig. 3 illustrates the flow of our algorithm: 1) preprocessing, 2) tri-sensor correspondence using ALs, 3) initial scale rectification and synchronization, and 4) optimal scale correction and data alignment.
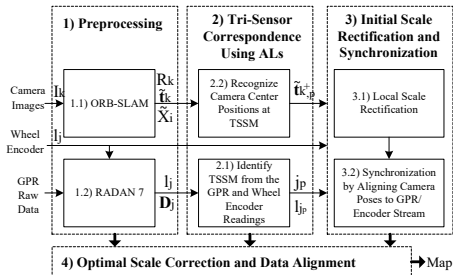

Fig. 3.   Algorithm block diagram.

A. Preprocessing

Camera and GPR data are pre-processed as follows to obtain initial reconstruction information in its individual modality. For camera images, we apply ORB-SLAM2 [22] to estimate camera poses and 3D landmarks at each frame. Let I_k be the camera image at time k, where $k = 1, 2, \ldots, N$. Let $\{W\}$ denote the 3D world coordinate system and $\{C_k\}$ denote the camera coordinate system at time k. Additionally, we define $\{W\}$ to be coinciding with $\{C_1\}$. Given sequential camera images I_k, we obtain camera poses $\{\mathbf{R}_k, \tilde{\mathbf{t}}_k\}$ with respect to $\{W\}$ at time k, where $\mathbf{R}_k$ is the camera rotation matrix and $\tilde{\mathbf{t}}_k \in \mathbb{R}^3$ is the translation vector. The ORB-SLAM2 algorithm also provides 3D landmark position $\tilde{\mathbf{X}}_i \in \mathbb{R}^3$, where index i denotes the i-th 3D landmark, and $i = 1, 2, \ldots, n$. Note that this is the outcome of the monocular visual SLAM. All 3D information is up-to-scale. As a convention in this paper, we use the tilde ˜ to indicate variables in 3D space that are up-to-scale. Therefore, $\tilde{\mathbf{t}}_k$ and $\tilde{\mathbf{X}}_i$ are up-to-scale while $\mathbf{t}_k$ and $\mathbf{X}_i$ are defined in the metric scale. For GPR raw data, we use GSSI RADAN 7 to export the GPR scans. Let $\mathbf{D}_j$ be the j-th GPR scan, which is a 1024-dimension vector containing depth information, and l_j be the GPR accumulated travel length measured by wheel encoder from $\mathbf{D}_1$ to $\mathbf{D}_j$, where $j = 1, 2, \ldots, m$. In addition, all $\mathbf{D}_j$ form the GPR image.

We obtain camera readings $\{\mathbf{R}_k, \tilde{\mathbf{t}}_k, \tilde{\mathbf{X}}_i\}$ and GPR/encoder readings $\{\mathbf{D}_j, l_j\}$ after the preprocessing step. At this moment, we know wheel encoder readings are synchronized with GPR data but we do not know how they correspond to camera readings yet, which is the focus of the next step.

B. Tri-Sensor Correspondence Using ALs

In our design, the camera center is directly above the origin of the GPR coordinate system. Therefore, the ground projection of the camera trajectory is the same as that of the GPR. Let us define AL anchor points to be the points on the AL edges created by the intersection between the projected camera/GPR ground trajectory and AL edges. The three sensor synchronization moment (TSSM) established by ALs is the moment that the camera center, GPR coordinate system origin, and AL anchor point are collinear as a line perpendicular to the ground (see Fig. 4). If we can recognize this moment across all sensor modalities, we can establish positional reference for correspondences among all sensors.

1) Identify TSSM from the GPR and Wheel Encoder Readings: When the GPR origin is directly above AL, the metal plate generates a strong response in the GPR readings and remains constant until GPR finishes traversing the AL. This is because the distance between the AL and the GPR does not change in this duration which leads to a clear short horizontal bar in the radargram (see Fig. 4). The beginning and the ending moments of the short horizontal bar are TSSMs in GPR readings. Let the number of detected edges be M for all ALs in the setting, we can obtain the corresponding GPR scan indexes $\{j_p | p = 1, 2, \ldots, M\}$. For all of these scan indexes, we have their corresponding encoder readings $\{l_{j_p} |, p = 1, 2, \ldots, M\}$. For the example in Fig. 2,
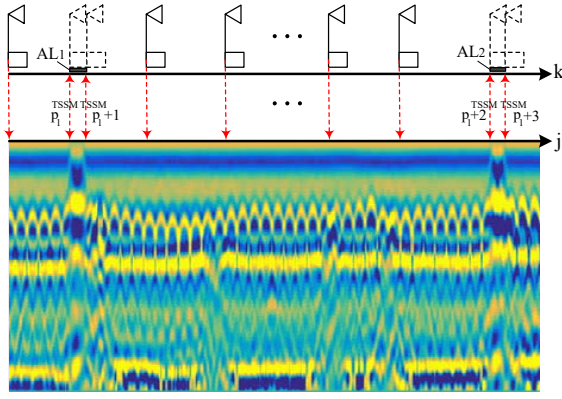
Fig. 4. ALs help us synchronize camera image stream and GPR/encoder data streams. Four TSSMs are shown here. Camera poses represented by small triangles and GPR poses represented by small rectangles are displayed on the top of the radargram. The poses drawn in dashed lines are virtual poses corresponding to TSSMs (best viewed in color).

we have the GPR scan indexes $\{j_1, \ldots, j_{12}\}$ due to $M = 12$ edges.

2) Recognize Camera Center Positions at TSSM: For now, we only have an up-to-scale 3D reconstruction from the ORB-SLAM2 results. We need to obtain TSSM camera center positions in this coordinate system. Note this camera pose is a virtual pose which means it does not correspond to an actual image due to the discretized time across frames. However, this pose can help us synchronize all sensors later. See Fig. 4 for examples of the visual poses.
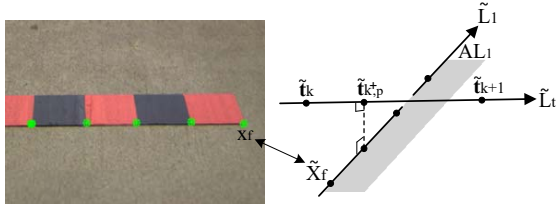

Fig. 5. Left: AL edge points in a 2D camera image. Right: two skew lines $\tilde{\mathbf{L}}_1$ and $\tilde{\mathbf{L}}_t$ are used to compute camera center position $\tilde{\mathbf{t}}_{k^+,p}$ (best viewed in color).

Fig. 5 illustrates the process of identifying the camera center position for the virtual pose. The high contrast pattern in AL allows us to recognize points on the edge (green points in the figure) across multiple frames. Based on the known camera poses, we can obtain their 3D points. Let $\{\mathbf{x}_f \leftrightarrow \tilde{\mathbf{X}}_f | f \in \mathscr{L}_1\}$ be the corresponding 2D and 3D points where $\mathscr{L}_1$ is the index set containing all edge points. These 3D points $\tilde{\mathbf{X}}_f$ allows us to establish AL edge line $\tilde{\mathbf{L}}_1 :$ $\tilde{\mathbf{X}} = \mathbf{p}_1 + t_1 \mathbf{d}_1$, where t_1 is a parameter, $\mathbf{p}_1 = \text{mean}(\tilde{\mathbf{X}}_f)$, and $\mathbf{d}_1$ is the singular vector corresponding to the largest singular value after performing SVD on normalized points $[\tilde{\mathbf{X}}_1 - \mathbf{p}_1, \ldots, \tilde{\mathbf{X}}_f - \mathbf{p}_1, \ldots]^\top$.

From images taken, we can identify two image indexes k and $k+1$ representing two immediate camera poses before and after crossing $\tilde{\mathbf{L}}_1$. The camera center positions at k and $k+1$ are $\tilde{\mathbf{t}}_k$ and $\tilde{\mathbf{t}}_{k+1}$, respectively. The camera trajectory

between k and $k+1$ can be approximated by a line $\tilde{\mathbf{L}}_t : \tilde{\mathbf{X}} = \mathbf{p}_2 + t_2 \mathbf{d}_2$, where t_2 is a parameter, $\mathbf{p}_2 = \tilde{\mathbf{t}}_k$, and $\mathbf{d}_2 = \tilde{\mathbf{t}}_k - \tilde{\mathbf{t}}_{k+1}$.

Finding the skew line between $\tilde{\mathbf{L}}_1$ and $\tilde{\mathbf{L}}_t$ allows us to obtain the camera center positions $\tilde{\mathbf{t}}_{k^+,p}$ at TSSM, where the combo subscripts k^+, p indicate this is slightly after time k and it is corresponding to p-th AL edge. We have

$$\tilde{\mathbf{t}}_{k^+,p} = \mathbf{p}_2 + \frac{(\mathbf{p}_1 - \mathbf{p}_2) \cdot \mathbf{n}_1}{\mathbf{d}_2 \cdot \mathbf{n}_1} \cdot \mathbf{d}_2, \tag{1}$$

where $\mathbf{n}_1 = \mathbf{d}_1 \times (\mathbf{d}_1 \times \mathbf{d}_2)$.

C. Initial Scale Rectification and Synchronization

1) Local Scale Rectification: Now we can recover the true scale by TSSM correspondence. When the sensing suite traverses two adjacent ALs, we can identify two virtual camera poses corresponding to the leading edge of each AL. If the first edge index is p_1, then the second edge index is $p_1 + 2$ due to the adjacency (see Fig 4). Say that the corresponding camera frame indexes are k_1 and k_2. Then the camera center positions of the two virtual poses are $\tilde{\mathbf{t}}_{k_1^+,p_1}$ and $\tilde{\mathbf{t}}_{k_2^+,p_1+2}$. The corresponding distances extracted from encoder readings are $l_{j_{p_1}}$ and $l_{j_{p_1+2}}$, respectively. Then a local scale ratio can be obtained as the ratio between the distance from the encoder and the distance from camera poses,

$$s = \frac{l_{j_{p_1+2}} - l_{j_{p_1}}}{||\tilde{\mathbf{t}}_{k_1+1} - \tilde{\mathbf{t}}_{k_1^+,p_1}|| + \sum_{k=k_1+1}^{k_2-1} ||\tilde{\mathbf{t}}_{k+1} - \tilde{\mathbf{t}}_k|| + ||\tilde{\mathbf{t}}_{k_2^+,p_1+2} - \tilde{\mathbf{t}}_{k_2}||}, \tag{2}$$

where $||\cdot||$ is l_2 norm. Then the scale of all $\tilde{\mathbf{t}}_k$ and $\tilde{\mathbf{X}}_i$ at frames k between k_1+1 and k_2 can be recovered by $\mathbf{t}_k = s\tilde{\mathbf{t}}_k$ and $\mathbf{X}_i = s\tilde{\mathbf{X}}_i$. For $\tilde{\mathbf{t}}_k$ and $\tilde{\mathbf{X}}_i$ before the first virtual camera pose or after the last virtual pose, we can also use the closest local scale ratio to rectify them. Note that we use a local scale ratio instead of a global scale ratio established by the first and the last ALs because monocular SLAM may not have a uniform scale ambiguity in its results due to scale drift. Using a local scale to correct the affected poses and 3D landmarks can alleviate the issue. Also, this is not a precise scale recovery because the encoder readings may be inaccurate due to skids and the scale drift variation may be big. We will handle it in Section IV-D.

2) Synchronizing Camera Poses to GPR/Encoder Data: With the scale rectified, we can align camera poses with the GPR/encoder data streams through distance matching. Let $\mathbf{t}'_k$ be GPR frame origin for the corresponding camera center position $\mathbf{t}_k$. The fixed frame mapping relationship is

$$\mathbf{t}'_k = \mathbf{R}_{ex}\mathbf{t}_k + \mathbf{t}_{ex}, \tag{3}$$

where extrinsic parameters $\{\mathbf{R}_{ex}, \mathbf{t}_{ex}\}$ are obtained by calibration [30]. Define the distance traveled from the first edge of the first AL to the current pose k_c to be $d_0(k_c)$. Say that k_0 is the index of the camera/GPR frame right before the first edge, $\mathbf{t}_{k_0^+,1}$ is the camera center of the first virtual pose right above the first edge and used as the starting point of the inspection. Then we have

$$d_0(k_c) = ||\mathbf{t}'_{k_0+1} - \mathbf{t}'_{k_0^+,1}|| + \sum_{k=k_0+1}^{k_c-1} ||\mathbf{t}'_{k+1} - \mathbf{t}'_k||. \tag{4}$$

Note that we do not have ~ over variables because they are in metric space. We can compute $d_0(k)$ for each camera/GPR pose k. For each $d_0(k)$, there exists an encoder reading l_{j_k}

$$d_l(k) = l_{j_k} - l_0 \cong d_0(k) \qquad (5)$$

where l_0 is the encoder reading corresponding to the first virtual pose $\mathbf{t}'_{k_0^+,1}$, and $d_l(k)$ is the cumulative distance traveled by the sensing suite from wheel encoder readings. Eq. (5) is true because encoder increment is 3mm, which is much less than the camera pose estimation error.

D. Optimal Scale Correction and Data Alignment

Now we can further synchronize sensor readings and perform metric reconstruction using an optimization framework. We formulate a constrained optimization problem as follows. Let the estimated parameters be 3D landmarks $\mathscr{X} = \{\mathbf{X}_i | i = 1, \dots, n\}$, camera orientations $\mathscr{R} = \{\mathbf{R}_k | k = 1, \dots, N\}$, and camera center positions $\mathscr{T} = \{\mathbf{t}_k | k = 1, \dots, N\}$. The cost function for camera re-projection errors is given by

$$C(\mathscr{X}, \mathscr{R}, \mathscr{T}) = \sum_{k=1}^{N} \sum_{i \in \mathscr{S}_k} \|\mathbf{x}_{i,k} - \hat{\mathbf{x}}_{i,k}\|^2_{\Sigma_{i,k}}, \qquad (6)$$

where $\mathscr{S}_k$ denotes the set containing all indexes of points visible by camera at time k, $\mathbf{x}_{i,k}$ is the image observation of $\mathbf{X}_i$ at time k, $\hat{\mathbf{x}}_{i,k} = f(\mathbf{R}_k, \mathbf{t}_k, \mathbf{X}_i)$ is the camera projection function, $\Sigma_{i,k}$ is the covariance of $\mathbf{x}_{i,k}$, and $\|\cdot\|_{\Sigma}$ denotes Mahalanobis distance. It is worth noting that points $\mathbf{x}_{i,k}$ are the surviving inline set from ORB-SLAM2 pose graph.

At each camera/GPR frame, we can incorporate encoder readings to capture the traveled distance. Note that encoder error accumulates linear to the distance traveled. We verify distances traveled between adjacent camera/GPR poses and formulate the follow objective function by considering relative error

$$F(\mathscr{X}, \mathscr{T}) = \sum_{k=2}^{N} \left\| \frac{[d_0(k) - d_0(k-1)] - [l_{j_k} - l_{j_{k-1}}]}{l_{j_k} - l_{j_{k-1}}} \right\|. \qquad (7)$$

Also, virtual poses at AL edges provide more constraints to this problem that can be used as penalty functions in the objective function. For each rectified camera center $\mathbf{t}_{k^+,p}$ for the virtual pose, we can find its corresponding virtual GPR pose at $\mathbf{t}'_{k^+,p}$ using (3). Similar to (4) and (5), we can define travel distance function $d_v(p)$ as

$$d_v(p) = \|\mathbf{t}'_{k_0+1} - \mathbf{t}'_{k_0^+,1}\| + \sum_{k=k_0+1}^{k_p-1} \|\mathbf{t}'_{k+1} - \mathbf{t}'_k\| + \|\mathbf{t}'_{k_p^+,p} - \mathbf{t}'_{k_p}\|, \qquad (8)$$

where k_p is the index of the camera/GPR frame right before p-th AL and $d_v(1) = 0$; and exact encoder reading l_{j_p} for each virtual pose. Then we have

$$G(\mathscr{X}, \mathscr{T}) = \sum_{p=2}^{M} \left\| \frac{[d_v(p) - d_v(p-1)] - [l_{j_p} - l_{j_{p-1}}]}{l_{j_p} - l_{j_{p-1}}} \right\|. \qquad (9)$$

Finally we formulate the optimization problem as

$$\min_{\mathscr{X}, \mathscr{R}, \mathscr{T}} C(\mathscr{X}, \mathscr{R}, \mathscr{T}) + \alpha \cdot F(\mathscr{X}, \mathscr{T}) + \beta \cdot G(\mathscr{X}, \mathscr{T}), \qquad (10)$$

where α and β are nonnegative weighting scalars. In (10), $F(\mathscr{X}, \mathscr{T})$ is a soft constraint due to potential synchronization errors; $G(\mathscr{X}, \mathscr{T})$ is a hard constraint because the AL correspondence at TSSM must be strictly preserved. Therefore, β has a higher value than α. We have to adjust α and β to solve (10). The optimization problem can be solved by first using a small positive weight for α and β, and then applying any nonlinear optimization solver, e.g. Levenberg–Marquardt. Then we gradually increase α and β and use the previous solution as the initial solution to solve (10) iteratively. Finally, we can obtain the solution as β is sufficiently large and the residual is converged. After estimating $\mathscr{X}, \mathscr{R}, \mathscr{T}$, we can repeat the synchronization procedure in Sections IV-C.2 to further remove errors caused by scale drift, and re-optimize to improve accuracy.

V. EXPERIMENTS

The proposed algorithm has been validated in physical experiments. The parameters for GSSI SIR-3000 are given as follows: the horizontal sample rate for the wheel encoder is 390 pulses per meter, the two-way travel time of a radar signal is 8 ns, the GPR sample rate is 1024 samples per scan, and the dielectric constant in air is 1. The resolution for the wheel encoder is 1785 pulses per meter, and the distance error does not exceed ±2% under ideal conditions (e.g. smooth surface and no skid). We have scanned a bridge deck at the Ernest Langford architecture center at Texas A&M University to test our system. The attached video illustrates the dataset and the 3D reconstruction result which shows that our algorithms are able to synchronize these data streams to create successful 3D metric reconstruction.

Furthermore, to validate if our algorithm can improve the accuracy for 3D reconstruction, we adopt the end-to-end GPR travel distance error between the first and last ALs as the metric. When collecting data, we manually measure the distance d_{GT} that GPR traveled from the first AL's edge ($p = 1$) to the last AL's edge ($p = M$) and treat it as the ground truth. We define the error metric as

$$e_{tri} = \left| \|\mathbf{t}'_{k^+,M} - \mathbf{t}'_{k^+,1}\| - d_{GT} \right|, \qquad (11)$$

where $\mathbf{t}'_{k^+,1}$ denotes the first virtual pose and $\mathbf{t}'_{k^+,M}$ denotes the last virtual pose on the trajectory. And we also define the corresponding error metric for the wheel encoder as

$$e_{wheel} = \left| [l_{j_M} - l_{j_1}] - d_{GT} \right|. \qquad (12)$$

Moreover, we know that e_{wheel} is bounded below $e_{max} = a \cdot l_m$, where l_m is the total traveling distance measured by the encoder and a is the relative accuracy (in our case, $a = 2\%$ according to GSSI manual). Finally, we use the bridge dataset to evaluate the accuracy and list results in Table I. The result shows that our algorithm improves reconstruction accuracy.

VI. CONCLUSIONS AND FUTURE WORK

We built an encoder-camera-GPR tri-sensor transportation infrastructure inspection sensing suite and developed a tri-sensor mapping algorithm. Our system design included hard-

TABLE I

ACCURACY TEST

	e_{tri}	e_{wheel}	e_{max}
Error(m)	0.0310	0.0745	0.0816
Accuracy(%)	0.76	1.83	2.00

ware configuration, software interface, ALs, and a data collection scheme. We designed ALs to assist the synchronization between two types of data streams: camera images that are temporally evenly-spaced and GPR/encoder data that are spatially evenly-spaced. We identified synchronization events created by ALs and used them as inputs to synchronize sensory inputs. The results lead to 3D metric reconstruction for synchronized data streams that covers both surface and subsurface structure. We tested our system in real physical experiments. Our system and algorithm have successfully achieved data synchronizations and metric reconstruction. In the future, we will conduct more physical experiments, especially in field tests. We will provide complexity analysis for the algorithm. We will further improve speed and accuracy of the proposed algorithm.

ACKNOWLEDGMENT

REFERENCES

[1] M. Scott, A. Rezaizadeh, A. Delahaza, C. Santos, M. Moore, B. Graybeal, and G. Washer, "A comparison of nondestructive evaluation methods for bridge deck assessment," *NDT & E International*, vol. 36, no. 4, pp. 245–255, 2003.
[2] N. Gucunski, F. Romero, S. Kruschwitz, R. Feldmann, A. Abu-Hawash, and M. Dunn, "Multiple complementary nondestructive evaluation technologies for condition assessment of concrete bridge decks," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2201, pp. 34–44, 2010.
[3] D. Huston, J. Cui, D. Burns, and D. Hurley, "Concrete bridge deck condition assessment with automated multisensor techniques," *Structure and Infrastructure Engineering*, vol. 7, no. 7-8, pp. 613–623, 2011.
[4] R. S. Lim, H. M. La, Z. Shan, and W. Sheng, "Developing a crack inspection robot for bridge maintenance," in *IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China*. IEEE, 2011, pp. 6288–6293.
[5] H. M. La, R. S. Lim, B. B. Basily, N. Gucunski, J. Yi, A. Maher, F. A. Romero, and H. Parvardeh, "Mechatronic systems design for an autonomous robotic system for high-efficiency bridge deck inspection and evaluation," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 6, pp. 1655–1664, December 2013.
[6] H. M. La, R. S. Lim, B. Basily, N. Gucunski, J. Yi, A. Maher, F. A. Romero, and H. Parvardeh, "Autonomous robotic system for high-efficiency non-destructive bridge deck inspection and evaluation," in *IEEE International Conference on Automation Science and Engineering (CASE), Madison, Wisconsin, USA*. IEEE, 2013, pp. 1053–1058.
[7] R. S. Lim, H. M. La, and W. Sheng, "A robotic crack inspection and mapping system for bridge deck maintenance," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 2, pp. 367–378, 2014.
[8] H. M. La, N. Gucunski, S.-H. Kee, and L. Van Nguyen, "Data analysis and visualization for the bridge deck inspection and evaluation robotic system," *Visualization in Engineering*, vol. 3, no. 1, p. 6, 2015.
[9] H. M. La, N. Gucunski, K. Dana, and S.-H. Kee, "Development of an autonomous bridge deck inspection robotic system," *Journal of Field Robotics*, 2017.
[10] D. J. Daniels, *Ground Penetrating Radar, 2nd Edition*. The Institution of Engineering and Technology, 2004, vol. 1.
[11] H. M. Jol, *Ground Penetrating Radar Theory and Applications*. Elsevier, 2008.
[12] A. Annan, *Electromagnetic Principles of Ground Penetrating Radar*. Elsevier, 2009, vol. 1.
[13] L. B. Conyers, *Ground-penetrating radar for archaeology*. AltaMira Press, 2013.
[14] D. Goodman and S. Piro, *GPR remote sensing in archaeology*. Springer, 2013, vol. 9.
[15] T. Fukuda, Y. Hasegawa, K. Kosuge, K. Komoriya, F. Kitagawa, and T. Ikegami, "Environment-adaptive antipersonnel mine detection system-advanced mine sweeper," in *IEEE/RSJ International Conference on Intelligent Robots (IROS), Beijing, China*, 2006, pp. 3618–3623.
[16] M. Freese, T. Matsuzawa, Y. Oishi, P. Debenest, K. Takita, E. F. Fukushima, and S. Hirose, "Robotics-assisted demining with gryphon," *Advanced Robotics*, vol. 21, no. 15, pp. 1763–1786, 2007.
[17] P. Furgale, T. D. Barfoot, N. Ghafoor, K. Williams, and G. Osinski, "Field testing of an integrated surface/subsurface modeling technique for planetary exploration," *The International Journal of Robotics Research*, vol. 29, no. 12, pp. 1529–1549, October 2010.
[18] T. Barfoot, P. Furgale, B. Stenning, P. Carle, L. Thomson, G. Osinski, M. Daly, and N. Ghafoor, "Field testing of a rover guidance, navigation, and control architecture to support a ground-ice prospecting mission to mars," *Robotics and Autonomous Systems*, vol. 59, no. 6, pp. 472–488, 2011.
[19] S. Thrun and Y. Liu, "Multi-robot slam with sparse extended information filers," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, P. Dario and R. Chatila, Eds. Springer Berlin / Heidelberg, 2005, vol. 15, pp. 254–266.
[20] J. Civera, O. G. Grasa, A. J. Davison, and J. Montiel, "1-point ransac for ekf-based structure from motion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), St. Louis, MO*. IEEE, 2009, pp. 3498–3504.
[21] Y. Lu and D. Song, "Visual navigation using heterogeneous landmarks and unsupervised geometric constraints," in *IEEE Transactions on Robotics (T-RO)*, vol. 31, no. 3, June 2015, pp. 736 —- 749.
[22] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, 2017.
[23] R. W. Wolcott and R. M. Eustice, "Visual localization within lidar maps for automated urban driving," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 176–183.
[24] Y. Lu, J. Lee, S.-H. Yeh, H.-M. Cheng, B. Chen, and D. Song, "Sharing heterogeneous spatial knowledge: Map fusion between asynchronous monocular vision and lidar or other prior inputs," in *The International Symposium on Robotics Research (ISRR), Puerto Varas, Chile*, December 2017.
[25] J. Biswas and M. Veloso, "Wifi localization and navigation for autonomous indoor mobile robots," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4379–4384.
[26] H.-M. Cheng and D. Song, "Localization in inconsistent wifi environments," in *The International Symposium on Robotics Research (ISRR), Puerto Varas, Chile*, December 2017.
[27] M. Achtelik, M. Achtelik, S. Weiss, and R. Siegwart, "Onboard imu and monocular vision based control for mavs in unknown in-and outdoor environments," in *Robotics and automation (ICRA), 2011 IEEE international conference on*. IEEE, 2011, pp. 3056–3063.
[28] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart, "Fusion of imu and vision for absolute scale estimation in monocular slam," *Journal of intelligent & robotic systems*, vol. 61, no. 1, pp. 287–299, 2011.
[29] C. Chou, S. Yeh, J. Yi, and D. Song, "Extrinsic calibration of a ground penetrating radar," in *IEEE International Conference on Automation Science and Engineering (CASE), Fort Worth, Texas*, Aug. 2016.
[30] C. Chou, S. Yeh, and D. Song, "Mirror-assisted calibration of a multi-modal sensing array with a ground penetrating radar and a camera," in *IEEE/RSJ International Conference on Intelligent Robots (IROS), Vancouver, Canada*, Sept. 2017.