Improved Mask R-CNN with Distance Guided Intersection over Union for GPR Signature Detection and Segmentation

Feifei Hou[1,2], Wentai Lei[1], Shuai Li[2], Jingchun Xi[1], Mengdi Xu[1], Jiabin Luo[1]
[1] School of Computer Science and Engineering, Central South University, Changsha, China (e-mail: houfeifei@csu.edu.cn; leiwentai@csu.edu.cn; xijingchun@csu.edu.cn; xumengdi@csu.edu.cn; 194711013@csu.edu.cn).
[2] Department of Civil and Environmental Engineering, University of Tennessee, Knoxville, USA (e-mail: sli48@utk.edu).

Abstract

Ground penetrating radar (GPR) has been used for non-destructive inspection of civil infrastructure systems such as bridges and pipelines. Manually extracting useful data from a large amount of non-intuitive GPR scans is tedious and error-prone. To address this challenge, a generalizable end-to-end framework is developed and implemented to simultaneously detect and segment object signatures in GPR scans. The proposed approach improves the Mask Region-based Convolutional Neural Network (R-CNN) by incorporating a novel distance guided intersection over union (*DGIoU*) as a new loss function for detection and segmentation. The *DGIoU* considers the center distance between two bounding boxes and overcomes the weakness of intersection over union (*IoU*) in training and evaluation. In addition, a new method is proposed to extract data points from the segmented mask patches containing both object signatures and background noises. The extracted data points can be further processed for object localization and characterization. Experiments were conducted using GPR scans collected from a concrete bridge deck. The hyperbolic signatures of rebars can be accurately detected and segmented using the proposed method. It was demonstrated that using *DGIoU* improves the regression effect of bounding box and mask. The improved Mask R-CNN achieved an average accuracy (AP) of 58.64% and 47.64% for the detection and segmentation task, respectively.

Keywords

Ground Penetrating Radar (GPR); Deep Learning (DL); Civil Infrastructure; Mask R-CNN; Detection and Segmentation; Intersection over Union (*IoU*)

1. Introduction

Ground penetrating radar (GPR), a non-destructive testing and evaluation technology, has been widely used in civil infrastructure inspection. Examples include underground asset mapping [1-4], rebar localization and corrosion inspection [5-7], concrete bridge deck deterioration assessment [8-10], sinkhole detection [11,12], and construction quality control [13]. The impulse GPR (hereafter GPR) can emit short electromagnetic (EM) pulses into the subsurface, propagating in a shape of a cone, and receive the signals reflected from the objects that have different EM properties from the subsurface medium [14]. The reflected signals will form a GPR scan, which is the main source for data visualization, analysis, and interpretation. The object signatures in a GPR scan can be of various shapes, depending on the object geometries and scanning trajectories [15]. The most common signature in a GPR scan is of a hyperbolic shape. For example, the signatures of underground utilities, rebars in bridge deck, sinkholes under pavement, voids in concrete may present as hyperbolic reflections in GPR scans. Detecting and segmenting the signatures of objects in GPR scans is an essential step to retrieve information. For instance, estimating the depth and dimension of an underground utility line requires the extraction of data points from the hyperbolic signature. However, given the large amount of data, manual processing of GPR scans is labor-intensive and error-prone.

There are two challenges in automating this process. The first challenge is the automatic detection of object signatures in GPR scans. Due to the signal interference among multiple adjacent objects in the subsurface, processing GPR scans are not amenable to the traditional methods such as edge detection [16], Hough transformation [17-19], and least squares method [20]. In addition, the distortion and incompleteness of hyperbolic signatures resulting from the subsurface objects may further impose difficulties in recognizing the object instances. The second challenge is the automatic segmentation of object signatures from noisy background in GPR scans. The existing methods are not adequate to segment the complete object signatures, which may hinder subsequent processing and interpretation. A novel method is needed to automatically detect and segment object signatures from noisy GPR scans. Therefore, in this study, a deep learning (DL) based framework is proposed to detect and segment object signatures, and a rule-based method is developed to extract data points on the segmented signatures. The integration of detection and segmentation in a unified framework will help advance the information extraction from coarse box-level instance recognition to precise pixel-level classification.

Fig. 1 presents the proposed framework. The contribution of this work is twofold. First, we developed a new computation for *IoU*, i.e. distance guided Intersection over Union (*DGIoU*), and incorporated it as a new metric and new loss function into the Mask R-CNN framework [21]. The existing *IoU* computation methods [22-23] only consider the sizes of bounding boxes (Bboxes) and neglect their locations. The proposed method can overcome the limitation and keep the discrepancy between predicted Bbox and real Bbox to a minimum during the training phase. Using the enhanced Mask R-CNN framework, GPR signatures can be automatically detected and the corresponding mask patches can be obtained. Second, we proposed a novel method to extract data points from the detected signatures, which can streamline subsequent analyses, such as hyperbola fitting, peak localization, and object depth and dimension estimation. Table 1 lists all the abbreviations and notations to ease the illustration.

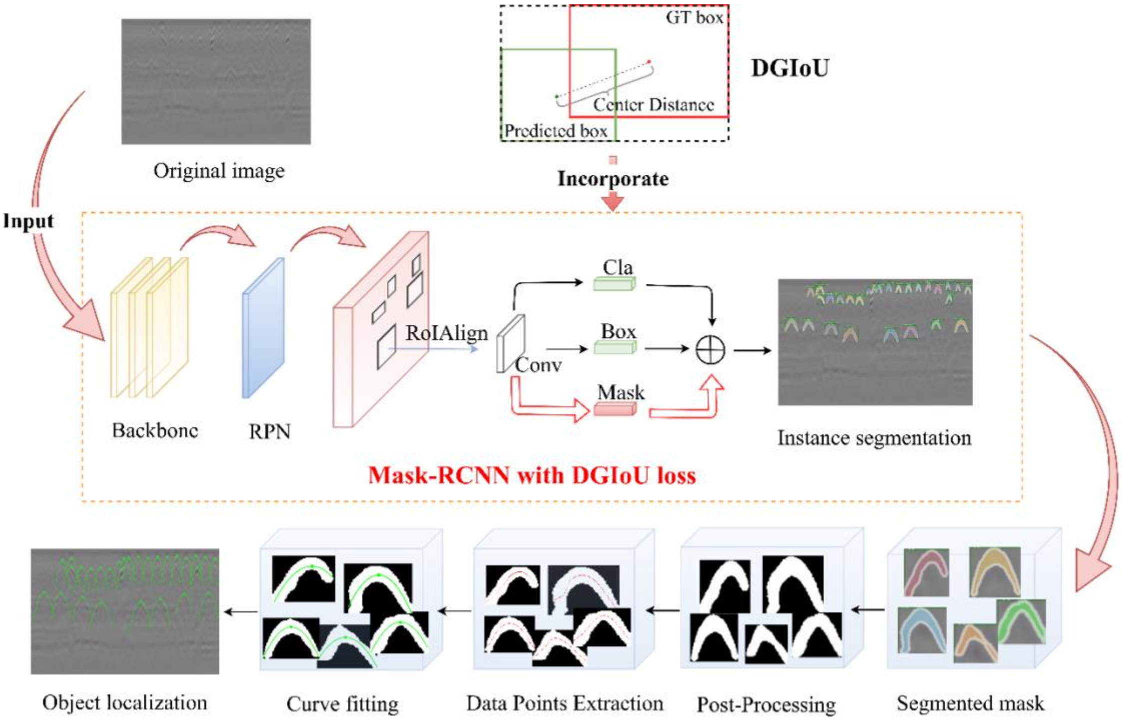


Fig. 1. GPR Signature Detection and Segmentation Framework.

Table 1. Summary of abbreviations and notations.

Notation(s) or Abbreviation	Description
AP	Average Precision
Bbox	Bounding Box
CD	Center Distance
DL	Deep Learning
EM	Electromagnetic
GPR	Ground Penetrating Radar
ROI	Region of Interest
RPN	Regional Proposal Network
IoU	Intersection over Union
$GIoU$	Generalized IoU
$DGIoU$	Distance Guided IoU
B^p , B^g , B^c	Predicted Bbox, Ground Truth Bbox, Smallest Enclosing Bbox
A^p , A^g , A^c	Area of B^p , B^g , B^c
$\boldsymbol{C}$	Smallest Enclosing box of Bbox ***A*** and Bbox ***B***
$\boldsymbol{U}$	Union of Bbox ***A*** and Bbox ***B***
$[k1, k2, k3]$	Coefficient matrix
L_{IoU} , L_{IoU}'	*IoU* loss in Bbox regression stage, in RPN stage
L_{GIoU} , L_{GIoU}'	*GIoU* loss in Bbox regression stage, in RPN stage
L_{DGIoU} , L_{DGIoU}'	*DGIoU* loss in Bbox regression stage, in RPN stage
$Lflag$	Searching to left
$Rflag$	Searching to right
$temp$	Step length for searching to left/right
P	Initial midpoint set
p	Peak of P
P'	Updated set
F	Final data point set for curve fitting

2. Literature Review

GPR has been used to detect, locate, and characterize buried objects [24-30]. Different methods have been developed to process different formats of GPR data (A-scan, B-scan, and C-scan) for various applications such as quality assessment, 2D target detection, and 3D mapping [31]. For example, A-scans were used in [32-35], B-scans were used in [36], and C-scan were used in [37]. For object signature detection, numerous studies focused on processing B-scans. Existing object detection methods are mainly categorized into conventional feature engineering-based methods and DL-based methods. Conventional feature engineering-based methods first extract handcrafted features from each patch, then the features are classified by a classifier. For example, in [15], an edge detection was performed to acquire simplified object signatures, which were then fed into a simple neural network. A detection technique is proposed in [38] for bridge condition assessment and deterioration analysis. In this method, the similarity between a GPR image patch and a hyperbolic signature template is computed. Then the Fast Fourier Transform (FFT) was integrated with hyperbola fitting to extract rebar peaks and signature parameters. However, the template-based method is not reliable for detecting object signatures in the presence of clutters. A face detection algorithm, namely Viola-Jones (VJ) [39], was first used in work [18] to pick the regions containing hyperbolic signatures. By narrowing down the region of interest (ROI) and using Hough transform, objects parameters such as hyperbola peak and EM wave velocity can be obtained. The study [40] first calculated the histogram of oriented gradient (HOG) features, and then fed these features into a support vector machine (SVM) for training. The trained model was used to classify the regions with and without object signatures.

Recently, DL-based method has gained its momentum in analyzing GPR data. Deep network has been used in classification tasks [41-43], box prediction tasks [44-48], and semantic segmentation tasks [49-52]. The deep network advances the image-level prediction [44] to box-level prediction [45], semantics-level prediction [49] and mask-level prediction [21]. As demonstrated in studies [53-55], DL methods do not require the traditional “feature engineering” step, and can directly extract the features from GPR scans. Moreover, revealed by a lot of studies, the deep networks outperform traditional feature engineering under certain circumstances. The method proposed in [36] used an object detection model to classify the buried targets and detect the target hyperbola region. This method was integrated with the double cluster seeking estimate (DCSE) to segment hyperbolic clusters, and with a novel column-based transverse filter points (CTFP) method to automate the task of extracting hyperbolic data points. The DCSE algorithm, and other recent clustering methods such as open scan clustering algorithm (OSCA) [56] and column-connection clustering (C3) method [57] were applied to segment hyperbolic signatures for data extraction. However, there are two limitations in the prior studies. First, there lacks a method to achieve simultaneous detection and segmentation of object signatures in GPR scans. Second, most existing methods are not reliable and accurate in extracting the data points from the segmented object signatures.

3. Distance Guided *IoU* in Mask R-CNN

This section first presents the *DGIoU* as a new metric to evaluate the similarity between two Bboxes. In addition, the *DGIoU* is integrated into the Mask-RCNN framework as a new loss function to enhance its performance.

3.1. Overview of Mask R-CNN and IoU Computation

The Mask R-CNN adds an additional mask prediction branch into the Faster R-CNN framework [44]. The Mask R-CNN architecture consists of three phases. The first phase is to input an original image into a pretrained neural network and obtain the corresponding feature map. In the second phase, the Regional Proposal Network (RPN) generates a large amount of proposals that are not related to the object category. The final R-CNN stage uses RoIAlign to extract features for each proposal and conducts three tasks: proposal classification, Bbox regression, and mask prediction.

As an instance level segmentation algorithm, the Mask R-CNN relies on accurate Bbox regression. Studies have shown that *IoU* could help refine Bbox and proven that improving *IoU* helps to reduce object uncertainty and increase detection accuracy. For most object detection tasks, *IoU* is a well-adopted evaluation metric and is used to compare the similarity of two arbitrary shapes and divide true positives and false positives. *IoU* is scale invariant, and thus is leveraged to verify the properties of object detection [61,62] and segmentation [58-61]. For axis aligned two-dimensional (2D) Bboxes, many frameworks used a surrogate loss of *IoU* to conduct the task of Bbox regression. For example, the authors in [63] introduced an *IoU* loss computation metric used for Bbox prediction by regressing the boundaries information of a predicted box. Their method converges fast and demonstrates robust performance in accurate and efficient localization of objects with different scales. Lachlan et al. [64] developed a bounded *IoU* loss for *IoU* maximization while keeping *IoU*'s good convergence properties. In [23], Rezatofighi et al. proposed a generalized version of *IoU* (*GIoU*) function to overcome the limitation of *IoU* under the circumstance of non-overlapping bounding boxes. Considering two bounding boxes ***A*** and ***B***, the smallest bounding box that encloses both ***A*** and ***B*** is defined as ***C***. ***C*** usually has the same scale with ***A*** and ***B***. The *IoU* and *GIoU* can be computed as in Eq. (1) and Eq. (2), where ***U*** is defined as the union of ***A*** and ***B***: $U = A \bigcup B = Area_A + Area_B - Area_{overlap}$.

$$IoU = \frac{A \bigcap B}{A \bigcup B} \tag{1}$$

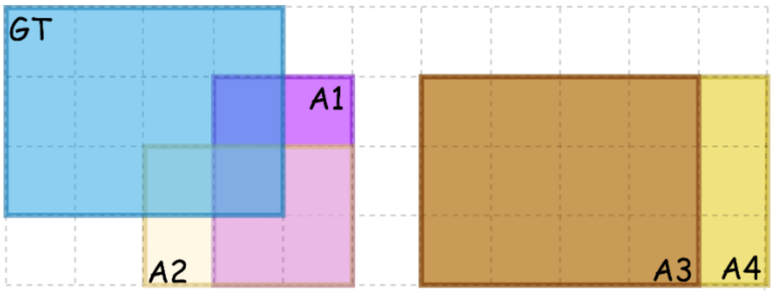
$$GIoU = IoU - \frac{C - A \bigcup B}{C} = IoU - \frac{C - U}{C} \tag{2}$$

3.2 Distance Guided IoU

However, *IoU* and *GIoU* value do not fully reflect the quality of predicted Bbox. For instance, for the simple overlapped scenario in the left part of Fig. 2, the predicted Bbox A1 has exactly same IoU and *GIoU* as the predicted Bbox A2. However, in fact, the center position of Bbox A2 is closer to ground truth (GT) Bbox than that of Bbox A1, because the center distance between A1/A2 and GT is $\sqrt{5} / \sqrt{4.5}$, respectively. For another non-intersected case in the right part of Fig. 2, it is obvious that Bbox A3 is a better choice than Bbox A4. But this observation cannot be reflected by computing *IoU* and *GIoU*. They have the same *IoU*. Moreover, the *GIoU* of Bbox A4 is higher than that of Bbox A3. The above observations indicate that *IoU* and *GIoU* can only consider the sizes but not the locations of two axis-aligned rectangles. In fact, several pairs of Bboxes with the same degree of overlap may have different center distance and therefore have different objective values. To solve this critical issue, we propose an extended version of *GIoU*, namely *DGIoU*, which considers an additional element called center distance in *GIoU.* Center distance information of two boxes is important for accurate object detection. The *DGIoU* redistributes the proportion of three elements: *IoU*, ***C*** excluding ***U*** and divide by ***C,*** and center distance (*CD*). The center distance is computed by the l2-norm between predicted Bbox and GT Bbox. The coefficient matrix $[k1, k2, k3]$ represents the weights of the above elements. Algorithm 1 gives the calculation of *DGIoU*.

Algorithm 1: *DGIoU*

```
Input  : Two rectangle Bboxes: A and B
Output: DGIoU
1. Find the smallest enclosing Bbox C of A and B
2. Find each center coordinate for two Bboxes: $C_A$ , $C_B$
3. Compute the center distance of A and B: $CD = \|C_A - C_B\|$
4. Compute $DGIoU = k1 * IoU - k2 * \frac{|C - U|}{|C|} - k3 * CD$
```



A1: IoU=0.1250 GIoU=-0.0750 DGIoU=-0.2711
A2: IoU=0.1250 GIoU=-0.0750 DGIoU=-0.2596
A3: IoU=0 GIoU=-0.4000 DGIoU=-0.2883
A4: IoU=0 GIoU=-0.3864 DGIoU=-0.3485

Fig. 2. Similarity comparison of predicted Bbox and GT Bbox using *IoU*, *GIoU* and *DGIoU*.

To maintain the properties of *GIoU*, we set k_1 and k_2 to 0.9 and 0.8, respectively. The *DGIoU* with small k_1 and k_2 values cannot reflect the properties of *GIoU*, while a large value will neglect distance information. For k_3, several values [0.05,0.1,0.15,0.2] are compared to select an appropriate one for our algorithm. The variations in average precision (AP) against different k_3 values for both box and mask tasks are plotted in Fig. 3. To balance the performance of box task and mask task, we set k_3 to 0.1 for *DGIoU* in this paper.

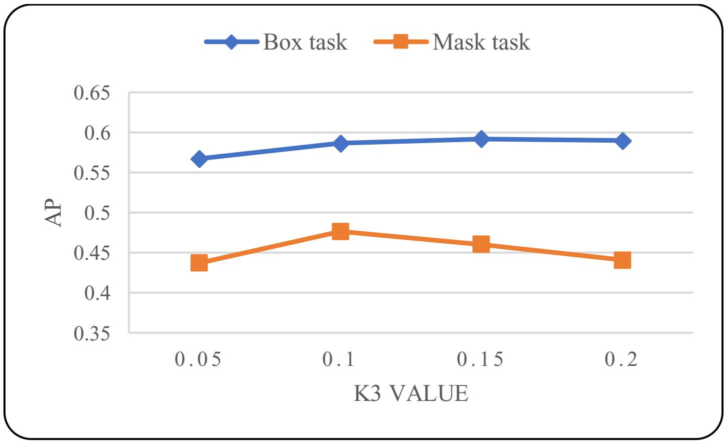


Fig. 3. AP value against different k3 evaluation curve of the proposed *DGIoU*.

3.3. DGIOU as loss for Bbox regression

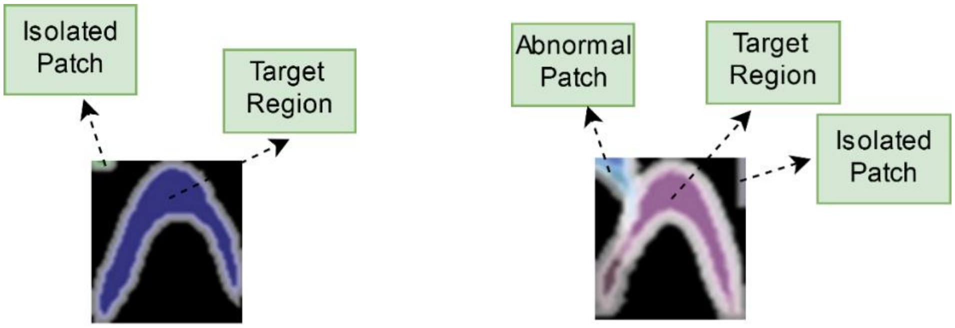
The task of detecting 2D object signatures in GPR scans can be converted into a task of comparing two axis-aligned Bboxes. Both the intersection and the smallest enclosing object have rectangular shapes. The center point is computed by using the mean value of two adjacent vertices coordinate in each Bbox, which can be obtained by adding this mean value to a minimum vertical coordinate. After obtaining the center coordinates of the two Bboxes separately, l2-norm is used to compute their center distance. Hence, a high-quality solution is proposed to calculate *DGIoU* detailed in Alg. 2. Similar to the *GIoU* loss in [23], the proposed *DGIoU* can be directly used as a loss into network. *DGIoU* as loss both in final Bbox regression stage (L_{DGIoU}) and in RPN stage (L_{DGIoU}') can be used to optimize Mask R-CNN framework. Since *IoU* and G*IoU* ignore the center coincidence between two Bboxes, the training quality and convergence rate will be limited. Compared to the above two criteria, the proposed *DGIoU* has considered this important element. It has three merits: 1) *DGIoU* has a strong correlation with *GIoU*; 2) *DGIoU* can further judge which one is the optimal solution when overlapped area levels between several predicted Bbox and GT box are the same; 3) *DGIoU* defines the representation of center distance as location space and incorporates it into size space. To this end, *DGIoU* is able to represent the real similarity among Bboxes.

```
Algorithm 2: DGIoU as Bbox loss and RPN loss
Input : Predicted Bbox $B^p$ and GT Bbox $B^g$ upper-left and lower-right coordinates:
        $B^p = \left(x_1^p, y_1^p, x_2^p, y_2^p\right)$ , $B^g = \left(x_1^g, y_1^g, x_2^g, y_2^g\right)$ .
Output: $L_{DGIoU}$
1.  $B^p$ needs to meet $x_2^p > x_1^p$ and $y_2^p > y_1^p$ :
        $\hat{x}_1^p = \min\left(x_1^p, x_2^p\right)$ , $\hat{x}_2^p = \max\left(x_1^p, x_2^p\right)$ ,
        $\hat{y}_1^p = \min\left(y_1^p, y_2^p\right)$ , $\hat{y}_2^p = \max\left(y_1^p, y_2^p\right)$
2.  Area of $B^g$ : $A^g = \left(x_2^g - x_1^g\right) \times \left(y_2^g - y_1^g\right)$
3.  Area of $B^p$ : $A^p = \left(\hat{x}_2^p - \hat{x}_1^p\right) \times \left(\hat{y}_2^p - \hat{y}_1^p\right)$
4.  Intersection $I$ between $B^p$ and $B^g$ :
        $x_1^I = \max\left(\hat{x}_1^p, x_1^g\right)$ , $x_2^I = \min\left(\hat{x}_2^p, x_2^g\right)$
        $y_1^I = \max\left(\hat{y}_1^p, y_1^g\right)$ , $y_2^I = \min\left(\hat{y}_2^p, y_2^g\right)$
        $I = \begin{cases} \left(x_2^I - x_1^I\right) \times \left(y_2^I - y_1^I\right) & if \quad x_2^I > x_1^I, y_2^I > y_1^I \\ 0 & otherwise. \end{cases}$
5.  Finding the coordinates of smallest enclosing box $B^c$ :
        $x_1^c = \min\left(\hat{x}_1^p, x_1^g\right)$ , $x_2^c = \max\left(\hat{x}_2^p, x_2^g\right)$
        $y_1^c = \min\left(\hat{y}_1^p, y_1^g\right)$ , $y_2^c = \max\left(\hat{y}_2^p, y_2^g\right)$
6.  Calculating area of $B^c$ : $A^c = \left(x_2^c, x_1^c\right) \times \left(y_2^c, y_1^c\right)$
7.  Finding center coordinates of $B^p$ and $B^g$ :
        $C^p = \left(C_x^p, C_y^p\right)$ , $C^g = \left(C_x^g, C_y^g\right)$
        $C_x^p = \left(\frac{\hat{x}_2^p - \hat{x}_1^p}{2} + \hat{x}_1^p\right)$ , $C_y^p = \left(\frac{\hat{y}_2^p - \hat{y}_1^p}{2} + \hat{y}_1^p\right)$
        $C_x^g = \left(\frac{x_2^g - x_1^g}{2} + x_1^g\right)$ , $C_y^g = \left(\frac{y_2^g - y_1^g}{2} + y_1^g\right)$
8.  Calculating center distance:
        $CD = \sqrt{\left(C_x^g - C_x^p\right)^2 + \left(C_y^g - C_y^p\right)^2}$
9.  $IoU = \frac{I}{U}$ , where $U = A^p + A^g - I.$
10. $GIoU = IoU - \frac{A^c - U}{A^c}$
11. $DGIoU = 0.9 * IoU - 0.8 * \frac{A^c - U}{A^c} - 0.1 * CD$
12. Final Bbox regression loss:
        $L_{DGIoU} = 1 - DGIoU$ , $L_{GIoU} = 1 - GIoU$ , $L_{IoU} = 1 - IoU$
13. RPN loss:
```

$$L_{DGIoU}{}' = 1 - DGIoU\ ,\ L_{GIoU}{}' = 1 - GIoU\ ,\ L_{IoU}{}' = 1 - IoU$$

4. Data Points Extraction Method

A new method is proposed to extract data points from each segmented mask patch for following hyperbolic fitting. Some post-processing steps are required, including thresholding and morphological operations. First, we define three concepts: ‘target region’, ‘isolated patch’, and ‘abnormal patch’. The ‘target region’ represents the hyperbolic target signature. The ‘isolated patch’ indicates an isolated non-target patch that is separated from the target region. The ‘abnormal patch’ represents a non-target patch that intersects with target region. Fig. 4 illustrates the concepts.



(a) (b)

Fig.4. Definitions of ‘isolated patch’, ‘abnormal patch’, and ‘target region’. (a) Example of mask patch with isolated patch and target region; (b) Example of mask patch with isolated patch, abnormal patch, and target region.

In the presence of isolated patch and abnormal patches, the proposed method searches and extracts a set of data points from the obtained mask patch. This method is an extension of the CTFP algorithm proposed in [36]. The CTFP algorithm extracts the midpoints along the hyperbolic signature region and employs a lower pass filter to smooth these points. Since CTFP cannot eliminate non-target patches, it is not robust when GPR signatures contain noises and clutter such as the isolated patches and abnormal patches. The new method overcomes this limitation and the steps for the algorithm are detailed as follows.

First, each target cluster is scanned column by column to search pixels with a value of 255 (hereafter valuable point) and store them. The midpoint of the valuable points in each column is obtained and stored in an array P. Then the peak Y can be identified from the array P and is stored. Next, taking the peak as the starting location, each column in the binary mask image is traversed to left and right based on the following rules to update and save the point set in F.

Rule 1: The searching will be stopped if there are no valuable points in the current column. An example is provided in Fig. 5. The gray cells denote valuable points. The original midpoint array $P = [L1, Y, R1, R2]$ is obtained. First, the searching traverses from peak Y to the left until the column where L1 is located (hereafter L1 column). Since there are no valuable points in the column to the left of L1 column, L1 point is retained and the searching to the left is stopped. Similarly, the searching to the right stops until the R2 column. Finally, the updated midpoint set is saved in $F = \left[L1, Y, R1, R2\right]$.

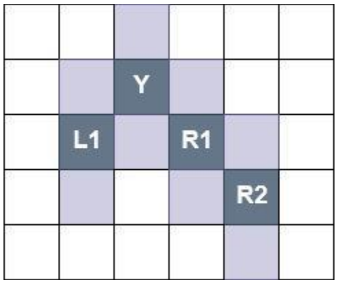


Fig. 5. An example of obtaining midpoint set $F=[L1,Y,R1,R2]$ using rule 1.

Rule 2: If searching from Y to both sides, the ordinate of midpoint in the current column is larger than that in previous column, then the F remains the same as the initial set, as shown in Fig. 6.

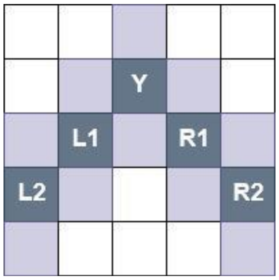


Fig. 6. An example of obtaining midpoint set $F=[L2,L1,Y,R1,R2]$ using rule 2.

Rule 3: If the ordinates of all valuable pixels in current column are smaller than the midpoint in previous adjacent column, then the search is stopped. For example, in Fig. 7, the original midpoint set is $P=[L2,L1,Y,R1,R2,R3,R4,\cdots]$, and all valuable pixels in the R4 column are above the midpoint R3. In this case, the rightward columns starting from R4 column are no longer considered (marked as blue area). Thereby, the updated set $F=[L2,L1,Y,R1,R2,R3]$ is obtained.

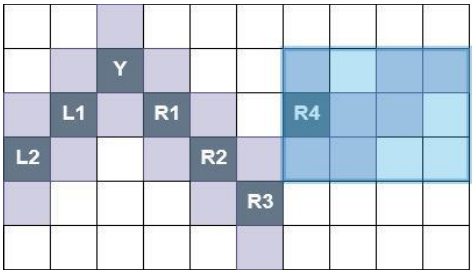


Fig. 7. An example of obtaining data point set $F=[L2,L1,Y,R1,R2,R3]$ using rule 3.

Rule 4: If the max ordinate in current column is larger than the ordinate of the midpoint in the previous column, a new midpoint is selected by adding 1 to the ordinate of the previous midpoint and is recorded into the updated set P'. Thereafter, the search continues. Fig. 8 provides an example where the initial midpoint set is $P=[L2,L1,Y,R1,R2,R3,R4,R5,\cdots]$, and valuable pixels in R4 column are divided into two parts that are above and below midpoint R3. According Rule 4, R4 will be replaced by R4', and R5 will be replaced by R5'. The following columns are no longer considered because all valuable pixels in next column are above R5', according to Rule 3. The final midpoint set $F=[L2,L1,Y,R1,R2,R3,R4',R5']$ is obtained.

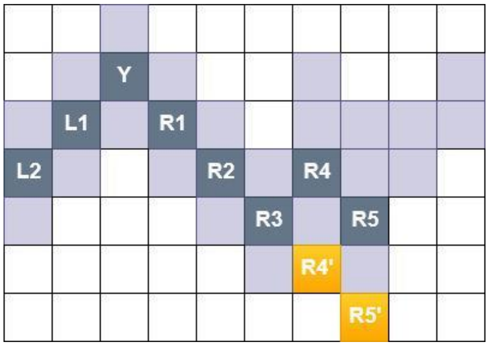


Fig. 8. An example of obtaining data point set $F = [L2, L1, Y, R1, R2, R3, R4', R5']$ using Rule 4.

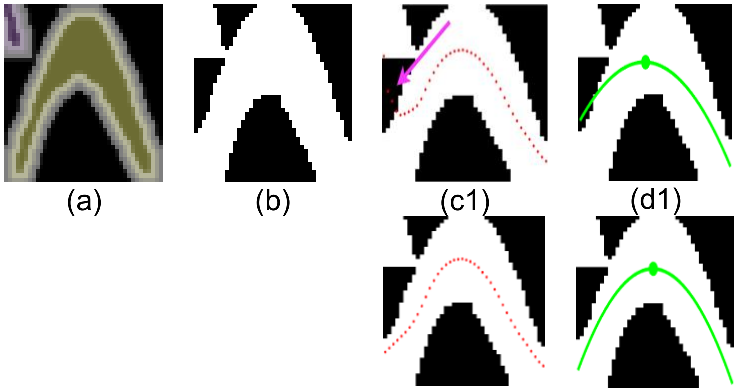
The pseudocode of the data point extraction method is presented in Alg. 3. By using the proposed method, data points are extracted from each mask patch, and outliers are separated from target region and discarded from patch. These extracted points are then fed into the subsequent hyperbola fitting step. Only the hyperbola with a downward opening will be reserved.

Algorithm 3: Data point Extraction Method

```
Input  : A binary image $I$.
Output: The extracted point set for curve fitting.
Define Int $temp = 1$. $temp$ indicates step length of searching to left/right.
  $Lflag = true$, $Rflag = true$.
  $Lflag = false$ and $Rflag = false$ indicate the searching towards to left/right is stopped.
for every column from 1 to $max\_column$ of $I$ do
    find all the $midpoint$ and store them in $P$.
    find the column $c$ where the peak $p$ of $P$ is located.
end
while (true)
   if $c - temp > 0$ then            ▹case1-traversing to the left
       find the valuable points in $(c - temp)$ and store them in $PL$.
       if $PL$ is empty then
           $Lflag = false$ end
   else $Lflag = false$ end
   if $c + temp \le max\_column\ of\ I$ ▹case1-traversing to the right
       find the valuable points in $(c + temp)$ and store them in $PR$.
       if $PR$ is empty then
           $Rflag = false$ end
   else $Rflag = false$ end
   if $Lflag = true$ and midpoint in $(c - temp)$ below the first value in updated midpoint set $F$
       ▹case2-traversing to the left
```

```
            if $max\_PL$ < $first\ value$ in $F$                ▷case3
                $Lflag = false$
            else                                                ▷case4
                store $(midpoint$ in $(c - temp)$ of $P) + 1$ in $F = [F; p]$
            end
        elseif ( $Lflag$ )
            store $midpoint$ in $(c - temp)$ of $P$ in $F = [F; p]$
        end
        if $Rflag = true$ and $midpoint$ in $(c + temp) < last\ value$ in $F$
            ▷case2-traversing to the right
            if $max\_PR > last\ value$ in $F$
                store $(midpoint$ in $(c + temp)$ of $P) + 1$ in $F = [p; F]$
            end
        elseif ( $Rflag$ )
            store $midpoint\ value$ in $(c + temp)$ of $P$ in $F = [p; F]$
        end
        $temp = temp + 1$
    end
    Output extracted data point set $F$.
```

Fig. 9 compares the proposed method and the original CTFP method, demonstrating the advantages of the proposed method. Fig. 9(a) is a typical mask region and Fig. 9(b) is the processed binary image. Fig. 9(c1) presents obtained data points by CTFP method, which contains some abnormal points (see purple arrow) due to the isolated patch at the top left. The corresponding fitted curve in Fig. 9(d1) does not match well with the hyperbolic signature. In addition, the peak of fitted curve and the apex of hyperbolic signature are not well aligned. As can be seen from Fig. 9(c2) and (d2), the proposed method handles the non-target patch and selects suitable data points, and the fitted curve accurately represents the hyperbolic signature.


(a) (b) (c1) (d1)

(c2) (d2)

Fig. 9. Performance Comparison between the proposed method and CTFP [36] method. (a) Mask patch; (b) Binary image; (c1) Data points (red dots) extracted by CTFP; (c2) Data points extracted by the proposed method; (d1-d2) Fitting result (peak marked as green dot).

5. Experiments and Results

A GSSI SIR-4000 GPR system with Model 62000 Palm antenna was used to scan a concrete bridge deck to collect B-scans (Fig. 10) to test the proposed methods. The 2GHz palm antenna has a pulse duration of 0.5 ns. The scanning distance is 455 cm and the time window is 8 ns. Each B-Scan has a size of 512×1676. Spatial sampling interval and time sampling interval are 0.27 cm and 0.016 ns, respectively. The distance between two adjacent B-scan is 50 cm and 95 B-scans are collected, covering an area of 4.55×47 m^2. These 95 B-scans contain thousands of annotated object instances. 78 B-scans were used in training, 9 B-scans were used in validation, and 8 were used in the test. We used the PyTorch 1.3.0 implementations of Mask R-CNN, released by Facebook research (https://github.com/roytseng-tw/Detectron.pytorch). The code was implemented using a NVIDIA GTX 1060 GPU. ResNet-50 was used with feature pyramid networks (FPN) as the backbone network and install it in Mask R-CNN, following its training protocol on each benchmark. To obtain the training results of Mask R-CNN using *IoU*, *GIoU*, *DGIoU* losses, the comparative tests were conducted by using L_{IoU}, L_{GIoU} and L_{DGIoU} losses in the final Bbox refinement stage and L_{IoU}', L_{GIoU}' and L_{DGIoU}' losses in RPN stage. To regularize the new regression loss, L_{IoU}, L_{GIoU} and L_{DGIoU} were multiplied by a factor of 10 for all experiments. Every anchor from feature map is compared with GT Bbox to obtain positive/negative samples, in which the threshold is set to 0.7 in the experiments.

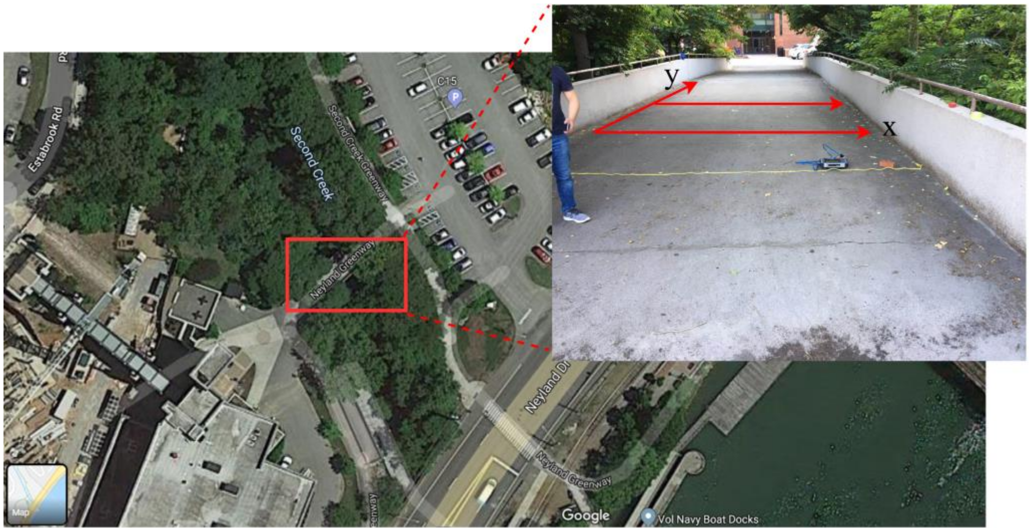
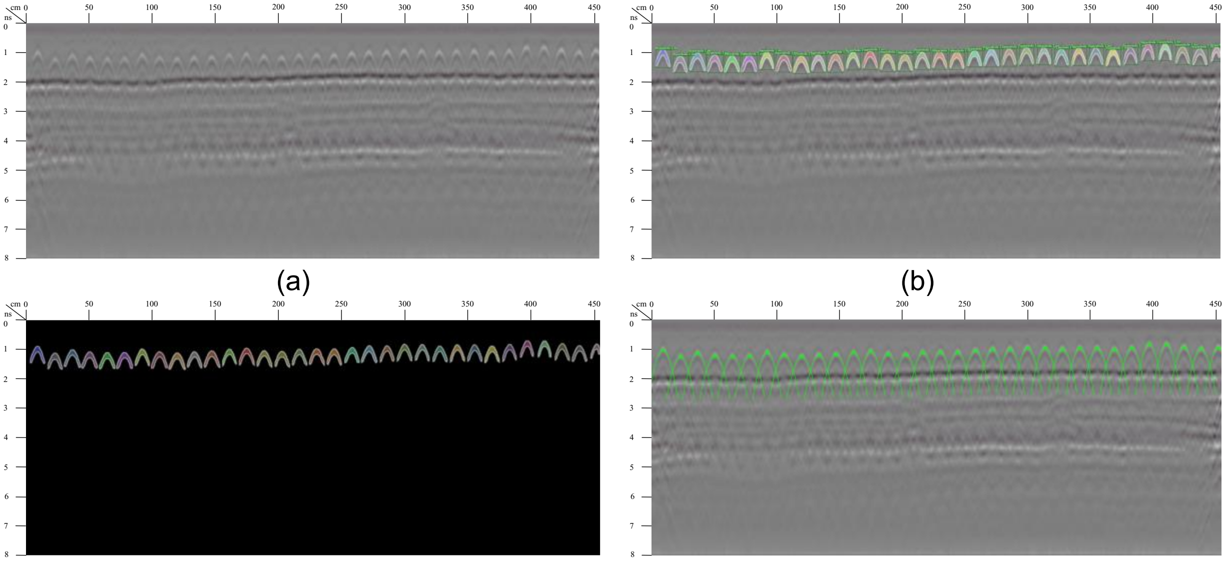


Fig. 10. Data collection from concrete bridge deck.

Three on-site trials on bridge deck datasets are conducted to verify the effectiveness of the proposed framework. The horizontal x-axis and vertical t-axis indicate GPR scanning trajectory (cm) and the two-way travel time of EM wave (ns), respectively. The results of first experiment are given in Fig. 11 and the hyperbolic signatures of 33 rebars were detected. The output of Mask R-CNN consists of 3 parts – the predicted rectangle box, confidence score, and mask patch, see Fig. 11(b). Fig. 11(c) shows the mask image on a black background with each individual instance marked with a different color. The fitting results are obtained and visualized in Fig. 11(d).

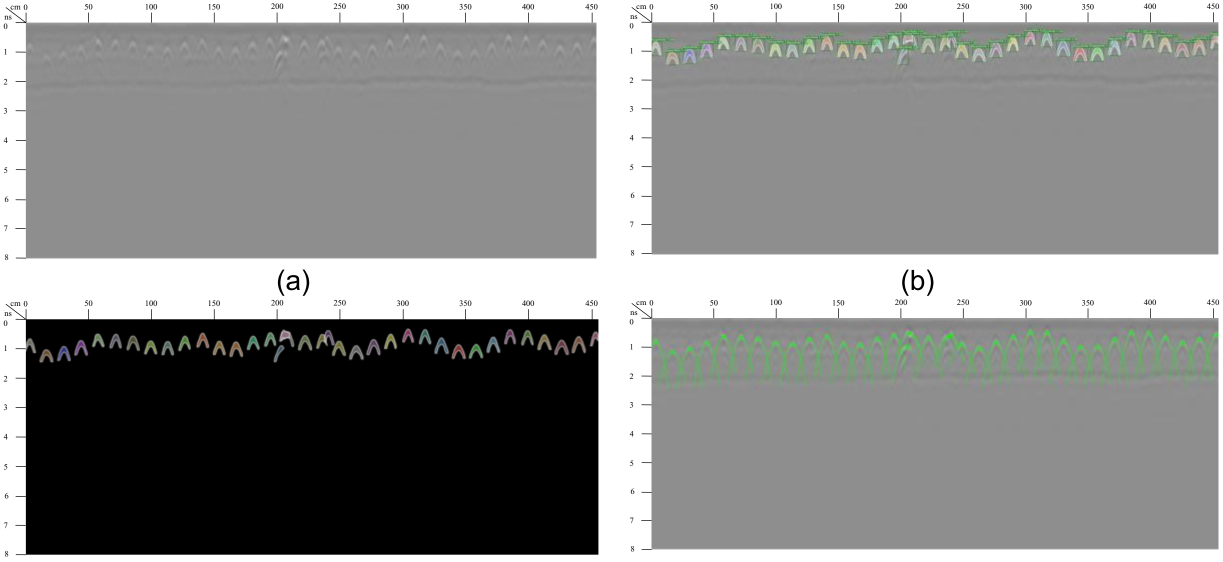


(a) (b)

(c) (d)

Fig. 11. First example of signature detection. (a) Original GPR scans; (b) Segmentation results with predicted box, confidence, and mask patch obtained by the enhanced Mask R-CNN; (c) Whole mask image; (d) Fitting results.

Fig. 12 presents the results of the second experiment. It can be found that the intersected targets can also be segmented by the proposed method, even if the targets have large overlapping area. The intermediate processing details are listed in Fig. 13. The picked patches in Fig. 13(a1-a2) are preprocessed and the corresponding binary images are yielded [Fig. 13(b1-b2)]. The proposed method extracts a series of data points (red dots) in Fig. 13(c1-c2), followed by the fitting curves for every target in Fig. 13(d1-d2).



(a) (b)

(c) (d)

Fig. 12. Second example of signature detection. (a) Original GPR bridge image; (b) Segmentation results with predicted box, confidence score, and mask patch obtained by the enhanced Mask R-CNN; (c) Whole Mask image; (d) Fitting results.

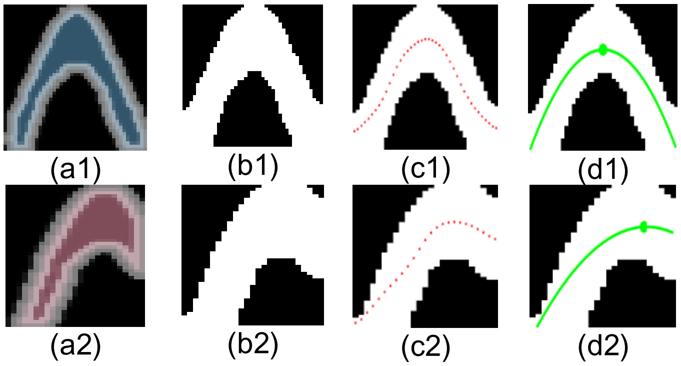


Fig. 13. Intermediate results of the second example. (a1-a2) Examples of mask region from Fig. 12(c); (b1-b2) Preprocessed binary image; (c1-c2) Extracted data points (marked as red dots); (d1-d2) Fitting results.

Fig. 14 shows the results of the third example with more complicated interference among the targets. In Fig. 14(a), there are six half-hyperbolae formed by the reflection of corner strap and are regarded as non-targets. Most of these non-targets are eliminated by Mask R-CNN but still two redundant detections left [see in Fig. 14(b)]. One of the non-target patches is eliminated, as illustrated by Fig. 15. The patch is eliminated because it does not conform to the hyperbolic downward-opening feature.

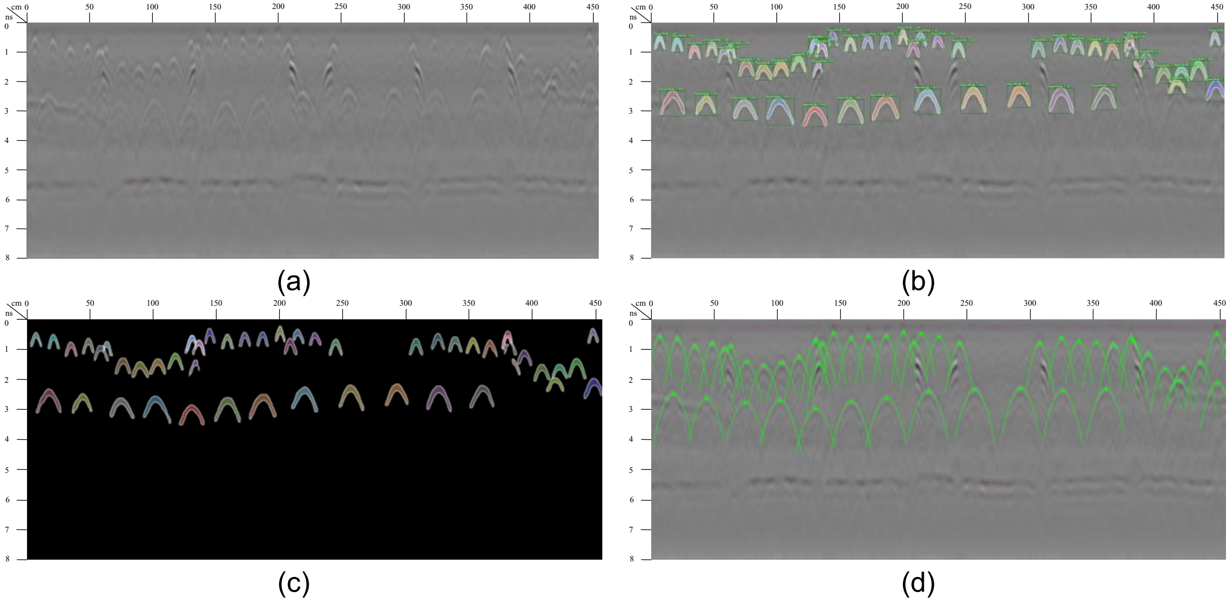


Fig. 14. Third example of signature detection. (a) Original GPR greyscale image; (b) Results with predicted box, confidence, and mask patch obtained by the improved Mask R-CNN; (c) Whole Mask image; (d) Fitting results.

(a) Mask region (b) Binary threshold patch (c) Processed patch after (d) Extracted data points (red dots) by the (e) Non-target fitting result open-close operation proposed method

Fig. 15. An example of eliminating non-target patch.

5.1 Evaluation of DGIoU loss in Mask R-CNN

The proposed *DGIoU* was evaluated both as Bbox loss and RPN loss in Mask R-CNN. RPN loss is first applied to correct the candidate box and Bbox loss is the second correction. Mask R-CNN was trained using L_{DGIoU} as Bbox regression loss and L_{DGIoU}' as RPN loss on the training dataset for 5000 iterations. Learning rate is an important parameter with the value of 0.001. *IoU* thresholds range is from 0.5 to 0.95. The best model results obtained on the validation set for Mask R-CNN have been reported in Table 2 for box task and Table 3 for mask task. By fixing L_{DGIoU} as the Bbox regression loss in final stage, Table 2 compares the detection performance based on three different RPN losses. The results indicate that the detection accuracy using L_{DGIoU}' performs better at AP and AP75 compared to the other two losses. From Table 3, it was found that the mask accuracy improved by using L_{DGIoU}' over L_{IoU}' and L_{GIoU}' losses. Particularly, comparing with L_{IoU}', the accuracy is improved to 82.91% at AP75.

Table 4 and Table 5 show the detection and segmentation results of training Mask R-CNN using three loss combinations: $L_{IoU}+L_{IoU}'$, $L_{GIoU}+L_{GIoU}'$, $L_{GDIoU}+L_{GDIoU}'$, where each group uses the same loss in Bbox refinement stage and RPN stage. According to the comparison of three groups of losses, Table 4 gives AP results, which indicates that combining the L_{DGIoU} and L_{DGIoU}' to train Mask R-CNN can consistently improves its box and mask performance compared to the other two groups. Moreover, especially in mask task [see in Table 5], compared with using *IoU* or *GIoU*, incorporating *DGIoU* both as Bbox loss and RPN loss can enhance the segmentation performance of Mask R-CNN. It improves both the detection and mask segmentation accuracy. That is because mask branches in later stages benefit from better localized Bboxes. Based the obtained best model, Fig. 16 plots the accuracy curve and four loss curves during training.

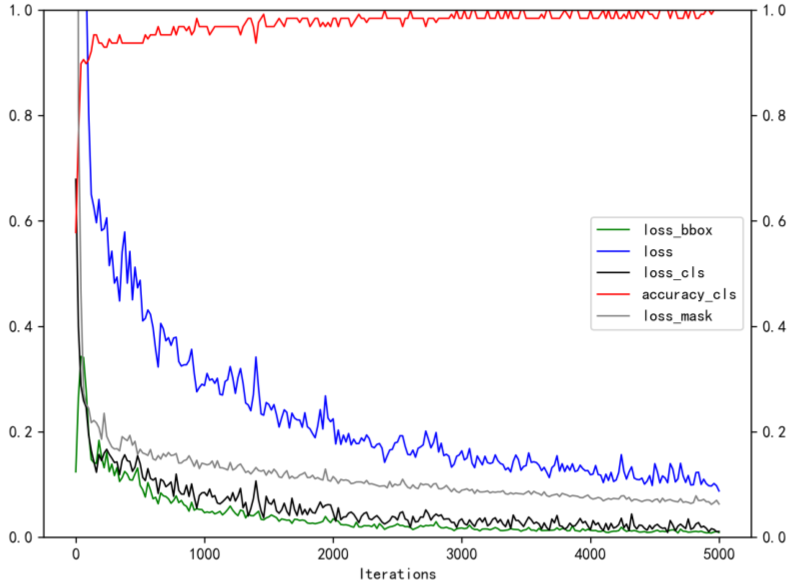


Fig. 16. Train accuracy and loss distributions of the enhanced Mask R-CNN.

TABLE 2. Detection performance comparison of training Mask R-CNN using different RPN losses: L_{IoU}' , L_{GIoU}' and L_{DGIoU}' . The results all are trained using L_{DGIoU} as Bbox loss (Task:box).

RPN loss	AP	AP75	APs	APm
L_{IoU}'	0.5749	0.5325	0.6751	0.4642
L_{GIoU}'	0.5807	0.6352	0.6711	0.4653
Relative improv.%	1.01%	19.29%	-0.59%	0.24%
L_{DGIoU}'	**0.5864**	**0.6455**	0.6651	0.4630
Relative improv.%	**2.00%**	**21.22%**	-1.48%	-0.26%

TABLE 3. Segmentation performance comparison of training Mask R-CNN using different RPN losses: L_{IoU}' , L_{GIoU}' and L_{DGIoU}' . The results all are trained using L_{DGIoU} as Bbox loss (Task: mask).

RPN loss	AP	AP75	APs	APm
L_{IoU}'	0.4280	0.1592	0.4321	0.3663
L_{GIoU}'	0.4282	0.1782	0.4345	0.3337
Relative improv.%	0.05%	11.93%	0.56%	-8.90%
L_{DGIoU}'	**0.4764**	**0.2912**	**0.4834**	**0.3673**
Relative improv.%	**11.3%**	**82.91%**	**11.87%**	**0.27%**

TABLE 4. Detection performance comparison of training Mask R-CNN using $L_{IoU} + L_{IoU}'$, $L_{GIoU} + L_{GIoU}'$ and $L_{DGIoU} + L_{DGIoU}'$ losses. (Task: box)

Loss and RPN loss	AP	AP75	APs	APm
$L_{IoU} + L_{IoU}'$	0.5761	0.6335	0.6650	0.4770
$L_{GIoU} + L_{GIoU}'$	0.5675	0.5723	0.6771	0.3777
Relative improv.%	-0.86%	-9.66%	1.82%	-20.82%
$L_{DGIoU} + L_{DGIoU}'$	**0.5864**	**0.6455**	0.6651	0.4630
Relative improv.%	**1.79%**	**1.89%**	0.02%	-2.94%

TABLE 5. Segmentation performance comparison of training Mask R-CNN using $L_{IoU} + L_{IoU}'$, $L_{GIoU} + L_{GIoU}'$ and $L_{DGIoU} + L_{DGIoU}'$ losses. (Task: mask)

Loss/Metric	AP	AP75	APs	APm
$L_{IoU} + L_{IoU}'$	0.3724	0.0829	0.3821	0.2337
$L_{GIoU} + L_{GIoU}'$	0.4259	0.1559	0.4304	0.3663
Relative improv.%	14.37%	88.06%	12.64%	56.74%
$L_{GDIoU} + L_{GDIoU}'$	**0.4764**	**0.2912**	**0.4834**	**0.3673**
Relative improv.%	**27.93%**	**251.27%**	**26.51%**	**57.17%**

5.2 Evaluation of data points extraction method

Fig. 17 explains the eliminating process of the non-target patch when picked mask region has more than one patch. The region [see in Fig. 17(a1-a3)] picked from whole mask image still contains two patches after preprocessing [see in Fig. 17(b1-b3)]. Our method traverses each patch separately and extract its data points using the proposed data points extraction method, as shown in Fig. 17(c1-c3) and (d1-d3). The fitting results for every object patch are shown in Fig. 17(e1-e3). Because the signature of rebar presents as a downward-opening hyperbola, therefore the patch with an upward-opening fitting curve is discarded. Even without the clustering method as used in [36,56,57], the proposed data extraction method still has the ability to handle unwanted interference and is robust in the presence of multi-patches.

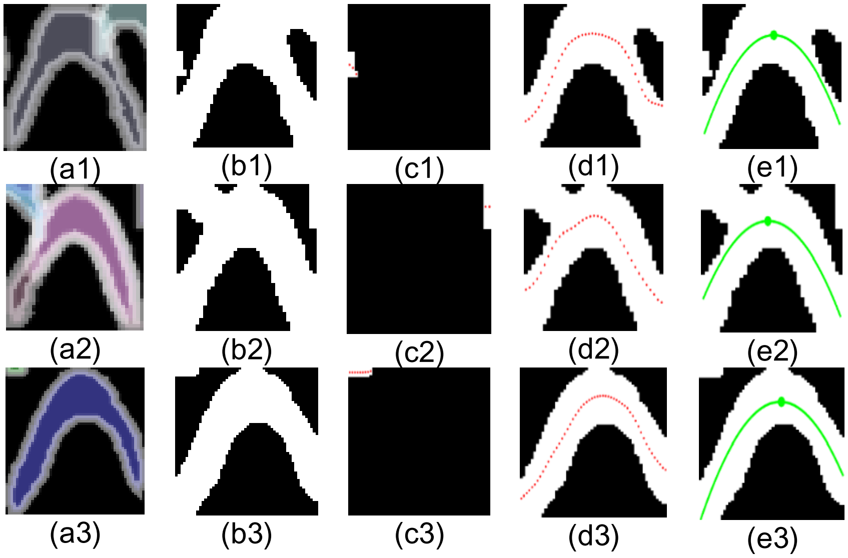


Fig.17. Some fitting results based on data points extraction and elimination of non-target patch. (a1-a3) Mask region with multiple patches; (b1-b3) Post-processed binary image; (c1-c3) Obtained data points from isolated patch (red points); (d1-d3) Obtained data points from target region with/without abnormal patch; (e3) Fitting result with downward opening and discarding the non-target patch.

5.3 Comparison experiments

Performance Comparison. The box task between the Mask R-CNN and the Faster R-CNN are compared. For baseline results, Faster R-CNN also uses the same backbone ResNet-50 and RPN as Mask R-CNN. They are evaluated on the same field dataset and training protocol.

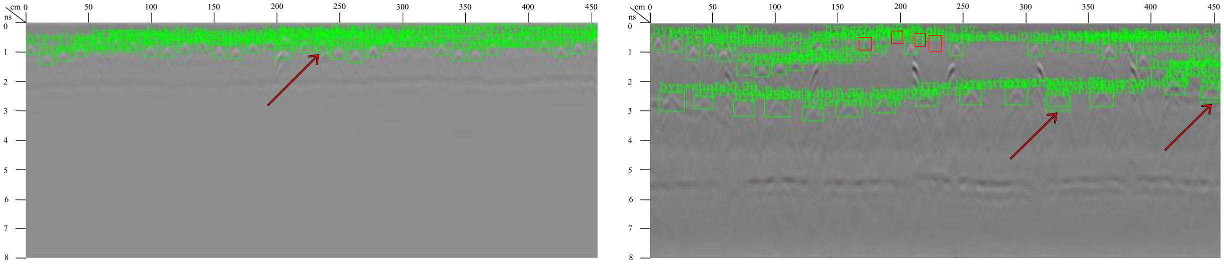
Table 6 represents the results of training Faster R-CNN using *IoU*, *GIoU,* and *DGIoU* losses in the final Bbox refinement and RPN stage, respectively. Using *DGIoU* as both Bbox regression loss and RPN loss, Faster R-CNN outperformed the other two loss combinations and achieved the AP of 55.08% and AP75 of 59.57%. Comparing with the evaluation results in Table 4, both frameworks improved the detection accuracy when using the proposed *DGIoU* as losses. Moreover, comparing with the Faster R-CNN with DGIoU, the enhanced Mask R-CNN has a higher AP of 3.56% and AP75 of 4.98%.

TABLE 6. Detection performance of training Faster R-CNN using $L_{IoU}+L_{IoU}'$, $L_{GIoU}+L_{GIoU}'$ and L_{DGIoU} + L_{DGIoU}' losses. (Task: box)

Loss and RPN loss	AP	AP75	APs	APm
$L_{IoU}+L_{IoU}'$	0.5393	0.5882	0.5782	0.4611

$L_{GIoU} + L_{GIoU}'$	0.5467	0.5413	0.5695	0.3927
Relative improv.%	1.37%	-7.97%	-1.50%	-14.8%
$L_{DGIoU} + L_{DGIoU}'$	**0.5508**	**0.5957**	0.5760	0.4588
Relative improv.%	**2.13%**	**1.28%**	-0.38%	-0.50%

Detection Result Comparison. To further visualize the comparison of the above two frameworks, Fig. 18 respectively shows the detection results using the Faster R-CNN framework with *DGIoU* on two original GPR images (Fig. 12(a) and Fig. 14(a)). Compared with the results in Fig. 12(b), Fig. 18(a) also detected all targets, whereas there exists a redundant box indicated by red arrow. Compared with Fig. 14(b), Fig. 18(b) omits four hyperbolic targets (marked by red rectangle), and there are two redundant boxes (shown by red arrows). The comparative results suggest the performance of the enhanced Faster R-CNN is inferior to the improved Mask R-CNN, and its performance is limited in the presence of hyperbolic signatures with weak reflection.



(a) (b)

Fig. 18. Detection results obtained by the Faster R-CNN with *DGIoU* on (a) original image Fig. 12(a), and (b) original image Fig. 14(a). (Red arrow indicates redundant box and red rectangle indicates missing target)

6. Conclusion

A DL based method is proposed to automate the detection and segmentation of object signatures from GPR scans. Mask R-CNN is adopted as the framework. To improve its performance, *DGIoU* is developed and incorporated in Mask R-CNN as a new loss computation to minimize the discrepancy between the predicted Bbox and the real Bbox in the training phase. Compared to the conventional *IoU* and *GIoU*, the *DGIoU* considers the center distance between two bounding boxes. The experiment results demonstrated the effectiveness of using *DGIoU* in enhancing the performance of Mask R-CNN. The segmented patches containing object signatures and background noises are then transformed into binary images for data points extraction. Our new data extraction method removes the interference among adjacent object signatures and the background. The extracted data points can be used in curve fitting to estimate object location and dimension. Real GPR scans collected from a concrete bridge deck were used in the experiments. The improved Mask R-CNN achieved a detection AP of 58.64% and a segmentation AP of 47.64%.

There are some limitations that could be addressed in future research. First, it requires vast amounts of real data for training the proposed method. Also, the publicly available GPR datasets are limited. Efforts need to be made in the future to collect and prepare GPR datasets to facilitate the implementation of the method. Unsupervised DL models and transfer learning techniques can also be leveraged to solve the problem of insufficient dataset. In addition, future studies can be conducted to explore the one-shot or few-shot detection and segmentation method. Second, this study mainly focused on detecting and segmenting rebar signatures from GPR scans. In future research, discriminative features can be learned from data to detect and evaluate potential rebar corrosions, which has not been achieved. Third, the proposed method has only been tested for rebar detection and localization. The scalability of the method in other challenging environments and applications needs further investigation. Future studies could test this method for detecting buried utilities in heterogeneous soils.

Acknowledgement

This work was funded by the National Natural Science Foundation of China (nos. 61102139), the U.S. National Science Foundation (nos. 1850008), and the China Scholarship Council (CSC).

Reference

1. S. Jazayeri, S. Kruse, I. Hasan, and N. Yazdani. Reinforced concrete mapping using full-waveform inversion of GPR data. Construction and Building Materials, 229 (2019), pp.117102, http://dx.doi.org/10.1016/j.conbuildmat.2019.117102.
2. A. Simi, G. Manacorda, M. Miniati, S. Bracciali, A. Buonaccorsi. Underground asset mapping with dual-frequency dual-polarized GPR massive array. In Proceedings of the XIII Internarional Conference on Ground Penetrating Radar, IEEE, June, 2010, pp. 1-5, http://dx.doi.org/10.1109/ICGPR.2010.5550236.
3. C. Ékes, B. Neducza, G.R. Henrich. GPR goes underground: Pipe penetrating radar. Proceedings of the North American Society for Trenchless Technology (NASTT) NoDig Show, March, 2011, pp. 27-31, https://www.sewervue.com/papers/B-3-02_Final_Paper_GPR_Goes_Underground-Pipe_Penetrating.pdf.
4. A. Sărăcin. Using georadar systems for mapping underground utility networks. Procedia Engineering, 209 (2017) pp.216-223, http://dx.doi.org/10.1016/j.proeng.2017.11.150.
5. N. Gucunski, A. Imani, F. Romero, S. Nazarian, D. Yuan, H. Wiggenhauser, P. Shokouhi, A. Taffe, D. Kutrubes. Nondestructive Testing to Identify Concrete Bridge Deck Deterioration. Transportation Research Board, Washington D.C., 2013, http://dx.doi.org/10.17226/22771. (ISBN: 780309129336, 0309129338)
6. K. Dinh, N. Gucunski, T. Duong. An algorithm for automatic localization and detection of rebars from GPR data of concrete bridge decks. Automation in Construction, 89 (2018) pp.292-298, https://doi.org/10.1016/j.autcon.2018.02.017.
7. N. Martino, K. Maser, R. Birken, M. Wang. Quantifying bridge deck corrosion using ground penetrating radar. Research in Nondestructive Evaluation, 27 (2) (2016) pp.112–124, http://dx.doi.org/10.1080/09349847.2015.1067342.
8. P. Kaur, K.J. Dana, F.A. Romero, N. Gucunski. Automated GPR rebar analysis for robotic bridge deck evaluation. IEEE Transactions on Cybernetics, 46 (10) (2015) pp.2265-2276, https://doi.org/10.1109/TCYB.2015.2474747.
9. N. Gucunski, B. Basily, J. Kim, G.Y. Jin, T. Duong, K. Dinh, S.H. Kee, A. Maher. RABIT: implementation, performance validation and integration with other robotic platforms for improved management of bridge decks. International Journal of Intelligent Robotics and Applications, 1 (3) (2017) pp.271–286, http://dx.doi.org/10.1007/s41315-017-0027-5.
10. N. Gucunski, B. Pailes, J. Kim, H. Azari, K. Dinh. Capture and quantification of deterioration progression in concrete bridge decks through periodical NDE surveys. Journal of Infrastructure Systems, 23 (1) (2016) pp.B4016005, http://dx.doi.org/10.1061/(ASCE)IS.1943-555X.0000321.
11. F. Gutiérrez, J.P. Galve, P. Lucha, C. Castañeda, J. Bonachea, J. Guerrero. Integrating geomorphological mapping, trenching, InSAR and GPR for the identification and characterization of sinkholes: A review and application in the mantled evaporite karst of the Ebro Valley (NE Spain). Geomorphology, 134 (1-2) (2011) pp.144-156, http://dx.doi.org/10.1016/j.geomorph.2011.01.018.
12. W. Y. Choi, K.K. Park. Array type miniaturized ultrasonic sensors to detect urban sinkholes. Measurement, 141 (2019) 371-379, http://dx.doi.org/10.1016/j.measurement.2019.04.043.
13. I.L. Al-Qadi, S. Lahouar. Measuring layer thicknesses with GPR–Theory to practice. Construction and Building Materials, 19 (10) (2005) pp.763-772, http://dx.doi.org/10.1016/j.conbuildmat.2005.06.005.
14. D.J. Daniels. Ground penetrating radar. Encyclopedia of RF and Microwave Engineering, 2005, John Willy & Sons, http://dx.doi.org/10.1002/0471654507.eme152.
15. N.R. Syambas. An Approach for Predicting the Shape and Size of a Buried Basic Object on Surface Ground Penetrating Radar System. International Journal of Antennas and Propagation, 2012 (2012), http://dx.doi.org/10.1155/2012/919741.
16. M.R. Shaw, S.G. Millard, T.C.K. Molyneaux, M.J. Taylor, J.H. Bungey. Location of steel

reinforcement in concrete using ground penetrating radar and neural networks. NDT & E International, 38 (3) (2005) pp. 203–212, http://dx.doi.org/10.1016/j.ndteint.2004.06.011.

17. D.L. Chen, C.L. Huang, Y.A. Su. An integrated method of statistical method and Hough transform for GPR targets detection and location. Acta Electronica Sinica, 32 (9) (2004) pp.1468-1471. http://en.cnki.com.cn/Article_en/CJFDTotal-DZXU200409014.htm.
18. C.G. Windsor, L. Capineri, P. Falorni. A data pair-labeled generalized Hough transform for radar location of buried objects. IEEE Geoscience & Remote Sensing Letters, 11 (1) (2013) pp.124-127, https://doi.org/10.1109/LGRS.2013.2248119.
19. C. Maas, J. Schmalzl. Using pattern recognition to automatically localize reflection hyperbolas in data from ground penetrating radar. Computers & Geosciences, 58 (2) (2013) pp. 116-125, https://doi.org/10.1016/j.cageo.2013.04.012.
20. A.C. Gurbuz, J.H. McClellan, W.R. Scott, Compressive sensing for GPR imaging. In 2007 Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers, IEEE, November, 2007, pp.2223-2227, http://dx.doi.org/10.1109/ACSSC.2007.4487636.
21. K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, 2017, pp.2961-2969, http://dx.doi.org/10.1109/ICCV.2017.322.
22. D. Zhou, J. Fang, X. Song, C. Guan, J. Yin, Y. Dai, R. Yang. IoU Loss for 2D/3D Object Detection. In 2019 International Conference on 3D Vision, IEEE, September, 2019, pp.85-94, http://dx.doi.org/10.1109/3DV.2019.00019.
23. H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, S. Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp.658-666, http://dx.doi.org/10.1109/CVPR.2019.00075.
24. Y. Wang, G. Cui, J. Xu. Semi-automatic detection of buried rebar in GPR data using a genetic algorithm. Automation in Construction, 114 (2020) pp.103186, http://dx.doi.org/10.1016/j.autcon.2020.103186.
25. H. Harkat, A.E. Ruano, M.G. Ruano, S.D. Bennani. GPR target detection using a neural network classifier designed by a multi-objective genetic algorithm. Applied Soft Computing, 79 (2019) pp. 310-325, http://dx.doi.org/10.1016/j.asoc.2019.03.030.
26. J. Zhang, C. Zhang, Y. Lu, T. Zheng, Z. Dong, Y. Tian, Y. Jia. In-situ recognition of moisture damage in bridge deck asphalt pavement with time-frequency features of GPR signal. Construction and Building Materials, 244 (2020) pp.118295, http://dx.doi.org/10.1016/j.conbuildmat.2020.118295.
27. L. Longoni, D. Arosio, M. Scaioni, M. Papini, L. Zanzi, R. Roncella, D. Brambilla. Surface and subsurface non-invasive investigations to improve the characterization of a fractured rock mass. Journal of Geophysics and Engineering, 9 (5) (2012) pp.461-472, http://dx.doi.org/10.1088/1742-2132/9/5/461.
28. F. Tosti, E. Slob. Determination, by using GPR, of the volumetric water content in structures, substructures, foundations and soil. In Civil Engineering Applications of Ground Penetrating Radar. Springer, Cham, 2015, pp. 163-194, http://dx.doi.org/10.1007/978-3-319-04813-0_7.
29. L. Crocco, V. Ferrara. A review on ground penetrating radar technology for the detection of buried or trapped victims. In 2014 International Conference on Collaboration Technologies and Systems (CTS). IEEE, May, 2014, pp.535-540, http://dx.doi.org/10.1109/CTS.2014.6867620.
30. A. Benedetto, F. Tosti, L.B. Ciampoli, F. D’amico. An overview of ground-penetrating radar signal processing techniques for road inspections. Signal Processing, 132 (2017) pp.201-209, http://dx.doi.org/10.1016/j.sigpro.2016.05.016.
31. W. W. L. Lai, X. Derobert, P. Annan. 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 (2018) pp.58-78, http://dx.doi.org/10.1016/j.ndteint.2017.04.002.

32. W. W. L. Lai, T. Kind, H. Wiggenhauser. Using ground penetrating radar and time–frequency analysis to characterize construction materials. NDT & E International, 44 (1) (2011) pp.111-120, http://dx.doi.org/10.1016/j.ndteint.2010.10.002.
33. B. Yoldemir, M. Sezgin. Peak scatter-based buried object identification using GPR-EMI dual sensor system. Nondestructive Testing and Evaluation, 34 (4) (2019) pp.339-353, http://dx.doi.org/10.1080/10589759.2019.1623213.
34. X. Wu, C.A. Senalik, J. Wacker, X. Wang, G. Li. Object detection of ground-penetrating radar signals using empirical mode decomposition and dynamic time warping. Forests, 11 (2) (2020) pp.230, http://dx.doi.org/10.3390/f11020230.
35. M. Kim, S.D. Kim, J. Hahm, D. Kim, S.H. Choi. GPR Image Enhancement Based on Frequency Shifting and Histogram Dissimilarity. IEEE Geoscience and Remote Sensing Letters, 15 (5) (2018) pp. 684-688, http://dx.doi.org/10.1109/LGRS.2018.2809720.
36. W. Lei, F. Hou, J. Xi, Q. Tan, M. Xu, X. Jiang, G. Liu, Q. Gu. Automatic hyperbola detection and fitting in GPR B-scan image. Automation in Construction, 106 (2019) pp.102839, http://dx.doi.org/10.1016/j.autcon.2019.102839.
37. M.S. Kang, N. Kim, J.J. Lee, Y.K. An. Deep learning-based automated underground cavity detection using three-dimensional ground penetrating radar. Structural Health Monitoring, 19 (1) (2020) pp.173-185, http://dx.doi.org/10.1177/1475921719838081.
38. Z. W. Wang, M. Zhou, G.G. Slabaugh, J. Zhai, T. Fang. Automatic detection of bridge deck condition from ground penetrating radar images. IEEE Transactions on Automation Science and Engineering, 8 (3) (2011) pp.633–640, http://dx.doi.org/10.1109/TASE.2010.2092428.
39. P. Viola, M. J. Jones. Robust real-time face detection. International Journal of Computer Vision, 57 (2) (2004) 137-154, https://doi.org/10.1023/b:visi.0000013087.49260.fb.
40. P. Kaur, K. J. Dana, F.A. Romero, N. Gucunski. Automated GPR rebar analysis for robotic bridge deck evaluation. IEEE Transactions on Cybernetics, 46 (10) (2016) pp.2265–2276, http://dx.doi.org/10.1109/TCYB.2015.2474747.
41. A. Krizhevsky, I. Sutskever, G.E. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems, 2012, pp.1097–1105, http://dx.doi.org/10.1145/3065386.
42. P. Tang, X. Wang, Z. Huang, X. Bai, W. Liu. Deep patch learning for weakly supervised object classification and discovery. Pattern Recognition, 71 (2017) pp.446–459, http://dx.doi.org/10.1016/j.patcog.2017.05.001.
43. R. Girshick, J. Donahue, T. Darrell, J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp.580–587, http://dx.doi.org/10.1109/CVPR.2014.81.
44. S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems, 2015, pp. 91–99, http://dx.doi.org/10.1109/TPAMI.2016.2577031.
45. L. Huang, Y. Yang, Y. Deng, Y. Yu. Densebox: Unifying landmark localization with end to end object detection. 2015, https://arxiv.org/pdf/1509.04874.pdf.
46. J. Redmon, S. Divvala, R. Girshick, A. Farhadi. You only look once: Unified, real-time object detection. In IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp.779–788, http://dx.doi.org/10.1109/CVPR.2016.91.
47. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg. Ssd: Single shot multibox detector. In European Conference on Computer Vision, 2016, 21–37, http://dx.doi.org/10.1007/978-3-319-46448-0_2.
48. J. Long, E. Shelhamer, T. Darrell. Fully convolutional networks for semantic segmentation. In IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp.3431–3440, http://dx.doi.org/10.1109/CVPR.2015.7298965.
49. L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille. Deeplab: Semantic image

segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on Pattern Analysis and Machine Intelligence, 2018, pp.834–848, http://dx.doi.org/10.1109/TPAMI.2017.2699184.

50. H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia. Pyramid scene parsing network. In IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp.2881–2890, http://dx.doi.org/10.1109/CVPR.2017.660.
51. Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, W. Liu. Ccnet: Criss-cross attention for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, 2019, pp.603-612, http://dx.doi.org/10.1109/ICCV.2019.00069.
52. L.E. Besaw, P.J. Stimac. Deep convolutional neural networks for classifying GPR B-scans. Proceedings of SPIE - The International Society for Optical Engineering, (2015) pp.9454, https://doi.org/10.1117/12.2176250.
53. J. Bralich, D. Reichman, L.M. Collins. Improving convolutional neural networks for buried threat in ground penetrating radar using transfer learning via pretraining. In Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXII, International Society for Optics and Photonics, 10182 (2017) pp.101820X, https://doi.org/10.1117/12.2263112.
54. D. Reichman, L.M. Collins, J.M. Malof. Some good practices for applying convolutional neural networks to buried threat detection in Ground Penetrating Radar. In 2017 9th International Workshop on Advanced Ground Penetrating Radar, IEEE, 2017, pp.1-5, https://doi.org/10.1109/IWAGPR.2017.7996100.
55. M.T. Pham, S. Lefèvre. Buried object detection from B-scan ground penetrating radar data using Faster-RCNN. 38th IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2018, pp.6804-6807, https://arxiv.org/pdf/1803.08414.pdf.
56. X. Zhou, H. Chen, J. Li. An automatic GPR B-Scan image interpreting model. IEEE Transactions on Geoscience and Remote Sensing, 56 (6) (2018) pp.3398-3412, https://doi.org/10.1109/TGRS.2018.2799586.
57. Q. Dou, L. Wei, D. R. Magee, A. Cohn. Real-time hyperbola recognition and fitting in GPR data. IEEE Transactions on Geoscience and Remote Sensing, 55 (1) (2016) pp.51-62, https://doi.org/10.1109/tgrs.2016.2592679.
58. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp.3213-3223, http://dx.doi.org/10.1109/CVPR.2016.350.
59. H. Alhaija, S. Mustikovela, L. Mescheder, A. Geiger, C. Rother. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. International Journal of Computer Vision, 126 (9) (2018) pp.961-972, http://dx.doi.org/10.1007/s11263-018-1070-x.
60. B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba. Scene parsing through ade20k dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp.633-641, http://dx.doi.org/10.1109/CVPR.2017.544.
61. T.Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C.L. Zitnick, P. Dollár. Microsoft COCO: common objects in context. In European conference on computer vision, Springer, Cham, September, 2014, pp.740-755, http://dx.doi.org/10.1007/978-3-319-10602-1_48.
62. M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman. The pascal visual object classes (voc) challenge. International Journal of Computer Vision, 88 (2) (2010) pp.303–338, http://dx.doi.org/10.1007/s11263-009-0275-4.
63. J. Yu, Y. Jiang, Z. Wang, Z. Cao, T. Huang. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, ACM, October, 2016, pp. 516–520, http://dx.doi.org/10.1145/2964284.2967274.
64. L. Tychsen-Smith and L. Petersson. Improving object localization with fitness nms and bounded iou loss. In Proceedings of the IEEE Conference on Computer Vision and Pattern

Recognition, 2018, pp.6877–6885, http://dx.doi.org/10.1109/CVPR.2018.00719.