A deep learning framework based on improved self-supervised learning for ground-penetrating radar tunnel lining inspection

Huang, Jian; Yang, Xi; Zhou, Feng; Li, Xiaofeng; Zhou, Bin; Lu, Song; Ivashov, Sergey; Giannakis, Iraklis; Kong, Fannian; Slob, Evert

DOI
10.1111/mice.13042

Publication date
2023

Document Version
Final published version

Published in
Computer-Aided Civil and Infrastructure Engineering

Citation (APA)
Huang, J., Yang, X., Zhou, F., Li, X., Zhou, B., Lu, S., Ivashov, S., Giannakis, I., Kong, F., & Slob, E. (2023). A deep learning framework based on improved self-supervised learning for ground-penetrating radar tunnel lining inspection. *Computer-Aided Civil and Infrastructure Engineering*, *39*(6), 814-833. https://doi.org/10.1111/mice.13042

DOI: 10.1111/mice.13042

RESEARCH ARTICLE

COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING WILEY

A deep learning framework based on improved self-supervised learning for ground-penetrating radar tunnel lining inspection

Jian Huang[1] | **Xi Yang**[1] | **Feng Zhou**[1,2] | **Xiaofeng Li**[1] | **Bin Zhou**[3] | **Song Lu**[3] | **Sergey Ivashov**[4] | **Iraklis Giannakis**[5] | **Fannian Kong**[6] | **Evert Slob**[7]

[1]School of Mechanical Engineering and Electronic Information, China University of Geosciences (Wuhan), Wuhan, China

[2]Guangdong Provincial Key Laboratory of Geophysical High-resolution Imaging Technology, Southern University of Science and Technology, Shenzhen, China

[3]China Railway Southwest Research Institute Co. LTD, Chengdu, China

[4]Remote Sensing Laboratory, Bauman Moscow State Technical University, Moscow, Russia

[5]School of Geosciences, University of Aberdeen, Aberdeen, UK

[6]Norwegian Geotechnical Institute, Oslo, Norway

[7]Department of Geoscience and Engineering, Delft University of Technology, Delft, The Netherlands

Correspondence
Feng Zhou, School of Mechanical Engineering and Electronic Information, China University of Geosciences (Wuhan), Wuhan 430074, China.
Email: zhoufeng@cug.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 41974165, 42241131; Guangdong Provincial Key Laboratory of Geophysical High-resolution Imaging Technology, Grant/Award Number: 2022B1212010002; CRSRI Open Research Program, Grant/Award Number: CKWV2021883/KY; Yunnan Provincial Science and Technology Key R&D Program, Grant/Award Number: 202203AA080006; Russian Science Foundation, Grant/Award Number: 211900043

Abstract

It is not practical to obtain a large number of labeled data to train a supervised learning network in tunnel lining nondestructive testing with ground-penetrating radar (GPR). To decrease the dependence of supervised learning on the number of labeled data, an improved self-supervised learning algorithm—self-attention dense contrastive learning (SA-DenseCL)—is proposed and incorporated with a mask region-convolution neural network (Mask R-CNN), which is trained by unlabeled and labeled GPR data. The proposed SA-DenseCL adds a self-attention-based relevant projection head to the DenseCL architecture of self-supervised learning, capturing the spatially continuing information between adjacent GPR traces. In the workflow, some unlabeled GPR images are used to pre-train the SA-DenseCL network for feature extraction and obtaining the backbone weights, which is superior to the conventional pre-training methods of supervised learning pre-trained by ImageNet images. The weights of the pre-trained backbone are then used to initialize the Mask R-CNN through transfer learning. Subsequently, a limited number of labeled GPR images are used to fine-tune the Mask R-CNN for automatically identifying the locations of the reinforcement bars and voids and estimating the secondary lining thickness. The experimental results show that the average precision reaches 96.70%, 81.04%, and 94.67% in identifying reinforcement bar locations, detecting void defects, and estimating secondary lining thickness, respectively, which

outperform the conventional methods that use ImageNet-based supervised learning or GPR image-based DenseCL for initializing the Mask R-CNN backbone weights. It is observed that the improved self-supervised learning-based framework can improve the detection and estimation accuracy in GPR tunnel lining inspection.

1 | INTRODUCTION

In tunnel engineering, lining plays a crucial role in bearing surrounding rock pressure and structure self-weight to maintain the stability of the tunnel (Balaguer, et al., 2014). Some key factors, exemplified by nonstandard construction, untimely maintenance, and geological environment changes, bring defects or damages to the tunnel lining and lead to potential dangers to the tunnel structure (Bergeson & Ernst, 2015; Zhang et al., 2014). Therefore, timely monitoring and maintenance of the tunnel lining are essential to prolong the service life of the tunnel (Menendez et al., 2018; Montero et al., 2015).

In the recent decade, deep learning algorithms, mainly based on convolutional neural network (CNN), have become the mainstream image processing algorithms in the field of computer vision, by which high-dimensional image features are readily extracted (Krizhevsky et al., 2012). Neural network and deep learning have been widely used in various fields of civil engineering fields (Adeli, 2002), such as real estate estimation (Rafiei & Adeli, 2016), bridge structure defect detection (Sajedi & Liang, 2021), and estimation of concrete compressive strength (Rafiei et al., 2017), showing gradually increasing attraction. In tunnel condition assessment, Xue and Li (2018) proposed to use a full convolution neural network to classify and identify various defects on the outer surface of tunnel lining and verified that the performance of this model is more accurate and faster than other several CNN models, such as AlexNet, GoogLeNet, and Visual Geometry Group network-16. Zhu et al. (2021) proposed the cloud model-based random forests to detect the loss of metro tunnels during the operational period, and they applied a semi-supervised approach to overcome the problem of too few data in the database. Zhou et al. (2022) proposed an enhanced you only look once version4 (YOLO v4) model for high-precision and real-time detection and identification of various defects in tunnel lining; compared with the original YOLO v4 and other conventional models, this model avoids the impact of different lighting and complex background and can achieve higher accuracy in localizing defects.

In addition to the visible structural defects on the surface, some potential problems are hidden inside the tunnel lining structure and cannot be straightforwardly observed, typically represented by excessive spacing of reinforcement bars and void forming (H. Wang et al., 2018). Moreover, the insufficient thickness of the secondary lining construction will also cause huge hidden danger to the safety of the tunnel structure. Although the initial lining mainly bears the pressure of the surrounding rock, the secondary lining controls the deformation of the surrounding rock, prevents the initial lining from being corroded, and bears the weight of ventilation and lighting equipment in the tunnel (Barpi & Peila, 2012). Figure 1 shows the diagrammatic sketch of the initial lining and secondary lining in the tunnel structure.

It is too expensive to drill a hole for the on-site check, and thus nondestructive testing (NDT) is significant in the internal inspection of the concrete structure, including sound diagnosis (Suda et al., 2004), infrared thermography (Sirca & Adeli, 2018), and ground-penetrating radar (GPR) (Parkinson & Ékes, 2008). As one of the mainstream NDT techniques, the GPR is characterized by good portability, rapid detection speed, and high spatial resolution. Therefore, GPR has been widely applied in civil and infrastructure engineering applications, including tunnel lining inspection (Lai et al., 2017; Montero et al., 2015). By scanning on the surface of the lining, GPR instru-

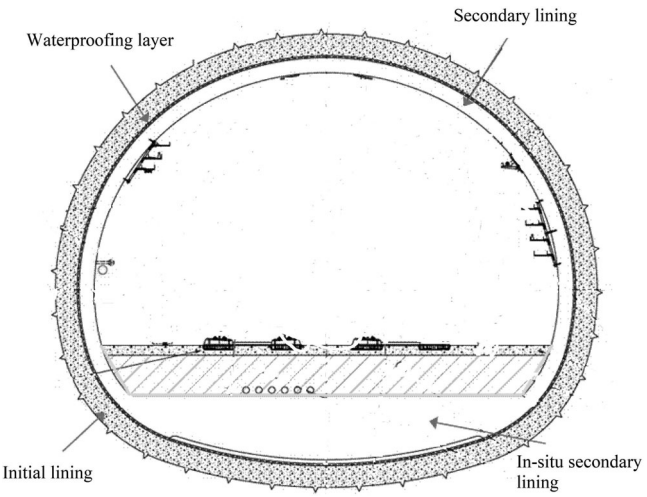


FIGURE 1 The schematic for the initial lining and secondary lining in a tunnel structure (Uhrin et al., 2017).

ments collect electromagnetic (EM) waves reflected from the objects inside the lining structure and form a GPR image by combining a serial of traces. These GPR images can not directly delineate the objects, wherefore data interpretations are crucial (Annan, 2003). Accurate and precise interpretations primarily rely on the experiences of an interpreter. However, once the data volume is huge, conventional manual interpretations seem impractical due to great time consumption and high labor costs (Chiaia et al., 2022).

In order to detect the target signals in GPR images quickly, accurately and automatically, Pham and Lefèvre (2018) used the faster region CNN (Faster-RCNN) to automatically extract hyperbolic signals reflected from the underground buried objects in GPR images, which demonstrates better performance than traditional algorithms. J. Gao et al. (2020) proposed an improved Faster-RCNN algorithm, called Faster R-ConvNet, for GPR pavement detection and achieved considerable accuracy in the inspections of cracks, water damage pits, and uneven settlements. Li et al. (2022) used YOLO v3 to automatically identify the signals reflected from reinforcement bars in GPR images and then combined GPR images with EM induction data to accurately estimate the cover thickness and reinforcement bar diameter through a one-dimensional CNN. In GPR tunnel lining assessment, Rosso et al. (2022) proposed to use the GPR data-based deep learning model to classify the internal damage degree of tunnel lining, and they find that compared with the ResNet-50, the more advanced vision transformer model showing an overwhelming advantage in classification accuracy. Marasco et al. (2022) studied the Fourier transform GPR pre-processing for data compression to improve the efficiency of deep learning model.

Although the examples mentioned above show successful trials of deep learning in various GPR scenarios, there are still some challenges existing in practical field applications, as typically exemplified with the identification and location of internal defects in tunnel lining using GPR. First, a well-trained deep learning architecture is greatly dependent on the number and types of labeled samples for representation learning. However, in the practical tunnel lining GPR images, the defect signals are not widespread, which means that it is impractical to collect enough GPR images to support the training of the supervised deep learning model. Second, it is expensive to label a large number of GPR samples containing defects for the labeling process is time-consuming, human-consuming, and experience-dependent. These two problems impose considerable restrictions on the applications of deep learning in practical tunnel lining inspection.

Transfer learning shows increasing attraction in alleviating the problem of insufficient training samples (Pan & Yang, 2009). In this process, open-source natural image datasets are used in classification pre-training of supervised learning to create the backbone parameters, and then realistic GPR data are used to fine-tune the model parameters (Rosso et al., 2023). By this means, general image representations are learned by the backbone. However, GPR images have considerably different features from the conventional images in the open-source datasets. Therefore, when labeled GPR samples are not adequate, the fine-tuning process may not explicitly improve the performance of the network.

Further, Qin et al. (2021) used deep convolutional generative adversarial network to expand GPR samples based on some obtained GPR data for Mask R-CNN training, and the verification in a realistic tunnel lining showed improved inspection accuracy, compared with the method that uses only real field data. However, these synthetic GPR data still require a lot of labor for manual labeling, and more essentially only the performance of the fine-tuning stage is improved. From another angle, J. Wang et al. (2022) replaced ImageNet with a large number of classified GPR simulated images for the classification pre-training of the backbone, which, to some extent, relieves the maladjustment of the feature extraction due to the feature differences between GPR and general images. However, this method is not suitable to pre-train real GPR data, for the reason that the classification pre-training is based on supervised learning and some a priori category information is needed for each sample, which does not save more labor than labeling each GPR images. In addition, the simulated data are derived from the simplified models relative to the real world, and thus the GPR images present more or less feature deviations from the complex real environments, which actually decline the capability of feature extraction of the backbone.

To develop a practical and inexpensive deep learning architecture for tunnel lining inspection application, it is crucial to decrease the dependence of the network on the sample number. Self-supervised learning, as one of the unsupervised learning methods, has the potential for solving the problem of insufficient labeled training data (Rafiei et al., 2023). Self-supervised learning uses unlabeled data for representation learning by constructing a specific pretext task (Hjelm et al., 2019). In the last 3 years, contrastive learning, as one of the self-supervised learning algorithms, is becoming popular in the field of computer vision because it takes training the capability of learning the similarity and dissimilarity among the unlabeled images as the pretext task, where the feature extraction ability of backbone is trained (T. Chen et al., 2020). He et al. (2020) proposed momentum contrast (MoCo), a contrastive learning algorithm, which shows comparable performance to supervised learning in many

downstream tasks. Afterward, X. Chen et al. (2020) proposed an improved MoCo, called MoCo v2, by integrating data augmentation and nonlinear transformation. Based on MoCo v2, X. Wang et al. (2021) proposed dense contrastive learning (DenseCL) algorithm, which adds pixel-level feature comparison into contrastive learning. Compared with supervised learning, DenseCL demonstrates great superiority in object detection and semantic segmentation (X. Wang et al., 2021). In recent years, self-supervised learning has also been applied to civil engineering (Van et al., 2022), but its applications in tunnel lining internal detection and GPR NDT is relatively rare.

Inspired by the successful applications of self-supervised learning method in the field of computer vision, this paper proposes an improved self-supervised learning algorithm—self-attention DenseCL (SA-DenseCL)—for GPR image applications, where a relevant projection head is designed and added to DenseCL. Although DenseCL has an excellent performance in processing natural images, it is considered that there is a great difference between the features of natural images and GPR images. For example, there is a strong correlation between each A-scan waveform of the GPR images, which is different from the correlation information between each pixel of the natural images. This means that DenseCL may have some deficiencies in processing the features of GPR images. Therefore, according to the characteristics of GPR image formation, on the basis of the original DenseCL, a relevant projection head based on self-attention is added in SA-DenseCL to enhance the learning of the correlation information of A-scan waveforms in GPR images. This algorithm is combined with Mask R-CNN (He et al., 2016) to construct a deep learning architecture, solving the problem of insufficient training samples in practical tunnel lining inspections. In this application environment, the deep learning network is expected to automatically identify reinforcement bars, localize void defects, and estimate secondary lining thickness through GPR data.

The proposed deep learning procedure includes two training steps: First, unlabeled GPR images are used to pre-train the backbone of SA-DenseCL, and the resulting backbone parameters are treated as the initial parameters of the backbone of Mask R-CNN; second, a limited number of labeled GPR images are used to fine-tune the parameters of Mask R-CNN for accurately identifying the inspected objects from GPR images. To assess the improved performance of SA-DenseCL in the constructed network architecture, this algorithm is compared with the supervised pre-training and the original DenseCL algorithm, respectively. The effectiveness of the proposed network architecture is testified by on-site drilling verifications of newly constructed tunnels.

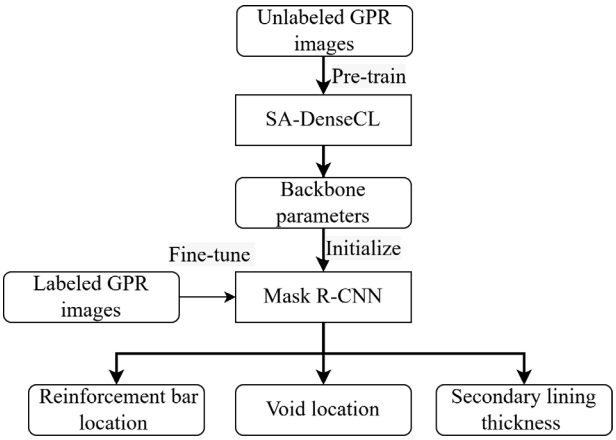


FIGURE 2 Flow chart of network training for tunnel lining inspection based on deep learning. GPR, ground-penetrating radar; Mask R-CNN, mask region convolution neural network; SA-DenseCL, self-attention dense contrastive learning.

2 | METHODOLOGY

A self-supervised learning-based algorithm for GPR data pre-training is combined with Mask R-CNN to form a deep learning workflow for the autonomous inspection of tunnel lining internal structure as shown in Figure 2. In this workflow, the improved self-supervised learning algorithm—SA-DenseCL—and Mask R-CNN work for the backbone pre-training and the fine-tuning, respectively. In the pre-training stage, SA-DenseCL is trained with a large number of unlabeled GPR data, including those without target signals, and the resulting backbone parameters are treated as the initial parameters of the subsequential Mask R-CNN. In the fine-tuning stage, a limited number of labeled GPR data are used to fine-tune the Mask R-CNN parameters, whereafter the trained Mask R-CNN is used to predict the reinforcement bar, void, and the secondary lining thickness. The architecture is composed of two main modules, that is, SA-DenseCL and Mask R-CNN. The Mask R-CNN is able to simultaneously implement image detection and image segmentation tasks, and thus suits well for solving the problems of object localizations and layering in GPR images. More details on the network of Mask R-CNN can be found in He et al., 2017. The details of SA-DenseCL are to be clarified in the following subsections.

2.1 | SA-DenseCL pipeline

In the pre-training stage, SA-DenseCL is used for representation learning with unlabeled GPR data, which are improved based on the self-supervised learning algorithm DenseCL by adding a self-attention module to fit the

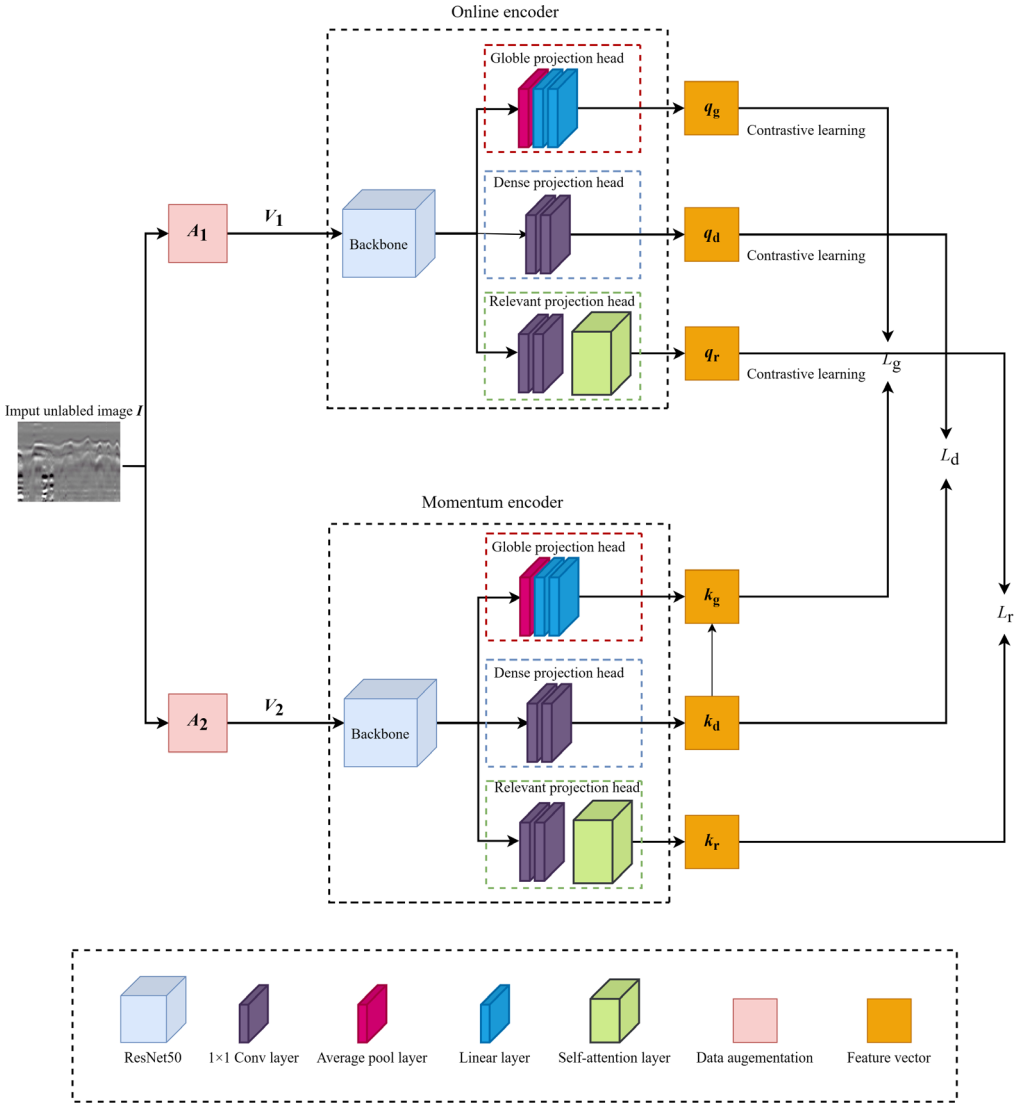
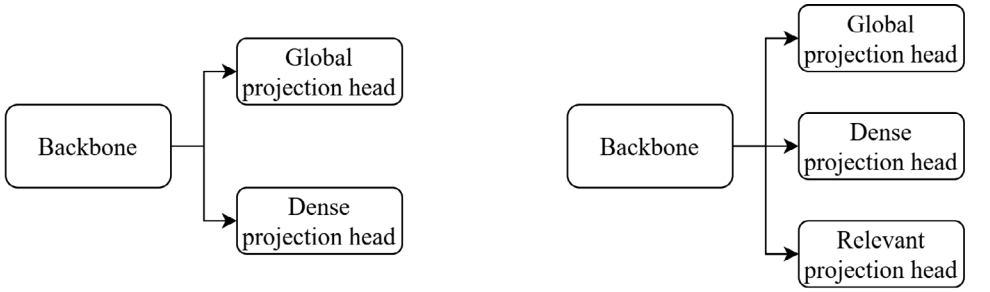


FIGURE 3 The network structure of the self-attention dense contrastive learning (SA-DenseCL).

features of GPR images. Figure 3 shows the overall architecture of SA-DenseCL. In addition to two random data augmentation combinations, the SA-DenseCL is mainly composed of an online encoder and a momentum encoder. Each encoder consists of a backbone for preliminary feature extraction and three feature projection heads for feature vector refinement. The feature vectors output by the encoders are constrained by the loss function to implement contrast learning. The backbones of the online encoder and momentum encoder both adopt Residual network-50 (ResNet-50) (He et al., 2016). After the pre-training of SA-DenseCL, only the parameters of the backbone are retained for the initialization of the parameters of Mask R-CNN. In view of the fact that each GPR image is composed of traces of waveforms, there exists strong relevance between adjacent trace of waveforms. The initial DenseCL algorithm aims to process general natural images. When processing GPR images, in addition to the global projection head and dense projection head, a relevant projection head based on self-attention is added to the encoder of SA-DenseCL, which aims to strengthen the learning capacity of the relevant information between traces of waveforms. The difference in encoder structure between SA-DenseCL and DenseCL is shown in Figure 4.

As the illustrated workflow for SA-DenseCL in Figure 3, when the SA-DenseCL starts to work, each unlabeled GPR image $\boldsymbol{I}$ is transformed into two views, $\boldsymbol{V}_1$ and $\boldsymbol{V}_1$, through two different data augmentation combinations, $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$. For clearer clarification, the two random data augmentation combinations $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ include one or several operations like random cropping, random flipping, contrast adjustment, saturation adjustment, Gaussian blur, and sunlight, with the aim to enlarge the difference between the two views, $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$, as much as possible. Afterward, the two views are sent to the online encoder and the momentum encoder for feature coding, and the feature



(a) The structure of encoder in DenseCL (b) The structure of encoder in SA-DenseCL

FIGURE 4 The structure of the encoders in DenseCL and SA-DenseCL.

vectors representing the whole view are put out. It is noted that the online encoder and the momentum encoder have the same structure, but their parameters are not shared.

In the process of network training, the gradient back-propagation is driven by calculating the similarity loss of the two output vectors, resulting in the update of the parameters of the online encoder. At the same time, the parameters of the momentum encoder are updated following the formula:

$$\theta_{i+1}^{m} = k\theta_{i}^{m} + (1-k)\,\theta_{i}^{o} \tag{1}$$

where θ^{o} and θ^{m} represent the parameters of the online encoder and momentum encoder, respectively, i represents the iterations in training, and k represents the momentum coefficient (ranging from 0 to 1). In contrastive learning, a larger momentum coefficient usually leads to stronger learning ability. By referring to X. Wang et al. (2021), the default value of k is 0.9 in this paper. Note that in the online encoder, the backbone is retained, while the rest are discarded after pre-training, for the reason that only the parameters of the backbone are needed in the downstream tasks.

Global project heads in the two encoders can project the global features of the feature maps exported from the backbones. To be more specific, the sizes of the feature maps output from the backbones are 7×7, and they subsequentially become 1×1 after downsampling by an adaptive average pool layer. Following the average pool layer, there are two linear layers linked by the activation function Rectified Linear Unit as proposed by Nair and Hinton (2010). At the end of the global projection head, a feature vector is output with a shape of $1 \times 1 \times 128$. An improved normalization method—shuffling batch normalization—is used to shuffle the vector order in the current mini-batch to increase the training difficulty of the whole SA-DenseCL so that the backbone is promoted for learning more features from more complexity (He et al., 2020).

For the task of objection detection and semantic segmentation, a dense projection head is used to project the feature map in a pixel level. In this module, two 1×1 convolution layers retain the size of the feature map, and then the feature map is divided into 49 local feature vectors with 128 channels, which are to be exported for the subsequential contrastive learning in a pixel level (X. Wang et al., 2021).

Significant improvement of the network is made in the projection head parts of the encoders, where a novel projection head called relevant projection head is added into the encoders as an individual module to fit the features special to GPR images. Specifically, the relevant projection heads aim to catch the relevant information of the local features of GPR images. In essence, a GPR image consists of a series of traces of echoes, and thus there is great relevance existing among the adjacent traces of echoes from the objects. This is different from the natural images, which are based on pixels and relevance existing in any direction. The relevant projection head is added to extract the relevant information along the traces to promote an additional level of contrastive learning among the local feature vectors.

In the following two subsections, detailed specifications are presented on the self-attention in the relevant projection head and the contrastive learning loss function for contrastive learning, for they are the primary improved parts in the SA-DenseCL module.

2.2 | Self-Attention in relevant projection head

The proposed relevant projection head consists of two 1×1 convolution layers and a self-attention module. The convolution layers adjust the channels of the feature map, which are exported from the backbone, to match the input of the self-attention module. Self-attention mechanism was originally used in the field of natural language processing to capture the relevance of the word sequence (Vaswani et al., 2017). In GPR images, the signals reflected from the objects present continuous spatial relevance along each trace, which can be in analogy with the temporal

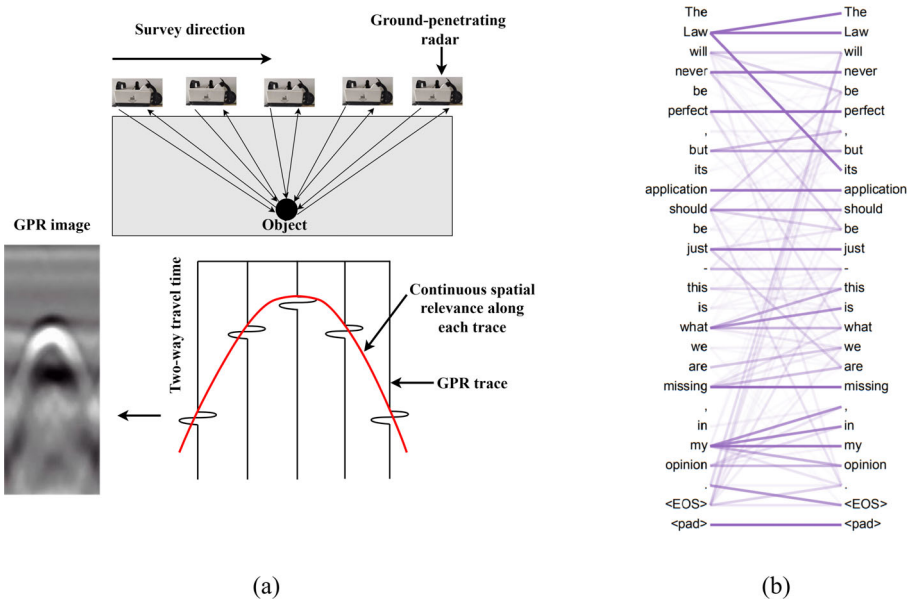


FIGURE 5 Schematic illustration for the analogy between ground-penetrating radar (GPR) images and natural language processing. (a) A GPR image consisting of traces of echoes, and (b) temporal relevance between words in natural language (Vaswani et al., 2017).

continuity of the natural language process. As illustrated by Figure 5, the reflected arc in the GPR image, consisting of a series of echo traces (Figure 5a), presents the relevance of spatial sequence along survey line direction, which is in analogy with the relevance of temporal continuity in natural language (Figure 5b). In the relevant projection head, each feature map that is transformed from the GPR image by backbone is divided into a series of feature sequences along the survey line direction. Then, the relevant information between these feature sequences is calculated by the self-attention module. Therefore, the self-attention module is introduced into the relevant projection head to capture the relevant information among the various local features in GPR images, enhancing the capability of the backbone in distinguishing object signals from clusters and noises.

Figure 6 shows the structure of the self-attention mechanism. The feature vector $\boldsymbol{X}$ is imported to three different linear layers for matrix multiplications obtaining the three eigenvectors ***Query***, ***Key*** and ***Value*** (hereinafter referred to as $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$, respectively). Then, the similarity between $\boldsymbol{Q}$ and $\boldsymbol{K}$ is calculated according to the function (Vaswani et al., 2017):

$$\boldsymbol{F} = \frac{\boldsymbol{Q} \cdot \boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_{\boldsymbol{K}}}}, \tag{2}$$

where $\boldsymbol{K}^{\mathrm{T}}$ represents the transpose of $\boldsymbol{K}$, and $d_{\boldsymbol{K}}$ represents the dimensions of $\boldsymbol{K}$. The softmax layer normalizes the similarity coefficient matrix of $\boldsymbol{Q}$ and $\boldsymbol{K}$. Subsequentially, the similarity coefficient matrix is multiplied by the eigenvectors $\boldsymbol{V}$ to obtain the vector $\boldsymbol{Y}$.

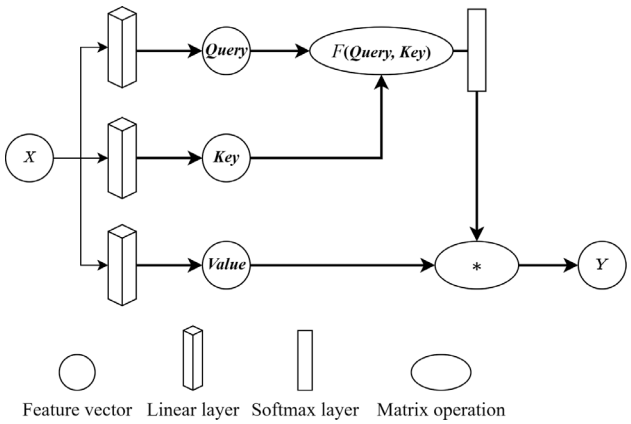


FIGURE 6 Structure of self-attention mechanism. The symbols $\boldsymbol{X}$ and $\boldsymbol{Y}$ represent the imported and exported feature vectors, respectively.

On the whole, the self-attention mechanism can be expressed by (Vaswani et al., 2017)

$$Attention = S\left(\frac{\boldsymbol{Q} \cdot \boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_{\boldsymbol{K}}}}\right)\boldsymbol{V}, \tag{3}$$

where S represents the softmax function. Through this procedure, the output feature vector $\boldsymbol{Y}$ extracts the self-correlation information from the input feature vector $\boldsymbol{X}$.

2.3 | Contrastive learning loss function

Essentially, the feature vectors from the three projection heads of SA-DenseCL are output for contrastive learning

by calculating their similarity losses with the loss function of L_g, L_d, and L_r as illustrated in Figure 3. Contrastive learning is an analogy with a dictionary look-up task (He et al., 2020). Thus, three dictionaries are established for contrastive learning of the three projection head branches.

In the dictionary of each projection head branch, the basic elements are queries and keys, and each query forms a pair of positive samples or negative samples with any key. As presented in Figure 3, two views obtained from image $\boldsymbol{I}$ are transformed into two feature representations by the online encoder and the momentum encoder. The feature representation exported from the online encoder is called query ($\boldsymbol{q}$), and at the same time, the other feature representation, which is exported from the momentum encoder, is denoted by the positive key ($\boldsymbol{k}^+$). For the eigenvectors $\boldsymbol{q}$ and $\boldsymbol{k}^+$ obtained from image $\boldsymbol{I}$, the feature representations from other import images are treated as the negative key ($\boldsymbol{k}^-$). In the dictionary composed of feature representations, for each query ($\boldsymbol{q}$)($\boldsymbol{q}$), there is a set of coded keys $\{\boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2 \ldots\}$, whereas only one positive key ($\boldsymbol{k}^+$) combined with $\boldsymbol{q}$ forms a pair of positive samples, and the rest $\boldsymbol{k}^-$s combined with $\boldsymbol{q}$s form many pairs of negative samples. According to the contrastive loss function, that is, info noise comparative estimation proposed by Oord et al. (2018), the goal of network training is to keep $\boldsymbol{q}$ close to $\boldsymbol{k}^+$ and away from any $\boldsymbol{k}^-$.

In this module, the loss function of the global projection head branch is expressed by (X. Wang et al., 2021)

$$L_g = -\ln \frac{\exp(\boldsymbol{q}_g \cdot \boldsymbol{k}_g^+/\tau)}{\exp(\boldsymbol{q}_g \cdot \boldsymbol{k}_g^+) + \sum_{\boldsymbol{k}_g^-} \exp(\boldsymbol{q}_g \cdot \boldsymbol{k}_g^-/\tau)}, \quad (4)$$

where a temperature hyper-parameter τ is introduced for the quick convergence of the loss function in the process of training, and it is set as 0.2 by default.

For the dense projection head branch, in order to achieve pixel-level feature through contrastive learning, the feature map output by the convolution layers is segmented into some small-size local feature vectors. In this paper, the output feature maps have sizes of 7×7, and they are segmented into 49 local feature vectors. In the network, there are two sets of local feature vectors output by the dense projection head branches of the online encoder and the momentum encoder. The query $\boldsymbol{q}_d$ represents the local feature vector from the online encoder, and its corresponding unique positive key $\boldsymbol{k}_d^+$ is selected from the local feature vectors $\boldsymbol{k}_d$s output by the momentum encoder. To determine $\boldsymbol{k}_d^+$, the largest similarity between $\boldsymbol{q}_d$ and $\boldsymbol{k}_d$ is searched following the formula:

$$\boldsymbol{k}_d^+ = \arg\max_i \left[sim\left(\boldsymbol{q}_d, \boldsymbol{k}_d^i\right)\right], \quad (5)$$

where *sim* represents the cosine similarity and $\boldsymbol{k}_d^i$ represents the *i*th local feature vector. The loss function of the dense projection head branch can be formulated by (X. Wang et al., 2021)

$$L_d = -\frac{1}{N} \sum_{\boldsymbol{k}_d^i} \ln \frac{\exp(\boldsymbol{q}_d \cdot \boldsymbol{k}_d^+/\tau)}{\exp(\boldsymbol{q}_d \cdot \boldsymbol{k}_d^+) + \sum_{\boldsymbol{k}_d^-} \exp(\boldsymbol{q}_d \cdot \boldsymbol{k}_d^-/\tau)}, \quad (6)$$

where N represents the number of local feature vectors.

For the relevant projection head branch, the structure of the loss function is similar to L_g, and it is formulated as

$$L_r = -\ln \frac{\exp(\boldsymbol{q}_r \cdot \boldsymbol{k}_r^+/\tau)}{\exp(\boldsymbol{q}_r \cdot \boldsymbol{k}_r^+) + \sum_{\boldsymbol{k}_r^-} \exp(\boldsymbol{q}_r \cdot \boldsymbol{k}_r^-/\tau)}. \quad (7)$$

The total loss function can be formulated by

$$L = (1 - \alpha - \beta) L_g + \alpha L_d + \beta L_r \quad (8)$$

where α and β represent the weight coefficients of the loss functions of the dense projection head branch and the relevant projection head branch, respectively. As discussed by X. Wang et al. (2021), in the DenseCL, the equal weight coefficients of L_g and L_d are able to acquire an optimal value. In this paper, α and β are set to 0.4 and 0.2, respectively, and these assignments are determined by the experiments that will be discussed in Section 3.4.

3 | EXPERIMENTS

To testify the superiority of the improved self-supervised learning algorithm and the proposed deep learning network architecture in the application of tunnel lining inspection with GPR data, a number of realistic GPR images, which were collected from several newly constructed tunnel sites, are used to train the SA-DenseCL and Mask R-CNN networks, in order to autonomously identify and estimate the reinforcement bars, void defects, and secondary lining thickness from the GPR images. As stated above, in the proposed architecture, SA-DenseCL acts as a self-supervised learning for pre-training with unlabeled GPR field images. By replacing the SA-DenseCL module with supervised ImageNet pre-training, the performance of the SA-DenseCL algorithm in GPR target signal identifications is compared. Further, ablation studies are conducted to verify the effectiveness of the SA-DenseCL algorithm, compared with its original form, that is DenseC. Finally, a field test with drilling verifications is conducted to further prove the practicability of the proposed workflow and algorithm.

3.1 | Data

All the data used in the network training and testing were collected from several newly constructed highway tunneling sites in Sichuan Province and Yunnan Province, China. The data were collected by commercial GPR devices SIR-3000 and SIR-4000 with the operation center frequency of 400 MHz. There are total of 13,365 GPR images used in the network training, among which 11,694 images are unlabeled for the pre-training of SA-DenseCL, and 1671 images are labeled for the fine-tuning of Mask R-CNN. It should be noted that in this study, the labeled dataset and the unlabeled dataset were divided unintentionally. A total of 1671 labeled GPR images were annotated by two data interpretation engineers from 13,365 GPR images within 3 months, and then the remaining GPR images were used as unlabeled datasets.

In the training of GPR images, we define the objects signals as the scattered or reflected waves from reinforcement bars, void defects, and secondary lining. In this paper, we do not further classify cracks, water-bearing voids, and air-bearing voids based on the considerations of the limited acquired data and the similarity of these kinds of objects, and therefore they are all categorized into void defects. The secondary line signals are actually referring to the signals reflected from the boundary between the secondary line and the initial lining, which are identified to infer the thickness of the secondary line. Note that void defects are not frequently found in the GPR images, and approximately 80% of unlabeled images are collected from real tunnel linings without void signals. Therefore, in the pre-training stage, the background is also deemed one of the features for the backbone to learn.

The labeled 1671 GPR images include 1671 secondary lining signals, 10,921 reinforcement bar signals, and 510 void defect signals. These labels are all identified and labeled by experienced engineers as the ground truth. It is acknowledged that artificial annotation may lead to some missed or misidentified object signals; however, it could be the only feasible way considering the unavailability of many drilled data. These labeled images are further divided into a training set and a test set with the number ratio of 7:3. In the training set, there are 1170 GPR images, including 7842 reinforcement bar signals, 371 void defect signals, and 1170 secondary lining signals. It is noted that 20% of GPR images in the training set are randomly allocated as the validation set, by which the optimal hyper-parameters are derived and over-fitting is avoided. In the test set, there are 501 GPR images, which include 3079 reinforcement bar signals, 139 void defect signals, and 501 secondary lining signals.

TABLE 1 Parameters settings of data augmentation for self-attention dense contrastive learning (SA-DenseCL) (X. Wang et al., 2021)

Parameter	A_1	A_2
Random crop probability	1.0	1.0
Flip probability	0.5	0.5
Color jittering probability	0.8	0.8
Brightness adjustment max intensity	0.4	0.4
Contrast adjustment max intensity	0.4	0.4
Saturation adjustment max intensity	0.2	0.2
Hue adjustment max intensity	0.1	0.1
Color dropping probability	0.2	0.2
Gaussian blurring probability	1.0	0.1
Solarization probability	0.0	0.2

3.2 | Implement and evaluation

The network training is implemented on a high-performance computer configured with two GTX3090-Ti GPUs, two Intel(R) Xeon(R) Silver 4114T CPUs and two Samsung 32GB DDR4 RAM memories. The network is built based on the open-source machine learning library—Detectron2 (Y. Wu et al., 2019).

In the pre-training of SA-DenseCL, the sizes of the input images are scaled to 224 × 224 for matching the input unit of ResNet-50 in the network, and the batch sizes and epochs are set to 64 and 800, respectively. A stochastic gradient descent optimizer is selected to optimize the training speed, where the initial learning rate is set to 3×10^{-4} and the weight decay to 1×10^{-4}. The parameters of the two data augmentation combinations are set by referring to X. Wang et al. (2021) as shown in Table 1.

In the fine-tuning stage, ResNet-50, after pre-trained by SA-DenseCL, is used as the backbone of Mask R-CNN. The batch sizes, the number of epochs, the initial learning rates, and the weight decay are set to 16, 120, 0.005, and 0.001, respectively. These parameters are determined by observing the changes in accuracy and the loss curve of the validation set. The loss curve of the training set and validation set are shown in Figure 7.

Average precision (AP) is a frequently used indicator for evaluating the effects of deep leaning in object detection and instance segmentation (He et al., 2017). AP can comprehensively reflect the precision and the recall rate under different thresholds of intersection over union (IoU). In this paper, AP is used as the evaluation index for the proposed deep learning algorithm and architecture. Equations (9)–(11), respectively, formulize precision, recall rate, and AP as

$$P = \frac{TP}{TP + FP} \tag{9}$$

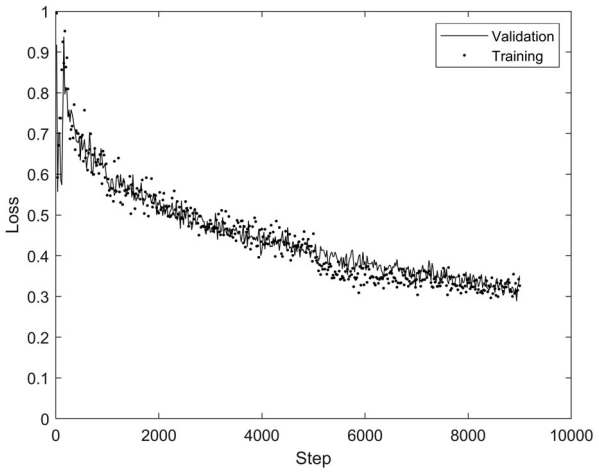


FIGURE 7 Loss curve of mask region convolution neural network.

TABLE 2 Criteria for the predicted result of the target signal

Ground truth / Predicted	Target signal	Others
Target signal	True positive	False positive
Others	False negative	True negative

$$R = \frac{TP}{TP + FN} \tag{10}$$

$$AP = \int_0^1 PR\mathrm{d}(R) \tag{11}$$

where *TP* represents the true positive, *FP* the false positive, and *FN* the false negative.

Table 2 shows the judgement standards of the positive and negative samples in the test results. If IoU exceeds the threshold value, then the predicted boxes are deemed target signals, and vice versa. In the inspections of reinforcement bars and void defects, the IoU threshold is set to 0.5 according to the rule of thumb.

3.3 | Results and comparisons

As illustrated in Figure 1, self-supervised learning is introduced to the deep learning-based workflow for GPR object identification. To assess the strength of SA-DenseCL in improving the backbone performance, the SA-DenseCL module is replaced by the ImageNet-based supervised pre-training and the DenseCL algorithm in the pre-training stage, while the remaining parts are kept unchanged. In the pre-training stage, the ImageNet-based supervised learning method extracts the features of the natural images of the ImageNet library and transfers them into the Mask R-CNN for backbone initialization, while the DenseCL and SA-DenseCL networks extract the features of the unlabeled GPR images and then transfer them into the Mask R-CNN backbone. When comparing with the supervised ImageNet pre-training, the parameters of the backbones are obtained through SA-DenseCL pre-trained with unlabeled GPR images and supervised ImageNet with ImageNet images. However, when compared with the DenseCL algorithm, the parameters of the backbones are both pre-trained with the same number of unlabeled GPR images. Finally, a total of 496 labeled GPR images are selected as the test sets, and the predicted results are compared with the evaluation indexes of recall, precision, and AP.

TABLE 3 Evaluation results of identifying the reinforcement bar signals with SA-DenseCL, DenseCL, and supervised pre-training-based workflows

Pre-training method	Recall	Precision	Average precision (AP)
Supervised pre-training	96.82%	97.68%	95.53%
DenseCL	97.23%	97.89%	96.42%
SA-DenseCL	97.09%	98.75%	**96.70%**

TABLE 4 Evaluation results of identifying the void signals with SA-DenseCL, DenseCL, and supervised pre-training-based workflows

Pre-training method	Recall	Precision	AP
Supervised pre-training	85.61%	78.23%	72.55%
DenseCL	88.54%	81.19%	78.12%
SA-DenseCL	92.07%	82.50%	**81.04%**

Tables 3 and 4 show the evaluation indexes of the three different methods in identifying the locations of reinforcement bars and recognizing void defects, respectively. Compared with the ImageNet-based supervised learning, the network frame based on SA-DenseCL gains an increased AP of 1.17% and 8.49% in identifying reinforcement bars and voids, respectively, reflecting a higher performance in GPR object identification. Compared with DenseCL, the AP by the SA-DenseCL-based method increases by 0.32%, 2.92%, and 2.13% in the identifications of reinforcement bars, voids, and second lining, respectively, demonstrating the effectiveness of the added relevant projection head branch in recognizing objects.

Figure 8a–j exemplifies the comparison results of identified signals reinforcement bars signals and void defect signals by supervised ImageNet pre-training-based and SA-DenseCL pre-training-based methods, respectively. The reinforcement bar reflected signals in the GPR images present hyperbolic shapes with a downward opening, and the alternating pattern of light and dark implies the reverse polarization of EM waves reflected from metal

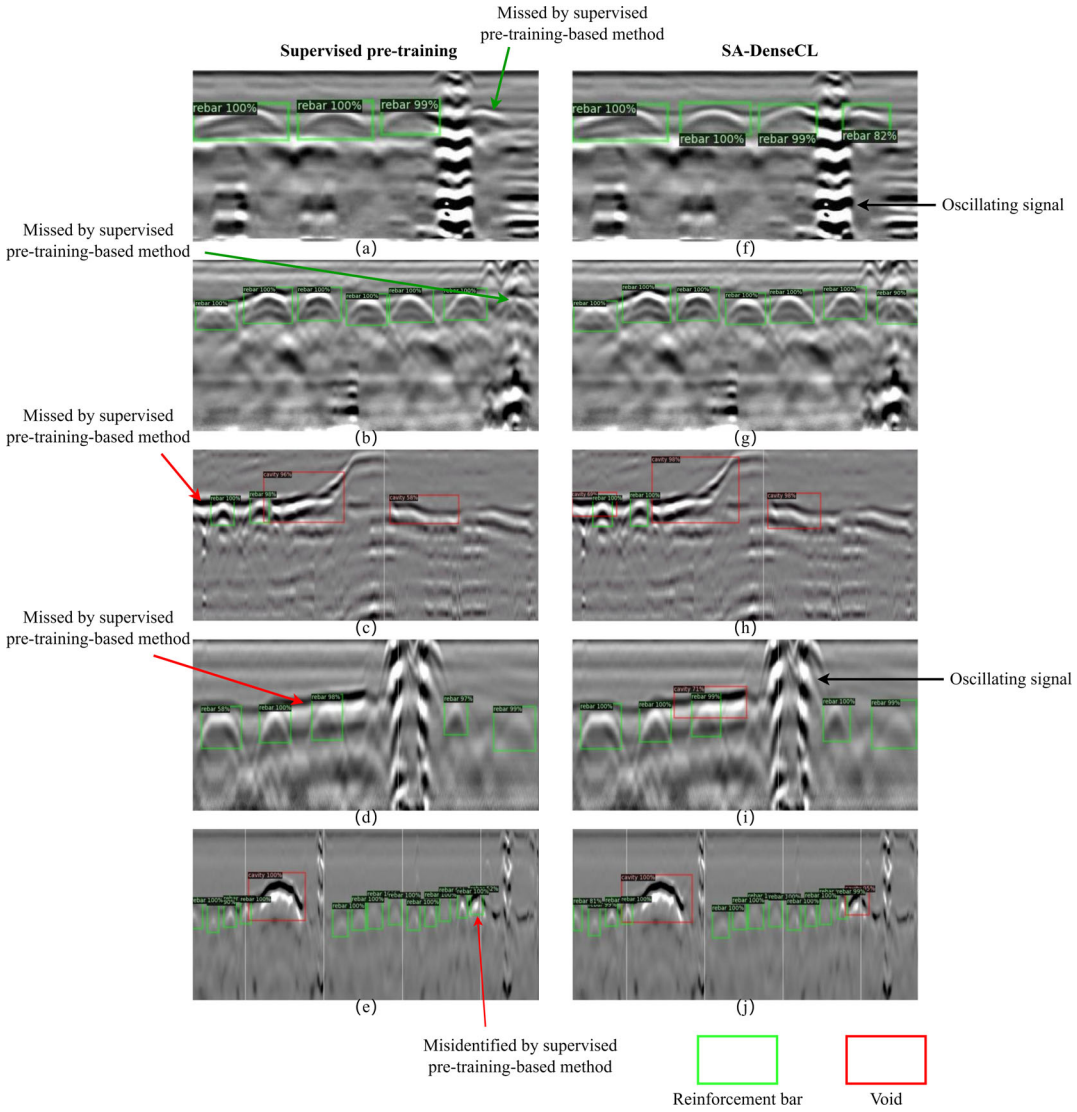


FIGURE 8 The predicted results of the reinforcement bar reflected signal and void reflected signal detection by the method SA-DenseCL-based and the method based on supervised ImageNet pre-training. Reinforcement bar reflected signals and void reflected signals are marked by green boxes and red boxes, respectively.

components. These features are easy to be identified by the network. Relative to the reinforcement bar signals, the void defects present complex reflected signals in the GPR images due to their irregular shapes, sizes, and different contained contents, which increase the difficulty of identifications, and therefore the identification accuracy relies on the experience of the operators during labeling in addition to the network performance. There are some strong oscillating signals found in Figure 8f,h, which are caused by the construction joints or drainage pipelines that are regularly distributed on the lining surface. These signals are not desired identification targets, wherefore they are excluded by the network because they are not labeled in the fine-training stage.

Figure 8a–e indicates the signals of reinforcement bars and voids identified by the supervised pre-training-based method. It can be seen that the majority of the reinforcement bar signals are successfully identified, which is contributed by a considerable number of labeled reinforcement bar signals labeled in the fine-tuning stage, while there are still a few reinforcement bar signals missed as indicated in Figure 8a,b. It is thought to be caused by the overlapping signals or clutters. Additionally, it can be found that several signals of voids are missed as indicated in Figure 8c–e. That is because the signal features reflected from void defects are relatively complex and the number of void signals in the labeled data is not adequate for fine-tuning, which leads to the image features extracted by the backbone being inaccurate. Figure 8f–j presents the signals of reinforcement bars and voids identified by SA-DenseCL-based workflow. It can be seen that the missed signals are correctly identified, which

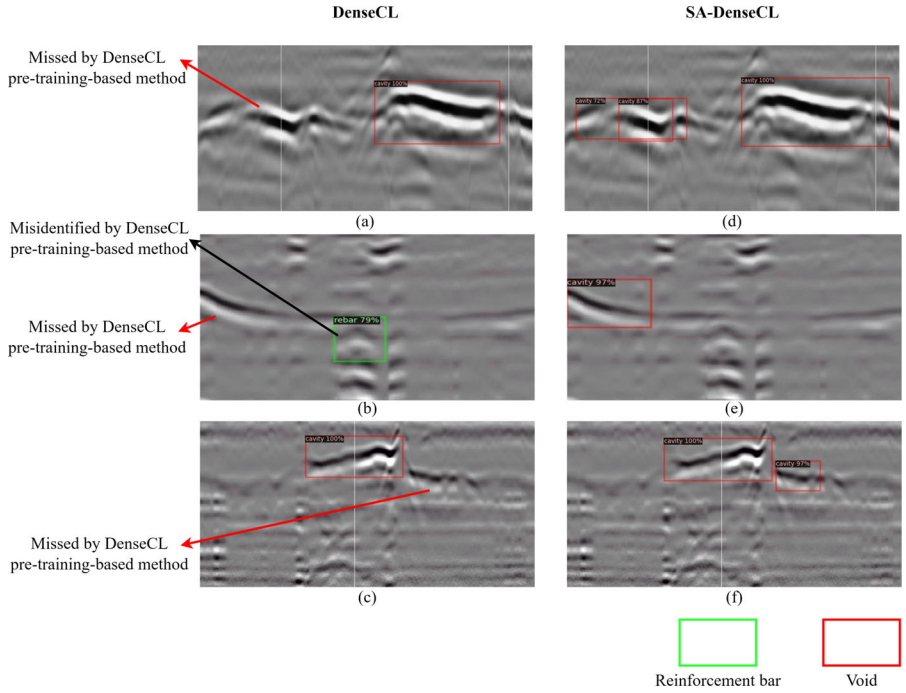


FIGURE 9 The identified results for the locations of reinforcement bars and voids by the SA-DenseCL-based and DenseCL-based methods, respectively. Reinforcement bar reflected signals and void reflected signals are marked by green boxes and red boxes, respectively.

reflects a higher recognition capacity of the SA-DenseCL-based workflow than the supervised pre-training-based method.

Figure 9a–f presents some examples containing missed and misidentified objects. By contrast, it can be seen that four objects are missed by the DenseCL-based method, while they are identified as voids by the SA-DenseCL-based method. The missed voids are caused by relatively weak or small reflected signals (Figure 9a,c). In addition, it can be seen that a hyperbolic-like signal is misidentified as a void (Figure 9b), whereas it is a part of oscillating signals arising from the measurement operation. It is obvious that the SA-DenseCL-based method has higher recognition accuracy than the DenseCL-based method in identifying void defect signals. This further proves that, compared with the conventional DenseCL algorithm, the relevant projection head in the SA-DenseCL module strengthens the learning capacity for continuing sequences of information in GPR images during the pre-training process, and thus can extract finer features for the backbone than DenseCL.

The comprehensive comparison of results shows that the performance of the supervised pre-training-based method relies heavily on the number of labeled GPR images. Due to the small number of void defect signals used for fine-tuning, the supervised pre-training-based method generally performs poorly in void defect signals recognition. Although DenseCL improves this problem to some extent, the performance of SA-DenseCL is better.

TABLE 5 Evaluation results of the identified secondary lining by the methods based on SA-DenseCL, DenseCL, and supervised pre-training

Pre-training method	Recall	Precision	AP
Supervised pre-training	94.02%	93.67%	91.23%
DenseCL	94.86%	94.91%	92.54%
SA-DenseCL	96.32%	95.83%	94.67%

The proposed workflow can greatly improve the feature extraction ability of the backbone using unlabeled GPR images, thus reducing the dependence of Mask R-CNN on the labeled GPR images. Therefore, after the backbone pre-training by SA-DenseCL, even with a limited number of labeled samples used for fine-tuning the Mask R-CNN, the Mask R-CNN can demonstrate adequate identification potential.

In the estimation of the secondary lining thickness, essentially, the network identifies the signals reflected from the boundary between the initial line and the secondary line, and then the secondary lining area is trapped in each GPR image by mapping the masked pixel. The masked regions of the secondary lining reflect the true thickness of the secondary lining in the constructed tunnels once the wave velocity is precisely determined.

Table 5 shows the evaluation indexes of the deep learning architectures based on three different pre-training methods in predicting the secondary lining thickness. It can be seen that the SA-DenseCL-based method

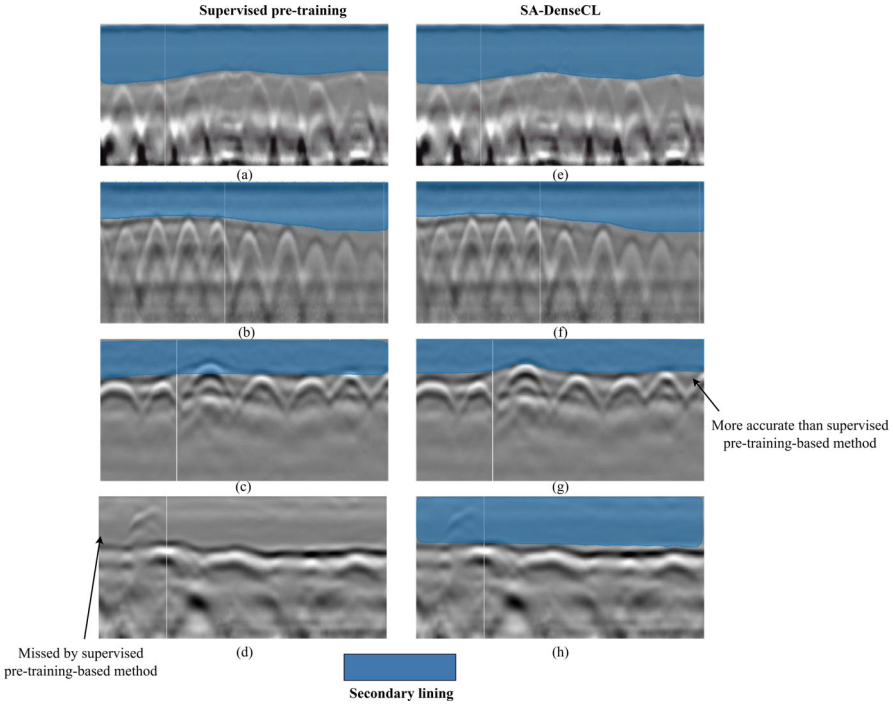


FIGURE 10 Results of the second lining reflected signal area identified by the method SA-DenseCL-based and the method based on supervised ImageNet pre-training. The areas of the second lining reflected signal are marked by blue masks.

outperforms the ImageNet-based and DenseCL-based methods by 3.44% and 2.13%, respectively, in AP, demonstrating the superiority of SA-DenseCL in the feature extraction of GPR images.

Figure 10 exemplifies the outlined second lining areas based on SA-DenseCL and supervised pre-training. For the GPR images with a relatively uniform distribution of reinforcement bar reflected signals, both methods can accurately predict the mask of the secondary lining area as shown in Figure 10a,b. However, for those with uneven distribution of reinforcement bar reflected signals, the SA-DenseCL-based workflow demonstrates better identification effects than the supervised learning-based method as shown in Figure 10c,g. Furthermore, for the GPR images without explicitly visible reinforcement bar reflected signals, the supervised pre-training-based method misidentifies some second lining areas, whereas the proposed workflow does not, as shown in Figure 10d,h.

Figure 11 exemplifies the outlined second lining areas by SA-DenseCL-based and DenseCL-based methods. By the comparisons, it is obvious that the SA-DenseCL-based method (Figure 11a,b) has higher accuracy than the DenseCL-based method (Figure 11d–e) in identifying the secondary lining area from the GPR images. In the GPR images with weakly visible reinforcement bar reflected signals, the second lining area is not marked by the DenseCL-based method (Figure 11c). The comparisons from Figures 8 and 11 indicate that SA-DenseCL can strengthen the capacity of the backbone for extracting more complete and accurate feature representations from GPR images, compared with the supervised pre-training and DenseCL.

To testify the underlying mechanism of SA-DenseCL in improving the feature extraction capacity of the backbone, a GPR image containing overlapping reinforcement bar reflected signals and void reflected signals is imported into the backbones that have been pre-trained by the SA-DenseCL and supervised pre-training for feature extraction, respectively. The feature maps in Conv 1 layer are specially visualized. Figure 12 shows the comparison results. The same experiment is also used for the comparison between SA-DenseCL and DenseCL as shown in Figure 13. It can be seen that the void defect signals, missed by the supervised pre-training-based method and DenseCL pre-training-based method, have a weaker characteristic intensity than the SA-DenseCL pre-training-based method, which is responsible for the missed object. In essence, the comparison reflects the strong feature extraction capacity of the backbone pre-trained by SA-DenseCL.

In order to investigate the added costs of the computational complexity brought about by the addition of relevant projection heads to DenseCL, the sizes of SA-DenseCL DenseCL and Mask R-CNN modules are compared as indicated in Table 6. It can be seen that the size of the SA-DenseCL module presents a slight increase relative to the DenseCL module, while the size of the Mask R-CNN

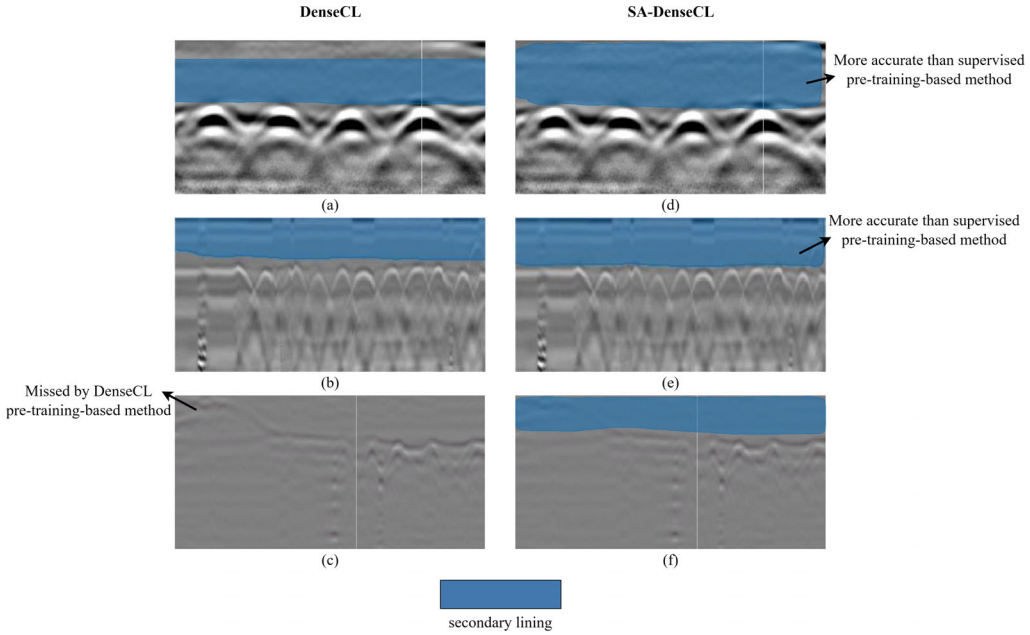


FIGURE 11 Results of the second lining reflected signal area identified by the method SA-DenseCL-based and the method based on DenseCL. The areas of the second lining reflected signal are marked by blue masks.

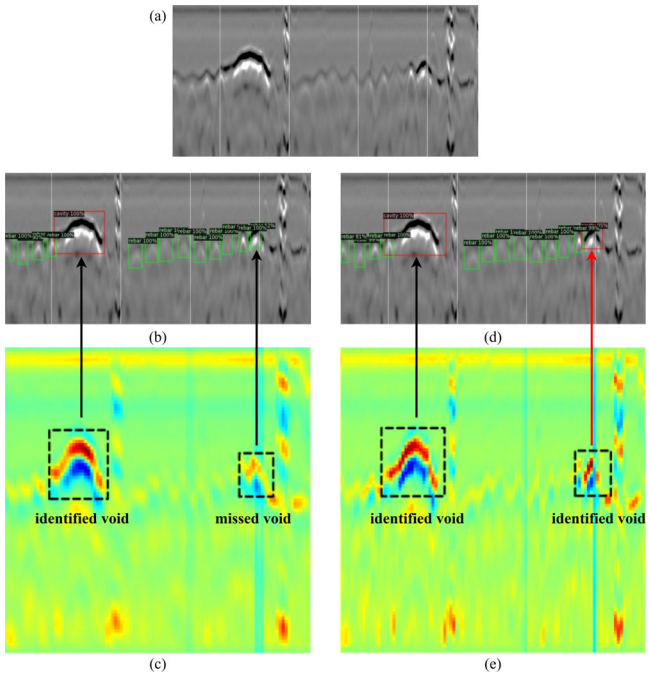


FIGURE 12 Comparison results of the visualization of the feature map in the backbone obtained based on SA-DenseCL and supervised pre-training respectively: (a) imported GPR image, (b) result identified by the method based on supervised pre-training in GPR image, (c) visualization of feature map in backbone pre-trained by supervised, (d) result identified by the method based on SA-DenseCL pre-training in GPR image, and (e) visualization of feature map in backbone pre-trained by SA-DenseCL. The characteristic region of the void reflection signals is manually framed with black dash boxes in the feature maps.

TABLE 6 Comparisons of the sizes of SA-DenseCL, DenseCL, and the mask region convolution neural network (Mask R-CNN) in the two methods, respectively.

	Model	Size (M)
Method 1	DenseCL	525.75 M
	Mask R-CNN	158.06 M
Method 2	SA-DenseCL	558.51 M
	Mask R-CNN	158.06 M

does not change, reflecting that the addition of the relevance project heads will not obviously introduce additional computational costs. (Table 6)

3.4 | Ablation study

Two hyper-parameters α and β in Equation (8) sever as the weights to balance the three contrastive loss terms, that is, the global term, dense term, and relevant term. Referring to the DenseCL (X. Wang et al., 2021), the weight of the global term is set equal to that of the dense term, and, thus, once β is determined, the weights of the three loss terms are determined. Compared with the DenseCL, extra information representing the coherence and relevance of the adjacent GPR traces is extracted by the added relevant projection head, which is reflected by the parameter β in this ablation experiment. Table 7 and Figure 14 present the evaluation results with different β. It can be seen that SA-DenseCL achieves the best performance when β is 0.2, while as β

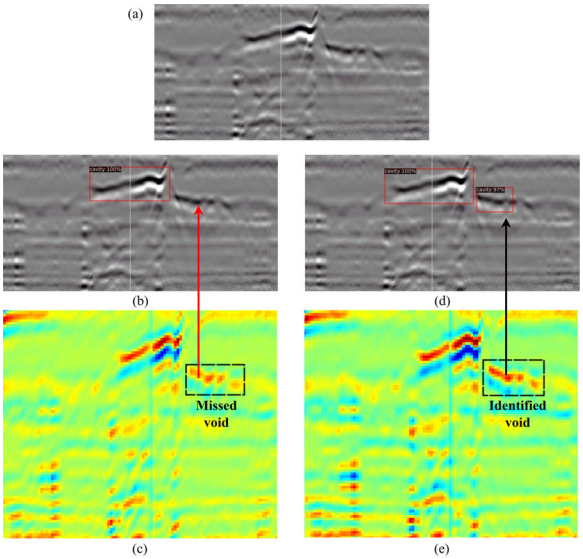


FIGURE 13 Comparison results of the visualization of the feature map in the backbone obtained based on SA-DenseCL and DenseCL pre-training respectively: (a) imported GPR image, (b) result identified by the method based on DenseCL pre-training in GPR image, (c) visualization of feature map in backbone pre-trained by DenseCL, (d) result identified by the method based on SA-DenseCL pre-training in GPR image, and (e) visualization of feature map in backbone pre-trained by SA-DenseCL. The characteristic region of the void reflection signals is manually framed with black dash boxes in the feature maps.

TABLE 7 Evaluation results of APs with different β.

	AP		
B	Reinforcement bar	Void	Secondary lining
0.05	96.49%	78.76%	92.87%
0.10	96.51%	79.52%	93.17%
0.15	96.40%	80.19%	93.86%
0.20	96.70%	**81.04%**	**94.46%**
0.25	**97.02%**	79.37%	93.09%
0.30	96.64%	77.50%	92.39%

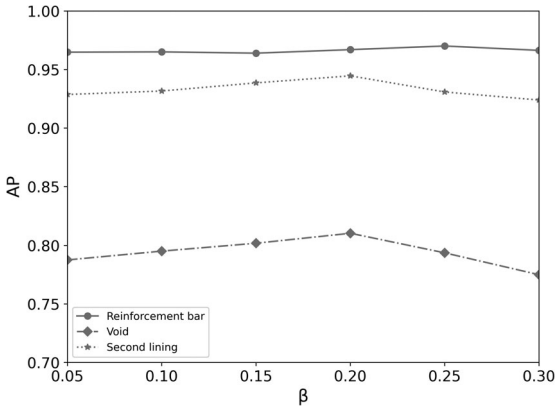


FIGURE 14 Trend of the average precision (AP) with the increased β.

TABLE 8 The evaluation results of SA-DenseCL with different numbers of unlabeled ground-penetrating radar (GPR) images for pre-training

	AP		
Number of GPR images	Reinforcement bar	Void	Secondary lining
3000	92.31%	69.49%	90.98%
6000	96.21%	78.64%	93.72%
9000	**96.82%**	80.09%	94.54%
11,694	96.70%	**81.04%**	**94.67%**

TABLE 9 The evaluation results of DenseCL with different numbers of unlabeled GPR images for pre-training

	AP		
Number of GPR images	Reinforcement bar	Void	Secondary lining
3000	91.69%	67.34%	87.18%
6000	96.10%	76.05%	91.97%
9000	96.36%	77.97%	92.38%
11,694	**96.42%**	**78.12%**	**92.54%**

goes up to 0.3, the relevant projection head decreases the performance of SA-DenseCL.

In the pre-training stage, a total of 11,694 unlabeled GPR images were collected, and these 11,694 unlabeled GPR images were all used for the pre-training of SA-DenseCL. In order to explore the influence of different numbers of unlabeled GPR images used in the pre-training of SA-DenseCL on its performance, and compare the performance between DenseCL and SA-DenseCL with different numbers of unlabeled GPR images, 9000, 6000, and 3000 unlabeled GPR images were used to pre-train SA-DenseCL and DenseCL. The training settings and training data used for fine-tuning the Mask R-CNN were kept unchanged. Tables 8 and 9, respectively, report the performance of SA-DenseCL and DenseCL using different numbers of unlabeled GPR images for pre-training.

It is obvious that with the decrease in the number of unlabeled GPR images used for self-supervised pre-training, the performance of the SA-DenseCL-based method and DenseCL-based method is getting worse. However, with the same number of unlabeled GPR images, the performance of SA-DenseCL always exceeds that of DenseCL, which confirms that the performance improvement of SA-DenseCL is significant. It is noted that once the number of unlabeled GPR images used in pre-training is too small, such as 3000, the pre-training effect of the two self-supervised learning algorithms is both worse than that of supervised pre-training. However, it is also surmised that more than 11,694 GPR images will improve the pre-training effect of the SA-DenseCL.

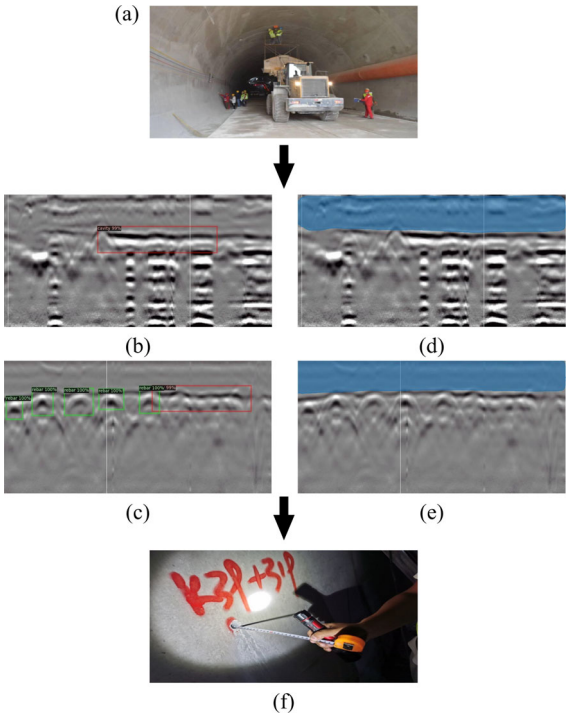


FIGURE 15 Implementation process and prediction results of tunnel on-site verification: (a) on-site data collection, (b–e) recognitions of reinforcement bar, void, and secondary lining, and (f) verifying the predicted results by drilling and measurement.

3.5 | Field tests

In order to further verify the effectiveness of the proposed SA-DenseCL and workflow for practical applications, a field test was conducted with drilling verifications on some constructed highway tunnels in Yunnan Province, China.

Figure 15 presents the on-site verification process and results, where GPR data collection (Figure 15a), autonomous identifications by deep learning network (Figure 15b–e), and drill hole verification (Figure 15f) are exemplified. In Figure 15b,c, there are five reinforcement bar signals and two void signals marked by the proposed network, respectively. The opening holes in the corresponding positions of the tunnel lining verify the identified targets. In Figure 15d,e, the secondary lining areas are identified and masked in the GPR images by the network, and the thickness of the secondary lining is estimated as 18 and 20 cm, respectively, by the determined wave velocity in the concrete. The measured thickness of the corresponding secondary lining after drilling is 20 and 22 cm, showing an acceptable agreement between the estimated and the true values of the secondary lining.

There are a total of 10 test points drilled and verified. All signals recognized as reinforcement bars are proven to be correct. The voids identified by the network and verified by drilling are shown in Table 10. It can be seen that all the drilled points, except for the eighth point, verify that voids are correctly identified by the network. Figure 16a presents the misrecognized void in the GPR image. The underlying reason is that the densely packed reinforcement bars lead to a continuous superposition of reflected signals, which presents a certain similarity to the void reflected signals. Table 11 shows the comparative validations of the second lining thickness by the proposed network and the drilling measurements. It can be seen that the majority of the errors

TABLE 10 The predicted and verified results of voids in the drilled points

Drilled point	Predicted void	Verified void
No. 1	True	True
No. 2	True	True
No. 3	True	True
No. 4	True	True
No. 5	True	True
No. 6	True	True
No. 7	True	True
No. 8	True	**False**
No. 9	True	True
No. 10	True	True

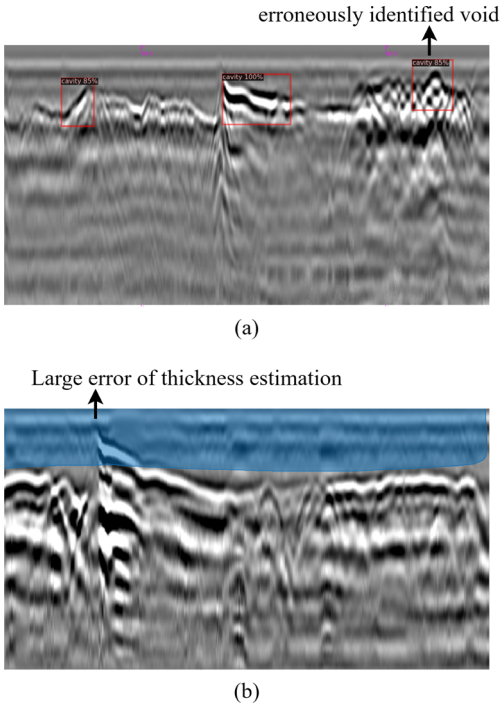


FIGURE 16 Examples of misrecognition of void signal and large error between the predicted and measured thickness of the second lining in the GPR images: (a) the misrecognition of void at No. 8 drilled point and (b) the large error of the thickness estimation at No. 6 drilled point.

TABLE 11 The predicted and verified results of thickness of the second lining in the drilled points

Drilled point	Predicted thickness (cm)	Verified thickness (cm)	Error (cm)
No. 1	18	20	2
No. 2	17	15	2
No. 3	17	19	2
No. 4	14	11	3
No. 5	17	18	1
No. 6	**21**	**11**	**10**
No. 7	16	15	1
No. 8	20	22	2
No. 9	15	14	1
No. 10	22	24	2

between the predicted and measured thickness of the second lining are not more than 3 cm. However, there is still a verification point presenting considerable error (sixth point) as seen in Figure 16b.

The incorrect identifications of the targets mainly stem from the limited number of labeled samples, which constrains the identification capacity of the network for more complicated scenarios. The erroneous estimation of the secondary lining thickness is caused by the experience or the rigorousness of the labeling operators.

4 | DISCUSSION

The experimental results in Section 3.3 show that the SA-DenseCL-based deep learning workflow is superior to the original DenseCL and supervised pre-training-based method in GPR tunnel lining inspection. Compared with supervised pre-training, the SA-DenseCL as a self-supervised learning algorithm has an excellent performance in pre-training the backbone for feature extraction through unlabeled GPR images, which brings great advantages in the following fine-tuning with a limited number of labeled data (He et al., 2020). To be more specific, in the previous supervised learning-based deep learning architecture for GPR target identification, the initial parameters of the backbone of Mask R-CNN are obtained through the supervised learning based on natural image datasets, such as ImageNet (Lei et al., 2019). Even with transfer learning, there are still differences existing in the features between the natural images and GPR images, and thus the performance of Mask R-CNN depends heavily on the number of labeled samples used for fine-tuning. In the self-supervised learning-based architecture, SA-DenseCL is used for pre-training, a large number of unlabeled GPR images are used for representation learning, and the extracted feature representations are closer to the features of the GPR images to be identified by Mask R-CNN.

Essentially, in the representation learning of supervised learning, an image is classified into either a positive or a negative sample according to whether there are targets existing in the image. This mechanism brings about a drawback in that only the images containing targets are learned. On the contrary, self-supervised learning uses two different data augmentations to generate twice as many views of the input images and construct two groups of positive samples and some negative samples by justifying whether the two sets of views come from the same input images. By this means, the backbone is able to focus on all the features in the images so that the target signal features are accurately distinguished from other signals and noises.

Compared with DenseCL, the improved self-supervised learning algorithm–SA-DenseCL—can enhance positive gain effects in the backbone of Mask R-CNN by adding the so-called relevant projection head module into the conventional DenseCL architecture. The underlying consideration is based on the fact that GPR images contain some additional relevant information between the adjacent traces of waveforms, compared with conventional images. Therefore, we propose the relevant projection head to capture this part of relevant information, by which the self-attention module is specially used to learn the similarity information between different sequences of GPR traces. However, the weights of the corresponding loss item—relevant loss— determine the gain effects. We obtained a relatively optimal weight of 0.2 through the ablation experiment in Section 3.4. These results indicate that the added relevant projection head enables the backbone to extract features special to GPR images.

In Section 3.5, the results of the field test prove the effectiveness of the proposed method in practical tunnel lining inspections with GPR data. The SA-DenseCL can only use the easily available unlabeled data for representation learning, reducing the dependence of the network on the number of labeled data during the fine-tuning of Mask R-CNN. The proposed workflow does not depend on a large number of simulation data or existing image databases, and a limited number of field data, with parts labeled, are able to train the network for accurately extracting the features of GPR images. Therefore, the proposed workflow has a good performance when applied to the practical tunnel lining inspection.

There exist some limitations in this study. Considering that collecting GPR data from tunnels in service will block the traffic, the GPR data used in this study were mainly obtained from the newly constructed tunnels, where a limited number of void defects are presented inside the tunnel lining. However, in the operational tunnels, there exist some different kinds of defects in the lining,

exemplified by cracks, leakages, and water-bearing cavities or fractures, especially for the aging tunnels. Therefore, the trained model may present decayed identification effects when applied to the aging tunnel lining inspection. In addition, the labels of GPR images are not real ground truth but are determined by experienced engineers, which causes the identification accuracy of the targets to be partly dependent on the elaborate degree of the labeling process. Laboratory settings seem to be the best way to obtain the ground truth, which allow for exactly obtaining a priori information of the targets. Therefore, in future work, it is considered to design some reinforced concrete specimens with a variety of defects inside and to collect GPR images with ground truth of targets for more accurate testification.

5 | CONCLUSION

A self-supervised learning-based deep learning framework for autonomous target identifications and estimation in tunnel lining inspection with GPR images is described. In the network architecture, SA-DenseCL, an improved self-supervised algorithm is proposed by introducing a self-attention mechanism to the DenseCL algorithm for pre-training the backbone with unlabeled GPR images; then, limited number of labeled GPR images are used to fine-tune the backbone of the Mask R-CNN to accurately estimate the distribution of reinforcement bars, the locations of void defects, and the thickness of secondary lining. By comparative analyses and field verification, the following conclusions are arrived:

1. Compared with supervised learning, self-supervised learning enables the backbone to extract more comprehensive and accurate features from many unlabeled GPR images thereby improving the recognition accuracy of the target signals. Specifically, the AP of the proposed SA-DenseCL outperforms that of the supervised learning by 1.17%, 8.49%, and 3.44%, respectively, in the recognition of reinforcement bar signals, void signals, and thickness of secondary lining.
2. The improved self-supervised learning algorithm, SA-DenseCL, enables the backbone to extract additional spatial correlation information between the sequences of GPR traces by adding the proposed relevant projection head, and thus increases the performance, compared with the conventional DenseCL algorithm.
3. The proposed workflow is successfully applied to the GPR inspection data in practical tunnel lining. The SA-DenseCL greatly reduces the dependence of deep learning on the number of labeled GPR data and promotes the applicability of deep learning in actual tunnel lining inspection.

Future work will pay more attention to the identification and classification of defects in tunnel lining. For example, cracks, air-bearing cavities, and water-bearing cavities are to be accurately identified and categorized. Furthermore, the sizes, shapes, and directions of these defects are also expected to be estimated. From the angles of deep learning, some newly developed algorithms may enhance the learning capacity in quantitatively recognizing and estimating defects, typically represented by neural dynamic classification algorithm (Rafiei & Adeli, 2017), dynamic ensemble learning algorithm (Pereira et al., 2020), and fine element machine for fast learning (Alam et al., 2020).

ACKNOWLEDGMENTS

This research was funded by the National Natural Science Foundation of China (41974165, 42241131), Guangdong Provincial Key Laboratory of Geophysical High-resolution Imaging Technology (2022B1212010002), CRSRI Open Research Program (CKWV2021883/KY), Yunnan Provincial Science and Technology Key R&D Program (202203AA080006), and Russian Science Foundation (211900043).

REFERENCES

Adeli, H. (2002). Neural networks in civil engineering: 1989–2000. *Computer-Aided Civil and Infrastructure Engineering*, *16*(2), 126–142.

Alam, K. M. R., Siddique, N., & Adeli, H. (2020). A dynamic ensemble learning algorithm for neural networks. *Neural Computing and Applications*, *32*, 8675–8690.

Annan, A. P. (2003). *GPR principles, procedures and applications.* Sensors and Software.

Balaguer, C., Montero, R., Victores, J. G., Martínez, S., & Jardón, A. (2014). Towards fully automated tunnel inspection: A survey and future trends. *ISARC. 31st Proceedings of the International Symposium on Automation and Robotics in Construction*, Sydney, Australia.

Barpi, F., & Peila, D. (2012). Influence of the tunnel shape on shotcrete lining stresses. *Computer-Aided Civil and Infrastructure Engineering*, *27*(4), 260–275.

Bergeson, W., & Ernst, S. L. (2015). Tunnel operations, maintenance, inspection, and evaluation (TOMIE) manual. *(No. FHWA-HIF-15-005)*. United States Federal Highway Administration.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., & Hinton, G. (2020). Big self–supervised models are strong semi–supervised learners. *Advances in neural information processing systems*, *33*, 22243–22255.

Chen, X., Fan, H., Girshick, R., & He, K. (2020). Improved baselines with momentum contrastive learning. ArXiv:2003.04297.

Chiaia, B., Marasco, G., & Aiello, S. (2022). Deep convolutional neural network for multi-level non-invasive tunnel lining assessment. *Frontiers of Structural and Civil Engineering*, *16*(2), 214–223.

Gao, J., Yuan, D., Tong, Z., Yang, J., & Yu, D. (2020). Autonomous pavement distress detection using ground penetrating radar and region–based deep learning. *Measurement*, *164*, 108077.

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. *Proceed-*

ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA (pp. 9729–9738).

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R–CNN. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy (pp. 2961–2969).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV (pp. 770–778).

Hjelm, R. D., Fedorov, A., Lavoie–Marchildon, S., Grewal, K., Bachman, P., Trischler, A., & Bengio, Y. (2019). Learning deep representations by mutual information estimation and maximization. ArXiv, abs/1808.06670.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*, Lake Tahoe, NV.

Lai, W. W., Dérobert, X., & Annan, P. (2017). A review of ground penetrating radar application in civil engineering: A 30–year journey from Locating and Testing to Imaging and Diagnosis. *NDT&E International*, *96*, 58–78.

Lei, W., Hou, F., Xi, J., Tan, Q., Xu, M., Jiang, X., Liu, G., & Gu, Q. (2019). Automatic hyperbola detection and fitting in GPR B-scan image. *Automation in Construction*, *106*, 102839.

Li, X., Liu, H., Zhou, F., Chen, Z., Giannakis, I., & Slob, E. (2022). Deep learning–based nondestructive evaluation of reinforcement bars using ground–penetrating radar and electromagnetic induction data. *Computer-Aided Civil and Infrastructure Engineering*, *37*(14), 1834–1853.

Marasco, G., Rosso, M. M., Aiello, S., Aloisio, A., Cirrincione, G., Chiaia, B., & Marano, G. C. (2022). Ground penetrating radar fourier pre-processing for deep learning tunnel defects' automated classification. *Engineering Applications of Neural Networks: 23rd International Conference, EAAAI/EANN 2022, Chersonissos, Crete, Greece* (pp. 165–176).

Menendez, E., Victores, J. G., Montero, R., Martínez, S., & Balaguer, C. (2018). Tunnel structural inspection and assessment using an autonomous robotic system. *Automation in Construction*, *87*, 117–126.

Montero, R., Victores, J. G., Martinez, S., Jardón, A., & Balaguer, C. (2015). Past, present and future of robotic tunnel inspection. *Automation in Construction*, *59*, 99–112.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. *International Machine Learning Society*, *2010*, Haifa, Israel (pp. 807–814).

Oord, A. V. D., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. ArXiv–1807.03748.

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359.

Parkinson, G., & Ékes, C. (2008). Ground penetrating radar evaluation of concrete tunnel linings. *12th International Conference on Ground Penetrating Radar*, Birmingham, UK (p. 11).

Pereira, D. R., Piteri, M. A., Souza, A. N., Papa, J. P., & Adeli, H. (2020). FEMa: A finite element machine for fast learning. *Neural Computing and Applications*, *32*, 6393–6404.

Pham, MT., & Lefèvre, S. (2018). Buried object detection from B–scan ground penetrating radar data using Faster–RCNN. *IGARSS 2018–2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia, Spain (pp. 6804–6807).

Qin, H., Zhang, D., Tang, Y., & Wang, Y. (2021). Automatic recognition of tunnel lining elements from GPR images using deep convolutional networks with data augmentation. *Automation in Construction*, *130*, 103830.

Rafiei, M. H., & Adeli, H. (2016). A novel machine learning model for estimation of sale prices of real estate units. *Journal of Construction Engineering and Management*, *142*(2), 04015066.

Rafiei, M. H., & Adeli, H. (2017). A new neural dynamic classification algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(12), 3074–3083.

Rafiei, M. H., Gauthier, L. V., Adeli, H., & Takabi, D. (2023). Self-supervised learning for electroencephalography. IEEE Transactions on Neural Networks and Learning Systems. Advance online publication. https://doi.org/10.1109/TNNLS.2022.3190448

Rafiei, M. H., Khushefati, W. H., Demirboga, R., & Adeli, H. (2017). Supervised deep restricted Boltzmann machine for estimation of concrete. *ACI Materials Journal*, *114*(2), 237.

Rosso, M., Marasco, G., Tanzi, L., Aiello, S., Aloisio, A., Cucuzza, R., Chiaia, B., Cirrincione, G., & Marano, G. (2022). Advanced deep learning comparisons for non-invasive tunnel lining assessment from ground penetrating radar profiles. *ECCOMAS Congress 2022–8th European Congress on Computational Methods in Applied Sciences and Engineering*, Oslo, Norway.

Rosso, M. M., Marasco, G., Aiello, S., Aloisio, A., Chiaia, B., & Marano, G. C. (2023). Convolutional networks and transformers for intelligent road tunnel investigations. *Computers & Structures*, *275*, 106918.

Sajedi, S. O., & Liang, X. (2021). Uncertainty-assisted deep vision structural health monitoring. *Computer-Aided Civil and Infrastructure Engineering*, *36*(2), 126–142.

Sirca, G. F. Jr., & Adeli, H. (2018). Infrared thermography for detecting defects in concrete structures. *Journal of Civil Engineering and Management*, *24*(7), 508–515.

Suda, T., Tabata, A., Kawakami, J., & Suzuki, T. (2004). Development of an impact sound diagnosis system for tunnel concrete lining. *Underground Space for Sustainable Urban Development. Proceedings of the 30th ITA-AITES World Tunnel Congress*, Singapore (pp. 328–329).

Uhrin, M., Brierley, R., Talliss, M., & Marchand, E. (2017). Amalgamation of tunnel secondary lining and first stage concrete at Whitechapel crossover. *In Crossrail Project: Infrastructure design and construction* (pp. 91–106).

Van Hauwermeiren, W., Filipan, K., Botteldooren, D., & De Coensel, B. (2022). A scalable, self-supervised calibration and confounder removal model for opportunistic monitoring of road degradation. *Computer-Aided Civil and Infrastructure Engineering*, *37*(13), 1703–1720.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, Long Beach, CA.

Wang, H., Huang, H., Feng, Y., & Zhang, D. (2018). Defects of concrete linings of road tunnels in China. *Journal of Risk Uncertainty in Engineering Systems. Part A: Civil Engineering*, *4*(4), 04018041.

Wang, J., Zhang, J., Cohn, A. G., Wang, Z., Liu, H., Kang, W., Jiang, P., Zhang, F., Chen, K., Guo, W., & Yu, Y. (2022). Arbitrarily–oriented tunnel lining defects detection from ground penetrating radar images using deep convolutional neural networks. *Automation in Construction*, *133*, 104044.

Wang, X., Zhang, R., Shen, C., Kong, T., & Li, L. (2021). Dense contrastive learning for self–supervised visual pre–training. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN (pp. 3024–3033).

Wu, Y., Kirillow, A., Massa, F., Lo, W., & Girshick (2019). *Detectron2*. https://github.com/facebookresearch/detectron2

Xue, Y., & Li, Y. (2018). A fast detection method via region–based fully convolutional neural networks for shield tunnel lining defects. *Computer-Aided Civil and Infrastructure Engineering*, *33*(8), 638–654.

Zhang, W., Sun, K., Lei, C., Zhang, Y., Li, H., & Spencer, B. F. Jr. (2014). Fuzzy analytic hierarchy process synthetic evaluation models for the health monitoring of shield tunnels. *Computer-Aided Civil and Infrastructure Engineering*, *29*(9), 676–688.

Zhou, Z., Zhang, J., & Gong, C. (2022). Automatic detection method of tunnel lining multi-defects via an enhanced You Only Look Once network. *Computer-Aided Civil and Infrastructure Engineering*, *37*(6), 762–780.

Zhu, M., Zhu, H., Guo, F., Chen, X., & Ju, J. W. (2021). Tunnel condition assessment via cloud model-based random forests and self-training approach. *Computer-Aided Civil and Infrastructure Engineering*, *36*(2), 164–179.

How to cite this article: Huang, J., Yang, X., Zhou, F., Li, X., Zhou, B., Lu, S., Ivashov, S., Giannakis, I., Kong, F., & Slob, E. (2023). A deep learning framework based on improved self-supervised learning for ground-penetrating radar tunnel lining inspection. *Computer-Aided Civil and Infrastructure Engineering*, 1–20. https://doi.org/10.1111/mice.13042