Small Object Segmentation Using Dilated Convolutions With Increasing-Decreasing Dilation

Ryuhei Hamaguchi, Aito Fujita, and Keisuke Nemoto

***Abstract*—This article presents a novel convolutional neural network (CNN) architecture for segmenting significantly small and crowded objects in remote sensing imagery. Although such small objects are characteristic in the remote sensing domain, the previous works mostly follow the state-of-the-art CNN models designed for ground-based images and have yet to fully explore the method for segmenting the small objects. To this end, we propose a network with no downsampling layers by utilizing dilated convolutions. We find that naive use of dilated convolutions with "increasing" dilation rates fails to capture local relationships among neighboring features, resulting in grid-like noise in the prediction. To alleviate this problem, we propose a novel scheme of "increasing-decreasing" dilation rates. Specifically, we propose a network module with decreasing dilation rates and attach it to the dilated backbone to reconnect the neighboring pixels of the backbone features. In the experiments, we evaluated the proposed model on six remote sensing datasets, where the model showed remarkably high performance, especially for small objects.**

***Index Terms*—Convolutional neural networks (CNNs), image analysis, semantic segmentation.**

I. Introduction

RECOGNITION of small objects is a fundamental problem for remote sensing (RS) image analysis. Many important targets in RS, such as buildings, cars, or roads, occupy only a tiny region in an image. For example, in a 50 cm resolution satellite imagery, buildings and cars have only 6–20 pixels on a side, and the width of roads is mostly less than 20 pixels. In this article, we define small objects as individual objects that occupy fewer than 400 pixels in the RS image (i.e., XS and S size in Fig. 2). Fig. 1 compares typical segmentation tasks in natural scenes and RS imagery. We see two difficulties in the RS image segmentation task: 1) Small and crowded objects and 2) cluttered background. Although convolutional neural networks (CNNs) have shown impressive performance on RS segmentation tasks, they are mainly based on architectures designed for natural scenes and thus still have difficulties with small objects.

As an example, Fig. 2 shows the performance of segmentation models for each building size in DeepGlobe Building Detection dataset [1]. Although many small buildings exist in the dataset, such as small housings or sheds, the state-of-the-art segmentation models perform poorly on the small buildings.

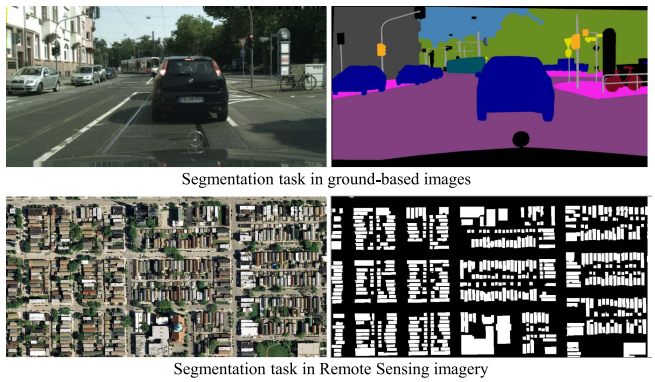

Fig. 1. Comparison of typical segmentation tasks in RS imagery (top) and ground-based images (bottom). The objects are significantly smaller and denser in RS imagery. Note that the image size is the same for both cases.

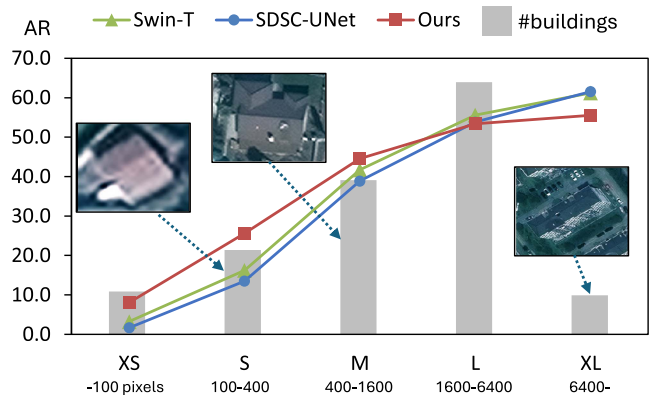

Fig. 2. Distribution of building size in DeepGlobe dataset and the performance of segmentation models for each size. While previous models perform well for large buildings, they perform poorly for small and medium size buildings. Our proposed method successfully enhances the accuracy for small buildings at the little expense of large buildings. ("Ours" shows the result of ResNet-D-LFE.)

This motivates us to explore a dedicated architecture to enhance the accuracy for these small objects, as achieved in the red plots in the figure.

The biggest obstacle for small object segmentation in RS imagery is downsampling. Downsampling layers such as max-pooling or strided convolutions are widely used for natural scenes to enlarge the receptive field (RF) of a CNN model. However, downsampling layers discard the detailed spatial information, leading to degraded performance for small objects. Recent RS image segmentation models try to recover the lost spatial information by reusing low-level features through skip

Received 3 April 2024; revised 4 July 2024 and 1 September 2024; accepted 23 September 2024. Date of publication 10 October 2024; date of current version 24 October 2024. *(Corresponding author: Ryuhei Hamaguchi.)*

The authors are with the Satellite Business Division, Pasco Corporation, Tokyo 153-0064, Japan (e-mail: riyhuc2734@pasco.co.jp).

Digital Object Identifier 10.1109/JSTARS.2024.3477606

connections [2], [3], [4], [5], [6], [7], [8]. However, the low-level features tend to be noisy because of the cluttered background, and the improvement for small objects is limited. A naive approach of removing downsampling layers is ineffective because a CNN model with a small RF cannot distinguish small objects from cluttered backgrounds [9]. Since small objects do not contain rich spatial information inside themselves, contextual cues around them are necessary for recognition.

One promising solution for the problem is the dilated convolutions [10], [11]. The dilated convolutions aggressively expand the RF by applying a dilated kernel. The kernel size is enlarged in the dilated kernel by setting an interval between the kernel weights. Unlike downsampling, the dilated convolutions can aggregate a wide range of information without losing resolution.

Nevertheless, we highlight that a naive application of dilated convolutions does not necessarily improve performance. Commonly, dilated convolutions are used by "increasing" dilation rates through layers. However, we find that such a design fails to capture the detailed structure of small objects due to increased intervals of the kernel weights. Whereas increasing dilation rates is essential in terms of resolution and context, it can be detrimental to small objects, which is especially undesirable in RS applications.

In this article, we solve this problem by a simple but effective design: "increasing-decreasing" dilation rates. We show that decreasing dilation rates afterward promotes feature sharing between nearby pixels, which helps refine fine details of small objects missed by the sparse dilated kernel. Specifically, we propose a local feature extraction (LFE) module that consists of several convolutional layers with decreasing dilation rates. The LFE module can be attached to any conventional architecture with increasing dilation rates to boost performance for small objects.

We have conducted comprehensive evaluations on six RS datasets. On most of the datasets, the proposed model outperforms state-of-the-art semantic segmentation models. We observe that the performance gain is especially high for small objects. We further analyze the problem of "increasing" dilation rates by visualizing the effective receptive field (ERF) [12], where we find undesirable grid-like patterns in the ERF when we use "increasing" dilation rates.

The contributions of the article are summarized as follows.

1) A dedicated architecture for extracting small objects from RS imagery is proposed. Unlike previous methods, the proposed architecture has no downsampling layers and can extract full-resolution features with large RFs by dilated convolutions;
2) We reveal the problems of the conventional dilated backbone and find a solution, i.e., "increasing-decreasing" dilation rates, based on an in-depth analysis of the network connectivity;
3) The proposed model outperforms the state-of-the-art models on instance-level metrics in multiple benchmark datasets. Despite its simplicity, it significantly improves the accuracy for small objects.

The rest of this article is organized as follows. Section II provides related works in RS and computer vision literature. Section III presents the proposed method in detail. In this section, we explain the problem of the naive application of dilated convolutions, i.e., the "increasing" dilation rate, and explain why the proposed "increasing-decreasing" dilation rate solves the problem. Section IV introduces instance-level evaluation metrics for RS object segmentation tasks and analyzes their importance compared to the conventional pixel-level metrics. Section V evaluates the proposed method against baselines and state-of-the-art segmentation models and reports the results of ablation studies. Section VI gives the interpretation of the experimental results and the limitation of the proposed method. Finally, Section VII concludes this article.

This article is an extended version of our prior work [13] accepted by WACV. The improvements from the conference version are as follows.

1) Proposal and evaluation of the alternative solution for the problem of dilated convolutions: We proposed an incrementally dilated (ID) backbone as an alternative solution for the problem of dilated convolutions in Section III-C. We compare the backbone to the LFE module in Section V-C5, where we find the LFE module is a better approach for addressing the problem of dilated convolutions.

2) Improved accuracy with modern CNN architectures: The proposed method is applied to the more recent CNN architectures, ResNet [14], and ConvNeXt [15], bringing significant performance improvement. The experiment also verifies the generalizability of the proposed method for different CNN architectures.

II. Related Works

In recent years, deep neural networks have achieved great success in the semantic segmentation of natural scene images [16], [17], [18], [19], [20], [21]. Also in RS, some techniques for natural scenes have been successfully applied to the RS imagery, such as multiscale feature fusion [2], [3], [4], [5], [6], [7], [8], [22], context aggregation [6], [23], [24], [25], and attention-based methods [26], [27], [28]. However, the accuracy for small objects (e.g., XS and S size objects in Fig. 2) is still insufficient due to the lost spatial information by downsampling layers. Our paper's novelty lies in its no-downsampling architecture to overcome this limitation, based on the dilated convolutions with novel "increasing-decreasing" dilation rates. Below, we first introduce the semantic segmentation methods in natural scenes in Section II-A. We then introduce the segmentation methods in RS and discuss the difference to our method in Section II-B. Finally, we introduce other related techniques to enhance the accuracy for small objects in Section II-C.

A. Semantic Segmentation of Natural Scene Images

Semantic segmentation is a task that assigns each pixel in an input image a semantic category. The pioneering work of Long et al. [16] first extended well-studied classification CNN architecture onto the semantic segmentation task. Since then, extensive studies have been made on developing CNNs for the task. A primary focus of the previous works is the utilization of higher resolution feature maps. Commonly, the resolution of feature maps in a CNN is gradually lost through layers due to subsampling operations. The resulting coarse-resolution feature

maps do not contain sufficient spatial information for accurate segmentation. To remedy this, Long et al. [16] proposed to fuse multiresolution predictions that are extracted from different stages of a network. In [29], a multipath refinement method is proposed to recover spatial fine details by fusing low-level encoder features. A line of works [17], [30], [31], [32] introduced decoder architecture in which resolution of feature maps was gradually recovered by employing inverse operation of pooling layer [30], [31] or skipping and fusing high-resolution feature maps from an encoder part [17], [32], [33]. The other line of works [11], [34], [35] maintained feature resolution through layers by utilizing dilated convolutions [10], [11] instead of subsampling operations. The works in [34] and [35] also proposed decreasing dilation rates to remedy grid noise in the predictions. Although the methods share a similar concept to our work, the target of the methods is natural scene images, and they still rely on subsampling operations. On the other hand, our method has no subsampling operations and is more suitable for handling small objects in RS imagery. Another focus of the previous works is the utilization of contextual information. The works of [18] and [32] constructed a multiscale feature pyramid to aggregate contextual information. More recent works [21], [36], [37], [38], [39] proposed an adaptive feature aggregation method based on feature relation. For instance, in OCNet [21], contextual features were aggregated through a self-attention mechanism.

B. Semantic Segmentation in RS

In the RS domain, semantic segmentation CNNs have been applied to various tasks, such as building detection, vehicle detection, or road extraction. The improvements in computer vision were also found to be effective for RS, e.g., FCN architecture [40], [41], multiscale feature fusion [42], and an encoder-decoder architecture [2].

Multiscale feature fusion: The most famous architecture for multiscale feature fusion is an encoder-decoder architecture. In the architecture, the encoder first extracts downsampled features with high-level semantics, and the decoder recovers the fine details by fusing encoder features in a coarse-to-fine manner. Many previous works tried to improve the architecture [2], [3], [4], [5], [6], [7], [8], [22], [43]. Several works improved skip connection by enhancing skipped features by attention mechanism [4], [5], [6], [22], [43]. In [43], a feature alignment method is proposed to align the decoder feature to the skipped feature using feature association. In [22], the feature fusion strategy is improved by utilizing intermediate building and edge predictions as supervised spatial attention maps. Other works improved encoder and decoder architecture by introducing residual learning [3], [7] and multiscale feature fusion [8]. In [3], DenseNet [44] was combined with an encoder-decoder architecture to improve building detection performance. In [8], a standard bottleneck layer in ResNet [14] was enhanced by a Res2Net module that extracts and fuses multiscale features.

HRNet [20] is also a widely used architecture in RS. Unlike the encoder-decoder, it extracts multiscale features in parallel and fuses them at intermediate stages. The architecture can better extract the spatial details in the high-resolution branch, resulting in better segmentation accuracy, especially around edges. HRNet has been improved for RS tasks, e.g., by employing a channel/spatial attention [45], [46] and a spatial pyramid module [47].

Recently, several methods proposed to use multiple convolution kernels, such as inception architecture to extract multiscale features [48], [49], [50]. In [49], the inception-like module and channel attention module are utilized to enhance multiscale features. Wang et al. [50] proposed a spatial pyramid block for multiscale feature extraction. In the block, multiscale features are extracted by multiple parallel depthwise separable convolutions with different dilation rates, and the features are then fused by a pointwise convolution and a scale-attention operation. Since the concept of the dilated spatial pyramid (DSP) can be an alternative solution to the problem of dilated convolutions, we conducted comparative experiments in Section V-C5.

Context aggregation: Contextual information is also beneficial for RS imagery. In [6] and [23] the multiscale feature pyramid module such as ASPP [32] was utilized for context aggregation. Zhou et al. [24] improved the pyramid module by applying self-attention between the multiscale features. Chen et al. [25] utilized vision transformers [51] for enlarging RF size, where they applied spatial and channel attention on the sparsely sampled tokens.

Attention-based methods: Recent methods utilize attention-based architectures such as Vision Transformers [51] as they perform better than CNNs in many cases. In [26], dual spatial attention was proposed that incorporates global context and local details by applying global self-attention with downsampling and local attention with efficient stripe convolution. Zhang et al. [27] proposed a shunted dual skip connection that enhances multiscale information inside the ViT backbone and skips encoder attention maps to the decoder to better exploit the similarity information inside the encoder. In [28], a ViT backbone is enhanced by complementing the local fine detail with a convolutional branch.

The most crucial difference between the works above and ours is that the previous works rely on downsampling operation at the encoder stage, which loses detailed spatial information of small objects. Although the multiscale feature fusion methods such as UNet [17] and HRNet [20] can partially alleviate the issue, fusing low-level features often introduces noisy information into the feature maps, which deteriorates the accuracy for small objects. On the other hand, our method does not use downsampling. It maintains full resolution throughout the network, enabling the model to extract deep features with high resolution and large RFs. Most related to our work is the work of Sherrah [52], which utilized dilated convolutions for the encoder. However, they used maxpooling layers with a stride of 1 after each dilated convolution layer, which decreases the actual resolution of the extracted feature maps.

C. Other Related Techniques

We summarize the other related techniques for addressing small objects in RS imagery. Note that the methods below are orthogonal to our work.

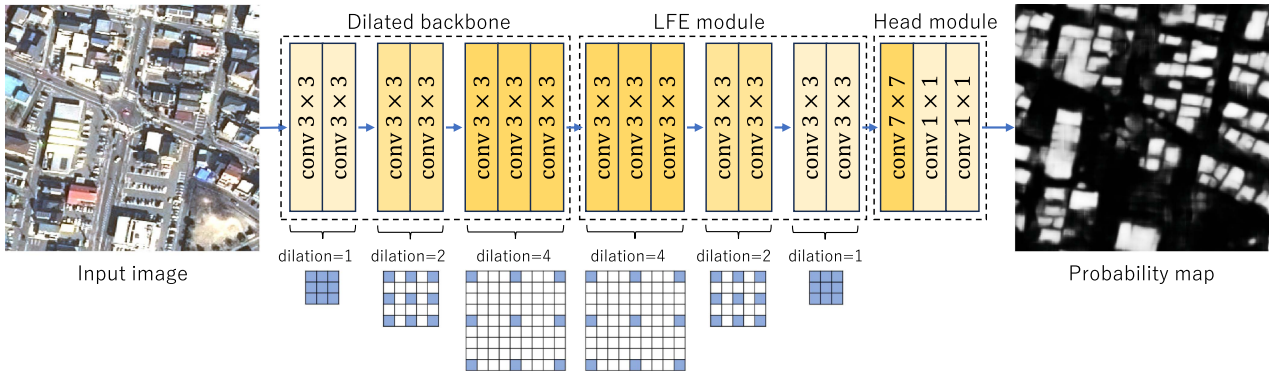

Fig. 3. Overview of the proposed network architecture. The network consists of a dilated backbone, a LFE module, and a head module. The dilated backbone consists of dilated convolutions with "increasing" dilation rates and has no downsampling layers. The LFE module also consists of dilated convolutions but with "decreasing" dilation rates. The dilated kernels are depicted below each block, where the blue cells in the grid represent the sampling location of the dilated kernel.

Boundary awareness: To accurately segment crowded small objects, several works performed boundary refinement by using boundary aware loss functions [8], [53], [54] or directly learning boundaries in multitask learning [46], [55]. In [55], semantic boundary extraction was simultaneously learned in a multitask manner to enhance vehicle segmentation. Also, Yan et al. [46] incorporated boundary information by introducing a boundary refinement module and fusing boundary-aware features for building detection. In [53], the network was trained to predict the distance from each pixel to the nearest object boundaries. The distance gives richer supervision about object boundaries than binary object masks, which results in precise localization of object boundaries.

Postprocessing: Postprocessing with conditional random fields [56] was also found to be effective for RS [57]. In [58], guided filter [59] was used as postprocessing to acquire enhanced predictions.

Super resolution: Enhancing the resolution of input images by super-resolution was found to be effective for small object detection, such as buildings [60], vehicles [61], and general objects [62]. Rabbi et al. [63] further proposed an edge-enhanced super-resolution method dedicated to small object detection.

III. Proposed Method

To detect small objects, we should pay attention to both RF size and the resolution of feature maps. High-level semantic features with large RFs are necessary even for small objects [9] because contextual information around them provides a crucial clue for recognition. When only given low-level local features, models will fail to distinguish features of small objects from other irrelevant features and produce many false alarms. A higher spatial resolution is also crucial. With a coarse-resolution feature map, a model will miss small objects or over-segment them together with adjacent objects.

The proposed model is designed to satisfy both demands. Fig. 3 illustrates a schematic of the model consisting of three parts: a dilated backbone, a LFE module, and a segmentation head. The dilated backbone extracts a high-resolution feature map with a large RF by utilizing dilated convolutions with increasing dilation rates (see Section III-A). As we will show in Section III-B, the dilated backbone with conventional "increasing" dilation rates fails to capture the local relationships between nearby features. To solve the problem, the LFE module with decreasing dilation rates is attached after the dilated backbone, which results in a novel scheme of "increasing-decreasing" dilation rates. Finally, the refined feature map from the LFE module is fed into the segmentation head that outputs a probability map of the target objects.

A. Dilated Backbone

As a preliminary, we first explain the dilated convolutions [11] in more detail. A standard $k \times k$ convolution computes the output by applying kernel weight $\boldsymbol{w}$ on the small $k \times k$ region sampled from the input. Let a grid $\mathcal{R} = \mathcal{X} \times \mathcal{Y}$ be the set of sampling positions centered at (0,0), that is, $\mathcal{X}, \mathcal{Y} \in \{-\frac{k-1}{2}, \ldots, 0, \ldots, \frac{k-1}{2}\}$. The convolution at position $\boldsymbol{p}$ can then be represented as follows:

$$\boldsymbol{z}(\boldsymbol{p}) = \sum_{\boldsymbol{\delta} \in \mathcal{R}} \boldsymbol{w}(\boldsymbol{\delta}) \cdot \boldsymbol{x}(\boldsymbol{p} + \boldsymbol{\delta}). \tag{1}$$

In a dilated convolution with a dilation rate r, the sampling positions are strided by a factor of r as follows:

$$\boldsymbol{z}(\boldsymbol{p}) = \sum_{\boldsymbol{\delta} \in \mathcal{R}} \boldsymbol{w}(\boldsymbol{\delta}) \cdot \boldsymbol{x}(\boldsymbol{p} + r\boldsymbol{\delta}). \tag{2}$$

As a result, the kernel size is enlarged from $k \times k$ to $(r(k-1)+1) \times (r(k-1)+1)$. Thanks to the strided sampling position and the large kernel size, the dilated convolution can enlarge the RF size without losing the resolution of the feature map.

The dilated backbone is acquired by applying dilated convolutions on the existing CNN backbones, such as VGG [64] and ResNet [14]. The role of the module is 1) extracting high-resolution feature maps and 2) aggregating a wide range of contextual information. To satisfy both of them, we remove all the downsampling layers from the module and use dilated convolutions instead. Accordingly, their dilation rates are increased by a factor of 2 at every removed pooling layer, resulting in "increasing" dilation rates. Note that the spatial resolution is kept throughout the module since the dilated kernels are densely applied to their input feature maps.

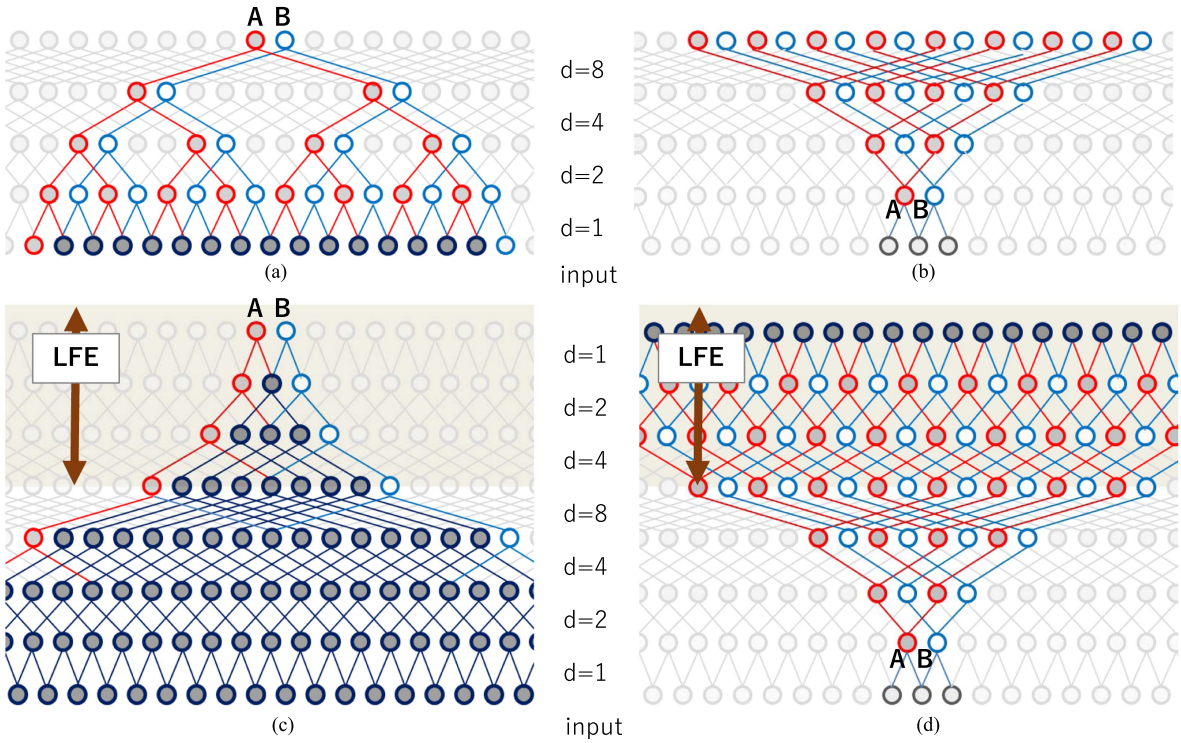

Fig. 4. (a) and (b) Describe the problems of the "increasing" scheme of dilation rates. (c) and (d) Describe how the problems are solved by the "increasing-decreasing" scheme with the LFE module. *(a) Problem on spatial inconsistency:* Focusing on the adjacent output units A and B, they share only the input features shown by dark color and share no intermediate features, which causes inconsistency between them. *(c) Solution:* Intermediate features are shared thanks to the LFE module, which ensures consistent output. *(b) Problem on local structure extraction:* Focusing adjacent intermediate features A and B, no units in higher layers receive the information from both A and B simultaneously, i.e., unaware of the local structure between A and B. *(d) Solution:* By adding the LFE module, features A and B are shared at the top layer (shown by dark color units), giving a chance to recognize the local structure between A and B.

B. LFE Module

Although the dilated backbone extracts high-resolution feature maps with a large RF, it fails to capture local relationships between nearby features due to its architecture with "increasing" dilation rates. Specifically, such an architecture causes two problems: 1) Spatial consistency between nearby features becomes weak, and 2) local structure cannot be extracted in higher layers. In this section, we first describe the problems in detail. We then introduce the LFE module as a solution for the problems. For simplicity, the analysis below is conducted for a 1-D convolution case, yet the results can be straightforwardly extended to a 2-D case.

1) Problem on Spatial Inconsistency: Fig. 4(a) shows a toy model with "increasing" dilation rates. The model consists of four 1-D convolutional layers with a kernel size of 2. The layers have increasing dilation rates of 1, 2, 4, and 8. In the figure, two outputs, "A" and "B," are extracted through the two extraction paths illustrated by red and blue edges. The problem is that the two paths only overlap at the input layer, meaning that the features in the middle layers are not shared at all for calculating the two adjacent outputs. As a result, the outputs "A" and "B" tend to have inconsistent values even though they are adjacent. In our experiments, we observe grid-like noise in the prediction of the network when we use "increasing" dilation rates.

2) Problem on Local Structure Extraction: Fig. 4(b) illustrates the second problem. In the figure, the two adjacent hidden features, "A" and "B," are propagated to higher layers through two propagation paths illustrated by red and blue edges. The problem is that the two paths do not overlap at all, i.e., units at higher layers receive information from either "A" and "B," not both. As a result, the network cannot recognize the local structure between "A" and "B," even though such information is essential for accurate boundary prediction in dense pixelwise labeling tasks. In RS scenarios where objects are small and crowded, the accuracy around boundaries is critical because crowded objects are easily over-segmented with inaccurate boundaries.

3) LFE Module: To solve the above two problems, we propose the LFE module that has *decreasing* dilation rate. Fig. 4(c) and (d) show network structures with the LFE module added on top of the toy models in Fig. 4(a) and (b), respectively. In the toy example, the LFE module has three convolutional layers with dilation rates of 4, 2, and 1. With such decreasing dilation rates, the above problems are successfully solved. The two extraction paths overlap sufficiently in Fig. 4(c), thus enhancing feature sharing between adjacent outputs. Moreover, the two propagation paths successfully overlap at the last layer in Fig. 4(d), giving chance to recognize local structure contained in the intermediate feature maps.

Below, we describe a more concrete explanation of why decreasing dilation rates solve the problem. Fig. 5 shows three convolutional networks, Net-d2, Net-d4, and Net-d8. Each network has a uniform dilation rate of 2, 4, and 8, respectively. Again, each path with the same color represents individual extraction

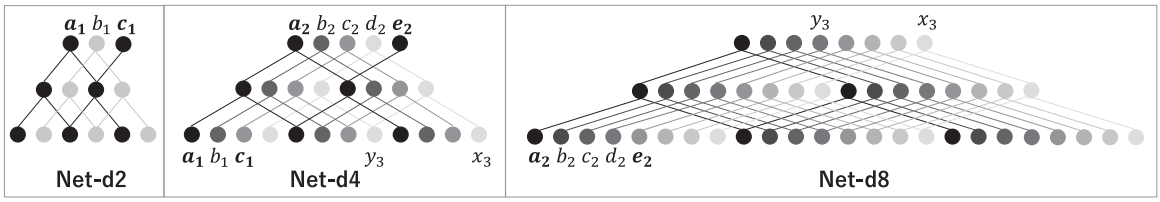

Fig. 5. Explains the detailed process on how the extraction and propagation paths become separated along with the increase of dilation rates. The figure shows three networks: Net-d2/d4/d8 with different dilation rates of 2, 4, and 8. Now, units a_1 and c_1 share the same path in Net-d2. When we stack Net-d4 on top of Net-d2, they belong to separate paths in Net-d4. When we further stack Net-d8, they are still on different paths and will not share the same path again as long as the dilation rate increases. Also, units a_2 and e_2 in Net-d4 become apart in Net-d8. In this way, increasing the dilation rate makes extraction paths (and propagation paths) branch off. In turn, the opposite happens if we stack in decreasing order (i.e., stacking Net-d4 on top of Net-d4 and Net-d2 on top of Net-d4). Although units x_3 and y_3 in Net-d8 belong to separate paths, they belong to the same path in Net-d4, and thus, their features can be shared in the layers above.

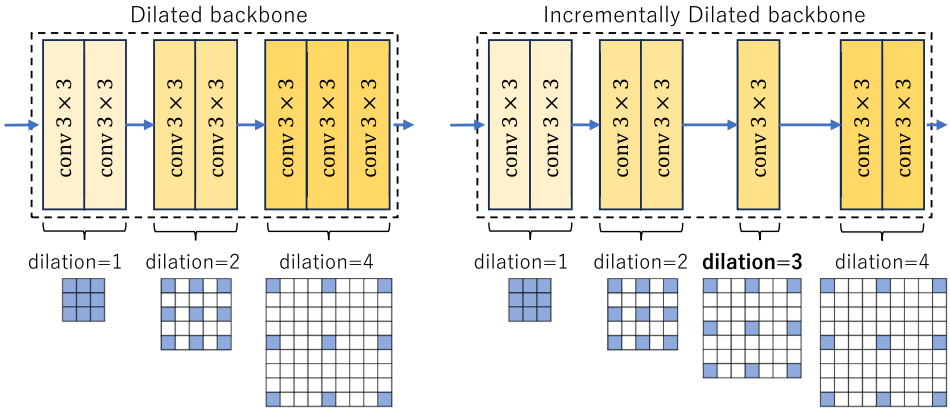

Fig. 6. Comparison between the dilated backbone and the ID backbone.

paths (propagation paths if seen upside down). As the dilation rate increases as 2, 4, and 8, the number of paths without overlap increases as 2, 4, and 8, accordingly. When these networks are stacked in increasing order, the two computation paths in Net-d2 branch off into four paths in Net-d4. Then, they further branch off into eight paths in Net-d8. In this way, computation paths become increasingly separated and never connected again as long as dilation rates are increased. In turn, if we stack these layers in decreasing order, the opposite happens: The eight paths in Net-d8 are connected into four paths in Net-d4 and then into two paths in Net-d2. Thus, the decreasing architecture gradually recovers the connection of the extraction/propagation paths. The connected extraction path promotes feature sharing among nearby positions and ensures consistent output. The connected propagation path allows local structures in the middle layers to be extracted in subsequent layers.

C. Alternative Solution: ID Backbone

Alternatively, we can also solve the problems by using incremental dilation rates. When dilated convolutions are used to replace downsampling layers, dilation rates are commonly increased by a factor of 2 at every downsampling layer to keep the same RF size. For example, let C_n be a conv3×3-ReLU block with a dilation rate of n and P be a maxpooling layer with a stride of 2, the architecture of VGG16 until third-stage can be written as C_1-C_1-P-C_1-C_1-P-C_1-C_1-C_1. Then, the corresponding dilated backbone has dilation rates increased by a factor of 2: C_1-C_1-C_2-C_2-C_4-C_4-C_4 (see Fig. 6, left). On the contrary, the ID backbone increases dilation rates one by one. For instance, we can build the backbone by changing C_4 of the above backbone into C_3: C_1-C_1-C_2-C_2-$\boldsymbol{C_3}$-C_4-C_4 (see Fig. 6, right).

Fig. 7 illustrates how the incremental design can solve the problems. The figure shows a 1-D toy architecture with incremental dilation rates of 1-2-3-4. As we can see, the two problems of the dilated backbone are solved; the extraction paths of outputs "A" and "B" overlap at the second and the third layer [see Fig. 7(a)]. Also, the propagation paths of the features "A" and "B" overlap at the third and the last layer [see Fig. 7(b)].

The key property is to have a dilation rate not divisible by the previous layer's dilation rate. To explain this more in-depth, we illustrate three convolutional networks, Net-d2, Net-d3, and Net-d4, with incremental dilation rates, 2, 3, and 4, in Fig. 7(c). When we stack the networks in increasing order, the units a_1 and d_1 are connected to the same path at Net-d3 while they are on a different path at Net-d2. Similarly, the units a_2 and e_2 separated at Net-d3 are connected at Net-d4. This can happen because the dilation rate of a layer is not divisible by that of the previous layer. Taking Net-d2 and Net-d3 as an example, units in each network have repeated structures; every two units share the same path in Net-d2, and every three units share the same path in Net-d3. Because three is not divisible by two, the repeated structure is not aligned between Net-d2 and Net-d3, resulting in the recovered connection at Net-d3. Obviously, this cannot happen when the dilation rates are increased by a factor of 2.

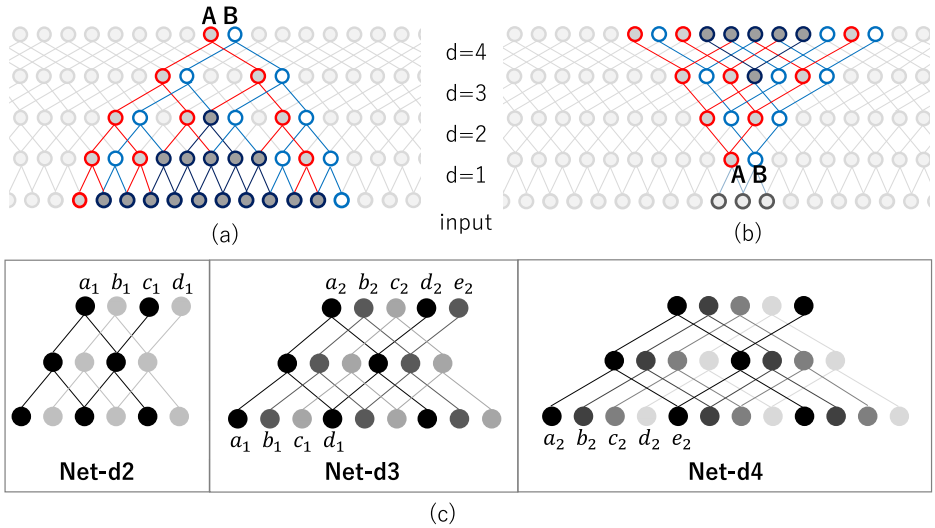

Fig. 7. Describes how the incremental dilation rate solves (a) problem on spatial inconsistency and (b) problem on local structure extraction. Instead of increasing dilation rates by a factor of 2 (i.e., 1-2-4), the architecture in the figure increases dilation rates one by one (i.e., 1-2-3-4). (c) Shows three toy networks with incremental dilation rates, 2, 3, and 4. The units in the same color share the same extraction/propagation path. As we can see, units a_1 and d_1 in Net-d2 are connected at Net-d3. Also, units a_2 and e_2 in Net-d3 are connected at Net-d4.

IV. Evaluation Metrics

Pixel-level metrics are widely used for evaluation in many ground object segmentation tasks. However, as we show in this section, the pixel-level metrics tend to ignore errors on small objects, which is not desirable for many practical applications. Instead, we propose utilizing instance-level metrics to evaluate the ground object segmentation.

Below, we first explain conventional pixel-level metrics in Section IV-A, demonstrate the problem of the metrics in Section IV-B, and introduce a method for utilizing instance-level metrics in Section IV-C.

A. Pixel-Level Evaluation Metrics

Pixel-level metrics such as pixel F1 score or intersection over union (IoU) are widely used in semantic segmentation tasks. In RS image analysis, the metrics are also standard for object segmentation tasks such as building footprint extraction. These metrics are computed as follows:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{3}$$

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \quad \mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad \mathrm{F1} = \frac{2\mathrm{PR}}{\mathrm{P} + \mathrm{R}} \tag{4}$$

where TP is the number of correctly classified foreground object pixels, FP is the number of background pixels wrongly classified as foreground objects, and FN is the number of foreground object pixels wrongly classified as background.

B. Problem on Pixel-Level Metrics

The problem on the pixel-level metrics is twofold: 1) Pixel-level metrics tend to neglect errors on small objects. Since small objects occupy few pixels, they have a relatively small impact on the metrics compared to large objects. Thus, the metrics prefer models that can correctly detect large objects even if they miss many small objects. 2) Pixel-level metrics tend to neglect errors on the boundaries between objects. This is because the thin boundary region consists of few pixels and has a relatively small impact on the metrics. As a result, the pixel-level metrics overlook the failure case where several small objects are detected as one large object.

Fig. 9 shows an example of the problem. The PRED1 overestimates several small buildings into one large mask, and the PRED2 better extracts the individual small buildings. However, the pixel-level metrics assign a higher score for PRED1 because the shape of the most prominent object on the top is more accurate in PRED1.

This behavior is not desirable in many practical applications because 1) the large object size does not necessarily mean the importance of the object, and 2) the number of objects is more meaningful than their shape in many cases. For instance, correctly recognizing and counting small buildings is essential to estimating the number of households and the population of a city. For monitoring activities of commercial facilities, the number of passenger cars is more important than the number of larger vehicles, such as buses or trucks. For such purposes, the performance of a model should be evaluated based on instance-level metrics.

C. Instance-Level Metrics

In this article, we propose to evaluate the ground object segmentation task with instance-level metrics, $\mathrm{AP}_{\mathrm{vol}}$ [65] and AR [44]. To do this, we developed several preprocessing steps for converting pixel-level output probability maps into instance-level detection results compatible with the instance-level evaluation pipeline.

Fig. 8 shows the specific steps for the preprocessing. We first extract binary masks by thresholding the probability maps and

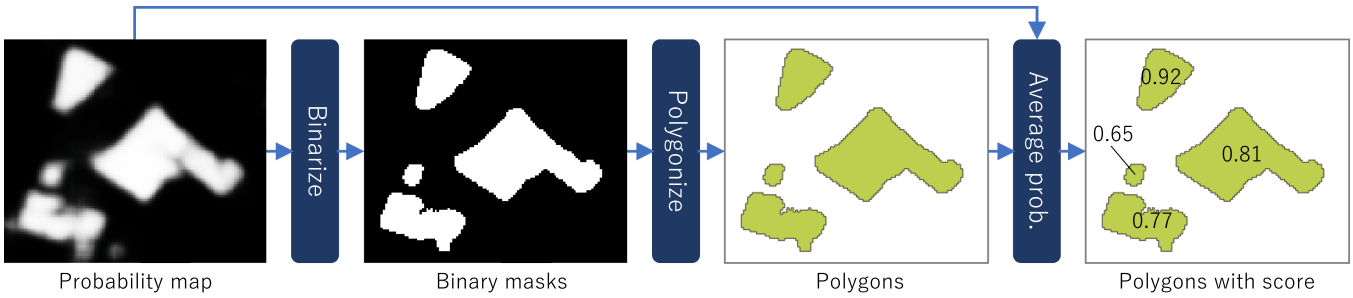

Fig. 8. Schematics of the preprocessing for evaluating ground object segmentation results with instance-level metrics such as AP_{vol}.

	PRED1	PRED2
F1 (pixel)	**97.1%**	93.3%
IoU (pixel)	**76.6%**	72.6%
AP_{vol} (instance)	10.2%	**37.5%**

Fig. 9. Comparison of evaluation metrics (pixel versus instance). The pixel-level metrics prefer PRED1, which accurately segments the large object, whereas the instance-level metric prefers PRED2, which accurately segments individual small objects. PRED1 and PRED2 are acquired from the prediction result of DeepLabV2 [32] and the proposed VGG-D-LFE on Massachusetts Buildings Dataset.

detect polygons from the binary masks. We then aggregate the probabilities of pixels inside each polygon and compute the confidence score as an average of the probabilities. Once we get the polygons and their confidence scores, we can compute instance-level metrics in the same way as instance segmentation.

For completeness, we explain the computation of the metrics in the following. We define a ground truth polygon as correctly predicted when the IoU with a predicted polygon is larger than a threshold τ_{iou}. Let s_n be the threshold for confidence score that satisfies $0 = s_0 < s_1 < \ldots < s_N < 1$, instance-level precision and recall can be computed for each threshold s_n as follows:

$$\mathrm{R}_n = \frac{\mathrm{TP}(s_n)}{\mathrm{TP}(s_n) + \mathrm{FN}(s_n)}, \quad \mathrm{P}_n = \frac{\mathrm{TP}(s_n)}{\mathrm{TP}(s_n) + \mathrm{FP}(s_n)} \tag{5}$$

where $\mathrm{TP}(s_n)$ is the number of correctly detected foreground object instances, $\mathrm{FP}(s_n)$ is the number of predicted instances wrongly detected as foreground, and $\mathrm{FN}(s_n)$ is the number of foreground object instances missed in prediction. The precision-recall curve is acquired by plotting the precision and recall for each score threshold s_n. The average precision is then computed as the area under the precision-recall curve as follows:

$$\mathrm{AP}_{\tau_{\text{iou}}} = \frac{1}{N} \sum_{n=1}^{N} (\mathrm{R}_n - \mathrm{R}_{n-1})\, \mathrm{P}_n. \tag{6}$$

Finally, the AP_{vol} is computed by averaging the AP evaluated on various IoU thresholds τ_{iou} sampled from [0.1,0.9] as follows:

$$\mathrm{AP}_{\text{vol}} = \frac{1}{|\mathcal{T}|} \sum_{\tau_{\text{iou}} \in \mathcal{T}} \mathrm{AP}_{\tau_{\text{iou}}} \tag{7}$$

where we set $\mathcal{T} = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. The average recall can be computed for each score threshold s_n as follows:

$$\mathrm{AR}_n = \frac{1}{|\mathcal{T}'|} \sum_{\tau_{\text{iou}} \in \mathcal{T}'} \mathrm{R}_n(\tau_{\text{iou}}) \tag{8}$$

where we set $\mathcal{T}' = \{0.5, 0.6, 0.7, 0.8, 0.9\}$. $\mathrm{R}_n(\tau_{\text{iou}})$ is a recall evaluated at the IoU threshold τ_{iou} and the score threshold s_n. In our experiments, we use the average recall at the lowest score threshold s_0 as follows:

$$\mathrm{AR} = \mathrm{AR}_0. \tag{9}$$

V. Experiments

In this section, we validate our method on six RS datasets and across three foreground object segmentation tasks: building detection, vehicle extraction, and road extraction. Fig. 10 shows the distribution of object size for four of the six datasets. We can see that the small objects (i.e., XS and S size) are not minor in the datasets; XS and S size objects occupy over 20% of the objects in the datasets, and the proportion reaches 95% in the Massachusetts Buildings Dataset. In the experiments, we focus especially on the small objects, showing how previous methods fail on these objects and how the proposed method can improve their performance.

As an additional attempt, we apply our method to the road extraction task. Although roads are not "small objects," they are characteristic in their thin structure. For instance, most of the roads in DeepGlobe Road Extraction datasets have widths of less than 20 pixels, which requires special care in the feature resolution and the RF size. Thus, we conduct experiments to evaluate the effectiveness of our method on such thin targets.

Below, Section V-A introduces datasets used in our experiments, Section V-B explains basic setups for all the experiments, and Section V-C–V-E describe the results on each of the three tasks.

A. Datasets

We used four building detection datasets (Toyota City Dataset, Massachusetts Buildings Dataset [66], DeepGlobe Building Detection Dataset [1], and Inria Aerial Image Labeling Dataset [67]), one vehicle detection dataset (Vaihingen Dataset [68]), and one road extraction dataset (DeepGlobe Road Extraction Dataset [1]).

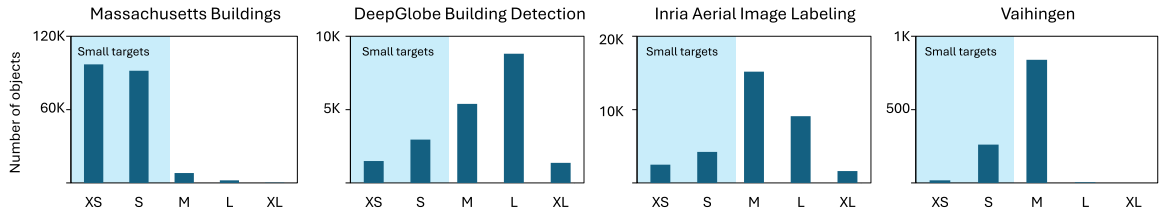

Fig. 10. Distribution of object size for each dataset. The five sizes {XS, S, M, L, XL} represent the object's area in a range of [0,100], [100,400], [400,1600], [1600,6400], and over 6400 pixels.

Toyota City Dataset: To evaluate the proposed method, we created the Toyota City Dataset, a dataset for building detection. The dataset is composed of satellite imagery around Toyota City, Japan. The images were acquired by Pleiades satellite in 2014. Training and test data covers roughly 200 km^2 and 20 km^2, each containing 100 000 and 15 000 buildings. The image resolution is 50 cm, and we used RGB bands for training and evaluation. Labels are provided for two classes: building or nonbuilding for each pixel. From the dataset, we collected 400 K patches for training. The dataset is our original dataset and is not publicly available at present.

Massachusetts Buildings Dataset: The dataset comprises 1 m spatial resolution aerial imagery with RGB bands. The dataset covers roughly 340 km^2 with 194 070 buildings for training and 23 km^2 with 15 261 buildings for testing. We collected training patches in the same way in [69] and did not conduct data balancing. From the dataset, we collected 5 M patches for training.

DeepGlobe Building Detection Dataset: The dataset comprises 30 cm resolution satellite images of size 640 × 640 and corresponding building footprints. The images are collected from four different cities in the world: Vegas, Paris, Shanghai, and Khartoum. In total, the number of training and validation images are 10 593 and 3526, respectively. From these images, we collected about 5 M patches for training.

Inria Aerial Image Labeling Dataset: Inria Aerial Image Labeling Dataset [67] is a dataset for building detection from aerial images. The dataset comprises image tiles of 5000 × 5000 pixels and 30 cm spatial resolution. The training set contains 180 images from five cities: Austin, Chicago, Kitsap Country, Western Tyrol, and Vienna, covering 405 km^2. As in [67], we used the first five image tiles of each city as a validation set. We collected about 5 M patches for training from the remaining image tiles.

Vaihingen Vehicle Dataset: For the vehicle detection task, we used the dataset curated from Vaihingen Dataset [68]. Vaihingen Dataset is provided by Commission III of the ISPRS [68]. The dataset is composed of 9 cm spatial resolution aerial imagery. We used near-infrared, red, and green bands and did not use a digital surface model. The dataset includes 16 labeled scenes covering roughly 0.6 km^2. As in previous works [42], [52], [70], we used five scenes (IDs: 11, 15, 28, 30, 34) for validation and the remaining 11 scenes for training. Labels are provided for six classes: impervious surface, building, low vegetation, tree, car, and clutter/background. We set car class as our target and used only car labels for training and testing. From the dataset, we collected 200 K patches for training.

DeepGlobe Road Extraction Dataset: For the road extraction task, we used the DeepGlobe Road Extraction Dataset [1]. The dataset contains 6226 satellite imagery in size 1024 × 1024 and corresponding mask images for road annotations. The imagery has a 50 cm resolution, captured from satellites over Thailand, Indonesia, and India. The dataset is divided into 4696 and 1530 images for training and validation sets. We evaluated our method on the validation set since labels for the testing set are not publicly available. We collected about 5 M patches for training from the dataset.

B. Basic Setups

We evaluated the proposed method on three types of base networks: VGG16 [64], ResNet18 [14], and ConvNeXt-Tiny [15]. For each base network, we built three types of backbones (plain, dilated, and ID) and two types of attachment modules (Keep and LFE). The architecture of the components is shown in Table I. Using the components, we built baselines and the proposed models as below.

Plain baseline (VGG-P/ResNet-P/ConvNeXt-P): To begin with, we built a plain model without the LFE module and dilated convolutions. The model consists of a plain backbone and a segmentation head with three convolution layers. As shown in Table I, the plain backbone has a similar architecture to VGG16/ResNet18/ConvNeXt-Tiny. Specifically, we utilized the architecture until the third stage of the base networks. For ResNet18 and ConvNeXt-Tiny, we removed the downsampling at the first convolution layer by changing the stride of the layer to 1.

Dilated baseline (VGG-D/ResNet-D/ConvNeXt-D): We built a model with a dilated backbone to evaluate the naive use of dilated convolution with "increasing" dilation rates. As shown in Table I, the dilated backbone is acquired by replacing all the downsampling layers of the plain backbone with the dilated convolutions.

Proposed model with LFE module (VGG-D-LFE/ResNet-D-LFE/ConvNeXt-D-LFE): Finally, we attached the LFE module on the dilated backbone to validate the effect of the proposed LFE module.

Strong baseline (VGG-D-Keep/ResNet-D-Keep/ConvNeXt-D-Keep): Because the LFE module increases the parameter size of the proposed model, we also built a strong baseline with the same parameter size as the proposed model for a fair comparison. Specifically, we attached the Keep module that has the same parameter size as the LFE module. The only but essential

TABLE I
ARCHITECTURES OF THE BACKBONES AND THE ATTACHMENT MODULES USED IN THE EXPERIMENTS

	Backbone									Attached module					
	Plain (P)			Dilated (D)			Incrementally Dilated (ID)			Keep			LFE		
	Layer	Width	Stride/Dilation	Layer	Width	Stride/Dilation	Layer	Width	Stride/Dilation	Layer	Width	Stride/Dilation	Layer	Width	Stride/Dilation
VGG	○ C3	64	s1 d1	○ C3	64	s1 d1	○ C3	64	s1 d1	● C3	256	s1 d4	● C3	256	s1 d4
	○ C3	64	s1 d1	○ C3	64	s1 d1	○ C3	64	s1 d1	● C3	256	s1 d4	● C3	256	s1 d4
	● Max Pooling			—			—			● C3	256	s1 d4	● C3	256	s1 d4
	○ C3	128	s1 d1	● C3	128	s1 d2	● C3	128	s1 d2	● C3	256	s1 d4	● C3	256	s1 d2
	○ C3	128	s1 d1	● C3	128	s1 d2	● C3	128	s1 d2	● C3	256	s1 d4	● C3	256	s1 d2
	● Max Pooling			—			—			● C3	256	s1 d4	○ C3	256	s1 d1
	○ C3	256	s1 d1	● C3	256	s1 d4	● C3	256	s1 d3	● C3	256	s1 d4	○ C3	256	s1 d1
	○ C3	256	s1 d1	● C3	256	s1 d4	● C3	256	s1 d4						
	○ C3	256	s1 d1	● C3	256	s1 d4	● C3	256	s1 d4						
ResNet	○ C7	64	s1 d1	○ C7	64	s1 d1	○ C7	64	s1 d1	● R	256	s1 d4	● R	256	s1 d4
	● Max Pooling			—			—			● R	256	s1 d4	● R	256	s1 d4
	○ R	64	s1 d1	○ R	64	s1 d1	○ R	64	s1 d1	● R	256	s1 d4	● R	256	s1 d2
	○ R	64	s1 d1	○ R	64	s1 d1	○ R	64	s1 d1	● R	256	s1 d4	● R	256	s1 d2
	● R	128	s2 d1	● R	128	s1 d2	● R	128	s1 d2	● R	256	s1 d4	○ R	256	s1 d1
	○ R	128	s1 d1	● R	128	s1 d2	● R	128	s1 d2	● R	256	s1 d4	○ R	256	s1 d1
	● R	256	s2 d1	● R	256	s1 d4	● R	256	s1 d3						
	○ R	256	s1 d1	● R	256	s1 d4	● R	256	s1 d4						
ConvNeXt	○ C4	96	s1 d1	○ C4	96	s1 d1	○ C4	96	s1 d1	● CX	384	s1 d4	● CX	384	s1 d2
	○ CX	96	s1 d1	○ CX	96	s1 d1	○ CX	96	s1 d1	● CX	384	s1 d4	● CX	384	s1 d2
	○ CX	96	s1 d1	○ CX	96	s1 d1	○ CX	96	s1 d1	● CX	384	s1 d4	○ CX	384	s1 d1
	○ CX	96	s1 d1	○ CX	96	s1 d1	○ CX	96	s1 d1	● CX	384	s1 d4	○ CX	384	s1 d1
	● C2	192	s2 d1	● C2	192	s1 d2	● C2	192	s1 d2						
	○ CX	192	s1 d1	● CX	192	s1 d2	● CX	192	s1 d2	Ck: $k \times k$ Convolutional Block					
	○ CX	192	s1 d1	● CX	192	s1 d2	● CX	192	s1 d2	R: Residual Block					
	○ CX	192	s1 d1	● CX	192	s1 d2	● CX	192	s1 d3	CX: ConvNeXt Block					
	● C2	384	s2 d1	● C2	384	s1 d4	● C2	384	s1 d3	● : Downsampling					
	○ CX	384	s1 d1	● CX	384	s1 d4	● CX	384	s1 d3	○ : Layers with normal convolution					
	○ CX	384	s1 d1	● CX	384	s1 d4	● CX	384	s1 d4	● ● ● : Layers with dilated convolution					
	○ CX	384	s1 d1	● CX	384	s1 d4	● CX	384	s1 d4						

TABLE II
BATCH SIZE AND THE NUMBER OF TRAINING ITERATIONS FOR THE EXPERIMENTS

	VGG16		ResNet18		ConvNeXt-Tiny	
Dataset	Batch size	Train iter	Batch size	Train iter	Batch size	Train iter
Toyota City	100	30 K	-	-	-	-
Massachusetts	50	30 K	32	156 K	32	156 K
DeepGlobe Building Detection	50	100 K	32	156 K	32	156 K
Inria Aerial Image Labeling	50	100 K	32	156 K	32	156 K
Vaihingen	50	4 K	32	6 K	32	6 K
DeepGlobe Road Extraction	32	156 K	32	156 K	32	156 K

difference is that the Keep module does not have decreasing dilation rates. Instead, it keeps the same dilation rate throughout the attached layers.

ID backbone: We also built a model with an ID backbone to evaluate the effectiveness of the solution. The ID backbone is acquired by changing the dilation rates of the dilated backbone so that the rates increase incrementally (see Table I).

DSP backbone: As another possible solution for the problem of dilated convolution, we built a backbone with DSP convolutions, which is conceptually similar to [50]. Specifically, we split each convolutional layer of a backbone into multiple parallel convolutions with different dilation rates and smaller depth, and then concatenate the outputs of the parallel convolutions at the output. We changed a convolution layer with dilation rate 2 into two parallel convolutions with dilation rates {1,2}, and that with dilation rate 4 into three parallel convolutions with dilation rates {1,2,4}.

For all the experiments, the backbones were initialized using ImageNet pretrained weights. The other modules were randomly initialized using "xavier" initialization [71]. We used Adam [72] with the linear learning rate decay. For VGG-based models, we set the initial learning rate as 1.0×10^{-5} and multiplied it by a factor of 10 for the scratch layers. For ResNet- and ConvNeXt-based models, we set the initial learning rate as 1.0×10^{-4} and did not use the increased learning rate for the scratch layers. When comparing with the baseline models (Tables III, VII, and IX), we used the learning rate settings of the VGG-based models also for the ResNet- and ConvNeXt-based models (i.e., initial learning rate of 1.0×10^{-5} and increased learning rate at scratch layers). We set the weight decay coefficient as 1.0×10^{-4}. The batch size and the number of training iterations are shown in Table II.

The training patches were randomly cropped from source images and augmented by random rotation. For building detection, we conducted data balancing by sampling a subset of the collected patches to balance the number of positive and negative pixels across the training patches. Specifically, the collected patches were first divided into five bins according to the ratio of positive class pixels in the patch. From the bins, an equal number of patches were collected.

We evaluated the models using both pixel-level metrics (pixel F1 and IoU) and instance-level metrics (instance F1, AP_{vol}, and AR). To evaluate the instance-level performance for each object size, we evaluated AR for five object sizes: XS, S, M, L, and XL, each corresponding to the object's area in a range of [0,100], [100,400], [400,1600], [1600,6400], and over 6400 pixels.

C. Building Detection

This section describes the experimental results of the building detection task. As a preliminary experiment, we first confirm the importance of enlarging the RF for small buildings in

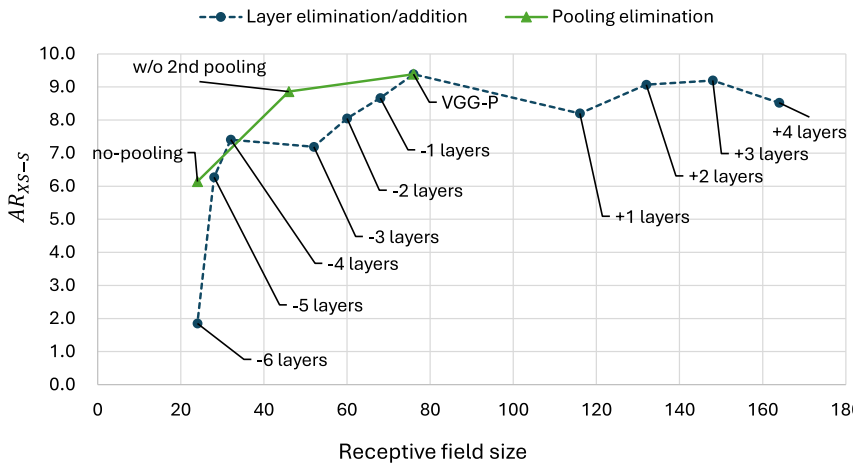

Fig. 11. Results of layer elimination/addition and pooling elimination. For each configuration, AR for small buildings (pixel area smaller than 1600 pixels) is shown. For layer elimination, we removed layers of the VGG-P one by one from top to bottom. For layer addition, we added layers in the 4th stage of the VGG16 one by one on top of the VGG-P. For pooling elimination, we eliminated the 2nd/all pooling layers from the VGG-P. The experiment is conducted on Toyota City Dataset.

Section V-C1. In Section V-C2, we empirically show the importance of high-resolution feature maps and the effectiveness of the proposed LFE module. In Section V-C5, we evaluate the effectiveness of the ID backbone and compare it with the LFE module. In Section V-C3, we compare the proposed model with the state-of-the-art segmentation models. In Section V-C6, we conduct an ablation study to validate the design choice of the proposed model. In Section V-C7, we visualize an ERF [12] of the models for further analyzing the effect of the LFE module.

1) Importance of Enlarging RF: In this section, we conducted preliminary experiments to verify the assumption that a model needs to have a sufficiently large RF to aggregate contextual information even for small objects. We have analyzed the impact of the RF size in two ways. First, we changed the depth of the backbone of VGG-P by eliminating/adding layers one by one (layer elimination/addition). Second, we eliminated pooling layers from VGG-P (pooling elimination). Fig. 11 shows the performance for small buildings (pixel area smaller than 1600 pixels) in Toyota City Dataset. In either experiment, the performance degrades as the RF size becomes small, showing the importance of enlarging the RF size for small objects.

2) Effect of the Dilated Backbone and the LFE Module: Table III shows the results on the four datasets. Below, we summarize the results.

(Plain versus Dilated): The use of the dilated backbone improves AP_{vol} of the plain baseline in many cases (seven out of ten cases), showing the importance of high-resolution feature maps. Significantly, the improvement on AP_{vol} reaches around 6% on Massachusetts Buildings Dataset (VGG-P versus VGG-D and ResNet-P versus ResNet-D).

(Keep versus LFE): The model with the LFE module outperforms its counterpart baseline (Keep) in nine of the twelve cases. The improvements are especially remarkable for small objects, showing the effectiveness of the LFE module.

We also applied dense-CRF [56] as a postprocessing on the Toyota City Dataset. The dense-CRF degrades instance-level performance because it over-segments a group of crowded small objects into one large mask due to their similar colors and ambiguous boundaries.

Fig. 12 qualitatively compares the outputs of VGG-D-Keep and VGG-D-LFE. As we see, grid noise appears in the prediction of VGG-D-Keep due to the spatial inconsistency problem discussed in Section III-B. On the contrary, the noise is successfully suppressed by utilizing the LFE module.

3) Comparison to the State-of-the-Art Models: Table IV compares the proposed models to the state-of-the-art semantic segmentation models. On instance-level metrics, the proposed model outperforms previous methods on most of the datasets. Focusing on the AR for each size, the proposed model performs remarkably well for small objects (see AR_{XS} and AR_S columns). On the other hand, previous methods such as DeepLab-V3+, PSPNet, and SwinTransformer show better accuracy for large objects (see AR_L and AR_{XL} columns). Accordingly, they also perform well on pixel-level metrics since the large objects significantly impact them. This result confirms that pixel-level metrics are preferred for large objects, while instance-level metrics are required to evaluate small objects.

4) Balancing Accuracy for Small and Large Objects With Model Ensembling: The proposed method performs better on small objects, while the previous methods perform better on large objects. Combining both methods can bring synergetic effects, leading to a method that is highly robust to the object size. Thus, we here evaluate the effectiveness of the combination by model ensembling. Fig. 13 compares the ensemble of the proposed and previous models ("Ours+PSPNet" and "Ours+Swin") with the ensemble of the previous models ("PSPNet+Swin"). As we see, the combination of previous models does not improve the accuracy for small buildings well. Conversely, combining the previous model with the proposed model significantly improves the accuracy for small buildings while keeping the accuracy for large buildings competitively well. The result shows that the proposed method is complementary to the previous methods in terms of object size, being a key technique to achieve robustness to object size.

TABLE III
EFFECT OF THE DILATED BACKBONE AND THE LFE MODULE FOR BUILDING DETECTION TASK

Dataset	Model	AP_{vol}	AR	AR_{XS}	AR_S	AR_M	AR_L	AR_{XL}
Toyota City Dataset	VGG-P	23.1	18.9	0.7	18.0	33.2	29.9	33.8
	VGG-D	25.3	21.9	0.9	23.5	36.2	29.2	26.5
	VGG-D-Keep	27.0	25.4	1.2	27.5	41.8	33.7	**40.0**
	VGG-D-LFE	**27.7**	**25.8**	**1.5**	**28.1**	**42.0**	**34.5**	33.1
	VGG-D-Keep-CRF	26.4	23.9	0.9	25.9	39.3	31.3	**42.5**
	VGG-D-LFE-CRF	**26.9**	**24.3**	**1.1**	**26.3**	**40.0**	**32.6**	28.1
Massachusetts Buildings Dataset	VGG-P	43.8	27.1	22.2	30.2	33.3	39.0	41.7
	VGG-D	49.9	33.4	21.9	37.9	38.0	41.7	36.2
	VGG-D-Keep	49.5	33.8	28.2	37.9	**38.5**	42.9	38.7
	VGG-D-LFE	**50.3**	**35.0**	**28.9**	**39.7**	38.2	**43.1**	**42.1**
	ResNet-P	40.7	23.1	15.0	25.7	30.4	34.2	31.1
	ResNet-D	**46.8**	**32.6**	**22.2**	**37.1**	31.8	38.0	37.8
	ResNet-D-Keep	44.8	30.3	19.9	34.2	**34.4**	**40.7**	**41.3**
	ResNet-D-LFE	45.8	30.3	19.9	34.4	32.4	39.5	38.1
	ConvNeXt-P	47.0	29.9	19.4	33.8	35.4	**40.8**	38.1
	ConvNeXt-D	45.9	28.5	18.2	32.2	32.9	40.7	**42.2**
	ConvNeXt-D-Keep	48.7	32.8	22.1	37.0	**36.3**	40.1	39.4
	ConvNeXt-D-LFE	**48.8**	**35.0**	**23.5**	**40.0**	35.9	38.7	41.9
DeepGlobe Building Detection	VGG-P	42.9	33.8	3.4	16.7	34.2	41.0	41.9
	VGG-D	42.8	34.9	4.9	18.9	35.2	41.5	42.2
	VGG-D-Keep	**50.8**	**44.1**	6.2	**25.9**	**45.6**	**52.7**	47.4
	VGG-D-LFE	49.8	41.3	**6.9**	23.9	41.1	49.4	**49.8**
	ResNet-P	47.3	37.0	3.1	16.8	37.0	46.5	43.6
	ResNet-D	46.3	37.4	5.3	20.2	37.7	44.7	46.4
	ResNet-D-Keep	47.4	41.3	5.8	**25.0**	**46.3**	47.5	33.0
	ResNet-D-LFE	**52.1**	**43.7**	**7.4**	24.5	43.5	**53.0**	**51.8**
	ConvNeXt-P	47.5	37.9	3.2	18.0	37.4	46.6	49.6
	ConvNeXt-D	48.3	38.7	4.1	18.4	37.7	47.9	50.1
	ConvNeXt-D-Keep	49.1	**41.0**	3.5	19.7	**41.0**	**52.0**	45.5
	ConvNeXt-D-LFE	**49.2**	40.2	**4.2**	**21.1**	39.4	49.1	**51.4**
	DeepLabV2	48.6	38.7	1.8	10.9	36.3	51.0	59.5
	DeepLabV2-Keep	49.6	39.4	1.8	10.7	36.9	52.4	58.4
	DeepLabV2-LFE	**50.1**	**40.4**	**2.2**	**13.3**	**38.0**	**52.8**	**60.7**
Inria Aerial Image Labeling Dataset	VGG-P	48.0	39.9	3.4	20.5	38.0	50.9	51.5
	VGG-D	49.9	42.9	5.5	24.1	41.3	52.9	52.5
	VGG-D-Keep	**56.9**	**50.2**	7.9	**30.2**	**48.3**	**60.3**	**60.7**
	VGG-D-LFE	56.2	49.2	**9.8**	**30.2**	46.5	59.2	59.6
	ResNet-P	51.8	43.3	3.7	21.5	40.4	54.8	56.6
	ResNet-D	52.6	45.2	6.9	26.1	43.1	55.3	54.7
	ResNet-D-Keep	54.8	48.9	6.0	28.7	**48.0**	59.1	53.3
	ResNet-D-LFE	**57.1**	**49.8**	**8.8**	**29.6**	47.0	**60.5**	**61.4**
	ConvNeXt-P	53.3	45.5	4.9	25.6	42.7	57.1	56.0
	ConvNeXt-D	54.5	46.9	6.6	26.7	43.7	58.1	53.5
	ConvNeXt-D-Keep	51.6	44.0	5.3	28.1	**47.3**	52.3	27.6
	ConvNeXt-D-LFE	**56.0**	**49.0**	**7.9**	**28.5**	46.2	**60.0**	**56.8**
	DeepLabV2	51.2	42.4	1.8	13.0	37.3	56.7	63.9
	DeepLabV2-Keep	**52.4**	**44.1**	**2.1**	**15.3**	**38.9**	**58.1**	**64.8**
	DeepLabV2-LFE	52.0	43.9	1.9	15.0	**38.9**	**58.1**	64.0

For ResNet- and ConvNeXt-based models, we use the same learning rate settings as the VGG-based models (initial learning rate of 1.0×10^{-5} and increased learning rate at scratch layers).
The bold values represent the best results.

5) Exploration of Alternative Solutions: Table V evaluates the effect of the ID backbone and the DSP backbone. In most cases, the DSP backbone performs worse than the dilated backbone. The effect of the ID backbone is inconsistent across datasets and architectures, showing improvement and degradation case by case. While the ID backbone performs well in some cases, the LFE module shows better performance and is, hence, a better choice for addressing the problem of dilated convolutions.

6) Design Choice of LFE Module: We further investigate the design choices of the LFE module. The investigations are conducted in three axes: 1) RF size of the module, 2) depth of the module, and 3) scheme for decreasing the dilation rate. We used VGG-D as the dilated backbone for all of the experiments.

First, we fixed the decreasing scheme to "monotonic decrease" and explored the RF size and the depth of the LFE modules. We controlled the RF size independently from the depth by adjusting the combination of dilation rates. For example, by changing dilation rates from 4-4-2-1 to 4-2-2-1, we can reduce the RF size while keeping the depth the same. In this way, we changed the RF size while keeping the depth the same and vice versa. Fig. 14 shows the result of the sensitivity analysis. We see that the performance is highly sensitive to the choice of the RF size, showing the importance of the parameter. In contrast, the performance is stable on the choice of the depth.

Next, in Table VI, we investigated several schemes of decreasing dilation rate: monotonic decrease (LFE-Dec), decrease twice (LFE-Dec-Dec), increase after dropping to 1 (LFE-Inc), and increase twice (LFE-Inc-Inc). The result shows that the models except LFE-Inc perform comparably well. The LFE-Inc might be inadequate because it decreases the dilation rate only at the beginning and increases the dilation rate in the latter part, which again causes the same problem stated in Section III-B. On the other hand, the repeated increasing architecture (LFE-Inc-Inc) performs well. Overall, we can conclude that any decreasing

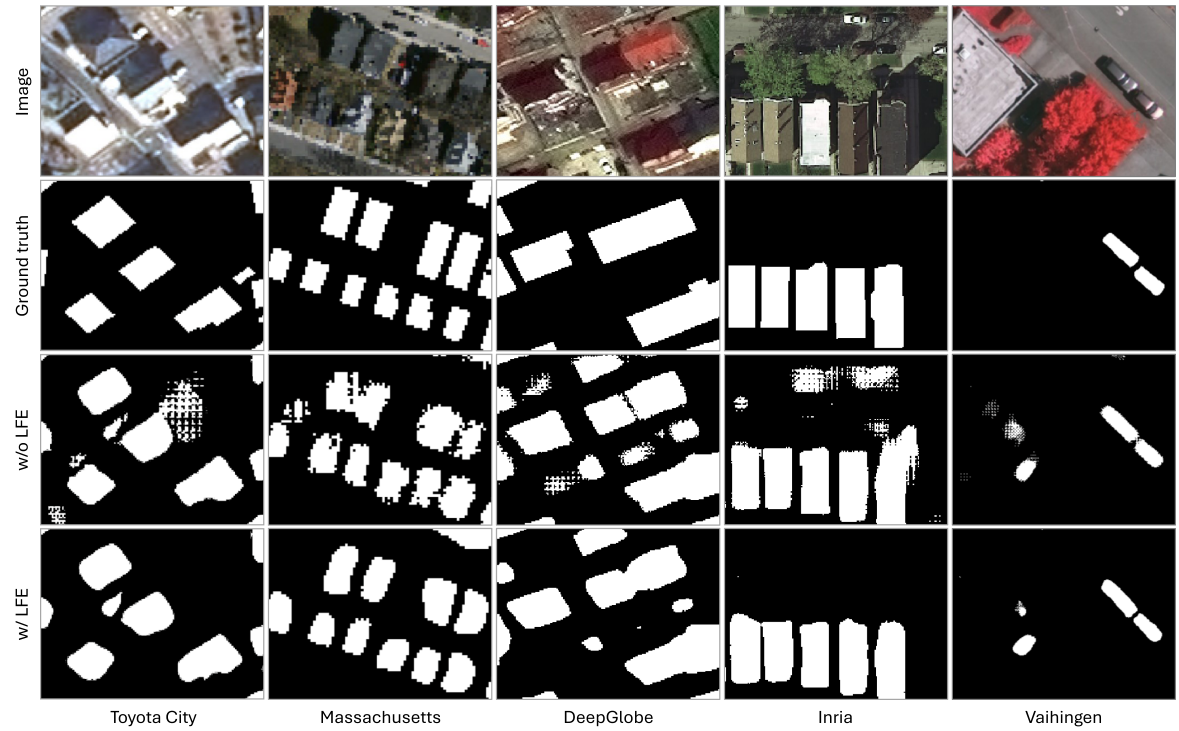

Fig. 12. Qualitative comparison of the prediction results with and without the LFE module. The results are shown for the building detection task (the 1st–4th row) and the vehicle extraction task (the last row). We can see grid-like noise without the LFE module (w/o LFE), but they disappear in the case with the LFE module (w/ LFE). The predictions of "w/o LFE" and "w/ LFE" are acquired from VGG-D-Keep and VGG-D-LFE.

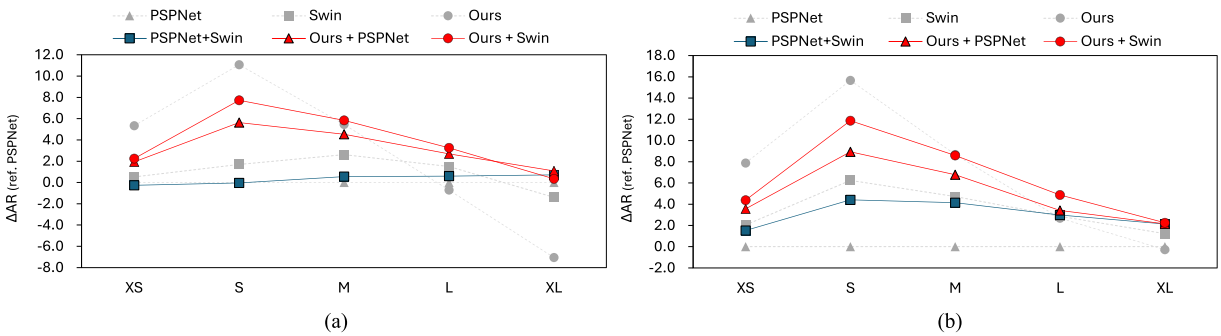

Fig. 13. Results of model ensembling. Each plot shows the performance difference from PSPNet (ΔAR) for each size of buildings. The red plots show the combination of the proposed and the previous model ("Ours+PSPNet" and "Ours+Swin"), the blue plot shows the ensemble of the two previous models (PSPNet+Swin), and the gray plots show the original three models. (a) DeepGlobe Building Detection Dataset. (b) Inria Aerial Image Labeling Dataset.

scheme performs well only if we do not use long-increasing architecture.

7) Analysis on ERF: To further analyze the effect of the LFE module, we visualized the ERF [12] of the models. As in [12], the ERF is acquired by averaging the gradient signals at the input layer that are back-propagated from the center unit in the output map. In Fig. 15, we compare the ERF for three models: the pooling-based model (VGG-P), the dilation-based model with and without the LFE module (VGG-D-Keep and VGG-D-LFE). To our surprise, grid patterns appear in the ERF when we use dilated convolutions without the LFE module (see VGG-D-Keep). The grid patterns mean that the units on the output are unaware of the local structures smaller than the grid size, as explained in Section III-B. More importantly, the grid patterns disappear when the LFE module is attached (see VGG-D-LFE), meaning that the LFE module successfully aggregates local information missed in the dilated backbone.

D. Vehicle Extraction

This section describes the results on the vehicle extraction task. Vehicles are one of the major targets for RS image analysis. It has a wide range of applications such as traffic monitoring and planning [75], defense, and trade area analysis. For such tasks, the model requires separately detecting small and often crowded vehicles on the roads and the parking lots.

Below, Section V-D1 evaluates the effect of the LFE module compared to the baselines, and Section V-D2 compares the

TABLE IV
COMPARISON TO THE STATE-OF-THE-ART MODELS ON BUILDING DETECTION TASK

	Pixel metrics		Instance metrics			AR for each size				
	F1	IoU	F1	AP_{vol}	AR	AR_{XS}	AR_S	AR_M	AR_L	AR_{XL}
Toyota City Dataset										
Sherrah [52]	62.0	46.8	42.4	21.3	17.0	0.2	15.1	30.8	34.0	32.5
U-Net [17]	62.7	**47.7**	47.8	24.9	21.6	0.8	21.9	36.8	34.5	27.5
FCN-8s [16]	56.2	38.3	16.7	9.8	4.2	0.0	2.1	8.4	21.5	17.5
Deeplab-LFOV [10]	61.7	45.0	29.1	15.4	7.3	0.0	4.7	14.4	24.8	24.4
Deeplab-V2 [32]	**62.8**	44.7	33.0	17.7	10.4	0.0	6.3	18.6	29.4	29.5
VGG-D-LFE	60.1	45.1	**51.5**	**27.7**	**25.8**	**1.5**	**28.1**	**42.0**	34.5	33.1
Massachusetts Buildings Dataset										
Mnih-CNN [66]	91.5	—	—	—	—	—	—	—	—	—
Saito-CNN-MA [69]	94.3	—	—	—	—	—	—	—	—	—
Sherrah [52]	93.0	63.3	66.1	43.0	26.5	21.9	29.3	33.6	37.8	38.2
U-Net [17]	94.1	67.6	74.0	49.3	32.9	27.0	36.9	38.2	45.0	52.9
FCN-8s [16]	93.1	61.0	46.4	29.5	12.3	10.6	16.3	27.4	39.2	45.4
Deeplab-LFOV [10]	89.7	53.6	17.4	13.0	4.4	1.5	4.2	18.7	28.5	34.3
Deeplab-V2 [32]	93.5	62.9	48.2	32.0	15.1	9.3	15.7	28.1	39.1	35.2
Deeplab-V3+ [19]	**95.4**	**70.3**	75.3	49.5	33.7	21.8	38.1	38.6	**48.6**	46.0
PSPNet [18]	94.8	66.1	51.0	31.6	18.1	9.9	19.3	32.4	47.8	**54.3**
VGG-D-LFE	93.4	64.1	75.1	**50.3**	35.0	**28.9**	39.7	38.2	43.1	42.1
ResNet-D-LFE	94.1	66.9	74.7	48.6	33.8	23.0	38.0	37.2	43.7	40.6
ConvNeXt-D-LFE	94.1	68.1	**76.9**	49.7	**36.1**	23.6	**41.2**	**39.2**	43.0	39.4
DeepGlobe Building Detection Dataset										
Sherrah [52]	79.2	65.2	52.9	35.5	25.7	0.3	2.6	20.2	36.0	43.7
U-Net [17]	84.6	72.5	67.4	48.1	38.4	2.1	13.0	36.8	49.6	55.3
FCN-8s [16]	80.4	65.8	55.3	36.9	27.0	1.2	6.8	23.2	36.0	44.5
Deeplab-LFOV [10]	78.7	60.0	46.9	29.6	19.1	0.1	0.8	12.0	28.0	40.2
Deeplab-V2 [32]	85.6	74.9	68.4	48.6	38.7	1.8	10.9	36.3	51.0	59.5
Deeplab-V3+ [19]	85.1	74.3	66.4	47.1	37.1	1.7	10.4	34.4	49.0	57.3
PSPNet [18]	86.3	75.9	70.0	50.6	41.5	2.7	14.5	39.1	54.1	**62.6**
SwinTransformer [73]	**87.1**	**77.3**	**71.9**	52.2	43.1	3.2	16.2	41.7	55.6	61.2
SegFormer-B5 [74]	86.9	77.0	71.0	51.9	43.3	3.9	17.5	40.8	**56.1**	62.3
DSAT-Net [26]	86.6	76.5	71.1	51.3	42.5	1.8	14.7	40.7	55.6	59.3
SDSC-Unet [27]	87.0	**77.3**	70.0	50.7	41.1	1.7	13.5	38.9	53.8	61.6
VGG-D-LFE	80.7	68.1	66.2	49.8	41.3	6.9	23.9	41.1	49.4	49.8
ResNet-D-LFE	83.4	71.9	69.9	**53.0**	**44.7**	**8.0**	**25.6**	**44.6**	53.4	55.6
ConvNeXt-D-LFE	82.2	70.2	67.4	50.3	41.9	6.2	22.6	40.3	51.3	52.9
Inria Aerial Image Labeling Dataset										
Sherrah [52]	79.2	66.7	62.6	41.5	32.0	0.2	7.5	27.3	44.5	46.1
U-Net [17]	84.9	76.3	73.6	53.2	46.2	3.0	18.2	40.3	59.9	63.7
FCN-8s [16]	82.4	72.2	65.8	44.7	35.8	1.2	9.8	29.1	50.2	58.4
Deeplab-LFOV [10]	77.5	50.7	38.3	23.8	14.7	0.0	1.2	10.0	24.0	37.1
Deeplab-V2 [32]	85.5	76.3	71.7	51.2	42.4	1.8	13.0	37.3	56.7	63.9
Deeplab-V3+ [19]	85.1	76.5	70.7	50.6	42.5	1.9	15.0	37.0	56.3	62.1
PSPNet [18]	86.1	77.7	73.5	53.3	45.5	2.5	17.1	40.3	59.8	65.3
SwinTransformer [73]	86.8	79.3	75.3	56.2	49.3	4.5	23.3	45.0	62.7	66.5
SegFormer-B5 [74]	**87.3**	**80.0**	75.6	56.5	49.7	4.2	22.2	45.1	**63.8**	**71.4**
DSAT-Net [26]	86.1	78.2	74.5	55.0	48.5	2.9	21.9	44.6	61.9	64.4
SDSC-Unet [27]	86.4	78.7	73.2	53.5	46.4	2.5	18.4	41.8	60.7	68.2
VGG-D-LFE	82.3	71.6	73.9	56.2	49.2	9.8	30.2	46.5	59.2	59.6
ResNet-D-LFE	84.7	75.5	**76.3**	**59.0**	**52.0**	**10.4**	**32.8**	**49.0**	62.5	65.0
ConvNeXt-D-LFE	84.6	75.3	74.7	57.7	50.8	8.9	30.7	47.5	61.7	63.3

The bold values represent the best results.

proposed model to the state-of-the-art semantic segmentation models.

1) Effect of the Dilated Backbone and the LFE Module: Table VII compares the proposed and baseline models. (Plain versus Dilated) The use of the dilated backbone consistently improves AP_{vol} of the plain baseline, showing the importance of high-resolution feature maps also for the vehicle extraction task. (Keep versus LFE) The LFE module improves the performance for most cases (two out of three). The last row of Fig. 12 shows the qualitative comparison of the predictions with and without the LFE module. While the model without the LFE module produces many false alarms due to the spatial inconsistency problem, they are successfully suppressed in the model with the LFE module. Moreover, the boundaries of the adjacent vehicles are more accurate with the LFE module.

2) Comparison to the State-of-the-Art Models: Table VIII compares the proposed models to the state-of-the-art segmentation models. We can see similar results as the building detection task. The proposed model outperforms the previous models on instance-level metrics. On the other hand, the previous methods, such as PSPNet, perform well on pixel-level metrics.

E. Road Extraction

This section describes the results on the road extraction task. Road extraction is a fundamental task in RS image analysis with various applications such as automatic map updates for

TABLE V
EXPLORATION OF ALTERNATIVE SOLUTIONS

	Backbone	Massachusetts	DeepGlobe	Inria
VGG	D	**49.9**	42.8	49.9
	ID	47.8	**43.2**	**50.0**
	DSP	-	42.1	**50.0**
ResNet	D	**46.8**	**46.3**	**52.6**
	ID	44.9	46.2	52.3
	DSP	-	45.9	50.9
ConvNeXt	D	45.9	48.3	54.5
	ID	**48.7**	**49.3**	**55.5**
	DSP	-	48.2	53.1
DeepLabV2	D	-	48.6	51.2
	ID	-	**49.9**	**52.4**

Table compares AP_{vol} of the dilated backbone (D), incrementally dilated (ID) backbone, and dilated spatial pyramid (DSP).
The bold values represent the best results.

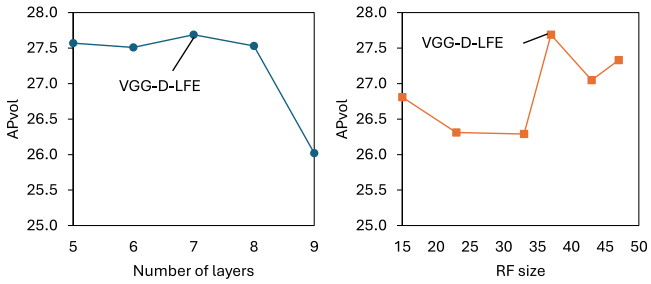

Fig. 14. Sensitivity analysis on the depth (left) and the RF size (right) of the LFE module.

TABLE VI
COMPARISON ON DIFFERENT SCHEMES OF DECREASING DILATION RATE IN LFE MODULE

Scheme	Dilation rate	AP_{vol}
LFE-Dec	4-4-4-2-2-1-1	27.7
LFE-Dec-Dec	4-4-2-1-4-2-1	27.5
LFE-Inc	1-1-2-2-4-4-4	27.1
LFE-Inc-Inc	1-2-4-4-1-2-4	27.8

The bold values represent the best results.

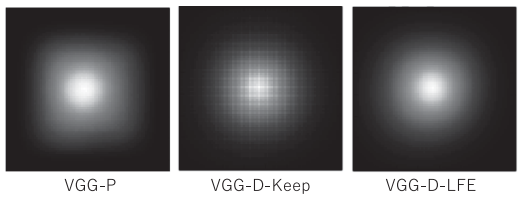

Fig. 15. ERF visualization results for VGG-P (left), VGG-D-Keep (mid), and VGG-D-LFE (right). Although the grid-like patterns appear in the ERF of VGG-D-Keep, they disappear in VGG-D-LFE thanks to the LFE module (mid versus right).

autonomous driving, damage assessment for disasters, and urban monitoring and planning. To accurately extract the long and thin structure of roads, the model requires special care on the resolution of the feature map and the receptive field size [76], similar to small object detection.

As an evaluation metric, the task puts more emphasis on the connectivity of roads than the pixel-level accuracy [77], [78], [79] because a small error on pixels can change the connectivity of roads resulting in completely wrong road networks. To evaluate the connectivity of roads, average path length similarity (APLS) [80] has been widely used. It measures the deviation of the shortest path length for every pair of nodes in the road graph, reflecting topological similarity between the predicted and the ground truth road networks. Following previous works, we focus on the APLS metric in this experiment.

1) Effect of the Dilated Backbone and the LFE Module: Table IX shows the comparison result between the proposed and the baseline models. (Plain versus Dilated) The use of the dilated backbone largely degrades the APLS of the plain model. The result contradicts the results of the previous sections, where the dilated backbone consistently improves the performance for small objects such as buildings and vehicles. A primary reason is that the spatial inconsistency caused by the dilated convolutions is more crucial for the road extraction task. We can see the impact of the problem in Fig. 16. The predicted roads are disconnected around the crossings due to the spatial inconsistency problem. (Keep versus LFE) Using the LFE module effectively solves the spatial inconsistency problem, significantly improving the APLS. As shown in Fig. 16, the disconnected roads are successfully connected with the LFE module.

2) Comparison to the State-of-the-Art Models: Table X shows the comparison result to the state-of-the-art segmentation models. The proposed models perform worse on both the pixel-level metrics and the APLS. One reason is that the proposed methods utilize shallow architecture (i.e., until stage3 of the base networks), and hence, the receptive field size is not sufficient for recognizing the long and thin structure of the roads [76]. Based on the observation, we extend ResNet-D-LFE by using deeper architecture, i.e., layers until stage4 of ResNet18, as the base network. As shown in the table, using the deeper architecture significantly improves the performance. However, the previous models still perform much better than the proposed model. The possible reason is that the receptive field size still needs to be increased because the previous model uses a much deeper architecture, such as ResNet101, as the backbone.

VI. DISCUSSION

A. Effectiveness of the Proposed LFE Module

To accurately segment small objects, not only high-resolution features but also large receptive fields are crucial. We confirm this by the results in Fig. 11, where the performance for small buildings is degraded by decreasing the receptive field size. This result motivates us to utilize dilated convolutions to extract high-resolution feature maps with large receptive fields. In previous works, dilated backbones commonly have increasing dilation rates [18], [21], [32], [37], [38]. However, as shown in Fig. 12, we find that the "increasing" architecture causes grid-like artifacts in the prediction results. The proposed LFE aims to solve this problem by introducing a novel scheme of "increasing-decreasing" dilation rates. The effect of the LFE module is evident from Figs. 12 and 16 and Tables III, VII, and IX, where grid-like artifacts successfully disappeared from the results, and the overall segmentation performance is significantly improved with the LFE module across different datasets and tasks.

TABLE VII
EFFECT OF THE LFE MODULE ON THE VEHICLE EXTRACTION TASK

	AP_{vol}	AR	AR_{XS}	AR_S	AR_M	AR_L	AR_{XL}
VGG-P	54.6	46.9	26.7	40.0	47.8	36.4	0.0
VGG-D	61.9	53.5	0.0	30.8	54.7	50.9	0.0
VGG-D-Keep	65.1	**57.5**	**40.0**	**43.3**	**58.9**	43.6	0.0
VGG-D-LFE	**65.7**	56.5	13.3	31.7	58.0	**56.4**	0.0
ResNet-P	59.2	45.2	0.0	23.3	46.6	34.6	0.0
ResNet-D	62.0	53.9	**40.0**	23.3	55.7	40.0	0.0
ResNet-D-Keep	53.7	**56.6**	**40.0**	**26.7**	**58.3**	**45.5**	0.0
ResNet-D-LFE	**62.4**	53.6	**40.0**	25.0	55.2	**45.5**	0.0
ConvNeXt-P	61.3	53.0	**40.0**	21.7	54.3	52.7	0.0
ConvNeXt-D	66.6	56.3	0.0	**35.0**	57.4	**56.4**	0.0
ConvNeXt-D-Keep	**67.9**	**59.6**	0.0	33.3	**61.2**	49.1	0.0
ConvNeXt-D-LFE	65.5	58.2	0.0	31.7	59.7	52.7	0.0

For ResNet- and ConvNeXt-based models, we use the same learning rate settings as the VGG-based models (initial learning rate of 1.0×10^{-5} and increased learning rate at scratch layers).
The bold values represent the best results.

TABLE VIII
COMPARISON TO THE STATE-OF-THE-ART MODELS ON VAIHINGEN DATASET

	Pixel metrics		Instance metrics			AR for each size				
	F1	IoU	F1	AP_{vol}	AR	AR_{XS}	AR_S	AR_M	AR_L	AR_{XL}
Sherrah [52]	77.7	58.9	67.3	48.0	33.4	0.0	18.7	34.5	25.5	0.0
U-Net [17]	83.5	70.3	79.7	62.6	52.0	0.0	30.0	53.9	47.3	0.0
FCN-8s [16]	81.0	67.1	63.6	47.6	32.5	0.0	21.4	33.7	27.3	0.0
DeepLab-LFOV [10]	84.9	73.0	72.1	54.8	43.6	0.0	23.3	39.2	43.6	0.0
DeepLab-V2 [32]	91.6	82.1	82.9	60.9	45.4	0.0	16.9	46.9	41.8	0.0
DeepLab-V3+ [19]	79.8	66.3	76.2	54.0	37.5	0.0	6.7	39.3	25.5	0.0
PSPNet [18]	**93.8**	**87.5**	**85.0**	64.9	51.6	0.0	**40.0**	52.6	41.8	0.0
VGG-D-LFE	77.9	40.7	80.5	**65.7**	56.5	13.3	31.7	58.0	**56.4**	0.0
ResNet-D-LFE	70.9	55.0	78.9	62.4	53.6	**40.0**	25.0	55.2	45.5	0.0
ConvNeXt-D-LFE	70.1	55.1	81.3	65.5	**58.2**	0.0	31.7	**59.7**	52.7	0.0

The bold values represent the best results.

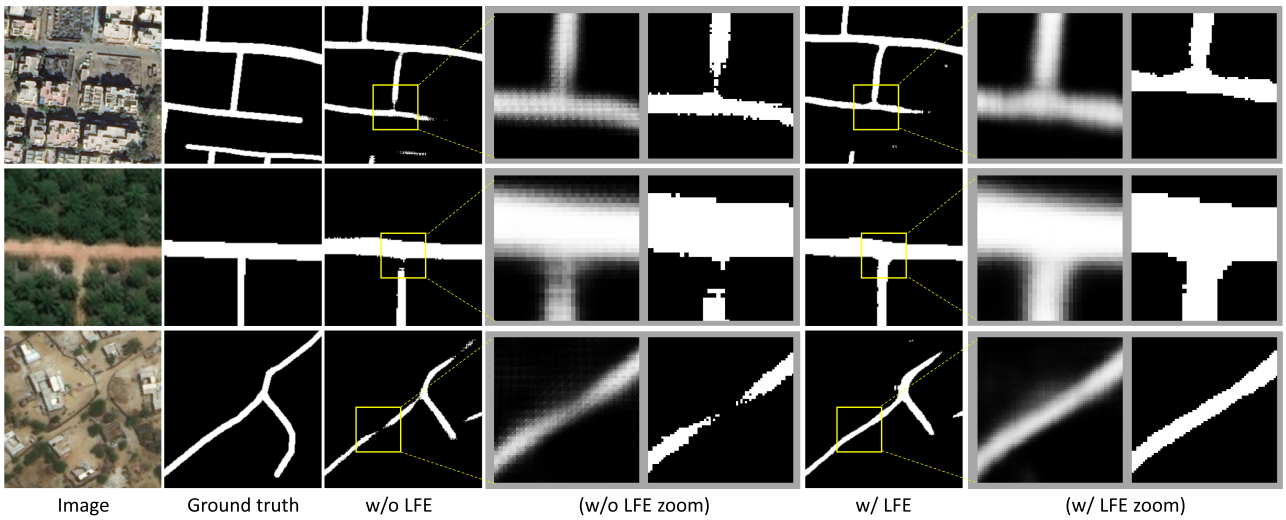

Fig. 16. Qualitative comparison of the prediction results with and without the LFE module on road extraction task. The region inside the yellow box is enlarged on the right, where the output probability and the predicted mask of the region are shown. The predictions of "w/o LFE" and "w/ LFE" are acquired from ResNet-D-Keep and ResNet-D-LFE, respectively.

B. Advantage Against Previous Methods

The main advantage of our model is its significantly enhanced performance on small objects. Although the previous methods perform well on pixel-level metrics in building detection, Table IV shows that they perform poorly on small buildings when evaluated on the instance-level metrics. On the other hand, our LFE module enables more aggressive use of dilated convolutions, i.e., replacing all the downsampling layers with the dilated convolutions to keep full resolution throughout the network. As a result, our model shows significantly enhanced performance on

TABLE IX
EFFECT OF THE LFE MODULE ON THE ROAD EXTRACTION TASK

	Acc.	IoU	APLS
VGG-P	97.2	50.2	0.358
VGG-D	97.1	49.1	0.254
VGG-D-Keep	**97.6**	**56.3**	0.351
VGG-D-LFE	**97.6**	55.8	**0.459**
ResNet-P	97.6	54.9	0.463
ResNet-D	97.3	51.7	0.290
ResNet-D-Keep	**97.8**	57.4	0.380
ResNet-D-LFE	97.7	**57.7**	**0.480**
ConvNeXt-P	97.6	55.5	0.459
ConvNeXt-D	97.5	54.8	0.366
ConvNeXt-D-Keep	**97.7**	**56.8**	0.402
ConvNeXt-D-LFE	97.6	56.2	**0.477**

The bold values represent the best results.

TABLE X
COMPARISON TO THE STATE-OF-THE-ART MODELS ON THE VALIDATION SET OF DEEPGLOBE ROAD EXTRACTION DATASET

	Acc.	IoU	APLS
Sherrah [52]	96.8	46.0	0.310
U-Net [17]	**98.1**	63.3	0.614
FCN-8s [16]	97.8	58.6	0.491
DeepLab-V2 [32]	97.9	61.0	0.578
DeepLab-V3+ [19]	**98.1**	**63.8**	**0.642**
PSPNet [18]	**98.1**	**63.8**	0.624
VGG-D-LFE	97.6	55.8	0.459
ResNet-D-LFE	97.8	59.3	0.521
ConvNeXt-D-LFE	97.7	57.9	0.494
ResNet-D-LFE (stage4)	97.9	60.0	0.557

The bold values represent the best results.

small buildings at the expense of little performance drop on large buildings. Moreover, thanks to the performance enhancement on small buildings, our model shows better overall performance than the previous models (see Tables IV and VIII). Still, our model performs worse than the previous methods in road extraction (see Table X). One reason is the shallow architecture of our model. The receptive field size of our shallow model might be insufficient for road extraction that requires a large receptive field [76].

The other advantage of our model is that it works complementary to the existing models in terms of object size. Because our model and existing models are good at small and large objects, respectively, the ensemble of both models can effectively enhance the performance of all kinds of objects (see Fig. 13).

C. Limitations and Future Work

A limitation of the proposed method is its high memory consumption. Since high-resolution feature maps are memory intensive, keeping full resolution throughout the network requires a large memory footprint during training and inference, which limits the application of the method for deeper architectures such as ResNet101. Architectural improvements might address the limitation, e.g., finding narrow and deeper architecture with architecture search or using memory efficient convolution methods such as depthwise separable convolutions. We leave these to our future work.

Our experiments focus on foreground object segmentation tasks, but the proposed method should also be effective for multiclass segmentation. Moreover, our method should be effective for tasks other than segmentation (e.g., object detection and change detection). We leave the extension to the different settings or tasks for our future work.

Another future work is an end-to-end model combining the conventional downsampling-based and the proposed dilation-based models. A feature fusion approach might be practical, in which the features from the downsampling and dilation branches are fused at intermediate layers. We expect that such a model will better exploit the synergetic effects of both approaches, achieving highly robust performance for objects of any size.

VII. CONCLUSION

This article presents a novel network architecture for segmenting small objects in RS imagery. Unlike previous approaches, the proposed architecture has no downsampling layers and uses dilated convolutions instead. We revealed that such aggressive use of dilated convolutions causes problems in local feature propagation and proposed a novel "increasing-decreasing" dilation rates to address the problems. We have evaluated the proposed method on six datasets across three tasks: building detection, vehicle detection, and road extraction, where the method has shown remarkable performance for small objects. The proposed method will apply to a wide range of remote sensing applications that require the detection of individual small objects, such as population estimation, automatic map creation, and urban monitoring.

ACKNOWLEDGMENT

The authors would like to thank Tomoyuki Imaizumi and Shuhei Hikosaka for their support and advice on this work.

REFERENCES

[1] I. Demir et al., "DeepGlobe 2018: A challenge to parse the earth through satellite images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2018, pp. 172–181, doi: 10.1109/CVPRW.2018.00031.
[2] N. Audebert, B. L. Saux, and S. Lefèvre, "Semantic segmentation of Earth observation data using multimodal and multi-scale deep networks," in *Proc. Asian Conf. Comput. Vis.*, 2016, pp. 180–196, doi: 10.1007/978-3-319-54181-5_12.
[3] L. Li, J. Liang, M. Weng, and H. Zhu, "A multiple-feature reuse network to extract buildings from remote sensing imagery," *Remote Sens.*, vol. 10, no. 9, 2018, Art. no. 1350, doi: 10.3390/RS10091350.
[4] H. Yang, P. Wu, X. Yao, Y. Wu, B. Wang, and Y. Xu, "Building extraction in very high resolution imagery by dense-attention networks," *Remote Sens.*, vol. 10, no. 11, 2018, Art. no. 1768, doi: 10.3390/RS10111768.
[5] M. Guo, H. Liu, Y. Xu, and Y. Huang, "Building extraction based on U-net with an attention block and multiple losses," *Remote Sens.*, vol. 12, no. 9, 2020, Art. no. 1400, doi: 10.3390/RS12091400.
[6] M. Yu, X. Chen, W. Zhang, and Y. Liu, "AGs-Unet: Building extraction model for high resolution remote sensing images based on attention gates U network," *Sensors*, vol. 22, no. 8, 2022, Art. no. 2932, doi: 10.3390/s22082932.
[7] H. Wang and F. Miao, "Building extraction from remote sensing images using deep residual U-net," *Eur. J. Remote Sens.*, vol. 55, no. 1, pp. 71–85, 2022, doi: 10.1080/22797254.2021.2018944.
[8] F. Chen, N. Wang, B. Yu, and L. Wang, "Res2-Unet, a new deep architecture for building detection from high spatial resolution images," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 15, pp. 1494–1501, 2022, doi: 10.1109/JSTARS.2022.3146430.
[9] P. Hu and D. Ramanan, "Finding tiny faces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1522–1530, doi: 10.1109/CVPR.2017.166.

[10] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.

[11] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 1–13.

[12] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 4905–4913. [Online]. Available: https://proceedings.neurips.cc/paper/2016/hash/c8067ad1937f728f51288b3eb986afaa-Abstract.html

[13] R. Hamaguchi, A. Fujita, K. Nemoto, T. Imaizumi, and S. Hikosaka, "Effective use of dilated convolutions for segmenting small object instances in remote sensing imagery," in *Proc. Winter Conf. Appl. Comput. Vis.*, 2018, pp. 1442–1450, doi: 10.1109/WACV.2018.00162.

[14] 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, doi: 10.1109/CVPR.2016.90.

[15] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11976–11986, doi: 10.1109/CVPR52688.2022.01167.

[16] 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, doi: 10.1109/CVPR.2015.7298965.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[18] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6230–6239, doi: 10.1109/CVPR.2017.660.

[19] C. L.-Chieh, Z. Yukun, P. George, S. Florian, and A. Hartwig, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 833–851, doi: 10.1007/978-3-030-01234-2_49.

[20] J. Wang et al., "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021, doi: 10.1109/TPAMI.2020.2983686.

[21] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, "OCNet: Object context for semantic segmentation," *Int. J. Comput. Vis.*, vol. 129, no. 8, pp. 2375–2398, 2021, doi: 10.1007/S11263-021-01465-9.

[22] M. Chen, T. Mao, J. Wu, R. Du, B. Zhao, and L. Zhou, "SAU-net: A novel network for building extraction from high-resolution remote sensing images by reconstructing fine-grained semantic features," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 17, pp. 6747–6761, 2024, doi: 10.1109/JSTARS.2024.3371427.

[23] Y. Liu et al., "ARC-Net: An efficient network for building extraction from high-resolution aerial images," *IEEE Access*, vol. 8, pp. 154997–155010, 2020, doi: 10.1109/ACCESS.2020.3015701.

[24] D. Zhou et al., "Robust building extraction for high spatial resolution remote sensing images with self-attention network," *Sensors*, vol. 20, no. 24, 2020, Art. no. 7241, doi: 10.3390/S20247241.

[25] K. Chen, Z. Zou, and Z. Shi, "Building extraction from remote sensing images with sparse token transformers," *Remote Sens.*, vol. 13, no. 21, 2021, Art. no. 4441, doi: 10.3390/rs13214441.

[26] R. Zhang, Z. Wan, Q. Zhang, and G. Zhang, "DSAT-Net: Dual spatial attention transformer for building extraction from aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 6008405, doi: 10.1109/LGRS.2023.3304377.

[27] R. Zhang, Q. Zhang, and G. Zhang, "SDSC-UNet: Dual skip connection ViT-based U-shaped model for building extraction," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 6005005, doi: 10.1109/LGRS.2023.3270303.

[28] H. Zhang, H. Dou, Z. Miao, N. Zheng, M. Hao, and W. Shi, "Extracting building footprint from remote sensing images by an enhanced vision transformer network," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5406814, doi: 10.1109/TGRS.2024.3421651.

[29] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5168–5177, doi: 10.1109/CVPR.2017.549.

[30] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1520–1528, doi: 10.1109/ICCV.2015.178.

[31] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[32] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018, doi: 10.1109/TPAMI.2017.2699184.

[33] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 483–499, doi: 10.1007/978-3-319-46484-8_29.

[34] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 636–644, doi: 10.1109/CVPR.2017.75.

[35] P. Wang et al., "Understanding convolution for semantic segmentation," in *Proc. Winter Conf. Appl. Comput. Vis.*, 2018, pp. 1451–1460, doi: 10.1109/WACV.2018.00163.

[36] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "A^2-Nets: Double attention networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 350–359. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/e165421110ba03099a1c0393373c5b43-Paper.pdf

[37] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 173–190, doi: 10.1007/978-3-030-58539-6_11.

[38] F. Zhang et al., "ACFNet: Attentional class feature network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6797–6806, doi: 10.1109/ICCV.2019.00690.

[39] H. Zhang, H. Zhang, C. Wang, and J. Xie, "Co-occurrent features in semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 548–557, doi: 10.1109/CVPR.2019.00064.

[40] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Convolutional neural networks for large-scale remote-sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 645–657, Feb. 2017, doi: 10.1109/TGRS.2016.2612821.

[41] 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. Workshop*, 2016, pp. 680–688, doi: 10.1109/CVPRW.2016.90.

[42] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "High-resolution aerial image labeling with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7092–7103, Dec. 2017, doi: 10.1109/TGRS.2017.2740362.

[43] W. Li, K. Sun, H. Zhao, W. Li, J. Wei, and S. Gao, "Extracting buildings from high-resolution remote sensing images by deep ConvNets equipped with structural-cue-guided feature alignment," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 113, 2022, Art. no. 102970, doi: 10.1016/j.jag.2022.102970.

[44] J. Hosang, R. Benenson, P. Dollar, and B. Schiele, "What makes for effective detection proposals?," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 814–830, Apr. 2016, doi: 10.1109/TPAMI.2015.2465908.

[45] Q. Zhu, C. Liao, H. Hu, X. Mei, and H. Li, "MAP-Net: Multiple attending path neural network for building footprint extraction from remote sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 6169–6181, Jul. 2021, doi: 10.1109/TGRS.2020.3026051.

[46] G. Yan, H. Jing, H. Li, H. Guo, and S. He, "Enhancing building segmentation in remote sensing images: Advanced multi-scale boundary refinement with MBR-HRNet," *Remote Sens.*, vol. 15, no. 15, 2023, Art. no. 3766, doi: 10.3390/RS15153766.

[47] S. Seong and J. Choi, "Semantic segmentation of urban buildings using a high-resolution network (HRNet) with channel and spatial attention gates," *Remote Sens.*, vol. 13, no. 16, 2021, Art. no. 3087, doi: 10.3390/RS13163087.

[48] Z. Li et al., "HCRB-MSAN: Horizontally connected residual blocks-based multiscale attention network for semantic segmentation of buildings in HSR remote sensing images," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 15, pp. 5534–5544, 2022, doi: 10.1109/JSTARS.2022.3188515.

[49] Y. Zhao, G. Sun, L. Zhang, A. Zhang, X. Jia, and Z. Han, "MSRF-net: Multiscale receptive field network for building detection from remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5515714, doi: 10.1109/TGRS.2023.3282926.

[50] H. Wang, M. Zhang, W. Li, Y. Gao, Y. Gui, and Y. Zhang, "Unbalanced class learning network with scale-adaptive perception for complicated scene in remote sensing images segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 4406712, doi: 10.1109/TGRS.2024.3388528.

[51] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–21. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[52] J. Sherrah, "Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery," 2016, *arXiv:1606.02585*. [Online]. Available: http://arxiv.org/abs/1606.02585

[53] J. Yuan, "Learning building extraction in aerial scenes with convolutional networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 11, pp. 2793–2798, Nov. 2018, doi: 10.1109/TPAMI.2017.2750680.

[54] V. Iglovikov, S. Seferbekov, A. Buslaev, and A. Shvets, "TernausNetV2: Fully convolutional network for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2018, pp. 2280–2284, doi: 10.1109/CVPRW.2018.00042.

[55] L. Mou and X. X. Zhu, "Vehicle instance segmentation from aerial image and video using a multitask learning residual fully convolutional network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6699–6711, Nov. 2018, doi: 10.1109/TGRS.2018.2841808.

[56] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2011, pp. 109–117. [Online]. Available: https://proceedings.neurips.cc/paper/2011/hash/beda24c1e1b46055dff2c39c98fd6fc1-Abstract.html

[57] S. Shrestha and L. Vanneschi, "Improved fully convolutional network with conditional random fields for building extraction," *Remote Sens.*, vol. 10, no. 7, 2018, Art. no. 1135, doi: 10.3390/RS10071135.

[58] 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, doi: 10.3390/RS10010144.

[59] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 1–14, doi: 10.1007/978-3-642-15549-9_1.

[60] S. Chen, Y. Ogawa, C. Zhao, and Y. Sekimoto, "Large-scale individual building extraction from open-source satellite imagery via super-resolution-based instance segmentation approach," *ISPRS J. Photogramm. Remote Sens.*, vol. 195, pp. 129–152, 2023, doi: 10.1016/j.isprsjprs.2022.11.006.

[61] H. Ji, Z. Gao, T. Mei, and B. Ramesh, "Vehicle detection in remote sensing images leveraging on simultaneous super-resolution," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 676–680, Apr. 2020, doi: 10.1109/LGRS.2019.2930308.

[62] J. Shermeyer and A. V. Etten, "The effects of super-resolution on object detection performance in satellite imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2019, pp. 1432–1441, doi: 10.1109/CVPRW.2019.00184.

[63] J. Rabbi, N. Ray, M. Schubert, S. Chowdhury, and D. Chao, "Small-object detection in remote sensing images with end-to-end edge-enhanced GAN and object detector network," *Remote Sens.*, vol. 12, no. 9, 2020, Art. no. 1432, doi: 10.3390/RS12091432.

[64] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.

[65] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 297–312, doi: 10.1007/978-3-319-10584-0_20.

[66] V. Mnih, "Machine learning for aerial image labeling," *Ph.D. dissertation*, Graduate Dept. of Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2013. [Online]. Available: http://hdl.handle.net/1807/35911

[67] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? The INRIA aerial image labeling benchmark," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2017, pp. 3226–3229, doi: 10.1109/IGARSS.2017.8127684.

[68] M. Cramer, "The DGPF-Test on digital airborne camera evaluation overview and test design," *PFG Photogrammetrie Fernerkundung Geoinf.*, no. 2, pp. 73–82, 2010, doi: 10.1127/1432-8364/2010/0041.

[69] S. Saito, T. Yamashita, and Y. Aoki, "Multiple object extraction from aerial imagery with convolutional neural networks," *J. Imag. Sci. Techn.*, vol. 60, no. 1, 2016, Art. no. 10402, doi: 10.2352/J.ImagingSci.Technol.2016.60.1.01040.

[70] S. Paisitkriangkrai, J. Sherrah, P. Janney, and A. V.-D. Hengel, "Effective semantic pixel labelling with convolutional networks and conditional random fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2015, pp. 36–43, doi: 10.1109/CVPRW.2015.7301381.

[71] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256. [Online]. Available: https://proceedings.mlr.press/v9/glorot10a.html

[72] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–13. [Online]. Available: http://arxiv.org/abs/1412.6980

[73] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10012–10022, doi: 10.1109/ICCV48922.2021.00986.

[74] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Álvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 12077–12090. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/64f1f27bf1b4ec22924fd0acb550c235-Abstract.html

[75] J. Leitloff, D. Rosenbaum, F. Kurz, O. Meynberg, and P. Reinartz, "An operational system for estimating road traffic information from aerial images," *Remote Sens.*, vol. 6, no. 11, pp. 11315–11341, 2014, doi: 10.3390/RS61111315.

[76] L. Zhou, C. Zhang, and M. Wu, "D-LinkNet: LinkNet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2018, pp. 182–186, doi: 10.1109/CVPRW.2018.00034.

[77] G. Máttyus, W. Luo, and R. Urtasun, "DeepRoadMapper: Extracting road topology from aerial images," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3438–3446, doi: 10.1109/ICCV.2017.372.

[78] A. Mosinska, P. M.-Neila, M. Koziński, and P. Fua, "Beyond the pixel-wise loss for topology-aware delineation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3136–3145, doi: 10.1109/CVPR.2018.00331.

[79] A. Batra, S. Singh, G. Pang, S. Basu, C. Jawahar, and M. Paluri, "Improved road connectivity by joint learning of orientation and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10385–10393, doi: 10.1109/CVPR.2019.01063.

[80] A. V. Etten, D. Lindenbaum, and T. M. Bacastow, "SpaceNet: A remote sensing dataset and challenge series," 2018, *arXiv:1807.01232*. [Online]. Available: http://arxiv.org/abs/1807.01232

Ryuhei Hamaguchi received the B.S. and M.Eng. degrees in aeronautics and astronautics from the Department of Aeronautics and Astronautics, the University of Tokyo, Tokyo, Japan, in 2011 and 2013, respectively.

He is currently a Technical Advisor with the Satellite Business Division, Pasco Corporation, Tokyo, and a Project Researcher with the National Institute of Advanced Industrial Science and Technology (AIST), Tokyo. His research interests include deep learning, scene understanding, change detection, and remote sensing image analysis.

Aito Fujita received the B.S. and M.Agr. degrees in agricultural engineering from the Department of Agricultural Engineering, Kobe University, Kobe, Japan, in 2011 and 2013, respectively.

From 2013 to 2018, he was a Research Engineer with the Satellite Business Division, Pasco Corporation. He is currently an Engineer with Synspective Inc., Tokyo, Japan, for SAR data analysis. His research interests include SAR data analysis, measure-theoretic statistics, geostatistics, and data assimilation.

Keisuke Nemoto received the B.S. degree in biomolecular functional engineering from the Department of Biomolecular Functional Engineering, Ibaraki University, Ibaraki, Japan, in 2012, and the M.S. degree in molecular and material sciences from the Department of Molecular and Material Sciences, Tokyo Metropolitan University, Tokyo, Japan, in 2014.

From 2014 to 2016, he was an Engineer with Simulation Corporation. From 2016 to 2018, he was a Research Engineer with the Satellite Business Division, Pasco Corporation. He is currently an Engineer with Synspective Inc., Tokyo, for SAR data analysis. His research interests include deep learning, change detection, and SAR data analysis.