Automatic Road Extraction From Remote Sensing Imagery Using Ensemble Learning and Postprocessing

Junjie Li, Yizhuo Meng, Donyu Dorjee, Xiaobing Wei, Zhiyuan Zhang, and Wen Zhang

***Abstract*—High-resolution satellite images contain valuable road semantic information, but the occlusion of vegetation and buildings and the sparse distribution and heterogeneous appearance of roads limit the accuracy of road extraction models. In this article, we propose a novel method for extracting roads using an ensemble learning model with a postprocessing stage. The network weights and biases of our proposed deep learning model are transmitted through the random combination of layers of different submodels during forward and backward propagation. In the gradient descent process, a superior loss function is designed to solve the problem of class imbalance caused by road sparseness, and more attention is given to hard classification samples to extract narrow and covered roads. In addition, we solve road disconnection issues in the results obtained with the neural network by extracting and analyzing the geometric structures and feature points of the roads. Experiments on two challenging datasets of remote sensing imagery show that the proposed method performs better than other models and can extract road information from complex scenes.**

***Index Terms*—Convolutional neural network (CNN), ensemble learning, remote sensing, road extraction, semantic segmentation.**

I. Introduction

AS AN essential part of basic geographical data at the national scale, roads play an important role in urban planning, transportation logistics, disaster assistance, emergency relief, navigation, etc. Automatically extracting and updating road information has always been a popular research topic. At present, methods for extracting and updating road information mainly involve traditional surveying and mapping, for which road information is generated from manual field measurements and recorded, or GPS trajectories. Road information is collected through professional GPS track acquisition devices for vehicles, taxis and individuals [1], [2]. Traditional methods require considerable manpower and material resources. Additionally, the extensive time requirement for early data collection leads to low-efficiency road extraction and updating. Thus, most methods are not suitable for the timely updating of road information over a large range. The temporal availability and wide coverage of remote sensing images support the large-scale extraction of object information. With the development of remote sensing technology, very high-resolution satellite images and aerial images have become important data sources for road extraction. However, there are various challenges associated with these sources. First, roads are often masked by the shadows of buildings and vegetation. Second, the colors, widths and shapes of roads in different regions vary greatly. For example, urban arterial roads are straighter and wider than rural roads. Third, relative to the distributions of vegetation, water and other objects in an image, the road distribution is generally sparser. All these factors increase the difficulty of automatically extracting roads from remote sensing images.

To solve these problems, many methods have been proposed to extract roads from complex backgrounds. Early methods can be divided into snake models [3], [4], dynamic programming methods [5], [6] and template matching methods [7], [8]. These methods extract the geometric and texture features of roads at pixel or object scale. However, road interference and connectivity problems are often encountered with these methods. In recent years, with the broad application of deep learning technology in computer vision, natural language processing and multimedia, convolutional neural networks (CNNs) have been verified as effective in mining contextual information from images. Some representative CNNs and algorithms include FCNs [9], UNet [10], PSPNet [11], and DeepLab [12], [13].

Based on the above networks, many road extraction neural networks have been proposed [14]–[19]. These algorithms treat road extraction as a semantic segmentation problem. Although these networks have achieved satisfactory results, there are still obvious limitations, such as the lack of sufficient reasoning ability. Feature extraction based on CNNs mainly relies on visual information, but the capture angle of an image, imaging time, and distribution of surface objects can influence the original visual information. Therefore, if a road is blocked by vegetation, buildings or other nonroad objects, obvious road distortion may occur, thus reducing the accuracy of extraction. Sparse distribution

Manuscript received March 28, 2021; revised May 17, 2021, June 9, 2021, and June 28, 2021; accepted July 1, 2021. Date of publication July 7, 2021; date of current version October 27, 2021. This work was supported by the Second Tibetan Plateau Scientific Expedition Program under Grant 2019QZKK0304 and the National Key Research and Development Program of China under Grant 2017YFC0405806. *(Corresponding author: Wen Zhang.)*

Junjie Li, Yizhuo Meng, Xiaobing Wei, and Wen Zhang are with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: junjieli1996@foxmail.com; yangguangmeng05@whu.edu.cn; 935880282@qq.com; wen_zhang@whu.edu.cn).

Donyu Dorjee is with the Climate Center of Tibet Autonomous Region, Lhasa 850000, China (e-mail: 398784231@qq.com).

Zhiyuan Zhang is with the Information Center (Hydrology Monitor and Forecast Center), Ministry of Water Resources, Beijing 100053, China (e-mail: zhangzhiyuan@mwr.gov.cn).

Digital Object Identifier 10.1109/JSTARS.2021.3094673

and class imbalance issues also influence these methods. Road extraction is essentially a problem of semantic segmentation, and road information is segmented from the complex background of remote sensing images. In this process, only two categories need to be considered: roads and nonroads. However, the road and nonroad pixels in a sample are often unbalanced due to the diversity of road widths and the sparse distribution of roads. Ordinary CNNs cannot solve this problem because they give the same amount of attention to each pixel.

To overcome such limitations, inspired by [20], we propose a road extraction framework based on ensemble learning that considers contextual information and road connectivity. A large number of studies and online competitions have proved that ensemble learning methods can effectively improve the accuracy and robustness of the model [17], [21]. Unlike ordinary CNNs, the network weights and biases of the proposed deep learning model are transmitted through the random combination of layers of different submodels during forward and backward propagation. By fusing the features of different layers of different submodels, low-level location information and high-level semantic information can be effectively extracted, thus enhancing the reasoning ability and robustness of the model. A new loss function is designed for road sparsity and diversity. Compared with the traditional cross-entropy (CE) loss function, this new function can effectively solve the problem of class imbalance and consider pixels that are difficult to classify. Moreover, due to the deceptiveness of visual information in an image, we design a postprocessing method based on geometric structure analysis and feature point extraction to help solve the problem of road connectivity. In general, our framework achieves performance improvements by ensemble learning and connecting broken roads. The main contributions of the article are summarized as follows.

1) A new ensemble learning model for road extraction is proposed, which improves the robustness and predictive performance through random combination and propagation of neural network layers.
2) A synthetic loss is designed to simultaneously focuses on class imbalance at the image level and the hard classification at the pixel level.
3) An improved multistage postprocessing algorithm is introduced to connect road breakpoints. A linear region growing algorithm is proposed to speed up the connection, and the width of the breakage can be automatically obtained by calculating the connected domain of road segments.

The remainder of this article is structured as follows. Section II explains the existing work related to road extraction. The details of the proposed method are presented in Section III. Section IV introduces the datasets and preprocessing method. Section V describes the experimental results and the corresponding analyses. Finally, in Section VI, conclusions are drawn, and future research is recommended.

II. Related Work

A. Feature-Based Approaches for Road Extraction

In the early stage of road extraction research, roads were mainly identified based on spectral, color, shape, edge, topological, and direction features. The process of road extraction involves feature extraction, feature fusion, and classification steps. Tupin *et al*. [22] extracted linear features from speckled radar images and then defined a Markov random field to identify real roads. Gaetano *et al*. [23] used an ad hoc skeletonization procedure to describe the linear structure of road segments. Liu *et al*. [24] constructed a geometric knowledge base for rural roads based on the linear characteristics of roads. Chaudhuri *et al*. [25] used a semiautomatic approach that accurately derives road segments by developing customized operators. These methods based on feature extraction are often semiautomatic methods that rely on manual selection, and although the required operations are simple, the experimental effect depends largely on the quality of feature selection and the feature fusion algorithm.

B. Object-Based Approaches for Road Extraction

Object-based image analysis (OBIA) uses the object region as the basic analysis unit. Compared with pixel-scale methods, OBIA can deeply extract shape, texture and other information for ground objects in an image to improve the smoothness of the road extraction results [26]. The image is first segmented to obtain many irregular homogeneous objects, and the pixels located in these objects have similar spectral and texture characteristics. These objects are then classified to extract the road. Shi *et al.* [27] first used a general adaptive neighborhood and local Geary's C to implement spectral-spatial classification to segment images. Then, road shape features, locally weighted regression and tensor voting were used to generate road centerlines and road networks. Maboudi *et al*. [28] applied a well-established multiresolution segmentation approach to create nonoverlapping regions, and then a fuzzy logic system and the ant colony optimization were used to analyze spatial, spectral, and textural object descriptors and extract road objects. The first image segmentation step is one of the most fundamental stages of OBIA, and it directly affects the precision and recall of road extraction.

C. Deep-Learning-Based Approaches for Road Extraction

With the great success of CNNs in the field of computer vision, deep learning approaches have been increasingly used in the field of remote sensing and have produced state-of-the-art results. Zhang *et al.* [14] proposed a deep ResUNet to extract roads from aerial images based on UNet. Tao *et al*. [29] proposed a spatial information inference network to capture and transmit road-specific contextual information, and this method displayed good road continuity performance. Zhang *et al*. [18] used a generative adversarial network to build an end-to-end framework for road extraction. Gao *et al*. [16] proposed a multiple feature pyramid network that combined feature pyramids and pyramid pooling to capture contextual information. Rezaee and Zhang [30] redesigned a patch-based deep neural network to detect roads in the Fredericton dataset, and the results showed that this method was better than SVM. Abdollahi *et al*. [31] proposed an end-to-end fully CNN to produce a high-resolution road segmentation map, and they combined CE and dice loss to

decrease the class imbalance influence. Li *et al*. [32] proposed an improved neural network D-Linknetplus based on D-LinkNet to extract roads from UAV remote sensing images. Wei *et al*. [33] designed a boosting segmentation framework to extract the road surface and road centerline simultaneously, and they also synthesized CE and dice loss. These deep-learning-based approaches achieved good results, but they only improved from the networks used and did not address road breakage issues caused by vegetation and buildings. They hardly simultaneously consider the hard-classified samples and class imbalance issue of the road. The above will reduce the accuracy of road extraction in complex environments.

D. Postprocessing of Road Extraction Data

In the traditional feature-based and object-based approaches and the current deep learning methods, the extracted roads are often noncontinuous due to vegetation and building occlusion issues. These road segments cannot be directly processed by computer vision. Spatial connectivity is an important attribute of roads, so postprocessing steps to assess road connections are necessary. Samet *et al.* [34] proposed a nontrivial semiautomatic approach to fill gaps in contour lines based on local and geometric properties. Gao *et al.* [35] used a tensor voting algorithm to reduce broken regions and improve the topological expressions of roads. Fan *et al*. [36] proposed an optimization method for broken road connections, and the approach included road breakpoint detection, polynomial fitting and pixel filling. In general, the postprocessing of extracted roads is commonly considered in traditional road extraction methods but less frequently used in deep learning methods because researchers have been more inclined to improve the structure of the neural network itself than to form an end-to-end processing system to improve accuracy and efficiency.

III. Method

In this article, a novel deep CNN (E-UNet) based on ensemble learning for semantic image segmentation is proposed to extract roads from remote sensing imagery. In the following subsections, we describe the basic structure of the proposed E-UNet and discuss the designed loss function and postprocessing method.

A. Structure of E-UNet

As one of the network structures widely used in academia and industry at present, UNet [10] has achieved good segmentation results through the feature fusion of different levels through skip connections. In our research, we use UNet as the basic model to build a more stable and robust ensemble learning model (E-UNet). We have simplified the process of ensemble learning proposed in [20], and resplit the various layers of UNet in our work. The construction process of E-UNet is as follows. In the initial stage, we make *N* copies of a single submodel to construct an integrated model *E* with *N* parallel submodels. Assuming that each submodel contains *Q* layers of CNNs, then *E* can be defined as

$$E = \begin{bmatrix} E_1 \\ E_2 \\ \dots \\ E_N \end{bmatrix} = \begin{bmatrix} L_{11} \dots L_{1Q} \\ L_{21} \dots L_{2Q} \\ \dots \dots \dots \\ L_{N1} \dots L_{NQ} \end{bmatrix} \tag{1}$$

where L_{nq} represents the weight and bias of the *q*th layer of submodel *n*. For the *q*th ($1 \leq q \leq Q$) layer of model *E*, we then construct a 1×*N* random matrix M_q to randomly select a layer of the submodels as the *q*th layer L_q for forward and backward propagation. L_q can be calculated as

$$L_q = M_q \begin{bmatrix} L_{1q} \\ L_{2q} \\ \dots \\ L_{Nq} \end{bmatrix} \tag{2}$$

$$M_q = \begin{bmatrix} X_{1q} \dots X_{Nq} \end{bmatrix} \tag{3}$$

where X_{nq} is a random variable and the value of X_{nq} is defined as

$$X_{nq} = \begin{cases} 1, if\ L_{nq}\ is \text{ selected.} \\ 0, \quad \text{otherwise.} \end{cases} \tag{4}$$

During the *i*th iteration of training, E-UNet can then be defined as

$$E_i = \begin{bmatrix} L_1 \dots L_q \dots L_Q \end{bmatrix}. \tag{5}$$

After forward propagation, the loss is calculated through the loss function, and the parameters of each layer in the submodels are updated by back propagation. During the *j*th iteration of training, a new random matrix is constructed for each layer to generate a new propagation path, and then the above steps are repeated. The overview of proposed E-UNet is shown in Fig. 1. The detailed training procedure of E-UNet is also given in Algorithm 1. In the model prediction phase, the prediction P_x of sample *x* is the average of each submodel, and the calculation formula is as follows:

$$P_x = \frac{1}{N} \sum_{i=1}^{N} E_i(x). \tag{6}$$

Therefore, the ensemble strategy of E-UNet is mainly embodied in two aspects. First, the random combination of different layers of submodels in the training process can improve the robustness of the model. Theoretically, for *N* submodels each with *Q* layers of CNNs, a total of N^Q new models may be randomly combined for training. Random combination is an effective mechanism to induce model diversity, which helps to extract complex and heterogeneous road features through more model parameters and more diversified model structures. The purpose of E-UNet is to search for a powerful ensemble model with strong generalization ability and high robustness in the hypothesis space by combining multiple simple models. Second, using the average output of the *N* submodels as the prediction result in the test process can reduce the deviation and improve the predictive performance.

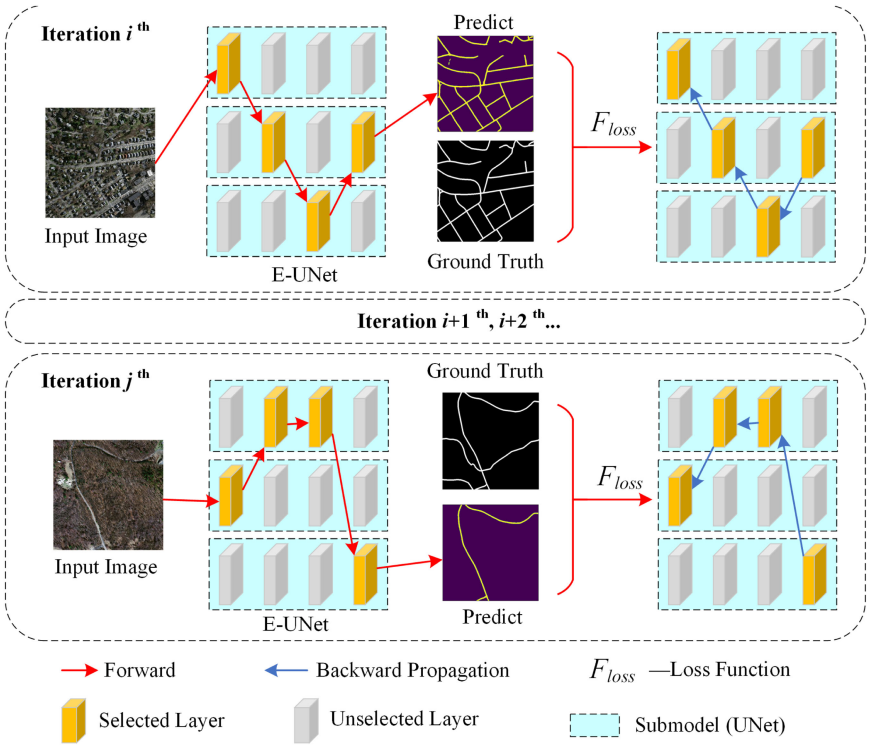


Fig. 1. Overview of E-UNet training.

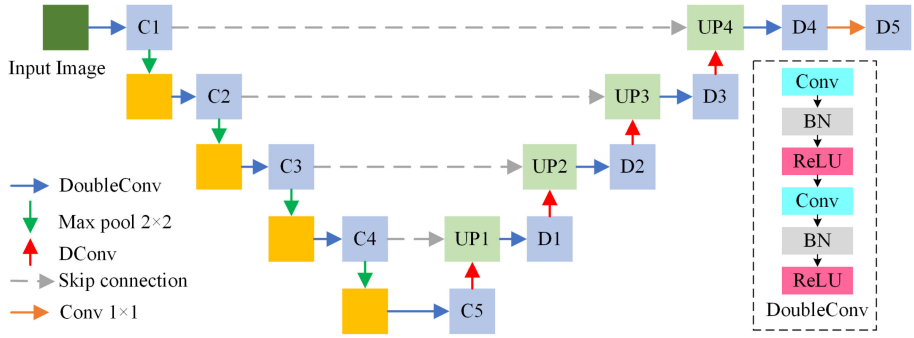


Fig. 2. Architecture of UNet.

Fig. 2 shows the basic model structure of the UNet we use. The contracting path performs deep feature extraction through convolution and pooling operations, and then the expansive path recovers the detailed information and structural information associated with roads through convolution and upsampling modules; its specific network parameters are given in Table I. Excluding the max pooling and concatenation operations, we divide UNet into 14 modules (layers), including 10 DoubleConvs in the contracting path and expansive path and 4 upsampling (deconvolution) layers in the expansive path. Therefore, $Q = 14$ in E-UNet.

B. Loss Function

Two issues in road semantic segmentation are the sparseness of roads and the imbalance of samples. The pixels in a certain road category account for a relatively small part of the entire image, and most of the pixels include the image background. In road extraction research, the traditional mean squared error (MSE) [14] and CE [15], [35] are mainly used.

For binary classification, CE can be defined as

$$\mathrm{CE}_{ij}^{(k)} = -\left[y_{ij}^{(k)} \log \hat{y}_{ij}^{(k)} + \left(1 - y_{ij}^{(k)}\right) \log\left(1 - \log \hat{y}_{ij}^{(k)}\right)\right] \tag{7}$$

where $y_{ij}^{(k)}$ and $\hat{y}_{ij}^{(k)}$ represent the ground truth and estimated probability for roads at location (i, j) in sample k, respectively. If the pixel represents a road, the ground truth value is 1; otherwise, it is 0. For minibatch training, the total loss of CE can be calculated as follows:

$$L_{\text{total}} = \frac{1}{K \times W \times H} \sum_{k=1}^{K} \sum_{i=1}^{W} \sum_{j=1}^{H} CE_{ij}^{(k)} \tag{8}$$

TABLE I
NETWORK PARAMETERS OF UNET

Module (Layer)	Operator	Kernel size	Stride	Padding	Output size (CxHxW)
C1	Conv1	3×3	1	1	64×512×512
	Conv2	3×3	1	1	64×512×512
	Pool1	2×2	2	0	64×256×256
C2	Conv1	3×3	1	1	128×256×256
	Conv2	3×3	1	1	128×256×256
	Pool2	2×2	2	0	128×128×128
C3	Conv1	3×3	1	1	256×128×128
	Conv2	3×3	1	1	256×128×128
	Pool3	2×2	2	0	256×64×64
C4	Conv1	3×3	1	1	512×64×64
	Conv2	3×3	1	1	512×64×64
	Pool4	2×2	2	0	512×32×32
C5	Conv1	3×3	1	1	1024×32×32
	Conv2	3×3	1	1	1024×32×32
UP1	DConv	2×2	2	0	512×64×64
	Concat	-	-	-	1024×64×64
D1	Conv1	3×3	1	1	512×64×64
	Conv2	3×3	1	1	512×64×64
UP2	DConv	2×2	2	0	256×128×128
	Concat	-	-	-	512×128×128
D2	Conv1	3×3	1	1	256×128×128
	Conv2	3×3	1	1	256×128×128
UP3	DConv	2×2	2	0	128×256×256
	Concat	-	-	-	256×256×256
D3	Conv1	3×3	1	1	128×256×256
	Conv2	3×3	1	1	128×256×256
UP4	DConv	2×2	2	0	64×512×512
	Concat	-	-	-	128×512×512
D4	Conv1	3×3	1	1	64×512×512
	Conv2	3×3	1	1	64×512×512
D5	Conv1	1×1	1	0	1×512×512

Algorithm 1: Training procedure of E-UNet.

Input: training dataset *X*. Batch size *K*. Training epochs *I*. The number of sub-models included in E-UNet is *N*. The number of network layers of UNet is *Q*.
Initialize a single submodel *U* (UNet)
Copy *N* submodels to form a model list *UList* (E-UNet)
for epoch = 01 … *I* **do**
  for batch = 12 … *X*/*K* **do**
    iteration = epoch∗ *X*/*K*+ batch
    Create an empty array *A* with size *Q*
    for layer *q* = 01 … *Q*-1 **do**
      A[*q*] = random(0, *N*-1)
      submodel *R* = *UList*[*A*[*q*]]
      Update weight, $Layer_U^q = Layer_R^q$
    end for
    outputs = forward (images, *U*)
    Loss (outputs, labels)
    Loss.backward()
    Optimizer.step()
    for layer *q* = 01 … *Q*-1 **do**
      submodel *R* = *UList*[*A*[*q*]]
      Update weight, $Layer_R^q = Layer_U^q$
    end for
  end for
end for

where *K*, *W*, and *H* represent the mini-batch size, width and height of the sample, respectively. As seen from the above formula, the loss function gives the same weight to each pixel in an image. The total loss is calculated as the average per pixel without considering the imbalance among classes or the difficulty of classifying various samples.

In fact, background objects and wide roads are easier to identify and classify than narrow and segmented roads; thus, the former is the focus of model optimization, and the latter is generally ignored when determining the overall loss. Inspired by focal loss [37], we first introduced a modulating factor for the CE loss to focus on the samples that were difficult to classify during training. The weighted CE loss can be defined as follows:

$$\begin{aligned}&\mathrm{WCE}_{ij}^{(k)}\\&=-\left[y_{ij}^{(k)}\left(1-\hat{y}_{ij}^{(k)}\right)^{\gamma}\log\hat{y}_{ij}^{(k)}+\hat{y}_{ij}^{(k)^{\gamma}}\left(1-y_{ij}^{(k)}\right)\log\left(1-\log\hat{y}_{ij}^{(k)}\right)\right]\\&=\begin{cases}-\left(1-\hat{y}_{ij}^{(k)}\right)^{\gamma}\log\hat{y}_{ij}^{(k)}\ if\ y_{ij}^{(k)}=1\\-\hat{y}_{ij}^{(k)^{\gamma}}\log\left(1-\log\hat{y}_{ij}^{(k)}\right)\ \text{otherwise}\end{cases}\end{aligned}\tag{9}$$

where $(1-\hat{y}_{ij}^{(k)})^{\gamma}$is a modulating factor that downweights the loss assigned to well-classified samples and upweights the loss assigned to hard-classified samples. γ is used to adjust the degree of downweighting and upweighting. We adopted the conclusion in [37] and set $\gamma=2$. $\mathrm{WCE}_{ij}^{(k)}$ is a pixel-level loss, while the problem of class imbalance is reflected at the image level, so we introduce dice loss [38] to compensate for the lack of WCE in representing class imbalance. The dice loss of sample *k* can be defined as

$$\mathrm{Dice}^{(k)}=1-\frac{2\sum_{i=1}^{W}\sum_{j=1}^{H}\left|y_{ij}^{(k)}\hat{y}_{ij}^{(k)}\right|}{\sum_{i=1}^{W}\sum_{j=1}^{H}\left(\left|y_{ij}^{(k)}\right|+\left|\hat{y}_{ij}^{(k)}\right|\right)}.\tag{10}$$

Finally, the complete loss function we designed is as follows:

$$L^{(k)}=(1-\alpha)\sum_{i=1}^{W}\sum_{j=1}^{H}\mathrm{WCE}_{ij}^{(k)}+\alpha\mathrm{Dice}^{(k)}\tag{11}$$

where $L^{(k)}$ is the loss of sample *k*. $\mathrm{Dice}^{(k)}$ is mainly used to mitigate the imbalance among classes, $\mathrm{WCE}_{ij}^{(k)}$ is used to improve the classification of hard pixels, and α is a hyperparameter used to adjust the contributions of these two losses to the total loss.

C. Postprocessing

A deep CNN cannot solve road breakage problems when extracting roads from an image. These broken roads may be due to narrow road widths or road blockages caused by vegetation, buildings, or other nonroad objects. Inspired by Manandhar *et al.* [39], we propose a linear region-growing algorithm to solve road breakage and disconnection problems.

1) Detection of Potential Road Breakpoints: The result of road extraction using a deep neural network is a binary image of a single band. Each pixel is either the foreground (road) or the background (nonroad). If a small change in any direction

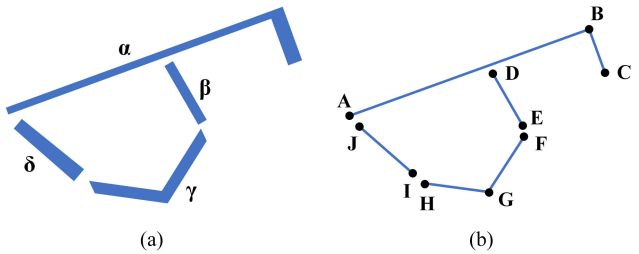


Fig. 3. Postprocessing of extracted roads. (a) Example of road segments. (b) Skeleton lines and nodes of road segments.

causes a large change in the grayscale, then the corresponding point is called a corner point or feature point. Road breakpoints and inflection points are obviously points of interest. Fig. 3(a) shows the road segments identified by the deep learning model. Each road segment can be considered an independent connected domain (α, β, γ, and δ). We first use the Zhang–Suen thinning algorithm [40] to obtain the skeleton lines of road segments, as shown in Fig. 3(b). We then use the Shi–Tomasi corner detector [41], which is improved on the basis of the Harris corner [42], to find the potential road breakpoints. The basic principle is as follows. Given a shift $(\Delta x, \Delta y)$ and a point (x, y) in a grayscale image, the change in pixel value is defined as

$$c\ (\Delta x, \Delta y) = \sum_{(x,y)} w\,(x,y)\,[I\,(x+\Delta x, y+\Delta y) - I\,(x,y)]^2 \tag{12}$$

where *w()* denotes the window function and the simplest case is $w = 1$. *I(x, y)* is the gray value at location *(x, y)*. The formula can be approximated by a Taylor expansion as follows:

$$c\ (\Delta x, \Delta y) = [\Delta x\ \Delta y]\ M \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{13}$$

where M is a matrix and λ_1 and λ_2 are the eigenvalues of M. These eigenvalues determine whether a region is a corner, an edge, or a plane. Finally, a corner response function RF is used to detect corners.

$$\mathrm{RF}\ = \min\left(\lambda_1, \lambda_2\right). \tag{14}$$

2) Find the Nearest Node: We regard the potential road breakpoints on each road segment as the nodes of the connected domain, represented by capital letters A to J in Fig. 3(b). Therefore, connecting broken roads actually connects nodes located on different connected domains. The closer two nodes are, the more similar the two nodes are, and the more likely they are to belong to the same road. For this hypothesis, we use the Euclidean distance as the connection cost, and two nodes with the minimum cost on different connected domains are connected. As shown in Fig. 3(b), we randomly select a node (for example, node A) in connected domain α as the initial node and then calculate the cost between the nodes in β, γ, and δ and node A until all nodes in all domains are calculated. The nearest node and minimum cost of each node are given in Table II.

We set a hyperparameter L; if the minimum cost of the two nodes is less than L, then the two nodes are added to the list to be connected; otherwise, they are abandoned. In addition, different connected domains are connected only once. For example, if we set $L = 10$, we first start from the connected domain α, nodes *A* and *J* are considered to be connectable, and the minimum cost of *B* and *C* is greater than 10 and is not considered. We then move to connected domain β, where we connect node E to node *F*. The complete node connection sequence is given in Table III.

TABLE II
NEAREST NODE AND MINIMUM COST OF EACH NODE

Road Segment (Connected Domain)	Node	Nearest Node	Minimum Cost
α	A	J	3
	B	D	20
	C	E	19
β	D	F	13
	E	F	2
γ	F	E	2
	G	E	14
	H	I	3
δ	I	H	3
	J	A	3

TABLE III
NODE CONNECTION SEQUENCE

Connected nodes	Connected Domain
A→J	α→δ
E→F	β→γ
H→I	γ→δ

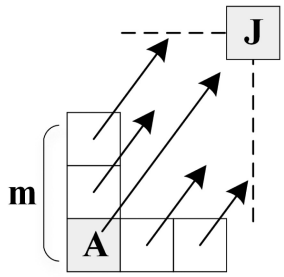


Fig. 4. Linear region growing.

3) Linear Region Growing: A linear region-growing algorithm is designed to connect breakpoints based on region growing [43]. Nodes in Fig. 3(b) are single-coordinate points, but the road in the input image is several pixels wide. In other words, the road area is a raster image rather than a line vector. Therefore, it is necessary to estimate the width of the roads and design an algorithm to connect the middle region of the fracture according to the two points. As shown in Fig. 4, each square represents a pixel, with point A as the center used to construct an "L"-shaped growth template. If *A* is in the upper left corner of *J*, then the template has an inverted "L" shape, and *m* is the template size. We use the average road width of two segments to be connected to calculate *m*, and the formula is as follows:

$$m_{AJ} = \frac{1}{2} \times \left(\frac{V_i}{T_i} + \frac{V_j}{T_j}\right) \tag{15}$$

where i and j represent the connected domain where nodes *A* and *J* are located, respectively. V_i is the number of pixels in connected domain i before thinning, and T_i is the number of pixels in connected domain i after thinning. Unlike the requirement in the region-growing algorithm, there is no need to select the

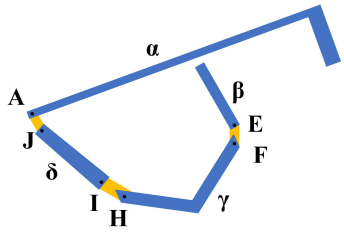


Fig. 5. Connection result.

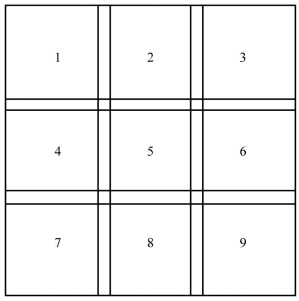


Fig. 6. Overlapping cropping for MRD images.

seed point in this case, and the pixels in the growth template are directly used as the initial seed points. Our linear region-growing algorithm uses the slope of the line *AJ* as the growth direction and stops the growth when the *x*–*y* coordinates of the pixel reach point *J*. Compared with the approach of growth in any direction for four neighbors or eight neighbors, the proposed algorithm is much more efficient. Finally, the region of growth between two nodes serves as the connection for the broken segments, as shown in Fig. 5.

IV. Datasets and Preprocessing

A. Datasets

To verify the accuracy of our model, the Massachusetts road dataset (MRD) [44] and the DeepGlobe 2018 road extraction challenge dataset (DRECD) [45] were used for road extraction. The MRD covers an area of approximately 2600 km^2 and spans urban, suburban and rural areas of Massachusetts, USA. The dataset consists of 1171 images in total, including 1108 images that were used for training, 14 images for validation, and 49 images for testing. Each image is 1500 × 1500 pixels with a resolution of 1.2 m/pixel. The DRECD consists of 6226 satellite images of 1024 × 1024 pixels and a resolution of 0.5 m/pixel. The dataset covers approximately 1632 km^2 of land in Thailand, Indonesia, and India. The original image size in these two datasets is large, and the roads are not evenly distributed; therefore, data preprocessing is necessary due to memory constraints.

B. Data Preprocessing

DRECD images (1024 × 1024) were directly divided into 4 blocks (512 × 512) of equal area. For MRD images, an overlapping cropping method was used to divide the sample into nine blocks with a size of 512 × 512 pixels, as shown in Fig. 6. After cropping, there were many samples with no roads or few roads in the new samples; such issues can lead to imbalanced samples in the training process and affect the convergence of the model. To solve this problem, we designed a ratio β to measure the sparseness of roads in a single sample. Assuming that the number of foreground pixels in a labeled image is x and the number of background (nonroad) pixels is y, then $\beta = x/y$. Through analyzing a large number of samples, we set the threshold to 0.02, and samples with $\beta > 0.02$ were retained. We finally obtained 11442 MRD samples and 14603 DRECD samples, which were divided into training set, validation set and test set according to 7:2:1, respectively.

V. Experiments and Analysis

A. Overall Details of the Experiments

1) Evaluation Metrics: To quantitatively evaluate the performance of different road extraction methods, precision, recall, overall accuracy (OA), F1 score [46] and intersection over union (IoU) [47] were used as metrics. OA measures the accuracy of road and nonroad identification at the pixel level and can be calculated as follows:

$$\mathrm{OA} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}} \tag{16}$$

where TP, FP, TN, and FN represent the numbers of true positives, false positives, true negatives and false negatives at the pixel level, respectively. The F1 score is an indicator used to measure the accuracy of binary classification in statistics, and it is calculated based on the precision (P) and recall (R)

$$F1 = \frac{2 \times P \times R}{P+R} \tag{17}$$

where

$$P = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \; R = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}. \tag{18}$$

The IoU is a commonly used evaluation metric in semantic segmentation, and it is the ratio of overlap between the true area and predicted area considering the total area. Specifically, IoU can be calculated as follows:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}. \tag{19}$$

2) Training Details: The proposed neural network was implemented using PyTorch [48], and all experiments were executed on a supercomputing platform with Nvidia Tesla V100 (16 GB) GPUs. We used L2 regularization [49] to prevent overfitting and used the Adam (adaptive moment estimation) [50] optimizer to minimize losses and update parameters. The learning rate was initially set to 0.001. The plateau decay strategy was used, and the learning rate was halved if the epoch loss did not decrease for three consecutive epochs. The network was trained on the training set with a batch size of 16 for 100 epochs, and if the model performance was not improved for five consecutive epochs, the model stopped training early. Finally, the accuracy of the network was evaluated based on the test set. According to the above details, we conducted road extraction experiments on the MRD and DRECD.

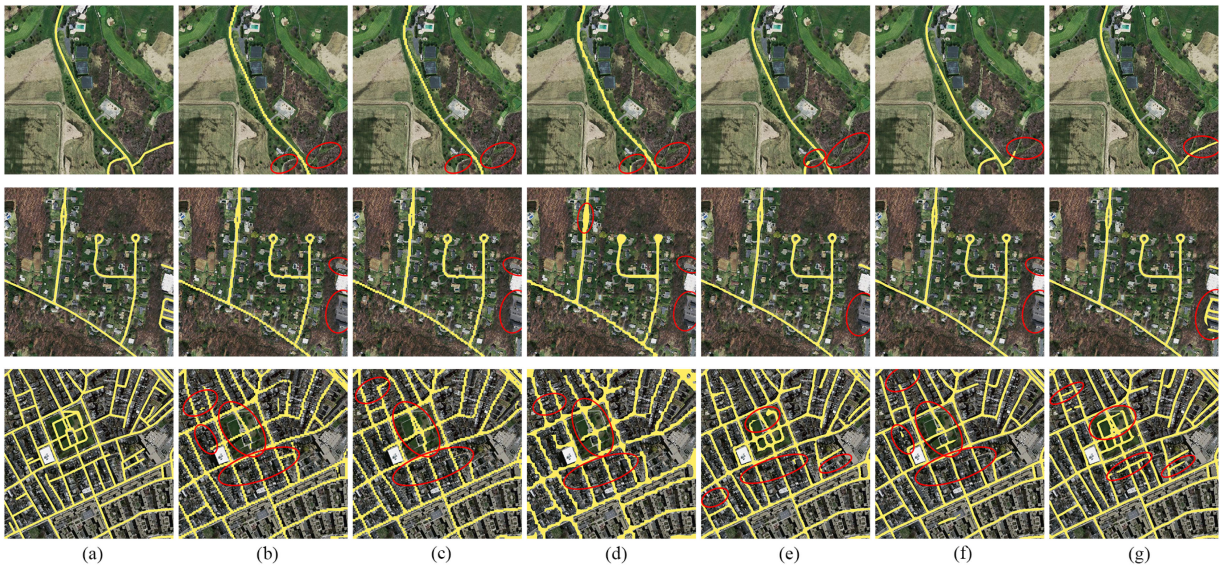


Fig. 7. Road extraction results for the MRD. (a) Ground truth. (b) HRNet-32. (c) HRNet-48. (d) DeepLab V3. (e) PSPNet. (f) UNet. (g) Our proposed model.

B. Experiment with the MRD

We compared our proposed network with five semantic segmentation-based road extraction methods: HRNet-32 [51], HRNet-48 [51], DeepLab V3, PSPNet, GAN [18], and UNet. As shown in Fig. 7, the yellow line is the actual road, and the red ellipses represent the areas where the roads extracted by different extraction methods have significant differences. The first row in Fig. 7 is an area with sparse buildings and dense vegetation coverage. Our model effectively mines high-level semantic information and accurately extracts roads blocked by vegetation. A suburban area where buildings and vegetation are evenly distributed is shown in the second row. Compared with other methods, our model is able to eliminate the interference caused by buildings and extract detailed and complex road information. The third row is a typical complex urban environment where roads and buildings are densely distributed and there is a high degree of similarity among spectral and texture characteristics; here, the roads are highly influenced by buildings. The results show that the proposed model has better spatial reasoning ability and multilevel contextual information mining ability than the other methods, and the results are approximate to the real road distribution. In addition, it can be seen from the comparison of different experimental results that the road boundaries identified by our model are smoother, which reduces the workload of postprocessing.

We also quantitatively analyzed the road extraction effects of different models. As given in Table IV, the precision, recall, OA, F1 score, and IoU of the proposed model are higher than those of the other models. From the experimental results, it can be found that UNet has a congenital advantage for road extraction on the MRD dataset, and the ensemble learning model we propose further magnifies this advantage. Specifically, E-UNet achieves increases of 2.18–12.43% for the F1 score and 3.0–17.02% for the IoU. In addition, our postprocessing method can improve the road extraction effect. The postprocessing actually corrects the pixels predicted by the CNN as the nonroad to road. As more roads on the image are extracted, the error rate also increases. Therefore, it is necessary to use comprehensive indicators such as F1 and IOU to balance precision and recall. As given in Table IV, after postprocessing, the recall increased while the precision and OA decreased, but F1 and IoU slightly improved. The postprocessing result will be shown and discussed in a later section.

C. Experiment with the DRECD

For the DeepGlobe dataset, we again compared the proposed network with HRNet-32, HRNet-48, DeepLab V3, PSPNet, GAN and UNet. As shown in Fig. 8, we chose areas with different road sparseness levels. Intuitively, the six models can effectively extract road information, but DeepLab V3 and UNet are relatively ineffective because they cannot identify road areas blocked by vegetation or buildings, and the extracted roads lack connectivity. In addition, the roads extracted by HRNet-32 and HRNet-48 have obvious jagged features. We further quantitatively analyzed the accuracies of different models, as shown in Table IV. In the existing models, UNet does not show similar predictive ability on the MRD dataset. In contrast, PSPNet performs better on the recall and F1 indicators. Our proposed method achieves better performance than the other methods based on precision, OA and IoU. Specifically, E-UNet achieves increases of 1.42%-6.13% for precision and 0.52–4.55% for the IoU. After postprocessing, compared with the other models, our model has better spatial reasoning ability for the DRECD dataset, thus ensuring the connectivity and integrity of the roads to the greatest extent.

D. Analysis of the Threshold α in the Loss Function

In this section, we assess the influence of the threshold α, an important parameter in the loss function. We use our model

TABLE IV
ROAD EXTRACTION RESULTS FOR MRD AND DRECD

Dataset	Method	P(%)	R(%)	OA (%)	F1 (%)	IoU (%)
MRD	HRNet 32	72.719	75.995	96.585	73.581	59.169
	HRNet 48	71.454	76.688	96.513	73.263	58.761
	PSPNet	71.985	70.927	96.300	69.613	55.138
	DeepLab V3	66.423	79.571	95.984	71.694	56.786
	GAN	68.965	67.114	95.716	68.027	51.546
	UNet	79.232	78.819	97.347	78.275	65.564
	E-UNet	**80.710**	**81.309**	**97.595**	**80.455**	**68.564**
	HRNet 32 + Postprocessing	72.574	76.185	96.586	73.609	59.201
	HRNet 48+ Postprocessing	71.451	76.725	96.513	73.284	58.783
	PSPNet+ Postprocessing	67.592	75.253	96.289	69.840	55.340
	DeepLab V3+ Postprocessing	68.184	77.907	95.982	71.736	56.833
	GAN+ Postprocessing	67.742	70.470	95.716	69.079	52.764
	UNet+ Postprocessing	77.842	80.285	97.347	78.348	65.653
	E-UNet+ Postprocessing	**79.772**	**82.335**	**97.594**	**80.522**	**68.650**
DRECD	HRNet 32	76.583	78.029	96.903	76.122	62.814
	HRNet 48	76.890	77.475	96.899	76.065	62.693
	PSPNet	78.158	**79.158**	97.188	**77.839**	65.141
	DeepLab V3	75.113	78.184	96.808	75.414	62.220
	GAN	75.000	76.744	96.133	75.862	61.111
	UNet	79.705	76.827	97.226	77.099	64.483
	E-UNet	**81.128**	77.021	**97.362**	77.810	**65.660**
	HRNet 32 + Postprocessing	75.998	78.718	96.905	76.192	62.891
	HRNet 48+ Postprocessing	76.144	78.312	96.901	76.128	62.767
	PSPNet+ Postprocessing	**77.455**	79.968	97.187	77.923	65.243
	DeepLab V3+ Postprocessing	73.939	79.495	96.803	75.575	62.383
	GAN + Postprocessing	73.799	78.605	96.096	76.126	61.455
	UNet+ Postprocessing	77.037	80.366	97.234	77.658	65.162
	E-UNet+ Postprocessing	76.938	**81.439**	**97.357**	**78.090**	**65.982**

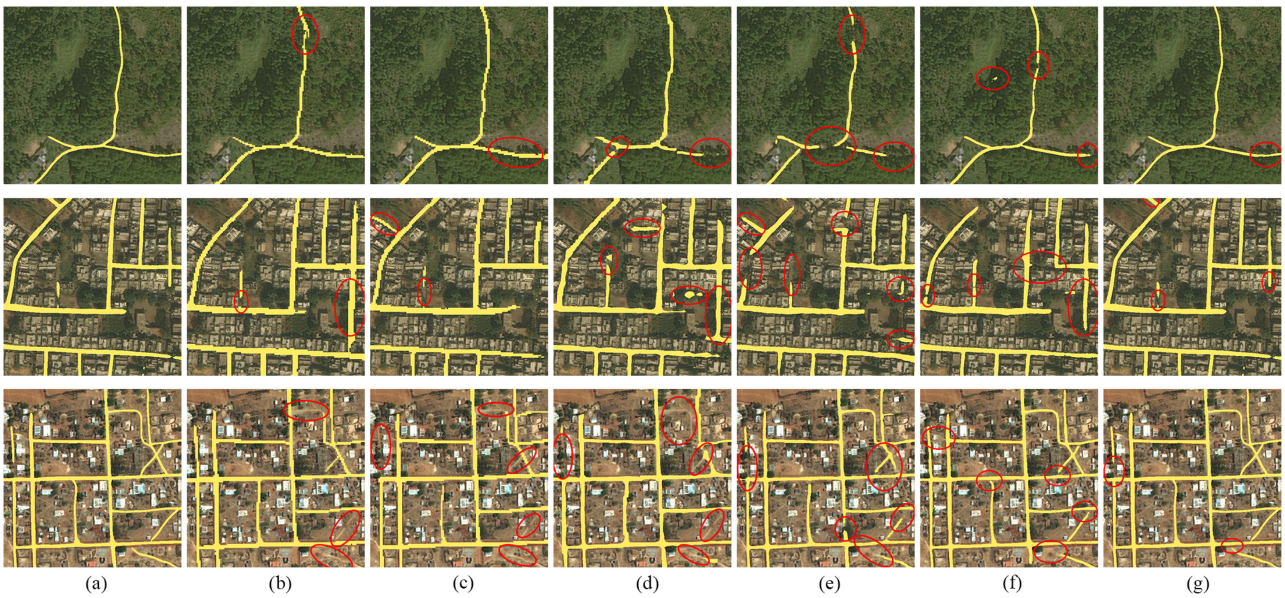


Fig. 8. Road extraction results for the DRECD. (a) Ground truth. (b) HRNet-32. (c) HRNet-48. (d) DeepLab V3. (e) PSPNet. (f) UNet. (g) Our proposed model.

to perform road extraction experiments based on the MRD and DRECD. The other parameters in the experiment are unchanged, and the value of α takes a value from 0 to 1, with an interval of 0.1. When $\alpha = 0$, our loss function becomes the focal loss function, and when $\alpha = 1$, it is the dice loss function. The visualization results during the model training process for Binary Cross Entropy loss, Dice loss, Focal loss and the loss we designed are shown in Fig. 9. The detailed experimental results are shown in Table V and Table VI. When $\alpha = 0.2$ for both datasets, the model achieves a balance between considering road sparsity and road classification. The dice loss function seeks to solve the problem of imbalance between the foreground and the background. The focal loss function considers the difficulty of classifying certain samples; for example, wide roads without vegetation are easier to identify than narrow roads with vegetation. When $\alpha = 0.2$, the accuracy is high, which indicates that the focal loss accounted for a larger proportion of the total loss than did the dice loss; notably, during the training process,

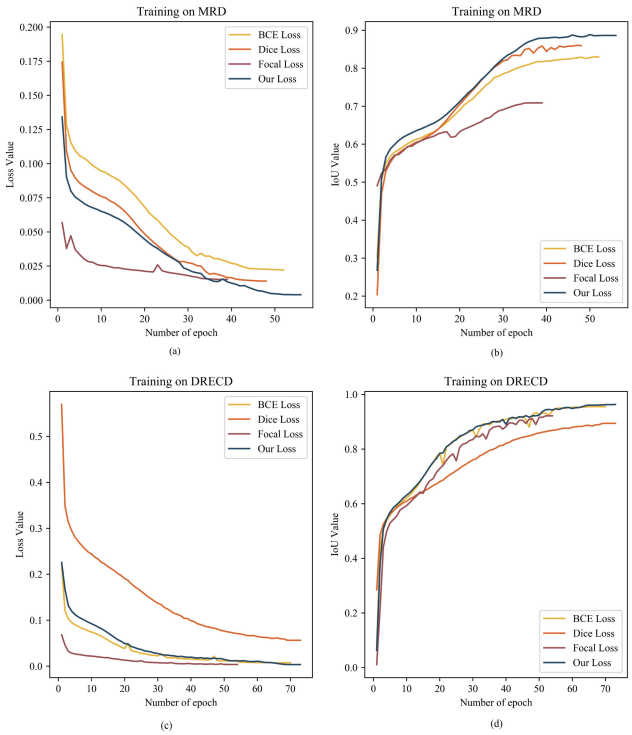


Fig. 9. Changes of loss and IoU while training on MRD and DRECD.

TABLE V
PERFORMANCE OF DIFFERENT THRESHOLDS α FOR THE MRD

loss	P(%)	R(%)	OA (%)	F1 (%)	IoU (%)
α=0 (focal loss)	80.203	74.112	97.189	76.122	62.843
α=0.1	80.685	78.232	97.445	78.789	66.266
α=0.2	80.710	**81.308**	97.596	**80.455**	**68.564**
α=0.3	81.468	80.383	**97.604**	80.226	68.370
α=0.4	79.790	80.204	97.465	79.319	67.016
α=0.5	79.686	78.501	97.371	78.260	65.623
α=0.6	80.775	78.795	97.474	78.966	66.638
α=0.7	81.515	79.649	97.574	79.884	67.866
α=0.8	81.261	79.529	97.533	79.635	67.518
α=0.9	79.918	78.916	97.412	78.599	66.104
α=1 (dice loss)	**81.573**	79.554	97.571	79.802	67.791

TABLE VI
PERFORMANCE OF DIFFERENT THRESHOLD α FOR THE DRECD

loss	P(%)	R(%)	OA (%)	F1 (%)	IoU (%)
α=0 (focal loss)	79.705	76.827	97.226	77.099	64.483
α=0.1	**81.135**	76.343	97.330	77.404	65.152
α=0.2	81.128	77.021	**97.362**	**77.810**	**65.659**
α=0.3	78.599	**78.074**	97.212	77.106	64.542
α=0.4	79.639	77.297	97.261	77.243	64.750
α=0.5	78.897	78.535	97.260	77.624	65.169
α=0.6	79.570	78.109	97.296	77.628	65.309
α=0.7	77.137	79.085	97.109	76.861	64.053
α=0.8	79.953	76.769	97.261	77.085	64.688
α=0.9	80.716	75.386	97.253	76.589	64.023
α=1 (dice loss)	80.268	76.382	97.259	77.134	64.586

the algorithm focuses more on difficult-to-classify samples to enhance the contextual reasoning ability of the model. This finding is closely related to our data preprocessing approach, in which data are filtered based on the foreground-to-background pixel ratio to solve the imbalance issue for roads and nonroads.

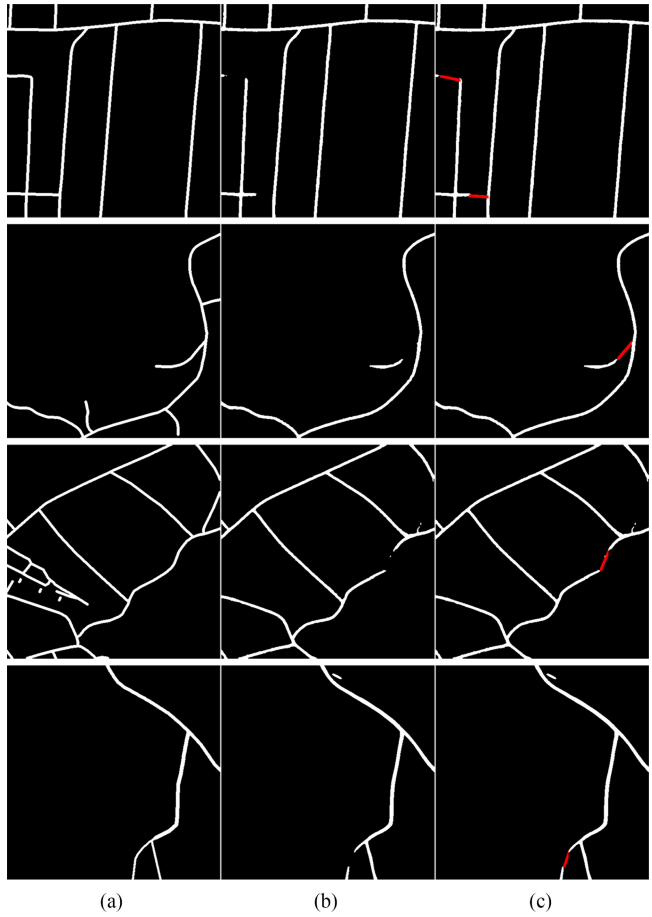


Fig. 10. Road extraction results after postprocessing. (a) Ground truth. (b) Model output. (c) Postprocessing result.

E. Analysis of Postprocessing

The purpose of the postprocessing stage is to connect road breakpoints. As shown in Fig. 10, the postprocessing algorithm we designed can automatically detect breakpoints and connect them. The accuracy of postprocessing depends on the hyperparameter L, which determines how far away two nodes are considered road breakpoints. If L is set too small, normal broken roads are not connected. If L is set too large, nonbroken points are connected. To select the optimal parameter value, we use the MRD and DRECD test sets to explore the relationship between the hyperparameter L and the accuracy of road extraction. According to our experience, we set the domain of L to $\{ L \mid L = 10\ l,\ 1 \leq l \leq 20\ \&\ l \in N* \}$. We use the precision, recall, F1 score and IoU as the evaluation metrics. The experimental results are shown in Fig. 11. In the four subfigures, the vertical axes on the left and right represent the evaluation values on the MRD dataset and DRECD dataset, respectively. We find that the optimal solution of the postprocessing hyperparameter L is the same when different datasets yield the best accuracy. Specifically, when L gradually increases, an increasing number of broken roads are connected, which means that an increasing number of roads in the remote sensing images are extracted, but at the same time, an increasing number of nonroad errors are introduced. Therefore, the recall rate increases, and the precision

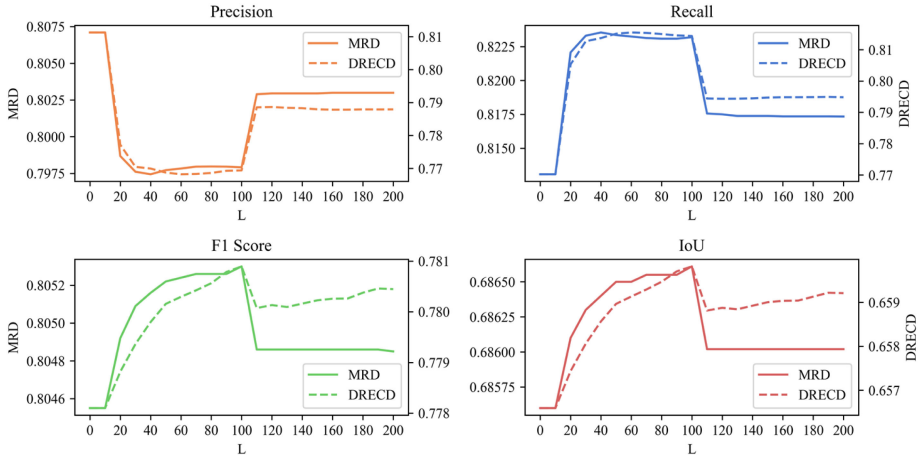


Fig. 11. Influence of L on the extraction accuracy for the MRD and DRECD.

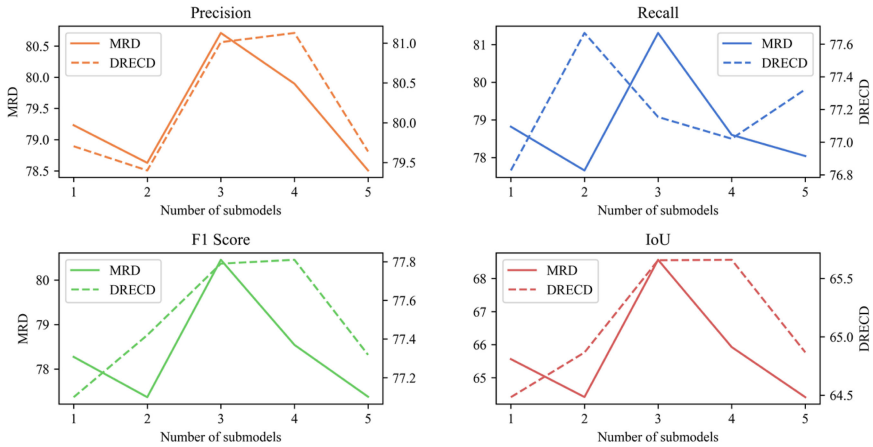


Fig. 12. Influence of number of submodels on the extraction accuracy for the MRD and DRECD.

TABLE VII
PERFORMANCE OF DIFFERENT N FOR MRD AND DRECD

dataset	N	P(%)	R(%)	OA (%)	F1 (%)	IoU (%)
MRD	N=1	79.232	78.819	97.347	78.275	65.564
	N=2	78.629	77.655	97.251	77.370	64.419
	N=3	**80.710**	**81.309**	**97.595**	**80.455**	**68.564**
	N=4	79.897	78.601	97.396	78.544	65.928
	N=5	78.507	78.038	97.231	77.380	64.414
DRECD	N=1	79.705	76.827	97.226	77.099	64.483
	N=2	79.401	**77.668**	97.256	77.420	64.866
	N=3	81.012	77.154	97.360	77.790	65.656
	N=4	**81.128**	77.021	**97.362**	**77.810**	**65.660**
	N=5	79.639	77.323	97.257	77.318	64.867

rate decreases, and when $L = 100$, the F1 score and IoU reach the maximum value. Subsequently, when L continues to increase, more fake breakpoints are connected, and finally, precision and recall reach a balance.

These results demonstrate that postprocessing steps can improve the connectivity of roads and compensate for the lack of visual features used by neural networks to retain road integrity. A road network with high connectivity is optimally generated by analyzing the geometric structures and feature points of roads. However, a drawback of the proposed method is that it relies too much on the feature point detection algorithm. For example, in Fig. 3(b), point D is likely to be connected to line segment AB, but no other feature points are detected on AB, which leads to this breakage not being connected. This issue is worthy of article in future research.

F. Structure Analysis of E-UNet

In our ensemble learning model, E-UNet is composed of N ordinary UNets. The value of N affects the fitting ability and

generalization ability of E-UNet. We used our model to perform road extraction experiments based on the MRD and DRECD. The other parameters in the experiment were unchanged, and the value of N was set from 1 to 5. The experimental results are given in Table VII and Fig. 12. With the increase in N, there are an increasing number of parameters of the model, and the fitting ability of the model becomes stronger. However, under the condition of a fixed number of training sets, the possibility of overfitting increases. In our experiment, for MRD and DRECD, when N is 3 and 4, the model has the highest accuracy. Although we used data augmentation, regularization and early stop strategies, we can see from the experimental results that overfitting occurs. How to simplify the model structure and adopt more effective training methods to avoid overfitting is the focus of our next study.

VI. Conclusion

In this article, we proposed a deep neural network based on ensemble learning for road extraction from remote sensing imagery. First, we use E-UNet to identify the road information in the image. In the process of encoding and decoding, the network weights and biases of E-UNet are transmitted through the random combination of layers of different submodels. A new loss function is designed to solve the class imbalance problem caused by road sparseness, and attention is given to samples that are difficult to classify to extract narrow and covered roads. We then use an effective algorithm to connect road breakpoints based on geometric features. Experiments on two datasets showed the advantages of the proposed method in road extraction from remote sensing imagery. Notably, our model infers the road information when roads are blocked by vegetation and buildings by integrating different models. The improved loss function can simultaneously solve the class imbalance issue associated with road sparseness and aid in narrow road classification. In addition, we investigated the performance of the model based on different values of the hyperparameter in the postprocessing step. In future work, we will focus on how to enhance the generalization ability of the model while simplifying the model parameters to avoid overfitting. In addition, we will apply the proposed method for other types of semantic segmentation and object detection.

Acknowledgment

The numerical calculations in this article were performed on the supercomputing system at the Supercomputing Center of Wuhan University.

References

[1] X. Yang, L. Tang, K. Stewart, Z. Dong, X. Zhang, and Q. Li, "Automatic change detection in lane-level road networks using GPS trajectories," *Int. J. Geographical Inf. Sci.*, vol. 32, no. 3, pp. 601–621, Mar. 2018.

[2] J. Qiu and R. Wang, "Automatic extraction of road networks from GPS traces," *Photogramm. Eng. Remote Sens.*, vol. 82, no. 8, pp. 593–604, Aug. 2016.

[3] M. Sushith and S. Sophia, "Extraction of road using soft computing techniques," *Soft Comput.*, vol. 23, no. 18, pp. 8487–8494, Apr. 2019.

[4] S. Leninisha and K. Vani, "Water flow based geometric active deformable model for road network," *ISPRS J. Photogramm. Remote Sens.*, vol. 102, pp. 140–147, Apr. 2015.

[5] A. Gruen and H. Li, "Semiautomatic linear feature extraction by dynamic programming and LSB-snakes," *Photogramm. Eng. Remote. Sens.*, vol. 63, no. 8, pp. 985–994, Aug. 1997.

[6] R. Fan, U. Ozgunalp, B. Hosking, M. Liu, and I. Pitas, "Pothole detection based on disparity transformation and road surface modeling," *IEEE Trans. Image Process.*, vol. 29, no. 1, pp. 897–908, Dec. 2020.

[7] J. Zhang, X. Liu, Z. Liu, and J. Shen, "Semiautomatic road tracking by template matching and distance transformation in urban areas," *Int. J. Remote Sens.*, vol. 32, no. 23, pp. 8331–8347, 2011.

[8] R. Zhang, J. Zhang, and H. Li, "Semiautomatic extraction of ribbon roads from high resolution remotely sensed imagery based on angular texture signature and profile match," *J. Remote Sens.*, vol. 12, no. 2, pp. 224–232, Mar. 2008.

[9] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[10] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervention*, 2015, pp. 234–241.

[11] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.

[12] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. "Semantic image segmentation with deep convolutional nets and fully connected CRFS," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[13] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[14] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual U-net," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 749–753, May 2018.

[15] Y. Xu, Z. Xie, Y. Feng, and Z. Chen, "Road extraction from high-resolution remote sensing imagery using deep learning," *Remote Sens*, vol. 10, no. 9, 2018.

[16] X. Gao *et al.*, "An end-to-end neural network for road extraction from remote sensing imagery by multiple feature pyramid network," *IEEE Access*, vol. 6, pp. 39401–39414, 2018.

[17] X. Zhang, W. Ma, C. Li, J. Wu, X. Tang, and L. Jiao, "Fully convolutional network-based ensemble method for road extraction from aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 10, pp. 1777–1781, Oct. 2020.

[18] X. Zhang, X. Han, C. Li, X. Tang, H. Zhou, and L. Jiao, "Aerial image road extraction based on an improved generative adversarial network," *Remote Sens.*, vol. 11, no. 8, pp. 930–949, 2019.

[19] Y. Liu, J. Yao, X. Lu, M. Xia, X. Wang, and Y. Liu, "RoadNet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2043–2056, Apr. 2019.

[20] H. J. Lee, S. T. Kim, H. Lee, N. Navab, and Y. M. Ro, "Efficient ensemble model generation for uncertainty estimation with Bayesian approximation in segmentation," 2020, *arXiv:2005.10754*.

[21] A. Ashukha, A. Lyzhov, D. Molchanov, and D. Vetrov, "Pitfalls of in-domain uncertainty estimation and ensembling in deep learning," in *Proc. 8th Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–11.

[22] F. Tupin, H. Maitre, J. Mangin, J. Nicolas, and E. Pechersky, "Detection of linear features in SAR images: Application to road network extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 2, pp. 434–453, Mar. 1998.

[23] R. Gaetano, J. Zerubia, G. Scarpa, and G. Poggi, "Morphological road segmentation in urban areas from high resolution satellite images," in *Proc. 17th Int. Conf. Digit. Signal Process.*, 2011, pp. 1–8.

[24] J. Liu, Q. Qin, J. Li, and Y. Li, "Rural road extraction from high-resolution remote sensing images based on geometric feature inference," *ISPRS Int. J. Geo-Inf.*, vol. 6, no. 10, p. 314, 2017.

[25] D. Chaudhuri, N. K. Kushwaha, and A. Samal, "Semiautomated road detection from high resolution satellite images by directional morphological enhancement and segmentation techniques," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 5, pp. 1538–1544, Oct. 2012.

[26] Y. Cao, Z. Wang, L. Shen, X. Xiao, and L. Yang, "Fusion of pixel-based and object-based features for road centerline extraction from high-resolution satellite imagery," *Acta Geodaetica et Cartographica Sinica*, vol. 45, no. 10, pp. 1231–1240, 2016.

[27] W. Shi, Z. Miao, and J. Debayle, "An integrated method for urban main-road centerline extraction from optical remotely sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3359–3372, Jun. 2014.

[28] M. Maboudi, J. Amini, S. Malihi, and M. Hahn, "Integrating fuzzy object based image analysis and ant colony optimization for road extraction from remotely sensed images," *ISPRS J. Photogramm. Remote Sens.*, vol. 138, pp. 151–163, Apr. 2018.

[29] C. Tao, J. Qi, Y. Li, H. Wang, and H. Li, "Spatial information inference net: Road extraction using road-specific contextual information," *ISPRS J. Photogramm. Remote Sens.*, vol. 158, pp. 155–166, 2019.

[30] M. Rezaee and Y. Zhang, "Road detection using deep neural network in high spatial resolution images," in *Proc. Joint Urban Remote Sens. Event*, 2017, pp. 1–4.

[31] A. Abdollahi, B. Pradhan, and A. Alamri, "VNet: An End-to-End fully convolutional neural network for road extraction from high-resolution remote sensing data," *IEEE Access*, vol. 8, pp. 179424–179436, 2020, doi: 10.1109/ACCESS.2020.3026658.

[32] Y. Li, B. Peng, L. He, K. Fan, Z. Li, and L. Tong, "Road extraction from unmanned aerial vehicle remote sensing images based on improved neural networks," *Sensors*, vol. 19, no. 19, pp. 4115, Sep. 2019.

[33] Y. Wei, K. Zhang, and S. Ji, "Simultaneous road surface and centerline extraction from large-scale remote sensing images using CNN-Based segmentation and tracing," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8919–8931, Dec. 2020.

[34] R. Samet and E. Hancer, "A new approach to the reconstruction of contour lines extracted from topographic maps," *J. Vis. Commun. Image Represent.*, vol. 23, no. 4, pp. 642–647, 2012.

[35] L. Gao, W. Song, J. Dai, and Y. Chen, "Road extraction from high-resolution remote sensing imagery using refined deep residual convolutional neural network," *Remote Sens*, vol. 11, no. 5, pp. 552–567, 2019.

[36] D. L. Fan, B. Wang, Z. L. Chen, and L. Wang, "Research on broken road connection method after road extraction from high-resolution remote sensing image," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 42, pp. 387–395, 2020.

[37] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[38] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. IC3DV*, pp. 565–571, Oct. 2016.

[39] P. Manandhar P, P. R. Marpu, Z. Aung, and F. Melgani, "Towards automatic extraction and updating of VGI-Based road networks using deep learning," *Remote Sens.*, vol. 11, no. 9, pp. 1012–1122, 2019.

[40] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Commun. ACM*, vol. 27, pp. 236–239, 1984.

[41] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 1994, pp. 593–600.

[42] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. 4th Alvey Vis. Conf,* 1988, vol. 15, pp. 147–151.

[43] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 641–647, Jun. 1994.

[44] V. Mnih and G. E. Hinton, "Learning to detect roads in high-resolution aerial images," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 210–223.

[45] I. Demir *et al.*, "DeepGlobe 2018: A challenge to parse the earth through satellite images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 172–181.

[46] D. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using brightness and texture," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 1–8.

[47] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.

[48] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8024–8035.

[49] C. Cortes, M. Mohri, and A. Rostamizadeh, "L2 regularization for learning kernels," in *Proc. 25th Conf. Uncertainty Artif. Intell.*, 2009, pp. 109–116.

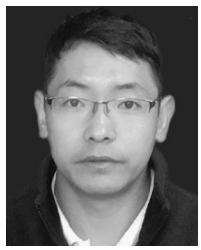
[50] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–41.

[51] J. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Apr. 2020.

Junjie Li received the B.S. degree in geographic information science from Central China Normal University, Wuhan, China, in 2018. He is currently working toward the Ph.D. degree at the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan.

His research interests include deep learning and image processing.

Yizhuo Meng received the M.S. degree and B.Eng. degree from Wuhan University, Wuhan, China, in 2020 and 2017. From 2017 to 2018 and 2018 to 2019, she was a Research Assistant with The Hong Kong Polytechnic University. Her research interests include image processing and analysis, image algorithm research.



Donyu Dorjee received the B.S. degree in remote sensing science and technology from the Chengdu University of Information Technology, Chengdu, China, in 2012.

From 2018 to 2019, he was a Visiting Scholar with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China. He is currently working on remote sensing applications with Climate Center of Tibet Autonomous Region, Lhasa, China.

Xiaobing Wei received the B.S. degree from the School of water Conservancy and Environment, China Zhengzhou University, Zhengzhou, China, in 2018. She is currently working toward the M.E. degree in photogrammeiry and remote sensing at the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China.

Her current research interests include multi-source remote sensing imagery classification and deep learning.

Zhiyuan Zhang received the Ph.D. degree from Wuhan University, Wuhan, China, in 2019.

He is currently an Engineer with the Information Center (Hydrology Monitor and Forecast Center), Ministry of Water Resources. His research interests include remote-sensing application in water conservancy and geographic information systems.

Wen Zhang received the Ph.D. degree from Wuhan University, Wuhan, China, in 2009.

She was a Visiting Scholar with the Joint Centre of Cambridge-Cranfield for High Performance Computing, Cranfield University, Cranfield, U.K., from 2007 to 2008. She is currently a Lecturer with the School of Remote Sensing and Information Engineering, Wuhan University. Her research interests include network GIS, remote-sensing applications, and spatial data analysis.