Building Extraction From Remote Sensing Images With DoG as Prior Constraint

Yujun Quan, Anzhu Yu, Xuefeng Cao, Chunping Qiu, Xiaoyi Zhang, Bing Liu, and Peipei He

Abstract—Building extraction from aerial and satellite remote sensing images is a basic component of social development. Compared to traditional feature extraction strategies, deep convolutional neural networks (CNNs) have the advantage of extracting deep high-level semantic features and efficient image processing capabilities. However, most works focus on developing end-to-end data-driven models without considering prior information, such as edges and the structures of buildings, which would cause loss of details and blurred boundaries in the prediction results. To alleviate this problem, we constructed a prior information module (PIM) as a constraint for feature map refinement in the training phase that uses the edge information extracted by the multiscale difference of Gaussian operator. We combined this module with the main network to form a dual-output training network. The module helps optimize the feature extraction process and solves the inaccurate and adhesion phenomenon of edge points in the building extraction results of the existing CNNs. Experiments illustrate that this PIM can improve the intersection over union of a lightweight network by 2.4% and 2.0% on an aerial remote sensing dataset and a satellite remote sensing dataset, respectively. The PIM can be embedded in other networks during the training phase to improve building extraction performance without any extra computational requirements during the inference phase. At the same time, we build a U-structure network with the proposed module, which outperforms another state-of-the-art building extraction approach with the extracted edge as prior information.

Index Terms—Building extraction, deep learning, difference of Gaussian (DoG) operator, semantic segmentation.

I. INTRODUCTION

Automatic extraction based on remote sensing images is an essential and basic task in the application fields of urban planning, land-use statistics, and city development evaluation [1]. In recent years, with the development of sensor and computer technologies, massive high-resolution remote sensing images with rich spatial details and semantic information have provided opportunities for high-precision and automatic building extraction [2], [3]. However, it remains a challenging task due to the large intra-class variance and the small inter-class variance when the foreground and background are complex [4]. The main categories of traditional building extraction methods are as follows:

1) Multiscale building segmentation first separates buildings from the background and then uses the buildings’ shape features [5], spectral features [6], and texture features [7] to extract buildings.

2) The building extraction methods based on edge detection [8], [9] use the difference in objects’ gray levels to extract boundary features and achieve segmentation. The building extraction methods based on corner matching [10] identify and extract corners with distinct features and, then, match buildings feature points to extract target points according to feature matching rules.

3) The building extraction methods based on regional segmentation [11] include the regional growth segmentation method, quadtree segmentation method [12], and watershed segmentation method [13].

4) Some building extraction methods use mathematical methods, such as moment invariant feature extraction [14], the gray-level co-occurrence matrix [15], histogram of oriented gradients [16], Haar space [17], etc.

5) Since the rise of machine learning, methods based on support vector machines [18] kernel function support vector machines [19], the k-means clustering method [20], and the conditional random field [21] have often been used for building extraction.

However, both the traditional building extraction methods and the machine learning methods are heavily dependent on prior knowledge and parameter initialization and consume significant time and human resources in practical applications [2]. In addition, hand-crafted features can only solve specific problems in specific scenarios, and their accuracy is not enough for applicable purpose [9].

In recent years, deep convolutional neural networks (DCNNs) for object detection, classification, and extraction have developed rapidly. The use of a DCNN for building feature extraction provides a solution to problems in the fields of artificial intelligence and machine learning. The studies in [22] offer an approach that can be employed in semantic segmentation to mitigate the issue that buildings are complex and alter in size and shape by utilizing low-resolution mapping for nonlinear upsampling. The shifted window transformer as a multiscale learning
Some scholars have used prior knowledge, such as shape and edge features, to improve the accuracy of building extraction and simplify the neural network structure. However, there are still some problems as follows: the proposed solutions are designed in a model-specific way, which did not provide general PIMs and could hardly be transferred to other networks to improve building extraction performance. Most solutions are based on a single-scale space, and the extraction results can hardly provide comprehensive information. Meanwhile, when buildings are extracted at a multiscale, the single-scaled edge information seems not sufficient enough for such tasks.

Recently, the self-supervised learning (SSL) has been a hot topic in deep learning. Many SSL approaches are based on the contrastive learning technique [39]–[41], which introduces an auxiliary network to generate a new view of the input image. Then, the output of the main network and the auxiliary network are used to form a contrastive loss, and no annotations are needed during the training phase. After training, only the trained main branch is kept for the downstream task, while the trained auxiliary network is dropped. Inspired by the SSL paradigm that introduces an auxiliary branch and only keeps the main branch for the downstream task, we propose a PIM to form an auxiliary branch as the training constraint that uses the difference of Gaussian (DoG) operator for multiscale edge extraction. This module could be embedded in most networks for feature maps refinement during the training phase to improve the accuracy of building extraction performance during the inference phase without extra computational complexity. The DoG operator is used by the famous scale-invariant feature transform key point detection algorithm [42] and provides stable features. It introduces the scale space and combines the edge information with different degrees of ambiguity, so that the edge extraction results include both global contour information and detail features, becoming more consistent with visual observation. In addition, it uses Gaussian kernels to avoid the influence of noise. The edge information extracted by the additional DoG operator improves boundary ambiguity and the adhesion phenomenon. Our work mainly includes the following three parts.

1) We propose a PIM as a constraint during the training phase that uses a multiscale DoG operator to extract stable edge information, which can be easily embedded in other networks to improve model performance.

2) We design an end-to-end network for building extraction, called the EfficientNet-UF, which combines a U-structured main network and the proposed PIM to use the DoG edges as the prior information for model performance improvement.

3) We conduct experiments on both an aerial remote sensing dataset and a satellite dataset. The results indicate that our proposed module can improve the building extraction performance without introducing extra computational requirements during the inference phase.

The rest of this article is organized as follows. Section II presents the proposed network and elaborates the materials and structure of our PIM. Section III explains the datasets and the experimental setup. Section IV discusses the results of the experiments. Finally, Section V concludes this article.
II. METHODOLOGY

A. Architecture Overview

The proposed module uses edge information as a constraint to optimize the model training procedure. The architecture of the proposed approach is shown in Fig. 1. It mainly consists of two parts.

1) The main network for building extraction: We design the main network structure for building extraction with a U-structure as the decoder. The backbone could be the popular ResNet series or the EfficientNet [43] for better performance. The EfficientNet is an efficient neural network that systematically scales the model and balances the comprehensive index of the network in three aspects: width, depth, and image resolution.

2) The PIM as the training constraint: We construct the module to encode and decode the multiscale edge information extracted by the DoG operator. The extracted feature maps are decoded and concatenated with the feature maps from the main network and then regressed to a building probability map for loss calculation. Thus, the features from the main network will be refined with the prior information via backpropagation during the training phase.

The structure of each module is described in detail in the following. Note that the PIM is only used for training, and we only kept the parameters of the main network for inference. Thus, the parameters and computational requirements are the same as the models that are trained without the PIM.

B. Main Network for Building Extraction

To better understand the operation mechanism of the main network of the EfficientNet-UF. We plot a schematic diagram of the main extraction network with a U-structure, as shown in Fig. 2. The U-structure is used to reduce information loss during the downampling and upsampling process [44]. EfficientNet could increase the receptive field and capture more fine-grained patterns. Thus, it is selected as the backbone for basic feature extraction. We select the feature maps with resolutions of 256 × 256, 128 × 128, and 64 × 64 in EfficientNet-b3 and use the residual module for downsampling to obtain more detailed information. The transposed convolution operation and the nearest-neighbor interpolation method are used to obtain the feature maps of the same resolution. Similar to FPN, the down-top pathway is selected to construct the pyramidal feature maps with the feature maps of 512 × 512, 256 × 256, and 128 × 128 pixels and to fully use the rich details in the high-resolution images. We adopt the skip-layer connection method to connect the low-level features and the high-level features [26] to achieve good robustness results and generalization ability in the field of semantic segmentation. Meanwhile, the building probability of each pixel could be achieved by regressing the fused feature maps with 1 × 1 sized convolution layers, as the fully convolutional networks do.

It is notable that the main network could be any other building extraction model as long as the feature maps could be concatenated correctly. In Section III, we will use the U-Net as a lightweight network to validate the performance of the proposed approach. The main network is designed exactly the same as the model shown in Fig. 2.

C. DoG Edge Extraction

DoG refers to the difference of Gaussian operation. The DoG operation refers to the subtraction of one Gaussian filtered version of an input image from another less blurred version of the original. For any 2-D images, the definition of the Gaussian filter is expressed as

\[
G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}
\]
Fig. 2. Detailed structure of the main network for building extraction. The features are extracted with the backbone (the EfficientNet here). Then, they are encoded and decoded with a U-structure network. The decoded feature maps are then sent to the PIM for refinement and also regressed to a building probability with $1 \times 1$ sized convolutional layers.

Fig. 3. Diagram of the edge information extracted by the DoG operator. After a lot of edge extraction experiments, there are three values of $k$ set empirically as follows: $k_1 = 0.9$, $k_1 = 1.1$, and $k_1 = 1.2$. The results of this operator could achieve edges at different scales as illustrated in [42]. They will be sent to the PIM after concatenation.

where $\sigma$ is the standard variance of the custom Gaussian kernel.

We use the Gaussian kernel in image convolution operations and obtain the representation of different scale spaces [45]

$$L(x, y, \sigma) = G(x, y, \sigma) \times I(x, y)$$  \hspace{1cm} (2)

where $\sigma$ is considered as the scale factor, $I$ denotes the input image, and $L$ represents the scale space. The DoG operator can be obtained by subtracting two adjacent scale spaces $L$, as follows:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) \times I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma).$$  \hspace{1cm} (3)

Fig. 3 shows the results of an aerial remote sensing image in the scale space generated by the DoG operator when the parameter $k$ is set to 0.9, 1.1, and 1.2, respectively. Then, the edges are concatenated to a 3-D image and sent to the PIM as the input, as shown in Fig. 1. This operation is implemented within the model, and the whole network is an end-to-end network.

D. Prior Information Module

Fig. 4 illustrates the proposed PIM. Since the input edges are lack of texture or color information, a lightweight model (such as the ResNet-18) could be selected as the backbone, which could effectively alleviate the vanishing gradient problem and have excellent generalization ability. A lightweight decoder, containing several layers such as convolutions, batchnorm, ReLU activation, and interpolation, is introduced to decode the extracted features. The edge information extracted by the DoG operator is sent to the prior embedding network as the input. The output of the PIM is the stacked feature maps with resolutions of $256 \times 256$ pixels and $128 \times 128$ pixels, which are concatenated with the features maps from the main network and regressed to a building probability map with $1 \times 1$ convolutional layers, as shown in Figs. 1 and 4. Thus, the prior information is used as a constraint for feature refinement. It is notable that the loss of PIM is also calculated during the training procedure to enhance the edge of buildings, while we only use the output of the main
Fig. 4. Details of the PIM. The input edges will be encoded and decoded by a lightweight module, and the decoded feature maps with high resolutions will be fused with the feature maps from the main network to regress a building extraction probability map during the training phase.

Fig. 5. Comparison of building extraction tested on the Aerial-WHU dataset and the Satellite-WHU dataset before and after adding the PIM to the U-Net network and EfficientNet-UF and other SOTA methods. The correctly extracted buildings are marked in green, the false positive is marked in red, and the false negative is marked in blue.

network as the extracted building probability map during the inference phase, as shown in Fig. 1.

E. Loss Function

Many experiments have proved that choosing an appropriate loss function for different training samples can effectively improve the prediction accuracy. Since we designed a dual network for building extraction, the loss function in our approach is intuitively formulated as

\[ \text{loss} = \text{loss}_1 + \beta \text{loss}_2 \]  

(4)

where \( \text{loss}_1 \) and \( \text{loss}_2 \) are the loss functions of the main network and the PIM, respectively, and the parameter \( \beta \) is the
Fig. 6. Comparison of building extraction tested on the Aerial-WHU dataset before and after adding the PIM constraint to the U-Net network. (a) Original images. (b) Features extracted without the PIM constraint. (c) Features extracted with the PIM constraint. All the feature maps are extracted from the last layer of the trained U-Net.

Fig. 7. Comparison of model complexity and inference accuracy. The radius of the circle increases when the model has larger MACs.

normalized weight to control the contribution of PIM. The binary cross-entropy (BCE) loss is widely used for binary semantic segmentation problems, which is chosen as the loss function of the main network. When using the BCE loss, we need to normalize each pixel in the last layer of the networks to a value between 0 and 1. Therefore, the sigmoid activation function is introduced to achieve the normalized output. Then, the BCE loss with sigmoid is formulated as

\[
\text{loss}_1 = -\sum_{i=1}^{n} \left[ y_i \log(\sigma(x_i)) + (1 - y_i)\log(1 - \sigma(x_i)) \right]
\]  

where \(x_i\) represents the predicted results, \(y_i\) denotes the corresponding ground truth, and the activation function \(\sigma(x_i)\) maps the \(x_i\) into the range of 0–1.
It is suggested by Milletari et al. [46] and He et al. [47] that the Dice loss is efficient for semantic segmentation and is suitable as an auxiliary loss. Thus, we introduce the Dice loss as for the PIM, which is calculated using the following equation [28], [47]:

$$\text{loss}_2 = \frac{2\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i^2 + \sum_{i=1}^{N} g_i^2}$$

(6)

where $N$ represents the total number of individuals involved in operation, $p_i \in P$ is the binary prediction unit, $g_i \in P$ is the true value prediction unit, and $i$ represents the $i$th element.

### III. EXPERIMENTAL SETUP

#### A. Datasets for the PIM Effectiveness Testing

The experiments are conducted on two public datasets, i.e., the Aerial-WHU and Satellite-WHU datasets, which were produced in [9]. The Aerial-WHU building dataset includes images from New Zealand. The ground resolution of the images in the dataset is 0.3 m after the downsampling process. In the experiments, we select approximately 22 000 independent buildings in Christchurch. At the same time, the shapfile data and rasterized data of the buildings are provided. The images are cropped into 8189 tiles with $512 \times 512$ pixels, and the samples are divided into 4736 images in the training set, 1036 images in the validation set, and 2416 images in the test set. The default dataset split is used in our experiments.

The Satellite-WHU building dataset includes images from East Asia, which consists of six neighboring satellite images with 0.45-m ground resolution, covering 860 km$^2$. The vector building map contains 25 749 buildings. The whole image is seamlessly cropped into 17 388 tiles with $512 \times 512$ pixels. It contains 13 662 images in the training set and 3726 images in the test set. There are 3135 positive samples containing buildings in the training set and 10 527 negative samples for training that do not contain buildings. There are 903 positive samples and 2823 negative samples in the test set. In the experiments, we select 95% of the positive training samples as the training set and the whole test samples as used as the test set, while 5% of the positive training samples and all the negative test samples are selected as the validation set to find the model with the best performance during training.

#### B. Implementation Details

During training on the Aerial-WHU dataset and the Satellite-WHU dataset, we trained models on one GPU with a total of three images per mini-batch with the AdamW optimizer for 150 epochs (the loss function basically converges after 150 epochs), and the learning rate was set to an initial value of $1 \times 10^{-3}$. Meanwhile, the learning rate decay amount is set to the value of 0.1 as the default. The version of Python is 3.7.3, and that of PyTorch is 1.9.1. As in most literature, image rotation, image cropping, and horizontal or vertical flipping are introduced for data augmentation during the training phase. During the inference phase, no data augmentation is used for simplicity and fair comparison.

#### C. Experimental Setup to Assess the PIM Effectiveness

To confirm the effectiveness of improving accuracy before and after adding the PIM and the excellent performance of our network compared to the most literature, the proposed...
networks, namely, the U-Net that adds skip-connection layers to improve localization accuracy, the SiU-Net [9] that alleviated the problem of large-scale building, the Farseg [48] that is committed to solving the problem of unbalanced foreground and background and constructing modules for foreground modeling, the MAP-Net [38] that utilizes the multiparallel path to reduce the incorrectly of large-scale and tiny buildings, and the proposed EfficientNet-UF are used in this experiments. It is notable that we dropped the foreground-aware optimization module during training phases compared to the original Farseg approach for a fair comparison, and we selected the ResNet-34 and ResNet-50 as the backbone of Farseg, respectively. Both the aerial and satellite datasets are selected as the experimental datasets. As a classic end-to-end network, the U-Net network has high feature extraction accuracy and high efficiency and is used as a lightweight network structure. Meanwhile, the EfficientNet-UF is used as the large and excellent performance structure. The PIM that contains edge information extracted by the DoG operator is added to the two networks during the training phase. We analyzed the building extraction accuracy before and after adding the PIM.

As in most literature, we report model performance with respect to building extraction, including intersection over union (IoU), precision, recall, and $F_1$-score. The precision, recall, $F_1$-score, and IoU score are defined as follows (Pr. and Rec. denote the precision and recall, respectively):

$$\text{Pr.} = \frac{TP}{TP + FP} \times 100\%$$

$$\text{Rec.} = \frac{TP}{TP + FN} \times 100\%$$

$$F_1 = \frac{2 \times \text{Pr.} \times \text{Rec.}}{\text{Pr.} + \text{Rec.}} \times 100\%$$

$$\text{IoU} = \frac{\text{Pr.} \times \text{Rec.}}{\text{Pr.} + \text{Rec.} - \text{Pr.} \times \text{Rec.}} \times 100\%.$$  

IV. EXPERIMENTAL RESULTS

The results after adding the PIM are introduced in this section. We listed the results of the primitive networks, which training without PIM, and the model training with PIM in Table I to illustrate and compare the effectiveness of the proposed prior information constraint. We set $\beta = 0.3$ in this section.

A. Quantitative and Comparative Analysis of Changes in the Accuracy of the Networks Before and After Adding the PIM

When extracting buildings based on the Aerial-WHU dataset, U-Net was used as a lightweight network structure. After adding the proposed PIM to U-Net, the IoU of building extraction increased by 2.5%. The EfficientNet-UF has a relatively complex structure and has achieved relatively high accuracy; after adding the proposed PIM, the IoU of building extraction increased by 0.2%. The U-Net network’s precision, recall, and $F_1$-score increased by 2.9%, 0.2%, and 1.5%, respectively. Compared to the network training without PIM, the accuracy metrics of the EfficientNet-UF network also improved. Meanwhile, we can conclude that the module is more efficient for the lightweight network than a complex one.

In the building extractions experiment based on the Satellite-WHU dataset, we also used U-Net as the lightweight model. Compared to the network training without the PIM, the accuracy of U-Net improved by 2.0% IoU. In terms of EfficientNet-UF, the IoU score increased by 0.5%. After introducing PIM, the recall and $F_1$-score accuracy metrics of the EfficientNet-UF network also improved. Only the precision was slightly lower. The reason for this may be that when adding prior knowledge as a constraint, the network highlights the buildings instead of the background information, which may lead to more false positives than the network without prior knowledge, which is similar to [26].

To prove the feasibility of the proposed PIM, we compare it with the JLCS-BE network proposed in the literature [37], which used the Sobel operator to extract single-scale edge information and embedded the extracted edges into a model that has a similar structure with the MAP-Net, as shown in Table II. In experiments conducted on the aerial images with densely distributed buildings and satellite images with low resolution, the proposed method outperforms the JLCS-BE network in all the accuracy metrics. This demonstrates the effectiveness of using the multiscale edge information extracted by the DoG operator and decoded by PIM as a training constraint. Among all accuracy

| Dataset  | Method                   | IoU%  | Pr. % | Rec. % | F1%  |
|----------|--------------------------|-------|-------|--------|------|
| Aerial   | U-Net                    | 86.2  | 91.6  | 93.6   | 92.6 |
|          | Farseg(ResNet-34) [49]   | 87.6  | 94.0  | 92.7   | 93.3 |
|          | SiU-Net [9]              | 88.4  | 93.8  | 93.9   | 93.8 |
|          | U-Net+PIM                | 88.7  | 94.5  | 93.8   | 94.1 |
|          | Farseg(ResNet-50) [49]   | 89.6  | 95.0  | 94.0   | 94.5 |
|          | EfficientNet-UF          | 90.1  | 95.3  | 94.2   | 94.8 |
|          | MAP-Net [39]             | 90.1  | 95.4  | 94.2   | 94.8 |
|          | EfficientNet-UF+PIM      | 90.3  | 95.3  | 94.4   | 94.9 |
| Satellite| SiU-Net [9]              | 61.1  | 72.5  | 79.6   | 75.9 |
|          | U-Net                    | 65.8  | 85.8  | 73.9   | 79.4 |
|          | Farseg(ResNet-34) [49]   | 66.5  | 80.8  | 78.9   | 79.9 |
|          | Farseg(ResNet-50) [49]   | 67.4  | 84.5  | 77.0   | 80.6 |
|          | EfficientNet-UF          | 67.8  | 86.4  | 75.8   | 80.8 |
|          | EfficientNet-UF+PIM      | 67.9  | 86.0  | 76.3   | 80.9 |
|          | MAP-Net [39]             | 71.8  | 84.9  | 82.3   | 83.6 |
|          | EfficientNet-UF+PIM      | 72.1  | 84.7  | 82.8   | 83.7 |

*The MAP-Net algorithm is reimplemented with officially released code. Pr. and Rec. denote the precision and recall, respectively.

| Dataset  | Method                   | IoU%  | Pr. % | Rec. % | F1%  |
|----------|--------------------------|-------|-------|--------|------|
| Aerial   | JLCS-BE [38]             | 89.6  | 94.7  | 94.3   | 94.5 |
|          | EfficientNet-UF+PIM      | 90.3  | 95.3  | 94.4   | 94.9 |
|          | improved                 | 0.7   | 0.6   | 0.1    | 0.4  |
| Satellite| JLCS-BE [38]             | 69.5  | 87.1  | 77.4   | 82.0 |
|          | EfficientNet-UF+PIM      | 72.1  | 84.7  | 82.8   | 83.7 |
|          | improved                 | 2.6   | -2.4  | 5.4    | 1.7  |

*The JLCS-BE algorithm is reimplemented with officially released code. The officially provided scale and contrast data augmentation operations are not used during training in our experiments for fair comparisons.

TABLE I

Comparison of Building Extraction Accuracy Tested on the Aerial-WHU Dataset and the Satellite-WHU Dataset

| Dataset  | Method                   | IoU%  | Pr. % | Rec. % | F1%  |
|----------|--------------------------|-------|-------|--------|------|
| Aerial   | U-Net                    | 86.2  | 91.6  | 93.6   | 92.6 |
|          | Farseg(ResNet-34) [49]   | 87.6  | 94.0  | 92.7   | 93.3 |
|          | SiU-Net [9]              | 88.4  | 93.8  | 93.9   | 93.8 |
|          | U-Net+PIM                | 88.7  | 94.5  | 93.8   | 94.1 |
|          | Farseg(ResNet-50) [49]   | 89.6  | 95.0  | 94.0   | 94.5 |
|          | EfficientNet-UF          | 90.1  | 95.3  | 94.2   | 94.8 |
|          | MAP-Net [39]             | 90.1  | 95.4  | 94.2   | 94.8 |
|          | EfficientNet-UF+PIM      | 90.3  | 95.3  | 94.4   | 94.9 |

TABLE II

Comparative Analysis of the Accuracy Tested on the Aerial-WHU Dataset and the Satellite-WHU Dataset
metrics, only the precision of the satellite dataset is lower than that of the JLCS-BE network. The reason for this is similar as mentioned above.

B. Qualitative and Comparative Analysis of Changes in the Accuracy of the Networks Before and After Adding the PIM

The images, labels, and inference results achieved with or without PIM and other state-of-the-art (SOTA) methods are shown in Figs. 5 and 6. Based on the comparative analysis, we obtain the following results.

1) Before adding the PIM, the extracted building edges are irregular and different from the actual shape of the buildings. After adding the module as a constraint, the building boundaries are more complete, the contours become clear, and the phenomenon of sawtooth edges is reduced.

2) In the satellite images, compare the images with and without the PIM. After adding the PIM constraint in the training phase, the probability of missing buildings becomes fewer. In addition, it becomes possible to extract some buildings that cannot be extracted by traditional neural networks, thus improving the building extraction accuracy. Compared with other networks, it is found that the false negative is reduced after adding the PIM.

3) In the aerial images, buildings have a relatively large size and dense distribution. Before adding the PIM for training, the extraction of densely distributed buildings has severe adhesion problems because their spectra are similar. After adding the PIM, the adhesion problems of some building boundaries are effectively solved.

4) As shown in Fig. 6, after adding the PIM constraint, the building boundary extraction results are clearer and more complete. The validity of the addition added to the prior constraint is demonstrated.

In summary, the quantitative and comparative analysis of various building extraction accuracy metrics, including IoU, precision, recall, and $F_1$-score, and the qualitative and comparative analysis of the neural networks’ prediction results before and after adding the PIM, we can draw the following conclusions. In terms of the quantitative metrics, adding the PIM can efficiently improve IoU and $F_1$-score, thereby proving the effectiveness of the experiments. At the same time, adding the PIM to the lightweight network for training can significantly improve its accuracy metrics. Adding the module to the network with a relatively complex structure and high accuracy for training can also slightly improve its building extraction accuracy. In the qualitative analysis of the visualization of the prediction results, we find that before adding the PIM, the prediction results of building boundaries have the problems of adhesive, irregular, and sawtooth edges. After adding the PIM, these problems are improved and can extract some missing buildings, and the prediction results are improved.

C. Comparing the Complexity of Networks in the Training Phase

To compare the complexity of networks and define the lightweight and the complex networks during training, we use multiply accumulate operations (MACs) and parameters (denoted as Params) as indicators for quantitative comparison. As shown in Table III and Fig. 7, after adding the PIM, there is about 25% increase in MACs, and as for the indicator of parameters, there is no significantly increased. Comparing the complexity of the proposed EfficientNet-UF, MAP-Net, and the JLCS-BE network, we find that the computational requirement and the number of parameters of the proposed network are lower than those of the MAP-Net and JLCS-BE networks. That is, in terms of the comprehensive indicators of complexity and performance, the proposed approach has less complexity than the MAP-Net and JLCS-BE networks, while it achieves better performance.

However, the complexity of our approaches is the same with the models trained without the PIM when inferencing, which is another advantage of introducing the training constraint to original models.

D. Ablation Experiments

We set up the ablation experiments using the Satellite-WHU dataset to verify the robustness of the different backbones of the module we proposed. We set the weight $\beta$ of the PIM to 0.1, 0.3, 0.5, 0.7, and 0.9 in the mixed loss function. The impact of varied weight parameter $\beta$ on building extraction accuracy was compared in Table IV and Fig. 8. To determine the efficiency of introducing the PIM and the impact of the main network’s complexity on the accuracy, we ran a comparative analysis on different backbone networks.

To clearly compare and analyze the change in the IoU before and after adding the PIM for training, we plotted an IoU statistics chart with the $\beta$ value varying between 0.1 and 0.9, as shown in Fig. 9. The comparative analysis of the above experiments show that for the ResNet-18 and ResNet-34 backbone networks, which have relatively simple structures, when the weight $\beta$ in the loss function is set to 0.1, 0.3, and 0.5, the IoU about ResNet-18 of building extraction is increased by 0.7%, 2.0%, and 0.8%, the recall is increased by 1.2%, 1.9%, and 0.1%, and the $F_1$-score is increased by 0.5%, 1.4%, and 0.5%, respectively. For the lightweight network, when the weight is within the range of 0.1–0.5, all the accuracy metrics of building extraction are improved. For the ResNet-50 as the backbone, which has much more learnable parameters, we notice that the ResNet-50 has a lower accuracy than the lightweight network mentioned above. The reason for this may be that the lack of training samples leads to the process of training unstably. It is not easy for the network to converge to the optimal solution with significantly increased parameters. Thus, the accuracy is decreased. Although

| Method         | MACs(G) | Params(M) |
|----------------|---------|-----------|
| U-Net          | 45.34   | 24.45     |
| EfficientNet-UF| 52.83   | 11.13     |
| U-Net+PIM      | 53.78   | 27.15     |
| EfficientNet-UF+PIM | 68.38 | 12.83     |
| JLCS-BE        | 71.00   | 31.80     |
| MAP-Net        | 92.69   | 24.00     |
TABLE IV
COMPARATIVE EXPERIMENTS BETWEEN U-NET AND THE U-NET WITH THE PIM WHEN THE WEIGHT PARAMETER $\beta$ VARIES

| Method            | $\beta$ | ResNet-x/y | IoU% | Pr.% | Rec.% | F1% |
|-------------------|---------|------------|------|------|-------|-----|
| U-Net             | 18      | 65.8       | 85.8 | 73.9 | 79.4  |     |
| U-Net             | 34      | 66.3       | 86.7 | 73.9 | 79.7  |     |
| U-Net             | 50      | 64.6       | 86.7 | 71.7 | 78.5  |     |
| U-Net+PIM         | 18+18   | 66.5       | 85.3 | 75.1 | 79.9  |     |
| U-Net+PIM         | 34+18   | 67.2       | 85.4 | 75.8 | 80.3  |     |
| U-Net+PIM         | 50+18   | 66.4       | 87.3 | 73.5 | 79.8  |     |
| U-Net+PIM         | 0.1     | 18+18      | 67.8 | 86.4 | 75.8  | 80.8 |
| U-Net+PIM         | 0.3     | 34+18      | 66.8 | 86.7 | 74.4  | 80.1 |
| U-Net+PIM         | 0.5     | 50+18      | 65.3 | 87.6 | 71.9  | 80.4 |
| U-Net+PIM         | 0.7     | 18+18      | 65.3 | 86.2 | 72.9  | 79.0 |
| U-Net+PIM         | 0.9     | 34+18      | 66.1 | 85.8 | 74.0  | 79.9 |
| U-Net+PIM         | 0.9     | 50+18      | 63.9 | 88.3 | 69.7  | 77.9 |
| U-Net+PIM         | 0.9     | 18+18      | 65.4 | 86.6 | 72.8  | 79.3 |
| U-Net+PIM         | 0.9     | 34+18      | 64.9 | 86.7 | 72.0  | 78.7 |
| U-Net+PIM         | 0.9     | 50+18      | 64.5 | 88.3 | 70.6  | 78.4 |

The ResNet-x/y indicates the backbone used in the main network (denoted as ResNet-x) and that used in the PIM (denoted as ResNet-y).

It cannot achieve high accuracy in the Satellite-WHU dataset, its building extraction accuracy metrics are also improved when $\beta$ is within the range of 0.1–0.5. This proves that adding the PIM as a constraint during the training phase can effectively improve the inference accuracy of building extraction. However, when $\beta$ is higher than 0.5, as the weight increases, the building extraction accuracy is decreased after adding the PIM. The degree of accuracy improvements after adding the PIM for different backbone networks and different weights of the PIM constraint is different. Fig. 9 shows that when the $\beta$ value in the loss function is within the range of 0.1–0.5, the height of the blue bar, which represents the network with the prior information, is greater than that of the red bar, which represents the network training without the PIM. When the $\beta$ value is within the range of 0.6–0.9, the height of the blue bar is less than that of the red bar.

Based on the qualitative and comparative analysis of the experimental data and the predicted building images, we find the following.

1) The analysis of extraction results shows that when the weight $\beta$ in the loss function is within the range of 0.1–0.5, adding the PIM constraint can significantly improve the building extraction accuracy. The buildings that traditional neural networks cannot extract are correctly extracted after adding the constraint for training. However, as the weight increases, the missing buildings cannot be extracted correctly.

2) At the same time, the comparative analysis shows that as the weights increases, the building boundaries are more regular, and the phenomenon of sawtooth edges is reduced. This proves that adding the PIM makes the building extraction results more standardized and effectively improves the problem of sawtooth edges in traditional building extraction methods. When the parameter $\beta > 0.5$, the building extraction accuracy decreased. We assume that the network highlights the prior knowledge as $\beta$ increases, making the training process pay more attention to the building structure and ignore the texture information. On the other hand, the extracted edges by the DoG results contain edges of all the objects in the images, and larger $\beta$ will introduce some irrelevant information.

3) As the weight increases, there are fewer boundary adhesion problems. That is, after adding the PIM, the phenomenon of boundary ambiguity in building extraction is improved.

In summary, the results of the ablation experiments prove the correctness of our conclusions. For the main network with a relatively simple structure, the accuracy metrics of building extraction are significantly improved. For the backbone with a complex structure, the degree of improvement is slight, but the accuracy metrics are still improved. Based on the ablation experiments, we find that when we use the prior knowledge as a constraint, add the PIM for training, and set the weight of the constraint in the mixed loss function to the range of 0.1–0.5, the accuracy metrics are significantly improved. As the weight value increases, the constraint affects the building extraction accuracy. However, based on the comparative analysis of the prediction results, we find that because of adding the PIM and highlighting the buildings, as the weight increases, the problems of adhesive, irregular, and sawtooth edges in building extraction results are improved. Therefore, we can conclude that adding prior knowledge as a constraint for training can improve building boundary irregularity, ambiguity, and the adhesion phenomenon.

V. CONCLUSION

Due to occlusion, shadows, and spectral similarity of different ground objects, the building extraction results of traditional convolutional neural networks have the problems of blurred and incomplete boundaries. To solve this problem, we constructed a PIM to integrate the building edge information in the multiscale space extracted by the DoG operator, combined PIM with the main network to form a dual output network, and selected a
mixed loss function for training. In terms of lightweight networks, the IoU and the $F_1$-score of building extraction were significantly improved. In terms of performance, the proposed PIM outperformed the other network, which added prior knowledge to the main network for training and has achieved relatively good results, outperforming another SOTA algorithm. In addition, due to its excellent adaptability, it can be adapted to the most building extraction main networks to improve extraction accuracy.

Finally, the comparative analysis of the visualization results has shown that the PIM can effectively improve the boundary ambiguity and adhesion phenomenon in the building extraction process of the traditional DCNN. In the future, our further work could be summarized as follows: We will introduce the Online Hard Example Mining [49] to improve the extraction accuracy of small-sized buildings. And more types of prior information, such as the vectors from Open Street Maps, will be considered, and a multimodal network will be built to improve the building extraction performance.

REFERENCES

[1] Z. Zhang and Y. Wang, “JointNet: A common neural network for road and building extraction,” Remote Sens., vol. 11, no. 6, 2019, Art. no. 1029.

[2] Y. Liu et al., “ARC-Net: An efficient network for building extraction from high-resolution aerial images,” IEEE Access, vol. 8, pp. 154997–155010, 2020.

[3] H. Guo, Q. Shi, B. Du, L. Zhang, D. Wang, and H. Ding, “Scene-driven multitask parallel attention network for building extraction in high-resolution remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 5, pp. 4287–4306, May 2021.

[4] C. Qiu et al., “Transferring transformer-based models for cross-area building extraction from remote sensing images,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 15, pp. 4104–4116, 2022.

[5] A. V. Dunaeva and F. A. Komilov, “Specific shape building detection from aerial imagery in infrared range,” Vestnik Yuzhno-Ural’skogo Gosudarstvennogo Universiteta. Seriya “Vychislitel’naya Matematika i Informatica,” vol. 6, no. 3, pp. 84–100, 2017.

[6] S.-H. Zhong, J.-J. Huang, and W.-X. Xie, “A new method of building detection from a single aerial photograph,” in Proc. IEEE 9th Int. Conf. Signal Process., 2008, pp. 1219–1222.

[7] M. Awarjirg, C. Zhang, and C. S. Fraser, “Improved building detection using texture information,” Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci., vol. 38, pp. 143–148, 2011.

[8] G. Ferraioli, “Multichannel InSAR building edge detection,” IEEE Trans. Geosci. Remote Sens., vol. 48, no. 3, pp. 1223–1231, Mar. 2010.

[9] S. Ji, S. Wei, and M. Lu, “Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set,” IEEE Trans. Geo. Remote Sens., vol. 57, no. 1, pp. 574–586, Jan. 2019.

[10] C. Jung and R. Schramm, “Rectangle detection based on a windowed hough transform,” in Proc. 17th Braz. Symp. Comput. Graph. Image Process., 2004, pp. 113–120.

[11] H.-D. Cheng and Y. Sun, “A hierarchical approach to color image segmentation using homogeneity,” IEEE Trans. Image Process., vol. 9, no. 12, pp. 2071–2082, Dec. 2000.

[12] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, “Learning a discriminative feature network for semantic segmentation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1857–1866.

[13] H. Samet, “Neighbor finding techniques for images represented by quadtreess,” Comput. Graph. Image Process., vol. 18, no. 1, pp. 37–57, 1982.

[14] A. Bieniek and A. Moga, “An efficient watershed algorithm based on connected components,” Pattern Recognit., vol. 33, no. 6, pp. 907–916, 2000.

[15] D. A. Clausi, “An analysis of co-occurrence texture statistics as a function of grey level quantization,” Can. J. Remote Sens., vol. 28, no. 1, pp. 45–62, 2002.

[16] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2005, vol. 1, pp. 886–893.
Yujun Quan received the bachelor’s degree in surveying engineering from the North China University of Water Resources and Electric Power University, Zhengzhou, China, in 2021. Currently, she is working toward the master’s degree with the PLA Strategic Support Force Information Engineering University, Zhengzhou. Her main research interests include land cover classification and semantic segmentation.

Anzhu Yu received the bachelor’s degree in remote sensing science and technology and the master’s degree in photogrammetry and remote sensing from PLA Strategic Support Force Information Engineering University, Zhengzhou, China, in 2011 and 2014, respectively, and the Ph.D. degree in photogrammetry and remote sensing from the Institute of Surveying and Mapping, PLA Strategic Support Force Information Engineering University, in 2017. Currently, he is an Associate Professor with PLA Strategic Support Force Information Engineering University. His research interests include signal processing in Earth observation.

Xuefeng Cao received the bachelor’s, master’s, and Ph.D. degrees in cartography and geographic information system from the Institute of Surveying and Mapping, PLA Strategic Support Force Information Engineering University, Zhengzhou, China, in 2006, 2009, and 2012, respectively. Currently, she is an Associate Professor with PLA Strategic Support Force Information Engineering University. Her research interests include signal processing and industry applications in Earth observation.

Chunping Qiu received the bachelor’s and master’s degrees in photogrammetry and remote sensing from the Zhengzhou Institute of Surveying and Mapping, Zhengzhou, China, in 2013 and 2016, respectively, and the Dr.-Ing. degree in signal processing in Earth observation from Technische Universität München, Munich, Germany, in 2020. Since 2021, she has been a Researcher with PLA Strategic Support Force Information Engineering University, Zhengzhou. In 2019, she was a Guest Researcher with the Telecommunications and Remote Sensing Laboratory, University of Pavia, Pavia, Italy. Her main research interests include big Earth data management and self-supervised learning for remote sensing applications, such as image retrieval and image scene classification.

Peipei He received the B.S. degree in photogrammetry and remote sensing from PLA Information Engineering University, Zhengzhou, China, in 2009, and the M.S. and Ph.D degrees in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2012 and 2015, respectively. Currently, he is a Lecturer with the College of Surveying and Geo-Informatics, North China University of Water Resources and Electric Power, Zhengzhou, China. Her research interests include multisource data registration and the precise classification of laser scanning data.