ISDNet: Integrating Shallow and Deep Networks for Efficient Ultra-high Resolution Segmentation

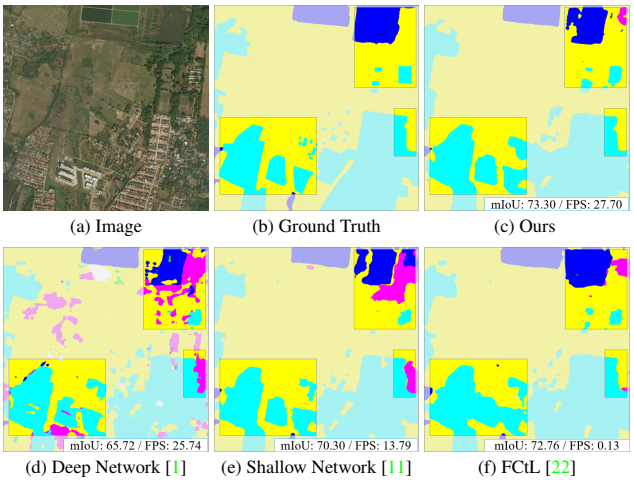
Shaohua Guo[1*], Liang Liu[2*], Zhenye Gan[2], Yabiao Wang[2], Wuhao Zhang[2],
Chengjie Wang[2], Guannan Jiang [5], Wei Zhang[5], Ran Yi[1†], Lizhuang Ma[1,3†], Ke Xu [4]
[1]Shanghai Jiao Tong University [2]Youtu Lab, Tencent
[3]East China Normal University [4]City University of Hong Kong [5]CATL
{guoshaohua, ranyi}@sjtu.edu.cn;{jianggn,zhangwei}@catl.com; ma-lz@cs.sjtu.edu.cn;
{leoneliu, winggzygan,caseywang, wuhaozhang,jasoncjwang}@tencent.com; kkangwing@gmail.com;

Abstract

The huge burden of computation and memory are two obstacles in ultra-high resolution image segmentation. To tackle these issues, most of the previous works follow the global-local refinement pipeline, which pays more attention to the memory consumption but neglects the inference speed. In comparison to the pipeline that partitions the large image into small local regions, we focus on inferring the whole image directly. In this paper, we propose ISDNet, a novel ultra-high resolution segmentation framework that integrates the shallow and deep networks in a new manner, which significantly accelerates the inference speed while achieving accurate segmentation. To further exploit the relationship between the shallow and deep features, we propose a novel Relational-Aware feature Fusion module, which ensures high performance and robustness of our framework. Extensive experiments on Deepglobe, Inria Aerial, and Cityscapes datasets demonstrate our performance is consistently superior to state-of-the-arts. Specifically, it achieves 73.30 mIoU with a speed of 27.70 FPS on Deepglobe, which is more accurate and 172 × faster than the recent competitor. Code available at https://github.com/cedricgsh/ISDNet.

1. Introduction

Semantic segmentation is a basic task that has been studied for decades. Unlike other vision tasks, such as image classification, segmentation needs to deal with small objects and fine boundaries that heavily rely on large-scale input images [1, 9, 16, 17, 24, 30, 40]. Especially, ultra-high resolution image with millions or even billions of pixels plays a vital role in the fields of remote sensing [34, 41, 42], autonomous driving [13, 25, 27], medical imagery applications [15, 19, 28].

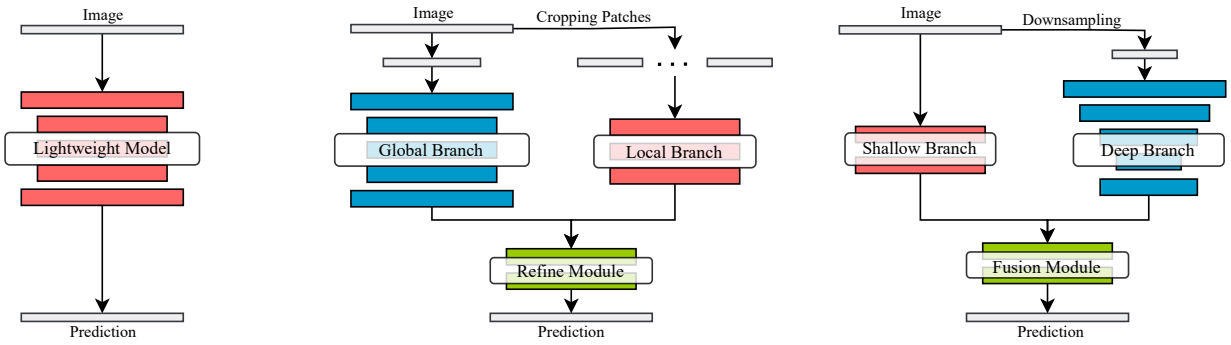


(a) Image (b) Ground Truth (c) Ours
(d) Deep Network [1] (e) Shallow Network [11] (f) FCtL [22]

Figure 1. Comparison of different model predictions on the ultra-high resolution image. (a) Input image. (b) The corresponding ground truth. (c) Prediction of our efficient segmentation method. (d) Prediction of a deep network input with the downsampled image. (e) Prediction of a shallow network input with the full-scale input. (f)Prediction of the latest ultra-high resolution segmentation method. Our method outperforms them in both speed and accuracy.

However, due to memory and computational limitations, general segmentation methods cannot well handle ultra-high resolution images input. Existing segmentation methods mainly focus on designing a neural network architecture for regular resolution images, but overlook the feasibility of larger scale input. As shown in Figure 1 (d), a deep and complicated model [1] needs to downsample the input image to meet the memory and speed requirements, but some detailed information is discarded during downsampling, resulting in poor performance. Although a shallow and lightweight model [11] can be adapted to process

*This work was done when S. Guo was an intern in Tencent Youtu Lab. S. Guo and L. Liu have equal contribution. † Corresponding Authors.



(a) Real-time Model Design (b) Global Local Refinement (c) Shallow and Deep Integration

Figure 2. Comparison for the schemes for ultra-high resolution image segmentation. (a) Design a lightweight model architecture to fit large scale images. (b) Global inference with multiple local patches refinement. (c) Our method by integrating the shallow and deep networks for the input of entire and downsampled image.

larger scale input as shown in Figure 1 (e), the performance is poor since it is hard to capture long-range and high-level semantic cues using a simple architecture.

Recently, some methods specially designed for ultra-high resolution segmentation tasks have been proposed [3, 4, 18, 22, 33]. These methods mainly follow the principle of global and local refinement. First, the entire image is input into the global network, and then the uncertain regions are refined through the local network multiple times. Although these methods require a lower memory consumption and reach higher accuracy in general, their inference speed is very slow. For example, Figure 1 (f) shows a recent method FCtL [22] that requires $\sim$8s to infer a image with 2448×2448 resolution and $\sim$26s for 5000×5000 resolution, which is intolerable in most applications.

To address the above limitations, we aim at achieving a better balance among accuracy, memory, and inference speed for ultra-high resolution segmentation. Instead of the scheme of global and local refinement, we propose ISDNet, a novel framework that infers the segmentation for ultra-high resolution inputs end-to-end. Inspired by the bilateral architecture [36, 37] widely used in lightweight segmentation model design, we proposed a framework to integrate shallow and deep networks for efficient segmentation. Different from the typical bilateral models which combine a shallow and a deep branch for the same input to model the spatial and context features respectively, we propose to input a different scale of input for shallow and deep branches. Besides, we empirically find that inputting heterogeneous information for shallow and deep branches and constructing an auxiliary learning task for another domain (e.g., super-resolution) can further help the training of our method.

For intuitive comparison, the prototype of three schemes for ultra-high resolution segmentation are shown in Figure 2. In summary, the contributions of this paper include:

- We propose a novel framework to integrate shallow and deep networks for efficient ultra-high resolution image segmentation. Besides, we empirically observe that heterogeneous inputs can improve accuracy.
- We present a Relation-Aware feature Fusion (RAF) module , which fuses features from shallow and deep branches based on their relationship along with auxiliary super-resolution and structure distillation losses to enhance the features learned from the deep branch.
- Extensive experiments show that our method achieves remarkable results on Deepglobe [7], Inria Aerial [26] and Cityscapes [6] datasets, while attaining both fast speed and low memory consumption in inference.

It is worth noting that our shallow and deep integration is a general framework, focusing particularly on efficient large scale segmentation, which can be leveraged to combine lots of general semantic segmentation networks including recent transformer based methods *e.g*. SegFormer [35].

2. Related works

2.1. Ultra High-resolution Segmentation

GLNet [3] proposes a collaborative global-local framework that combines the context of the global branch and details of the local branch to improve the segmentation results. Based on the GLNet, PPN [33] presents a classification branch to select important local patches to fuse with global images. Besides, CascadePSP [4] employs a general cascade structure to refine the coarse segmentation map both globally and locally. Similarly, MagNet [18] introduces a novel multi-scale architecture. The output rough result will be progressively refined from the coarsest to the finest scale. Recently, FCtL [22] leverages the locality-aware contextual correlation and the adaptive feature fusion scheme, which associates and combines local-context information to

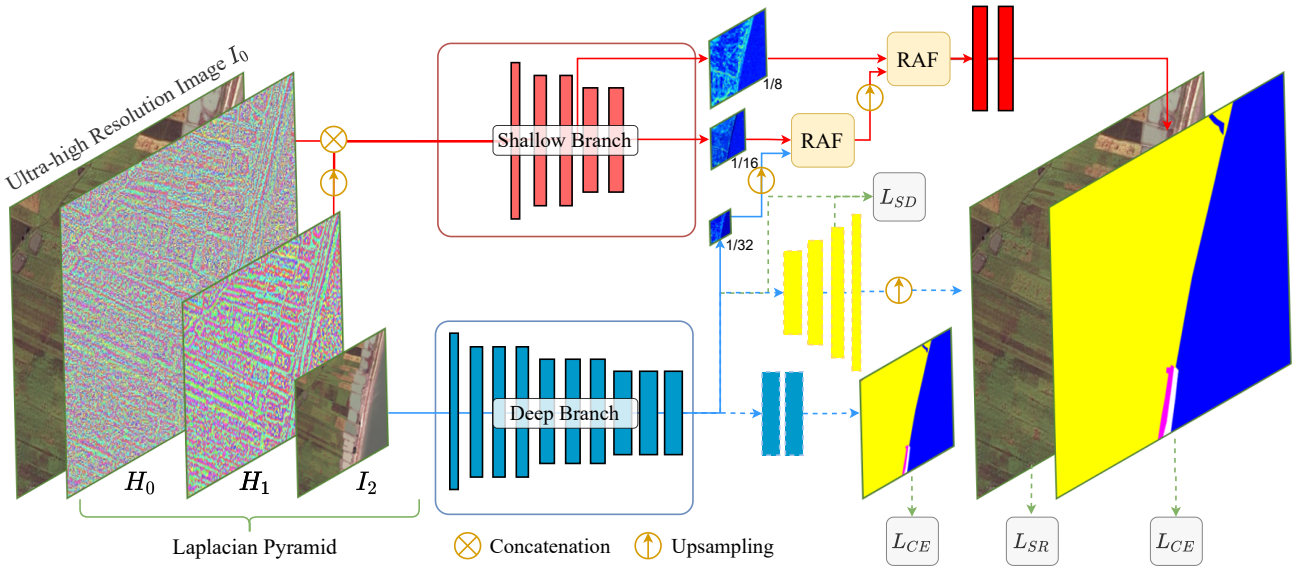


Figure 3. Pipeline of the proposed ISDNet for high-resolution image segmentation. Given a image $I_0 \in \mathbb{R}^{H\times W\times 3}$, we first decompose it into a Laplacian pyramid (e.g., n = 2). Let $I_i, H_i \in \mathbb{R}^{\frac{H}{2^i}\times\frac{W}{2^i}\times 3}(i = 0, 1, ..., n)$ denote RGB images and high-frequency residuals, respectively. For the high-frequency residual H_i , we process it by the shallow branch, marked by red arrows. Blue arrows represent the deep branch takes the downscale RGB image I_2. Then the RAF module fuses the feature from different branches. Green arrows stand for the optimization loss functions for each module. Dotted lines and bounding boxes represent modules only utilized in training.

strengthen local segmentation. However, the above ultra-high resolution segmentation methods crop input images into patches towards low GPU memory, which leads to redundant calculation and very slow inference speed. In contrast, without cropped patches, our method directly processes the full-scale and downsampled inputs by integrating shallow and deep networks, significantly accelerating the inference speed.

2.2. Generic Semantic Segmentation

With the development of convolution neural networks, FCN-based [25] methods [8, 10, 12, 16, 24] achieve impressive performance on various benchmarks. Deeplabv3 [1] employs an atrous spatial pyramid pooling module to capture multi-scale context. PSPNet [40] devises a pyramid pooling to capture both local and global context information in the dilation backbone. However, most approaches require a large computation costs due to the high-resolution feature and the complicated network connections. To address these limitations, ICNet [39] employs a multi-scale image cascade structure to achieve a good speed-accuracy trade-off. Besides, BiSeNetV1 [37] proposes a two-stream paths for low-level details and high-level context information, respectively. On the basis of BiSeNetV1 [37], STDC [11] presents a low-latency backbone to achieve fast speed and high accuracy. Moreover, it also introduces a boundary map as supervision so that the shallow stage of backbone can obtain edge-aware feature representation. However, the above methods fail to handle ultra-high resolution segmentation well. Differently, we design a novel ultra-high resolution segmentation framework with the shallow and deep networks and a novel RAF module, achieving state-of-the-art segmentation accuracy.

3. Methodology

3.1. Overview

In this method, we propose a novel framework to address the efficiency problem of ultra-high resolution segmentation methods. As shown in Figure 3, a deep network takes downsampled image to extract high-level semantic information, while a shallow network directly processes *full-scale* inputs with enhanced spatial details (Section 3.2). Besides, a novel feature fusion module (Section 3.3) is introduced to fuse these branches based on their relationship. Moreover, we employ auxiliary segmentation and super-resolution tasks to learn better features for the deep branch (Section 3.4).

3.2. Architecture

Typically, higher resolution or deeper networks [1] lead to better performance. However, they are also slow and memory-intensive in inference. Previous methods reduce inference memory by combining cropped and downsampled inputs, but this solution is still slow due to the fusion of cropped patches [3, 22]. To further speed-up inference, we design a bilateral architecture, integrating a pair of deep and shallow networks, denoted by $\mathcal{D}$ and $\mathcal{S}$, respectively. Since the shallow branch $\mathcal{S}$ has fewer layers and

faster inference speed, its input does not need any downsampling or cropping. We input full resolution images to $\mathcal{S}$ to extract detailed spatial information. To further force the shallow branch to learn the complementary spatial details, we replace RGB images with high-frequency residuals as inputs. High-frequency residuals $\{H\}_{i=0}^{n}$ (*e.g.* $n = 1$) are computed by Laplacian pyramid:

$$H_i = g_i(I) - \text{Upsample}(g_{i+1}(I))), \quad (1)$$

where I represents the full scale image, $g(.)$ denotes guassian blur and i is the number of levels in the pyramid.

The outputs of shallow branch ($\mathcal{S}$) are two feature maps with $\frac{1}{8}$ and $\frac{1}{16}$ resolution of the original image. For the deep branch $\mathcal{D}$, since the inference is slow for high-resolution images, we input downsampled RGB images, similar to previous approaches [3, 4], and outputs $\frac{1}{32}$ feature map which extracts high-level semantic information. Different from [3, 4], to better fuse this branch's features with the detailed shallow branch, during training, we introduce an auxiliary segmentation head, a super-resolution head, and a structure distillation loss, for deep supervision of this branch (Section 3.4), while these modules are not used in inference.

The three feature maps extracted from these two branches are then fused by a cascade of feature fusion modules (Section 3.3). Lastly, a standard segmentation head produces the final prediction from the fused feature map.

3.3. Relation-Aware feature Fusion

Common approaches employ addition or concatenation [3] to fuse features from different branches. Some methods [11, 37] apply attention mechanism to re-weight different channels, for each feature map separately. However, it is unreasonable to assume that features from deep and shallow branches contribute equally to feature fusion.

Therefore, we introduce Relational-Aware feature Fusion (RAF), to exploit the relationship between the shallow feature (detailed spatial information) and the deep feature (high-level semantic information) (Figure 4). Let $F_s \in \mathbb{R}^{C\times H_s\times W_s}$, $F_d \in \mathbb{R}^{C\times H_d\times W_d}$ denote feature maps from shallow and deep networks, respectively. First, channel-wised attention att is computed as follows:

$$att = fc(GAP(F)). \quad (2)$$

Thus, attention vectors $att_s, att_d \in \mathbb{R}^C$ are generated for F_s, F_d, respectively. att_s, att_d are then orderly divided into k groups with length r [23], denoted by $G_s, G_d \in \mathbb{R}^{k\times r}$.

Then, we define the relationship matrix $R \in \mathbb{R}^{k\times k}$ between G_s and G_d by inner product for each group pairs:

$$R = G_s G_d^T. \quad (3)$$

Define modulation factor $M \in \mathbb{R}^C$ as:

$$M = \sigma(att + \alpha fc(flatten(R))), \quad (4)$$

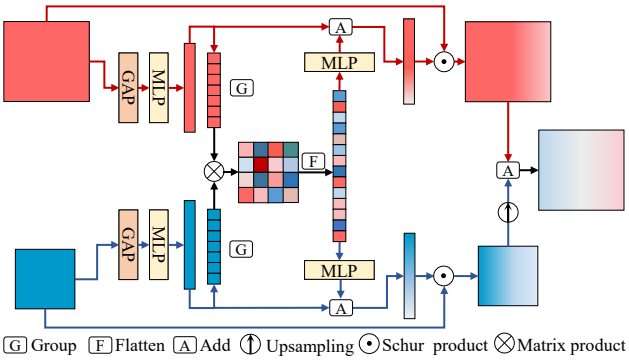


Figure 4. Illustration of Relation-Aware Feature Fusion module. Blue and red represent feature maps produced by deep and shallow branches, respectively.

where σ is sigmoid function and α is a learnable parameter.

After that, the fused feature F_{fusion} is computed as follows:

$$F_{fusion} = M_s \cdot F_s + \text{Upsample}(M_d \cdot F_d). \quad (5)$$

where $\cdot$ denotes element-wise multiplication.

3.4. Loss Functions

Segmentation loss. The standard cross-entropy loss is used for the final segmentation results ($\mathcal{L}_{SEG}$) and the auxiliary segmentation head after the deep branch ($\mathcal{L}_{AUX}$).

Super-resolution loss. The deep branch uses low-resolution images as input, thus producing noisy features, especially around boundaries. To learn a more accurate representation, we add a super-resolution head to reconstruct the original image I_0. The common mean squared error loss L_{SR} is used to supervise the reconstructed image I_{rec}:

$$\mathcal{L}_{SR} = \|I_0 - I_{rec}\|_2^2. \quad (6)$$

Structure distillation loss. Directly adding the above super-resolution task without interaction brings limited improvement. To strengthen the interaction between the super-resolution and semantic segmentation tasks, inspired by [31], we propose to distill structural information from the last layer of the super-resolution head, to strengthen the deep branch feature. Specifically, we denote by F_d the deep branch feature, and F_{sr} the super-resolution head feature down-sampled to the same resolution as F_d, the structure distillation loss $\mathcal{L}_{SD}$ is defined as follows:

$$\mathcal{L}_{SD} = \left\| F_d^T F_d - F_{sr}^T F_{sr} \right\|. \quad (7)$$

Overall loss. The overall loss $\mathcal{L}$ is a weighted combination of all above losses:

$$\mathcal{L} = \mathcal{L}_{SEG} + \lambda_1 \mathcal{L}_{AUX} + \lambda_2 \mathcal{L}_{SR} + \lambda_3 \mathcal{L}_{SD}. \quad (8)$$

Note that both the super-resolution head and the segmentation head for the deep branch are only used in training.

4. Experimental Results

We first introduce the datasets and implementation details. Then, we make a contrastive analysis with other methods. We adopt mean of class-wise intersection over union (mIoU), memory consumption, and Frames-per-second (FPS) as the main metrics. Finally, we discuss the impact of each component in our proposed approach.

4.1. Datasets

To evaluate the proposed method, we carry out comprehensive experiments on two widely used ultra-high resolution image segmentation datasets: DeepGlobe [7] and Inria Aerial [26]. In addition, we use a popular generic dataset Cityscapes [6] to verify the generality of our method.

DeepGlobe. The DeepGlobe dataset contains 803 images with 2448×2448 resolution. It contains 7 classes of landscape regions, in which the class named "unknown" is not considered in the evaluation. We follow the protocol as [3], by splitting images all of the images into training, validation and test set with 455, 207 and 142 images respectively.

Inria Aerial. The Inria Aerial dataset provides 180 images with 5000×5000 resolution and dense annotations with a binary mask for building and non-building areas. Following [3], we split images into training, validation and test set with 126, 27 and 27 images respectively.

Cityscapes. The Cityscapes dataset is a popular generic dataset for semantic segmentation, which has 5,000 fine annotated images with 1024×2048 resolution. We follow the official data split for our experiments, which contains 2,975 images for training, 500 images for validation and the rest 1525 images for testing.

4.2. Implementation Details

Our method integrates the deep and shallow models, which can be leveraged to combine lots of general semantic segmentation networks. Without specifical statement, we employ DeepLabv3 [1] with ResNet18 [14] as the deep branch, in which the segmentation head will be discarded during the inference. Besides, we utilize a recent lightweight model STDC [11] as the shallow branch, in which only the first four stages are used. We initialize both branches by the corresponding pretrained models on ImageNet. It is worth noting that we copy the weights 2 times for the first layer in the pretrained model to match the 6 channels heterogeneous input for the shallow branch. We use $\lambda_1 = 1, \lambda_2 = 0.1, \lambda_3 = 1$ for all experiments.

We adopt the mmsegmenation [5] toolbox as our codebase and follow the default augmentations without bells and whistles. All parameters are optimized by SGD with momentum 0.9. The initial learning rate is configured as 10^{-3} with the polynomial decay parameter of 0.9, together with the maximum iteration number set to 40k, 80k and 160k for Inria Aerial, DeepGlobe and Cityscapes respectively. All experiments use a batch size of 8 for training on a DGX-1 workstation with Tesla V100 GPUs. We use the command line tool "gpustat" to measure the GPU memory. Memory and Frames-per-second (FPS) are measured on a RTX 2080Ti GPU with a batch size of 1. [1]

4.3. Experiments on the DeepGlobe Dataset

We first apply our framework to DeepGlobe [7], an aerial dataset with ultra-high resolution images. Due to the diversity of land cover types and the high density of annotations, this dataset is very challenging.

Firstly, we compare our method with several generic and specifically designed segmentation methods. Table 1 shows comparison results. On the one hand, since generic methods are not suitable to input images with a large scale, there are two common ways to segment large scale images for generic models: (1) ***Global Inference***, which train and test the model on a downsample scale. (2) ***Local Inference***, which train and test the model on cropped images, require multiple times inference then merge local results. On the other hand, we also compare with methods specifically for ultra-high images, denoted as ***UHR Methods***, including GLNet [3], MagNet [18] and FCtL [22] *etc*.

Compared with approaches in Table 1, our method not only achieves the highest mIoU, but also attains a better balance among accuracy, speed and memory. Concretely, there are two critical observations: 1) Processing patches will increase inference time. Recent UHR methods require abundant refinement on uncertain regions, which limits the inference speed. Besides, for generic methods, local inference need to process more pixels than global inference, causing a slow inference speed. 2) Using down-sample inputs causes the missing of small objects and reduced accuracy at semantic boundaries. As shown in Table 1, for generic models, local inference is better than global inference on mIoU since that global inference loses much detailed information.

Qualitative results are shown in Figure 5. The first row shows that our method achieves more detailed segmentation results on "ubran" class, represented by cyan. In the second row, compared with FCtL [22] and STDC [11], our method attains a clear segmentation result on the boundary between "agriculture" and "forest" classes, marked by yellow and green, respectively.

In conclusion, our method utilizes the deep branch to extract semantic context from downsampled images and em-

[1] It isn't recommended to compare FPS from different papers: speed is related on environments, so we measured most of the competitors under our environments. We provide this script in the supplementary.

[2] The results of FCtL are slightly different from the original paper, in which test time augmentation(TTA) are used. For a fair comparison with other methods, we evaluate it with the checkpoints provide by the official repository without TTA.

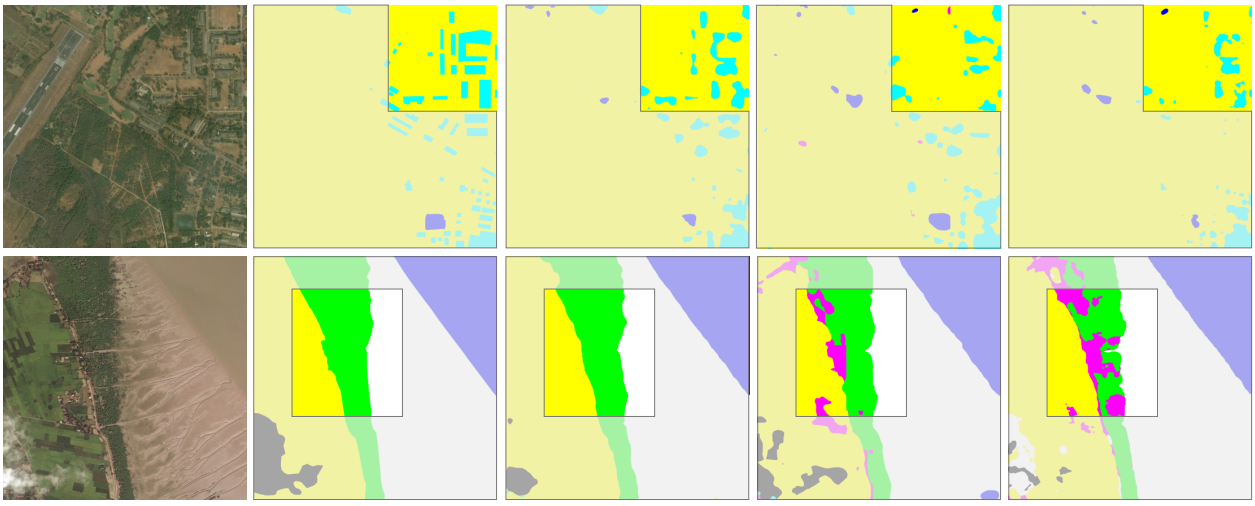


Figure 5. We illustrate several examples of the DeepGlobe dataset, comparing with the SOTAs. In this figure, masks with varied colors represent different semantic regions. Particularly, cyan represents "urban", yellow represents "agriculture", purple represents "rangeland", green represents "forest", blue represents "water", white represents "barren" and black represents "unknown".

Method	mIoU ↑	FPS ↑	Memory(MB) ↓
Local Inference			
UNet [29]	46.53	1.26	1741
FCN-8s [25]	62.43	4.55	**970**
DeepLabv3+ [2]	69.69	1.60	1541
Global Inference			
UNet [29]	50.11	3.54	7627
FCN-8s [25]	52.86	7.91	1984
DeepLabv3+ [2]	63.50	4.44	3226
BiSeNetV1 [37]	53.00	14.20	1801
STDC [11]	70.30	14.00	2580
UHR Methods			
GLNet [3]	71.60	0.17	1865
CascadePSP [4]	68.50	0.11	3236
PPN [33]	71.90	12.90	1193
PointRend [20]	71.78	6.25	1593
MagNet [18]	72.96	0.80	1559
MagNet-Fast [18]	71.85	3.40	1559
FCtL^2 [22]	72.76	0.13	4332
Ours(ISDNet)	**73.30**	**27.70**	1948

Table 1. Segmentation results on the DeepGlobe dataset. We evaluate the speed and memory under our environment, and the accuracy of competitors are collected from [18].

ploy the shallow branch to inference whole images. Therefore, without inputting cropped patches, our method can achieve 2.5× faster than PPN [33] with higher accuracy.

4.4. Experiments on the Inria Aerial Dataset

To further illustrate the effectiveness of our method, we apply our method to Inria Aerial [26]. In this dataset, the number of pixels for each image has reached 25 million which is around four times than that in the DeepGlobe. Besides, the foreground regions in each image are finer, which makes it more challenging for segmentation methods.

Similarly, we compare our method with generic and UHR methods. As shown in Table 2, our method achieves the best performance on both mIoU and FPS. In general, UHR methods are more accurate than generic methods along with lower memory consumption. However, most of UHR methods suffer from the local refinement are too slow to real-world application. Compared with FCN-8s [25] that occupies the least memory, our method attains a clear improvement on both mIoU and FPS. Compared with FCtL [22], a recent UHR method, the inference speed (6.90 FPS) of our method is nearly 172× than FCtL (0.04 FPS). More importantly, our method only increases a few memory. Besides, Figure 6 shows qualitative results. From the cropped patches that marked by orange bounding boxes, we can see that our segmentation results are more precise in contrast to other methods. In a nutshell, our method also achieves a better balance among accuracy, speed and memory on the Inria Aerial dataset.

4.5. Experiments on the Cityscapes Dataset

The Cityscapes [6] is a high resolution datasets for autonomous driving, which is popular in semantic segmentation community. Therefore, we also apply our framework to this datasets to evaluate the model generality.

We conduct two experiments on the Cityscapes dataset. Table 3 shows the comparison of our method with generic and UHR segmentation methods. Our method significantly boosts the accuracy among UHR methods, and achieves comparable performance compared with generic methods. Compared with deep models such as Deeplabv3 [1], our

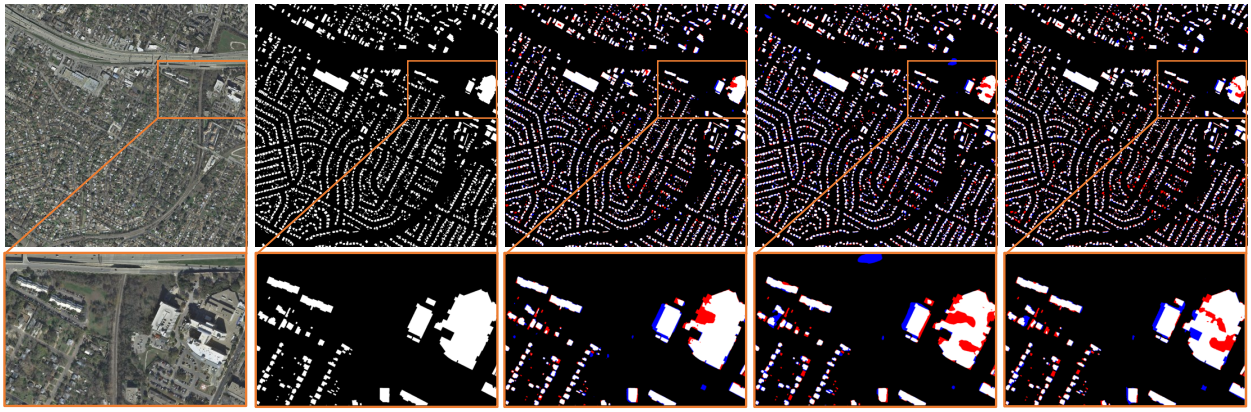


Figure 6. We illustrate several examples of the Inria Aeril dataset, comparing with the SOTAs. In this figure, white and black represent building and non-building respectively. Besides, in the segmentation results, we employ red and blue to mark the area with misclassification. Specifically, red represents foreground is classified into background, and vice versa for blue.

Method	mIoU ↑	FPS ↑	Memory(MB) ↓
Generic Methods			
DeepLabv3+ [2]	55.90	1.67	5122
FCN-8s [25]	69.10	1.90	**2447**
STDC [11]	72.44	4.97	7410
UHR Methods			
CascadePSP [4]	69.40	0.03	3236
GLNet [3]	71.20	0.05	2663
FCtL^2 [22]	72.87	0.04	4332
Ours (ISDNet)	**74.23**	**6.90**	4680

Table 2. Segmentation results on the Inria Aerial dataset. We evaluate the speed and memory under our environment, and the accuracy of competitors are collected from [22].

method attain a clear improvement on the speed with similar accuracy and memory. On the other hand, compared with lightweight methods, our method still maintains an advantage in accuracy and memory consumption.

Table 4 shows the generality of our method. We integrate PSPNet [40] (CNN-based) and Segformer [35] (Transformer-based) as the deep branch in our framework. For full scale input, we use the setting of mmsegmentation toolbox to train the model. For fair comparison, we utilize random scale $[0.5, 1]$ and $[0.125, 0.5]$ respectively to train models with downscale 2 and 4 input. The results show that our mIoU is higher than PSPNet with downscale 2 and downscale 4 input. Compared with PSPNet using full size inputs, our method is $3.8\times$ faster than PSPNet although we loss some accuracy. Moreover, we also utilize Segformer [35], a transformer-based method, to validate the generality. From the Table 4 we can obtain a similar conclusion with CNN-based methods.

In a summary, our method achieves high accuracy with less inference time on Cityscapes dataset with a good generality to extend existing segmentation models.

Method	mIoU ↑	FPS ↑	Memory(MB) ↓
Generic Methods			
BiSeNetV1 [37]	74.44	42.43	2147
BiSeNetV2 [36]	75.80	43.07	1602
PSPNet [40]	74.87	15.15	1584
ICNet [39]	74.43	**68.55**	**1390**
STDC [11]	74.5	62.15	1536
DeepLabv3 [1]	**76.70**	13.32	1468
UHR Methods			
DenseCRF [21]	62.95	0.04	1575
DGF [32]	63.33	3.13	1727
SegFix [38]	65.83	2.63	2033
PointRend [20]	64.39	7.14	2052
MagNet [18]	67.57	0.34	2007
MagNet-Fast [18]	66.91	3.13	2007
Ours (ISDNet)	**76.02**	**50.79**	**1510**

Table 3. Segmentation results on the CityScapes dataset. We evaluate the speed and memory under our environment, and the accuracy of UHR competitors are collected from [18].

Method	mIoU	FPS	Mem(MB)
PSPNet [40]	74.87	15.15	1584
PSPNet [40] (½ scale)	72.87	54.99	1160
PSPNet [40] (¼ scale)	65.20	169.91	1076
PSPNet [40] + ISD	74.30	58.29	1540
Segformer-b0 [35]	73.45	13.70	3114
Segformer-b0 [35] (½ scale)	71.20	65.49	1174
Segformer-b0 [35] (¼ scale)	51.19	76.22	1032
Segformer-b0 [35] + ISD	72.99	41.82	1500

Table 4. Comparison of existing models integrating with our framework. We evaluate the corresponding methods with different scales to compare the accuracy and inference cost.

4.6. Ablation Study

Effectiveness of our architecture. We conduct an ablation experiment on each branch to evaluate the effectiveness of

$\mathcal{D}$	$\mathcal{S}$	Full Scale	¼ Scale	mIoU	FPS
✓			✓	61.40	64.64
✓		✓		73.23	5.40
	✓		✓	56.39	200.64
	✓	✓		70.30	13.79
✓	✓	✓	✓	71.69	31.69

Table 5. Effectiveness of our architecture. $\mathcal{D}$ and $\mathcal{S}$ denote the deep and shallow branch, respectively. Scale means the input size compared with original images.

the trade-off between accuracy and speed. Table 5 shows the comparison result. We only use the baseline network for a fair comparison. Specifically, the baseline model contains deep and shallow branches, optimized by L_{SEG} and $\mathcal{L}_{AUX}$. Besides, we utilize simple addition to fuse F_s and F_d instead of RAF. As shown in Table 5, whether it is the deep or shallow networks, it is hard to reach a desirable trade between speed and accuracy. For the deep branch, full-scale inputs have satisfied accuracy but slow speed. But $\frac{1}{4}$ scale input cannot maintain the accuracy with high speed. And the shallow branch has a similar conclusion. However, the baseline achieves a proper balance between speed and accuracy. The baseline increases speed for the deep branch with full-scale inputs by nearly $6\times$. Besides, we significantly improve the accuracy compared to the shallow branch with $\frac{1}{4}$ scale inputs. Hence, our architecture can attain a satiable trade-off between accuracy and speed.

Comparison of feature fusion methods. We conduct an ablation experiment to assess the effectiveness of Relation-Aware feature Fusion module. Table 6 shows results. This experiment employs the baseline with the high-frequency residual input H, optimized by overall loss functions in Section 3.4. Besides, the addition and concatenate with channel-wise attention are ARM and FFM in [37], respectively. As shown in Table 6, our relation-aware attention strategy achieves a satisfying trade-off among accuracy, speed, and memory. Compared with naively addition and ARM, our module has better performance on accuracy. Besides, the RAF requires less memory and inference faster than concatenation and FFM. Therefore, the proposed module is suitable to fuse the F_s and F_d from deep and shallow branches, respectively.

Effectiveness of losses and input types. We carry out an ablation experiment to validate the usefulness of $\mathcal{L}_{SR}$ and $\mathcal{L}_{SD}$ in our method. And we also evaluate the utility of high-frequency residual inputs. In this experiment, we train the ISDNet with $\mathcal{L}_{SEG}$ and $\mathcal{L}_{AUX}$ as the baseline. As shown in Table 7, the $\mathcal{L}_{SR}$ and $\mathcal{L}_{SD}$ increase the accuracy with $+0.39$. Moreover, for the input of the shallow input, replacing the RGB image with high-frequency residuals can obtain the improvement of $+0.6$. In conclusion, Both the auxiliary super-resolution task and high-frequency inputs can increase the performance.

ADD	CAT	CW	M_s	M_d	mIoU	FPS	Mem(MB)
✓			-	-	72.20	31.69	-
✓		✓	-	-	72.42	29.73	1891
	✓		-	-	71.88	23.98	-
	✓	✓	-	-	72.57	25.76	2204
✓		✓	✓		72.63	28.93	-
✓		✓	✓	✓	73.30	27.70	1948

Table 6. Comparison of feature fusion methods. ADD and CAT represent two simple fusion strategies: addition and concatenation. CW means channel-wise attention mechanism. M_s and M_d denote the relation-aware attention for deep and shallow branch.

Baseline	$\mathcal{L}_{SR}$	$\mathcal{L}_{SD}$	H	mIoU
✓				72.31
✓	✓			72.55
✓	✓	✓		72.70
✓	✓	✓	✓	73.30

Table 7. Comparison of loss components and heterogeneous input. H indicates high-frequency residual inputs for the shallow branch.

5. Conclusion and Limitations

This paper has explored integrating deep and shallow networks for efficient ultra-high resolution image segmentation. To exploit relational information between branches, we have introduced a novel feature fusion module: Relation-Aware feature Fusion (RAF). To further enhance the shallow branch, we have proposed to use high-frequency residuals as input to strengthen spatial details. Besides, super-resolution loss and structure distillation loss are introduced to enhance features from the deep branch. Our method substantially speeds up ultra-high image segmentation and has achieved state-of-the-art performance across three popular datasets.

Nevertheless, there are some limitations to this work. For example, uniform down-sampling is used for the deep branch. Replacing it with other adaptive down-sampling methods (*e.g.*, deformable down-sampling) might improve performance. Besides, we only provided one type of shallow network. A systematic exploration of more architecture is worthy of future research with more resources.

6. Acknowledgements.

This work was supported by the National Key Research and Development Program of China (2019YFC1521104), National Natural Science Foundation of China (72192821, 61972157), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), Shanghai Science and Technology Commission (21511101200, 22YF1420300), and Art major project of National Social Science Fund (I8ZD22). We thank Zhengyang Feng for insightful discussions.

References

[1] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 1, 3, 5, 6, 7

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 6, 7

[3] Wuyang Chen, Ziyu Jiang, Zhangyang Wang, Kexin Cui, and Xiaoning Qian. Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8924–8933, 2019. 2, 3, 4, 5, 6, 7

[4] Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. Cascadepsp: toward class-agnostic and very high-resolution segmentation via global and local refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8890–8899, 2020. 2, 4, 6, 7

[5] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 5

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2, 5, 6

[7] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 172–181, 2018. 2, 5

[8] Xiaohan Ding, Yuchen Guo, Guiguang Ding, and Jungong Han. Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 3

[9] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Diverse branch block: Building a convolution as an inception-like unit. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10886–10895, June 2021. 1

[10] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13733–13742, 2021. 3

[11] Mingyuan Fan, Shenqi Lai, Junshi Huang, Xiaoming Wei, Zhenhua Chai, Junfeng Luo, and Xiaolin Wei. Rethinking bisenet for real-time semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9716–9725, 2021. 1, 3, 4, 5, 6, 7

[12] Guang Feng, Zhiwei Hu, Lihe Zhang, and Huchuan Lu. Encoder fusion network with co-attention embedding for referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15506–15515, 2021. 3

[13] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019. 1

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[15] Yufan He, Dong Yang, Holger Roth, Can Zhao, and Daguang Xu. Dints: Differentiable neural network topology search for 3d medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5841–5850, 2021. 1

[16] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 1, 3

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 1

[18] Chuong Huynh, Anh Tuan Tran, Khoa Luu, and Minh Hoai. Progressive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16755–16764, 2021. 2, 5, 6, 7

[19] Wei Ji, Shuang Yu, Junde Wu, Kai Ma, Cheng Bian, Qi Bi, Jingjing Li, Hanruo Liu, Li Cheng, and Yefeng Zheng. Learning calibrated medical image segmentation via multi-rater agreement modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12341–12351, 2021. 1

[20] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9799–9808, 2020. 6, 7

[21] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24:109–117, 2011. 7

[22] Qi Li, Weixiang Yang, Wenxi Liu, Yuanlong Yu, and Shengfeng He. From contexts to locality: Ultra-high resolution image segmentation via locality-aware contextual correlation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7252–7261, 2021. 1, 2, 3, 5, 6, 7

[23] Xiangtai Li, Xia Li, Ansheng You, Li Zhang, Guang-Liang Cheng, Kuiyuan Yang, Y. Tong, and Zhouchen Lin. Towards efficient scene understanding via squeeze reasoning. *ArXiv*, abs/2011.03308, 2020. 4

[24] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 1, 3

[25] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 3, 6, 7

[26] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez. Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3226–3229. IEEE, 2017. 2, 5, 6

[27] Xingang Pan, Jianping Shi, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Spatial as deep: Spatial cnn for traffic scene understanding. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1

[28] Simon Reiß, Constantin Seibold, Alexander Freytag, Erik Rodner, and Rainer Stiefelhagen. Every annotation counts: Multi-label deep supervision for medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9532–9542, 2021. 1

[29] O Ronneberger, P Fischer, and T Brox. U-net: Convolutional networks for biomedical image segmentation. arxiv. *Lecture Notes in Computer Science*, 2015, 2015. 6

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 1

[31] Li Wang, Dong Li, Yousong Zhu, Lu Tian, and Yi Shan. Dual super-resolution learning for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3774–3783, 2020. 4

[32] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1838–1847, 2018. 7

[33] Tong Wu, Zhenzhen Lei, Bingqian Lin, Cuihua Li, Yanyun Qu, and Yuan Xie. Patch proposal network for fast semantic segmentation of high-resolution images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12402–12409, 2020. 2, 6

[34] Xiao Wu, Ting-Zhu Huang, Liang-Jian Deng, and Tian-Jing Zhang. Dynamic cross feature fusion for remote sensing pansharpening. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14687–14696, October 2021. 1

[35] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021. 2, 7

[36] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *International Journal of Computer Vision*, pages 1–18, 2021. 2, 7

[37] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018. 2, 3, 4, 6, 7, 8

[38] Yuhui Yuan, Jingyi Xie, Xilin Chen, and Jingdong Wang. Segfix: Model-agnostic boundary refinement for segmentation. In *European Conference on Computer Vision*, pages 489–506. Springer, 2020. 7

[39] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 405–420, 2018. 3, 7

[40] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1, 3, 7

[41] Zhuo Zheng, Ailong Ma, Liangpei Zhang, and Yanfei Zhong. Change is everywhere: Single-temporal supervised object change detection in remote sensing imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15193–15202, October 2021. 1

[42] Zhuo Zheng, Yanfei Zhong, Junjue Wang, and Ailong Ma. Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1