# Easy-Net: A Lightweight Building Extraction Network Based on Building Features

Huaigang Huang, Jiabin Liu, and Ruisheng Wang, *Senior Member, IEEE*

***Abstract*— The efficient, accurate, and automatic extraction of buildings from remote sensing imagery is a key task in the intelligent extraction of remote sensing information owing to its importance in applications including urban planning, change detection, and unmanned aerial vehicle (UAV) navigation. However, the fast and accurate extraction of buildings from remote sensing images remains difficult owing to the complex, variable nature of geographic information, and variable external appearances of buildings. This is because many existing building extraction networks fail to incorporate building features into their design. Also, generally, simple lightweight networks do not accurately identify buildings, while large complex networks have high operational costs. Therefore, in this article, we proposed a simple, effective feature fusion strategy based on the building features extracted by the lightweight backbone network; also we have improved the feature fusion performance by combining the advantages of a convolutional neural network (CNN) and transformer; and presented the lightweight building extraction network called Easy-Net. We conducted experiments comparing Easy-Net with existing high-performing networks on the public dataset WHU and self-made datasets; results showed the efficiency and accuracy of our method in the task of building extraction from remote sensing images. Thus, Easy-Net was found to be a promising alternative to existing building extraction networks. Code has been released at: github.com/teddy132/EasyNet_for_building_extraction.**

***Index Terms*— Building extraction, building features, lightweight network, transformer.**

## I. Introduction

In RECENT years, with the rapid development of various computer vision algorithms based on deep learning, the intelligent extraction of remote sensing information—a popular research area of remote sensing—has made great progress. As buildings are closely related to human activities, the extraction of buildings from remote sensing images has been a popular research area in this field. Building information is important for many downstream applications and tasks, such as urban modeling, urban planning, urban disaster emergency response, and change monitoring. As cities are undergoing rapid renewal, it is critical to extract building information quickly and accurately.

Traditional building extraction methods [1], [2], [3], [4], [5], [6], [7], [8] primarily employ some unique building features to represent buildings through experience and complete the extraction of buildings through feature sets composed of these features, which generally include color, texture, shadow, shape, length, and edge [9]. Zhang [10] proposed a building extraction method combining the multispectral rule classification features of buildings and texture filtering methods. Further, Lin and Nevatia [11] introduced a building detection method for aerial images based on shadows and roof geometry. Later, researchers took the extracted building texture and geometric features and trained them using high-performing machine learning algorithms to obtain building classifiers [12], [13], [14], which improved the building extraction results to some extent. However, building extraction using these low- and medium-level features is not ideal, as they are affected by many external factors, including weather, sensors, and illumination, in addition to having poor generalization performance across images. Extracted buildings were found to be insufficiently accurate for practical applications, making it difficult to automate the extraction of buildings using empirically designed features [9].

It was only through a breakthrough in computer hardware performance that convolutional neural networks (CNNs) [15], which had long been proposed, reappeared on researchers' horizons. Subsequently, with the rapid development of computer vision, opportunities for significant advances in originally limited building extraction methods emerged. The advantage of CNNs [15], [16], [17], [18], [19], [20] is that in addition to extracting shallow features, such as texture, color, and shape through convolution, they can also use their deeper network structures [18], [19], [20] to extract deeper features that are more stable and have stronger representation capabilities. CU-Net [21] introduces multiple constraints based on fully convolutional networks (FCNs) to strengthen the extraction of multiscale features, enhancing the ability of the network to extract buildings and achieve better performance than FCN. Moreover, DE-Net [22] uses a more robust backbone to improve the network's ability to extract features and to achieve better building extraction results. MA-FCN [9] enables multiscale feature extraction based on a feature pyramid network and post-processes the building boundary refinement to achieve good results on the WHU aerial image dataset [23]. MTPA-Net [24] features a scene-driven multitask

Manuscript received 17 July 2023; revised 26 October 2023; accepted 22 December 2023. Date of publication 29 December 2023; date of current version 11 January 2024. This work was supported in part by the National Natural Science foundation of China under Grant 42071443 and in part by the National Key Research and Development Program of China under Grant 2022YFB2602105. *(Corresponding author: Ruisheng Wang.)*

Huaigang Huang is with the Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: hhgteddy@163.com).

Jiabin Liu is with the Faculty of Geography, Yunnan Normal University, Kunming 650500, China (e-mail: zz05231638@163.com).

Ruisheng Wang is with the School of Architecture and Urban Planning, Shenzhen University, Shenzhen 518060, China, and also with the Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: ruiswang@ucalgary.ca).

Digital Object Identifier 10.1109/TGRS.2023.3348102

parallel attention mechanism to improve the network's ability to extract buildings from various scenes. MAP-Net [25] has an improved high-resolution backbone and fuses feature maps at different scales using the channel attention module and the pyramid pooling module to capture the association between feature channels and feature space, effectively improving the building extraction accuracy. MHA-Net [26] introduces multi-path hybrid dilated convolution (HDC) based on MAP-Net to improve the network's ability to recognize buildings at different scales. Further, RSR-Net [27] was motivated by the lightweight backbone network [28] and features a lightweight feature fusion block. With the addition of a large number of SE blocks, the network achieved better performance in building extraction tasks with a small number of parameters and low computational complexity. LRAD-Net [29] enlarges the receptive field of the network by optimizing the atrous spatial pyramid pooling (ASPP) block, which has improved building extraction capability owing to its addition of attention blocks.

Although the abovementioned CNN-based building extraction methods can perform well, they still have the following two limitations.

1) Most building extraction methods were derived from improvements or combinations of upstream computer vision algorithms or tricks, so they lack consideration of building features in network design and are more akin to designing a network to apply to building extraction rather than designing a network for building extraction.
2) The size of the convolutional kernel determines the range of features extracted by the convolutional layer, allowing CNNs to capture a large number of local features. Although CNNs can obtain global features by stacking multiple convolutional layers, the global features extracted from local features and appearing only in the deep layers of the network often lead to a loss of some crucial and effective features [30]. Therefore, limited by the structure of the convolution, CNNs are more powerful in local modeling but less effective in global modeling [31], [32], and the ability of CNN-based networks to recognize buildings may be reduced as a result.

For these two limitations, we provide the corresponding solutions in this article.

1) Observe the building features extracted by the backbone and develop a feature fusion strategy suitable for building extraction.
2) Introduce a transformer structure with strong global modeling capability due to self-attention based on a CNN, thus improving the performance of the network.

According to the above solutions and for the consideration of the practical application requirements of the algorithm, we propose a lightweight network based on the observation of building features called Easy-Net. We conducted experiments comparing Easy-Net, mainstream semantic segmentation networks, and building extraction networks on public datasets and self-made building datasets. The results show the increased extraction accuracy and overall score, considering the operational cost, of Easy-Net compared to other networks. The contributions of this article are as follows.

1) Based on the observation of building features extracted from the backbone network, we propose a simple, effective feature fusion strategy applicable to building extraction. We discard features that are too deep, and intra-class features of different stages are fused first by an add operation. The fused intra class and inter class features are concatenated together and then fused through the feature fusion block.
2) Combining the individual advantages of a CNN and transformer, we propose a feature fusion block CTF that mixes these two network structures to improve the ability of the network to extract buildings.
3) We propose a lightweight building extraction network for remote sensing images: Easy-Net, which can achieve far better performance than other mainstream networks on multiple building datasets, with only a small cost.

## II. Relative Work

### A. Semantic Segmentation

Building extraction from remote sensing images can be seen as a task for pixel-by-pixel classification of images, which is similar to the goal of semantic segmentation in basic computer vision tasks. The difference is that building extraction is a pixel binary classification problem, while most well-known, well-performing semantic segmentation networks [33], [34], [35], [36], [37], [38], [39], [40], [41] are based on large public datasets for pixel multiclassification. Therefore, networks or methods used for semantic segmentation can also be applied to building extraction in a remote sensing image [42], [43], but their performance may not be consistent with that of large semantic segmentation public datasets [44], [45], [46]. This is one of the implications of our study of building extraction networks.

### B. Extraction and Fusion of Features

Semantic segmentation has two main steps: feature extraction and feature fusion. Between them, feature extraction is usually done by the backbone of the semantic segmentation network, which is the only source material for subsequent feature fusion. The backbone generally originates from the upstream classification algorithm [16], [17], [18], [19], [20], [47], [48] and significantly impacts the performance of the segmentation network. As the final output of the classification algorithm loses more detailed information on the features, its performance in the task of semantic segmentation decreases [33]. For this problem, the aggregation of semantic information using the features at each stage in the classification network is an important solution. This information aggregation is also known as feature fusion, which aims to improve the performance of the network on the semantic segmentation task by setting up a reasonable fusion strategy and functional modules to fully exploit the features extracted from the backbone at each stage and multiple scales. U-Net [34] uses skip connections implementing a level-by-level fusion strategy of deep and shallow features, and Deeplabv3+ [36] uses an

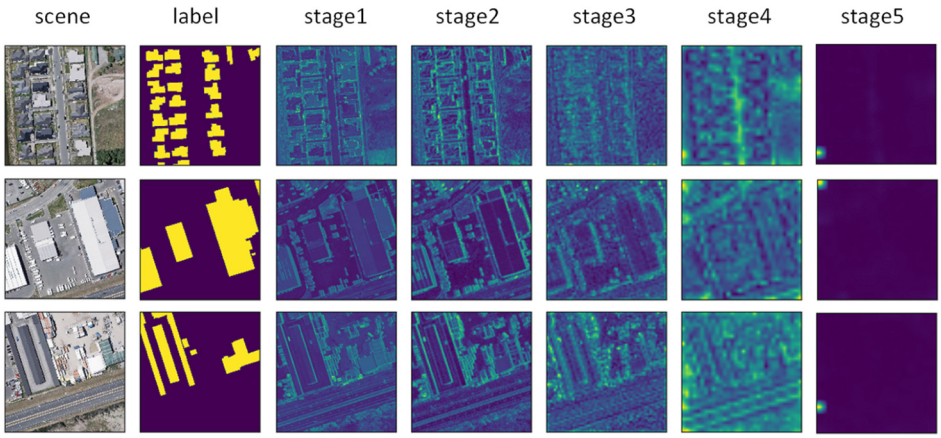


Fig. 1. Baseline features of each stage and feature fusion strategy.

improved ASPP module to enhance the network's ability to aggregate feature information and combine deep and shallow feature fusion strategies. These networks have achieved great success in semantic segmentation tasks.

### C. Transformer

Compared to the limited receptive field of CNN networks, a transformer has a larger receptive field owing to its nonlocal attention structure [41]. ViT [49] attempted to apply the transformer structure to computer vision, and it was successful. Subsequently, a series of networks [50], [51], [52], [53] based on the transformer structure were proposed and applied to various computer vision tasks to tackle the problems of insufficient local modeling capabilities and the high computational cost of the transformer. DERT [54] is an end-to-end network designed for object detection, which is composed entirely of transformer units and without non-maximum suppression (NMS). Segformer [41] uses deep-wise convolution to learn the location information of patches and combines efficient self-attention and a simple multilayer perceptron (MLP) decoder to achieve good performance in semantic segmentation.

### D. Lightweight Networks

Neural network–based algorithms are often limited by the capability of the operating equipment in real-world applications, so the cost of performing the network, including the running time and the memory required to load the network, becomes one of the most important indicators for evaluating the performance of the network in real-world applications. Two efficient convolution modules—deep-wise convolution and point-wise convolution—have enabled MobileNet-series networks [47], [48] to perform well in industrial projects, changing the previous stereotype of CNN networks as bulky and slow. The automatic search networks [28], [55], [56], [57] can obtain a series of networks at different computational complexity by setting up the network base unit and an automatic search of hyperparameters, and practical application personnel can choose a lightweight network or a larger network not only with higher accuracy but also with more complexity according to the needs of the application scenario. BiseNetv2 [36] greatly reduces the model inference time through a lightweight bilateral network structure with multiple channels of shallow detail branches and fewer channels of deep semantic branches, enabling semantic segmentation to make the same real-time inferences as object detection.

## III. Method

This section is divided into two main subsections to introduce Easy-Net and its construction process. In the first subsection, we propose a feature fusion strategy through feature visualization, which is important for Easy-Net, and in the second subsection, we detail the overall network structure of Easy-Net.

### A. Simple, Effective Feature Fusion

*1) Visualization of Building Features Extracted by Backbone:* Feature fusion significantly impacts semantic segmentation performance, so observing the features used for feature fusion is important for proposing useful fusion strategies. We trained baseline network of this article (with RegNet08 [31] as the backbone and U-Net as the fusion strategy) on the WHU satellite image dataset. Then, we fed the building scenes from test set into the trained model and visualized the features extracted by the backbone at each stage. The visualization method we employed involves computing the mean value for each pixel of features along the channel dimension, resulting in a 2-D feature map. Brighter regions on the 2-D feature map indicate higher feature values at those locations, while darker regions indicate lower feature values. The visualization is shown in Fig. 1.

*2) Feature Fusion Strategy:* According to Fig. 1, features extracted by backbone from stage 1 to stage 3 have strong similarity, at which point the network focuses more on the edges of the buildings and we refer to these as intra-class features, and by stage 4, the network pays significantly more attention to features with similar spectral information to the buildings, such as concrete surfaces similar to roads, we refer to these as inter-class features. For the deepest level of the network, stage 5, the features extracted by the network do

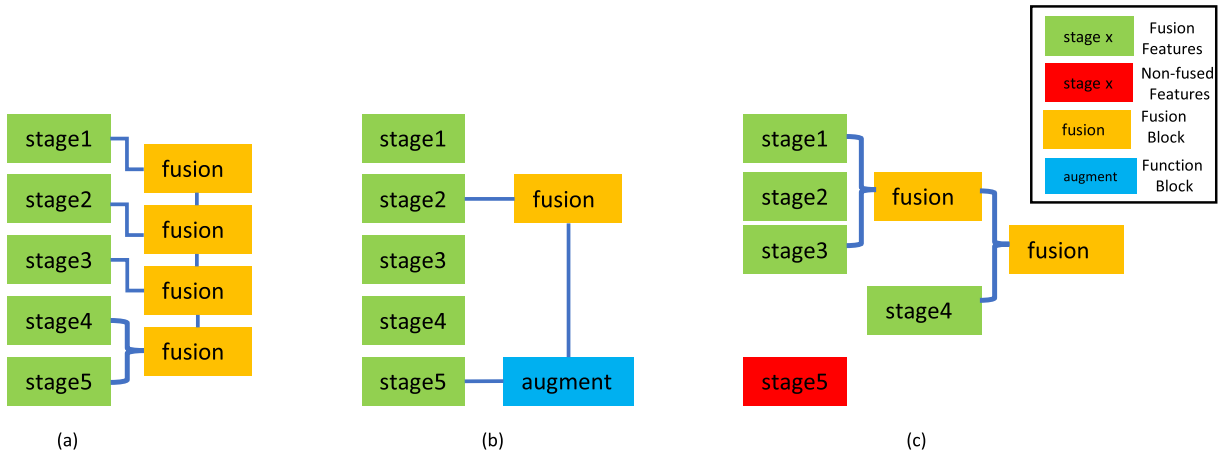


Fig. 2. Feature fusion strategies of different networks. (a) U-Net or FPN. (b) Deeplabv3+. (c) Easy-Net.

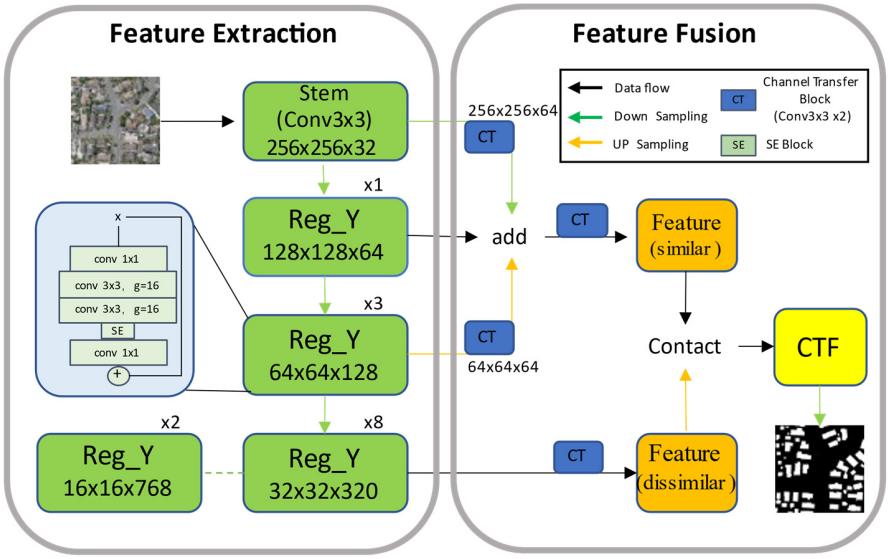


Fig. 3. Network structure of Easy-Net.

not directly observe the relevant properties. Based on the properties of each stage of feature, and combined with the conclusion that performing the fusion of deep features may reduce the accuracy of building extraction [27], we propose a simple, effective feature fusion strategy for building extraction: discard overly deep stage features, fuse stage features that show similar properties first, and then fuse stage features that show different properties, as shown in Fig. 2(c). It should be noted that, unlike the indiscriminate bottom-up feature fusion strategies shown in Fig. 2(a) employed by FPN [59], U-Net, the fusion strategy proposed in this article, is based on observations of the extracted building features from the backbone. Specifically, the intra-class features are fused using an additive operation, and the fused intra-class features are then combined with the inter-class features using a concatenation operation. The fusion strategy of Deeplabv3+ shown in Fig. 2(b) augments the final stage features through functional block similar to ASPP, and then fuses them with the second-stage features. In order to show the effectiveness of the proposed feature fusion strategy in this article, Easy-Net does not use advanced functional block to augment features. For a detailed description, please refer to Section III-B.

### B. Network Structure

The proposed network structure consists of two main parts: feature extraction and feature fusion. According to our proposed fusion strategy, the backbone used for feature extraction discards the last stage compared to the baseline. In addition, the feature fusion part consists of a composite of two types of base units, the feature transfer block and the feature fusion block, as shown in Fig. 3.

*1) Backbone Network:* Based on the consideration of the network operation cost and extraction performance, we choose RegNet08, a lightweight network for image classification tasks, as the baseline for our backbone. Regnet08 is derived from the RegNet series, which is searched by network automatic search methods at different floating-point operations (Flops), while RegNet08 is proven to be good and efficient at building extraction tasks [27]. The original RegNet08 consists

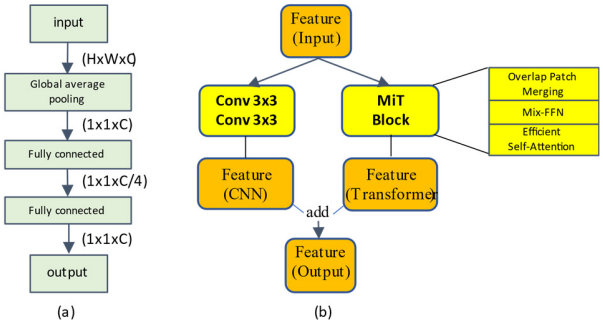


Fig. 4. Network structure of block in Easy-Net. (a) SE block in RegY unit. (b) CTF block.

of five stages (i.e., the green blocks in Fig. 3). The first stage is the stem module, and each remaining stage consists of a different number of RegY units stacked atop each other. The structure of the squeeze-and-excitation (SE) module in the RegY unit is shown in Fig. 4(a). However, according to our proposed feature fusion strategy, we use the modified RegNet08 as our backbone (i.e., discarding the last stage), as shown by the green dashed line in Fig. 3.

*2) Feature Fusion Block Combining the CNN and Transformer:* We use our proposed hybrid block, CNN and transformer fusion (CTF), combining the CNN and transformer in feature fusion, which is divided into two parts, CNN-based fusion, and transformer-based fusion, as shown in Fig. 4(b). The CNN-based fusion part uses the commonly used double-layer $3 \times 3$ convolution, which aims to strengthen the local detail feature fusion in feature fusion. The mix vision transformer (MiT) block in Segformer [38] is used to enhance the global view in feature fusion using self-attention in the block. The structure of the MiT block is shown in Fig. 4(b).

The MiT block uses overlapped patch merging to strengthen the connection between patches, which is achieved by equating the size of the convolution kernel in the convolution to the patch size and then controlling the degree of overlap by stride. The MiT also uses $3 \times 3$ convolution instead of position encoding for learning position information [i.e., Mix-FFN in Fig. 4(b)], which is implemented in the following equation:

$$f_{\text{out}} = \text{MLP}(\text{GELU}(\text{Conv}_{3\times3}(\text{MLP}(f_{\text{in}})))) + f_{\text{in}} \tag{1}$$

where $f_{\text{in}}$ is the input feature, and $f_{\text{out}}$ is the output feature. Note that the self-attentive mechanism in the MiT module performs the sequence reduction mentioned in PVT [50] by setting a hyper-parameter $R$. This operation effectively reduces the high computational cost of self-attentiveness, which has been criticized. In this article, $R$ is set to 4 in the CTF block. The implementation of sequence reduction is shown in the following equation:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}_{\text{head}}}\right)V \tag{2}$$

$$K' = \text{Reshape}\left(\frac{N}{R}, C \cdot R\right)(K) \tag{3}$$

$$K = \text{Linear}(C \cdot R, C)\left(K'\right) \tag{4}$$

where $K$ in (2) and (3) is the sequence to be reduced, and $K$ in (4) is the reduced sequence. $C$ is the dimensional tensor. These improvements allow the MiT block to be more efficient and perform better than the transformer block in ViT [49]. Finally, we combine the two types of feature information fused by CNN fusion and transformer fusion with a concise add operation to obtain a feature fusion block CTF with both short- and long-range modeling capabilities.

*3) Feature Fusion Part:* We designed the feature fusion part of Easy-Net based on the proposed feature fusion strategy, as shown in the right-hand box in Fig. 4. The design idea is simple: the intra-class features of stages 1–3 show similar attributes and are close in spatial location, so we use the add operation to fuse them, ensuring that similar features are only fused in the same channel to increase the amount of information. By fusing them with the add operation, these similar features can complement each other and result in more significant similar features, without increasing the number of channels as the concatenate operation. This in turn helps to prevent an increase in parameter count and larger computational costs. Features with different attributes (i.e., the fused features of stages 1–3 and the inter-class features of stage 4) are different in spatial distribution, so we concatenate them together to preserve as much complete feature information from both sides as possible, and then the fusion is carried out by the feature fusion block (CTF). The concatenate operation only stacks features along the channel dimension without changing the original feature information, which can fully retain the spatial distribution differences. We believe that learning these differences of features can improve the network's performance in recognizing the target, while the add operation will change the feature information within the channel, which to some extent disrupts the spatial distribution differences of the fused features. The highlighted areas in Fig. 1 represent prominent features with larger feature values, but the darker areas, which are non-prominent features, also have feature values. The distributions of the two are not completely consistent in different stages, so directly adding the features may reduce the difference in feature values. In this regard, the features of stages 1–3 are shallow, and the add operation requires the same number of channels. Thus, we use the common two-layer $3 \times 3$ convolution as the channel transfer block to reduce the noise impact of shallow features while completing the conversion of feature channels.

## IV. Experiment

This section presents the design, results, and analysis of our experiments in detail. Section IV-A–IV-D describes the datasets, settings, networks, and evaluation metrics of our experiments. In Section IV-E–IV-H, we show the results of the comparison experiments for each dataset, the ablation experiments with Easy-Net and comparison of different fusion strategies, with relevant analyses.

### A. Datasets

To demonstrate the superiority of our method for building extraction tasks, we compared Easy-Net with other high-performing mainstream networks on a variety of datasets, including four building datasets from two public [23] and two

self-made datasets, covering various sensor types, data sources, resolutions, building cases, and data volume levels. We believe that these widely varying datasets provide a good evaluation of the building extraction ability of networks.

*1) WHU Aerial Image Dataset:* The dataset covers an area of 450 $km^2$ of Christchurch, New Zealand. The production team made high-quality corrections for the numerous labeling errors in the original data. The original image resolution of this dataset was 0.075 m, but the final dataset provided was downsampled to a resolution of 0.3 m, and the entire image was cropped to 8139 512 × 512 images. All images were divided into three sets: the training set, the validation set, and the test set. There were 130 500 buildings in the training area (4736 images), 14 500 buildings in the validation area (1036 images), and 42 000 buildings in the test area (2416 images). It should be noted that, in order to better test the performance of building extraction methods, we have merged the original test set and validation set into a new test set.

*2) WHU Satellite Image Dataset:* This dataset covers an area of 860 $km^2$ in East Asia and was divided into a training set and a test set, consisting of six cropped remote sensing images with a spatial resolution of 0.45 m, which are adjacent to each other and have visible variations in color. As there are more suburban forest areas in the coverage area, which have no buildings or few buildings, the production team screened out the cropped images without buildings, and the remaining images were divided into a training set (3135 images) and test set (903 images), with each image sized at 512 × 512.

*3) Unmanned Aerial Vehicle Image Dataset (Self-Made):* This dataset covers unmanned aerial vehicle (UAV) images of Haizhu District, Guangzhou City, China, in 2012 as the data source. By preprocessing and filtering the image data, we selected 13 high-quality UAV images with various types of ground features, including buildings (single-story houses, multistory houses, schools, irregular buildings), bare ground, water areas, and fields. Each image has a size of 10 401 × 10 401 and a spatial resolution of 0.1 m. We used ArcGIS to create vector building labels, and finally, we cut the images and the produced building labels into 512 × 512 patches, with 3612 patches for the training set and 800 patches for the test set.

*4) University City Dataset (Self-Made):* This dataset contains Google images of University City, Guangzhou, China, in 2018 as the data source. The dataset production process was the same as that of the UAV image dataset. The resolution of this dataset is 0.275 m. We cropped the whole image and the corresponding labels into 851 patches of 512 × 512, 680 for training, and the remaining 171 for testing.

### B. Experimental Settings

Our work was implemented in Pytorch using a single GTX 1660Ti with 6 GB of memory. We used the dice loss function and Adam optimizer (parameter settings remained at the defaults), with an initial learning rate of 0.0001. The image size of the input network during all network training was 512 × 512. The batch size of training was 4. Before each batch of images had entered the network, the data augmentation processes of rotation, flipping, and scaling were carried out randomly. All encoders of the proposed network were initialized with pretraining parameters on ImageNet, and 150 epochs were trained.

### C. Networks for Comparison

To evaluate the capability of Easy-Net, we introduced several mainstream semantic segmentation networks as well as building extraction networks for comparison, including networks with mainstream backbones paired with mainstream feature fusion strategies, lightweight CNN-based or transformer-based semantic segmentation networks, and lightweight building extraction networks.

1) Resnet50 [18], mobilev2 [48], Swin_T [40] with the U-Net feature fusion strategy [34]: denoted as U(res50), U(mv2), and U(Swin-T), respectively. The U-Net feature fusion strategy involves a stage-by-stage contact of different stages of features through skip-connection and uses a decoder block for fusion. Backbone networks are generally used for feature extraction in semantic segmentation and originate from networks for upstream image classification tasks. Resnet50 is a popular CNN-based mainstream backbone network. MobileNetv2 is also a CNN-based network, but its efficient convolutional block design makes it an extremely popular lightweight backbone network with high accuracy. Swin transformer is a series of improved transformer-based backbone networks. The design of windowed sliding slices compensates well for the lack of local modeling capability of transformer-based networks. Owing to the limitation of equipment performance and the fairness of the experiment, we chose the least computationally costly method, Swin-T, for our experiment.
2) Resnet50, mobilenetv2 with Deeplabv3+ feature fusion strategy [36], denoted as D3 + (res50), D3 + (mv2), respectively. The Deeplabv3+ feature fusion strategy aims to use the ASPP module with hole convolution to expand the perceptual field of the network. Thus, it can aggregate more effective feature information, and finally, the fusion is carried out with a decoder module similar to the U-Net shallow features and deep features' contact.
3) *BiseNetv2 [39]:* A CNN-based network for real-time semantic segmentation. The bipartite structure consisting of detail and semantic branches makes it computationally efficient and highly accurate.
4) *Segformer [41]:* Segformer is a series of semantic segmentation networks based on an efficient transformer backbone network and a simple MLP decoder. For the sake of experimental fairness, we only chose Segformer (b0), which has the computational cost closest to Easy-Net, for our experiments, denoted as Seg(b0).
5) *RSR-Net [27]:* This is a lightweight building extraction network based on the improved U-Net feature fusion strategy.
6) *LRAD-Net [29]:* This is a lightweight building extraction network based on the improved Deeplabv3+ feature fusion strategy.

### D. Evaluation Metrics

*1) Accuracy Evaluation Metrics:* In the experiments in this article, we used the following common accuracy evaluation metrics.

1) *IoU:* An important accuracy evaluation metric in semantic segmentation referring to the proportion of the intersection of the pixels predicted by the network as buildings and the pixels of the real buildings to their merged sets.
2) *Recall:* The proportion of the number of pixels correctly identified by the network as buildings to the total number of pixels of real buildings.
3) *Precision:* The proportion of the pixels identified by the network as building pixels that are actually building pixels.
4) *F1 Score:* Also known as the balanced score, defined as the harmonized mean of the recall and precision and related to recall and precision, as shown in the following equation:

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{5}$$

*2) Computational Cost Evaluation Metrics:* The operational cost is critical in measuring the performance of the network in practice. The following metrics of the operational cost were used in this work.

1) *Number of Parameters:* Since the data type of all experimental network parameters in this article is float32 (i.e., each parameter needs to occupy four bytes), the memory required to load the network is four times the number of parameters.
2) *Single Image Inference Speed (CPU):* In this article, this is the inference speed of each image of size 512 × 512 on the i5 8500 CPU.
3) *Single Image Inference Speed (GPU):* In this article, this is the inference speed of each image of size 512 × 512 on a GTX1660Ti GPU.
4) *FLOPs:* Floating-point operations per second, which can be used to measure the complexity of algorithms or models.

*3) Comprehensive Evaluation Metric:* To evaluate the performance of the network comprehensively, we designed a comprehensive evaluation metric called the relative comprehensive score (RCS). We calculated the scores of all the networks in this article in four metrics: IoU (the most important accuracy evaluation metric), number of parameters, CPU inference speed, and GPU inference speed. The scoring rule is 100 for the best performance and 0 for the worst. The rest is interpolated in proportion to the difference, resulting in a relative score based only on the networks used in the experiments. The RCS is obtained by multiplying each score with its assigned weight, and based on our experience, the weights of the RCS in this article are expressed below

$$\text{RCS} = 0.5\text{iou} + 0.2\text{param} + 0.15\text{cpu} + 0.15\text{gpu}. \tag{6}$$

Note that we did not use common FLOPs in RCS. This is because a related study [58] indicated that FLOPs do not consider access memory consumption and computational parallelism and operations such as Add and Contact, which do not require parameters, can cause significant access memory consumption and lead to longer inference time consumption. However, these costs cannot be reflected in FLOPs.

### E. Comparison Experiments

On each of the four building datasets mentioned in Section IV-A with their own characteristics, we conducted comparison experiments between Easy-Net and the networks mentioned in Section IV-C, and we recorded the performance of the networks using the evaluation metrics in Section IV-D. The following is a presentation and analysis of our experimental results for these networks on each dataset.

*1) Experiments on the WHU Aerial Image Dataset:* This dataset has the advantages of a large number of buildings, rich building types, and good labeling quality, and it can be a good test of the comprehensive capability of each building extraction network. The experimental results for each network on this dataset are shown in Table I.

It is important to note that the performance of the CPU used in our experiments, i5 8500, is only approximately one-sixth of the performance of the i9 13900k (one of the high-performance CPUs currently available). Additionally, the GPU used, 1660ti, has a single-precision computational power of only 5.5TFLOPS, whereas one of the highest-performance GPUs currently available, the GTX 4090, has a single-precision computational power of 100TFLOPS. Therefore, we should acknowledge that the values in Table I regarding network inference time may differ significantly from the results reported in the corresponding published papers. This discrepancy is primarily due to the limitations of our experimental equipment rather than the inherent capabilities of the network itself. However, in order to ensure fairness, all of our data were obtained using the same device throughout the experiments.

It can be seen that BisegNetv2, which is known for its speed, has a significantly higher inference speed than other networks. However, a significant gap exists between its important accuracy evaluation metric (IoU) and that of the other networks, indicating its lack of building extraction ability. From the experimental results of U(res50), U(mv2), D3 + (res50), and D3 + (mv2), for the U-Net fusion strategy, the gap in extraction capability between the lightweight backbone network and the backbone network with a larger number of parameters is not large (IoU gap of 0.1), and conversely, a gap exists in the extraction capability between the lightweight backbone network and the backbone network with a larger number of parameters with the D3+ fusion strategy (IoU difference of 0.73). However, the use of the lightweight backbone network has obvious advantages in terms of the number of parameters and inference speed, with U(mv2) and D3 + (mv2) having only about one-sixth of the number of parameters of U(res50) and D3 + (res50). Moreover, the CPU inference speed is nearly twice as fast. U(Swin-T) has a strong building extraction capability, achieving the highest IoU and Recall, 89.16% and 94.2%, respectively. However, because the transformer structure is more computationally costly than the CNN and Swin-T is built from multiple transformers, U(Swin-T) has

TABLE I
COMPARISON OF DIFFERENT NETWORKS AND EASY-NET ON WHU AERIAL IMAGE DATASET

| Method | IoU (%) | Recall (%) | Precision (%) | F1 Score | Parameters (M) | CPU Infer (MS) | GPU Infer (MS) | FLOPs(G) | RC Score |
|---|---|---|---|---|---|---|---|---|---|
| U5(res50) | 87.57 | 93.67 | 93.51 | 93.59 | 32.5 | 1001 | 58.5 | 42.7 | 35 |
| D3+(res50) | 87.9 | 93.1 | 94.09 | 93.59 | 26.7 | 980 | 59.3 | 36.8 | 44 |
| U5(mv2) | 87.67 | 93.33 | 93.43 | 93.38 | 6.6 | 550 | 55 | 13.5 | 65 |
| D3+(mv2) | 87.17 | 92.07 | 94.21 | 93.13 | 4.4 | 500 | 55 | 6.2 | 59 |
| U5(Swin_T) | **89.16** | **94.2** | 94.21 | **94.20** | 31.6 | 1142 | 72.6 | 39.1 | 51 |
| SegFormer | 87.56 | 92.31 | 94.27 | 93.28 | 3.7 | 585 | 54 | 6.8 | 65 |
| BiseNetV2 | 85.94 | 92.54 | 92.45 | 92.49 | 5.2 | **344** | **44** | 17.7 | 48 |
| LRAD-Net | 87.67 | 92.78 | 94.03 | 93.40 | 7.1 | 599 | 55 | 13.4 | 63 |
| RSR-Net | 88.09 | 93.16 | 93.97 | 93.56 | **2.9** | 521 | 54 | **6.0** | 75 |
| **Easy-Net**(ours) | **89.16** | 93.48 | **94.94** | **94.20** | 3.9 | 583 | 53 | 17.3 | **90** |

the slowest inference speed among these networks. Overall, Easy-Net and U(Swin-T) are comparable and significantly better than the other networks in building extraction, with the highest IoU of 89.16%. In addition, Easy-Net achieves the highest precision and F1 scores. In terms of application performance, Easy-Net's number of parameters and inference speed are both good, so it scores the highest (90) in the RCS; this indicates the high overall capability of the network and its ability to clearly outperform other networks.

The above is our analysis of the results of the experiments in Table I, reflecting the general performance of networks on the WHU aerial image dataset. To determine the reason for this result, we visualized representative building scenes taken from the test set, as shown in Fig. 5, and we used them to develop the analysis again to further understand the building extraction capability of each network. In Fig. 5, three types of buildings are shown [i.e., dense buildings, large buildings, and sparse buildings, with a total of six scenes (two in each type)]. The yellow area represents the range of correctly predicted pixels, the red area represents the range of incorrectly predicted pixels, the orange area represents the range of missed predictions, and the IoU of the current scene prediction is marked in the upper right corner of each scene.

1) *Dense Building Scenes:* For the U-Net fusion strategy, no difference exists between the performance of the lightweight backbone and the backbone with a larger number of parameters [i.e., U(mv2) and U(res50)], but U(Swin-T) with a larger number of parameters and a backbone network constructed from a pure transformer visibly outperforms them. However, for the Deeplabv3+ fusion strategy, a clear difference exists in performance between the lightweight backbone and the backbone with a larger number of parameters, with D3 + (mv2) performing significantly worse than D3 + (res50). RSR-Net and LRAD-Net are lightweight backbone networks combined with the improved U-Net fusion strategy and the D3+ fusion strategy, respectively, and their performance in this scenario is also consistent with our analysis above. The performance of Easy-Net is as good as U(Swin-T), both of which are better than other networks.
2) *Large Building Scenes:* The networks are comparable in their ability to extract large buildings, except for BiseNetv2, which produces some discontinuous holes in the second scene owing to omitted predictions. In addition, some interesting phenomena are seen in the first scenario. Containers are sometimes found near large buildings in factories for storing raw materials or manufactured goods, as shown in the red box, and they are not part of the building but look similar to it. Thus, distinguishing between containers and buildings is challenging for networks. In this challenge, U(mv2) and D3 + (mv2), using a lightweight CNN backbone network, discriminate containers much more poorly than U(res50) and D3 + (res50) paired with a larger parametric backbone network. Moreover, the Deeplabv3+ fusion strategy also performs better than the U-Net fusion strategy. Notably, Segformer (b0), consisting of only transformer and MLP, has the best performance in this challenge, indicating that the combination of the transformer and MLP can better overcome the challenges posed by containers than the CNN network, given the same parameter magnitude. However, Easy-Net has the highest accuracy for container recognition when using a lightweight CNN backbone network. In addition, BiseNetv2 and U(Swin-T), as well as Segformer (b0) with transformer as the backbone network, are better at recognizing elongated buildings than the other networks. Still, this ability seems to limit the network's ability to distinguish between building edges and shadows, as shown in the green box. This may be because of the elongated shape of the building edges and shadows or the fact that the shadows are covered by impervious surfaces of a similar color and texture. Easy-Net is poorer than the three networks mentioned above at recognizing elongated buildings, but its ability to distinguish between building edges and shadows is superior to theirs.
3) *Sparse Building Scenes:* All networks can accurately identify the approximate location of buildings, but the small percentage of pixels in buildings leads to large fluctuations in IoU owing to subtle differences in recognition. However, Easy-Net is still better at recognizing the edges of buildings than other networks. For example, in the enlarged building pointed at by the arrow in scene 2, although the edge recognition of RSR-Net and U(Swin-T) is at the level of almost only a red line of

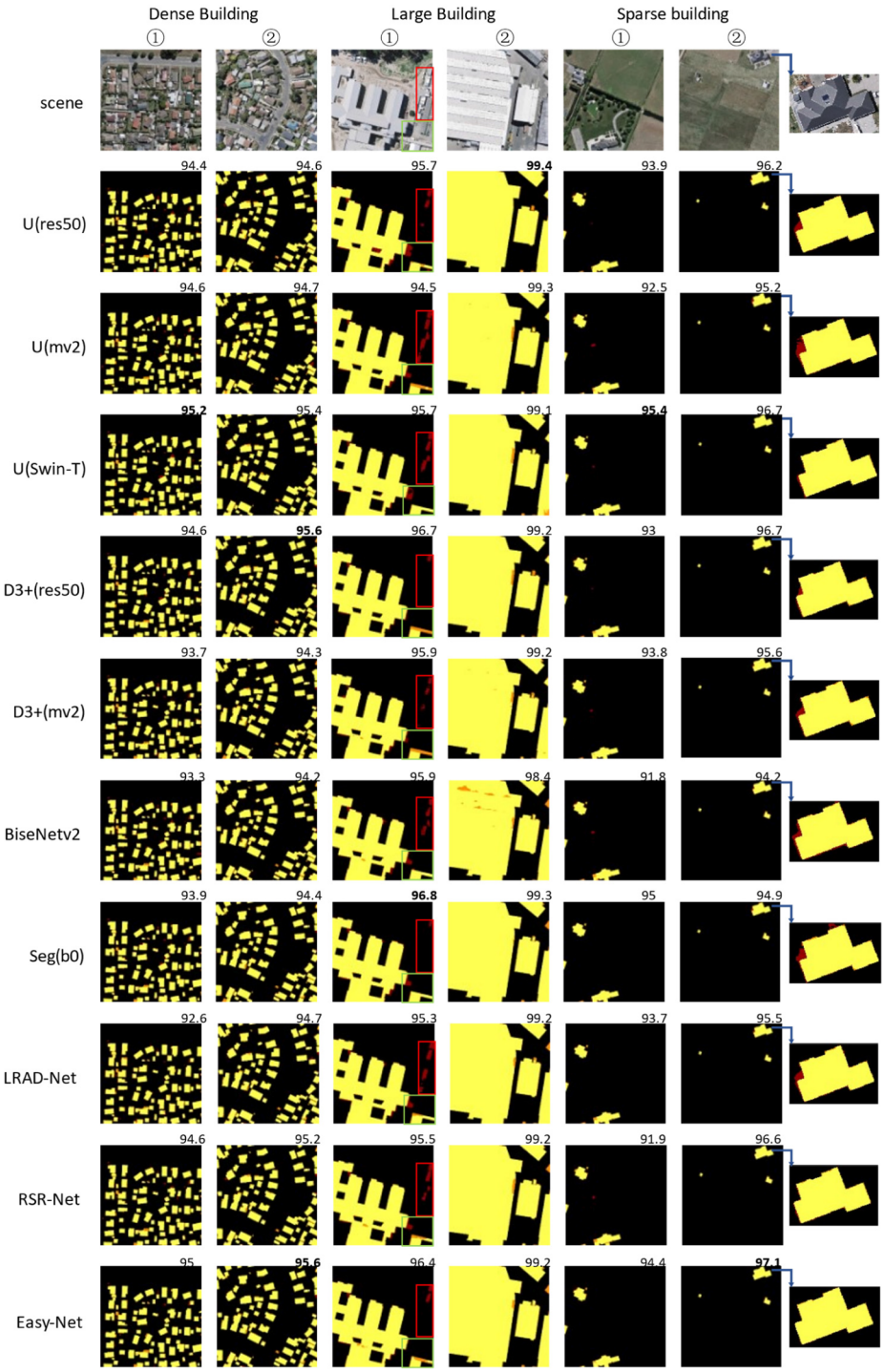


Fig. 5. Performance of networks on WHU aerial image dataset.

false predictions, Easy-Net is accurate to the extent that the edges almost completely overlap.

*2) Experiments on the WHU Satellite Image Dataset:* This dataset is characterized by sparse buildings and significant color differences due to different data sources, which is a real situation commonly arising in remote sensing image mosaic applications. Thus, it is a good evaluation of the network's ability to extract buildings from different data sources but with similar styles in the same geographical area. The experimental results of each network on this dataset are shown in Table II.

As the input image size remains the same, the number of parameters and the inference speed of the network do not change, and Table II does not repeat what is listed and repeated in Table I. From Table II, the lightweight building extraction networks LRAD-Net, RSR-Net, and Easy-Net have relatively obvious advantages over the rest of the networks, and Easy-Net is the best performer among these three, achieving

TABLE II
COMPARISON OF DIFFERENT NETWORKS AND EASY-NET ON WHU AERIAL IMAGE DATASET

| Method | IoU (%) | Recall (%) | Precision (%) | F1 Score | RC Score |
|---|---|---|---|---|---|
| U5(res50) | 66.12 | 76.66 | 83.98 | 80.15 | 10 |
| D3+(res50) | 68.64 | 79.43 | **84.76** | 82.01 | 34 |
| U5(mv2) | 66.97 | 77.55 | 84.22 | 80.75 | 45 |
| D3+(mv2) | 66.9 | 79.27 | 82.3 | 80.76 | 46 |
| U5(Swin_T) | 66.77 | 79.57 | 82.32 | 80.92 | 6 |
| SegFormer | 69.78 | 81.38 | 83.61 | 82.48 | 68 |
| BiseNetV2 | 67.49 | 79.8 | 82.2 | 80.98 | 59 |
| LRAD-Net | 71.38 | 83.08 | 83.89 | 83.48 | 78 |
| RSR-Net | 71.57 | 83.6 | 83.53 | 83.56 | 84 |
| **Easy-Net** (ours) | **72.5** | **84.21** | 84.31 | **84.26** | **90** |

the highest IoU, recall, F1, and RCS of all networks, at 72.5, 84.21, 86, and 90, respectively. In the visual presentation in Fig. 6, two representative types of scenes are shown. One type is three scenes with normal color, which include one scene with relatively dense buildings and two scenes with sparse buildings. The other type is the scenes with abnormal color, which include two scenes with sparse buildings and a special sparse scene with sparse buildings at the junction of different data sources.

As shown in Fig. 6, in scenes with normal color, the performance is similar to the network's performance in the WHU aerial image dataset, with the exception of D3 + (mv2) in the first scene. Here, the IoU is much lower than in the other networks because of the incorrect prediction of the road as a building, and all scenes in the aerial dataset are of normal color. However, the result is different in the color anomaly scenes. For U(res50), D3 + (res50), and U5(Swin-T), the networks with the larger number of parameters show significantly worse performance than the rest of the lightweight networks. In particular, U(Swin-T) drops from good performance in the color normal scenes to the worst; this is barely effective in identifying the color anomalies in the buildings. In contrast, Easy-Net shows strong building extraction ability in each scene, especially in the last scene at the junction of different data sources, where it far outperforms the other networks. The result demonstrates that Easy-Net has good robustness in building extraction for remote sensing images stitched with different data sources.

*3) Experiments on the UAV Image Dataset:* This dataset is characterized by high resolution and also covers a large area of a particular scenario, urban villages, which are densely populated with residential buildings. Therefore, this dataset can be a good evaluation of the network's ability to extract extremely dense buildings in high-resolution remote sensing imagery. The experimental results of the various networks on this dataset are shown in Table III.

As can be seen from the table, the Deeplabv3+ fusion strategy has an advantage over the U-Net fusion strategy in processing high-resolution remote sensing images, with resolutions up to 0.1 m. This phenomenon is also reflected in RSR-Net and LRAD-Net, where the IoU of RSR-Net based

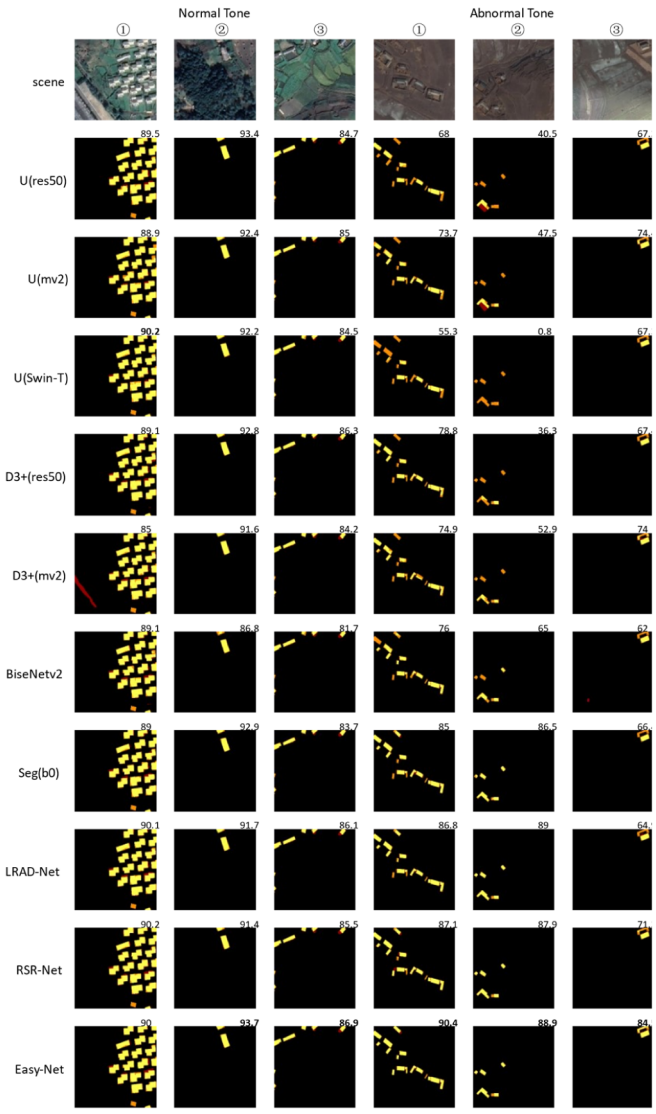


Fig. 6. Performance of networks on WHU satellite image dataset.

TABLE III
COMPARISON OF DIFFERENT NETWORKS AND EASY-NET ON UAV DATASET

| Method | IoU (%) | Recall (%) | Precision (%) | F1 Score | RC Score |
|---|---|---|---|---|---|
| U5(res50) | 77.55 | 88.06 | 86.78 | 87.42 | 25 |
| D3+(res50) | 78.73 | 88.76 | 87.64 | 88.20 | 49 |
| U5(mv2) | 77.52 | **88.83** | 86.48 | 87.64 | 53 |
| D3+(mv2) | 78.38 | 88.63 | 87.59 | 88.11 | 69 |
| U5(Swin_T) | 78.5 | 87.63 | 88.46 | 88.04 | 32 |
| SegFormer | 78 | 86.84 | 89.06 | 87.94 | 62 |
| BiseNetV2 | 76.64 | 87.82 | 86.27 | 87.04 | 48 |
| LRAD-Net | 79.42 | 88.72 | 88.97 | 88.84 | 83 |
| RSR-Net | 78.9 | 88.34 | 88.71 | 88.52 | 79 |
| **Easy-Net** (ours) | **79.64** | 88.47 | **89.66** | **89.06** | **90** |

on the U-Net optimization is 78.9%, which is lower than that of LRAD-Net, 79.42%. In addition, similar to the WHU satellite image dataset, the lightweight building extraction networks LRAD-Net, RSR-Net, and Easy-Net have better

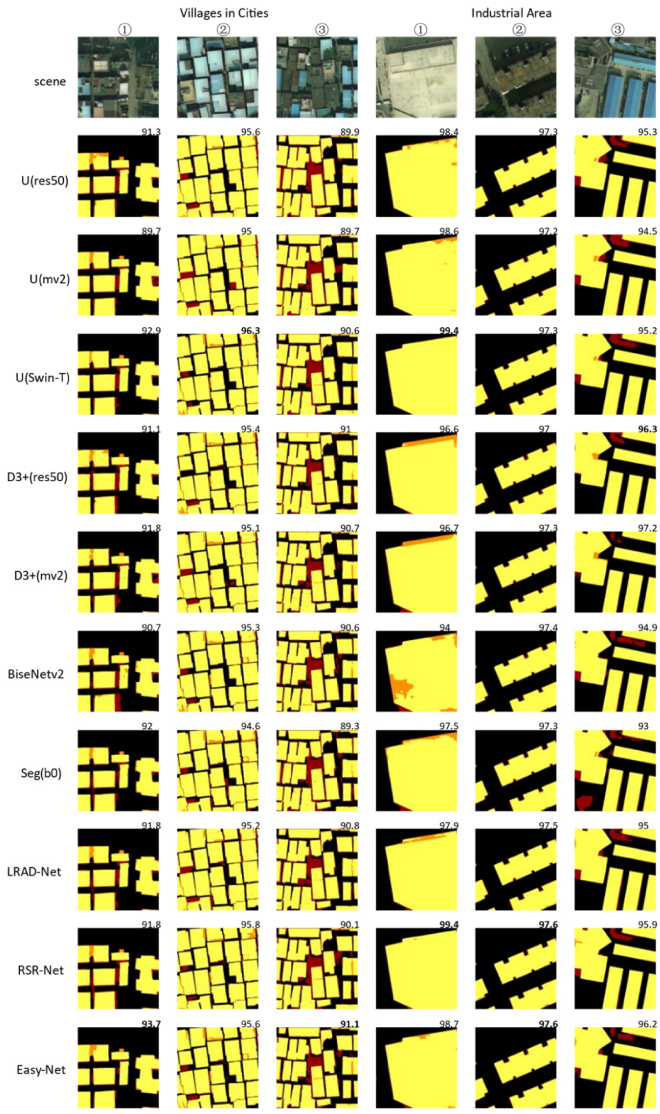


Fig. 7. Performance of networks on UAV dataset.

performance compared to the rest of the networks. Easy-Net is the best performer among these three, achieving the highest IoU, precision, F1, and RCS among all networks, at 79.64%, 89.66%, 84.89%, and 90, respectively.

In the visual presentation in Fig. 7, we show two representative types of scenarios. One type is the urban village scene with three scenes, and the other is the industrial area scene, also with three scenes. In the urban village scenes, two difficulties arise in building extraction: 1) how to distinguish the narrow roads between buildings and 2) how to distinguish hidden open spaces hidden between buildings from the actual buildings. As can be observed from the urban village scene 1, Easy-Net performs best in the challenge of distinguishing narrow roads from buildings, as shown in the red box. However, for the challenge of distinguishing open space from buildings in a dense cluster of buildings, it can be seen from scenes 2 and 3 that no network can identify the open space in the green box. This is because remote sensing images do not carry height information, while the spectral information of the open space is almost identical to that of the buildings. In the industrial area scenes, for scene 1, where the buildings cover almost the whole scene, both U(Swin_T) and RSR-Net can ensure the integrity of the buildings well. In contrast, BiseNetv2 performs the worst. Easy-Net's performance is not the best in each of these six scenes, but it is at a high level, which is why it has the best performance, as shown in Table III.

TABLE IV
COMPARISON OF DIFFERENT NETWORKS AND EASY-NET ON UNIVERSITY DATASET

| Method | IoU (%) | Recall (%) | Precision (%) | F1 Score | RC Score |
|---|---|---|---|---|---|
| U5(res50) | 77.56 | **86.87** | 85.06 | 85.96 | 60 |
| D3+(res50) | 75.79 | 82.61 | 88.97 | 85.67 | 57 |
| U5(mv2) | 75.71 | 84.99 | 84.88 | 84.93 | 81 |
| D3+(mv2) | 75.49 | 84.01 | 86.03 | 85.01 | 82 |
| U5(Swin_T) | 75.06 | 87.8 | 83.68 | 85.69 | 41 |
| SegFormer | 73.31 | 82.88 | 87.41 | 85.08 | 74 |
| BiseNetV2 | 63.96 | 90.77 | 69.28 | 78.58 | 48 |
| LRAD-Net | 74.81 | 84.25 | 86.29 | 85.26 | 76 |
| RSR-Net | 76.78 | 84.3 | 86.86 | 85.56 | 88 |
| **Easy-Net** (ours) | **77.7** | 84.84 | **89.02** | **86.88** | **90** |

*4) Experiments on the University City Dataset:* This dataset is characterized by a small amount of data, enabling the evaluation of the building extraction ability of the network in real-world situations where only a small number of samples are available. The experimental results of the various networks on this dataset are shown in Table IV.

As can be seen from the table, the performance of BiseNetv2 is significantly lower when the data volume is small, with an IoU of only 63.96%. In addition, the U-Net fusion strategy can cope with the challenge of a small data volume better than the Deeplabv3+ fusion strategy. Easy-Net is still the best performer, achieving the highest IoU, precision, and RCS of all networks. Owing to the small sample size and single scenario in this dataset, we only randomly selected six scenarios in the visual presentation to show Fig. 8. It can be seen that BiseNetv2 has a lot of false predictions in each scene. In contrast, Easy-Net can still perform well in each scene, with only a small number of samples, especially in scenes 5 and 6. It shows a strong ability to distinguish between buildings and adjacent impermeable ground.

### F. Practical Application Performance Evaluation

In the practical application of neural network algorithms, we must consider both application performance metrics, such as computational cost, as well as generalization performance, which is highly valued because it reflects the stability of the algorithm. Therefore, we evaluated the performance of all networks in practice based on the four abovementioned datasets and the RCS of each network. We also added generalization performance to this evaluation. We calculated the $RCS_{mean}$ and standard deviation of IoU based on the RCS and $Score_{iou}$ for each network on these four datasets, and then we obtained the actual application performance score (APS) based on these

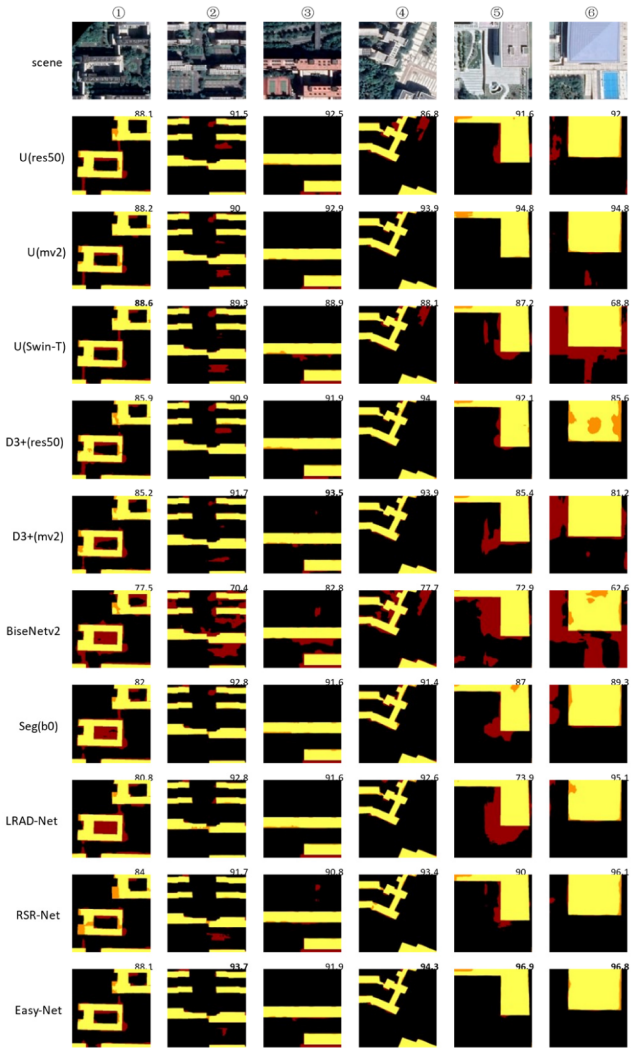


Fig. 8. Performance of networks on University Town Dataset.

two calculations. The relevant equations are given below

$$\mathrm{RCS}_{\mathrm{mean}} = \frac{\mathrm{RCS}_1 + \mathrm{RCS}_2 + \mathrm{RCS}_3 + \mathrm{RCS}_4}{4} \tag{7}$$

$$\sigma_{\mathrm{iou}} = \sqrt{\frac{\sum_{i=1}^{4}(\mathrm{Score}_i - \mathrm{Score}_m)^2}{4}} \tag{8}$$

$$\mathrm{APS}_i = \mathrm{RCS}_{\mathrm{mean}i} \times \frac{\mathrm{Sum}(\sigma_{\mathrm{iou}1}, \sigma_{\mathrm{iou}2}, \ldots) - \sigma_{\mathrm{iou}i}}{\mathrm{Sum}(\sigma_{\mathrm{iou}1}, \sigma_{\mathrm{iou}2}, \ldots)}. \tag{9}$$

We then plotted Fig. 9 based on the various data obtained. As shown, the lightweight building extraction networks LRAD-Net, RSR-Net, and Easy-Net outperform the other networks in practice. Easy-Net also performs significantly better than the other two networks.

### G. Ablation Experiments

The comparative experiments in the previous subsection proved the advantages of Easy-Net over other networks. To investigate the mechanism underlying these advantages, we performed an ablation experiment based on the WHU aerial image dataset, evolving from baseline to Easy-Net. At the same time, we carefully observed and summarized

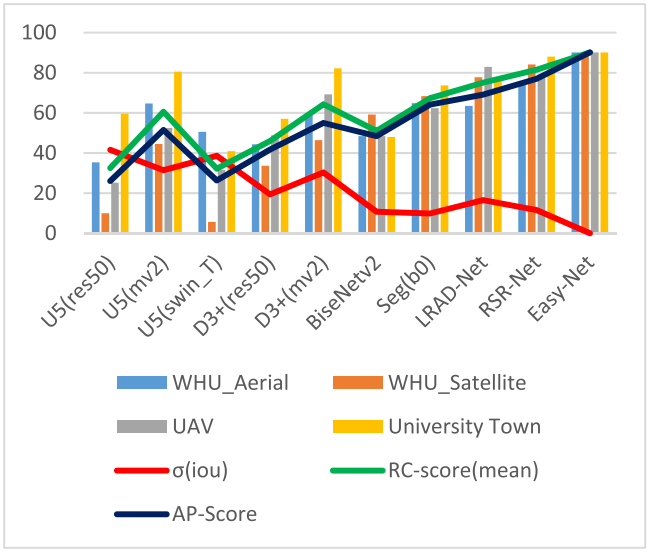


Fig. 9. Application performance of networks.

TABLE V
ABLATION EXPERIMENT OF EASY-NET ON WHU AERIAL IMAGE DATASET

| Method | IoU (%) | Recall (%) | Precision (%) | F1 Score | Parameters (M) |
|---|---|---|---|---|---|
| Baseline | 87.9 | 93.52 | 93.52 | 93.52 | 7.49 |
| Baseline+S1 | 87.96 | 92.91 | 94.11 | 93.51 | 3.42 |
| Baseline+S1+S2 | 88.67 | 92.82 | **95.1** | 93.95 | 3.68 |
| Baseline+S1+S2+CTF **(Easy-Net)** | **89.16** | **93.48** | 94.94 | **94.20** | 3.9 |

the impact and contribution of each part of our method, aiming to inspire better building extraction methods in the future.

In the ablation experiment, our baseline was U(reg08) (i.e., the backbone network being the complete RefNet08), paired with the U-Net fusion strategy. The first step was to discard features that were too deep, denoted as Baseline + S1. The second step was to fuse similar features with dissimilar features, denoted as Baseline + S1 + S2. Note that the unit used for fusion is the most common double-layer 3 × 3 convolution. Finally, the CTF module was used to replace the double-layer 3 × 3 convolution module used to fuse the dissimilar features, which is denoted as Baseline + S1 + S2 + CTF (i.e., the proposed Easy-Net).

The experimental results are shown in Table V. The results show that each step of our method can effectively improve the building extraction ability of the network. Then, combining these results with the visualization results of the selected scenes in Fig. 10, we performed an analysis of the mechanism underlying each step of the improvement. It can be seen that the baseline's ability to extract dense small buildings similar to scene 1 decreases after discarding features that are too deep. However, this also significantly reduces the number of false predictions in scenes with no buildings but only neat textures, as in scene 2.

The result of these two offsetting each other is that there is only an extremely slight rise in baseline + S1 over the baseline on IoU. However, a lighter network with only 45%

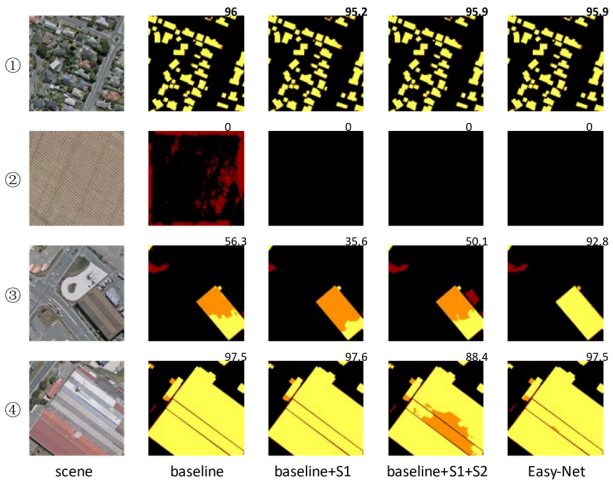


Fig. 10. Performance of Easy-Net in the ablation experiment.

parameters compared to the baseline is obtained because the deeper network used to extract this part of the features is also discarded, which has important implications for reducing the computational cost. After the second step of the fusion strategy, we can see that baseline + S1 + S2 reduces the impact of the first step well, bringing the network back to the same level of extraction capability for dense small buildings as done for the baseline, with an improvement of 0.71% in IoU over that at the first step. However, throughout the process from the baseline to the fusion strategy in this article, the network does not satisfactorily identify large buildings, as shown in scenes 3 and 4. Even in scene 4, baseline + S1 and S2 are the poorest, which also shows that the proposed fusion strategy not only improves the recognition of dense small buildings but also reduces the recognition of large buildings to a certain extent. However, this changes after using the CTF block. At this point, our proposed network Easy-Net can recognize the complete large buildings well, and the IoU on the whole dataset rises by 0.49%. We infer that this is because the global perceptual field of self-attention reinforces the network's focus on the connections between pixels on a large scale, and that the powerful local modeling capability of the CNN in the CTF block also ensures that the network has a strong extraction capability for small buildings. As can be seen, the fusion strategy and the proposed CTF block complement each other well, which is the main reason why Easy-Net performs well in various scenes across datasets.

### *H. Comparison of Different Feature Fusion Strategies*

In this subsection, we compared the performance of Easy-Net and other fusion strategies' networks on WHU aerial image dataset. The backbone of all networks is Regnet08, and the fusion strategies are FPN, U-Net, Deeplabv3+, and Easy-Net, respectively. The experimental results are shown in Table VI.

We can observe that when using the same lightweight feature extraction network, U-Net performs better than Deeplabv3+, which aligns with the results obtained in Section IV-E of our experiments. However, when using the same bottom-up feature fusion approach, FPN and U-Net exhibit differences. We attribute this disparity to the fact that, under the condition of indiscriminate fusion, the add operation used in FPN tends to compromise more valuable feature information compared to the Contact operation used in U-Net. Easy-Net, which performs the best, is based on the observation of building features. It utilizes the add fusion of intra-class features and the fusion of concatenated intra-class features and inter-class features through CTF. This approach allows Easy-Net to effectively leverage the features extracted by the feature extraction network, resulting in superior performance in building extraction tasks.

TABLE VI
DIFFERENT FUSION STRATEGIES ON WHU AERIAL IMAGE DATASET

| Method | IoU (%) | Recall (%) | Precision (%) | F1 Score |
|---|---|---|---|---|
| Deeplabv3+ | 87.34 | 92.91 | 93.42 | 93.16 |
| FPN | 87.35 | 92.97 | 93.49 | 93.25 |
| U-Net | 87.9 | 93.52 | 93.52 | 93.52 |
| **Easy-Net** | **89.16** | **93.48** | **94.94** | **94.20** |

## V. CONCLUSION

In this article, we proposed a novel, simple, and effective feature fusion strategy based on the observation of features extracted from each stage of the backbone network. We used the strategy to construct a lightweight network for building extraction from remote sensing images: Easy-Net. In Easy-Net, we incorporated a feature fusion block (CTF) that combined the short-range modeling capability of a CNN with the long-range modeling capability of a transformer, effectively enhancing the feature fusion capability of the network. To validate the effectiveness of our method, we conducted experiments comparing Easy-Net with several mainstream semantic segmentation networks and excellent lightweight building extraction networks on four building datasets, including both public and self-made datasets, even under limited equipment conditions. The results, based on a comprehensive evaluation, showed the superior performance of Easy-Net compared to these networks. Finally, the results of the ablation experiments and comparison of different feature fusion strategies also showed the effectiveness of our proposed feature fusion strategy and CTF.

## REFERENCES

[1] M. Awrangjeb, C. Zhang, and C. S. Fraser, "Improved building detection using texture information," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 38, pp. 143–148, Apr. 2011.

[2] M. Cote and P. Saeedi, "Automatic rooftop extraction in nadir aerial imagery of suburban regions using corners and variational level set evolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 313–328, Jan. 2013.

[3] J.-P. Burochin, B. Vallet, M. Brédif, C. Mallet, T. Brosset, and N. Paparoditis, "Detecting blind building façades from highly overlapping wide angle aerial imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 96, pp. 193–209, Oct. 2014.

[4] G. Zhou and X. Zhou, "Seamless fusion of LiDAR and aerial imagery for building extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7393–7407, Nov. 2014.

[5] Z. Li, W. Shi, Q. Wang, and Z. Miao, "Extracting man-made objects from high spatial resolution remote sensing images via fast level set evolutions," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 883–899, Feb. 2015.

[6] D. Chaudhuri, N. K. Kushwaha, A. Samal, and R. C. Agarwal, "Automatic building detection from high-resolution satellite images based on morphology and internal gray variance," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 5, pp. 1767–1779, May 2016.

[7] N. L. Gavankar and S. K. Ghosh, "Automatic building footprint extraction from high-resolution satellite image using mathematical morphology," *Eur. J. Remote Sens.*, vol. 51, no. 1, pp. 182–193, Jan. 2018.

[8] S. Xia and R. Wang, "Extraction of residential building instances in suburban areas from mobile LiDAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 144, pp. 453–468, Oct. 2018.

[9] 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.

[10] Y. Zhang, "Optimisation of building detection in satellite images by combining multispectral classification and texture filtering," *ISPRS J. Photogramm. Remote Sens.*, vol. 54, no. 1, pp. 50–60, Feb. 1999.

[11] C. Lin and R. Nevatia, "Building detection and description from a single intensity image," *Comput. Vis. Image Understand.*, vol. 72, no. 2, pp. 101–121, Nov. 1998.

[12] J. Inglada, "Automatic recognition of man-made objects in high resolution optical remote sensing images by SVM classification of geometric image features," *ISPRS J. Photogramm. Remote Sens.*, vol. 62, no. 3, pp. 236–248, Aug. 2007.

[13] Y. Meng and S. Peng, "Object-oriented building extraction from high-resolution imagery based on fuzzy SVM," in *Proc. Int. Conf. Inf. Eng. Comput. Sci.*, Dec. 2009, pp. 1–6.

[14] S. Du, F. Zhang, and X. Zhang, "Semantic classification of urban buildings combining VHR image and GIS data: An improved random forest approach," *ISPRS J. Photogramm. Remote Sens.*, vol. 105, pp. 107–119, Jul. 2015.

[15] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, vol. 25, 2012, pp. 1097–1105.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. ECCV*, 2016, pp. 630–645.

[20] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5987–5995.

[21] G. Wu et al., "Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks," *Remote Sens.*, vol. 10, no. 3, p. 407, Mar. 2018.

[22] H. Liu et al., "DE-Net: Deep encoding network for building extraction from high-resolution remote sensing imagery," *Remote Sens.*, vol. 11, no. 20, p. 2380, Oct. 2019.

[23] 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.

[24] 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.

[25] Q. Zhu, C. Liao, H. Hu, X. Mei, and H. Li, "MAP-Net: Multiple attending path neural network for building footprint extraction from remote sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 6169–6181, Jul. 2021.

[26] J. Cai and Y. Chen, "MHA-Net: Multipath hybrid attention network for building footprint extraction from high-resolution remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5807–5817, 2021.

[27] H. Huang, Y. Chen, and R. Wang, "A lightweight network for building extraction from remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Jan. 2022.

[28] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10425–10433.

[29] J. Liu, H. Huang, H. Sun, Z. Wu, and R. Luo, "LRAD-Net: An improved lightweight network for building extraction from remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 675–687, 2023.

[30] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, "Do vision transformers see like convolutional neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 12116–12128.

[31] Y. Zhou et al., "BOMSC-Net: Boundary optimization and multi-scale context awareness based building extraction from high-resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Jan. 2022.

[32] L. Wang, S. Fang, X. Meng, and R. Li, "Building extraction with vision transformer," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Jan. 2022.

[33] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[34] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.

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

[36] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. CVPR*, 2018, pp. 801–818.

[37] W. Wang et al., "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8439–8448.

[38] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.

[39] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "BiseNet v2: Bilateral network with guided aggregation for real-time semantic segmentation," in *Proc. CVPR*, 2021, pp. 1–18.

[40] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[41] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. NIPS*, vol. 34, 2021, pp. 12077–12090.

[42] W. Sun and R. Wang, "Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with DSM," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 3, pp. 474–478, Mar. 2018.

[43] E. Maltezos, A. Doulamis, N. Doulamis, and C. Ioannidis, "Building extraction from LiDAR data applying deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 155–159, Jan. 2019.

[44] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. CVPR*, 2011, pp. 991–998.

[45] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[46] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ADE20K dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5122–5130.

[47] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[48] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[49] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2021, pp. 1–22.

[50] L. Yuan et al., "Tokens-to-token ViT: Training vision transformers from scratch on ImageNet," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 538–547.

[51] X. Chu, Z. Tian, B. Zhang, X. Wang, and C. Shen, "Conditional positional encodings for vision transformers," 2021, *arXiv:2102.10882*.

[52] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in transformer," in *Proc. NIPS*, vol. 34, 2021, pp. 15908–15919.

[53] W. Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 548–558.

[54] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. CVPR*, 2020, pp. 213–229.
[55] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *J. Mach. Learn. Res.*, vol. 20, no. 1, pp. 1997–2007, 2019.
[56] M. Tan et al., "MnasNet: Platform-aware neural architecture search for mobile," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2815–2823.
[57] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. ICML*, 2019, pp. 6105–6114.
[58] P. K. A. Vasu, J. Gabriel, J. Zhu, O. Tuzel, and A. Ranjan, "MobileOne: An improved one millisecond mobile backbone," 2022, *arXiv:2206.04040*.
[59] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.

**Huaigang Huang** received the master's degree from the School of Geographic Science and Remote Sensing, Guangzhou University, Guangzhou, China, in 2021. He is currently pursuing the Ph.D. degree with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada.

His research interests include computer vision and intelligent extraction of remote sensing information.

**Jiabin Liu** received the B.S. degree in remote sensing science and technology from Chang'an University, Xian, China, in 2020, the M.S. degree in surveying and mapping engineering from Guangzhou University, Guangzhou, China, in 2023. He is currently pursuing the Ph.D. degree with the Department of Cartography and geographic information systems, Yunnan Normal University, Kunming, China.

His major research interests include remote sensing, image processing, and computer vision.

**Ruisheng Wang** (Senior Member, IEEE) received the B.Eng. degree from Wuhan University, Wuhan, China, in 1993, the master's degree from the University of New Brunswick, Fredericton, NB, Canada, in 1993, and the Ph.D. degree from McGill University, Montreal, QC, Canada, in 1993.

He is currently a Professor with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada. His research interests include geomatics and computer vision, especially point cloud processing.