Vector Mapping Method for Buildings in Remote Sensing Images Based on Joint Semantic-Geometric Learning

Jichong Yin, Fang Wu, and Yuyang Qi

***Abstract*—An important high-precision building vector mapping method automatically delineates building polygons from high-resolution remote sensing images. Deep learning methods have greatly improved the accuracy of automatic building segmentation in remote sensing images. However, building polygons in vector forms have a compact and regular structured expression effect, which corresponds more with the application requirements of cartography and geographic information systems (GIS). We propose a vector mapping method for buildings in remote sensing images with joint semantic-geometric learning to generate building polygon vectors in remote sensing images automatically. The method, aiming to provide cartography and GIS data sources, consists of three modules: multi-task segmentation, contour regularization, and polygon optimization. To reduce missing extractions and misextractions and obtain a complete building segmentation mask, the multitask segmentation module performs joint semantic-geometric learning on three related tasks: building instance detection, pixel-wise contour segmentation, and edge extraction. The regularization module normalizes the segmentation mask expression using geometric constraints and image information, whereas the polygon optimization module combines geometric constraints with deep learning methods to ensure vectorization quality. The experimental results show that the proposed method adapts well to building vector extraction tasks under different scenarios and can generate building vector polygons that match the ground truth labels. This method offers significant advantages in solving problems, such as building polygon irregularity and vertex offset.**

***Index Terms*—Contour regularization, joint semantic-geometric learning, multitask segmentation, polygon optimization, remote sensing images.**

I. Introduction

BUILDINGS are the main cartographic element; therefore, accurate building outlines and shapes are critical for cartography andgeographic information system (GIS) applications because they can provide important reference values for applications such as urban planning, three-dimensional (3-D) modeling, change detection, and disaster assessment [1], [2]. High-resolution remote sensing images have become a primary data source for the vector mapping of building polygons as remote sensing and earth observation technology have advanced [3]. Automatic building outline extraction from high-resolution remote sensing images is an important method for improving vector mapping efficiency [4], [5] and has been both the focus and challenge of remote sensing applications and cartographic research [6], [7], [8].

In recent years, deep learning-based building segmentation methods have vastly improved the accuracy of building outline extraction from remote sensing images [6], [9]. However, building segmentation masks derived from this pixel-wise segmentation method often suffer from two major issues.

1) Building segmentation masks have blurred boundaries, redundant points, and weak right-angled features.
2) In practical applications, raster building segmentation masks are not ideal formats for cartographic and GIS applications, necessitating post-processing to convert them to ideal building vector polygons.

Therefore, increasing research is being conducted on regular building polygon extraction from remote sensing images. Jung et al. [10] used the minimum description length technique [11] to regularize the building roof shapes based on airborne LiDAR data. In addition, Zhao et al. [8] improved the minimum description length on this basis, making it suitable for the image domain and using it to regularize the building segmentation mask generated by Mask R-CNN [12]. Wei et al. [13] used the main direction concept to perform fine regularization using the improved Douglas–Peucker algorithm [14], whereas [15] polygonized building segmentation masks using polygon partition refinement. The methods described above effectively address the boundary irregularity of building segmentation masks; however, geometric boundary optimization methods are highly contingent on design features and rules and require a greater degree of manual intervention. Several studies have used deep learning methods to solve the problem of regularizing building outlines. Girard and Tarabalka [16] introduced the polygon boundary loss into the convolutional neural network architecture to attain the regular expression of rectangular buildings. Zorzi and Fraundorfer [17] used generative adversarial networks to regularize building segmentation masks, thereby optimizing their boundaries. Although these methods achieved

Manuscript received 14 July 2023; revised 7 September 2023; accepted 20 September 2023. Date of publication 27 September 2023; date of current version 5 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 42201491 and in part by the Natural Science Foundation for Distinguished Young Scholars of Henan Province under Grant 212300410014. *(Corresponding author: Fang Wu.)*

The authors are with the Institute of Geospatial Information, PLA Strategic Support Force Information Engineering University, Zhengzhou 450001, China (e-mail: jichongy@whu.edu.cn; wufang_630@126.com; qi_yuyang@163.com).

Digital Object Identifier 10.1109/JSTARS.2023.3319605

decent regularization results, their output results remain in raster form, making them unsuitable for practical applications. The regularization effect is also dependent on segmentation mask quality.

Another class of methods learns vector representations of building polygons from remote sensing images. Polygon-RNN [18] and Polygon-RNN++ [19] are semiautomatic polygon labeling methods that use recurrent neural networks to extract polygons by their vertices. PolyMapper [20] builds on this foundation and uses it to extract buildings and roads. Zhao et al. [1] upgraded feature extraction by introducing global context and boundary refinement modules based on PolyMapper, added a channel and spatial attention module to improve the detection module's effectiveness, and obtained building vector representations by learning to predict the locations of key building vertices and connecting them sequentially. The PolygonCNN proposed by [3] first performs segmentation to extract initial building outlines. Then, it utilizes a modified PointNet [21] to learn the shape prior and predict polygon vertices to generate precise building vector results by encoding the vertices of building polygons and merging image features extracted from the segmentation step. Girard et al. [22] proposed a building contour method based on frame field learning to extract regular building footprints directly from remote sensing images as vector polygons. Li et al. [6] designed a multitask learning network for joint semantic-geometric learning with pixel-level building segmentation, multiclass vertex and edge direction prediction and used the vertex generation module to convert segmented contours into high-quality polygon vertices and the polygon refinement network to adjust polygon vertices to more accurate positions automatically. The above methods can obtain vector representations of building polygons but continue to face some limitations. For example, vertex redundancy occurs when dealing with simple shapes, insufficient vertices or self-intersection may occur when extracting complex buildings, and building polygons are irregular or deviate from the ground truth.

To address the aforementioned issues, a joint semantic-geometric learning method for building vector mapping in remote sensing images (JSGLNet) is proposed in this article. The main contributions of this article are described as follows.

1) JSGLNet strives to solve a series of problems in building extraction from remote sensing images, including incomplete segmentation, missing extractions, mis-extractions, blurred edges, irregular boundaries, and nonvector output results. Compared with existing advanced building extraction methods, JSGLNet achieves satisfactory results on two public datasets.
2) To generate building outlines with regular boundaries without losing their geometric details, a contour regularization module is designed, and the rich semantic information provided by the remote sensing image input and the geometric constraint knowledge of the boundaries provided by the segmentation mark is used to solve the blurred building edges problem effectively.
3) We propose a polygon optimization module combining geometry and deep learning, which is used to convert the segmentation mask into a set of valid vertices that represent building instances and predict the offset of each vertex to generate more accurate polygon vertices.

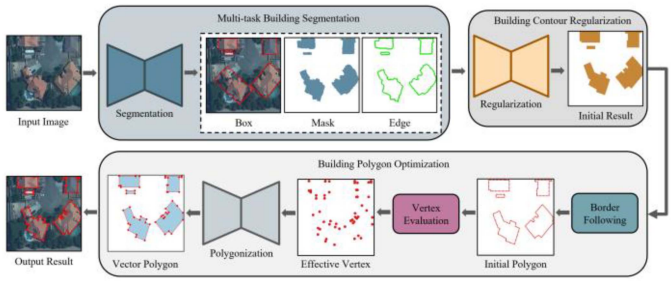

Fig. 1. Overall building vector mapping framework.

II. Methods

A. Overall Structure

Fig. 1 shows the overall JSGLNet framework, including three main parts.

1) *Multitask Building Segmentation Module (MBS)*: The multitask segmentation module takes remote sensing images as input with multiple building instances for joint semantic-geometric learning of three tasks: building instance detection; pixel-wise contour segmentation; and edge extraction. This module improves building segmentation performance by learning shared information across multiple related tasks, resulting in a more accurate building segmentation mask.
2) *Building Contour Regularization Module (BCR)*: Given that polygon reduction techniques applied to irregular boundaries produce inaccurate vector polygons, a contour regularization module for generating normalized building representations is proposed in this article. The contour regularization module normalizes building mask representations by effectively utilizing image information and geometric constraint knowledge of the boundary from the remote sensing image input.
3) *Building Polygon Optimization Module (BPO)*: The polygon optimization module converts the regularized mask into a valid set of vertices that represent building instances and predicts the offset of each vertex to generate more accurate polygon vertices.

The framework takes high-resolution remote sensing images as input and employs a multitask segmentation module to generate an initial building segmentation mask. The contour regularization module is then used to generate a regularized building mask. Finally, the polygon optimization module generates polygon vectors with more accurate vertices, providing data sources for mapping. The multitask segmentation and contour regularization modules are used to generate a complete and standardized initial polygon for each building instance. The polygon optimization module is used to refine the obtained initial polygon so that the initial polygon can better capture the building instance shape and generate a vector polygon that is more suitable for the building contour.

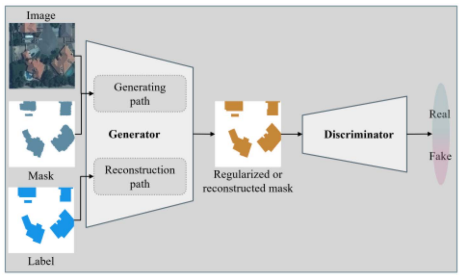

Fig. 2. Overall framework of the building contour regularization module.

B. Multi-Task Building Segmentation Module

Building segmentation detects and extracts building instances from high-resolution remote sensing images. This is analogous to a semantic segmentation or instance segmentation task in computer vision [7], [23], which many deep learning networks can solve. This paper used MultiBuildNet [24] to obtain a complete building segmentation mask. The model employs a multitask learning strategy to complete the building detection, contour segmentation, and edge extraction tasks simultaneously. It achieves significant performance advantages on many open-source building datasets and large-scale high-resolution remote sensing images by using shared information between multiple tasks.

C. Building Contour Regularization Module

Using a multitask segmentation module can ensure high building segmentation accuracy, vastly improving common issues of mis-extractions, missing extractions, and incomplete building extractions from high-resolution remote sensing images. However, such pixel-based building methods cannot effectively resolve the irregular contours and unclear boundaries issues in building segmentation masks, which frequently have rounded corners and irregular edges [25], [26]. Accurate building polygon vector extraction from the initial building segmentation mask is challenging and may result in incorrect building vertices. As a result, a contour regularization module is used to achieve the normalized expression of the building segmentation mask, which is then used to generate building outlines with regular boundaries. This article designs a contour regularization module based on the generative adversarial network [27], with the overall framework shown in Fig. 2.

The contour regularization module comprises a generator network and a discriminator network, with the former attempting to generate ideal building labels and the latter adversarially trained to distinguish generated masks from ideal labels, thereby encouraging the generation of more realistic and reliable building masks. Unlike the standard generation countermeasure network, our contour regularization module consists of a dual-path boundary constraint generator network and a relativistic average discriminator network; its detailed structure is shown in Fig. 3.

In the generator network, we adopt a dual-path design, including two paths: the generating path and the reconstructing path. First, we use two subnets to extract the features of the input remote sensing image, the building noise segmentation mask and the real building label. Both subnetworks comprise two continuous convolution layers, followed by a leaky rectified linear unit (leaky ReLU) and a down-sampling layer. After passing through two subnetworks, the feature map is first connected and then fused by the subsequent convolution layer. Next, a network structure similar to a decoder consisting of two transposed convolution layers and three tiled convolution layers is used to generate regular building masks or reconstruct real building masks. Inspired by U-Net, we also adjust the generator network by adding jump connections. Finally, because the high-frequency details of the generated regularized building mask include positive and negative pixels, we use ReLU as the activation function to ensure no negative pixels in the output results.

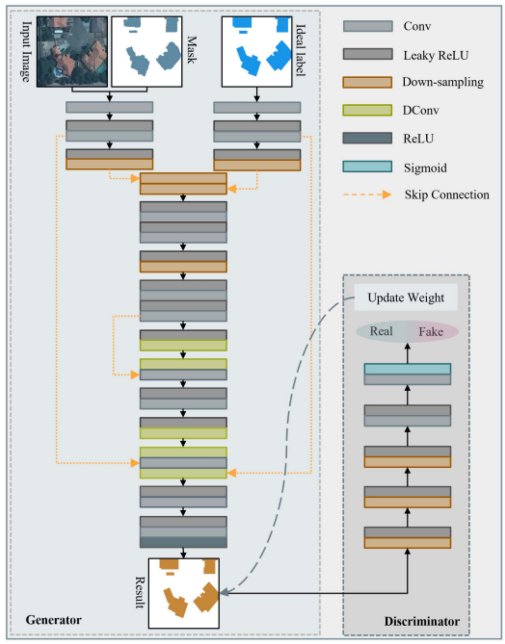

Fig. 3. Detailed structure of the generator and discriminator networks.

Based on the input of the building noise segmentation mask, the generating path adds the remote sensing image as the input to obtain the low-level features and spatial detail information in the remote sensing image to overcome the vertex offset error caused by the inaccurate segmentation mask in the regularization process. The real building label is also used as the input to the reconstruction path to learn the missing advanced features and building semantic-geometric information in the generation path. The boundary loss is used to design the boundary constraint on the generated label. To ensure the model can use the building boundary information in remote sensing images to improve the building regularization effect further, we design the boundary loss to calculate the boundary difference between the generated mask and the ideal label. We used a linear combination of Potts loss L_{potts} [28] and a normalized cut loss L_{ncut} [29] as the boundary loss, which can be expressed as:

$$L_{\text{edge}} = \alpha L_{\text{potts}} + \beta L_{\text{ncut}} \tag{1}$$

$$L_{\text{potts}} = \sum_{k} V^{k'} W (1 - V^{k}) \tag{2}$$

$$L_{\text{ncut}} = \sum_{k} \frac{V^{k'} W (1 - V^{k})}{d' V^{k}} \tag{3}$$

where k represents a certain label and V^k represents a binary indicator vector denoting a split sample with label k. d represents the identity matrix. All the elements on its main diagonal are 1, and all the other elements are 0. W represents the affinity matrix, which reflects the similarity between two samples (generated mask and ideal mask). For any two samples T_i and $\tilde{T}_j$, we can define an affinity matrix $W = [W_{ij}]$, and the elements in the matrix are denoted as W_{ij}. We use a Gaussian kernel as the kernel function. Given the bandwidth σ, there are as follows:

$$W_{ij} = \begin{cases} \exp\left(-\frac{||T_i - \tilde{T}_j||_2^2}{2\sigma^2}\right), \tilde{T}_j \approx T_i \\ 0, \text{otherwise} \end{cases}. \tag{4}$$

The dual-path generator network can obtain not only the low-level and detailed features in remote sensing images, but also the high-level and semantic-geometric features in real labels to achieve the effect of boundary constraints; however, it can also prevent the generated tag and the real tag from being easily recognized by the discriminator due to different codes. By including a reconstruction path, the ideal labels are encoded and decoded as the generator network's input, allowing for the reconstruction mask of the desired labels to be obtained. The reconstruction loss is used to calculate the difference in information before and after the label input goes through the generator network to ensure that the model can generate building labels similar to the generated mask. This paper used the binary cross-entropy loss function to calculate the reconstruction loss L_{rec} of the model, which can be expressed as follows:

$$L_{\text{rec}} = -\sum_{i}^{n} S_i \cdot \log G(S_i, I_i) - \sum_{i}^{n} T_i \cdot \log R(T_i) \tag{5}$$

where n represents the total number of training samples, i represents a training sample, and I_i, S_i and T_i represent the remote sensing image, segmentation mask and ideal label, respectively, corresponding to a training sample. $G(\cdot)$ represents the generation path for regularized masks as similar as possible to that for the ideal labels; $R(\cdot)$ represents the reconstruction path for encoding and reconstructing ideal labels.

In the discriminator network, we use the relativistic average discriminator network. Instead of simply outputting the binary classification result of 0 or 1, we obtain the probability that the real label is more real on average than the generated label, and the value range is between 0 and 1. The discriminator network structure consists of five continuous convolutions with a kernel of 3×3, and the number of filters is doubled. Except for the last layer, each convolution layer is followed by a Leaky ReLU activation function. The last convolution layer is followed by a sigmoid activation function, which is used to evaluate the probability of each input image as a regular building mask or a reconstructed real building mask generated by the generator network. Adversarial loss is used to learn the mapping function between the training domain and the target domain and "cheat" the discriminator network by measuring the probability that the real building label is more real than the generated building mask to prompt the generator network to generate building labels similar to the target domain samples. This process can be realized by symmetrically minimizing the discrimination loss of the discriminator L_D and the adversarial loss of the generator L_G, which is defined as follows:

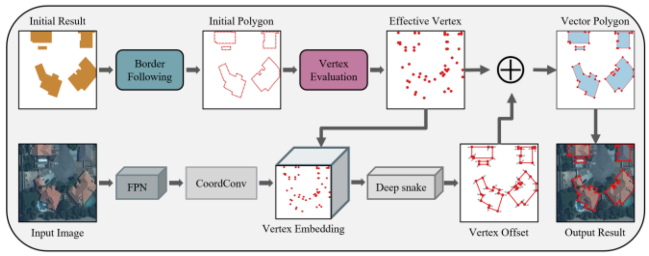

Fig. 4. Building polygon optimization process.

$$L_{adv} = L_D + L_G \tag{6}$$

$$L_G = -\mathrm{E}_{S,I}\left[\log(1 - D(G(S, I)))\right] - \mathrm{E}_{S,I}\left[\log(D(G(S, I)))\right] \tag{7}$$

$$L_D = -\mathrm{E}_{S,I}\left[\log(D(G(S, I)))\right] - \mathrm{E}_{S,I}\left[\log(1 - D(G(S, I)))\right] \tag{8}$$

where $\mathrm{E}[\cdot]$ represents the average operation of a specific small batch of images and L_{adv} represents the total adversarial loss. $G(\cdot)$ represents the generator network, which generates regular building masks. $D(\cdot)$ represents the discriminator network, which aims to distinguish generated masks and reconstructed ideal labels to make the generated building mask as close as possible to the ground truth label.

The total loss of the contour regularization module is the weighted sum of boundary loss, reconstruction loss, and adversarial loss; it can be expressed as follows:

$$L_{\text{reg}} = \alpha L_{\text{potts}} + \beta L_{\text{ncut}} + \gamma L_{\text{rec}} + \eta L_{\text{adv}} \tag{9}$$

where α, β, γ, and η are weighting factors.

D. Building Polygon Optimization Module

GIS and cartographic applications often require vector polygons instead of building masks in raster format generated by multitask segmentation or contour regularization modules. Therefore, a polygon optimization module is designed to convert building masks into vector polygons. The polygon optimization module effectively combines geometric methods with deep learning. The process is shown in Fig. 4 and includes the following steps.

1) *Border Following:* For the regularized building mask, the border following algorithm [30] is adopted to extract the building outlines and obtain the initial polygons.
2) *Vertex Evaluation:* Based on the initial polygons, an initial set of vertices is obtained by placing a vertex every 20 pixels along the contour line. Furthermore, this paper introduced an orientation difference threshold [6] to serve

as a vertex evaluation criterion for selecting valid vertices from the initial set of vertices. For the vertex evaluation criterion, the absolute difference in orientation angles between two adjacent vertices for each initial vertex candidate is calculated, and vertices whose absolute difference exceed the orientation difference threshold are chosen as valid vertices.

3) *Feature Extraction:* The feature pyramid network (FPN) [31] is utilized to learn multiscalar features in remote sensing images to capture the boundary features of complex buildings, which can further improve the results.
4) *Vertex Embedding:* In this article, the vertex representation is based on the feature map extracted from the FPN, and the CoordConv layer is used to introduce coordinate features to provide the network with a concept of location.
5) *Vertex Offset:* The deep snake model [32] is used in this paper to learn the offset of each vertex, in other words, the relative displacement between the predicted valid vertices and the real vertices of their corresponding building polygons. When a polygon vertex moves, so do the two edges that connect to it. To avoid overlapping and the instability caused by the movement process, the attention mechanism [33] is utilized to transmit positional information between vertices and improve predicted vertex offsets. After retrieving the vertex offsets, they are fed back to the original polygons to update their respective shapes. This allows for polygon vector creation with more precise positions.

Our proposed polygon optimization module combines the semantic segmentation method of deep learning with the polygon simplification method based on geometry, a novel deep learning framework. The deep learning semantic segmentation method is good at extracting various features from remote sensing images, while the polygon simplification method based on geometry can obtain polygons with regular boundaries. By combining the two methods, we make full use of the advantages of their respective methods, better capture the geometric shape of building examples, and generate high-quality building polygons in various challenging scenes.

During training, the vertex offset loss and polygon update loss are used to learn and update the weights.

The vertex offset loss is used to prevent unstable offsets of vertices, and the standard deviation loss is used to define the edge length between vertices, which is expressed as follows:

$$L_{\text{cor}} = \sqrt{\frac{\sum \left\| b_i - \bar{b} \right\|_2}{n}} \tag{10}$$

where b_i represents the length of the side between two vertices, and $\bar{b}$ represents the average length of the side.

The polygon update loss is used to restrict deviations in shape between the predicted and ground truth polygons; it is expressed using the Chamfer distance loss [34], [35] and calculated as follows:

$$L_{\text{tra}} = \frac{\sum_i \min\limits_{p \in P} \|\tilde{p}_i - p\|_2}{\left|\tilde{P}\right|} + \frac{\sum_j \min\limits_{\tilde{p} \in \tilde{P}} \|\tilde{p} - p_j\|_2}{|P|} \tag{11}$$

where $\tilde{p}$ and p are the rasterized boundary pixels of the predicted polygon and ground truth polygon, respectively.

III. Datasets and Experimental Details

A. Description of Datasets

Two challenging benchmark datasets of high-resolution remote sensing building images were used in this paper to test the method's performance and generalizability: the AIcrowd mapping challenge dataset (AIcrowd) [36] and the WHU building dataset (WHU) [23], [37]. Both datasets cover different regions (the United States of America and New Zealand) and have different image sources (satellite and aerial images). Furthermore, the spatial resolution of the images and scene complexity vary.

1) AIcrowd Mapping Challenge Dataset: The AIcrowd Mapping Challenge dataset is a large-scale dataset of satellite images with a spatial resolution of 0.3 m and a sample size of 300 × 300. The training set consists of 280 741 images and 2 400 000 annotated building instances, with the validation set containing 60 317 images and 515 000 building instances and the test set containing 60 697 images.

2) WHU Building Dataset: The WHU building dataset primarily comprises an aerial dataset and a satellite dataset, with aerial images of the WHU building dataset used in the experiment. The WHU building dataset aerial images contain a greater number of building instances with varying styles, scales, uses, and colors. They can be used for large-scale building extraction from high-resolution remote sensing images. The images in this dataset have an original spatial resolution of up to 0.075 m, and the sample resolution was down-sampled to 0.3 m, with a sample size of 512 × 512. The dataset's coverage area includes 220 000 building instances, with a training set, a validation set, and a test set containing 4736, 1036, and 2416 images, respectively.

B. Experimental Details

1) Experimental Setup: The proposed framework for building vector mapping was deployed and tested with PyTorch 1.7.1 on a 64-bit Ubuntu system equipped with an NVIDIA Corporation UTTU102GL [Quadro RTX 6000/8000] GPU. To ensure the objectivity of the test results, all test networks were optimized using the Adam algorithm, with the initial learning rate and number of batches set to 0.00004 and 4, respectively. All experimental networks iterated over the dataset 150 times from zero during training.

2) Evaluation Indicators: Based on the three building vector mapping modules, the performance of the proposed method was evaluated at the pixel, object and vector levels.

At the pixel level, intersection over union (IoU), recall and precision were used to evaluate the overall building segmentation performance [38], [39], [40]. IoU represents the ratio of the intersection over the union of predicted building and real building pixels. *Recall* represents the ratio of correctly predicted building pixels to real building pixels. *Precision* represents the proportion of correctly predicted building pixels among predicted building pixels.

At the object level, curvature error (E_{cur}) was introduced to evaluate the accuracy of the building boundary representations [41], [42]. For a regularized object on the generated label $\hat{T}$ and a reference object $B_i (i = 1, 2, 3, \ldots, n)$ on the ground truth label T, the curvature error (E_{cur}) can be used to measure the difference in boundaries between the regularized building and ground truth building by using

$$E_{\text{cur}}(B_i, M_i) = \|f_c(M_i) - f_c(B_i)\| \tag{12}$$

for calculation, where f_c represents the curvature function of the contour. $f_c(B_i)$ is usually small when B_i is manually annotated. If $E_{\text{cur}}(B_i, M_i)$ is very large, it indicates that the building boundaries in the generated labels are very uneven.

At the vector level, considering the vectorization performance of the generated vector buildings, the shape error (E_{shp}) [42], [43] and vertex offset error (E_{ver}) [3] were calculated to evaluate the similarity in shape between the generated polygons and ground truth polygons and the vertex accuracy of the generated polygons. The shape error (E_{shp}) was used to evaluate the difference in the building shapes, with a calculation formula of

$$E_{\text{shp}}(B_i, M_i) = \|f_s(M_i) - f_s(B_i)\| \tag{13}$$

$$f_s(M_i) = \frac{4\pi |M_i|}{p_{M_i}^2} \tag{14}$$

where p_{M_i} represents the perimeter of M_i. The value of $f_s(M_i)$ is 1 for circles and $\frac{\pi}{4}$ for rectangles. In practical cartographic applications, structured and simplified representations are required for building polygon vectors. Therefore, using vertices for evaluation can better reflect the performance of generated results in practical engineering applications. The Hausdorff distance [44] was used to calculate the vertex offset error of building polygons. The Hausdorff distance reflects the vertex accuracy of the generated building polygons by taking the largest of the smallest distances measured between each vertex of the generated building and ground truth building polygon vectors.

IV. Experimental Results and Analysis

To verify the effectiveness of the proposed method, it was tested on two large-scale building datasets and compared with other advanced building extraction methods, including pixel-wise segmentation methods (AFL-Net [45] and the mask-based boundary segmentation and regularization method [BSR] [13]) as well as polygon extraction methods (based on frame field learning [FFL] [22] and PolygonCNN [3]). Furthermore, a corresponding ablation study was conducted to investigate the contribution of each module to the proposed method.

A. Results of the Aicrowd Mapping Challenge Dataset

Table I compares the results of the proposed method to four other advanced methods on the AIcrowd mapping challenge dataset. Our method achieved the best results and had the smallest shape and vertex offset errors on the AIcrowd mapping challenge dataset. This indicates that compared to other methods, the proposed method had the lowest missed detection rate,

TABLE I
Accuracy Evaluation Results of the AIcrowd Mapping Challenge Dataset

Methods	*IoU*	*Recall*	*Precision*	E_{cur}	E_{shp}	E_{ver}
AFL-Net	**0.846**	0.906	0.917	6.63	5.39	5.72
BSR	0.825	0.896	0.887	**2.89**	6.21	4.31
FFL	0.837	0.886	**0.937**	4.32	4.36	2.69
PolygonCNN	0.829	0.895	0.884	3.96	3.67	2.35
Ours	0.838	**0.913**	0.896	3.28	**2.64**	**1.74**

The bold entities represent the best accuracy evaluation results.

the highest shape similarity with the ground truth labels, and the highest positional accuracy. As our method regularized the segmentation mask and optimized the polygons while generating building polygon vectors, the building expression effect was more simplified, which also reduced the IoU of the buildings to a certain extent. While AFL-NET produced the best IoU result, it is a pixel-wise segmentation method that produces many irregular building outlines, resulting in a higher curvature, shape, and vertex offset errors. Postprocessing was used in the mask-based boundary segmentation and regularization method to regularize building outlines and eliminate jagged edges. While this reduced curvature error, it increased shape error.

Fig. 5 gives the results of a visual comparison of the proposed method with four other advanced methods tested on the AIcrowd mapping challenge dataset. The other four methods produced more compact and regularized representations than AFL-Net. While the mask-based boundary segmentation and regularization method generated correct regularized representations for simple buildings, simplification errors were common when dealing with complex buildings, resulting in large shape errors. The method based on FFL generated more regular representations, but the generated polygons differed from ground truth labels, and processing building vertices was not sufficiently smooth. Although both PolygonCNN and our method produced compact and regular representations, PolygonCNN was error-prone when dealing with buildings with 'holes'. Furthermore, our method better processed the building polygon details, generating building polygons highly consistent with ground truth labels. In summary, according to visual interpretation, it is evident that our method can adapt well to different building vector extraction task scenarios for the AIcrowd Mapping Challenge dataset. It can improve irregular boundaries and excessive sharp corners in building extraction to some extent and generate results consistent with the shape of ground truth labels, thereby being more advantageous than other comparison methods.

B. Results of the WHU Building Dataset

Fig. 6 shows the visual comparison results of the proposed method r and four other advanced methods tested on the WHU building dataset. As illustrated in the figure, the proposed method can accurately describe buildings of various sizes and shapes in complex scenes compared with other methods. Furthermore, it can provide precise geometric details and vertex positions, allowing ground truth labels to maintain highly consistent shapes with building polygons.

Fig. 5. Experimental results of various methods on the AIcrowd mapping challenge dataset.

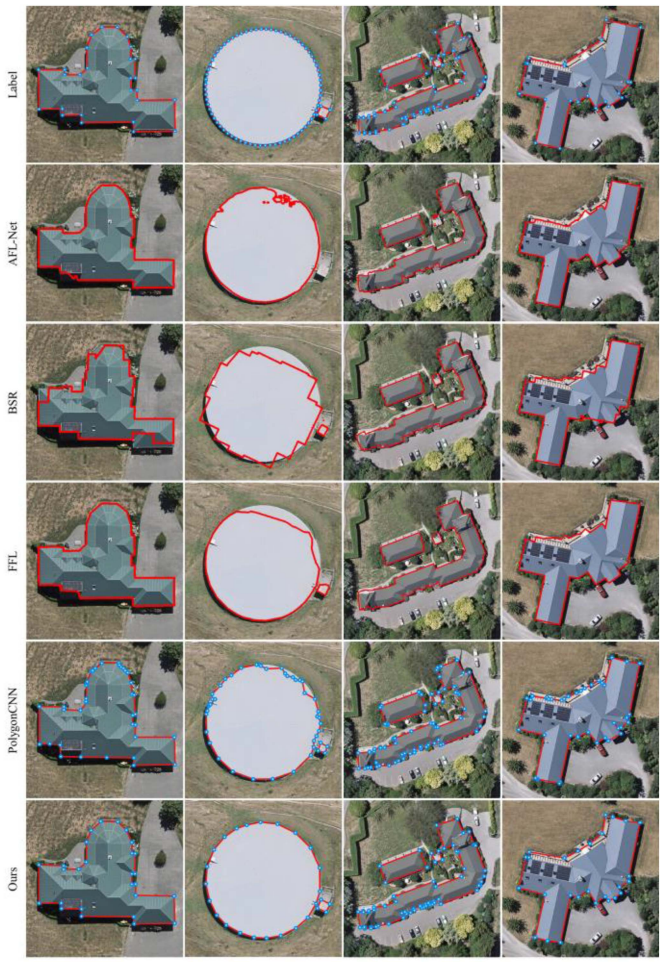

Fig. 6. Experimental results of various methods on the WHU building dataset.

TABLE II
ACCURACY EVALUATION RESULTS OF THE WHU BUILDING DATASET

Methods	*IoU*	*Recall*	*Precision*	E_{cur}	E_{shp}	E_{ver}
AFL-Net	0.914	0.954	0.956	6.32	5.31	1.79
BPR	0.892	**0.966**	0.920	**2.52**	5.76	2.47
FFL	0.906	0.942	0.957	3.23	3.43	1.34
PolygonCNN	0.886	0.938	0.942	2.68	3.18	1.68
Ours	**0.926**	0.957	**0.958**	2.79	**2.67**	**0.76**

The bold entities represent the best accuracy evaluation results.

TABLE III
QUANTITATIVE EVALUATION RESULTS OF EACH MODULE ON THE AICROWD DATASET

Methods	*IoU*	*Recall*	*Precision*	E_{cur}	E_{shp}	E_{ver}
MBS	0.849	0.912	0.923	6.35	5.32	5.68
MBS+BCR	**0.857**	**0.928**	**0.936**	3.29	3.14	1.96
MBS+BPO	0.833	0.904	0.889	4.72	4.19	1.81
MBS+BCR+BPO	0.838	0.913	0.896	**3.28**	**2.64**	**1.74**

The bold entities represent the best accuracy evaluation results.

Table II reports the quantitative comparison results between the proposed method and four other advanced methods on the WHU building dataset. Our method outperformed others on the WHU building dataset, with good results in the three evaluation indicators based on pixels, objects, and vectors. The method presented in this paper achieved the best results in IoU, precision, shape error, and vertex offset error and is only second to the mask-based boundary segmentation and regularization method in recall and curvature error. While the mask-based boundary segmentation and regularization method achieved the smallest curvature error, the shape error increased due to the excessive pursuit of simplified representation, such that its results were too different from ground truth labels. Our method attained the lowest shape and vertex offset errors while ensuring a low curvature error, effectively maintaining an equilibrium between simplified representation and shape preservation.

C. Ablation Study

As the proposed building vector mapping framework operates in a pipeline, with the three modules, MBS, BCR, and BPO, independent of each other, we conducted a combined test on these three modules to further evaluate the effect of each module. Tables III and IV report the quantitative evaluation results of the accuracy of each module on the AIcrowd and WHU datasets, respectively. Fig. 7 shows the comparison results of the ablation experiment accuracy. Fig. 8 shows the visual comparison results of the ablation experiment.

TABLE IV
QUANTITATIVE EVALUATION RESULTS OF EACH MODULE ON THE WHU DATASET

Methods	*IoU*	*Recall*	*Precision*	E_{cur}	E_{shp}	E_{ver}
MBS	0.883	0.941	0.935	5.34	4.85	1.73
MBS+BCR	0.888	0.938	0.947	3.13	2.79	1.35
MBS+BPO	0.895	0.943	0.957	4.35	3.56	0.89
MBS+BCR+BPO	**0.926**	**0.957**	**0.958**	**2.79**	**2.67**	**0.76**

The bold entities represent the best accuracy evaluation results.

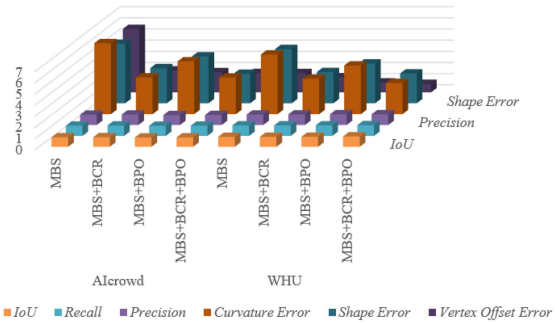

Fig. 7. Accuracy comparison results of the ablation experiment.

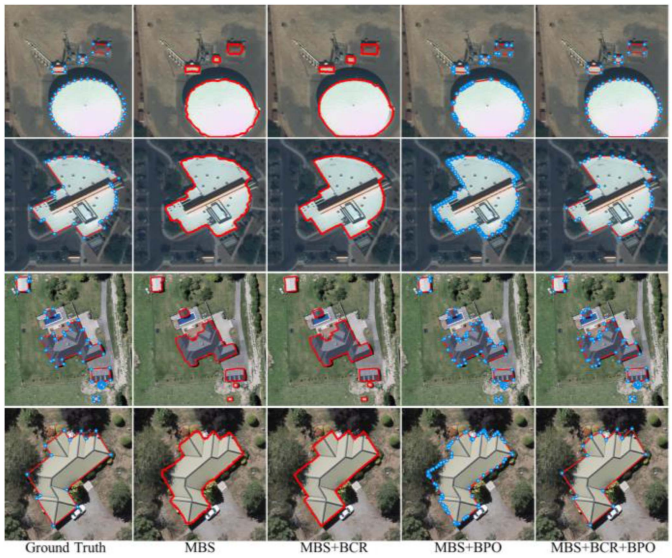

Fig. 8. Visual comparison results of the ablation experiment.

As Tables III and IV and Fig. 7 show, the BCR module effectively reduced the curvature and shape errors of the building segmentation mask, while the BPO module effectively reduced the vertex offset error of the building segmentation mask. This indicates that the contour regularization module helped to eliminate the jagged edges and sharp corners of the segmentation mask to obtain smooth and regular buildings. The polygon optimization module was robust at adjusting the building polygon vertices to more accurate positions to ensure excellent positional accuracy.

From Fig. 8, the multitask segmentation module obtained relatively accurate segmentation masks even under different scenes, such as occluded buildings, large differences in building sizes in the image, and complex building shapes. This demonstrates that the multitask segmentation module ensured complete and accurate segmentation results while resolving misextractions, missing extractions, and incomplete segmentation. The contour regularization module eliminated the segmentation mask's jagged edges and solved the problems of blurred edges and an excessive number of sharp corners. The polygon optimization module further improved the positional accuracy of building polygon vertices, which effectively solved the problems of redundant points in building polygons and insufficient vertex accuracy. This allowed for more compact and regular vector representations to be obtained. To achieve automatic building vector extraction from remote sensing images, the building vector mapping framework relied on three modules: multitask segmentation; contour regularization; and polygon optimization.

TABLE V
COMPARISON RESULTS OF THE COMPLEXITY OF EACH MODEL

Methods	*Parameters (M)*	*FLOPs (G)*	*Training speed (FPS)*	*Inference speed (FPS)*
AFL-Net	10.13	26.16	32.83	64.43
FFL	51.87	103.26	22.64	53.07
PolygonCNN	56.85	193.36	22.57	44.36
MBS	20.68	60.34	30.39	53.74
BCR	30.74	84.52	24.25	46.06
BPO	**9.43**	**23.56**	**33.65**	**67.41**

The bold entities represent the best accuracy evaluation results.

D. Complexity Comparison

Complexity is a critical factor that affects the practical application of a model. In building extraction tasks based on deep learning methods, lower numbers of parameters and floating-point operations (FLOPs) often result in faster training and inference speeds. A model with lower complexity is more convenient for practical applications. To objectively evaluate the complexity of each model, the number of parameters, the number of FLOPs, the training speed and the inference speed are calculated separately for each model. On an NVIDIA RTX 3090 GPU, the training speed is expressed as the number of frames per second (FPS) required for training an input image of size $3 \times 512 \times 512$, and the inference speed is expressed as the number of FPS required for inferring an input image of size $3 \times 512 \times 512$. Because our method relies on three independent modules for building vector mapping, we compared the complexity of the three modules in JSGLNet with other building extraction models based on deep learning. The results of the quantitative comparison of the complexity of each model are given in Table V.

In Table V, we can easily find that the polygon optimization module (BPO) obviously achieved the fastest training speed and reasoning speed with few parameters and FLOPs. The multitask segmentation module (MBS) and the contour regularization module (BCR) were not as efficient as AFL-Net in training and inferring. However, they still achieved faster training and inference speeds than FFL and PolygonCNN.

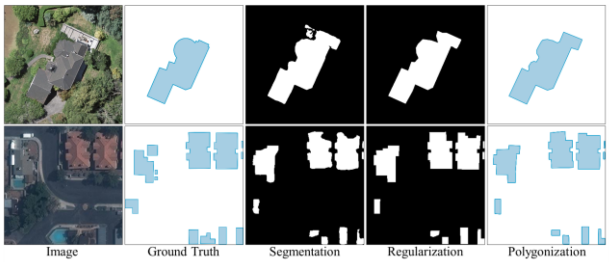

Fig. 9. Examples of errors in builder vector mapping.

V. Discussion

According to the performance comparison of our method on the AIcrowd and WHU datasets, it is obvious that when the remote sensing image quality is good, we can obtain a high level of automation and a high-quality building vector polygon. In contrast, when the quality of the remote sensing image is poor and it is difficult to obtain a complete or correct building segmentation mask due to occlusion or shadow, our method cannot correctly generate the building vector polygon, as shown in Fig. 9. In Fig. 9, we can easily find that when the segmentation effect is good, our method can eliminate the jagged edges and sharp corners of the segmentation mask, obtain a smooth and regular building, and move the vertices of the building polygon to a more accurate position. However, the subsequent contour regularization module and polygon optimization module cannot deal with the false extraction, missing extraction and incomplete segmentation problems in the segmentation results. This requires an MBS module to ensure complete and correct building segmentation results. Therefore, our method can ensure a high-quality building vector drawing effect on the basis of obtaining a complete building segmentation mask. Once there is a problem with the segmentation quality, the subsequent contour regularization module and polygon optimization module will continue this error and accumulate errors, leading to incorrect building vector mapping results.

In addition, our method relies on three independent modules for building vector mapping, which requires many training and inference steps. Although the training and inference speed of a single module can meet the practical application requirements, repeated training and inferring are cumbersome. Therefore, in the future, we hope to develop an end-to-end method from remote sensing images directly to vector maps to integrate segmentation and vectors.

VI. Conclusion

This article proposed a joint semantic-geometric learning method for building vector mapping in remote sensing images to extract building polygon vectors from high-resolution remote sensing images automatically. The method relies on three modules, multitask segmentation, contour regularization, and polygon optimization, to accurately extract building polygon vectors. Additionally, this method provides data sources for cartography and GIS. The multitask segmentation module performs joint semantic-geometric learning on three related tasks, building instance detection, pixel-wise contour segmentation, and edge extraction, to obtain accurate and complete building segmentation masks. The regularization module uses boundary, reconstruction and adversarial losses to effectively fuse image information with geometric constraints to attain the regular expression of building segmentation masks. Finally, the polygon optimization module effectively combines geometric constraints with deep learning methods to solve redundant building polygon vertices and positional offsets problems, thereby obtaining vector representations that conform to the ground truth. Experiments on open-source building datasets and ablation tests show that compared to other building extraction methods, the proposed method has significant performance advantages because it can better adapt to the building vector mapping task under various scenarios. The proposed method effectively solves the incomplete segmentation, irregular edges, and non-compact results problems in building extraction, and it is a good first step toward automatically generating building vector maps from remote sensing images. However, for vector mapping, our method depends on three independent modules. While these are simple to combine and migrate, the research paradigm of such pipeline modes accumulates errors during the extraction process, lowering polygon quality. Furthermore, segmentation mask quality is critical in this paradigm, with our method relying on the entire segmentation mask to ensure vectorization quality. Therefore, future research will focus on developing an end-to-end research paradigm for directly extracting building vector polygons from remote sensing images.

References

[1] W. Zhao, C. Persello, and A. Stein, "Building outline delineation: From aerial images to polygons with an improved end-to-end learning framework," *ISPRS J. Photogramm. Remote Sens.*, vol. 175, pp. 119–131, May 2021.

[2] K. Zhao, M. Kamran, and G. Sohn, "Boundary regularized building footprint extraction from satellite images using deep neural network," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. V-2–2020, pp. 617–624, Aug. 2020.

[3] Q. Chen, L. Wang, S. L. Waslander, and X. Liu, "An end-to-end shape modeling framework for vectorized building outline generation from aerial images," *ISPRS J. Photogramm. Remote Sens.*, vol. 170, pp. 114–126, Dec. 2020.

[4] M. Persson, M. Sandvall, and T. Duckett, "Automatic building detection from aerial images for mobile robot mapping," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Automat.*, 2005, pp. 273–278.

[5] H. L. Yang, J. Yuan, D. Lunga, M. Laverdiere, A. Rose, and B. Bhaduri, "Building extraction at scale using convolutional neural network: Mapping of the United States," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2600–2614, Aug. 2018.

[6] W. Li, W. Zhao, H. Zhong, C. He, and D. Lin, "Joint semantic-geometric learning for polygonal building segmentation," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1958–1965.

[7] S. Zorzi, K. Bittner, and F. Fraundorfer, "Machine-learned regularization and polygonization of building segmentation masks," in *Proc. Int. Conf. Pattern Recognit.*, 2021, pp. 3098–3105.

[8] K. Zhao, J. Kang, J. Jung, and G. Sohn, "Building extraction from satellite images using mask R-CNN with building boundary regularization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 242–246.

[9] J. Huang, X. Zhang, Q. Xin, Y. Sun, and P. Zhang, "Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network," *ISPRS J. Photogramm. Remote Sens.*, vol. 151, pp. 91–105, May 2019.

[10] J. Jung, Y. Jwa, and G. Sohn, "Implicit regularization for reconstructing 3D building rooftop models using airborne LiDAR data," *Sensors*, vol. 17, no. 3, Mar. 2017, Art. no. 621.
[11] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, Sep. 1978.
[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.
[13] S. Wei, S. Ji, and M. Lu, "Toward automatic building footprint delineation from aerial images using CNN and regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2178–2189, Mar. 2020.
[14] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica, Int. J. Geographic Inf. Geovisualization*, vol. 10, no. 2, pp. 112–122, Dec. 1973.
[15] M. Li, F. Lafarge, and R. Marlet, "Approximating shapes in images with low-complexity polygons," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2020, pp. 8630–8638.
[16] N. Girard and Y. Tarabalka, "End-to-end learning of polygons for remote sensing image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp .*, 2018, pp. 2083–2086.
[17] S. Zorzi and F. Fraundorfer, "Regularization of building boundaries in satellite images using adversarial and regularized losses," in *Proc. IEEE Int. Geosci. Remote Sens. Symp .*, 2019, pp. 5140–5143.
[18] L. Castrejón, K. Kundu, R. Urtasun, and S. Fidler, "Annotating object instances with a polygon-RNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2017, pp. 4485–4493.
[19] D. Acuna, H. Ling, A. Kar, and S. Fidler, "Efficient interactive annotation of segmentation datasets with polygon-RNN++," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2018, pp. 859–868.
[20] Z. Li, J. D. Wegner, and A. Lucchi, "Topological map extraction from overhead images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis .*, 2019, pp. 1715–1724.
[21] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2017, pp. 77–85.
[22] N. Girard, D. Smirnov, J. Solomon, and Y. Tarabalka, "Polygonal building extraction by frame field learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2021, pp. 5887–5896.
[23] S. Ji and S. Wei, "Building extraction via convolutional neural networks from an open remote sensing building dataset," *Acta Geodaetica et Cartographica Sinica*, vol. 48, no. 4, pp. 448–459, 2019.
[24] J. Yin, F. Wu, Y. Qiu, A. Li, C. Liu, and X. Gong, "A multi-scale and multi-task deep learning framework for automatic building extraction," *Remote Sens.*, vol. 14, no. 19, Jan. 2022, Art. no. 4744.
[25] H. Guo, Q. Shi, A. Marinoni, B. Du, and L. Zhang, "Deep building footprint update network: A semi-supervised method for updating existing building footprint from bi-temporal remote sensing images," *Remote Sens. Environ.*, vol. 264, Oct. 2021, Art. no. 112589.
[26] Z. Wang et al., "A multi-scale edge constraint network for the fine extraction of buildings from remote sensing images," *Remote Sens.*, vol. 15, no. 4, Jan. 2023, Art. no. 927.
[27] I. J. Goodfellow et al., "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[28] M. Tang, F. Perazzi, A. Djelouah, I. Ben Ayed, C. Schroers, and Y. Boykov, "On regularized losses for weakly-supervised CNN segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 507–522.
[29] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised CNN segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1818–1827.
[30] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," *Comput. Vis. Graph. Image Process.*, vol. 29, no. 3, Mar. 1985, Art. no. 396.
[31] T. - Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2017, pp. 936–944.
[32] S. Peng, W. Jiang, H. Pi, X. Li, H. Bao, and X. Zhou, "Deep snake for real-time instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2020, pp. 8530–8539.
[33] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[34] J. Liang, N. Homayounfar, W. - C. Ma, Y. Xiong, R. Hu, and R. Urtasun, "PolyTransform: Deep polygon transformer for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2020, pp. 9128–9137.
[35] N. Homayounfar, W. - C. Ma, S. K. Lakshmikanth, and R. Urtasun, "Hierarchical recurrent attention networks for structured online maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2018, pp. 3417–3426.
[36] S. P. Mohanty et al., "Deep learning for understanding satellite imagery: An experimental survey," *Front. Artif. Intell.*, vol. 3, 2020, Art. no. 534696.
[37] 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. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019.
[38] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2016, pp. 724–732.
[39] Z. He, H. Ding, and B. An, "E-Unet: A atrous convolution-based neural network for building extraction from high-resolution remote sensing images," *Acta Geodaetica et Cartographica Sinica*, vol. 51, no. 3, pp. 457–267, Mar. 2022.
[40] Y. Xie et al., "Refined extraction of building outlines from high-resolution remote sensing imagery based on a multifeature convolutional neural network and morphological filtering," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1842–1855, 2020.
[41] C. Persello and L. Bruzzone, "A novel protocol for accuracy assessment in classification of very high resolution images," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 3, pp. 1232–1244, Mar. 2010.
[42] L. Ding, H. Tang, Y. Liu, Y. Shi, X. X. Zhu, and L. Bruzzone, "Adversarial shape learning for building extraction in VHR remote sensing images," *IEEE Trans. Image Process.*, vol. 31, pp. 678–690, 2022.
[43] I. Lizarazo, "Accuracy assessment of object-based image classification: Another STEP," *Int. J. Remote Sens.*, vol. 35, no. 16, pp. 6135–6156, Aug. 2014.
[44] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the hausdorff distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 9, pp. 850–863, Sep. 1993.
[45] Y. Qiu et al., "AFL-Net: Attentional feature learning network for building extraction from remote sensing images," *Remote Sens.*, vol. 15, no. 1, Jan. 2023, Art. no. 1.

Jichong Yin received the B.Sc. degree in surveying and mapping engineering from Wuhan University, Wuhan, China, in 2019. He is currently working toward the Ph.D. degree in surveying and mapping science and technology with the Institute of Geospatial Information, PLA Strategic Support Force Information Engineering University, Zhengzhou, China.

His research interests include remote sensing image processing and deep learning.

Fang Wu received the Ph.D. degree in cartography and geographic information engineering from Information Engineering University, Zhengzhou, China, in 2000.

She is currently a Professor with the Institute of Geospatial Information, PLA Strategic Support Force Information Engineering University. She has co-authored more than 100 papers. Her research interests include mapping generalization and spatial data mining.

Yuyang Qi received the B.Sc. degree in surveying and mapping engineering from the School of Surveying and Mapping Engineering, North China Institute of Science and Technology, Langfang, China, in 2020. He is currently working toward the M.Sc. degree in surveying and mapping science and technology with the Institute of Geospatial Information, PLA Strategic Support Force Information Engineering University, Zhengzhou, China.

His research interests include spatial data mining and natural language processing.