# ESPC_NASUnet: An End-to-End Super-Resolution Semantic Segmentation Network for Mapping Buildings From Remote Sensing Images

Penglei Xu, Hong Tang, Jiayi Ge, and Lin Feng

***Abstract*—Higher resolution building mapping from lower resolution remote sensing images is in great demand due to the lack of higher resolution data access, especially in the context of disaster assessment. High resolution building layout map is crucial for emergency rescue after the disaster. The emergency response time would be reduced if detailed building footprints were delineated from more easily available low-resolution data. To achieve this goal, we propose a super-resolution semantic segmentation network called ESPC_NASUnet, which consists of a feature super-resolution module and a semantic segmentation module. To the best of authors' knowledge, this is the first work to systematically explore a deep learning-based approach to generate semantic maps with higher spatial resolution from lower spatial resolution remote sensing images in an end-to-end fashion. The experimental results for two datasets suggest that the proposed network is the best among four different end-to-end architectures in terms of both pixel-level metrics and object-level metrics. In terms of pixel-level $F1$-score, the improvements are greater than 0.068 and 0.055. Regarding the object-level $F1$-score, the disparities between ESPC_NASUnet and other end-to-end methods are more than 0.083 and 0.161 in the two datasets, respectively. Compared with stage-wise methods, our end-to-end network is less impacted by low-resolution input images. Finally, the proposed network produces building semantic maps comparable to those generated by semantic segmentation networks trained with high-resolution images and the ground truth utilizing the two datasets.**

***Index Terms*—Building extraction, end-to-end network, remote sensing, super-resolution semantic segmentation (SRSS).**

## I. Introduction

REMOTE sensing image interpretation is an important way to delineate buildings for urban planning. The poor efficiency and time-consuming nature of artificial interpretation have made automatic and semiautomatic building extraction algorithms hot topics in the last decades [1], [2]. With the development of remote sensing imaging technologies, the spatial resolution of acquired data continues to improve. Thus, building footprints extracted from remote sensing images are becoming more detailed. For instance, images with resolution of hundreds or tens of meters, e.g., MODIS, are often exploited to identify large-scale built-up areas on the earth's surface [3], [4]. Individual buildings can be delineated from meter- or submeter-resolution images, e.g., WorldView, QuickBird, or UAV aerial images [5], [6]. In some emergencies with time limitations such as disaster assessment, individual buildings need to be delineated [7] as quickly as possible. However, it is difficult to obtain high-resolution (HR) image quickly. By contrast, some data with lower spatial resolution are open access. If these data could be utilized to produce semantic maps of buildings, the difficulty of HR data acquisition could be avoided.

Previous research works have focused on combining traditional machine learning algorithms such as support vector machine [8] and handcrafted features such as the morphological building index [9] and morphological shadow index [10] to solve the problem of building extraction [11]. As remote sensing data volume and complexity increase, traditional methods cannot obtain superior performance. However, the development of deep learning (DL) has catalyzed a great revolution in the processing of remote sensing data and building extraction. The application of convolutional neural networks (CNNs) to semantic segmentation (SS) can tremendously increase the accuracy of built-up mapping. The fully convolutional neural network (FCN) is the first high-profile CNN-based SS network [12]. An encoder–decoder structure further improves the effect of SS; typical networks are SegNet [13] and U-Net [14]. The latest DeepLab V3+ [15] of the DeepLab series outperformed many state-of-the art SS networks on two widely used datasets in 2018. Motivated by the abovementioned work, various DL-based methods aimed at building footprint extraction have been proposed. Paisitkriangkrai *et al.* [16] presented the first attempt to apply CNN and conditional random fields to remote sensing image pixel labeling. The work demonstrated the effectiveness of CNNs for building extraction. However, handcrafted features and random forest were still utilized to increase the performance due to the weak representation ability of shallow CNNs. Subsequently, an end-to-end learning method based on of FCN was proposed in [17] to delineate different objects on earth.

Manuscript received January 7, 2021; revised March 4, 2021 and April 22, 2021; accepted May 8, 2021. Date of publication May 12, 2021; date of current version June 4, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 41971280 and in part by the National Key R&D Program of China under Grant 2017YFB0504104. *(Corresponding author: Hong Tang.)*

The authors are with the State Key Laboratory of Remote Sensing Science, Faculty of Geographical Science, Beijing Normal University, Beijing 100875, China, with the Beijing Key Laboratory for Remote Sensing of Environment, and Digital Cities, Faculty of Geographical Science, Beijing Normal University, Beijing 100875, China (e-mail: 201921051188@mail.bnu.edu.cn; tanghong@bnu.edu.cn; 202021051203@mail.bnu.edu.cn; 202021051202@mai.bnu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2021.3079459

The method performed well on the land cover mapping task, although multinetwork integration was required to obtain the best results. Inspired by encoder–decoder networks such as U-Net [18], the traditional convolutional layer with an inception module and proposed an enhanced hourglass-shaped network. The newly added modules simultaneously improved accuracy and reduced the network volume. In [19], traditional U-Net was adapted with the ResNet [20] backbone to extract information from handcrafted features. In [21], the combination of U-Net and ResNet was further explored and an end-to-end deeper CNN network named DeepResUnet was proposed. Inspired by NASNet-Mobile [22], a new U-Net variant called U-NASNetMobile was proposed to extract building footprints in [6]. Since 2015, the performance of DL-based building extraction methods has continued to improve due to advances in computer vision technology and SS networks. However, these methods still suffer from the limitations of the SS framework: the spatial resolution of the generated semantic maps strictly corresponds to that of the input raster data. For instance, only semantic maps with 10-m spatial resolution were generated when given raster data with 10-m spatial resolution. This limits the application of traditional SS methods in disaster assessment.

Image super-resolution (SR) is an alternative technique to address the abovementioned dilemmas. Traditional interpolation-based SR methods such as BiCubic interpolation [23] suffer from the poor reconstruction performance and blurry results. Great achievements in DL-based SR networks have been made recently. The pioneering work on DL-based SR methods can be traced to 2014, when SRCNN made the first attempt to employ a CNN for SR [24]. Additional DL-based SR methods with a better performance were subsequently proposed. In [25], the efficient subpixel convolutional neural network (ESPCN) was proposed to avoid adverse artificial effects in the deconvolutional layer. The introduction of designed backbones, e.g., VGG-net [26] and ResNet [20], made SR networks deeper and more powerful. For example, VGG-net was applied on VDSR [27], and ResNet was applied on SRDenseNet [28] and the residual dense network (RDN) [29]. SR techniques have also been applied for remote sensing image processing: SRCNN was exploited to construct HR satellite-derived sea surface temperature data and obtained a considerable improvement of the peak signal-to-noise ratio compared with traditional methods [30]. A multifork network named LGCNet was designed in [31] to improve the resolution of remote sensing data; in this method, multilevel features were learned to reconstruct HR images. In other work [32], the visual attention mechanism was integrated within residual learning to make the network focus on high-frequency details in land-cover components. The method exhibited competitive performance in two remote sensing datasets with three scaling factors. Further recent explorations of SR include the application of the visual attention mechanism to single remote sensing image SR in [33].

Based on research on SR and SS, a super-resolution semantic segmentation (SRSS) framework could be built to generate higher resolution pixel-level semantic labels from lower resolution images. To the best of authors' knowledge, there are the following two kinds of approaches to obtain HR SS maps from low-resolution (LR) images: 1) stagewise approaches, i.e., carrying out SS and SR independently [34]; (2) end-to-end approaches, i.e., integrating SS and SR in a network. Two variants of stagewise SRSS approaches are depicted in Fig. 1. As shown in Fig. 1(a), input LR images are first processed by various SR approaches in stage 1, and then, SS models are fed the generated HR images. The preprocessing SR methods range from traditional interpolation-based methods to DL-based SR networks, e.g., ESPCN and RDN. Similar to the method shown in Fig. 1(a), Fig. 1(b) depicts another stagewise form. The segmentation maps are first generated from LR images with pretrained SS models in stage 1. SR approaches, e.g., nearest interpolation, are exploited to obtain HR segmentation maps from LR segmentation maps directly. Few works, except for nearest interpolation, have explored ways to enhance the resolution of segmentation maps because rich context features are discarded in the process of generation of LR maps. Therefore, the quality of the final results is dominated by the LR semantic maps.

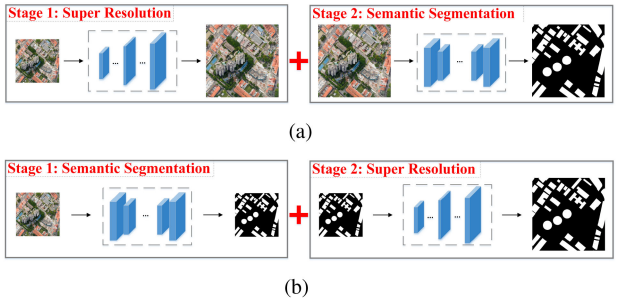



Fig. 1. Sketches of stagewise SRSS approaches.

In the stagewise methods, SR is exploited to reconstruct images with enhanced spatial resolution in terms of visual effect and SS is exploited to produce class labels for each of the raster pixels. However, the images processed by SR are not certainly beneficial for SS [34] due to the lack of cooperativity between the two separated tasks. In other words, combining well-behaved image SR models and SS models would produce superior semantic maps with higher resolution from lower resolution data. Therefore, an end-to-end SRSS approaches are in need. In such methods, the SR process is integrated in SS networks as a special module to enhance the resolution of features from intermediate layers. The two main elements are named the front component and rear component in terms of the element position in an end-to-end network. There are different ways to combine both of the SR and SS modules. In the end-to-end network, the SR module could be the front component as depicted in Fig. 2(a) and the SS module could also be the front component as depicted in Fig. 2(b). In the recently proposed end-to-end SRSS network [35] named dual super-resolution learning (DSRL), the SR module is employed as the rear component. In DBPN-SegNet [36], the SR module is employed as the front component. However, HR images are needed to train DSRL and DBPN-SegNet, which greatly hinders practical applications. To address the dilemma, a unified framework in which both SR and SS modules are integrated is proposed in this article.

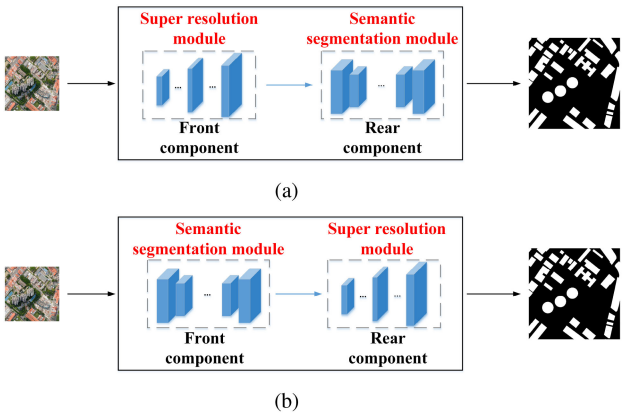



Fig. 2. Sketches of end-to-end SRSS approaches.

Our main contributions in this article are threefold.

1) We propose an effective end-to-end SRSS network, i.e., ESPC_NASUnet, to extract HR building maps from LR remote sensing images, in whose training phase HR images are not in need.
2) We compare different methods, including stage-wise and end-to-end approaches, to obtain building maps whose spatial resolution is higher than that of the input. The rules to construct a well performed end-to-end SRSS framework are systematically compared and analyzed using multiple datasets.
3) We design three new metrics, i.e., object recall (OR), object precision (OP), and object $F1$-score (OF1), to evaluate the SRSS results, which can reflect the quality of building prediction result from the object level.

The remainder of this article is organized as follows. In Section II, the proposed ESPC_NASUnet is described in detail. The datasets, implementation details, metrics, and comparison networks are described in Section III. Both quantitative evaluation and qualitative analysis are presented in Section IV. In Section V, the validity and limitations are discussed. Finally, some conclusions are drawn in Section VI.

## II. Method

In the proposed ESPC_NASUnet, an SR module is employed as the front component and the rear component is an SS module. Specifically, the efficient subpixel convolution (ESPC) and U-Net with NASNet-Mobile [37] backbone (NASUnet) are used as SR and SS modules, respectively.

### A. SR Module: ESPC Module

Inspired by ESPCN [25], we design an ESPC feature SR module. It consists of three convolution layers. The first two are normal 2-D convolution layers with 64 and 32 convolutional kernels of size $3 \times 3$. These two layers extract valuable information to facilitate resolution enhancement and SS in the following layers. In addition, the two convolutional layers can enlarge the receptive field of the next subpixel convolution. The third layer is a special subpixel convolution layer that consists of the following three steps: convolution, periodic shuffle, and concatenation. As shown in Fig. 3, $w_{\text{in}}$ and $h_{\text{in}}$ are the width and height of the input feature maps, respectively. $b_{\text{in}}$ and $b_{\text{out}}$ denote channel numbers of input features and output features, respectively. $s$ means SR scale. The input features are first convoluted by $b_{\text{out}}s^2$ filters. Then, the convoluted features of size $w_{\text{in}} \times h_{\text{in}} \times b_{\text{out}}s^2$ are rearranged with the periodic shuffle operation. The re-aligned features are $b_{\text{out}}$ feature maps with size $\text{sw}_{\text{in}} \times \text{sh}_{\text{in}}$. Finally, the feature maps are concatenated in the channel dimension.

In the process of subpixel convolution, spatial information is first stored in the channel dimension and then restored to the spatial dimension. This process can enhance spatial resolution without artificial interruption, such as zero internal padding in the deconvolution layer. The subpixel convolution can be described as follows:

$$x^{\text{SR}} = \mathcal{PS}\left(x_{\text{in}}\right) = \mathcal{PS}\left(W_{\text{SR}} * x^{\text{LR}} + b_{\text{SR}}\right) \tag{1}$$

where $\mathcal{PS}$ denotes the periodic shuffle operand, $x_{\text{in}}$ are features generated from former convolution layers, and $W_{\text{SR}}$ and $b_{\text{SR}}$ are the weight and bias of the convolution layer, respectively. As shown in Fig. 3, $x_{\text{in}}$ is a $w_{\text{in}} \times h_{\text{in}} \times (b_{\text{out}}s^2)$ tensor, and the final output tensor is $\text{sw}_{\text{in}} \times \text{sh}_{\text{in}} \times b_{\text{out}}$, i.e., the $b_{\text{out}}s^2$ tensor is grouped into $b_{\text{out}}$ parts, and each part consists of $s^2$ feature maps. Then, features with the same spatial location in feature maps from one part are placed adjacently in the output feature map. The process can be described mathematically as follows:

$$\begin{aligned}&\mathcal{PS}\left(x_{\text{in}}\right)_{x,y,z}\\&= \left(x_{\text{in}}\right)_{\left[\frac{i}{s}\right],\left[\frac{j}{s}\right],k*s^2+\bmod(i,s)*s+\bmod(j,s)}\end{aligned} \tag{2}$$

where $(i, j, k)$ indicates the coordinates of the processed feature in the output feature maps $x^{\text{SR}}$.

### B. SS Module: NASUnet

The SS module acting as the rear component is named NASUnet. As shown in Fig. 4, layers next to the ESPC module represent a stem convolution, which consists of a convolution layer with stride 2 and a batch normalization layer. The other blocks are structures obtained via neural architecture searching [22], e.g., reduction cells and normal cells. These cells are first searched for image classification on ImageNet. In [6], they proved effective in an SS network. As shown in Fig. 4, a normal cell consists identity layers, depthwise-separable convolution layers with different kernel sizes and average pooling layers with different window sizes. Similarly, there are identity layers, depthwise-separable convolution layers, average and max pooling layers with different kernal sizes in the reduction cell. The kernel sizes vary among values of $3 \times 3$, $5 \times 5$, and $7 \times 7$. In addition, the normal cell will repeat 4 times in the network once it is used. For these two kinds of cells, the inputs are the output tensors of the previous layer and the layer before previous layer, i.e., skip connection is employed among the layers. Note that the spatial resolution will be reduced twofold through the reduction cell and will not change through the normal cell. The combination of these structures has strong feature extraction ability and functions as the encoder in this

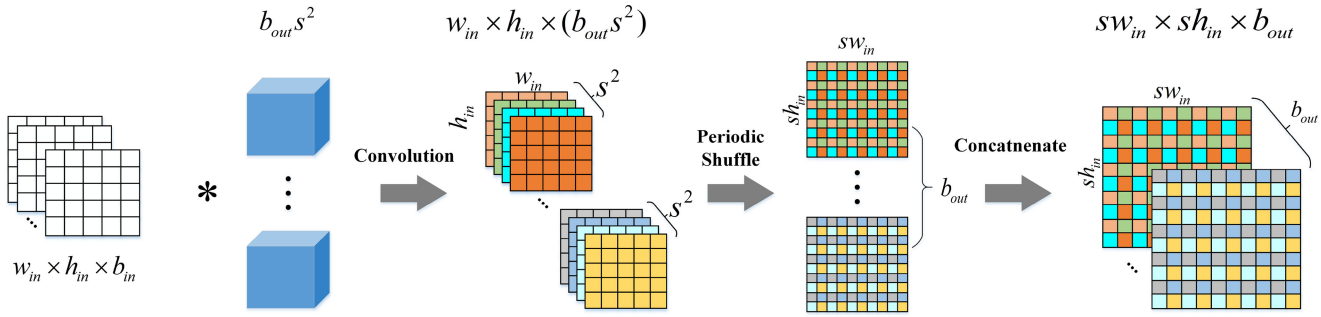



Fig. 3. Workflow of subpixel convolution. The sign $*$ denotes the convolution operation. As an intuitive representation, $w_{\text{in}}$, $h_{\text{in}}$, and $s$ are, respectively, set as 5, 5, and 2 in this figure.

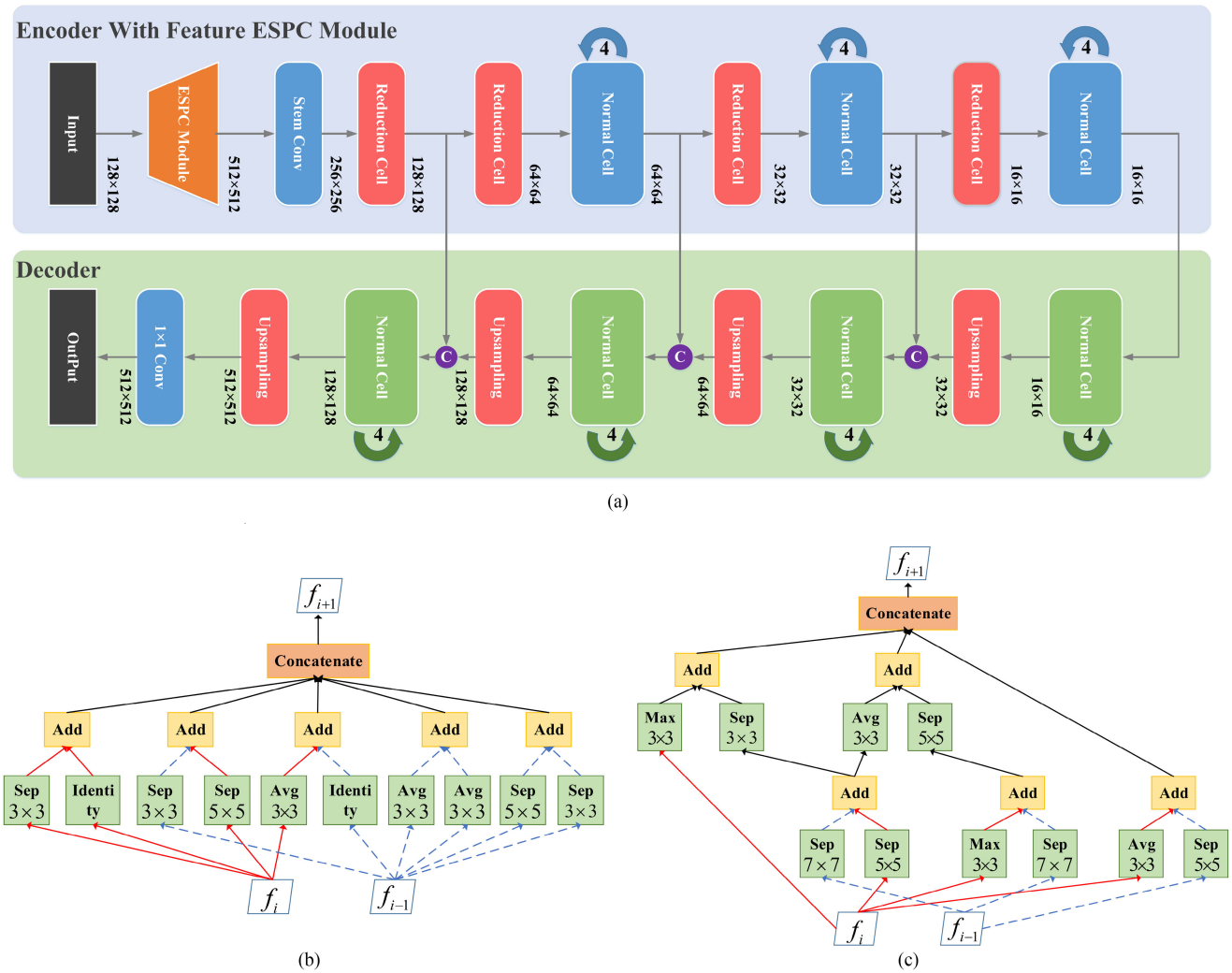



Fig. 4. Specification of ESPC_NASUnet, normal cell, and reduction cell. The purple circles with the letter C in ESPC_NASUnet indicate concatenation. Sep, Avg, Max, and Identity in two cells indicate depthwise-separable convolution layer, average pooling layer, max pooling layer, identity layer, respectively. The number below the text in the green box is the kernel size of each layer. $f_{i-1}$, $f_i$ and $f_{i+1}$ are the output of the layer before the previous layer, the previous layer and this layer, respectively.

"U" shaped SS module; the SR feature acts as the input for the encoder. The decoder of ESPC_NASUnet is the same as that of U-NASNetMobile [6] and can generate superior building semantic maps with the help of the NAS mobile backbone. There are four upsampling blocks to recover resolution from encoded feature maps to the final semantic maps whose size equals that of the SR feature maps. In detail, the upsampling block consists of one bilinear interpolation layer, four normal cells and one SE block [38]. As shown in Fig. 4, the input of ESPC_NASUnet is LR images with size 128 × 128. The ESPC block with 4×

SR factor converts the LR image into $512 \times 512$ again. Finally, softmax is used to complete the pixel-level classification of the decoded feature maps.

The specific architecture of ESPC_NASUnet is shown in Fig. 4, where the ESPC module is placed next to the input layer. In a deep SS neural network, the feature maps output from layers that are close to the network input are comprehensible [39]. In other words, these feature maps contain more detailed information about buildings [40], e.g., morphological and geometric information, if given a remote sensing image. In contrast, the feature maps from layers close to the output layer are much more abstract and usually contain information that benefits the segmentation task. In other words, low-level feature maps are gradually refined to generate superior semantic maps at the cost of information loss. The lost details may be crucial for the SR task. Accordingly, the ESPC module is employed as the front component so that the details can be effectively exploited to reconstruct HR features without extra loss. In addition, the ESPC module is a shallow feature SR module, whose number of convolution layers is only three. It can be considered as a special feature extractor. Differently, the task of such extractor is to enhance the spatial resolution of feature maps rather than condense it. NASUnet module is an encoder–decoder architecture. Feature maps in such module are first compressed in the spatial dimension via encoder then recovered via decoder. The shallow SR module will not greatly increase the depth of feature extraction part, which could result in the imbalance between encoder and decoder and unstable gradient propagation. Accordingly, ESPC module and NASUnet module are chosen as the front and the rear components to construct the SRSS network, i.e., ESPC_NASUnet.

## III. Experiments

In this section, we first specify some experimental settings, such as the building datasets, training strategy, and other detailed information. Then, pixel-level metrics and new object-level metrics are described for quantitative evaluation. For comparison, the architectures of three additional end-to-end SRSS networks are ultimately presented.

### A. Datasets

For SRSS of buildings, due to the lack of directly available datasets, we create two datasets by downsampling the images in the DREAM-B dataset [6] and Massachusetts buildings (MBs) dataset [41].

*DREAM-B*: The original dataset contains 626 image tiles from 100 different cities around the world. The size of each original image tile is $4096 \times 4096$. Among them, 250 tiles are randomly selected for training, 63 tiles for validation, and 313 tiles for testing. The image tiles and corresponding ground truths share the same spatial resolution of 30 cm. To meet the demands of the SRSS task, all image tiles are resampled as $1024 \times 1024$ with the BiCubic method to simulate the images whose spatial resolution is 1.2 m.

MB: The original dataset consists of 151 aerial images with the corresponding ground truth of size $1500 \times 1500$. In the dataset, 137 images, 4 images, and 10 images are employed for training, validation, and testing, respectively. Like DREAM-B, the images in this dataset are resampled as $375 \times 375$ with the BiCubic method. Note that the original spatial resolution is 1 m and that of the transformed samples is approximately 4 m. Therefore, the building patches are relatively smaller and more ambiguous, and thus, the task is more challenging.

All image tiles are composed of 3 bands: red(R), green(G), and blue(B). Only two classes, buildings and nonbuildings, are interpreted in the semantic ground truth.

### B. Implementation Details

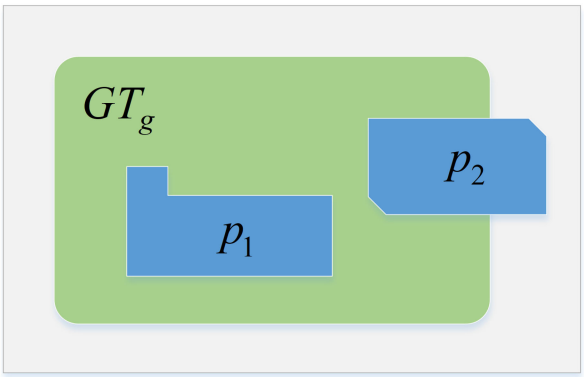
In our experiments, one NVIDIA Tesla V100 GPU is utilized to train ESPC_NASUnet and other networks. The value of $b_{\text{out}}$ in the ESPC module is set as 3. The size of each input batch is $128 \times 128$, and the batch size is set as 16. Categorical cross entropy is used as the loss function. We apply the optimizer Adaptive moment estimation (Adam) [42] to optimize the model parameters. The cosine learning rate annealing schedule [43] includes a maximum learning rate of $3 \times 10^{-4}$ and a minimum learning rate of $1 \times 10^{-6}$. The annealing period is set as 100 and the number of total epochs is 210. The first 10 epochs are used for model warm-up. In other words, the learning rate schedule in this article consists of one warm-up process and two annealing cycles. In addition, data augmentations are employed to reduce the impacts of overfitting, including random flipping vertically and horizontally, random rotation, and random brightness jittering.

### C. Evaluation Metrics

For detailed comparison, pixel-level and object-level evaluation metrics are exploited. In pixel-level evaluation, overall accuracy (OA) is the most widely used evaluation metric. Nevertheless, the number of buildings is small compared with that of other background objects. OA is not able to reflect the actual performance of different methods if there is a huge gap in the percentages of the building class and the background class. Other metrics are also provided for inference, including recall, precision, $F1$ Score, and Kappa coefficient. In addition, the intersection over union (IoU) [44] is employed as a monitoring index in the training process. More specifically, the IoU can be calculated by

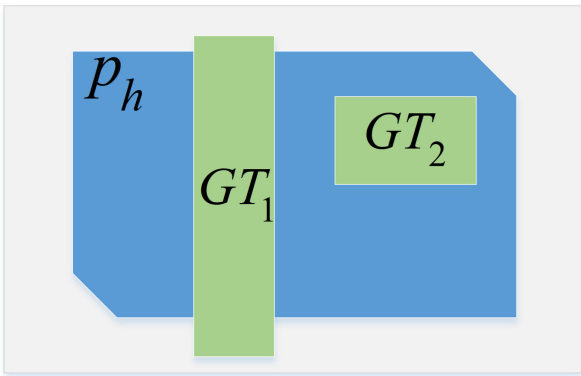
$$\text{IoU} = \frac{\text{Prediction } \cap \text{ Ground Truth}}{\text{Prediction } \cup \text{ Ground Truth}}. \tag{3}$$

SS is a pixel-level classification task and corresponding metrics are all based on pixel-level statistics. SRSS for building extraction is a special task in which the degree of separation of building objects matters. For instance, distinguishing two buildings located 2 m apart is easy in a remote sensing image with 0.3 m spatial resolution but may be a challenge in an image with 3 m spatial resolution. The correspondence between predictions and building objects is always many-to-many in SS, in contrast to the one-to-one or zero-to-one relationship in instance segmentation. Therefore, object-level metrics in instance segmentation, e.g., mean average precision, cannot be applied




$$OR_g = \frac{\sum_{i=1}^{n_g} S_{GT_g \cap p_i}}{n_g S_{GT_g}} = \frac{S_{GT_g \cap p_1} + S_{GT_g \cap p_2}}{2 S_{GT_g}}$$

(a)




$$OP_h = \frac{\sum_{j=1}^{n_h} S_{p_h \cap GT_j}}{n_h S_{p_h}} = \frac{S_{p_h \cap GT_1} + S_{p_h \cap GT_2}}{2 S_{p_h}}$$

(b)

Fig. 5. Examples of OR and OP. (a) Two predicted patches intersect with $\mathrm{GT}_g$, i.e., $n_g$ is 2. (b) Two building objects and predicted patches $p_h$ intersect each other, i.e., $n_h$ is 2.

in SRSS. To address the abovementioned dilemma, three new object-level metrics are proposed based on the pixel-level recall, precision, and $F1$-score. In such an evaluation pattern, irregular patches whose prediction labels are all building classes are considered independent building objects. For a single building object in the ground truth, if an area of predicted building patches approximates the true area, it can be considered appropriately segmented. A large amount of building patches inside the true object can also harm segmentation performance, giving rise to the fragmentation phenomenon. To summarize, the segmentation performance can be described quantitatively by the metric OR, which for the $g$th building object $\mathrm{GT}_g$ is calculated as follows:

$$\mathrm{OR}_g = \frac{1}{n_g} \sum_{i=1}^{n_g} \frac{S_{\mathrm{GT}_g \cap p_i}}{S_{\mathrm{GT}_g}} \tag{4}$$

where $n_g$ denotes the quantity of predicted patches that intersect the $g$th building object, and $p_i$ is the $i_{\mathrm{th}}$ predicted intersectant patch among predicted patches. $S$ in the formula means the value of the patch area. Similarly, the metric OP for the $h$th predicted patch can be calculated based on the predicted patches as follows:

$$\mathrm{OP}_h = \frac{1}{n_h} \sum_{j=1}^{n_h} \frac{S_{p_h \cap \mathrm{GT}_j}}{S_{p_h}} \tag{5}$$

where $n_h$ is the quantity of real building objects that intersect the $h$th predicted patch. Correspondingly, $\mathrm{GT}_i$ denotes the $j$th real building object among the abovementioned objects. Note that a large $n_h$ indicates that many buildings are predicted inside one patch, which is called prediction adhesion. Obviously, this should be avoided in SRSS. To provide a better description, the calculation example is shown in Fig. 5.

Similar to pixel metrics, OR and OP are able to reflect performance unilaterally. Thus, a comprehensive index akin to the $F1$-score is needed. We apply microstatistic patterns to calculate the OF1 as follows:

$$\mathrm{OF1} = \frac{2 * \sum_{g=1}^{m} \mathrm{OR}_g * \sum_{h=1}^{n} \mathrm{OP}_h}{n \sum_{g=1}^{m} \mathrm{OR}_g + m \sum_{h=1}^{n} \mathrm{OP}_h} \tag{6}$$

where $m$ and $n$ denote the total quantities of building objects and predicted patches, respectively.

### D. Structures of Other SRSS Networks

ESPC_NASUnet is the first SRSS network that is not trained with HR images. In other words, there is no method that can be compared with ESPC_NASUnet under the same conditions. Therefore, we constructed three other SRSS networks that are similar to ESPC_NASUnet for comparison. There are two SR module integration modes. The first mode is like ESPC_NASUnet, in which the SR module is employed as the front component. In the second mode, the SR module is placed between the SS module and output layer to act as the rear component, is designed. For the sake of comparison, the network NASUnet_ESPC, in which the ESPC module is the rear component and NASUnet is the front component. In such a network, the spatial size of features first falls to $4 \times 4$ via the encoder and then rises to $512 \times 512$ with the decoder and ESPC module.

The development of SR approaches from VDSR [27] to RDN [29] has proved the theory that deeper or wider networks obtain the better performance in terms of image SR. It is unclear whether a more powerful image SR model can help produces better SRSS maps when it is modified as a feature SR block. Therefore, the residual dense super-resolution (RDSR) block is designed for comparison, which is transformed from a complex and outstanding image SR model RDN.

The architecture of RDSR block is depicted in Fig. 6. There are many convolution layers and residual dense blocks (RDBs) in front of subpixel convolution layers. Dense connections and residual learning are exploited inside of RDB. Features from

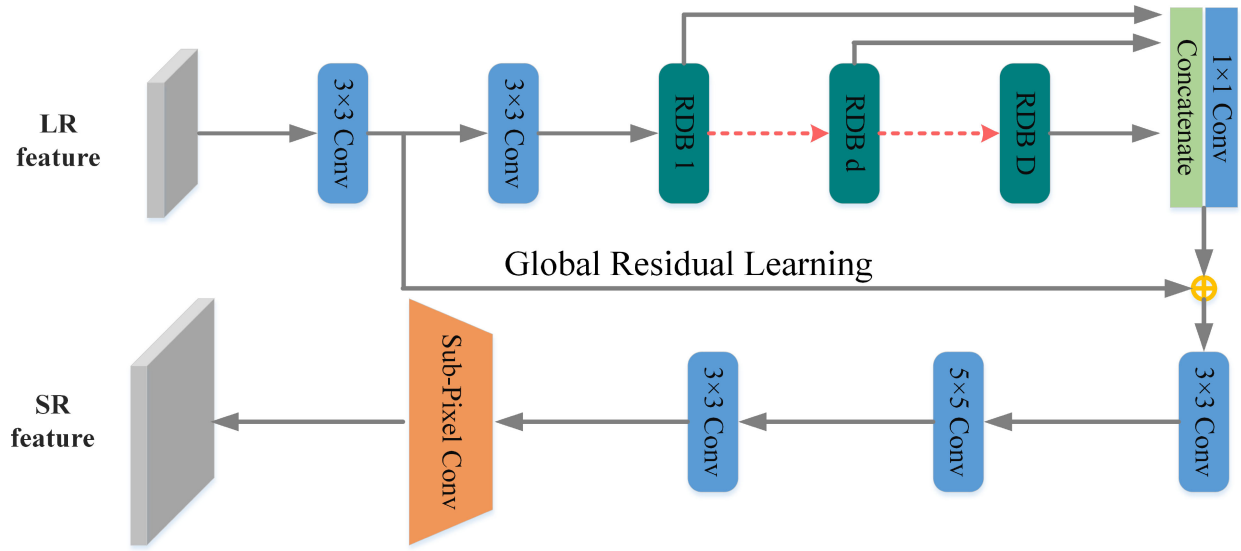



Fig. 6. Architecture of RDSR block. RDB is the block proposed in RDN. $D$ RDBs are exploited in the RDSR block. The yellow circle with the add symbol denotes the element addition operand.

TABLE I
PIXEL-LEVEL EVALUATION RESULTS

| Datasets | Methods | OA | Recall | Precision | F1-score | IoU | Kappa |
|---|---|---|---|---|---|---|---|
| DREAM-B | RDSR_NASUnet | 0.8443 | 0.6515 | 0.7403 | 0.6931 | 0.5303 | 0.8640 |
| | NASUnet_RDSR | 0.8244 | 0.5090 | 0.7614 | 0.6101 | 0.4390 | 0.8353 |
| | NASUnet_ESPC | 0.8530 | 0.6871 | 0.7477 | 0.7161 | 0.5578 | 0.8733 |
| | ESPC_NASUnet | **0.8837** | **0.7850** | **0.7842** | **0.7846** | **0.6456** | **0.9023** |
| MB | RDSR_NASUnet | 0.8712 | 0.4525 | 0.7577 | 0.5666 | 0.3953 | 0.4965 |
| | NASUnet_RDSR | 0.8573 | 0.3293 | 0.7739 | 0.4620 | 0.3004 | 0.3948 |
| | NASUnet_ESPC | 0.8494 | 0.2758 | 0.7634 | 0.4053 | 0.2541 | 0.3401 |
| | ESPC_NASUnet | **0.8900** | **0.4864** | **0.8627** | **0.6221** | **0.4515** | **0.5635** |

shallow layers are added to features from deep layers via skip connection. Similarly, RDSR modules are employed as the front component or rear component in the newly designed end-to-end networks, termed RDSR_NASUnet and NASUnet_RDSR, respectively.

## IV. EXPERIMENTAL RESULTS

In this section, the proposed ESPC_NASUnet is compared with the three other end-to-end SRSS methods quantitatively and qualitatively. The settings are the same as that of ESPC_NASUnet described in Section III-B when training the other end-to-end networks.

### A. Quantitative Evaluation

Table I presents the quantitative evaluations of the two datasets using six pixel-level metrics. In the DREAM-B dataset, ESPC_NASUnet obtains the highest values of all pixel-level metrics among the four end-to-end SRSS networks. In terms of recall and precision, the proposed ESPC_NASUNet obtains values of 0.7850 and 0.7842, respectively, which are high for the SS task. In addition, the difference value of 0.0008 indicates stability of the network. The precision values obtained by the other three networks are all higher than the recall values by more than 0.05. Therefore, the $F1$-score of ESPC_NASUnet exceeds those of the other methods by more than 0.0685. Similarly, the difference values of IoU and Kappa coefficient reflect the effectiveness on this dataset: the values are greater than 0.0878 and 0.0290, respectively. Compared to the spatial resolution of 1.2 m in DREAM-B, the spatial resolution of 4 m in the MB dataset presents a great challenge for the SRSS task. The values of all metrics except for OA and precision in the MB dataset considerably decline. In such situations, ESPC_NASUnet still outperforms the other three networks. The value of recall is 0.4864, which is much lower than that of precision with respect to DREAM-B. This phenomenon suggests that ESPC_NASUnet tends to predict pixels that are blurry in lower resolution images as background. Nevertheless, the values of the proposed network are higher than those of the other three models. For the three comprehensive metrics, i.e., $F1$-score, IoU, and Kappa coefficient, ESPC_NASUnet yields great superiority despite the low values arising from the challenging dataset.

Table II lists the quantitative evaluations on the simulation datasets using three object-level metrics. On the DREAM-B dataset, the results of the comparison are similar to those for pixel-level metrics in terms of OF1. The values of the OR metric and OP metric differ greatly. ESPC_NASUnet obtains the highest OR value. This indicates that ESPC_NASUnet tends

TABLE II
OBJECT-LEVEL EVALUATION RESULTS

| Datasets | Methods | OR | OP | OF1 |
|---|---|---|---|---|
| **DREAM-B** | RDSR_NASUnet | 0.3612 | 0.6034 | 0.4519 |
| | NASUnet_RDSR | 0.1894 | **0.6768** | 0.2960 |
| | NASUnet_ESPC | 0.4593 | 0.5980 | 0.5195 |
| | ESPC_NASUnet | **0.5862** | 0.6211 | **0.6031** |
| **MB** | RDSR_NASUnet | 0.3035 | 0.6793 | 0.4195 |
| | NASUnet_RDSR | 0.2633 | 0.6432 | 0.3737 |
| | NASUnet_ESPC | 0.1463 | 0.6094 | 0.2359 |
| | ESPC_NASUnet | **0.4588** | **0.7912** | **0.5808** |

to predict an integrated patch to approximate the true building patch, i.e., it predicts with low fragmentation. In terms of OP, NASUnet_RDSR, which produces the lowest OR value, obtains the highest OP value because the patches predicted by NASUnet_RDSR are always accompanied by a fragmented distribution, i.e., it is common for NASUnet_RDSR to predict a single building in the ground truth as multiple objects. ESPC_NASUnet yields an OP value of 0.6211, which is a relatively good result. OF1 can reflect the object-level result comprehensively and ESPC_NASUnet obtains the highest OF1 value of 0.6031. In other words, ESPC_NASUnet obtains a superior balance between a low degree of fragmentation and adhesion. On the MB dataset, ESPC_NASUnet obtains the highest values among all methods in terms of all three object-level metrics: 0.4588, 0.7912, and 0.5808, respectively. In terms of OF1, the value of ESPC_NASUnet is 0.1613 higher than that of RDSR_NASUnet, whose value of OF1 is the second greatest. This is significantly superior with respect to the OF1 metric.

The abovementioned comparisons demonstrate the better performance of ESPC_NASUnet in generating HR semantic maps from LR images without the help of HR images.

### B. Qualitative Analysis

In Fig. 7(a)–(e), the courtyards inside the building in the yellow circle are clearly recognized by ESPC_NASUnet, whereas the other three methods misclassified parts of these courtyards as buildings. In Fig. 7(f)–(j), the shadow in the circle is easily misclassified as a building by RDSR_NASUnet and NASUnet_ESPC. Despite the prediction of the shadow by NASUnet_RDSR, ESPC_NASUnet can produce semantic maps with better approximation of building edges. In the MB dataset, most buildings in downsampled images with a spatial resolution of 4 mare small objects. There are the following two challenges when generating semantic maps for such small objects: 1) the adjacent objects are classified as one object, i.e., prediction with great adhesion; 2) many small objects are omitted. As shown in Fig. 7(k)–(o), the former situation is common in semantic maps generated by the other three methods, whereas ESPC_NASUnet can generate prediction results with low adhesion. In Fig. 7(p)–(t), many buildings are misclassified as background by the other three models. ESPC_NASUnet can generate semantic maps with fewer omissions. These visualization results are merely for reference, and the quantitative evaluations in Section IV-A are more reliable.

### C. Position and Complexity of the Feature SR Module

The position of the feature SR module is a significant factor that affects the performance of the final result. ESPC_NASUnet and NASUnet_ESPC share the same feature SR module with different positions in the network. Similarly, both RDSR_NASUnet and NASUnet_RDSR include the same RDSR module with different positions in the network. The architectures of the four networks are described in detail in Section III-D. Loss and $F1$-score curves are depicted in Fig. 8. In the training process, ESPC_NASUnet converges with a lower loss and higher $F1$-score, whereas RDSR_NASUnet and NASUnet_RDSR converge at similar points. In terms of validation curves, networks in which the feature SR module is employed as the front component, i.e., ESPC_NASUnet and RDSR_NASUnet, obtain better $F1$-scores than networks with rear-placed SR modules. This result suggests that more beneficial segmentation information is preserved when SR modules are employed on the low-level features and that placing the feature SR block in front of the SS module is a practical design idea.

The complexity of the SR block is another factor to be considered when constructing an SRSS network. The ESPC module and RDSR module are two SR modules with different complexities: the latter is more complex, and the prototype RDN has stronger image SR capability. SRSS networks with the ESPC module reach the convergence point with lower loss and higher $F1$-scores with respect to the DREAM-B dataset, which diverges from the rule in image SR tasks [27]–[29]. In addition, the validation curves of the networks in both datasets with RDSR as the front component are unstable because the number of layers in networks is too large to maintain normal gradient propagation. Note that the convergence point of NASUnet_ESPC is higher than that of NASUnet_RDSR in terms of training loss on the MB dataset. For datasets with low spatial resolution, a more complex SR rear component with stronger representational capacity is needed. This phenomenon suggests that the complexity of the feature SR block cannot dominate SRSS performance. In other words, an SR block transformed from a high-performing image SR model may not necessarily provide superior performance. The final result may be affected by the position of the SR module and the spatial resolution of the dataset.

## V. DISCUSSION

In this section, some conventional methods are evaluated for the sake of discussion. The settings of these methods are listed in Table III. The HR or LR in the brackets next to the SS model refer to the dataset used for model training. For instance, U-NASNetMobile(HR) is trained with HR images and the corresponding HR ground truth. In the DREAM-B dataset, the resolution is 0.3 m for HR data and 1.2 m for LR data. The resolutions of HR data and LR data in the MB dataset are 1 m and 4 m, respectively. All stagewise methods and end-to-end methods are evaluated on the LR test images. To distinguish between the end-to-end approaches and stagewise approaches, the latter methods are named by joining the names of the applied SR method and SS method with a plus sign.

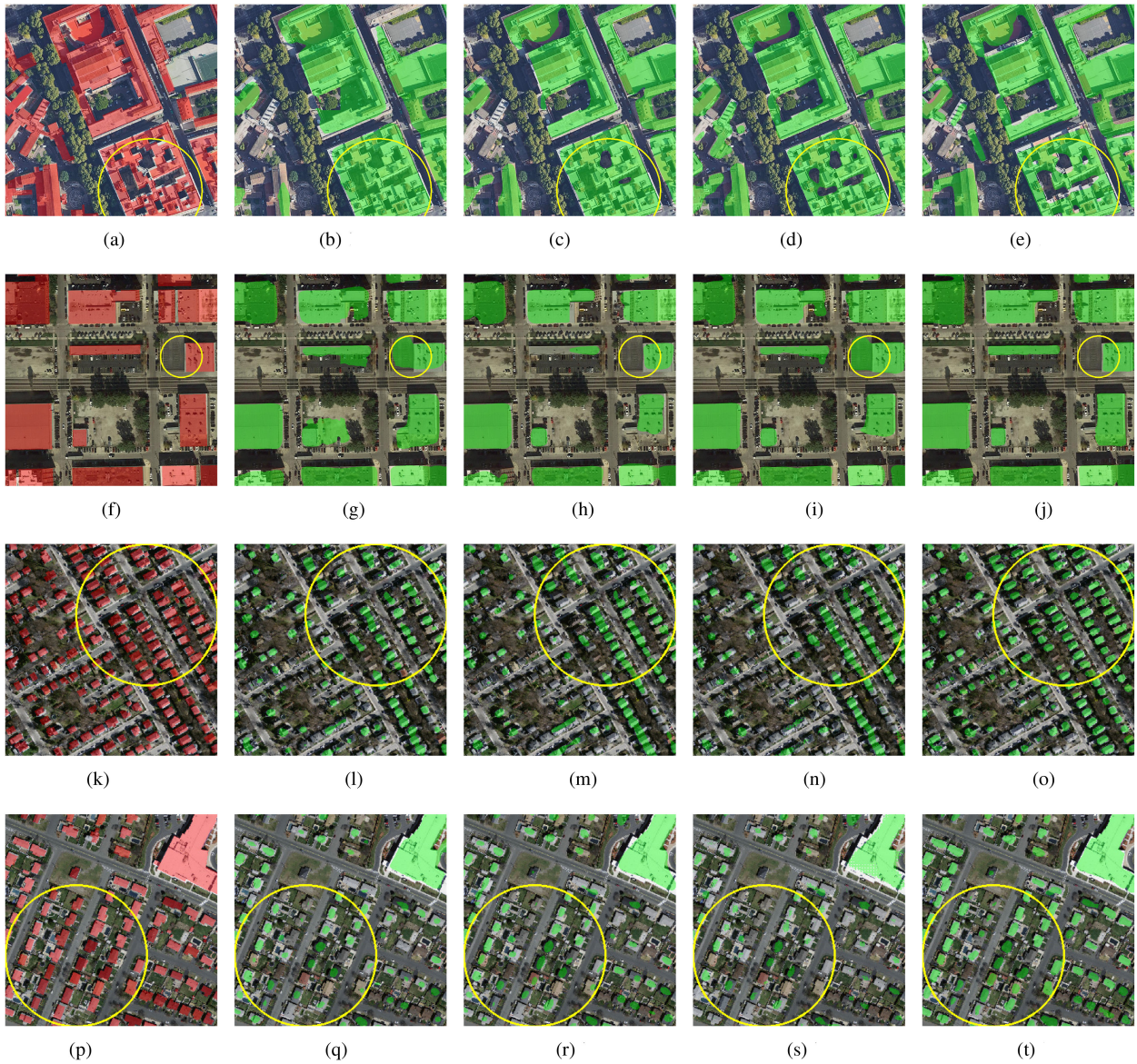



Fig. 7. SS results of different end-to-end SRSS methods. The subfigures in the first column are the building ground truth. The subfigures in the other columns are generated by RDSR_NASUnet, NASUnet_ESPC, and ESPC_NASUnet, respectively. Some notable details are marked with yellow circles. The background of each figure is the original HR image.

TABLE III
GENERAL SETTINGS OF THE METHODS IN THE SECTION V

| Methods | Training dataset | Test data | Pre-Processing | Post-Processing | End-to-end |
|---|---|---|---|---|---|
| U-NASNetMobile(HR) | HR images+HR GT | HR images | - | - | - |
| U-NASNetMobile(HR)+Nearest | HR images+HR GT | LR images | - | Nearest Resampling | ✗ |
| U-NASNetMobile(LR)+Nearest | LR images+LR GT | LR images | - | Nearest Resampling | ✗ |
| BiCubic+U-NASNetMobile(HR) | HR images+HR GT | LR images | BiCubic Resampling | - | ✗ |
| RDN+U-NASNetMobile(HR) | HR images+HR GT+LR images | LR images | RDN | - | ✗ |
| ESPCN+U-NASNetMobile(HR) | HR images+HR GT+LR images | LR images | ESPCN | - | ✗ |
| DBPN-SegNet | HR images+HR GT+LR images | LR images | - | - | ✗ |
| MDBPN-SegNet | LR images+HR GT | LR images | - | - | ✗ |
| ESPC_NASUnet | LR images+HR GT | LR images | - | - | ✓ |

GT in the second column refers to ground truth.

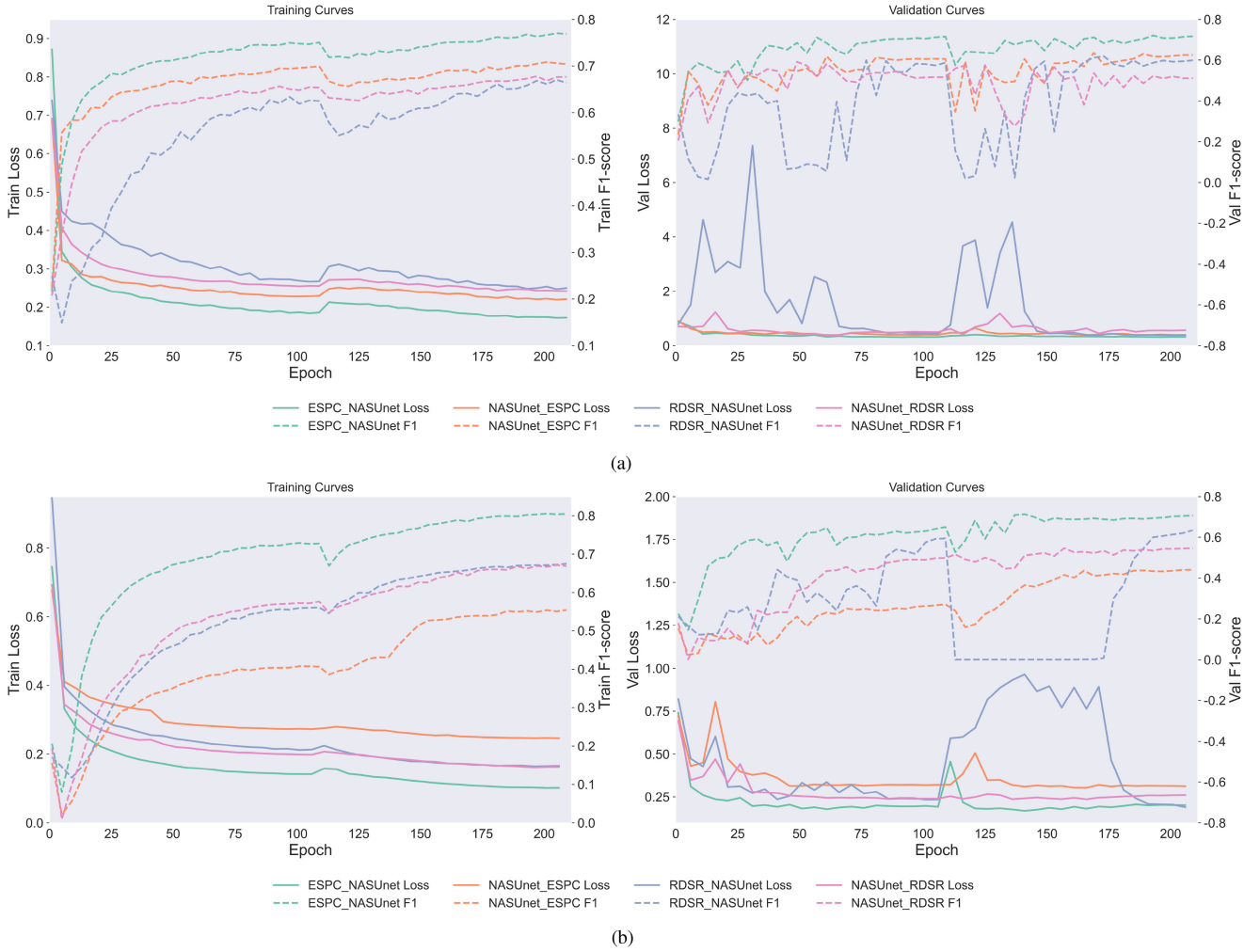



Fig. 8. Loss and $F1$-score curves of end-to-end SRSS networks in training process. Training and validation curves are plotted in the same subfigure.(a) Training process in DREAM-B dataset. (b) Training process in MB dataset.

### *A. Comparison Between ESPC_NASUnet and Stagewise Methods*

The proposed ESPC_NASUnet is an efficient end-to-end model for generating HR semantic maps from LR images directly. SR and SS are carried out independently in stage-wise methods. To compare the two kinds of models, four stagewise methods are tested on the two datasets. There are the following three methods of preprocessing: interpolation-based BiCubic resampling, shallow SR network ESPCN, and complex network RDN. Only one method with postprocessing nearest to resampling is included due to the lack of related research. The same data augmentation as ESPC_NASUnet is exploited in the training phase of U-NASNetMobile(LR), U-NASNetMobile(HR), RDN, and ESPCN. Following [29] and [25], the loss functions of RDN and ESPCN are set as the mean absolute error and mean square error, respectively. For the methods RDN+U-NASNetMobile(HR) and ESPC+U-NASNetMobile(HR), the training process is divided into two parts. First, HR images and corresponding LR images are used to train the supervised SR models RDN and ESPCN. Then, the supervised SS models are trained with HR images and HR labels. Therefore, HR images are reused in the training process.

As shown in Table IV, stagewise methods can obtain higher precision values, except for U-NASNetMobile(LR)+Nearest in MB. However, the values of other metrics vary greatly between the two datasets. In DREAM-B dataset, the degeneration of spatial resolution from 0.3 to 1.2 m makes little difference for building recognition. Although the quality of the final results mainly depends on that of the generated LR semantic maps, U-NASNetMobile(LR)+Nearest obtains a relatively good result. Among the methods with preprocessing, DL-based SR methods are better than interpolation-based methods. Note that RDN+U-NASNetMobile(HR) obtains better results than the proposed ESPC_NASUnet, which is reasonable for the recycled usage of HR images in the training process of RDN and U-NASNetMobile(HR). By contrast, none of the HR images are exploited to train ESPC_NASUnet. As depicted in Fig. 9(b)–(d), DL-based SR networks can recover details from LR images better than BiCubic resampling. In terms of visual effect, the image recovered by RDN in Fig. 9(d) exhibits little difference from the original image in Fig. 9(a), which leads to superior semantic maps in Fig. 9(h). By contrast, many buildings are misclassified as background in Fig. 9(f) due to the poor performance of BiCubic resampling in Fig. 9(b).

TABLE IV
RESULTS OF DIFFERENT STAGEWISE METHODS IN THE TWO DATASETS

| Datasets | Methods | Pixel-level metrics | | | | | | Object-level metrics | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | OA | Recall | Precision | F1-score | IoU | Kappa | OR | OP | OF1 |
| DREAM-B | U-NASNetMobile(LR)+Nearest | 0.8730 (-0.011) | 0.7041 (-0.081) | 0.8012 **(+0.017)** | 0.7495 (-0.035) | 0.5994 (-0.046) | 0.8891 (-0.013) | 0.4641 (-0.122) | 0.6203 (-0.001) | 0.5310 (-0.072) |
| | BiCubic+U-NASNetMobile(HR) | 0.8541 (-0.030) | 0.5794 (-0.206) | 0.8283 **(+0.044)** | 0.6819 (-0.103) | 0.5173 (-0.128) | 0.8646 (-0.038) | 0.4392 (-0.147) | 0.6862 **(+0.065)** | 0.5356 (-0.068) |
| | RDN+U-NASNetMobile(HR) | 0.8927 **(+0.009)** | 0.7644 (-0.021) | 0.8251 **(+0.041)** | 0.7936 **(+0.009)** | 0.6578 **(+0.012)** | 0.9077 **(+0.005)** | 0.5529 (-0.033) | 0.6723 **(+0.051)** | 0.6068 **(+0.004)** |
| | ESPCN+U-NASNetMobile(HR) | 0.8861 **(+0.002)** | 0.7174 (-0.068) | 0.8371 **(+0.053)** | 0.7726 (-0.012) | 0.6295 (-0.016) | 0.8998 (-0.002) | 0.5185 (-0.068) | 0.6805 **(+0.059)** | 0.5886 (-0.014) |
| MB | U-NASNetMobile(LR)+Nearest | 0.8308 (-0.059) | 0.1493 (-0.337) | 0.7183 (-0.144) | 0.2472 (-0.375) | 0.1410 (-0.310) | 0.1957 (-0.368) | 0.0557 (-0.403) | 0.5425 (-0.249) | 0.1010 (-0.480) |
| | BiCubic+U-NASNetMobile(HR) | 0.8443 (-0.046) | 0.1793 (-0.307) | 0.9184 **(+0.056)** | 0.3000 (-0.322) | 0.1765 (-0.275) | 0.2547 (-0.309) | 0.1808 (-0.278) | 0.8935 **(+0.102)** | 0.3007 (-0.280) |
| | RDN+U-NASNetMobile(HR) | 0.8460 (-0.044) | 0.1929 (-0.293) | 0.9029 **(+0.040)** | 0.3178 (-0.304) | 0.1889 (-0.263) | 0.2700 (-0.293) | 0.1872 (-0.272) | 0.8625 **(+0.071)** | 0.3076 (-0.273) |
| | ESPCN+U-NASNetMobile(HR) | 0.8406 (-0.049) | 0.1586 (-0.328) | 0.9133 **(+0.051)** | 0.2703 (-0.352) | 0.1563 (-0.295) | 0.2278 (-0.336) | 0.1916 (-0.267) | 0.8941 **(+0.103)** | 0.3156 (-0.265) |

Pixel-level and object-level metrics are all employed for evaluation. The numbers in brackets are difference values between the methods. A positive number indicates a higher value than that of ESPC_NASUnet for that such metric. All positive numbers are in bold.

TABLE V
QUANTITATIVE RESULTS OF U-NASNETMOBILE(HR) AND U-NASNETMOBILE(HR)+NEAREST

| Datasets | Methods | Pixel-level metrics | | | | | | Object-level metrics | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | OA | Recall | Precision | F1-score | IoU | Kappa | OR | OP | OF1 |
| DREAM-B | U-NASNetMobile(HR)+Nearest | 0.7962 (-0.088) | 0.4320 (-0.353) | 0.6977 (-0.087) | 0.5336 (-0.251) | 0.3639 (-0.282) | 0.8055 (-0.097) | 0.1802 (-0.406) | 0.5881 (-0.033) | 0.2759 (-0.327) |
| | U-NASNetMobile(HR) | 0.9029 **(+0.019)** | 0.8073 **(+0.022)** | 0.8286 **(+0.044)** | 0.8178 **(+0.033)** | 0.6918 **(+0.046)** | 0.9178 **(+0.015)** | 0.6300 **(+0.044)** | 0.6640 **(+0.043)** | 0.6466 **(+0.043)** |
| MB | U-NASNetMobile(HR)+Nearest | 0.8278 (-0.062) | 0.0826 (-0.404) | 0.9094 **(+0.047)** | 0.1514 (-0.471) | 0.0819 (-0.370) | 0.1243 (-0.439) | 0.0049 (-0.454) | 0.8710 **(+0.080)** | 0.0097 (-0.571) |
| | U-NASNetMobile(HR) | 0.9240 **(+0.034)** | 0.6850 **(+0.199)** | 0.8800 **(+0.017)** | 0.7703 **(+0.148)** | 0.6265 **(+0.175)** | 0.7257 **(+0.162)** | 0.6272 **(+0.168)** | 0.8155 **(+0.024)** | 0.7091 **(+0.128)** |

The numbers in brackets are difference values between the methods and ESPC_NASUnet. A positive number indicates a higher value of the metric than that of ESPC_NASUnet. All positive numbers are in bold.

For the MB dataset, the performance of stagewise methods degenerates considerably. The visual effect changes substantially when the spatial resolution declines from 1 to 4 m. This challenge results in significant omission in the prediction results of U-NASNetMobile(LR)+Nearest. In addition, the tremendous degeneration of resolution makes image SR a tough task. The listed BiCubic resampling and ESPCN encounter difficulty in rebuilding images from those with 4-m resolution. In spite of the superior performance for DREAM-B, RDN is unable to produce clear SR images from the MB dataset. As shown in Fig. 9(k)–(m), the SR results from the three methods are all blurry and result in inferior semantic maps in Fig. 9(o)–(q). By contrast, ESPC_NASUnet obtains the superior performance for such datasets.

### *B. Lower Bound and Upper Bound of the SRSS Task in Terms of Performance*

For SRSS, the lower bound and upper bound are produced in two different situations. For the upper bound, data with HR are tested on U-NASNetMobile(HR). Data with LR are tested on U-NASNetMobile(HR) without any preprocessing to produce the lower bound. For a fair comparison, nearest resampling is exploited to enhance the resolution of the generated semantic maps. The name of the lower bound is U-NASNetMobile(HR)+Nearest.

Quantitative results are listed in Table V. In DREAM-B, the values of all metrics for U-NASNetMobile(HR) are higher than those of ESPC_NASUnet. However, the difference values are smaller than 0.05. By contrast, the difference values between U-NASNetMobile(HR)+Nearest and ESPC_NASUnet exceed 0.3 for some metrics. In Fig. V-B(a) –(d), prediction results on a region from Nepal are depicted. In terms of delineation of small buildings, the results of ESPC_NASUnet exhibit greater degrees of adhesion than those of U-NASNetMobile(HR), whereas all buildings are predicted as background by U-NASNetMobile(HR)+Nearest. In MB, the upper bound method produces much higher values of the metrics due to the great difference between the input data of the two methods. Correspondingly, ESPC_NASUnet obtains much better values of some metrics than U-NASNetMobile(HR)+Nearest. The disparities among the three methods are enlarged in the MB dataset due to the degradation of spatial resolution. The prediction results for a subregion in Massachusetts are shown in Fig. V-B(e)–(h): more small buildings are misclassified as background by ESPC_NASUnet than by

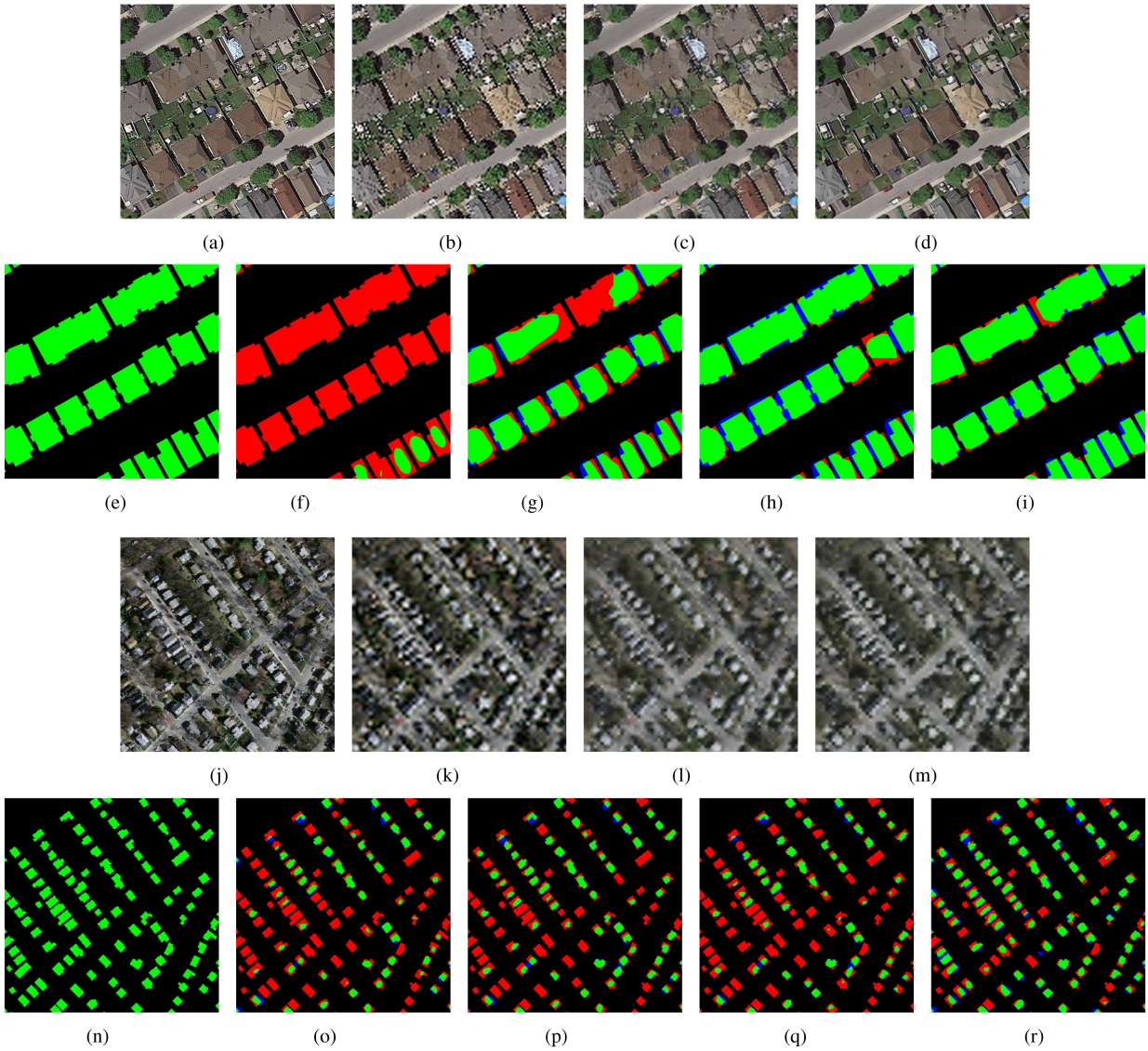



Fig. 9. Visual examples of stagewise methods with preprocessing and ESPC_NASUnet. (a)–(i) From DREAM-B and (j)–(r) MB. (a) and (j) Original HR images. (e) and (n) Corresponding ground truths. (b)–(d) and (k)–(m) Preprocessed images in two datasets and the three preprocessing methods are BiCubic resampling, ESPCN and RDN, respectively. (f)–(h) and (o)–(q) Semantic maps generated by BiCubic+U-NASNetMobile(HR), ESPCN+U-NASNetMobile(HR), and RDN+U-NASNetMobile(HR) in two datasets. (i) and (r) are generated by ESPC_NASUnet. In semantic maps, pixels in green and blue are predicted correctly and incorrectly, respectively. Pixels in red are omitted in the prediction.

U-NASNetMobile(HR). The comparison between the upper bound U-NASNetMobile(HR), ESPC_NASUnet, and the lower bound NASNetMobile(HR)+Nearest shows that the performance of ESPC_NASUnet is better than the lower bound. However, the gap between the upper bound and ESPC_NASUnet is hard to ignore. Therefore, further improvement of the approximation effect of SRSS models is needed.

TABLE VI
QUANTITATIVE RESULTS OF ESPC_NASUNET, DBPN-SEGNET, AND MDBPN-SEGNET ON THE MA DATASET

| Methods | F1-score | IoU | Kappa | OF1 |
|---|---|---|---|---|
| ESPC_NASUnet | **0.6221** | **0.4515** | **0.5635** | **0.5808** |
| DBPN-SegNet | 0.4576 | 0.2967 | 0.3993 | 0.4633 |
| MDBPN-SegNet | 0.5864 | 0.4148 | 0.5257 | 0.5542 |

### C. Comparison With Another End-to-End SRSS Network

If an SRSS network is trained without HR images, the network can be called an HRI-free network. Otherwise, it is called an HRI-need network. However, no HRI-free SRSS network was proposed before ESPC_NASUnet. To compare the ESPC_NASUnet with HRI-need network, we choose DBPN-SegNet [36] as a base model and remove the SR branch of it to construct a new modified DBPN-SegNet (MDBPN-SegNet). As shown in Table III, the training data of the modified network are the same as that of ESPC_NASUnet. We trained MDBPN-SegNet with the same settings as ESPC_NASUnet on the MB dataset. DBPN-SegNet is also evaluated for comparison.

As shown in Table VI, the performance of ESPC_NASUnet is the best among the three networks in terms of the four metrics. In

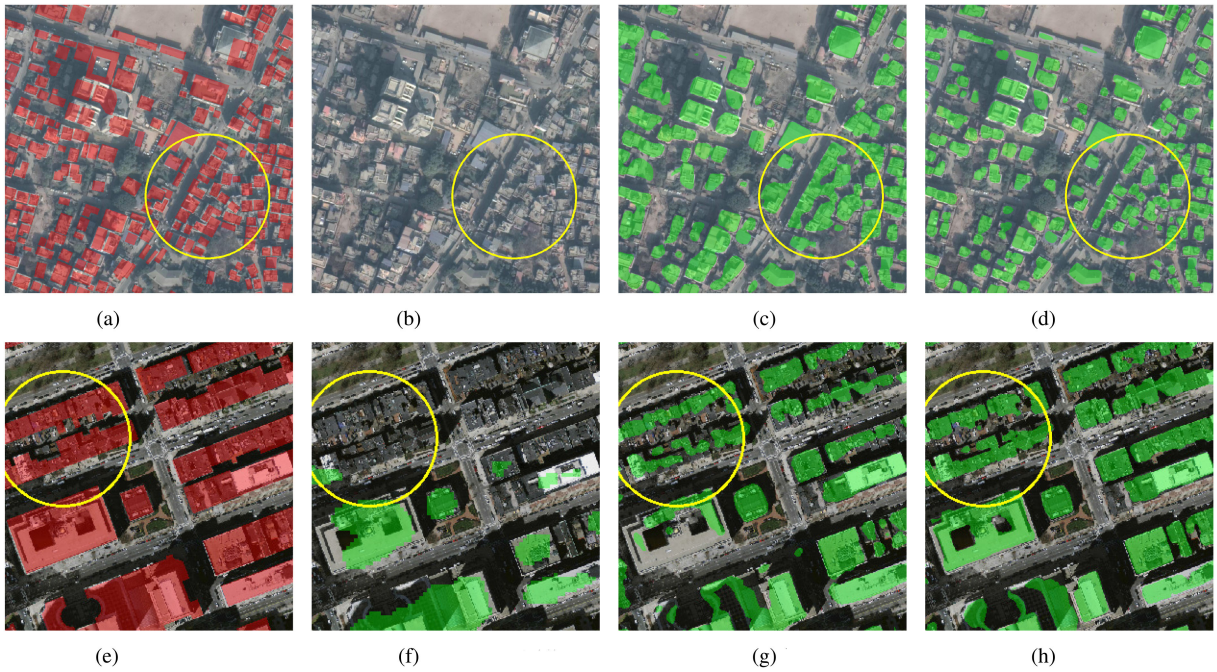



Fig. 10. SS examples of U-NASNetMobile(HR)+Nearest, U-NASNetMobile(HR), and ESPC_NASUnet. The subfigures in the first column are the buildings' ground truth. The results generated by U-NASNetMobile(HR)+Nearest, U-NASNetMobile(HR), and ESPC_NASUnet are listed in the second, third, and fourth columns, respectively.

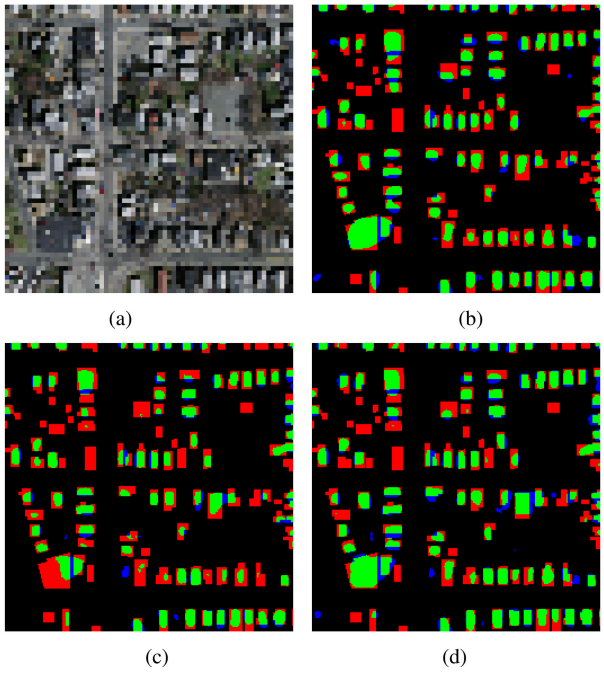



Fig. 11. Visual examples of ESPC_NASUnet, DBPN-SegNet, and MDBPN-SegNet. (a) LR image. (b)–(d) Semantic maps generated by ESPC_NASUnet, DBPN-SegNet, and MDBPN-SegNet, respectively. In semantic maps, pixels in green and blue are predicted correctly and wrongly. Pixels in red are omitted in the prediction.

addition, the performance of DBPN-SegNet is worse than that of MDBPN-SegNet, i.e., the SR task of DBPN-SegNet harms the SS task in this dataset. As shown in Section V-A, it is a difficult task to super-resolve the resampling images four times in the MA dataset. In other words, the multitasks cannot promote each other when some tasks are difficult to complete. Therefore, HRI-need networks are not necessarily effective for datasets with LR. By contrast, the proposed ESPC_NASUnet can be applied to more scenarios.

TABLE VII
EVALUATION OF ESPC_NASUNET WITH DIFFERENT SR SCALE FACTORS

| Scale Factor | F1-Score | IoU | Kappa | OF1 |
|---|---|---|---|---|
| 4 | 0.6221 | 0.4515 | 0.5635 | 0.5808 |
| 6 | 0.4640 | 0.3020 | 0.3999 | 0.4537 |
| 8 | 0.3025 | 0.1782 | 0.2451 | 0.1716 |
| 10 | 0.1866 | 0.1029 | 0.1473 | 0.0824 |

### *D. ESPC_NASUnet With Other SR Factors*

The network is proposed for data with LR. To demonstrate the efficiency of ESPC_NASUnet, more large SR scales are employed on the MB dataset, including 6, 8, and 10. In the training phase, the input images are resized with a large factor, and the input labels retain the original spatial resolution. All the training settings are the same as in the experiment with 4× ESPC_NASUnet. The metrics of $F1$-score, IoU, and OF1 are used for evaluation. The result is shown in Table VII. When the scale factor changes from 4 to 6, the performance drops slightly. However, the metric values drop sharply when the scale factor is 8 or 10. The main reason is that extensive information loss occurs in the process of shrinking the images. The larger the factor is, the more information is lost. Remote sensing data with LR, e.g., Sentinel-2A images, could alleviate this dilemma.

TABLE VIII
QUANTITATIVE RESULTS OF ESPC_NASUNET ON SENTINEL-2 IMAGES

| Method | F1-Score | IoU | Kappa | OF1 |
|---|---|---|---|---|
| ESPC_NASUnet | 0.4726 | 0.3094 | 0.3779 | 0.3449 |

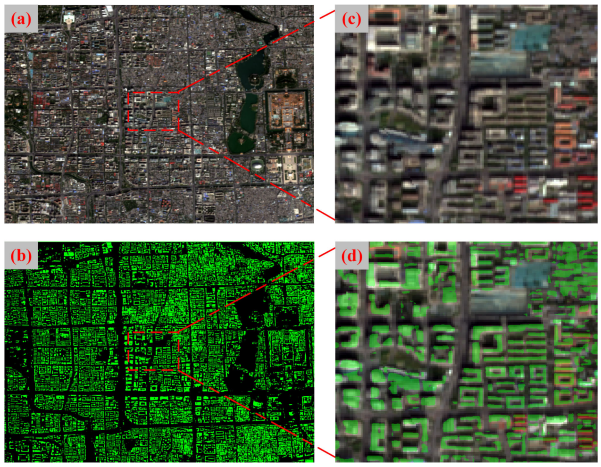



Fig. 12. Semantic maps of predicted buildings from a Sentinel-2 image. (b) Predicted buildings with 2.5 m spatial resolution by the ESPC_NASUnet from the Sentinel-2 image in (a). (c) Enlarged subimage from (a) marked with red box; (d) Image in (c) overlaid with buildings predicted by the ESPC_NASUnet. The sentinel images displayed in (a), (c), and (d) Resampled four times to correspond the spatial resolution of predicted semantic maps.

### E. ESPC_NASUnet on Sentinel-2 Images

In order to further analyze the practicability of ESPC_NASUnet, we employed the network on Sentinel-2 images. The images used in our experiments are the level-2A of Sentinel-2 with red, green, blue, and near infrared bands, whose spatial resolution is 10 m. The ground truth is extracted based on open street map (OSM). Beijing in China is chosen as an experimental area. In consideration of that some buildings in Beijing are not labeled in the build-up coverage of OSM, we select 1075 patches with correct semantic labels as the training samples and 145 patches for validation. The pixel size of all the samples are $64 \times 64$ and that of corresponding ground truth is $256 \times 256$.

All the training settings are the same with that in Section III. The quantitative results of validation samples are listed in Table VIII. The 2.5 m predicted building maps of beijing are depicted in Fig. 12. As we can find that some small buildings are delineated well in the Fig. 12(d). Therefore, the ESPC_NASUnet can also work well on Sentinel-2 images, which could extract precise 2.5 m building map from 10 m Sentinel-2 image.

## VI. CONCLUSION

Inspired by the image SR technologies and SS, an end-to-end SRSS network called ESPC_NASUnet is proposed in this article that makes full use of the relationships between higher resolution ground truth and lower spatial resolution images. The main conclusions of this article are as follows:

1) employing the SR module as the front component improves the performance of end-to-end SRSS;
2) In contrast to SR approaches in stagewise SRSS methods, SR modules with low complexity can help maintain the stability and superiority of end-to-end methods;
3) the HRI-free end-to-end methods are less impacted by the degeneration of spatial resolution;
4) the proposed ESPC_NASUnet is capable of capturing details in LR images for fine recognition of individual buildings.

This article is the first attempt at HRI-free SRSS for building delineation from remote sensing images and can be used as a reference for higher resolution mapping from lower resolution satellite images. However, there is still a gap in the performance between the proposed method and the SS model trained by the HR dataset. Although the results in Section V-E have proved the practicability of ESPC_NASUnet on Sentinel-2 satellite images of Beijing City, the dataset is small and a larger real satellite datasets are in need. In the future, we would furthermore explore the practical application of the proposed method to produce semantic maps of buildings from large-scale satellite images with middle level resolution, for example, rapid mapping from Sentinel-2 images for unexpected national disasters in China.

## REFERENCES

[1] X. Jin and Curt H Davis, "Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual, and spectral information," *EURASIP J. Adv. Signal Process.*, vol. 2005, no. 14, 2005, Art. no. 745309.
[2] Y. Xu, L. Wu, Z. Xie, and Z. Chen, "Building extraction in very high resolution remote sensing imagery using deep learning and guided filters," *Remote Sens.*, vol. 10, no. 1, 2018, Art. no. 144.
[3] T. Zhang and H. Tang, "A comprehensive evaluation of approaches for built-up area extraction from Landsat OLI images using massive samples," *Remote Sens.*, vol. 11, no. 1, 2019, Art. no. 2, doi: 10.3390/rs11010002.
[4] X. Liu *et al.*, "High-resolution multi-temporal mapping of global urban land using landsat images based on the Google Earth Engine platform," *Remote Sens. Environ.*, vol. 209, pp. 227–239, 2018.
[5] M. Pesaresi, Georgios K Ouzounis, and L.Gueguen, "A new compact representation of morphological profiles: Report on first massive VHR image processing at the JRC," *Proc. SPIE*, vol. 8390, 2012, Art. no. 839025.
[6] N. Yang and H. Tang, "Geoboost: An incremental deep learning approach toward global mapping of buildings from VHR remote sensing images," *Remote Sens.*, vol. 12, no. 11, 2020, Art. no. 1794.
[7] L. Dong and J. Shan, "A comprehensive review of earthquake-induced building damage detection with remote sensing techniques," *ISPRS J. Photogrammetry Remote Sens.*, vol. 84, pp. 85–99, 2013.
[8] S. F. Ding, B. J. Qi, and H. Y. Tan, "An overview on theory and algorithm of support vector machines," *J. Univ. Electron. Sci. Technol. China*, vol. 40, no. 1, pp. 2–10, 2011.
[9] X. Huang and L. Zhang, "A multidirectional and multiscale morphological index for automatic building extraction from multispectral geoeye-1 imagery," *Photogrammetric Eng. Remote Sens.*, vol. 77, no. 7, pp. 721–732, 2011.
[10] X. Huang and L. Zhang, "Morphological building/shadow index for building extraction from high-resolution imagery over urban areas," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 1, pp. 161–172, Feb. 2012.
[11] R. AvudaiammalS. and V. Rajangam, "Extraction of buildings in urban area for surface area assessment from satellite imagery based on morphological building index using SVM classifier," *J. Indian Soc. Remote Sens.*, vol. 48, no. 9, pp. 1325–1344, 2020.
[12] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
[13] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Analysis Mach. Intell.*, 39, no. 12, pp. 2481–2495, 2017.

[14] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.
[15] L.-C. Chen, Y.Zhu, G. Papandreou, F. Schroff, and H.Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.
[16] S. Paisitkriangkrai, J. Sherrah, P. Janney, and A. Van-Den Hengel, "Effective semantic pixel labelling with convolutional networks and conditional random fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2015, pp. 36–43.
[17] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, "Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2016, pp. 1–9.
[18] Y. Liu, D. M. Nguyen, N. Deligiannis, W. Ding, and A. Munteanu, "Hourglass-Shapenetwork based semantic segmentation for high resolution aerial imagery," *Remote Sens.*, vol. 9, no. 6, 2017, Art. no. 522.
[19] Y. Xu, L. Wu, Z. Xie, and Z. Chen, "Building extraction in very high resolution remote sensing imagery using deep learning and guided filters," *Remote Sens.*, vol. 10, no. 1, 2018, Art. no. 144.
[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[21] Y. Yi, Z. Zhang, W. Zhang, C. Zhang, W. Li, and T. Zhao, "Semantic segmentation of urban buildings from VHR remote sensing imagery using a deep convolutional neural network," *Remote Sens.*, vol. 11, no. 15, 2019, Art. no. 1774.
[22] B. Zoph, V. Vasudevan, J. Shlens, and Q. V Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8697–8710.
[23] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981.
[24] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.
[25] W. Shi *et al.*, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1874–1883.
[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[27] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1646–1654.
[28] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4799–4807.
[29] J. Xu, Y. Chae, B. Stenger, and A. Datta, "Dense Bynet: Residual dense network for image super resolution," in *Proc. 25th IEEE Int. Conf. Image Process.*, 2018, pp. 71–75.
[30] A. Ducournau and R. Fablet, "Deep learning for ocean remote sensing: An application of convolutional neural networks for super-resolution on satellite-derived SST data," in *Proc. IEEE 9th IAPR Workshop Pattern Recognit. Remote Sens.*, 2016, pp. 1–6.
[31] S. Lei, Z. Shi, and Z. Zou, "Super-resolution for remote sensing images via local-global combined network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 8, pp. 1243–1247, Aug. 2017.
[32] J. M. Haut, R. Fernandez-Beltran, M. E Paoletti, J. Plaza, and A. Plaza, "Remote sensing image superresolution using deep residual channel attention," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9277–9289, Nov. 2019.
[33] D. Zhang, J. Shao, X. Li, and H. T. Shen, "Remote sensing image super-resolution via mixed high-order attention network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5183–5196, doi: 10.1109/TGRS.2020.3009918.
[34] Z. Guo *et al.*, "Super-resolution integrated building semantic segmentation for multi-source remote sensing imagery, *IEEE Access*, vol. 7, pp. 99381–99397, 2019.
[35] L. Wang, D. Li, Y. Zhu, L. Tian, and Y. Shan, "Dual super-resolution learning for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3774–3783.
[36] M. B. Pereira and J. A. d. Santos, "An end-to-end framework for low-resolution remote sensing semantic segmentation," in *Proc. IEEE Latin Amer. GRSS ISPRS Remote Sens. Conf.*, 2020, pp. 6–11.
[37] B. Zoph, V. Vasudevan, J. Shlens, and Q. V Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8697–8710.
[38] Abhijit Guha Roy, N. Navab, and C. Wachinger, "Concurrent spatial and channel 'squeeze & excitation' in fully convolutional networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervention*, 2018, pp. 421–429.
[39] X. Li, H. Zhao, L. Han, Y. Tong, S. Tan, and K. Yang, "Gated fully fusion for semantic segmentation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, pp. 11418–11425, 2020.
[40] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun, "Exfuse: Enhancing feature fusion for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 269–284.
[41] V. Mnih, *Machine Learning for Aerial Image Labeling*, Canada, 2013.
[42] D. P. Kingma and J.Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[43] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.
[44] M. Everingham, L. V. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.

**Penglei Xu** received the B.S. degree in engineering of surveying and mapping from Hohai University, Nanjing, China, in 2019. He is currently working the M.S. degree in photogrammetry and remote sensing with the Faculty of Geographical Science, Beijing Normal University, Beijing, China.

His research interests include remote sensing image processing and pattern recognition.

**Hong Tang** received the B.S. and M.S. degrees from the China University of Mining and Technology, Xuzhou, China, in 1998 and 2001, respectively, and the Ph.D. degree in pattern recognition and intelligence systems from the Shanghai Jiao Tong University, Shanghai, China, in 2006.

From 2006 to 2008, he was a Postdoc with the IMEDIA Project, INRIA Paris, Paris, France. He is currently a Professor with the Beijing Normal University, Beijing, China. His research interests include remote sensing image processing, pattern recognition, and natural disaster reduction.

**Jiayi Ge** received the B.S. degree in geographic information science from Dalian Maritime University, Dalian, China, in 2020. He is currently working the M.S. degree in cartography and geographic information engineering with the Faculty of Geographical Science, Beijing Normal University, Beijing, China.

His research interests include remote sensing image processing and pattern recognition.

**Lin Feng** received the B.S. degree in engineering of surveying and mapping from Wuhan University, Wuhan, China, in 2018. She is currently working toward the M.S. degree in cartography and geographic information engineering with Beijing Normal University, Beijing, China.

Her research interests are remote sensing big data and machine learning.