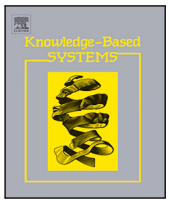
Knowledge-Based Systems 317 (2025) 113413

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



SFGNet: Salient-feature-guided real-time building extraction network for remote sensing images

Jin Kuang [a,b,c], Dong Liu [a,b,*]

[a] School of Computer and Artificial Intelligence, Xiangnan University, 423300, Chenzhou, Hunan, China
[b] Hunan Engineering Research Center of Advanced Embedded Computing and Intelligent Medical Systems, Xiangnan University, Chenzhou, 423300, Hunan, China
[c] School of Geosciences, Yangtze University, 430100, Wuhan, Hubei, China

ARTICLE INFO

Dataset link: https://github.com/gasking/SFGNet

Keywords:
Building extraction
Real-time semantic segmentation
Lightweight model design
Feature extraction and fusion

ABSTRACT

Building extraction is crucial for interpreting remote-sensing images. However, existing methods struggle to balance accuracy with inference speed, limiting their support for high concurrency and real-time processing. Although recent approaches have improved segmentation, significant hurdles remain in feature lightweighting, capturing salient features, and ensuring semantic coherence across different characteristics. This paper presents a salient-feature-guided real-time building extraction network (SFGNet), designed to investigate and integrate salient information, such as semantics, details, and borders, thereby improving segmentation performance. First, an effective feature extraction module called Dual-branch Cascade Module (DCM) was developed to extract relevant channel information by learning the shallow details and boundary features of buildings. Additionally, an Offset Feature Alignment Module (OFAM) is designed to minimize the feature offset in both high- and low-frequency connection zones to capture detail and contour edge feature information. A lightweight Context Feature Aggregation Module (CFAM) was implemented in the decoder stage to consolidate local and global features. Finally, a novel hybrid loss function was designed to address the imbalance in single-view, high-density distributions. On the three public datasets (Massachusetts Builds, WHU Aerial Image, and Potsdam Dataset), our model achieves mIoU scores of 75.45%, 89.40%, and 93.16%, respectively. Furthermore, an additional cross-domain experiment on an external untrained real dataset demonstrated outstanding generalization performance. With only 2.397 M parameters, the model reaches an 130.62 FPS, outperforming current state-of-the-art models in terms of both segmentation accuracy and inference speed. These results demonstrate the potential of SFGNet for real-time building segmentation. The Code is available at https://github.com/gasking/SFGNet.

1. Introduction

Building extraction constitutes a key link in the intelligent interpretation of remote sensing images and plays an extremely significant role in urban planning, urban change monitoring, topographic map updating, population density estimation, and urban disaster emergency response [1].

Currently, deep learning methods, including encoder–decoder convolutional neural networks (CNNs), self-attention-based transformer frameworks [2], and the recently developed context-aware Mamba model [3], are widely employed in remote sensing image interpretation. For the task of building extraction, these methods significantly outperform traditional handcrafted-feature-based approaches in terms of segmentation accuracy, as a result of their ability to extract deep features. However, these methods rely heavily on convolutional or self-attention operations as core feature extractors, resulting in substantial computational costs. This limitation makes it difficult to meet the requirements of actual engineering applications such as high-concurrency urban disaster response and millimeter-level military drone monitoring tasks.

Several lightweight segmentation models have emerged in recent years to achieve a balance between accuracy and real-time performance, endeavoring to enhance performance through efficient feature extraction. For example, a Dual Attention Network [4] enhances semantic associations by adaptively aggregating global and local features. DDRNet [5] boosts feature extraction efficiency by incorporating ResNet residual blocks [6] during encoding. Inspired by HRNet [7], FFNet [8] expands the receptive field to increase feature

* Corresponding author.
E-mail addresses: gasquue@gmail.com (J. Kuang), liudong@xnu.edu.cn (D. Liu).
URLs: https://github.com/gasking (J. Kuang), https://github.com/promisedong/ (D. Liu).

https://doi.org/10.1016/j.knosys.2025.113413
Received 19 November 2024; Received in revised form 11 January 2025; Accepted 22 March 2025
Available online 2 April 2025

interaction. However, for lightweight models, the relatively coarse features extracted to guarantee real-time performance result in weak discriminative ability, negatively impacting segmentation accuracy.

Recently, some studies have demonstrated that extracting salient features (such as details, context, and boundaries) to guide the training of deep networks can effectively improve segmentation accuracy without increasing the time complexity. For example, BiseNetv1 [9] uses a dual-branch structure with a spatial branch capturing fine details and a context branch focusing on global features, thereby achieving lightweight and effective feature aggregation. BiseNetv2 [10] further refined the backbone and feature fusion of BiseNetv1, improving segmentation performance while maintaining real-time efficiency. PIDNet [11] introduced a three-branch architecture, combining detail, context, and boundary extraction to enhance shallow salient feature collaboration in decision-making.

Through further research, we found that the effectiveness of salient feature guidance depends not only on capturing salient information across multiple levels but also on the design of an effective strategy that ensures seamless interaction among these diverse salient features. However, current models still face several limitations: (1) Multi-level salient features have not been fully exploited and feature guidance for the boundaries of objects is simply conducted based on shallow salient features without considering efficient feature interaction guidance between deep features (position information of the objects) and shallow features; (2) When multiple features convey the same semantic information, effectively guiding the model to focus on and extract the most salient features remains a challenge requiring further exploration. (3) Existing networks often struggle with redundant features during extraction, hindering their ability to self-optimize and select high-response features during training, which compromises model generalization performance.

To address these challenges, we propose a salient-feature-guided real-time building extraction network (SFGNet) that enables efficient feature extraction. This network facilitates fast responses to critical building information using effective feature extraction modules and optimal interactions of boundary and spatial information characteristics, enabling collaborative functionality and improving real-time segmentation performance. Specifically, we developed an efficient lightweight dual-branch cascade module (DCM) to extract salient channel information from shallow details and boundary features. The shallow contour details and high-level semantic information are combined using a cascaded skip connection to form the backbone output features. The primary component of the DCM is the self-decision salient feature convolution module (SDCM), which leverages a Gaussian sequence [12] for self-optimization and high-response feature extraction. To address feature offsets in both the high- and low-frequency regions, an offset feature alignment module (OFAM) is introduced. In the decoder, we developed a streamlined context feature aggregation module (CFAM) for synthesizing local and global features, including weighting procedures for features across several channels, to consolidate feature information effectively. The number of convolutional processes in these modules was carefully optimized to reduce computational complexity. Finally, a hybrid loss function is implemented to enhance the model's ability to capture building boundary details, particularly in single-view, high-density, and long-tailed object distributions. By combining the above modules, SFGNet strikes an excellent balance between segmentation accuracy and FPS, making it highly suitable for real-time building extraction.

To summarize, the primary contributions in this study are as follows:

(1) We propose a novel real-time building segmentation network called SFGNet guided by salient features. It can fully exploit feature information such as the geometric position, details, context, and boundaries of buildings, and effectively enhance the segmentation accuracy without increasing the inference speed.
(2) We designed an efficient and lightweight feature extraction module (DCM), feature decision module (SDCM), and a feature aggregation module (CFAM), which can synergistically extract salient features from superficial details and boundary information, thereby maximizing the functionality of these salient features for boundary extraction.
(3) We propose a hybrid loss strategy combining self-distillation segmentation alignment loss (SSA loss), boundary loss, and location loss (Loc loss). This approach eliminates the impact of semantic expression ambiguity among feature maps of different scales on the performance of building segmentation, enabling the model to focus more on learning building foreground objects.
(4) The novel SFGNet outperformed other real-time segmentation networks in terms of both inference speed and segmentation accuracy. With 2.397 M parameters and 14.086 GFLOPs, SFGNet achieved impressive mIoU of 90.06% and 89.40% on the Potsdam and WHU Aerial Image Dataset, respectively, while reaching inference speeds of 130.62 FPS and 114.52 FPS. It achieves a remarkable 75.46% mIoU on the Massachusetts Builds Dataset, which features dense urban environments. Additionally, the superior generalization ability of our model was validated through cross-domain building detection using unlabeled data from real-world scenarios, further confirming its superiority over current mainstream real-time segmentation algorithms.

2. Related work

2.1. Building extraction

Building extraction, which is a descending task of semantic segmentation in remote sensing, is distinct from multi-class semantic segmentation. It can be considered as a classical binary classification task. Remote sensing images, with their high resolution, high density, and single-target perspective, pose a significant challenge to model design. However, in recent years, exciting advancements have been made in building extraction techniques, particularly in the widespread use of CNNs. For example, by merging different convolutional layers, as reported in the HF-FCN [13], a CNN can express the contextual feature information of feature maps through a cascade of local features. MAP-Net [14] gradually extracts high-level semantic features through multiple parallel paths with fixed-spatial-resolution features at each stage to obtain refined building features. GCCINet [15] attempts to extract salient and fused features using an attention mechanism and dilated convolution. The Siamese UNet [16] uses dual branch of weight sharing to combine the segmentation prediction of original images and corresponding sub-sampled images to improve the classification accuracy of large buildings. To extract more refined building contours, DMBC-Net [17] utilizes a multi-task learning framework, including semantic segmentation, direction prediction, and distance estimation, allowing the network to segment buildings accurately. In combination with edge refinement prediction, CBR-Net [18] optimizes building boundaries in a coarse-to-fine stage-by-stage fashion, improving boundaries by sensing the orientation of image pixels relative to the nearest object center in the network. Line2Poly [19] uses feature lines as geometric relationships in an end-to-end manner and extracts the contours of buildings by recovering the topological relationships of lines in the current building. When developing building extraction algorithms using CNNs, it is crucial to enhance the significant features of network models, including contextual semantic and edge features [20], or introduce prior knowledge, including geometric relationships and building boundary information, to improve building extraction performance.

2.2. Real-time semantic segmentation

In recent years, the practical application of real-time semantic segmentation has developed rapidly and has become a focal point for many researchers. Many methods have focused on optimizing segmentation accuracy and inference speed. For example, ENet [21] is a lightweight network with extremely fast response times. ICNet [22] accelerates inference through image concatenation. SwiftNet [23] reduces computational overhead using a lightweight feature extraction network and upsampling decoder. DFANet [24] enhances feature expression and reduces model complexity by reusing features. ESPNet [25] uses efficient pyramid expansion convolutions to enhance multi-scale contextual features for real-time segmentation performance. Additionally, ShuffleSeg [26] employs a lightweight ShuffleNet [27] backbone network, reducing computations and improving feature interaction through depthwise-separable convolutions and channel-shuffling mechanisms. DWR-Seg [28] introduces novel expansion residual and simple inverse residual modules, utilizing multi-rate deep expansion convolutions for feature extraction. DDRNet [5] implements a deep dual-branch network with multiple bilateral fusions to enhance feature fusion. LPSNet [29] uses progressively scalable networks, optimizing the trade-off between the number of convolution blocks and channels for the best balance between speed and accuracy. Moreover, PIDNet combines proportional–integral–derivative control with CNNs, utilizes a three-branch structure to analyze detail, context, and boundary features, effectively preventing the drowning of detail features; FFNet reconsiders network design by simplifying the architecture to optimize the receptive field and improve inference speed for large images. Inspired by these approaches, the SFGNet incorporates an efficient and lightweight feature extraction module as its core component. We combined Gaussian sequences to enable the network to select salient features adaptively for decision-making. Additionally, we draw on the design principles of PIDNet to refine building boundary features using boundary information. Innovatively, we introduce the geometric position information of buildings to guide the network in fully exploring the semantic differences between the foreground and background of buildings. By integrating these techniques, SFGNet achieves an optimal balance between segmentation accuracy and inference speed, offering an advanced solution for real-time building extraction.

2.3. Lightweight network architectures

Following research on group/depth-separable convolutions, the development of lightweight network architectural designs has accelerated, as illustrated by Xception [30], MobileNet [31], and ShuffleNet, which achieve a balance between computational complexity and memory access costs. GhostNet [32] uses depthwise-separable convolution to process a portion of the channel features while the other features are mapped linearly, effectively reducing the FLOPs of the model. FasterNet [33] utilizes a novel partial convolution that can extract spatial features by reducing redundant computation and memory access costs. Inspired by FasterNet, we designed a real-time semantic segmentation network with high accuracy and low parameters by balancing model complexity, memory access cost, and actual inference speed.

2.4. Feature utilization modules

The keys to semantic segmentation are how to use the features captured by the network efficiently and how to capture richer contextual information for feature fusion.

(1) Feature extraction modules: The Stage-aware Feature Alignment Network [34] introduces a stage-aware feature alignment module to align and aggregate the feature maps from the two adjacent layers. The Feature Augmentation Block further enhances the spatial and contextual features of the encoders. PP-MobileSeg [35] uses an Aggregated Attention Module to filter detail features by evaluating a semantic feature set. SegNetXt [36] demonstrates that convolutional attention outperforms self-attention for encoding contextual features and, achieves better performance with a novel convolutional computation. Non-local Neural Networks [37] employ non-local operations to capture remote dependencies by computing the response of a location as the weighted sum of the feature responses from all locations in the feature locality, thereby improving feature effectiveness.

(2) Feature aggregation modules: Inspired by the optical flow alignment in video frames, SFNet [38] introduces a Flow Alignment Module to capture the semantic flow between high-resolution semantic and low-resolution detail information. AligSeg [39] utilize the Alignment Feature Aggregation (AlignFA) and Alignment Context Modeling modules (AlignCM) to address feature misalignment. AlignFA uses a learnable interpolation offset to alleviate misalignment from different resolutions, wthereas AlignCM adaptively selects context positions for embedding. OCNet [40] leverages self-attention to explore object context, defining it as a set of pixels within the same category.

Through our literature review and preliminary-experiments, we found that increasing the response degree of features makes the network more likely to learn important feature information. By effectively aggregating high- and low-resolution features, a model can enhance its ability to extract fine-grained features. Inspired by this concept, we developed the lightweight CFAM to facilitate the weighted aggregation of local and global features. Additionally, we designed the OFAM to mitigate noise interference during the feature up-sampling stage, addressing the issue of poor fine-grained feature extraction in lightweight models.

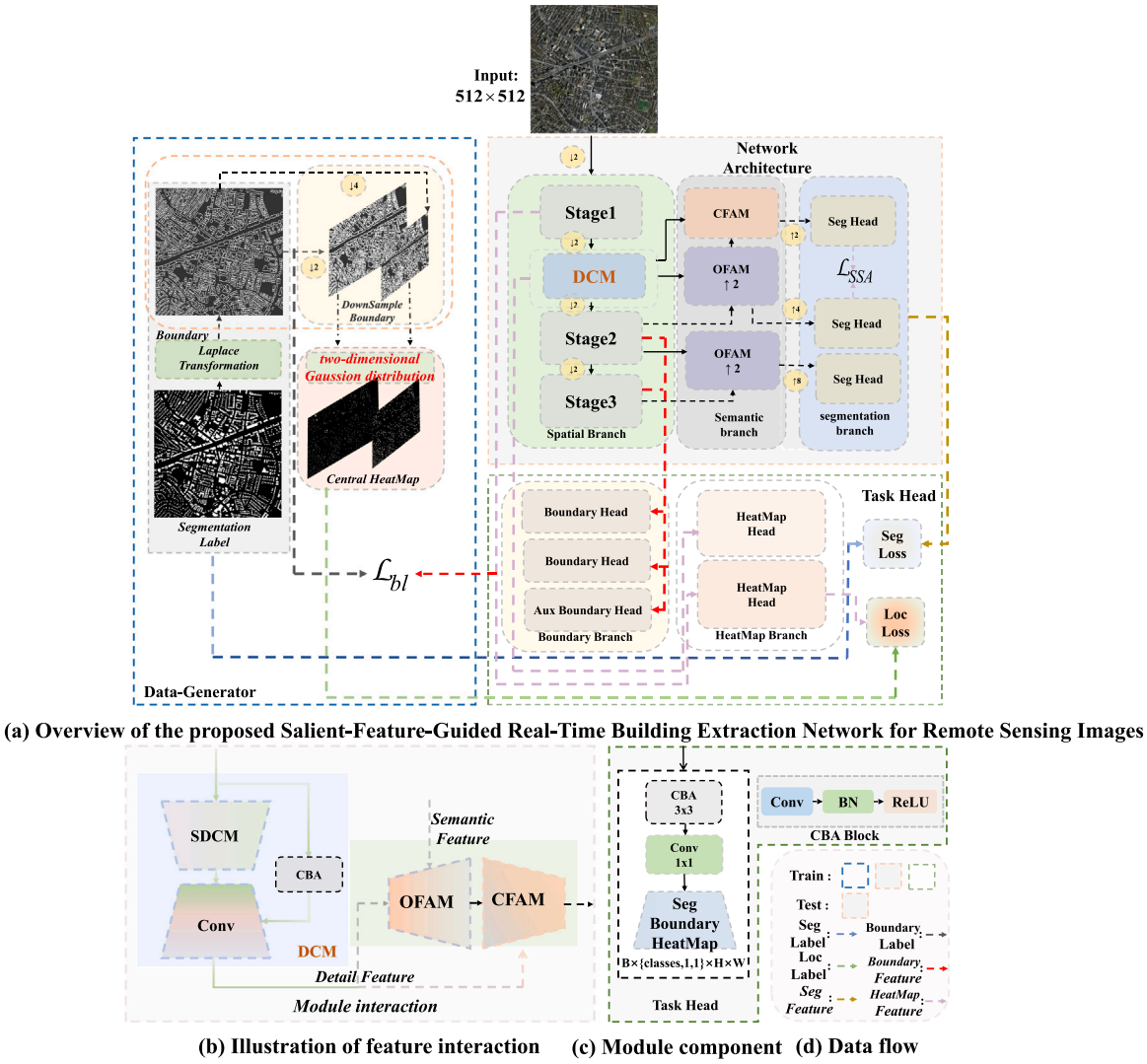
3. Methodology

3.1. Overview of SFGNet

Reinforcing the learning of salient features is crucial in lightweight models. However, many current models lose detail features during extraction, negatively affecting global feature responses. Additionally, boundary, semantic, geometric, and contextual spatial information are vital for semantic segmentation tasks. However, some networks struggle to extract these features collaboratively from high-resolution remote sensing images, hindering effective feature responses. To address this issue, we designed a real-time building extraction network called SFGNet, which is guided by salient features to better capture building outlines, incorporate geometric positions, and enable interactive feature guidance. As shown in Fig. 1, SFGNet consists of four components, namely training data generator, lightweight feature extraction architecture, segmentation head, and hybrid loss function.

(a) **Training data generator:** An input Segmentation label is transformed via Laplace transformation to generate the contour binary mask boundary label image of the corresponding scale of the image. Subsequently, the mask boundary label image is down-sampled by factors of 2 times and 4 times to generate two different mask boundary label images. The mask boundary label images after down-sampling are used as 2D Gaussian distributions for mathematical modeling. Specifically, contours are generated as minimum bounding rectangles, where the center of each rectangle is considered as the mean and the variance of the rectangles is calculated. Subsequently, a 2D Gaussian image is generated using the mean value as the adjacent boundary. The mask boundary label generation process is defined in Algorithm 1.

(b) **Lightweight feature extraction module:** The encoder in our network primarily consists of the following three stages and the DCM for feature downsampling with a stride of two. The decoder mainly consists of the OFAM and CFAM.



(a) Overview of the proposed Salient-Feature-Guided Real-Time Building Extraction Network for Remote Sensing Images

(b) Illustration of feature interaction (c) Module component (d) Data flow

Fig. 1. (a) Overall architecture of the proposed SFGNet, consisting of a Data-Generator, Network Architecture, and Task Head; (b) the feature interactions between SDCM, OFAM, and CFAM; (c) Module component; (d) Data flow.

(1) **Spatial Branch:** Stage 1 use the CBA module (Fig. 1(c)), while stages 2 and −3 use a dual-branch multi-layer convolution to extract edge features, followed by a 3×3 global pooling layer to fuse global features. After downsampling by a factor of two, the DCM focuses on learning significant edge details and channel feature information from the shallow-layer features. A detailed explanation of the DCM is provided in Section 3.3.

(2) **Semantic Branch:** The semantic branch receives semantic feature information from the spatial branch. The OFAM performs offset feature correction across different scales, thereby reducing the artifacts introduced by high-level semantic features during upsampling. The CFAM aggregates the original feature map with OFAM-calibrated features by combining contextual and semantic information. This approach strengthens the aggregation of local low-frequency features during decoding and helps capture global high-frequency boundary features.

(3) **Segmentation Branch:** The 1/8 and 1/16 features aggregated by the OFAM are up-sampled by 4 times and 8 times, and then aggregated with the features outputted by the CFAM, which acts the segmentation head.

(c) **Task Head:** The model consists of three task heads, as shown in Fig. 1: the segmentation head (responsible for pixel classification), the boundary detection head (regresses building boundaries), and building localization head (provides feature guidance for coarse building localization).

3.2. Self-decision salient feature convolution module

Existing lightweight designs such as, MoibleNet [31] and ShuffleNet focus on depthwise-separable convolution and reduce the number of parameters and computations through channel-by-channel convolution and 1×1 mixed-channel convolution. However, based on the use of depthwise-separable convolutions, frequent memory access is required. Therefore, although these models have relatively small numbers of computations, their operational efficiency is not significantly improved. Our study was inspired by FasterNet and GhostNet. There is a certain amount of redundancy in feature channels, so extracting features from all channels during the feature extraction process is unnecessary. FasterNet uses manual regulations to select the most effective channel features for convolution calculations and performs a linear channel superposition of the remaining channel features. However, this approach weakens the ability of the model to understand adjacent features in an image and removes its ability to fit the spatial invariance of an image.

Algorithm 1 Generating mask boundary labels

Input: Building center semantic information label $heat_{b2} \in \mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times 1}$
Output: Binary image of a building $x_b \in \mathbb{R}^{H\times W}$

Initialize: $gray_b \leftarrow Readbinary(x_b)$;
$binary_b \leftarrow Threshold(gray_b)$;
$BBoxes \leftarrow GetBoxes(FindContours(Canny_b))$
for Box **in** $BBoxes_b$ **do**
  Get a single rectangle data: $C_{xy} \leftarrow (lt+rb)\times 0.5$; $Box_{wh} \leftarrow (rb-lt)$.
  Get the label-center mask with 2x Down-sampling: $label^2_{xy} = \frac{C_{xy}}{2}$; $label^2_{wh} = \frac{Box_{wh}}{2}$
  Get 2x Down-sampling Gaussian kernel radius: $radius_2 = 0.5 \times gaussian\_radius(label^2_{wh}) \div 3$
  Plot the center of the Gaussian distribution:
  for $j = label^2_{xy}[1] - 3\times radius_2$ **to** $label^2_{xy}[1] + 3\times radius_2$ **do**
    for $i = label^2_{xy}[0] - 3\times radius_2$ **to** $label^2_{xy}[0] + 3\times radius_2$ **do**
      if $i < \frac{W}{2}$ **and** $j < \frac{H}{2}$ **then**
        $heat_{b2}[j,i] = exp(\frac{-(i-label^2_{xy}[0])^2-(j-label^2_{xy}[1])^2)}{2\times {radius_2}^2}$
      end if
    end for
  end for
end for
return $heat_{b2}$

Therefore, this paper proposes a method to select effective features adaptively using an Embedding Query vector Gaussian sequence [12] as an optimized parameter to make feature selection decisions. The structure of the proposed SDCM illustrated in Fig. 2 and its main features are described below:

1. For ordinary convolution, the input feature $F \in \mathbb{R}^{h\times w\times c}$, input scale is equal to the output scale, input channel is C_{in}, output channel is C_{out}, and convolution kernel size is $k\times k$. The FLOPs calculation for ordinary convolution is shown in Eq. (1).

$$FLOPs_{conv} = (K\times K\times C_{in} + (K\times K\times C_{in} - 1) + 1) \times h\times w\times C_{out} \quad (1)$$

   (a) According to Eq. (1), FLOPs can be reduced by reducing the size of the input channel C_{in}. In this study, we selected a number channels of $c_p = 1/4\times C_{in}$ as the salient features (salient features can effectively make feature decisions without redundant features and reduce the computational complexity of the model). Eq. (2) represents the number of computations required for the SDCM.

$$FLOPs_{SDCM} = (K\times K\times C_p + (K\times K\times C_p - 1) + 1) \times h\times w\times C_p \approx \frac{1}{16}FLOPs_{conv} \quad (2)$$

2. Aiming at the problem of C_p channel selection, this paper proposes a method based on the Gaussian score to obtain the optimal parameter model that conforms to the data distribution, allowing the proposed module to make optimal selection decisions with a large number of features. The Gaussian score is defined in Eq. (3).

$$Gaussian\ score = Randn(c,) \quad (3)$$

   (a) The Gaussian scores are sorted in descending order and the high response feature $F_1 \in \mathbb{R}^{h\times w\times c_p}$ composed of the unmasked sequence is selected, as defined in Eqs. (4)–(5).

   (b) The low-response feature $F_2 \in \mathbb{R}^{h\times w\times(c-c_p)}$, which is composed of the channel feature information of the remaining sequence $c - c_p$ does not participate in feature extraction, as depicted in Eq. (6).

   (c) Subsequently, we performed a convolution operation on $F1$ to extract low-frequency detail feature information to obtain $P_1 \in \mathbb{R}^{h\times w\times c_p}$. P1 and F2 are superimposed in the channel dimensions to obtain $P_2 \in \mathbb{R}^{h\times w\times c}$, which constitutes a lightweight high-response feature extraction operation, as shown in Eq. (7).

   (d) P2 operates through the CBA module and convolution to obtain an efficiently corrected feature vector $P_3 \in \mathbb{R}^{h\times w\times c}$. To prevent the effective features with very weak responses from being lost during the convolution operation, P3 is constructed from $F \in \mathbb{R}^{h\times w\times c}$ through a pixel-by-pixel summation operation, as shown in Eq. (8).

$$score = argsort(Gaussian\ score) \quad (4)$$

$$unmask = score[0 : c_p]$$
$$mask = scorep[c_p : c - c_p] \quad (5)$$

$$f_1 = f[:, unmask, :, :], f_2 = f[:, mask\ :, :, :] \quad (6)$$

$$P_1 = Conv(F_1), P_2 = cat((F_2, P_1), axis = -1) \quad (7)$$

$$P_3 = Conv(CBA(P2)) \oplus F \quad (8)$$

3.3. Dual-branch cascade module

The proposed SDCM can effectively reduce the number of model computations; however, this reduction may be accompanied by the loss of important features. The challenge of designing an efficient feature extraction module to ensure real-time performance while maintaining the accuracy of segmentation is significant. Previous semantic segmentation methods used residual skip connections for feature aggregation or convolution kernels with receptive fields of different sizes for refined local feature extraction. However, the use of convolution kernels of different scales for local feature extraction introduces redundant local detail features and generates error gradients during the model training process. The proposed DCM is designed to extract local features efficiently and reduce the feature response failures caused by error gradients, as shown in Fig. 3.

First, redundancy in the input feature $f_1 \in \mathbb{R}^{h\times w\times c}$ is eliminated through ordinary convolution to obtain $\hat{f}_1 \in \mathbb{R}^{h\times w\times c1}$. Then $\hat{f}_2 \in \mathbb{R}^{h\times w\times c2}$ is obtained by the SDCM and ordinary convolution, and $\hat{f}_3 \in \mathbb{R}^{h\times w\times c3}$ is obtained through feature cascade superposition in the channel dimension for effective feature fusion. Moreover, the semantic information of the context of the feature map is obtained over long distances was captured. $Q_1 \in \mathbb{R}^{h\times w\times c3}$ is obtained through low-pass filtering performed on f_1 with dimensions of 1×1 kernel and $Q_2 \in \mathbb{R}^{h\times w\times c3}$ is obtained through high-pass filtering performed with dimensions of $k\times k$ kernel on f_1. Finally, $\hat{f}_3, Q_1, Q_2$ are added pixel-by-pixel to obtain $\hat{f} \in \mathbb{R}^{h\times w\times c}$. The overall description is given in Eqs. (9)–(11), where $\oplus$ indicates the adding of element-wise values.

$$\hat{f}_1 = Conv(f_1),$$
$$\hat{f}_2 = Conv(SDCM(f_1)), \quad (9)$$
$$\hat{f}_3 = Conv(cat((\hat{f}_1, \hat{f}_2), axis = -1))$$

$$Q_1 = CBA(f_1)_{1\times 1}, Q_2 = CBA(f_1)_{k\times k} \quad (10)$$

$$\hat{f} = \hat{f}_3 \oplus Q_1 \oplus Q_2 \quad (11)$$

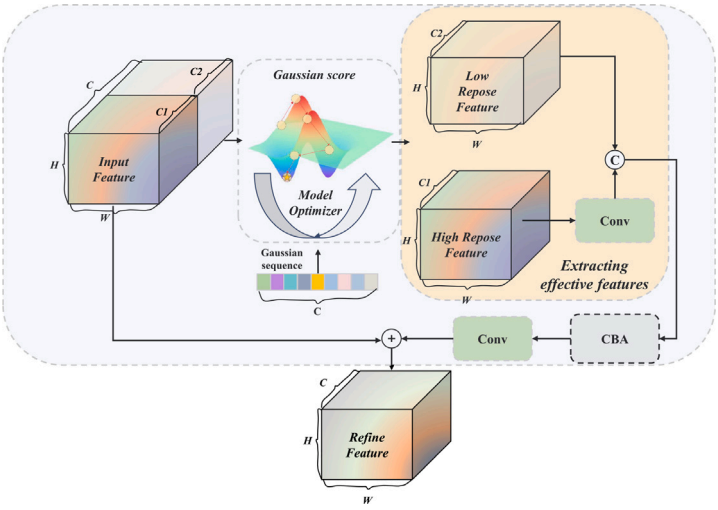


Fig. 2. Structural diagram of the SDCM.

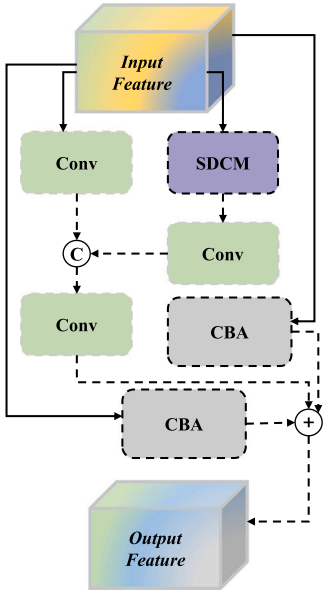


Fig. 3. Structural diagram of the DCM.

3.4. Offset feature alignment module

In addition to feature extraction, for semantic segmentation, effectively aggregating features of different sizes from different modalities is necessary to enhance segmentation performance. In previous methods, the high-level feature $Z_1 \in \mathbb{R}^{h_1 \times w_1 \times c_1}$ is up-sampled to the $h_2 \times w_2$ scale of the shallow feature $Z_2 \in \mathbb{R}^{h_2 \times w_2 \times c_2}$, and then the features are superimposed with $Z_1 \in \mathbb{R}^{h_1 \times w_1 \times c_1}$ in the channel dimension. Finally 1×1 convolution is used to change the output channel to remove redundant feature information. In combination with the activation function, this operation allows the model to focus on salient features. However, directly up-sampling the high-level feature $Z_1 \in \mathbb{R}^{h_1 \times w_1 \times c_1}$ and aligning the shallow feature $Z_2 \in \mathbb{R}^{h_2 \times w_2 \times c_2}$ produces feature artifacts and causes feature offset, negatively affecting segmentation performance. This issue is particularly significant in high- and low-frequency connection regions.

In this study, we designed OFAM, which uses two sets of learnable parameters offset_x, offset_y to fit the feature offset caused by the up-sampling process of the high-level feature $Z_1 \in \mathbb{R}^{h_1 \times w_1 \times c_1}$. The structure of the OFAM is illustrated in and operations are described below.

(1) First, $Z_1 \in \mathbb{R}^{h_1 \times w_1 \times c_1}$ is up-sampled and 3×3 convolution is performed to obtain $w_1 \in \mathbb{R}^{h_2 \times w_2 \times c_1}$. The shallow $Z_2 \in \mathbb{R}^{h_2 \times w_2 \times c_2}$ is also convolved by a 3×3 convolution kernel to suppress the special segmentation information of the detailed edge contours, which makes it difficult to express semantics to obtain $w_2 \in \mathbb{R}^{h_2 \times w_2 \times c_2}$.

$$w_1 = Conv_{3\times3}(Upsample(Z_1)), w_2 = Conv_{3\times3}(Z_2) \tag{12}$$

(2) Second, the features are fused in the channel dimension to obtain $\delta \in \mathbb{R}^{h_2 \times w_2 \times c}$, and after performing two 3×3 convolution operations on δ, the up-sampling offset difference $\Delta_{high}^{h_2 \times w_2 \times 2}$ of the original high-level feature $Z_1 \in \mathbb{R}^{h_1 \times w_1 \times c_1}$ and shallow detail lossless offset $\Delta_{low}^{h_2 \times w_2 \times 2}$ of the shallow layer $Z_2 \in \mathbb{R}^{h_2 \times w_2 \times c_2}$ are obtained.

$$\begin{aligned} \delta &= cat((w1, w2), axis = -1), \\ \Delta_{high}^{h_2 \times w_2 \times 2} &= Conv_{3\times3}(\delta), \\ \Delta_{low}^{h_2 \times w_2 \times 2} &= Conv_{3\times3}(\delta) \end{aligned} \tag{13}$$

(3) After obtaining the two offset feature tensors, $u(\cdot,\cdot)$ is used for up-sampling offset calibration. The alignment function is defined as follows:

$$u(F^{h\times w\times c}, \Delta^{h\times w\times 2}) = X^{h\times 1} @ Y^{1\times w} + \Delta^{h\times w\times 2} \tag{14}$$

where @ represents the tensor multiplication operation, and $X^{h\times1}, and\ Y^{1\times w}$ are the 1D offset sequence generated by obtaining h and w from $F^{h\times w\times c}$ and performing the mean sampling of h and w on $[-1,1]$. The value range of $[-1,1]$ indicates that the value at a certain point after up-sampling may come the upper, lower, left, or right neighborhoods of the current point, and the feature offset differences of $\Delta_{high}^{h_2 \times w_2 \times 2}$ and $\Delta_{low}^{h_2 \times w_2 \times 2}$ are added. Finally, the offset features of $\Delta_{high}^{h_2 \times w_2 \times 2}, \Delta_{low}^{h_2 \times w_2 \times 2}$ are outputted after adding the features element-wise using the $u(\cdot,\cdot)$ function. The network updates the $\Delta_{high}^{h_2 \times w_2 \times 2} and \Delta_{low}^{h_2 \times w_2 \times 2}$ parameters of the learnable offset tensor, and then corrects and aligns the semantic information from different scales (see Fig. 4).

3.5. Context feature aggregation module

Semantic segmentation depends heavily on both local feature information and contextual feature information. The feature fusion module (FFM) introduced by BiseNet effectively balances the expression of various feature scales through channel superposition and the convolution of feature maps using the sigmoid function for regularization. As a result, features are extracted with high efficiency. Regardless, the FFM

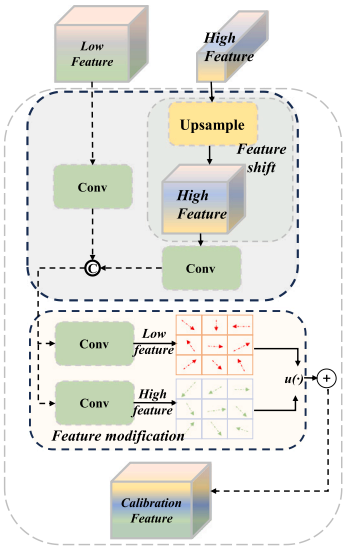


Fig. 4. Structural diagram of the OFAM.

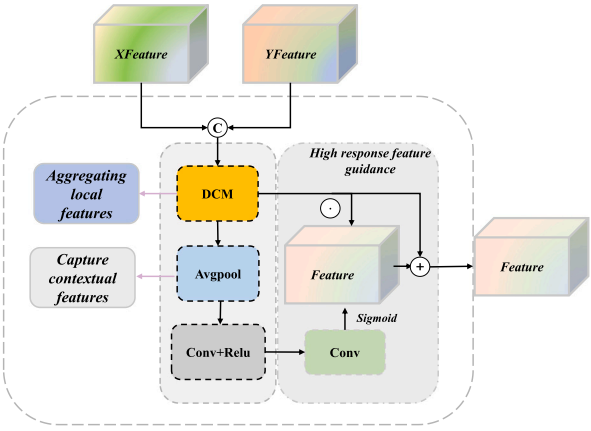


Fig. 5. Structural diagram of the CFAM.

conducts single-layer convolution processing on fused high-dimensional complex features to maintain a lightweight approach, limiting its ability to extract deep semantic information from fused features. Therefore, we modified the FFM to create the CFAM and implemented multi-branch convolution with varying receptive fields to enhance the semantic understanding of the model for shallow features. Furthermore, the SDCM is integrated to facilitate feature redundancy calculations and the effective extraction of high-saliency features, ensuring that the accuracy of the model remains intact while maintaining a lightweight design. Fig. 5 presents the structure of the CFAM.

3.6. Loss function

In this subsection, we describe the loss function used in this study, as well as the inference stage. To improve the feature learning performance of dense buildings in complex scenes, we designed a hybrid collaborative loss function strategy that mainly includes the following loss types.

(a) SSA loss: To align semantic information at different scales, inspired by the CWD and KL losses, we calculate the alignment loss on the 4× and 8× down-sampled segmentation feature maps. The SSA loss $\mathcal{L}_{SSA}$ is defined as follows.

$$\phi(x_c) = \frac{exp(\frac{x_{c,i}}{\tau})}{\sum_{i=1}^{W\times H} exp\frac{x_{c,i}}{\tau}} \tag{15}$$

$$\mathcal{L}_{kl} = \sum_{c=0}^{C}\sum_{i=1}^{H\times W} \phi(f^{c,i}) \cdot log\frac{\phi(f^{(c,i)})}{g^{(c,i)}} \tag{16}$$

$$\mathcal{L}_{SSA}(f,g) = \lambda_1(\mathcal{L}_{kl}(f,g) + \mathcal{L}_{kl}(g,f)) \tag{17}$$

Let f, g denote the subs-sampled segmentation feature maps at 4× and 8×, respectively. Here, $c = 1, 2, 3, \ldots, C$ is the channel index, and $i = 1, 2, 3, \ldots, H \times W$ represents the spatial resolution of the feature map.τ is the hyper-parameter of distribution smoothing; λ_1 is the SSA loss factor, which is taken as 1.0 and 0.5 in this study; ϕ represents the probability distribution map that converts the feature map into the category channels, reduce the impact of scale variations on segmentation. In the early stage of training, $\phi(f^{c,i})$ and $\phi(g^{c,i})$ are large, preserving the relationship between the scales. In the later stages of training, $\phi(f^{c,i})$ is relatively small and the gradient value of $\phi(g^{c,i})$ has no effect on model optimization. Segmentation alignment via self-distillation forces the network to eliminate noise at different scales and learn the salient feature distribution of the foreground.

(b) boundary loss: To fit and optimize building contour boundaries with segmentation loss and capture the edge contour information of buildings, we utilize boundary loss. For the two and four times of the boundary feature map in small and medium down-sampling, which can express the contour detail boundary information of the building to a certain extent, the $\mathcal{L}_{heat}$ is used. However, for the boundaries feature map with 8 times down-sampling, only the semantic information of the boundary is available, and binary loss is used to optimization. The boundary loss $\mathcal{L}_{bl}$ is defined as follows:

$$\begin{aligned}\mathcal{L}_{bl}(\sum_{f=2}^{2,4,8} p_f, g_f) =& \alpha \cdot \mathcal{L}_{heat}(p_2, f_2) \\ &+ \mathcal{L}_{heat}(p_4, f_4) \\ &+ \mathcal{L}_{bce}(p_8, f_8)\end{aligned} \tag{18}$$

$$\alpha = pow((g_2, label_2).mean, 2) \tag{19}$$

where $\sum_{f=2}^{2,4,8}$ represents the down-sampling factor relative to the original image; α represents the adaptive weight coefficient adjustment; $p_f \in \mathbb{R}^{H\times W\times 1}$ represents the predicted boundary probability; $g_f \in \mathbb{R}^{H\times W\times 1}$ represents the true boundary label. $\mathcal{L}_{bce}$ is the binary cross-entropy loss and $\mathcal{L}_{heat}$ is defined as follows.

$$\begin{aligned}\mathcal{L}_{heat}(p_f, g_f) =& -\frac{1}{N}(\{g_f > 0\} \sum_{i}^{H\times W} g_f^i log(p_f^i) \\ &+ (1 - g_f^i) log(1 - g_f^i)) \\ &+ (\{g_f = 0\}\lambda_{noobj} \sum_{i}^{H\times W} g_f^i log(p_f^i) \\ &+ (1 - g_f^i) log(1 - g_f^i))\end{aligned} \tag{20}$$

where $\{g_f > 0\}$ represents the loss of positive samples; $\{g_f = 0\}$ represents the loss of negative samples; N represents the number of positive samples; λ_{noobj} is set to 0.5.

(c) Loc loss: Considering the high density of remote sensing images with complex scene characteristics, the specific performance of a building's foreground and background pixel distribution is extremely unbalanced. When the feature responses are insufficient, it is difficult to balance samples. In this study, the Loc loss function was designed to use the geometric position information of a building itself to learn the foreground features of the building, making up for the loss of feature information of small building

targets caused by high density. The Loc loss $\mathcal{L}_{loc}$ [41] is defined as follows.

$$\mathcal{L}_{loc}(\sum_{i}^{2,4} p_i, l_i) = \mathcal{L}_f(p_i^2, l_i^2) + \lambda_{loc4}\mathcal{L}_f(p_i^4, l_i^4) \tag{21}$$

where $\sum_i^{2,4} p_i, l_i$ represents the predicted localization feature map and the mask localization label with two and four-times downsampling relative to the original image, respectively; λ_{loc4} is a hyperparameter set to 0.2. $\mathcal{L}_{loc}$ is defined as follows.

$$\mathcal{L}_f(p_i, l_i) = \begin{cases} \delta \cdot (1-p_i)^{\beta} \cdot log(p_i), if\ l_i != 0 \\ (1-\delta) \cdot p_i^{\beta} \cdot log(1-p_i), if\ l_i == 0 \end{cases} \tag{22}$$

where δ, β are hyperparameters that are set to 0.25 and 2.0 in this study, respectively.

The overall loss is defined as follows, where $\mathcal{L}_{seg}$ is cross-entropy loss:

$$\mathcal{L}_{total} = \underset{\theta=1}{\overset{\theta}{\operatorname{argmin}}}(\mathcal{L}_{seg} + \mathcal{L}_{SSA} + \mathcal{L}_{bl} + \mathcal{L}_{loc}) \tag{23}$$

The fusion of the loss function strategies described above can effectively guide our SFGNet model to extract the most discriminative features of a building collaboratively to improve the segmentation performance of the model, while ensuring that it lightweight. Our loss functions are examined in greater detail in our ablation experiments.

4. Experiments and analysis

This section presents our experimental design, results, and analysis. Sections 4.1–4.3 cover the dataset, evaluation metrics, experimental environment, and parameter settings used in this study. The comparison results for different datasets and cross-domain experiment for generalization are presented in Section 4.4–4.5.

4.1. Dataset description

To confirm the effectiveness of our method for building extraction, we compared SFGNet with other prominent lightweight real-time segmentation models using three publicly available building extraction datasets: the WHU Aerial Image Dataset, the Massachusetts Builds Dataset, and Potsdam Dataset. Additionally, real-world building image data were used to validate the generalizability of the model. During the model testing phase, we merged the validation and testing sets to create a new validation set to assess the accuracy of the model. Links to the training and testing sets used during training are provided to ensure the reproducibility of our results.

(1) **Massachusetts Builds Dataset** [42]: This dataset consists of 151 aerial images of the Boston area, each with a resolution 1500 × 1500 pixels, covering a total area of approximately 340 km^2. It includes 131 training images, 10 testing images, and 4 validation images, all cropped to 512 × 512 pixels. The training set contains 1066 image pairs, testing contains 90, and validation set contains 36. Target maps were derived from the OpenStreetMap project with data limited to areas with less than 5% missing noise. The dataset covers Boston and its suburbs, featuring buildings of various sizes, including homes and garages. Notably, our model was trained from scratch without pre-training to ensure strong performance and fair evaluation.

(2) **WHU Aerial Image Dataset** [43]: This dataset contains 220,000 buildings extracted from aerial images of Christchurch, New Zealand with a spatial resolution of 0.075 m, covering 450 km^2. The images were corrected for mislabels and most images including 187,000 buildings were down-sampled to a resolution of 0.3 m. The dataset consists of 8189 images (512 × 512), split into 4736 training images (130,500 buildings), 1036 validation images (14,500 buildings), and 2416 testing images (42,000 buildings).

(3) **Potsdam Dataset** [44]: The Potsdam Dataset, which was collected by the International Society for Photogrammetry and Remote Sensing, covers a diverse range of areas, including urban, suburban, and natural regions of Potsdam, Germany. It provides detailed ground truth data with annotations for various categories, buildings, roads, and vegetation. The dataset consists of 36 high-resolution remote sensing images (6000 × 6000 pixels, 5 cm resolution), offering rich scene diversity. In this study, we used RGB images from the Potsdam dataset and focused solely on the building category of annotations. The images were split into patches with a sliding window approach using a 0.1 overlap and a window size of 512 × 512, resulting in 4056 image patches. These patches were then randomly divided into training and testing sets with an 8:2 ratio, resulting in 3244 images for the training set and 812 images for the testing set.

(4) **CScity Dataset:** These data were collected from real-world remote sensing images of Changsha in Hunan Province of China. Four remote sensing images were downloaded from 4D WorldView, with satellite types of SVN1-01 and SVN3-01, sensor types of MUX and PMS, a resolution of 0.5 m, and single spectral band. Images were captured between September 25 of 2024, and November 2 of 2024. The four images were cropped and resized 2048 × 2048 images. These images had no labels and were used as third-party, unseen data to test the generalizability of our method.

4.2. Experimental setting

The hardware used in for this study included an Intel Xeon Gold 6348 CPU, and Nvidia A30 and Nvidia RTX 3090 GPUs. The software environment was based on Windows 10, Python 3.7 and PyTorch 1.13.1+cu117. Adam was used as the optimizer, with a cosine decay learning rate schedule, ranging from a minimum of 1e−6 to a maximum of 2.5e−4, with a weight decay of 1e−3. Images are randomly sampled and resized to 512 × 512 pixels for training. The batch size was 128 with 2.7K iterations for the Massachusetts Buildings Dataset, 2.59K iterations for the WHU Aerial Image Dataset , and 2.53K iterations for the Potsdam Dataset. Data augmentation included random flips, affine transformations, and normalization.

4.3. Evaluation metrics

(1) Accuracy Evaluation Metrics: In this paper, the IoU, Pixel Accuracy (PA), Recall, F1-Score, and mIoU were used as the accuracy evaluation indices, with all units being percentages.

(2) Comprehensive evaluation metric: For lightweight real-time semantic segmentation models, the numbers of parameters and FLOPs are key factors affecting performance. To evaluate the models comprehensively, we adopted the equilibrium index (ES) inspired by Easy-Net [45]. This index considers five key metrics: IoU (the primary metric for segmentation performance), F1-Score (a comprehensive performance measure), Parameters (Param), FLOPs (computational complexity), and FPS (the inverse of image processing time). Param and FLOPs are inversely related with lower values indicating better performance. The function $f(x)$ maps these values, and $f(x)$ and ES are defined as follows:

$$f(x) = \frac{100.}{x} \tag{24}$$

$$ES = \alpha_1 \times IoU + \alpha_2 \times F1\_Score + \beta_1 \times f_1(Param) + \beta_2 \times f(FLOPs) + \lambda \times FPS \tag{25}$$

Here $\alpha_1, \alpha_2, \beta_1, \beta_2, \lambda$ are harmonic coefficients values of 0.35, 0.15, 0.3, 0.1, and 0.1, respectively, are used as the coefficients of the equilibrium index. Notably, CPU and GPU inference

Table 1
Comparison of various real-time semantic segmentation algorithms on the Massachusetts Builds Dataset (retaining BN layers during FPS testing, the optimal results are shown in bold and suboptimal results are shown underlined).

Method	Reference	Param (M)	FLOPs (G)	Resolution	FPS (Torch)	PA	Recall	F1-Score	IoU	mIoU	ES
BiseNetv1	ECCV 2018	13.419	15.261	512 × 512	234.03	92.00	79.36	73.95	58.66	74.82	**57.92**
BiseNetv2	IJCV 2021	5.182	17.699	512 × 512	155.70	87.42	79.83	44.88	28.93	57.84	38.78
PIDNet	CVPR 2023	7.717	6.341	512 × 512	113.27	86.74	72.14	43.55	27.84	56.93	33.07
DDRNet	Arxiv 2021	5.693	4.580	512 × 512	132.92	87.50	68.86	53.30	36.33	61.44	41.46
FFNet	CVPR 2022	28.544	16.466	512 × 512	104.26	90.36	81.58	64.41	47.51	68.47	38.46
STDC	CVPR 2021	10.637	15.900	512 × 512	**240.17**	88.77	71.51	60.46	43.33	65.52	51.70
LPSNet	IJCV 2023	3.372	**2.229**	512 × 512	103.22	86.31	73.07	38.53	23.87	54.87	37.84
Ours	–	**2.397**	14.086	512 × 512	114.52	**92.35**	**81.84**	**74.64**	**59.54**	**75.46**	56.71

Table 2
Comparison of various real-time semantic segmentation algorithms on the WHU Aerial Image Dataset (retaining BN layers during FPS testing, the optimal results are shown in bold and suboptimal results are shown underlined).

Method	Reference	Param (M)	FLOPs (G)	Resolution	FPS (Torch)	PA	Recall	F1-Score	IoU	mIoU	ES
BiseNetv1	ECCV 2018	13.419	15.261	512 × 512	234.41	97.40	87.56	88.42	79.25	88.18	**67.33**
BiseNetv2	IJCV 2021	5.182	17.699	512 × 512	156.89	96.21	86.02	82.16	69.72	82.79	58.77
PIDNet	CVPR 2023	7.717	6.341	512 × 512	114.00	95.55	81.03	79.58	66.09	80.61	51.93
DDRNet	Arxiv 2021	5.693	4.580	512 × 512	134.01	96.17	84.60	82.28	69.89	82.85	57.66
FFNet	CVPR 2022	28.544	16.466	512 × 512	104.50	96.92	84.65	86.41	76.08	86.33	51.70
STDC	CVPR 2021	10.637	15.900	512 × 512	**242.29**	97.00	88.26	86.15	75.67	86.18	67.09
LPSNet	IJCV 2023	3.372	**2.229**	512 × 512	139.74	96.22	85.74	82.24	69.84	82.85	64.14
Ours	–	**2.397**	14.086	512 × 512	114.26	**97.72**	**89.67**	**89.71**	**81.34**	**89.40**	66.58

speeds were not used as evaluation indicators in this study because the performance difference between different hardware devices are significant. Therefore, ES was used directly for the comprehensive evaluation of inference speed in this study.

(3) Visual qualitative evaluation metrics: The label is positive, and the prediction is positive (TP). The label is positive and the prediction is negative (FN). The label is negative and the prediction is positive (FP). The label is negative and the prediction is negative (TN).

4.4. Comparative experimental analysis

(1) Results on the Massachusetts Builds Dataset: To evaluate our method comprehensively, we compare it with mainstream lightweight models. In Table 1, one can be see that our model achieved the best results in terms of PA, Recall, F1-Score, IoU, mIoU, and Param, which reflecting its excellent segmentation performance and model complexity, while achieving suboptimal results in the overall ES index. Specific results are as follows. (1) IoU: Our model leads with an IoU of 59.54%, surpassing BiseNetv1 (which uses pre-trained weights) by 0.88%. This improvement can be attributed to DCM, which focuses on foreground building features, helping the model accurately identify small buildings obstructed by trees or dense structures; (2) Param: Our model has the fewest parameters (2.397M), outperforming LPSNet's more complex design. This reduction was achieved by optimizing the receptive field and balancing segmentation performance with reduced model complexity; (3) FPS: Our model yield slightly lower FPS than STDC [46] (approximately 120 fewer FPS) because STDC uses large convolution kernels to reduce memory access complexity at the cost of accuracy. The IoU of STDC is 43.33%, which is 16.21% lower than that of our model, and its ES is 5.01 lower than that of our model; (4) BiseNetv1 achieved an FPS of 234, maintaining good segmentation performance with an efficient feature extraction module. Inspired by this efficiency, we introduced the DCM to optimize feature extraction further and reduce model complexity while improving feature accuracy. Overall, our model achieved the best balance between segmentation performance and inference time. More intuitive visualization results are presented in Fig. 6. In addition to the quantitative analysis, we conducted qualitative visual analysis using various methods. As shown in Fig. 7 (lines 1–4), BiseNetv2, PIDNet, and LPSNet struggle to identify dense buildings, often misrecognize blocks, and fail to segment clear building contours. These issues complicate building extraction. In contrast, our method accurately segments small buildings and fits building contours well, producing clear boundaries between buildings. Additionally examples in lines 5–6 of Fig. 7 show that although LPSNet can segment large buildings, it misses many smaller buildings. BiseNetv1 and STDC learn building contours better by enhancing detail features. However, their segmentation boundaries are less uniform than those of our model. Rows 7–9 in Fig. 7 highlight dense buildings with clear boundaries. The models other than our model struggle to identify small buildings in blocks with uneven boundaries in their segmentation.

(2) *Results on the WHU Aerial Image Dataset:* From Table 2, we can draw the following conclusions: (1) Our method outperforms mainstream models in terms of PA, Recall, F1-Score, IoU, mIoU, and other segmentation metrics. With an IoU of 81.34%, our method surpasses the state-of-the-art (SOTA) PIDNet by 15.25%, demonstrating the model's effective interaction with salient building features; (2) The IoU of other models, such as BiseNetv2 (69.72%), DDRNet (69.89%), LPSNet (69.84%), and FFNet (76.08%) show a common trend. These models use various convolution kernels to enhance detail, context, and local feature learning, thereby improving pixel discrimination and reducing computational costs. However, they struggle to segment boundaries accurately in complex remote sensing scenes, leading to lower IoU scores. Our model addresses this issue by leveraging building contours and geometric spatial features to mitigate information loss caused by receptive field changes, particularly for dense small buildings. We further improve segmentation accuracy using Loc loss and boundary loss, which efficiently guide gradient optimization; (3) Another key advantage is that our model achieves 114.26 FPS with only 2.397M parameters, which is 1/5 of STDC's parameter count. Therefore, our model strikes an optimal balance between inference speed and segmentation accuracy; (4) Finally, in terms of the overall ES index, our model ranks second with a score of 66.58, only 0.75 points lower than STDC. Regardless, our model has lower FLOPs (1.814G) than STDC, achieving a strong balance in the comprehensive evaluation. For the qualitative analysis of the WHU Aerial Image Dataset, we selected nine pairs of images

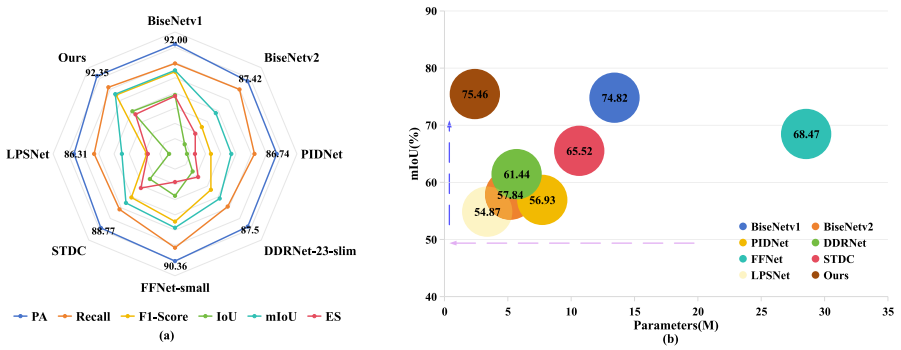


Fig. 6. Visualization of Massachusetts Builds Dataset performance. (a) Radar chart with lines farther from the center indicating better performance and (b) bubble chart with the arrow directions indicating performance.

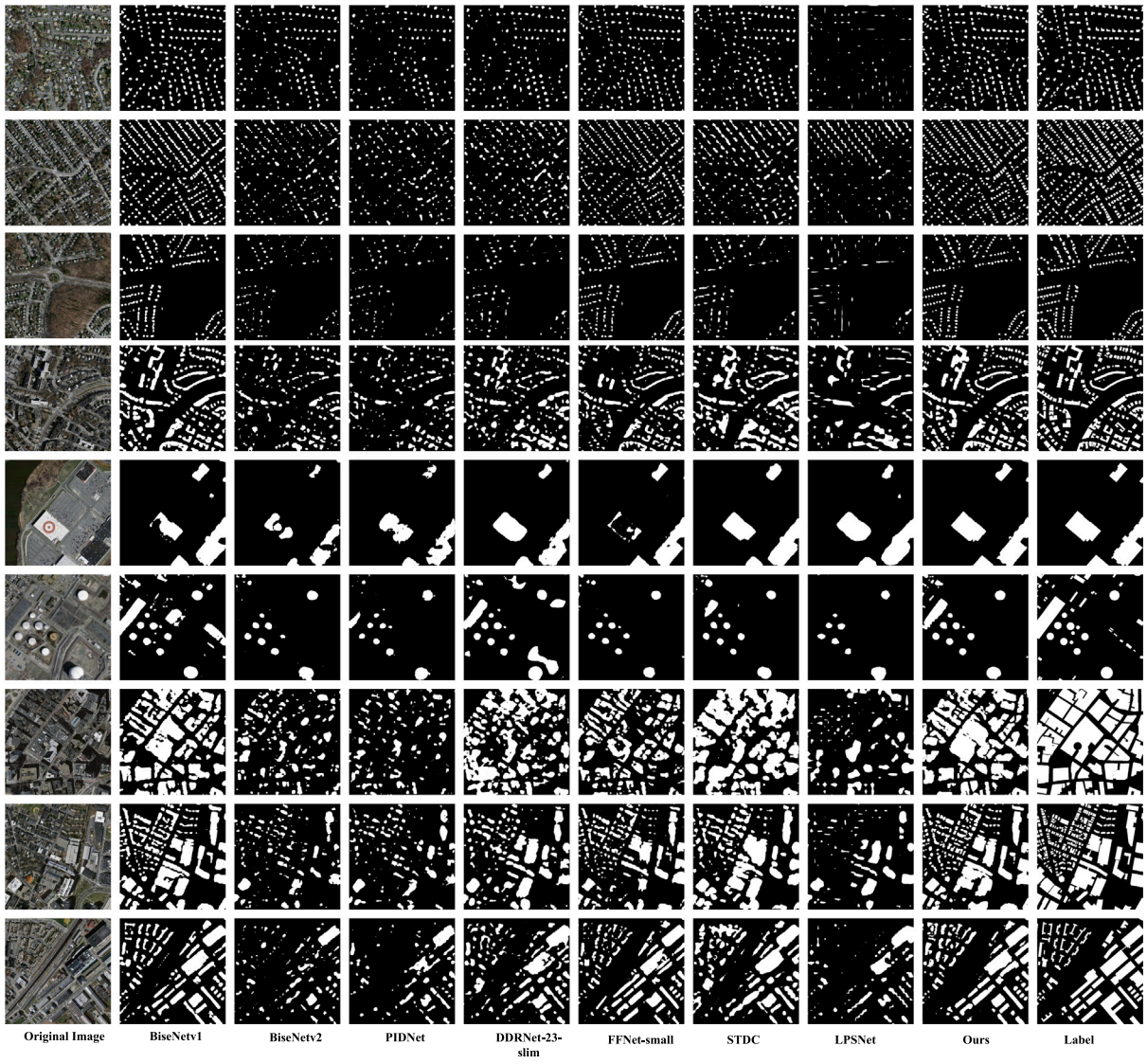


Fig. 7. Massachusetts builds dataset visual comparisons.

covering multi-view, regular/irregular, and dense buildings, as shown in Fig. 8. Rows 1–4 feature dense buildings with abundant vegetation and similarly shaped structures, posing significant challenges for accurate segmentation. LPSNet and STDC tend to merge continuous buildings into blocks, whereas BiseNetv1, DDRNet, and FFNet can segment medium-to-large buildings, but produce irregular building contours with occasional missed and false segmentations. In contrast, our method reduces missed segmentations and fits building contours more precisely, particularly for dense buildings. Row 4 highlights how our method accurately segments non-salient buildings and conforms to building boundaries, where other methods struggle. Rows 5–9 test

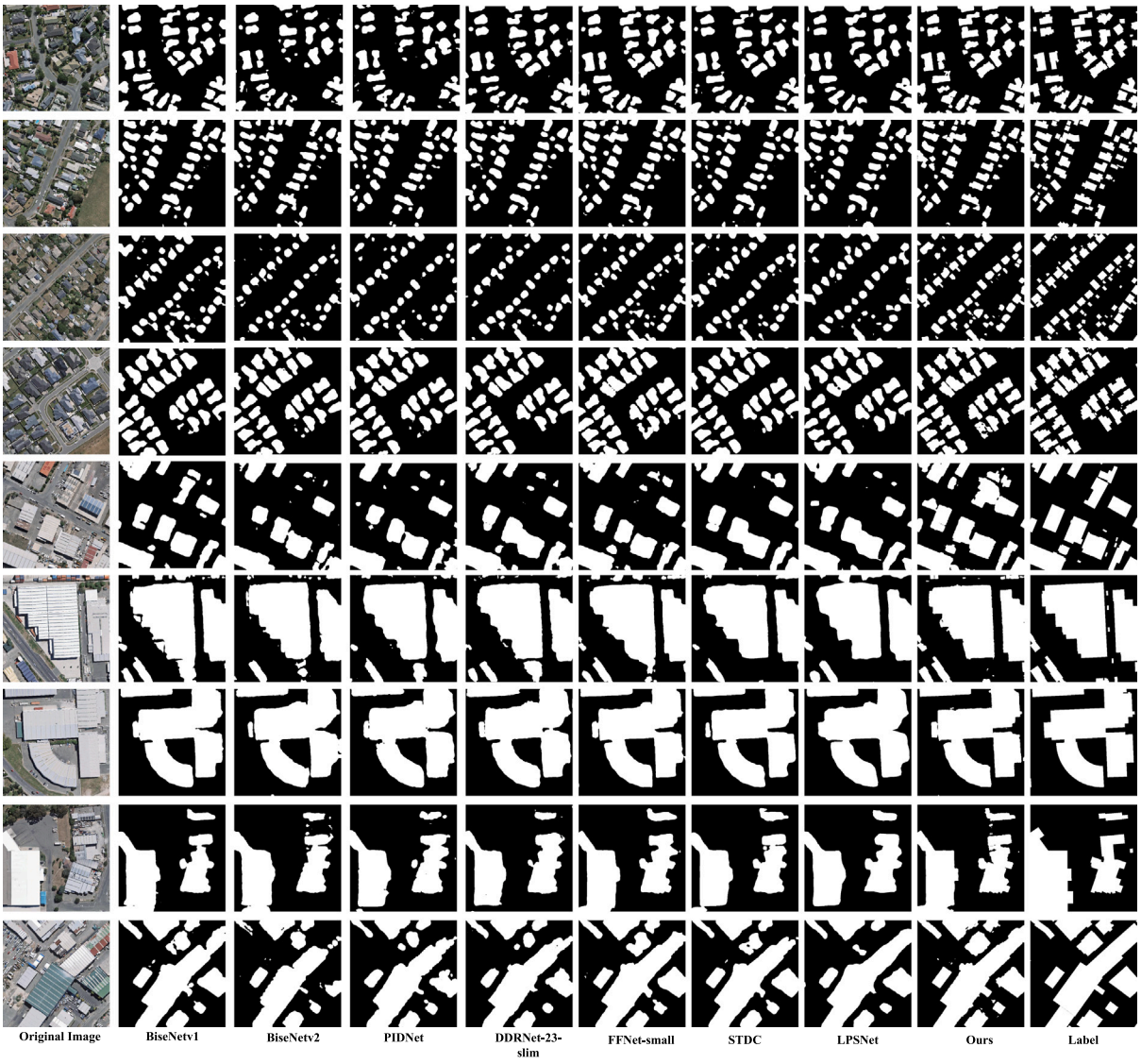


Fig. 8. WHU aerial image dataset visual comparisons.

Table 3
Comparison of various real-time semantic segmentation algorithms on the Potsdam Dataset (retaining BN layers during FPS testing, the optimal results are shown in bold and suboptimal results are shown underlined).

Method	Reference	Param (M)	FLOPs (G)	Resolution	FPS (Torch)	PA	Recall	F1-Score	IoU	mIoU	ES
BiseNetv1	ECCV 2018	13.419	15.261	512 × 512	238.55	95.46	**95.58**	93.26	87.37	91.34	71.31
BiseNetv2	IJCV 2021	5.182	17.699	512 × 512	162.49	92.98	90.55	86.32	75.94	83.46	62.13
PIDNet	CVPR 2023	7.717	6.341	512 × 512	118.56	92.12	87.19	84.98	73.89	81.87	55.93
DDRNet	Arxiv 2021	5.693	4.580	512 × 512	136.70	93.84	88.55	88.57	79.48	84.70	62.23
FFNet	CVPR 2022	28.544	16.466	512 × 512	147.79	94.15	89.07	89.12	80.37	86.33	51.70
STDC	CVPR 2021	10.637	15.900	512 × 512	**245.76**	95.73	91.19	92.14	85.43	89.87	71.76
LPSNet	IJCV 2023	3.372	**2.229**	512 × 512	138.03	94.34	90.20	89.37	80.78	86.68	68.86
Ours	–	**2.397**	14.086	512 × 512	130.62	**97.20**	95.35	**94.77**	**90.06**	**93.16**	**72.02**

the ability of the models to handle large, regular buildings. Excluding our method, all other models struggled to fit the building contours properly, resulting in irregular and overly smooth boundary details. Our approach, aided by the SDCM and OFAM, performs multi-scale feature learning, shallow detail extraction, and high-level semantic fusion, significantly improving contour fitting and model robustness.

(3) ***Results on the Potsdam Dataset:*** From Table 3, we can draw the following conclusions: (1) Our model outperforms SOTA lightweight models in terms of four key segmentation metrics: PA, F1-Score, IoU, and mIoU. Notably, our method achieved an impressive IoU of 90.06%, surpassing the SOTA models PIDNet and LPSNet by 16.17% and 9.28%, respectively. This result highlights the exceptional ability of the DCM to make precise decisions based on salient features when handling large buildings, enabling the model to focus on and accurately distinguish similar features of larger structures effectively; (2) Representative models, such as BiseNetv1, STDC, and FFNet achieved IoUs of 87.37%, 85.43%, and 80.37%, respectively. BiseNetv1 and STDC enhance the interactions between high-level semantics and low-level details, whereas FFNet employs various convolutional kernels for global feature extraction. However, for remote-sensing

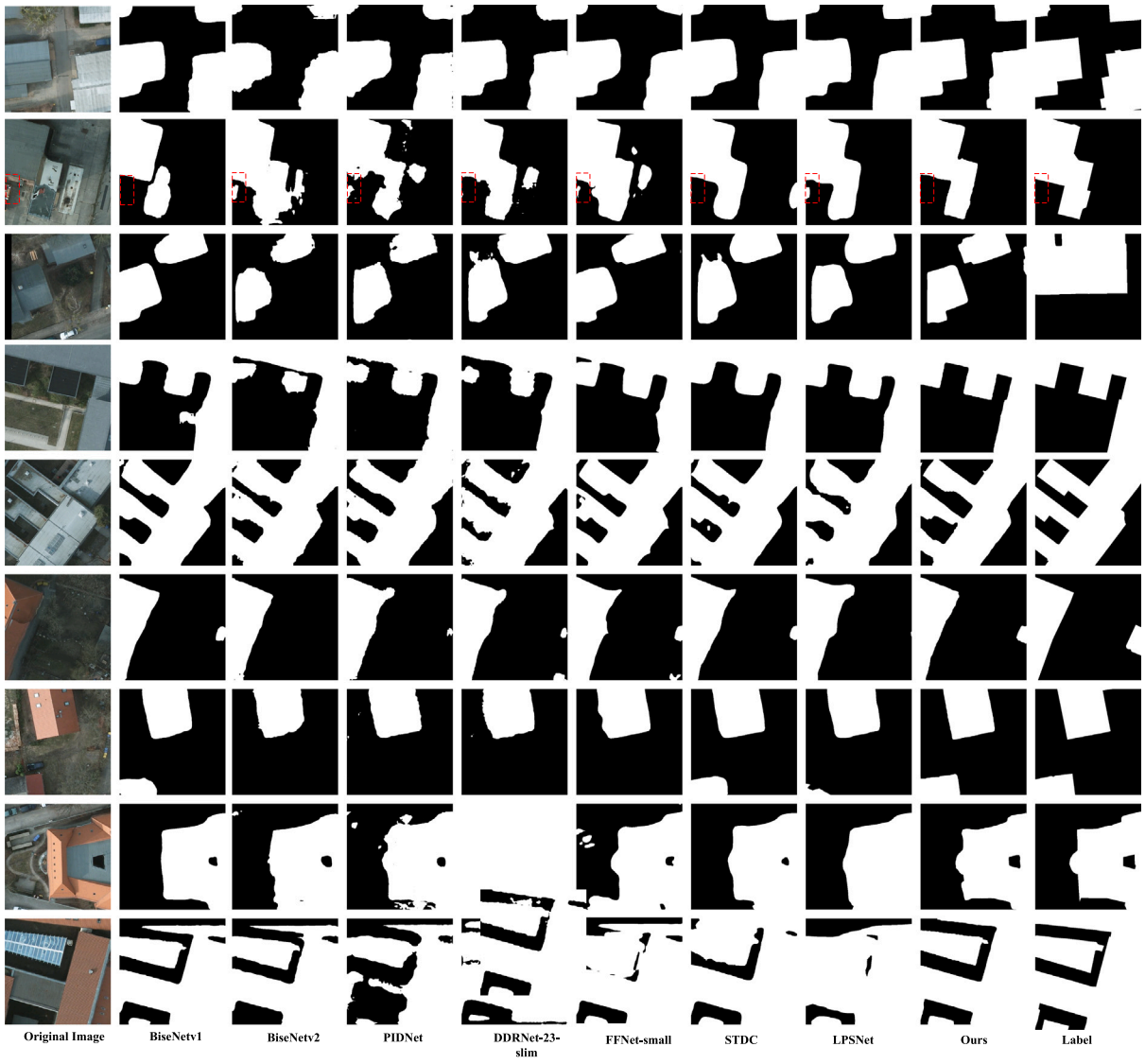


Fig. 9. Potsdam dataset visual comparisons.

images of large buildings, the key challenge is to distinguish similar features between adjacent structures. Previous methods fall short in terms of effective feature decision-making. In contrast, our model introduces a more efficient feature interaction mechanism. We utilize the OFAM to address up-sampling artifacts, utilize learnable offset parameters for self-guided feature alignment, and integrate the lightweight CFAM in the decoding stage. The DCM enables the network to leverage deep semantic features fully, achieving the precise discrimination of similar building features; (3) Finally, regarding the overall ES metric, our model achieved an ES of 72.02, which is the best result among all compared methods, effectively balancing Param, FLOPs, and accuracy. Although our model has the lowest Param (2.397M) among all methods, its FPS performance ranks in the middle. This result can be attributed to the SDCM's efficient guidance of salient features, which incurs the cost of frequent channel-adaptive decision-making, leading to higher memory access overhead. For the qualitative analysis of the Potsdam Dataset, we selected nine images featuring similar terrains, regular structures, and complex buildings, as shown in Fig. 9. Rows 1 to 5 feature gray buildings, where the terrain and buildings have similar distributions, presenting a significant challenge for accurate segmentation. In row 2, only our method, STDC, and LPSNet achieve fine-grained boundary segmentation. Other models such as BiseNetv2, PIDNet, DDRNet, and FFNet not only fail at boundary segmentation but also misclassify background features as buildings. However, our method accurately distinguishes the non-salient buildings (highlighted in red boxes). A notable result is shown in row 3, where despite incorrect ground-truth labels, our method successfully segments two non-contiguous buildings and fits the contours precisely. This result can be attributed to the effective guidance of the DCM and feature supervision provided by the boundary loss. Finally, in rows 5 to 9, we assess the ability to discern boundaries in complex structures with well-defined contours. The building in row 8, which contains trapezoidal non-building pixels, challenges the feature extraction and contour fitting abilities of the models. Excluding BiseNetv1 and the proposed method, the other models struggled with contour fitting and produced blocky boundary artifacts. Our method excels at distinguishing non-contiguous non-building pixels, a task that BiseNetv1 cannot handle, a result of the efficient interaction between detail and contextual semantic features through the DCM and OFAM, which allows the proposed network to select salient features for supervised training adaptively.

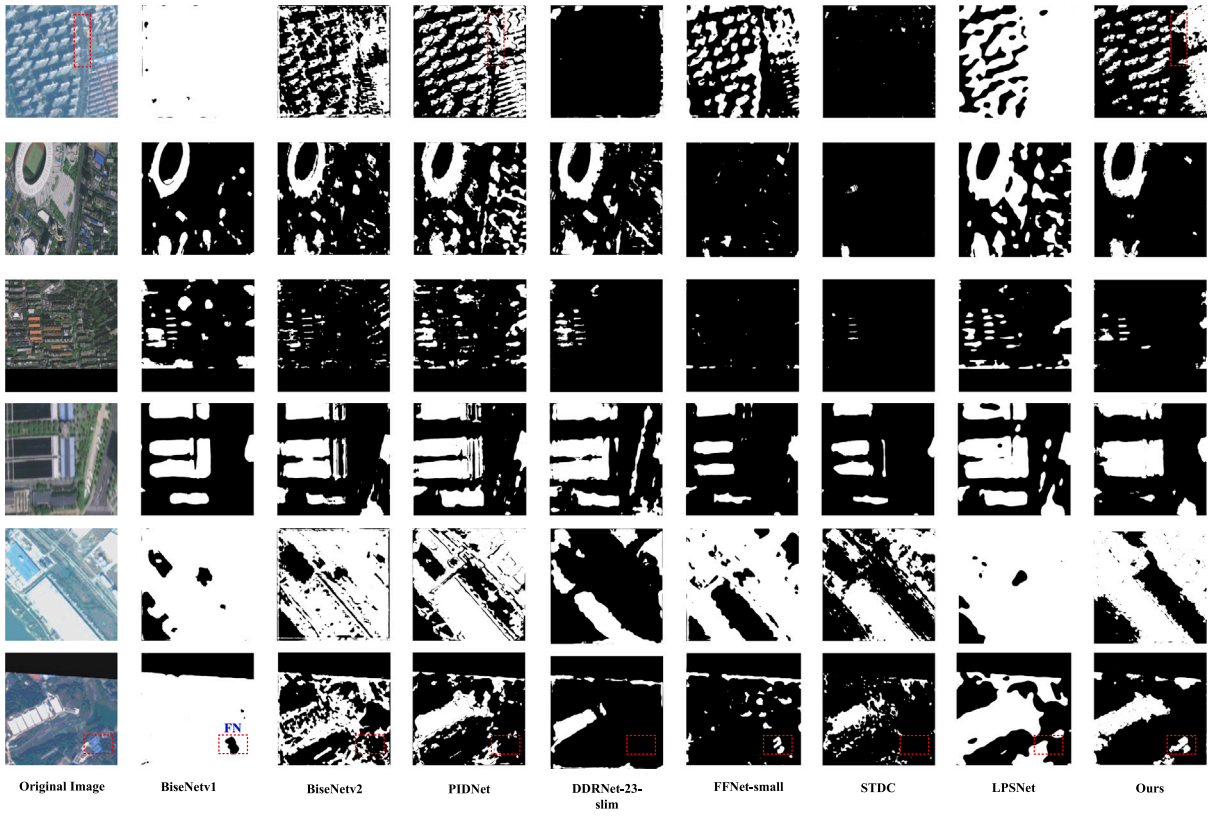


Fig. 10. Cross-domain building detection (Massachusetts → CScity) visualization results.

4.5. Cross-domain building detection for generalization performance analysis

(1) ***Qualitative Analysis of Direct Cross-domain Building Detection with Multiple Models:*** For cross-domain building detection analysis, we considered a model trained on the Massachusetts Builds Dataset as a case study for direct inference and qualitative evaluation. We selected six real-world images, covering both urban and suburban areas with diverse scenarios, including large buildings, complex and dense urban scenes, and regular buildings. This selection provided as a rigorous test of the generalization capabilities of the models. As shown in Fig. 10, in rows 2, 5, and 6, our model significantly outperformed the other methods for detecting buildings such as the He Long Sports Center and suburban factory buildings. In particular, for the He Long Sports Center, our model not only fits the circular shape of the building more accurately but also reduces the misclassification of surrounding dense building clusters, an issue commonly observed in the other models. This improvement can mainly be attributed to the effectiveness of our SDCM, which successfully captured the salient features of buildings while suppressing irrelevant features. In rows 5 and 6, excluding our model, the other models tended to misidentify elongated regions with spatial distributions similar to those of buildings. Additionally, they failed to recognize buildings in suburban areas, where the distribution was more scattered (as indicated by the red boxes). In contrast, our OFAM uses learnable offset tensors to aggregate multi-scale features efficiently and semantically align them. For dense building clusters, as shown in row 1, both our model and PIDNet successfully detected buildings. However, PIDNet tended to group contiguous buildings into blocks (highlighted by red boxes), whereas the proposed method detected individual buildings more accurately. Furthermore, our model successfully perceived low-rise buildings along roads, whereas PIDNet misclassified them. This result can be attributed to the CFAM in the decoding stage of our model, which effectively utilizes salient features for guidance and combines multi-branch convolutions with varying receptive fields to enable multi-scale feature understanding. However, as shown in row 4, when the distribution of building features differs significantly from that of other land cover types, excessive reliance on salient feature guidance and multi-scale feature aggregation can cause the proposed model to converge to a local optimum, resulting in reduced segmentation performance.

(2) Cross-domain building detection feature response analysis across various models: In this section, we further explore how features respond when a model is trained on the Massachusetts Builds, WHU Aerial, and Potsdam datasets, and then applied to real-world scenarios, as shown in Fig. 11. (1) When using the Massachusetts Builds Dataset as the training set, all models correctly responded to the dense building clusters in columns 1 and 3. However, in columns 2 and 4–6, excluding our method, the other models incorrectly responded to non-building features and exhibited weak responses to large buildings. The Massachusetts Builds dataset primarily consists of dense building clusters, which lack the diverse large building features necessary for effective learning; (2) When using the WHU Aerial Dataset as the training set, as shown in column 1, only our model and PIDNet were able to provide localized responses, while other models failed to focus on the building clusters. In column 3, for densely distributed and regularly shaped buildings, our model not only successfully segmented four buildings with significant differences from the surrounding land cover features but also accurately segmented buildings with features similar to those of the land cover, demonstrating strong feature response amplitudes. This ability is not possessed by the other methods and can

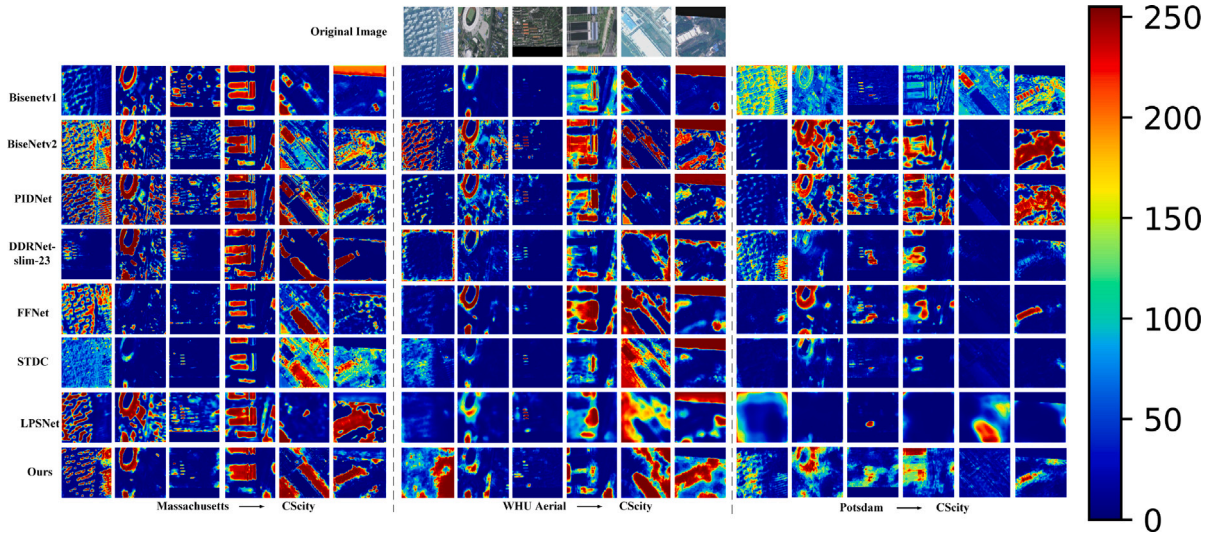


Fig. 11. Cross-domain building detection feature response visualization across multiple datasets.

be attributed to the efficient feature interaction and aggregation enabled by our DCM; (3) When using the Potsdam Dataset for training, the lack of dense small buildings and significant resolution inconsistencies result in ineffective feature responses during cross-domain inference. Excluding the proposed method, the other models failed to produce meaningful responses. In contrast, our method consistently achieved effective feature responses across all three datasets. This result can be attributed to our efficient feature aggregation module and feature guidance techniques (e.g., $\mathcal{L}_{bl}$ and $\mathcal{L}_{loc}$), which enhance feature interactions and improve model generalization. As shown in Fig. 12, our approach facilitates multi-scale feature aggregation and responses through collaboration between boundary features and heatmap center features, ensuring effective feature alignment and addressing the issue of insufficient feature learning in cross-domain building detection.

5. Ablation studies

(1) Effectiveness of the proposed loss function: To validate the effectiveness of the proposed hybrid loss function, we conducted ablation experiments using the WHU Aerial Image Dataset with experimental parameters matching those described in Section 4.2. Table 4 presents the quantitative results of these experiments and Fig. 13 presents the visual performance improvements at different loss stages. Starting with L1, the IoU was only 42.98%, and the Recall was 43.23%, indicating poor performance in dense urban areas, as shown in line 3 of Fig. 13. The model failed to respond to dense buildings accurately, instead picking up noise from the surrounding shadows, trees, and vegetation, leading to erroneous gradients and negative optimization. However, the model performed better for regular large buildings, although it still struggled with background features and treated them similarly to buildings. For L2, the IoU increases to 75.41% and Recall rises to 81.56%, showing significant improvement. The SSA loss helps reduce noise across multi-scale feature maps, making the network more sensitive to foreground features. As shown in line 4 of Fig. 13, L2 effectively extracts features from dense buildings and improves the contour fitting for regular buildings by focusing on building features and minimizing attention on non-building areas. L3 further improved the performance, yielding an IoU of 76.66% and Recall of 83.00%, a 1.26% and 1.44% increase over L2. L3 incorporates the SSA loss to refine building boundaries at the pixel level and reduce noise from small-scale feature upsampling. As shown in line 6 of Fig. 13, L3 responds better to dense buildings and accurately extracts regular building features. Finally, adding Loc loss to L3 for L4 yielded an IoU of 81.34% and Recall of 89.67%, significantly improving feature extraction, particularly for dense buildings. Loc loss addresses the complexity of remote sensing images, guiding the model to focus on foreground features. Line 7 of Fig. 13 shows that L4 enhances the model's ability to fit building contours, particularly for regular large buildings, by prioritizing building centers during feature extraction.

Table 4
Loss function ablation experiment (training on the WHU Aerial Image Dataset, optimal results are shown in bold).

Method	$\mathcal{L}_{seg}$	$\mathcal{L}_{SSA}$	$\mathcal{L}_{bl}$	$\mathcal{L}_{loc}$	PA	Recall	F1-Score	IoU	mIoU
(L1)	✓				85.48	43.23	60.12	42.98	63.34
(L2)	✓	✓			96.71	81.56	85.99	75.41	85.88
(L3)	✓	✓	✓		96.93	83.00	86.79	76.66	86.62
(L4:Ours)	✓	✓	✓	✓	**97.72**	**89.67**	**89.71**	**81.34**	**89.40**

Table 5
Module ablation experiment (Baseline: BiseNetv1, replacing modules in BiseNetv1 with modules designed by us stage by stage, the optimal results are shown in bold and suboptimal results are underlined).

Method	DCM	OFAM	CFAM	PA	Recall	F1-Score	IoU	mIoU
Baseline				97.40	87.56	88.42	79.25	88.18
Stage1	✓			98.12	**94.17**	91.28	83.96	90.94
Stage2	✓	✓		<u>98.14</u>	91.77	<u>91.58</u>	**84.47**	<u>91.20</u>
Stage3	✓	✓	✓	**98.17**	<u>93.27</u>	**91.59**	<u>84.45</u>	**91.22**

(2) Effectiveness of the proposed modules: To evaluate the effectiveness of each module in our network, we conducted an ablation study using BiseNetv1 as the baseline on the WHU Aerial Image Dataset, with parameters consistent with those described in Section 4.2. Feature visualization was performed using the CAM to observe the response of the model to building features. The baseline was pre-trained, whereas all other models were trained from scratch. Table 5 reveals that the baseline achieved an IoU of 79.25% and Recall of 87.56%. As shown in Fig. 14 (column 2), BiseNetv1 has a limited response to building foregrounds, especially in dense areas. In Stage 1, replacing the Conv + BN + ReLU block with the DCM improved the IoU to 83.96% (increase of 4.71%) and Recall to 94.17% (increase of 6.61%). This result demonstrates that the DCM enhances the ability of the model to focus on building features, particularly for dense structures. In Stage 2, the IoU increased to 84.47% and Recall increased to 91.77%,

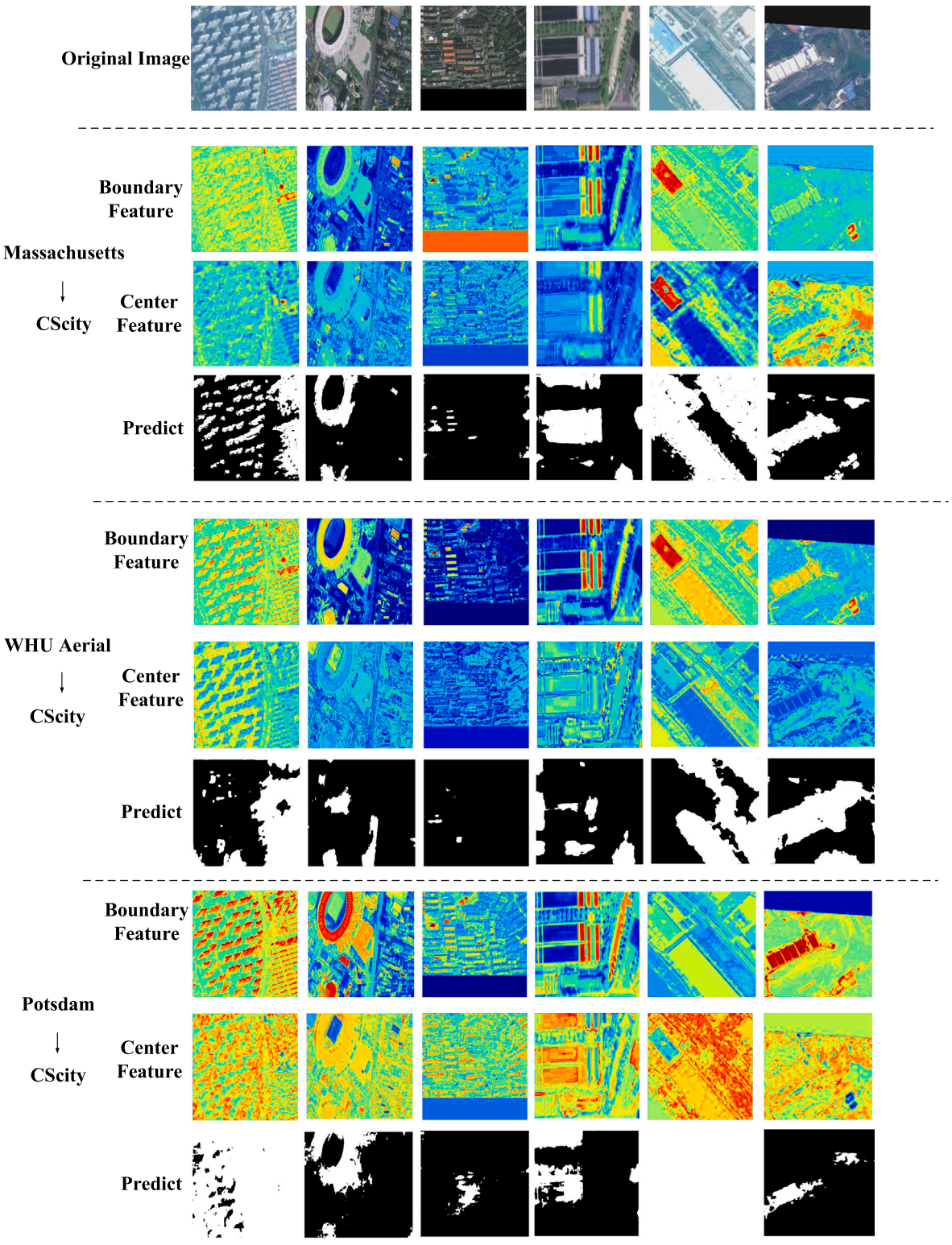


Fig. 12. Cross-domain building detection feature visualization results of our method across multiple datasets.

demonstrated that the OFAM reduced feature ambiguity between the foreground and background, improving the segmentation performance for both dense and large buildings. Stage 3 achieved an IoU of 85.45% and Recall of 93.27%, providing optimal performance across all five segmentation metrics (PA: 98.17%, F1-Score: 91.59%, mIoU: 91.22%). These result confirm that the proposed method effectively reduces the model parameters while maintaining high segmentation accuracy. Finally, adding the CFAM (column 5 in Fig. 14) further improves feature attention, particularly for building contours, highlighting its ability to extract both local and global contextual features.

(3) Analysis of key parameters: In this section, sensitivity analysis is performed on the $\mathcal{L}_{SSA}$ weight λ_1 and loss function weights $\lambda_{loc4}, \delta, \beta$ in $\mathcal{L}_{loc}$. λ_1 ranges from 0.1 to 2.0, λ_{loc4} ranges from 0.01 to 1.5, δ ranges from 0.05 to 1.0, and β ranges from 0.5 to 4.0. $\lambda_1 = 0.5$, $\lambda_{loc4} = 0.2$, $\delta = 0.25$, and $\beta = 2.0$ were used as baseline parameters. The

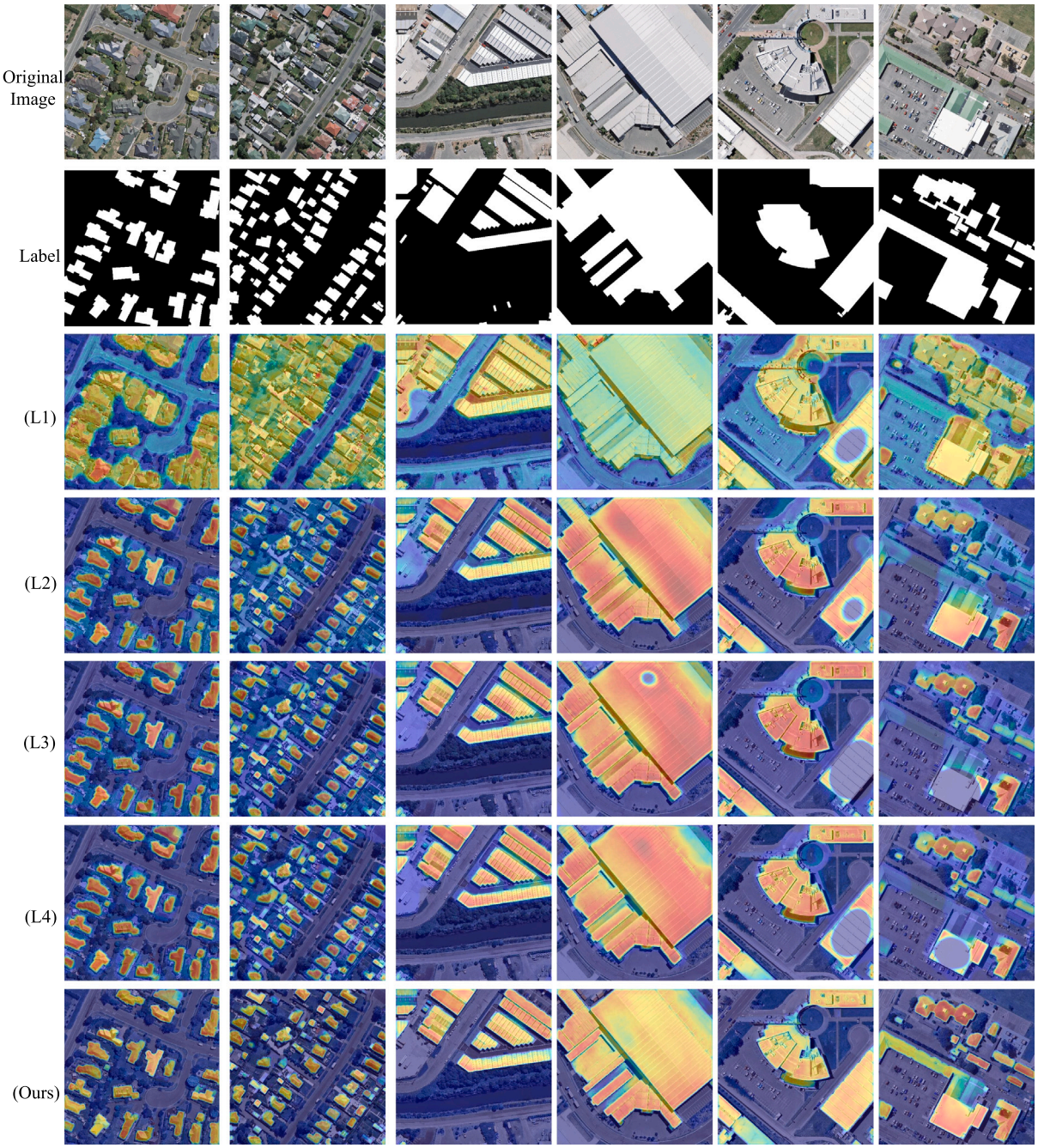


Fig. 13. Results of loss function ablation experiment visualization.

Table 6
Ablation experiments with sensitive parameters (the optimal results are shown in bold).

Method	λ_1	λ_{loc4}	δ	β	PA	Recall	F1-Score	IoU	mIoU
(I)	0.10	0.01	0.05	0.50	96.09	76.28	84.18	72.68	84.16
(II)	0.20	0.05	0.10	1.00	96.19	77.03	84.50	73.15	84.45
(III)	0.25	0.10	0.20	1.50	96.39	78.25	85.19	74.20	85.09
(IV) Baseline	**0.50**	**0.20**	**0.25**	**2.00**	**97.72**	**89.67**	**89.71**	**81.34**	**89.40**
(V)	1.00	0.50	0.30	2.50	96.85	81.99	86.58	76.33	86.41
(VI)	1.25	0.80	0.40	3.00	96.43	79.08	85.12	74.09	85.06
(VII)	1.50	1.00	0.50	3.50	96.83	81.83	86.55	76.29	86.38
(VIII)	2.00	1.50	1.00	4.00	96.70	80.64	86.14	75.65	85.99

experimental parameter configuration was consistent with Section 4.2, and the experimental results are listed in Table 6. With decreases in $\lambda_1, \lambda_{loc4}, \delta, \beta$, we found that the values of PA, Recall, F1-Score, and IoU all decreased to a certain extent. However, PA decreased slowly, whereas Recall, F1-Score, and IoU decreased rapidly, indicating that the $\mathcal{L}_{SSA}$ factor in λ_1 has a significant impact on the segmentation performance of the network. $\mathcal{L}_{SSA}$ can eliminate the influence of different scales of feature distributions on the segmentation performance

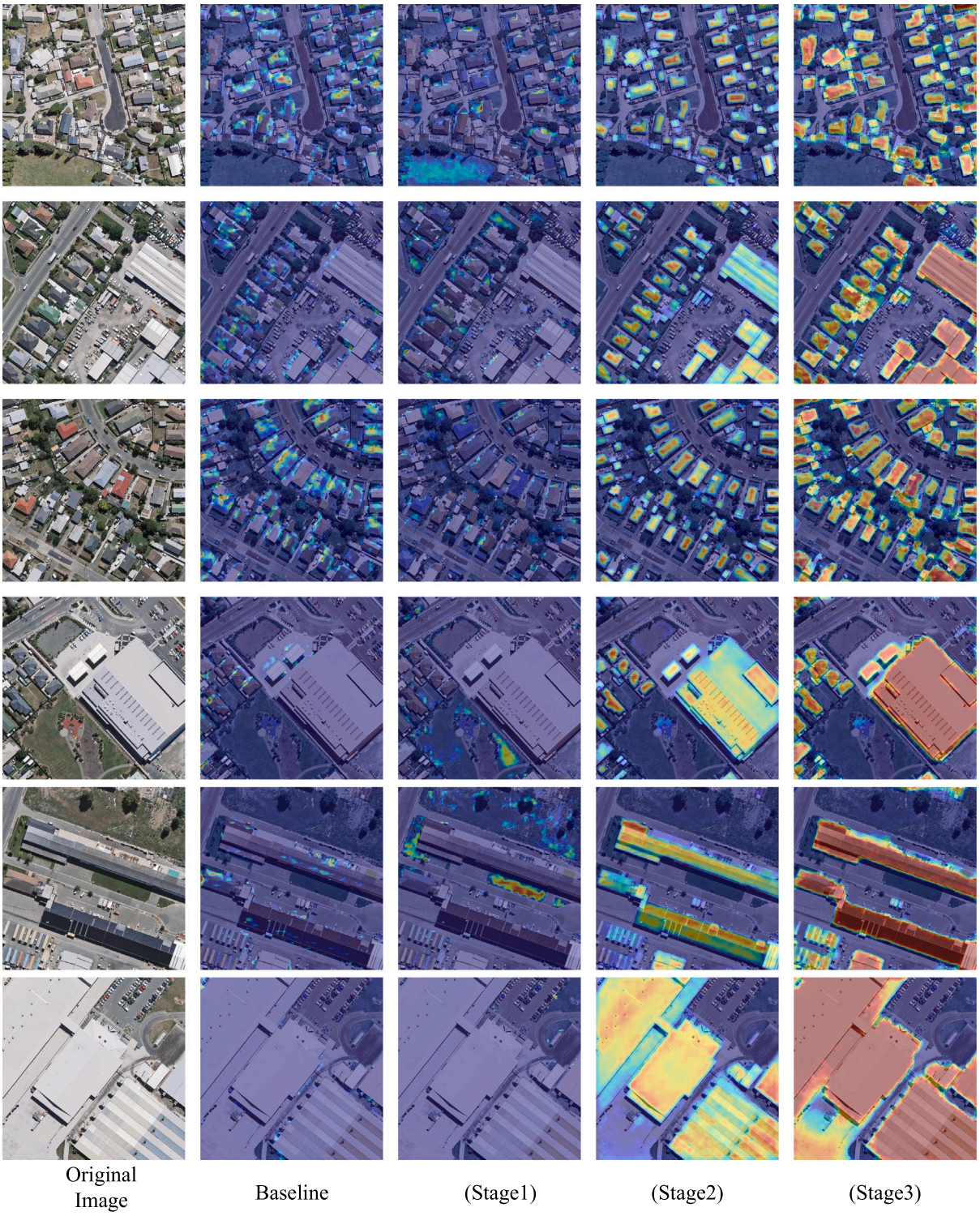


Fig. 14. Visualization of the results of the module ablation experiment.

through feature alignment causing the network to pay more attention to salient foreground features. As λ_1 decreases, the model's focus on background features grows, leading to a drop in overall segmentation accuracy. With increases of $\lambda_{loc4} and \beta$, the Recall and F1-Score decrease more steeply, whereas PA decreases slowly. This result indicates that the model has a certain pixel discrimination ability; however, the Recall is poor as a result of the insufficient extraction of building features. As shown in Fig. 15 (column 6 to both ends), the increases in FP(blue) and FN(yellow) indicate pixel discrimination errors, suggesting that the salient foreground features did not effectively guide the network during training.

6. Limitations and future research

Although we achieved competitive performance on several benchmark public datasets, there are still some limitations of our study. (1) Real-time performance requires improvement. The SDCM splits the channel dimensions during the encoding-decoding process to identify

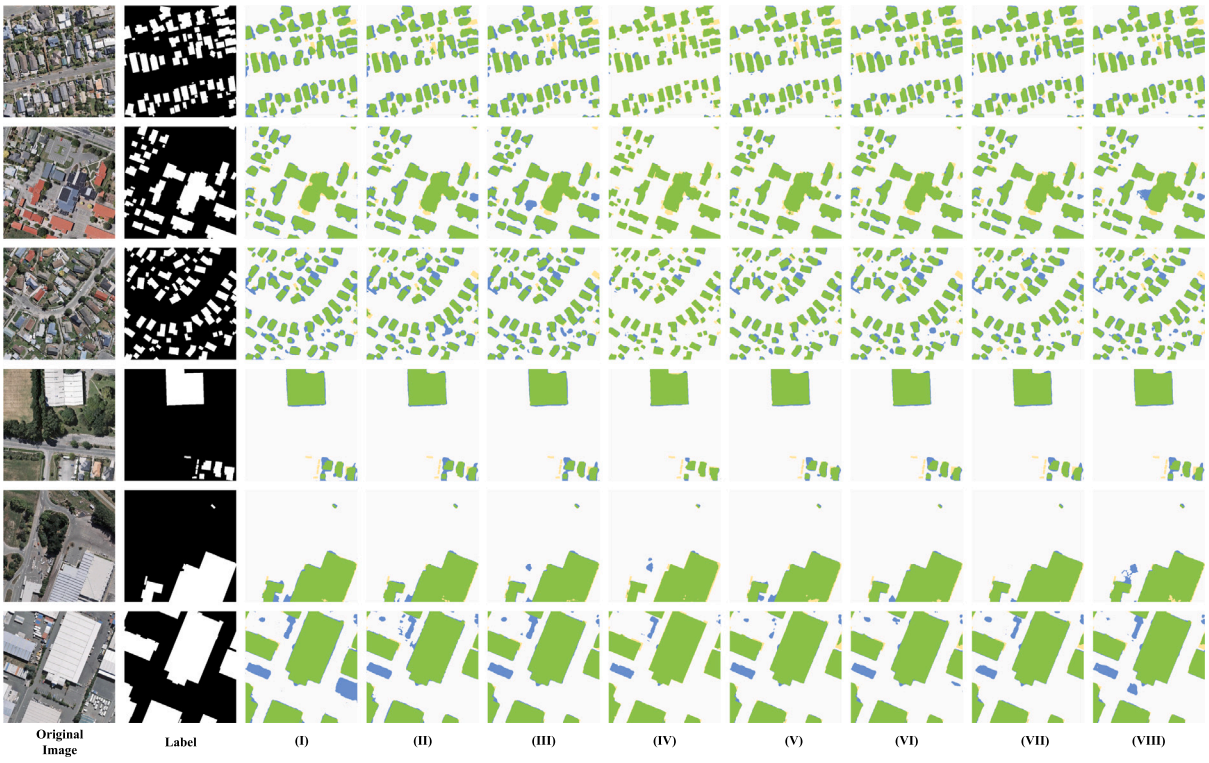


Fig. 15. Visualization results of ablation experiments for key parameters.

the optimal salient features that affect real-time efficiency, resulting in reduced processing speed. However, considering both the overall parameter count and segmentation accuracy, our method remains the optimal choice in most cases. (2) In some specific scenarios, such as when there is a significant disparity between the distribution of buildings and other objects, excessive salient feature guidance may cause the model to become trapped in a local minimum, negatively impacting segmentation performance. (3) Our experimental analysis revealed that the model tends to smooth the boundaries at the building junctions, leading to blocky boundary pixels.

To address these limitations, our future research will focus on the following three areas: (1) In the model inference phase, we will explore using tensor dot product operations combined with re-parameterization techniques to avoid frequent channel splitting while maintaining salient feature guidance to improve real-time performance; (2) We will designing prior knowledge to enhance the model's feature perception of buildings in complex scenes, thereby improving its generalization ability. (3) Future work will explore the use of diffusion models to generate boundary guidance, thereby enhancing the model's ability to fit irregular building boundaries.

7. Conclusion

In this paper, we proposed SFGNet, which is a salient-feature-guided real-time building extraction network. By designing the efficient DCM, the model effectively learns shallow details and contour features, whereas the SDCM adaptively makes high-response feature decisions by self-learning Gaussian sequence parameters to reduce computational complexity. As a plug-and-play component, the SDCM can be integrated into any network. To aggregate high-level semantic and shallow detail information, the OFAM minimizes feature offset during up-sampling, thereby enhancing the model's ability to capture detail and contour features. In the decoding stage, the lightweight CFAM aggregates local and global features. SFGNet achieved a strong balance between inference speed and accuracy on both the Massachusetts Builds and WHU Aerial Image Datasets. Through effective supervision and the utilizing of efficient modules, SFGNet significantly improves real-time building segmentation by leveraging geometric features.

CRediT authorship contribution statement

Jin Kuang: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation. **Dong Liu:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was funded by Natural Science Foundation of Hunan Province (No. 2023JJ50392, No. 2023JJ50393), Scientific Research Fund of Hunan Provincial Education Department (No. 23A0588), and Aid Program for Science and Technology Innovative Research Team in Higher Educational Institutions of Hunan Province.

Data availability

Data will be made available on request.
Dataset linking: https://github.com/gasking/SFGNet.

References

[1] M. Gong, T. Liu, M. Zhang, Q. Zhang, D. Lu, H. Zheng, F. Jiang, Context–content collaborative network for building extraction from high-resolution imagery, Knowl.-Based Syst. 263 (2023) 110283.

[2] J. Wang, C. Gou, Q. Wu, H. Feng, J. Han, E. Ding, J. Wang, RTFormer: Efficient design for real-time semantic segmentation with transformer, Adv. Neural Inf. Process. Syst. 35 (2022) 7423–7436.

[3] Y. Hu, X. Ma, J. Sui, M.-O. Pun, PPMamba: A Pyramid Pooling Local Auxiliary SSM-Based Model for Remote Sensing Image Semantic Segmentation, 2024, arXiv preprint arXiv:2409.06309.

[4] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.

[5] Y. Hong, H. Pan, W. Sun, Y. Jia, Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes, 2021, arXiv preprint arXiv:2101.06085.

[6] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[7] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al., Deep high-resolution representation learning for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 43 (10) (2020) 3349–3364.

[8] D. Mehta, A. Skliar, H. Ben Yahia, S. Borse, F. Porikli, A. Habibian, T. Blankevoort, Simple and efficient architectures for semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2628–2636.

[9] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, N. Sang, Bisenet: Bilateral segmentation network for real-time semantic segmentation, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 325–341.

[10] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, N. Sang, Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation, Int. J. Comput. Vis. 129 (2021) 3051–3068.

[11] J. Xu, Z. Xiong, S.P. Bhattacharyya, Pidnet: A real-time semantic segmentation network inspired by pid controllers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 19529–19539.

[12] Q. Zou, C. Brenner, M. Sester, Gaussian process mapping of uncertain building models with GMM as prior, IEEE Robot. Autom. Lett. (2023).

[13] T. Zuo, J. Feng, X. Chen, HF-FCN: Hierarchically fused fully convolutional network for robust building extraction, in: Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part I 13, Springer, 2017, pp. 291–302.

[14] Q. Zhu, C. Liao, H. Hu, X. Mei, H. Li, MAP-net: Multiple attending path neural network for building footprint extraction from remote sensed imagery, IEEE Trans. Geosci. Remote Sens. 59 (7) (2020) 6169–6181.

[15] D. Feng, H. Chen, Y. Xie, Z. Liu, Z. Liao, J. Zhu, H. Zhang, Gccinet: Global feature capture and cross-layer information interaction network for building extraction from remote sensing imagery, Int. J. Appl. Earth Obs. Geoinf. 114 (2022) 103046.

[16] Y. Song, Z. Jing, M. Li, Siamese u-net with attention mechanism for building change detection in high-resolution remote sensing images, in: International Conference on Aerospace System Science and Engineering, Springer, 2021, pp. 487–503.

[17] F. Shi, T. Zhang, A multi-task network with distance–mask–boundary consistency constraints for building extraction from aerial images, Remote. Sens. 13 (14) (2021) 2656.

[18] X. Wang, B. Ma, Z. Qing, Y. Sang, C. Gao, S. Zhang, N. Sang, Cbr-net: Cascade boundary refinement network for action detection: Submission to activitynet challenge 2020 (task 1), 2020, arXiv preprint arXiv:2006.07526.

[19] S. Wei, T. Zhang, D. Yu, S. Ji, Y. Zhang, J. Gong, From lines to polygons: Polygonal building contour extraction from high-resolution remote sensing imagery, ISPRS J. Photogramm. Remote Sens. 209 (2024) 213–232.

[20] Y. Gao, X. Luo, X. Gao, W. Yan, X. Pan, X. Fu, Semantic segmentation of remote sensing images based on multiscale features and global information modeling, Expert Syst. Appl. 249 (2024) 123616.

[21] A. Paszke, A. Chaurasia, S. Kim, E. Culurciello, Enet: A deep neural network architecture for real-time semantic segmentation, 2016, arXiv preprint arXiv:1606.02147.

[22] H. Zhao, X. Qi, X. Shen, J. Shi, J. Jia, Icnet for real-time semantic segmentation on high-resolution images, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 405–420.

[23] H. Wang, X. Jiang, H. Ren, Y. Hu, S. Bai, Swiftnet: Real-time video object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 1296–1305.

[24] H. Li, P. Xiong, H. Fan, J. Sun, Dfanet: Deep feature aggregation for real-time semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9522–9531.

[25] S. Watanabe, T. Hori, S. Karita, Hayashi, et al., ESPnet: End-to-end speech processing toolkit, 2018, arXiv preprint arXiv:1804.00015.

[26] M. Gamal, M. Siam, M. Abdel-Razek, Shuffleseg: Real-time semantic segmentation network, 2018, arXiv preprint arXiv:1803.03816.

[27] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.

[28] H. Wei, X. Liu, S. Xu, Z. Dai, Y. Dai, X. Xu, DWRSeg: Rethinking efficient acquisition of multi-scale contextual information for real-time semantic segmentation, 2022, arXiv preprint arXiv:2212.01173.

[29] Y. Zhang, T. Yao, Z. Qiu, T. Mei, Lightweight and progressively-scalable networks for semantic segmentation, Int. J. Comput. Vis. 131 (8) (2023) 2153–2171.

[30] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

[31] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, et al., Searching for mobilenetv3, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1314–1324.

[32] Y. Tang, K. Han, J. Guo, C. Xu, C. Xu, Y. Wang, GhostNetv2: Enhance cheap operation with long-range attention, Adv. Neural Inf. Process. Syst. 35 (2022) 9969–9982.

[33] J. Chen, S.-h. Kao, H. He, W. Zhuo, S. Wen, C.-H. Lee, S.-H.G. Chan, Run, don't walk: chasing higher FLOPS for faster neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12021–12031.

[34] H. Liu, S. Ma, D. Xia, S. Li, Sfanet: A spectrum-aware feature augmentation network for visible-infrared person reidentification, IEEE Trans. Neural Netw. Learn. Syst. 34 (4) (2021) 1958–1971.

[35] S. Tang, T. Sun, J. Peng, G. Chen, Y. Hao, M. Lin, Z. Xiao, J. You, Y. Liu, Pp-mobileseg: Explore the fast and accurate semantic segmentation model on mobile devices, 2023, arXiv preprint arXiv:2304.05152.

[36] M.-H. Guo, C.-Z. Lu, Q. Hou, Z. Liu, M.-M. Cheng, S.-M. Hu, Segnext: Rethinking convolutional attention design for semantic segmentation, Adv. Neural Inf. Process. Syst. 35 (2022) 1140–1156.

[37] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.

[38] X. Li, J. Zhang, Y. Yang, G. Cheng, K. Yang, Y. Tong, D. Tao, Sfnet: Faster and accurate semantic segmentation via semantic flow, Int. J. Comput. Vis. 132 (2) (2024) 466–489.

[39] Z. Huang, Y. Wei, X. Wang, W. Liu, T.S. Huang, H. Shi, Alignseg: Feature-aligned segmentation networks, IEEE Trans. Pattern Anal. Mach. Intell. 44 (1) (2021) 550–557.

[40] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, J. Wang, OCNet: Object context for semantic segmentation, Int. J. Comput. Vis. 129 (8) (2021) 2375–2398.

[41] T.-Y. Ross, G. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2980–2988.

[42] V. Mnih, Machine Learning for Aerial Image Labeling, University of Toronto (Canada), 2013.

[43] S. Ji, S. Wei, M. Lu, Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set, IEEE Trans. Geosci. Remote Sens. 57 (1) (2018) 574–586.

[44] Y. Li, T. Shi, Y. Zhang, J. Ma, SPGAN-DA: Semantic-preserved generative adversarial network for domain adaptive remote sensing image semantic segmentation, IEEE Trans. Geosci. Remote Sens. (2023).

[45] H. Huang, J. Liu, R. Wang, Easy-net: A lightweight building extraction network based on building features, IEEE Trans. Geosci. Remote Sens. (2023).

[46] M. Fan, S. Lai, J. Huang, X. Wei, Z. Chai, J. Luo, X. Wei, Rethinking bisenet for real-time semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 9716–9725.