Research on the Extraction of Hazard Sources along High-Speed Railways from High-Resolution Remote Sensing Images Based on TE-ResUNet

Xuran Pan 1, Lina Yang 2, Xu Sun 3,*, Jingchuan Yao 4 and Jiliang Guo 4

Abstract: There are many potential hazard sources along high-speed railways that threaten the safety of railway operation. Traditional ground search methods are failing to meet the needs of safe and efficient investigation. In order to accurately and efficiently locate hazard sources along the high-speed railway, this paper proposes a texture-enhanced ResUNet (TE-ResUNet) model for railway hazard sources extraction from high-resolution remote sensing images. According to the characteristics of hazard sources in remote sensing images, TE-ResUNet adopts texture enhancement modules to enhance the texture details of low-level features, and thus improve the extraction accuracy of boundaries and small targets. In addition, a multi-scale Lovász loss function is proposed to deal with the class imbalance problem and force the texture enhancement modules to learn better parameters. The proposed method is compared with the existing methods, namely, FCN8s, PSPNet, DeepLabv3, and AEUNet. The experimental results on the GF-2 railway hazard source dataset show that the TE-ResUNet is superior in terms of overall accuracy, F1-score, and recall. This indicates that the proposed TE-ResUNet can achieve accurate and effective hazard sources extraction, while ensuring high recall for small-area targets.

Keywords: high resolution remote sensing image; hazard source extraction; semantic segmentation; texture enhancement; Lovász loss function

1. Introduction

Hazard sources, such as aging color-coated steel houses, poorly fixed dust-proof nets, and mulch films, along a high-speed railway are prone to be blown up by the wind and fall on the high-speed rails or grids, thereby hindering the normal operation of high-speed railway, or even causing major security incident. Notifying the relevant departments and units to strengthen the hazard sources through the ground investigation can effectively prevent accidents from occurring. For a long time, the ground investigation method has been the main means to detect the hazard sources around the railway. However, the traditional ground investigation requires a lot of manpower and material resources, and is limited by the environmental conditions, resulting in low efficiency and poor investigation results [1]. High-resolution remote sensing images have the advantages of large-scale, continuous time and space, periodicity, easy acquisition, and low cost. The spatial resolution of high-resolution remote sensing images has reached the submeter level, which can clearly present the hazard sources along railway, as shown in Figure 1. Therefore, high-resolution remote
sensing image earth object extraction technology can be applied to accurately extract and dynamically monitor information on hazard sources along the railway line.

![Figure 1](image1.png)

**Figure 1.** Hazard source along high-speed railways in high resolution remote sensing image: (a) color-coated steel house; (b) dust-proof net.

High-resolution remote sensing image earth object extraction aims to assign a certain earth object or background label to each pixel of the remote sensing images [2], mainly using semantic segmentation technology [3–8]. In recent years, deep learning technology has developed rapidly. Compared with traditional machine learning algorithms, deep learning can extract high-level semantic features hidden in images, and has better performance in complex scenes. Therefore, deep learning based semantic segmentation methods open up a new way for remote sensing image earth object extraction [9–17].

The earliest semantic segmentation method used is the patch-based method. Saito et al. [18] used patch-based convolutional neural network (CNN) to extract roads and buildings from high-resolution images, and achieved good results on Massachusetts roads and buildings datasets [19]. Alshehhi et al. [20] proposed a single patch-based CNN architecture for the extraction of roads and buildings from high-resolution remote sensing data, and low-level features were integrated with CNN features in the post-processing stage to improve the performance. However, these patch-based methods suffer from limited receptive fields and large computational overheads, so it was soon surpassed by pixel-based methods. The pixel-based methods mostly follow an encoding–decoding paradigm. Maggiori et al. [21] proposed a baseline architecture for building extraction based on fully convolutional network (FCN) [3] combined with multi-layer perceptron, and provided a building extraction dataset which was named the Inria aerial image labeling dataset. Bischke et al. [22] used a cascaded multi-task loss to optimize the SegNet [8] architecture to take advantage of the boundary information of segmentation masks, and the performance can achieve certain improvements without any changes of the network architecture. In [23], Khalel et al. proposed a stack of U-Nets to automatically label the buildings from high aerial images, of which each U-Net can be regarded as the post-processor of the previous U-Net to improve the accuracy [4]. Chen et al. [24] proposed a CNN architecture to produce fine segmentation results of high-resolution remote sensing images. The architecture introduced the cascaded relation attention module to determine the relationship among different channels or positions, and adopted information connection and error correction capture, and fused the features of geo-object details. Diakogiannis et al. [25] proposed a reliable framework for semantic segmentation of remotely sensed data consisting of a ResUNet-based architecture and a novel loss function based on the Dice loss. Lan et al. [26] proposed a novel global context based dilated convolutional neural network for road extraction. In order to cope with the complex background, the architecture introduced residual dilated blocks to enlarge the receptive field, and performed pyramid pooling module to capture the multiscale global context features. Gao et al. [27] proposed an object-oriented deep learning framework to effectively extract both low-level and high-level features with limited training data. The framework leveraged residual networks with different depths
to learn adjacent feature representations by embedding a multibranch architecture in the deep learning pipeline. Although the above methods have achieved the improvement of the extraction accuracy within a certain range, there are still issues with low precision of ground objects boundaries and small objects.

The distribution of hazard sources along the high-speed railway on the high-resolution remote sensing images is sparse, with many small targets. Meanwhile, the intra-class variation of the same hazardous sources is relatively large. This brings certain technical difficulties to the hazard sources extraction from high-resolution remote sensing images, and puts forward higher requirements for the accuracy and stability of the extraction model. In this paper, we propose a texture-enhanced ResUNet (TE-ResUNet) architecture for high-resolution remote sensing image railway hazard source extraction. The network follows UNet architecture, and introduces texture enhancement module to enhance the texture details of low-level features to improve the hazard source extraction accuracy, especially the boundary extraction accuracy. Moreover, the multi-scale Lovász loss function is proposed to improve the supervision of the texture enhancement modules, and further improve the stability of the model.

2. Materials and Methods

2.1. Data Labeling

Our sample data come from the multi-spectral remote sensing images of a GF2 satellite, acquired over Beijing, China. We used the LabelMe annotation tool to manually label hazard sources along the high-speed railway and to generate binary label images. The categories of hazard sources in this paper include color-coated steel houses and dust-proof nets.

Color-coated steel houses are typical urban temporary buildings, with color-coated steel sheets as the raw material. The colors are mainly blue, white, and red. The color-coated steel houses in high resolution remote sensing images usually have high brightness, and the shape is mostly rectangular or rectangular combination. Dust-proof nets are mainly used for dust control and environmental protection. Exposed surfaces, dust-prone materials, etc., in cities are usually required to be covered by dust-proof nets. Dust-proof nets are generally made of polyethylene material in a light green color. Here, we define hazard source pixels as positive samples and other pixels as negative samples, and generate binary label images, as shown in Figure 2.

Figure 2. Training samples of a hazard source dataset: (a) color-coated steel houses; (b) dust-proof nets.
The original images in the hazard source dataset are the fusion images of the panchromatic and multi-spectral bands at a spatial resolution of 0.8 m, using the red, green, and blue bands. A total of 153 image patches of \(1000 \times 1000\) pixels containing color-coated steel houses and 139 image patches containing dust-proof nets were selected for the dataset. Divided by the proportion 7:2:1 for training set, validation set, and testing set, we finally obtained 153, 43, and 21 pieces of color-coated steel house samples and 139, 39, and 19 pieces of dust-proof net samples. Data augmentation was performed on the training samples, after rotation by \(90^\circ\), \(180^\circ\), and \(270^\circ\), horizontal and vertical inversion, and random cropping, and 3672 and 3336 image patches of \(512 \times 512\) pixels were obtained, respectively, and the specific information is shown in Table 1.

### Table 1. Training samples of hazard source dataset.

| Hazard Source       | Number of Training Samples | Original Images (1000 \(\times\) 1000) | Data Augmentation (512 \(\times\) 512) |
|---------------------|----------------------------|----------------------------------------|----------------------------------------|
| Color-coated steel house | 153                       | 3672                                    |                                         |
| Dust-proof net      | 139                       | 3336                                    |                                         |

#### 2.2. Architecture of TE-ResUNet

The proposed TE-ResUNet is a segmentation network with an encoder–decoder structure based on the UNet family model-ResUNet. ResUNet is inspired by residual network, replacing each sub-module of UNet with residual connected network, which allows the network to be deepened to obtain higher-level semantic features and with a reduced risk of gradient disappearance. Typically, the UNet family models skip connect the low-level features directly to high-level features to maintain accurate texture features. However, the low-level features extracted by the shallow layer of the network are often of low quality, especially in terms of contrast, resulting in blurred texture details that negatively affect the extraction and utilization of low-level information. This, in turn, affects the extraction accuracy of the segmentation boundary. In order to make full use of the low-level texture features, a texture-enhanced ResUNet (TE-ResUnet) is proposed for fast-speed railway hazard source extraction from high-resolution remote sensing images.

The architecture of TE-ResUNet is shown in Figure 3, with a contracted path on the left side performs residual connected convolutions to produce feature maps, and an extended path on the right side to recover detailed and structural information related to hazard sources via convolution and upsampling modules. The whole network presents a U-shape. In particular, the texture enhancement modules are introduced to enhance the texture of the low-level features to assist the expansion path to better recover the hazard boundary, as well as the small targets.

The orange module D represents residual connected down-sampling module, including convolution, atrous convolution, and ReLU layers. The blue module U represents the upsampling module, including the convolution and upsampling layer, and the yellow module C represents the convolution, activation function, and batch normalization layers. The dark gray arrow represents skip connected operation. The low-level features at each scale are skip-connected to the higher layers, allowing the model to make full use of the low-level features. The low-level features extracted from the shallow layers are often of low quality, especially with low contrast, which leads to blurred texture details, and brings negative impacts on extraction and utilization of low-level information. Therefore, this paper introduces the texture enhancement module (purple module TE) to perform texture enhancement on the low-level features before they are delivered to the deeper layers by skip connections.

The texture enhancement (TE) module is inspired by histogram equalization, a classical method of image quality enhancement [28]. The TE module aims to convert histogram equalization into a learnable manner. It first encodes the feature to a learnable histogram, during which the input features are quantized into multiple level intensities, and each level
can represent a kind of texture statistics. Then, a graph is built to reconstruct each gray level by propagating statistical information of all original levels for texture enhancement.

![Figure 3. Architecture of TE-ResUNet.](image)

Specifically, the texture enhancement module starts by generating a histogram with the horizontal and vertical axes representing each gray level and its count value, respectively. These two axes are represented as two feature vectors, \( G \) and \( F \). Histogram equalization aims to reconstruct the levels as \( G' \) using the statistical information contained in \( F \). Each level \( G'_n \) is converted to \( G''_n \) by Equation (1):

\[
G''_n = \frac{(N - 1) \sum_{i=0}^{n} F_n}{\sum_{n=0}^{N} F_n}
\]

where \( N \) represents the number of gray levels.

The structure of the texture enhancement module is shown in Figure 4. The input feature of TE module is of high channel dimensionality. In order to quantize and count the high dimensional representation, we compute the similarity between each vector and average feature as the counted object instead of quantizing each channel separately. The input feature map \( A_C \times H \times W \) is converted into the global averaged feature \( g_C \times 1 \) by global average pooling. Each spatial position \( A_{ij} \) of \( A \) has dimension \( C \times 1 \), and the cosine similarity of each \( A_{ij} \) and the mean \( g \) is then calculated to obtain the similarity map \( S_1 \times H \times W \), where \( S_{ij} \) can be calculated by Equation (2).

\[
S_{ij} = \frac{g \cdot A_{ij}}{\|g\|_2 \|A_{ij}\|_2}
\]

\( S \) is then reshaped to \( 1 \times HW \) and quantized with \( N \) gray levels \( L \). \( L \) is obtained by equally dividing \( N \) points between the minimum and maximum values of \( S \). The quantization encoding map \( E_N \times HW \) is thus obtained. The histogram \( C_N \times 2 \) is represented by concatenating \( E \) and \( L \), where \( L \) is the horizontal coordinate indicating the number of quantized gray levels, and \( E \) is the vertical coordinate indicating the weight corresponding to each level. The quantization counting map \( C \) reflects the relative statistics of input feature map. To further obtain absolute statistical information, the global average feature \( g \) is encoded into \( C \) to obtain \( D \).
The specific model parameters of TE-ResUNet are shown in Table 2.

Afterwards, according to the histogram quantization method, a new quantization level $L'$ needs to be computed from $D$. Each new level should be obtained by perceiving the statistical information from all the original levels, and can be treated as a graph. For this purpose, a graph is constructed to propagate the information from all levels. The statistical characteristics of each quantified level are defined as a node. In a traditional histogram quantization algorithm, the adjacency matrix is a manually defined diagonal matrix, which is extended to a learning matrix as follows:

$$X = Soft\ max(\phi_1(D)^T \cdot \phi_2(D))$$

(3)

where $\phi_1$ and $\phi_2$ represent two different $1 \times 1$ convolutional layers, and after performing Softmax operation in the first dimension as a nonlinear normalization function, each node is then updated to obtain the reconstructed quantization level $L'_C \times N$ by fusing the features of all other nodes.

$$L' = \phi_3(D) \cdot X$$

(4)

where $\phi_3$ represents another $1 \times 1$ convolutional layer.

Subsequently, the reconstruction level $L'$ is assigned to each pixel using the quantization encoding mapping $E_{N \times HW}$ to obtain the final output $R$, since $E$ reflects the original quantization level of each pixel. $R$ is obtained by:

$$R = L' \cdot E$$

(5)

$R$ is then reconstructed into $R_{C \times H \times W}$, which is the final texture-enhanced feature map.

Texture enhancement of features $F_1$ and $F_2$ is performed by TE1 and TE2, which are then connected to the high-level features by depth to make full use of the low-level texture details to assist the network in generating more accurate hazard source extraction results. The specific model parameters of TE-ResUNet are shown in Table 2.

2.3. Multi-Scale Lovász Loss Function

The hazard source dataset suffers from the class-imbalance, mainly due to the sparsity of the hazard source pixel distribution. This makes the pixel-based loss function, such as cross entropy loss, focus training on background pixels that contribute less valid information, resulting in low training efficiency and model degradation. Lovász loss [29] is a metric-based measurement that focuses more on the metrics of the entire image instead of a single pixel. This means that there is no need to consider the problem of balanced sample distribution, and it works better for binary classification problems where the proportion of foreground samples is much smaller than the background.
Table 2. The architecture of TE-ResUNet.

| Layer | Operation | Filters’ Size | Filters’ Number | Stride | Layer | Operation | Filters’ Size | Filters’ Number | Stride |
|-------|-----------|---------------|-----------------|--------|-------|-----------|---------------|-----------------|--------|
| C1    | Conv1     | 3             | 64              | 1      | D4    | Conv3     | 1             | 2048           | ×3     |
|       | Pool1     | 2             | 64              | 1      |       | Pool1     | 2             | 192            | 1      |
|       | Conv1     | 1             | 64              | 1      |       | Conv1     | 3             | 128            | 1      |
|       | Conv2     | 3             | 64              | 1      |       | DConv     | 4             | 96             | 2      |
| D1    | Conv1     | 1             | 256             | 1      |       | Conv1     | 3             | 128            | 1      |
|       | Conv2     | 3             | 128             | 2,1    |       | DConv     | 4             | 96             | 2      |
|       | Conv3     | 1             | 512             | 1      |       | Conv1     | 3             | 64             | 1      |
| D2    | Conv1     | 1             | 256             | 2,1    |       | DConv     | 4             | 48             | 2      |
|       | Conv2     | 3             | 256             | 1      |       | Conv1     | 3             | 48             | 1      |
|       | Conv3     | 1             | 1024            | 2,1    |       | DConv     | 4             | 32             | 2      |
| D3    | Conv1     | 1             | 512             | 2      |       | Conv1     | 3             | 32             | 1      |
|       | Conv2     | 3             | 512             | 1      |       | Conv2     | 1             | 2              | 1      |
| D4    | Conv1     | 1             | 512             | 2      |       |           |               |                 |        |
|       | Conv2     | 3             | 512             | 1      |       |           |               |                 |        |

Based on the above analysis, we propose to address the class-imbalance problem by applying the Lovász loss. The Lovász loss function is a smoothed expansion of the Jaccard loss for the Jaccard index. The expression for Jaccard loss is given in Equation (6):

$$\Delta_{J_{c}}(\tilde{y}*_{c}, \tilde{y}) = 1 - \frac{|\{y^*_{c} \cap \tilde{y} = c\}|}{|\{y^*_{c} \cup \tilde{y} = c\}|}$$  \hspace{1cm} (6)

where $y^*$ represents the predicted result of the network model, and $\tilde{y}$ represents the ground truth. As the Jaccard loss function is only applicable to the discrete case, a Lovász expansion of it can transform the input space from discrete to continuous, and the output value is equal to the output of the original function on the discrete domain. In this paper, the Lovász loss function is denoted as $\Delta_{L_{c}}$.

In order to make the network optimization go toward the accurate direction of loss decline, and to enhance the supervision of parameter learning of the two texture enhancement modules, we performed loss function calculations based on three scales of feature maps. In addition to calculating the loss of the final prediction results and the labels, we also performed channel dimensionality reduction on F8 and calculated Lovász loss with the downsampled labels and perform channel dimensionality reduction on U4 and calculate Lovász loss with the downsampled labels. The final loss is the weighted sum of the three components, as follows.

$$L_{Total} = \lambda_{1} \Delta_{L_{c}}(y^*, \tilde{y}) + \lambda_{2} \Delta_{L_{c}}(y^{D2*}, \tilde{y}_{U4}) + \lambda_{3} \Delta_{L_{c}}(y^{D4*}, \tilde{y}_{F8}),$$  \hspace{1cm} (7)

where $\tilde{y}_{U4}$ represents the result of dimensionality reduction on the output of U4, $y^{D2*}$ and $y^{D4*}$ denotes the result of downsampling the original labels by a factor of 2 and 4, $\tilde{y}_{F8}$ denotes the result of channel dimensionality reduction on F8, and $\lambda_{1}$, $\lambda_{2}$ and $\lambda_{3}$ are the weights of the three scale loss functions.
2.4. Metric

For a binary-segmentation assignment, the prediction results are divided into four sets, i.e., true positive (TP), false negative (FN), false positive (FP), and true negative (TN). TP means inferring a positive sample as positive correctly, FN means inferring a positive sample as negative wrongly, FP means inferring a negative sample as positive wrongly, and TN means inferring a negative sample as negative correctly. For evaluating the performances of different networks, we choose three metrics to assess the hazard source extraction result, which are overall accuracy (OA), $F_1$-score, and recall, as defined in Equations (8)–(11). Here, OA is the proportion of correctly inferred samples to all samples; precision is the proportion of correctly inferred positive samples to all inferred positive samples; recall is the proportion of correctly inferred positive samples to all actual positive samples; $F_1$-score is the harmonic average of precision and recall.

\[
OA = \frac{TP + TN}{TP + FN + TN + FP} \tag{8}
\]

\[
F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2TP}{2TP + FN + FP} \tag{9}
\]

\[
\text{precision} = \frac{TP}{TP + FP} \tag{10}
\]

\[
\text{recall} = \frac{TP}{TP + FN} \tag{11}
\]

3. Experiments
3.1. Implementation Details

The implementation is based on Pytorch 1.8.0, and the training machine is a server equipped with Intel(R) Xeon(R) Gold 5218 CPU, GeForce RTX 2080 Ti GPU. We use SGD with a total of four images per minibatch. All models are trained for 29,000 iterations with an initial learning rate of 0.001. Weight decay of 0.0001 and momentum of 0.9 are used. The three weights of the loss function, $\lambda_1$, $\lambda_2$, and $\lambda_3$, were set to 0.7, 0.2, and 0.1, respectively.

3.2. Ablation Study
3.2.1. Tradeoff between Training Data Size and Encoder Complexity

In order to investigate the parameter scale of the encoder, Vgg16, ResNet50 and ResNet101 acted as the encoder of the UNet architecture, and trained and validated on the color-coated steel house dataset, respectively. The experimental results on the validation dataset are shown in Table 3. The ResNet50 and ResNet101 obtained higher overall accuracy (OA), $F_1$-score and recall compared to Vgg16. ResNet50 had slightly lower OA than ResNet101, but the more important metrics, $F_1$-score and recall, were higher than ResNet101. It can be concluded that the model depth of ResNet50 is more appropriate for the scale of the hazard source dataset. Vgg16 has fewer layers and cannot extract enough abstract high-level semantic features, while the ResNet101 model is too deep for this dataset and is prone to overfitting. Therefore, the encoder of TE-ResUNet proposed in this work is constructed based on ResNet50.

Table 3. Network performance with different encoders.

| Encoder   | OA (%) | $F_1$-Score (%) | Recall (%) |
|-----------|--------|----------------|------------|
| Vgg16     | 95.32  | 60.22          | 68.30      |
| ResNet50  | 95.79  | 64.85          | 72.97      |
| ResNet101 | 95.98  | 63.78          | 70.25      |
3.2.2. The Effectiveness of Texture Enhancement Module

In order to verify whether the texture enhancement module can bring gains to the performance of hazard source extraction, comparative experiments of ResUNet, TE-ResUNet with TE1 module removed (TE-ResNet-A), TE-ResUNet with TE2 module removed (TE-ResNet-B), and TE-ResUNet were conducted on the validation set of color-coated steel house dataset. As shown in Table 4, the performance of the models with the addition of the texture enhancement module are improved, compared to the original ResUNet, which indicates that the texture enhancement module can bring benefits to hazard source extraction. In terms of the texture enhancement module settings, the texture enhancement for the shallower layers (TE1) is better than the texture enhancement for the deeper layers (TE2), and the two texture enhancement modules work together obtained the optimal performance.

Table 4. Network performance with texture enhancement module.

| Models        | TE1 | TE2 | OA (%) | F1-Score (%) | Recall (%) |
|---------------|-----|-----|--------|--------------|------------|
| ResUNet       | ×   | ×   | 95.79  | 64.85        | 72.97      |
| TE-ResUNet-A  | ×   | √   | 96.22  | 66.13        | 73.12      |
| TE-ResUNet-B  | √   | ×   | 96.83  | 71.13        | 77.38      |
| TE-ResUNet    | √   | √   | 97.01  | 75.89        | 77.43      |

Figure 5 shows the extraction results of each method on the validation set of color-coated steel house dataset. The extraction results of the original ResUNet (Figure 5c) show that the boundaries of the color-coated steel houses are blurred, and there are obvious omissions (see red box). The experimental results of TE-ResUNet-A (Figure 5d) and TE-ResUNet-B (Figure 5e), with the help of the texture enhancement module, have better accuracy of the boundary extraction, but still have obvious omissions. The visual effect of TE-ResUNet-B is slightly better than TE-ResUNet-A, indicating that the texture enhancement is more effective for lower-level features. As shown in Figure 5f, with the help of texture enhancement modules of two scales, TE-ResUNet has greatly improved the boundary extraction results of the color-coated steel house, and the omissions has been significantly improved.
3.2.3. Network Performance with Different Loss Functions

For our dataset, the number of positive samples is significantly smaller than the negative samples. Moreover, many hazard sources are very small or with complicated and blurred boundaries, which are difficult for a network to identify. Therefore, we tested several specific loss functions to address such class imbalance problem, including the cross-entropy loss, the focal loss, the dice loss, the Lovász loss, and the proposed multi-scale Lovász loss.

As different loss functions have different rules for calculating loss values, the loss curves are not comparable, therefore the mean IoU curves of the training data are shown in Figure 6. As can be seen that the mean IoU converges to around 75% on the training data when the network is trained using pixel-based loss functions, the cross-entropy loss, and the focal loss. The curve of the cross-entropy loss is smoother and convergences to a higher mean IoU score than the focal loss. This is because the focal loss mainly solves the imbalance of the number of hard and easy samples, and has a limited effect on the imbalance of the number of positive and negative samples in this work. In contrast, the metric-based loss function, Lovász loss, and multi-scale Lovász loss outperform the pixel-based loss function, with the mean IoU converging at around 78% on the training data, demonstrate that metric-based loss function can better overcome the problem of positive and negative sample imbalance. Further, training the network with the proposed multi-scale Lovász loss results in faster convergence and a higher mean IoU score compared to the Lovász loss. This is mainly due to the multi-scale Lovász loss allows better supervision of the texture enhancement modules, which enables the network to better recover texture details and improve the accuracy of hazard source boundaries and small targets. Table 5 shows the network performance on the validation set of color-coated steel house dataset with different loss functions, further proving the superiority of the proposed loss function.

![Figure 6. Mean IoU curves on 29 k iterations with different loss function.](image)

Table 5. Network performance with different loss functions.

| Loss Function             | OA (%) | F1-Score (%) | Recall (%) |
|---------------------------|--------|--------------|------------|
| Cross-entropy Loss        | 95.91  | 72.44        | 75.68      |
| Focal Loss                | 95.32  | 70.32        | 75.00      |
| Lovász Loss               | 97.01  | 75.89        | 77.43      |
| Multi-Scale Lovász Loss   | 96.93  | 76.21        | 77.76      |

It is worth mentioning that we also conducted experiments on dice loss, which also belongs to metric-based loss, but it cannot converge on the training data. Therefore, the experimental results are not shown in Figure 6 and Table 5. The main reason may be that dice loss is more suitable for extremely imbalance samples. Under normal circumstances, using dice loss will adversely affect backpropagation, and make training unstable.
3.3. Comparing Network Performance with Other Method

To further validate the effectiveness of the proposed method, the experimental results on two testing sets of TE-ResUNet are compared with existing methods in this section, including FCN8s [3], PSPNet [7], DeepLabv3 [8], and AEUNet [16]. Among these methods, FCN8s is a base semantic segmentation network with 8x upsampled prediction. PSPNet extends pixel-level features to global pyramid pooling to improve the segmentation result. DeepLabv3 introduces a decoder, which can achieve accurate semantic segmentation and reduce the computational complexity. AEUNet is a spatial-channel attention-enhanced building extraction model, which introduces Resnet and attention models into UNet architecture, and proposes a multi-scale fusion module to retain the local detail characteristics. To ensure the fairness of the comparison, ResNet50 is adopted as the backbone networks of PSPNet, DeepLabv3, and AEUNet.

Table 6 shows the comparison results of each method on the color-coated steel house testing set. It can be seen that TE-ResUNet outperforms other comparison methods on all evaluation metrics. There are many small targets in the color-coated steel house dataset, so most methods have relatively serious omissions of small targets, and therefore a lower recall. While TE-ResUNet can guarantee a relatively high accuracy for small targets, and recall is improved by more than 5% compared to other methods. FCN8s has a relatively poor performance in color-coated steel house extraction due to its shallow network layers and relatively simple decoder design. The pyramid pooling module in PSPNet further downsamples the feature map on the basis of 32 times the downsampling of Resnet50, which affects the extraction accuracy of small color-coated steel houses, therefore the $F_1$-score and recall are relatively low. In addition, PSPNet is difficult to train with a small dataset. Due to the atrous convolution structure of DeepLabv3, the downsampling loss of small targets is small, therefore the recall and other evaluation indicators are relatively high. AEUNet is a promising architecture with spatial and channel wise attention mechanisms to improve the representation of targets, but it requires a relatively large amount of training data to motivate its performance. The hazard source dataset is relatively small, therefore AEUNet achieves better performance in OA but lower in $F_1$-score and recall when compared to the TE-ResUNet. The proposed TE-ResUNet proposed makes full use of multi-scale features through skip connections, and performs texture enhancement on low-level features to make full use of texture detail features. The texture enhancement module is lightweight and only brings very little extra cost, which can better overcome the overfitting problem caused by the insufficient data amount. Therefore, TE-ResUNet obtains more accurate boundaries and improves the recall of small objects, as well as other evaluation metrics.

### Table 6. Network performance on color-coated steel house testing set.

| Models    | OA (%) | $F_1$-Score (%) | Recall (%) |
|-----------|--------|-----------------|------------|
| FCN8s     | 95.95  | 65.62           | 68.15      |
| PSPNet    | 97.08  | 70.32           | 60.30      |
| DeepLabv3 | 96.76  | 72.06           | 73.68      |
| AEUNet    | 96.51  | 74.93           | 76.58      |
| TE-ResUNet| 97.49  | 77.55           | 78.64      |

Figures 7 and 8 show the extraction results of each method on the testing set of color-coated steel house dataset. The color-coated steel houses in the testing set are mostly adjacent to or scattered in residential areas. The spectral characteristics of the white color-coated steel houses are similar to those of the white buildings and bare soil, which are easily confused by the model. Meanwhile, there are many small targets in this dataset, which are prone to be mis-extracted by models. Figures 7c and 8c are the extraction results of FCN8s. It can be seen that the extraction results of color-coated steel houses are relatively coarse. As it does not make full use of the low-level features, the boundary is blurred, and the omissions of small targets is high. Figures 7d and 8d show the extraction results of PSPNet, the accuracy of the boundary is improved compared with FCN8s, but there is
also the issue of omissions. Figures 7e and 8e show the extraction results of DeepLabv3, the accuracy of the boundary has been greatly improved, but the problem of omissions of small targets is still serious. Figures 7f and 8f show the extraction results of AEUNet, the extraction results of small targets is notably better than FCN8s, PSPNet and Deeplabv3, but the mis-extraction still exists. Figures 7g and 8g show the extraction results of the proposed method, the accuracy of the boundary and the detection rate of small targets have been greatly improved (in the red box), especially for the small color-coated steel houses scattered in residential areas.

![Figure 7. The example results of color-coated steel house testing set: (a) original images; (b) ground truths; (c) FCN8s; (d) PSPNet; (e) DeepLabv3; (f) AEUNet; (g) TE-ResUNets.](image1)

![Figure 8. The example results of color-coated steel house testing set: (a) original images; (b) ground truths; (c) FCN8s; (d) PSPNet; (e) DeepLabv3; (f) AEUNet; (g) TE-ResUNets.](image2)
On the dust-proof net dataset, FCN8s, PSPNet, DeepLabv3, and AEUNet are also performed to compare with the proposed TE-ResUNet. Table 7 displays the experimental results of each method on the testing set of dust-proof net dataset. It can be seen that TE-ResUNet obtained the best scores for all metrics compared with other methods, which further proves the rationality of the network structure design of the proposed network. Compared to the color-coated steel houses, the dust-proof nets are generally larger in size and fewer in number. Therefore, TE-ResUNet performs well in all metrics, all of which are above 84%.

Table 7. Network performance on dust-proof net testing set.

| Models          | OA (%) | F1-Score (%) | Recall (%) |
|-----------------|--------|--------------|------------|
| FCN8s           | 95.04  | 83.26        | 83.64      |
| PSPNet          | 95.30  | 82.57        | 71.97      |
| DeepLabv3       | 95.89  | 86.44        | 80.69      |
| AEUNet          | 97.60  | 83.35        | 82.31      |
| TE-ResUNet      | 96.62  | 88.10        | 84.52      |

Figures 9 and 10 show the extraction results of the dust-proof net testing set. The example images contain dust-proof nets with a long coverage time, which is prone to be confused with bare soil due to fading. Figures 9c and 10c show the extraction results of FCN8s—the boundary of dust-proof nets is also blurred, but the detection rate is good; Figures 9d and 10d show the extraction results of PSPNet—the boundary accuracy has been improved, but the omissions are relatively high (in the green box); Figures 9e and 10e show the extraction results of DeepLabv3—the boundary accuracy and omissions are improved, but omissions still occur; Figures 9f and 10f show the extraction results of AEUNet—the results of dust-proof nets are more complete and fine; Figures 9g and 10g show the extraction results of the proposed method—the boundary accuracy is more satisfactory and most of the dust-proof nets are extracted completely.

![Figure 9. The example results of dust-proof net testing set: (a) original images; (b) ground truths; (c) FCN8s; (d) PSPNet; (e) DeepLabv3; (f) AEUNet; (g) TE-ResUNets.](image)
4. Conclusions

In this paper, we propose a TE-ResUNet architecture for fast-speed railway hazard source extraction from high-resolution remote sensing imagery. The TE-ResUNet is based on the UNet framework, and adopts residual connected network to deepen the model in the encoding stage. In the decoding stage, in order to make full use of the low-level texture features, we propose to introduce texture enhancement modules to enhance the texture features of the shallow layers to improve the boundary accuracy of hazard source extraction. In addition, a multi-scale Lovász loss function is proposed for model optimization, which can better cope with the sample imbalance problem, as well as perform better supervision on the texture enhancement module. The testing results on the GF-2 railway hazard extraction dataset show that the performance of the TE-ResUNet exceeds several baseline methods across all evaluation metrics.

Author Contributions: Conceptualization, X.P., L.Y. and X.S.; software X.P. and L.Y.; validation, X.P.; investigation, J.Y. and J.G.; writing—original draft preparation, X.P.; writing—review and editing, X.S. and L.Y.; supervision, X.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Scientific Research and Development Program of China Railway (K2019G008), and the China Academy of Railway Sciences Foundation (2019YJ028).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: The authors would like to thank the anonymous reviewers for their constructive comments and recommendations.

Conflicts of Interest: The authors declare no conflict of interest.
References

1. Gao, S. Research on the Technical System of Railway Remote Sensing Geological Exploration. Railw. Stand. Des. 2014, 58, 18–21. [CrossRef]

2. Yang, H.L.; Yuan, J.; Lunga, D.; Laverdiere, M.; Rose, A.; Bhaduri, B. Building extraction at scale using convolutional neural network: Mapping of the United States. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2018, 11, 2600–2614. [CrossRef]

3. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.

4. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Cham, Switzerland, 2015; pp. 234–241.

5. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. Comput. Sci. 2014, 4, 357–361.

6. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 2016, 40, 834–848. [CrossRef] [PubMed]

7. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. arXiv 2017, arXiv:1706.05587.

8. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; IEEE: Los Alamitos, CA, USA, 2017; pp. 6230–6239.

9. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 13–16 December 2015; pp. 1520–1528.

10. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 2017, 39, 2481–2495. [CrossRef]

11. Liu, Y.; Piramanayagam, S.; Monteiro, S.; Saber, E. Dense semantic labeling of very-high-resolution aerial imagery and LiDAR with fully-convolutional neural networks and higher-order crfs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 1561–1570.

12. Pan, X.; Gao, L.; Marinoni, A.; Zhang, B.; Yang, F.; Gamba, P. Semantic labeling of high resolution aerial imagery and LiDAR data with fine segmentation network. Remote Sens. 2018, 10, 743. [CrossRef]

13. Girard, N.; Charpiat, G.; Tarabalka, Y. Aligning and updating cadaster maps with aerial images by multi-task, multi-resolution deep learning. In Proceedings of the Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018; pp. 675–690.

14. Griffiths, D.; Boehm, J. Improving public data for building segmentation from Convolutional Neural Networks (CNNs) for fused airborne lidar and image data using active contours. ISPRS J. Photogramm. Remote Sens. 2019, 154, 70–83. [CrossRef]

15. Vargas-Muñoz, J.E.; Lobry, S.; Falcão, A.X.; Tuia, D. Correcting rural building annotations in OpenStreetMap using convolutional neural networks. ISPRS J. Photogramm. Remote Sens. 2019, 147, 283–293. [CrossRef]

16. Li, C.; Fu, L.; Zhu, Q.; Zhu, J.; Fang, Z.; Xie, Y.; Guo, Y.; Gong, Y. Attention Enhanced U-Net for Building Extraction from Farmland Based on Google and WorldView-2 Remote Sensing Images. Remote Sens. 2021, 13, 4411. [CrossRef]

17. Yang, N.; Tang, H. GeoBoost: An incremental deep learning approach toward global mapping of buildings from VHR remote sensing images. Remote Sens. 2020, 12, 1794. [CrossRef]

18. Saito, S.; Yamashita, T.; Aoki, Y. Multiple Object Extraction from Aerial Imagery with Convolutional Neural Networks. Electronic Imag. 2016, 60, 10402.

19. Mnih, V. Machine Learning for Aerial Image Labeling. Ph.D. Thesis, University of Toronto, Toronto, ONT, Canada, 2013.

20. Alshehhi, R.; Marpu, P.R.; Wei, L.W.; Mura, M.D. Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks. ISPRS J. Photogramm. Remote Sens. 2017, 130, 139–149. [CrossRef]

21. Magnieri, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Can semantic labeling methods generalize to any city? The Inria aerial image labeling benchmark. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3226–3229.

22. Bischke, B.; Helber, P.; Kolb, J.; Borth, D.; Dengel, A. Multi-task learning for segmentation of building footprints with deep neural networks. arXiv 2017, arXiv:1709.05932.

23. Khalel, A.; El-Saban, M. Automatic pixelwise object labeling for aerial imagery using stacked u-nets. arXiv 2018, arXiv:1803.04953.

24. Chen, J.; Wang, H.; Guo, Y.; Sun, G.; Deng, M. Strengthen the feature distinguishability of geo-object details in the semantic segmentation of high-resolution remote sensing images. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2021, 14, 2327–2340. [CrossRef]

25. Diakogiannis, F.I.; Waldner, F.; Caccetta, P.; Wu, C. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. ISPRS J. Photogramm. Remote Sens. 2020, 162, 94–114. [CrossRef]

26. Lan, M.; Zhang, Y.; Zhang, L.; Du, B. Global context based road segmentation via dilated convolutional neural network. Inform. Sci. 2020, 535, 156–171. [CrossRef]

27. Gao, H.; Guo, J.; Guo, P.; Che, X. Classification of very-highspatial-resolution aerial images based on multiscale features with limited semantic information. Remote Sens. 2021, 13, 364. [CrossRef]
28. Zhu, L.; Ji, D.; Zhu, S.; Gan, W.; Wu, W.; Yan, J. Learning Statistical Texture for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 19–25 June 2021.

29. Berman, M.; Triki, A.R.; Blaschko, M.B. The Lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4413–4421.