PointFlow: Flowing Semantics Through Points for Aerial Image Segmentation

Anonymous CVPR 2021 submission

Paper ID 3065

Abstract

Aerial Image Segmentation is a particular semantic segmentation problem and has several challenging characteristics that general semantic segmentation does not have. There are two critical issues: The one is an extremely foreground-background imbalanced distribution, and the other is multiple small objects along with the complex background. Such problems make the recent dense affinity context modeling perform poorly even compared with baselines due to over-introduced background context. To handle these problems, we propose a point-wise affinity propagation module based on the FPN framework, named PointFlow. Rather than dense affinity learning, a sparse affinity map is generated upon selected points between the adjacent features, which reduces the noise introduced by the background while keeping efficiency. In particular, we design a dual point matcher to select points from the salient area and object boundaries, respectively. Experimental results on three different aerial segmentation datasets suggest that the proposed method is more effective and efficient than state-of-the-art general semantic segmentation methods. Especially, our methods achieve the best speed and accuracy trade-off on three aerial benchmarks. Further experiments on three general semantic segmentation datasets prove the generality of our method. Both code and models will be available for further research.

1. Introduction

High spatial resolution (HSR) remote sensing images contain various geospatial objects, including airplanes, ships, vehicles, buildings, etc. Understanding these objects from HSR remote sensing imagery has great practical value for urban monitoring and management. Aerial Image segmentation is an important task in remote sensing understanding that can provide semantic and localization information cues for interest targets. It is a specific semantic segmentation task that aims to assign a semantic category to each image pixel.

However, besides the large scale variation problems in most semantic segmentation datasets [12, 61, 35, 6], aerial images have their own challenging problems including high background complexity [60], background and foreground imbalance [45], tiny foreground objects in high resolution images. As shown in the first row of Fig.1, the red boxes show the tiny objects in the scene while the yellow box and blue box show the complex background context, including houses and trees receptively. Current general semantic segmentation methods mainly focus on scale variation in the natural scene by building multi-scale feature representation [59, 8] or enhancing the object boundaries with specific design module [23, 40]. They fail to work well due to the lack of explicit modeling for the foreground objects. For example, several dense affinity-based methods [16, 52] also obtain inferior results mainly because the imbalanced and complex background will fool the affinity learning on small objects. For example, both yellow boxes and blue boxes have the same semantic meaning of background but with

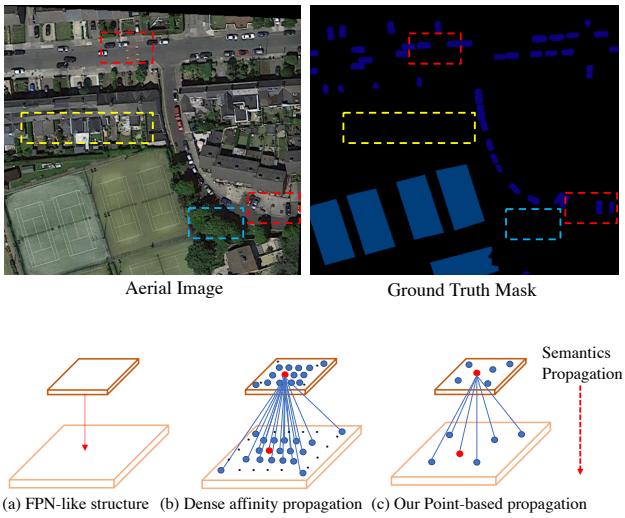


Figure 1: Illustration of an aerial image segmentation example and our proposed module. The first row presents the input image and ground truth with complex backgrounds and small objects. The second row indicates the schematic diagram on *dense affinity propagation* and our proposed *point-based propagation* module.

a huge appearance change. Dense affinity learning forces pixels on small objects to absorb such noisy context which leads to inferior segmentation results. FarSeg [60] adopts FPN-like [30] design and solve the background and foreground imbalance problems by introducing a foreground-aware relation module. However, for small objects, there still exists some semantic gaps in different features in FPN. Namely, the gap is between the high-resolution features with low semantic information and low-resolution features with high semantic information. As shown in Fig.1, tiny objects like cars need more semantic information in lower layers with high resolution.

In this paper, we propose a point-based information propagation module to handle the previous problems stated above. We propose PointFlow Module (PFM), a novel and efficient module for specific semantic points propagation between adjacent features. Our module is based on the FPN framework [30, 22] to make up the semantic gap. As shown in the last row of Fig.1, rather than simple fusion or dense affinity propagation on each point as the previous work Non-local module [43], PointFlow selects the several representative points between any adjacent feature pyramid levels. In particular, we design Dual Point Matcher by selecting matched point features from the salient area and object boundaries receptively. The former is obtained from explicit max pooling operation on learned salient map. The latter is conditioned on the predicted object boundaries where we adopt a subtraction-based prediction. Then the point-wised affinity is estimated according to the point features that are sampled from both adjacent features. Finally, the higher layer points are fused into lower layers according to the affinity map. Our PFMs select and propagate points on foreground objects and sampled background areas to simultaneously handle both the semantic gap and foreground-background imbalance problem.

Then we carry out detailed studies and analysis on PFM in the experiment part, where it improves the various methods by a large margin with negligible GFlops increase. Based on the FPN framework, by inserting PFMs between feature pyramids, we propose the PFNet. In particular, PFNet suppresses previous method FarSeg [60] by **3.2%** point on iSAID [45]. Moreover, we also benchmark the recent state-of-the-art general semantic segmentation methods[42, 52, 26] on three aerial segmentation datasets including iSAID, Vaihingen and Postdam for the community. Benefited from efficient FPN design [22], our PFNet also achieves the best speed and accuracy on three benchmarks. Finally, we further verify the effectiveness of PFM on general semantic segmentation benchmarks, including Cityscapes [12], ADE-20k [61], and BDD [49] and it achieves considerable results with previous work [8, 52] with fewer GFLOPs. Our main contributions are three-fold:

1) We propose PointFlow Module (PFM), a novel and efficient module for poise-wised affinity learning, and we design a Dual Point Matcher to select the matched sparse points from salient areas and boundaries in a complementary manner.

2) We append PFM into PFN architecture and build a pyramid propagation network called PFNet.

3) Extensive experiments and analysis indicate the efficacy of PFM. We benchmark 15 state-of-the-art general segmentation methods on three aerial benchmarks. Our PFNet achieves state-of-the-art results on those benchmarks also with the best speed and accuracy trade-off. We further prove the generality of our method on three general semantic segmentation datasets.

2. Related Work

General Semantic Segmentation The general semantic segmentation has been eminently motivated by the fully-convolutional networks (FCNs) [32]. The following works [59, 7, 8, 9, 48, 42] mainly exploit the spatial context to overcome the limited receptive field of convolution layer which leads to the multi-scale feature representation. For example, ASPP [8] utilizes atrous convolutions [50] with different atrous rate to extract features with the different receptive field, while PPM [59] generates pyramidal feature maps via pyramid pooling. Several work [38, 2, 46, 58, 4, 22] use the encoder-decoder architecture to refine the output details. Recent works [26, 52, 20, 55, 57, 63, 51, 10, 27] propose to use non-local-like operator [41, 44] to harvest the global context of input images. Meanwhile, several works [23, 40, 53] propose to refine the object boundaries via specific designed processing. These general semantic segmentation methods ignore the special issues including imbalanced foreground-background pixels for modeling the context and increased small foreground objects in the Aerial Imagery. Thus these methods get inferior results which will be shown in the next section.

Semantic Segmentation of Aerial Imagery Several earlier works [21, 34, 33] focus on using multi-level semantic features on local patterns of images using deep CNN. Also, there exist a lot of applications, such as land use [19], building or road extraction [14, 28, 47, 3], agriculture vision [11]. They design specific methods based on existing semantic segmentation methods for special application scenario. In particular, relation net [36] captures long-range spatial relationships between entities by proposing spatial and channel relation modules. Recently, FarSeg [60] propose relation-based and optimization-based foreground modeling to handle the foreground-background imbalance problems in remote sensing imagery. However, the missing explicit exploration of semantics propagation between adjacent features limits the performance on the segmentation of small objects.

Multi Scale Feature Fusion Based on the FPN frame-

Method	OS	mIoU	Δ
dialated FCN[32, 50](baseline)	8	59.0	-
DAnet[59]	8	30.3	28.7↓
OCnet(ASP-OC) [52]	8	40.2	18.8↓
DAnet+FPN [30]	8	59.3	0.3↑
DAnet+our PFNet decoder	8	65.6	6.6↑
SemanticFPN [22](baseline)	32	61.3	-
+dense affinity [54]	32	58.9	2.4↓
+our PFM	32	65.0	3.7↑

Table 1: Simple experiment results on iSAID validation dataset. The dense affinity results in inferior results over various baselines. Appending our proposed PointFlow module results in a significant gain. OS: Output Stride in backbone.

work [30], rather than simple top-down additional fusion, several works propose to fuse feature through gates [15, 24], neural architecture search [17], pixel-level alignment [25] or adding bottom up path [31], dense affinity learning propagation[54]. Such full fusion methods may emphasize background objects like roads where the imbalance problem exists widely in aerial images. Our proposed PFM follows the design of FPN by propagating the semantics from the top to bottom. In contrast, rather than full fusion like previous works, our methods are based on point-level which select the several representative points to overcome the pixel imbalance problems in aerial imagery and leads to better results.

3. Method

In this section, we will first introduce some potential issues on dense point affinity learning for aerial segmentation task. Then we will provide detailed descriptions of our PointFlow module(PFM) to resolve the issues by selecting key semantic points for propagation efficiently. Finally, we will present our PFNet for aerial imagery segmentation.

3.1. Preliminary

Recent dense affinity based methods [43, 52, 16, 41, 54] have shown progressive results for semantic segmentation. The core idea of these methods is to model the pixel-wised relationship to harvesting the global context. As shown in Equ. 1, in the view of self-attention [41], each pixel p in 2-D input feature $F \in \mathbb{R}^{C\times H\times W}$ is connected to all the other pixels to calculate the pixel-wised affinity where A is the affinity function and it outputs affinity matrix $\in \mathbb{R}^{HW\times HW}$. C, H, and W denote the channel dimension, height, and width, respectively. Note that definitions of A can be different; we use the same label for simplicity.

$$F^r(p) = A(F(p), F(p))F(p) \tag{1}$$

However, applying these methods directly on the iSAID dataset leads to inferior results even compared with various baseline methods, as shown in Tab. 1 whether such module is appended after FCN backbone or is inserted into feature pyramids. The reason has two folds: (1) There exist extremely imbalanced foreground-background objects in the iSAID dataset. Explicit affinity modeling on complex background brings noise for outputs. (2) Too many small objects exist on the iSAID dataset, which requires high resolution and high semantic representation.

To solve the first problem, rather than dense affinity modeling, we can use a point sampler β to select matched representative points $\hat{p}$ to balance the background context ratio while keeping efficiency. For the second problem, to fill the semantic gap on small objects, we adopt the FPN framework and change the inputs of A by using adjacent features in a top-down manner shown in Equ. 2:

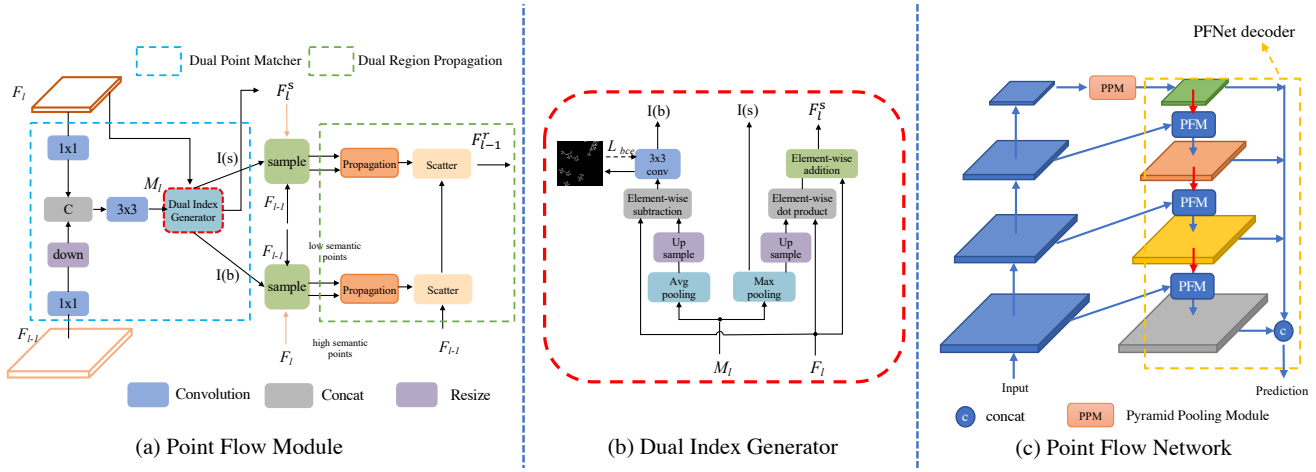
$$F^r(\hat{p}) = A(\beta(F_l(\hat{p})), \beta(F_{l-1}(\hat{p})))\beta(F_l(\hat{p}))) \tag{2}$$

where F_l and F_{l-1} are adjacent features in the FPN framework and $\hat{p}$ is sampled pixels for affinity modeling. We will detail the β in the following part. As shown in Tab.1, our method improves the baselines by a significant margin.

3.2. PointFlow Module

Motivation and Overview As the previous section shows the limitation of dense affinity on aerial image segmentation, we argue that unnecessary background pixels context may bring noises for foreground objects. Considering this, we propose to propagate context information through selective points, which can keep the efficiency in both speed and memory. Meanwhile, the semantic gap problems can also be fixed after propagation leading to high-resolution feature representation with high semantics, which is why we adopt FPN-framework design [30] in a top-down manner. Since our framework works in a top-down manner, and the semantics flow into low-level features through points, we name our module PointFlow. Our PointFlow is built on the FPN framework [30], where the feature map of each level is compressed into the same channel depth through two 1×1 convolution layers before entering the next level. Our module takes two adjacent feature maps as inputs $F_{l-1} \in \mathbb{R}^{C\times H\times W}$ and $F_l \in \mathbb{R}^{C\times H/2\times W/2}$ as the inputs where l means the index of feature pyramid and output refined $F^r_{l-1} \in \mathbb{R}^{C\times H\times W}$. For modeling β, we propose the Dual Point Matcher to select the points, and then the point-wise affinity can be calculated between adjacent points. Finally, the points with high-resolution and low semantics can be enhanced by the points with low-resolution high semantics according to the estimated affinity map. The process is shown in Fig. 2(a).

Dual Point Matcher The critical issue is how to find the corresponding points between two adjacent maps. We argue that most salient areas can be represented as key points for



(a) Point Flow Module (b) Dual Index Generator (c) Point Flow Network

Figure 2: (a), The overal pipeline of our proposed PointFlow Module. Left: Two adjacent features with one salient map are sent to the Dual Index Generator to obtain the sampled indexes. Right: The sampled point features are propagated from top to the bottom and finally scattered into the low level features point-wisely. (b), The detailed operation on proposed Dual Index Generator. (c), We design the PF Network Architecture by inserting PF modules into FPN-like framework.

balanced pixel-level propagation due to the unbalanced pixels between the foreground and background. Meanwhile, since there are many small objects in aerial scenes that need more fine-grained location cues, the boundary areas can also be considered the key points. Thus we design a novel Dual Point Matcher to consider the most salient part of inputs and object boundaries at the same time. The Dual Point Matcher has two steps: (1) Generate the salient map. (2) Generate sampled indexes from Dual Index Generator.

For the first step, we combine the input feature maps where the high-resolution part F_{l-1} is downsampled into the same low resolution through bilinear interpolation. The resized feature is denoted as $\widetilde{F}_{l-1}$. Then we perform one 3×3 convolution following with sigmoid function to generate the saliency map M_l. The process is shown as follows:

$$M_l = Sigmoid(\text{conv}_l(\text{Concat}(F_l, \widetilde{F}_{l-1}))), \quad (3)$$

For the second step, we take F_l and M_l as the inputs of the Dual Index Generator. We perform the adaptive max pooling on such map to obtain the most salient points. To highlight the salient part of foreground objects, we multiply such map on F_l with residual design as attention map shown in Equ. 4:

$$F_l^s = MaxPool(M_l) \times F_l + F_l, \quad (4)$$

We simply choose the salient indexes from $MaxPool(M_l)$. K is the number of pooled points, and it equals to the product of adaptive pooling kernels. We denote the salient indexes as $I(s)$ for short.

For boundary point selection, rather than simply using the binary supervision on the input feature F_l or F_{l-1} for boundary prediction, we propose to adopt residual prediction on the F_l. Our method is motivated by Laplacian pyramids in image processing [1, 5]. In Laplacian pyramids, the edge part of original images can be obtained by subtracting the smoothed upsampled images. Motivated by that, we use the average pooling on saliency map M_l and multiply the pooled map on F_l for smoothing inner content, then we subtract the such smoothed part from F_l to generate the sharpened feature $\widetilde{F}_l^b$ for boundary prediction. The process is shown in Equ. 5:

$$\widetilde{F}_l^b = F_l - AvgPool(M_l) \times F_l, \quad (5)$$

After the boundary prediction using $\widetilde{F}_l^b$, we obtain the boundary map B_l. Following the previous step, we simply sample Top-K points from the edge maps(K=128 by experiment) according to their confidence scores. We denote the boundary indices $I(b)$ for short. In total, the Dual Index Generator samples the key points in an orthogonal way by selecting points from specific regions according to the salient map M_l. The total process of Dual Index Generator is shown in Fig. 2(b).

Dual Region Propagation After the point matcher, we obtain the indexes $I(s)$ and $I(b)$, respectively. Then we sample the points from map from salient feature F_l^s and original input feature F_{l-1}. For each selected point, a point-wise feature representation is extracted on both adjacent input features. Note that features f for a real-value point are computed by bilinear interpolation of 4 nearest neighbors that are on the regular grid. We use normalized girds during the implementation. We denote f_l^s and f_l^b as sampled feature point at stage l for salient part and boundary part. We propagate those sampled points independently. For each sam-

pled point $\hat{p}$, the top-down propagation process is shown in Equ 6.

$$f_{l-1}(\hat{p})^r = \sum_{i \in \{I(b), I(s)\}} A(f^i_{l-1}(\hat{p}), f^i_l(\hat{p})) f^i_l(\hat{p}) + f^i_{l-1}(\hat{p}), \tag{6}$$

where A is affinity function, i means the indexes whether from $I(s)$ or $I(b)$. For A, we use the point-wise matrix multiplication along with softmax function for normalization. Following the previous work [18], we adopt the residual design for easier training. We calculate the sampled high semantic points through point-wise affinity according to the semantic similarity on sampled points with low semantics, which avoids the redundant background information in the aerial scene. Since we propagate semantics two times independently, we term two flows as *salient point flow* and *boundary point flow*, respectively. Finally, the refined feature F^r_{l-1} is obtained by scattering the f^r_{l-1} into F_{l-1} according to the indices $I(s)$ and $I(b)$.

3.3. Network Architecture

Overview Fig. 2 illustrates the our network architecture, which contains a bottom-up pathway as the encoder and a top-down pathway as the decoder. The encoder is backbone network with multiple feature pyramid outputs while the decoder is lightweight FPN equipped with our PFMs.

Network Architecture The encoder uses the ImageNet pretrained backbone with OS 32 rather dilation stragety with OS 8 for efficient inference. We additionally adopt the Pyramid Pooling Module (PPM) [59] for its superior efficiency and effectiveness to capture contextual information. In our setting, the output of PPM has the same resolution as that of the last stage. PFNet decoder takes feature maps from the encoder and uses the refined feature pyramid for final aerial segmentation according to previous work design [60, 22]. By simply replacing normal bilinear up-sampling with our PF module in top-down pathway of FPN, the PFNet decoder finally concatenates all the refined F^r_l(where l ranges from 2 to 5) by upsampling the inputs to the same resolution(1/4 resolution of input) and perform prediction. Note that our module can also be integrated into other architectures including Deeplabv3 [8] with a slight modification by appending such decoder after its head. More details can be found in the experiment part.

Loss Function For edge prediction in each PFM, we adopt binary BCE loss L_{bce}. For final segmentation prediction, we adopt the cross-entropy loss. The two losses are weighted to 1 by default.

4. Experiments

Overview: We will firstly perform ablation studies on iSAID dataset and give detailed analysis and comparison on PFM. Then we benchmark several recent works on Vaihingen and Potsdam datasets. Finally, we prove the generalizability of our module on general segmentation datasets.

4.1. Aerial Image Segmentation

DataSets: We use iSAID [45] dataset for ablation studies and report results on remaining datasets. iSAID [45] consists of 2,806 HSR images. The iSAID dataset provides 655,451 instances annotations over 15 categories of the object and it is the largest dataset for instance segmentation in the HSR remote sensing imagery. We also use Vaihingen and Postdam datasets[1] for benchmarking.

Implementation detail and Metrics: We adopt ResNet-50 [18] by default. Following the same setting [60], for all the experiments, these models are trained with 16 epoch on cropped images. For data augmentation, horizontal and vertical flip, rotation of 90 · k (k = 1, 2, 3) degree were adopted during training. For data preprocessing, we crop the image into a fixed size of (896, 896) using a sliding window striding 512 pixels. We use the mean intersection over union (mIoU) as the main metric for object segmentation to evaluate the proposed method if not specified. The baseline for ablation studies is Semantic-FPN [22] with OS 32.

Effectiveness on baseline models: In Tab. 2(a), adopting our PFMs leads to better results than appending PPM [59] shown in both 2nd and 3rd rows with about 1.2 % gap. After applying both PPM and PFM, there is a significant gain over the baseline models shown in the last row. Only applying boundary flow is slightly better than applying salient point flow which indicates the small object problems are more severe than foreground-background imbalance problems in this dataset. In Tab. 2 (b), we explore the effect on insertion position with our PFMs. From the first three rows, PF improves all stages and gets the greatest improvement at the first stage, which shows that the semantic gap is more severe for small objects in lower layers. After appending all FPMs, we achieve the best result shown in the last row.

Comparison with feature fusion methods: Tab. 2(c) give several feature fusion methods [13, 25, 54] used on scene understanding tasks. For all the methods, we replace these modules into the same position on PFnet decoder as in Fig 2(c) for fair comparison. Compared with DCN-like methods [13, 62, 25], our method leads to significant gain over them since our method can better handle the foreground semantics propagation.

Ablation on design choices: We give more detailed design studies in the second row of Tab. 2. Tab. 2(d) explores several sampling methods for salient points sampling. Attention based method is directly selecting top-K(K=128) points from M_l while uniform random sample is done by randomly selecting one pixels from 7×7 neighbor region of M_l(We report average result of 10 times experiments). Our max

[1] https://www2.isprs.org/commissions/comm2/wg4/benchmark/

+PPM	+salient point flow	+ boundary point flow	mIoU(%)
-	-	-	61.3
✓	-	-	63.8
	✓	✓	65.0
✓	✓	-	64.8
✓	-	✓	66.2
✓	✓	✓	66.9

(a) Effect of dual flow propagation on baseline.

Method	$\hat{\mathbf{F}}_3$	$\hat{\mathbf{F}}_4$	$\hat{\mathbf{F}}_5$	mIoU(%)
Baseline+PPM				63.8
	✓			65.8
		✓		65.6
			✓	65.5
		✓	✓	66.5
	✓	✓	✓	66.9

(b) Effect of Insertion Position. $\hat{\mathbf{F}}_l$ means the position between $\mathbf{F}_l$ and $\mathbf{F}_l - 1$.

Settings	mIoU(%)
baseline+PPM	63.8
+DCNv1 [13]	65.2
+DCNv2 [62]	65.6
+desne affinity flow [44]	62.0
+FAM [25]	65.7
+ Ours	66.9

(c) Comparison with Other Propagation Methods.

Sampling Method	mIoU(%)
baseline+PPM	63.8
uniform random	64.0
attention based	64.2
Our max pooling	64.8

(d) Effect of salient point sampling in Dual Index Generator.

Settings	mIoU(%)
baseline + PPM	63.8
top-down(td)	66.9
bottom-up(bu)	47.3
td then bu	54.5

(e) Effect of propagation direction.

Settings	mIoU(%)
baseline+PPM	63.8
direct prediction	65.7
addition prediction	65.5
Our subtraction based	66.2

(f) Effect of edge generation module in Dual Index Generator.

Network	Backbone	mIoU(%)	GFlops
Deeplabv3 [8]	ResNet50	60.4	168.4
Deeplabv3 [8]	ResNet101	61.5	264.1
+FPN	ResNet50	62.3	183.4
+PF decoder	ResNet50	65.6	185.2
CCNet [20]	ResNet50	58.3	206.5
+FPN	ResNet50	60.2	220.8
+PF decoder	ResNet50	65.3	223.2

(g) Application on Other Architectures.

Table 2: **Ablation studies.** We first verify the effect of each module and comparison results in the first row. Then we verify several design choices and generality of our module in the second row. Best view it in color and zoom in.

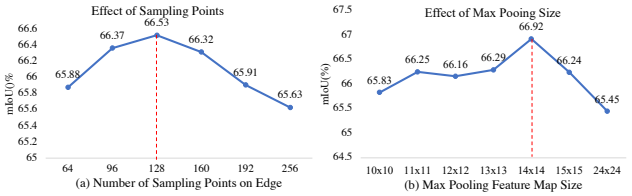


Figure 3: Ablation studies on the number of sampled point for both point flows. Best view it in color and zoom in.

Method	mIoU	F1(12px)	F(9px)	F1(5px)	F1(3px)
baseline+PPM	63.8	88.2	86.2	85.6	84.3
+salient point flow:	64.8	88.9	88.1	87.0	85.4
+boundary point flow	66.2	93.2	91.2	89.0	88.4
+both	66.9	94.2	93.2	90.2	89.0
direct prediction	65.7	89.6	87.5	86.4	85.8
subtraction prediction	66.2	93.2	91.2	89.0	88.4

Table 3: Ablation study on semantic boundaries where we adopt 4 different thresholds for evaluation.

pooling based methods work the best among them. Tab. 2(e) shows the propagation direction of PFM. Adding bottom up fusing leads to bad results mainly because more background context is introduced into the head which verifies our motivation of flowing semantics into the bottom. Tab. 2 (f) shows the effect results on edge prediction. Our subtraction based prediction has better results mainly due to better boundary prediction. This is also verified in Tab. 3.

Ablation on Number of Sampled Points: We first verify the best number of sampled points on boundary point flow in Fig. 3(a) by increasing the number of sampled pixels where we find the best number is 128. Sampling more points leads to inferior results which indicates missing background context is also important. Appending the boundary flow as the strong baseline, we explore the kernel size of salient point flow where we find the best kernel size 14×14(256 points in total) in Fig. 3(b). After selecting more points(24×24, 576 points in total), the performance drops a lot since the imbalance problems exist. This verifies the same conclusion that the dense affinity leads to bad results.

Application on Various Methods: Our PFM can be easily adopted into several existing networks by extending PFNet decoder(shown in Fig. 2(c) yellow box) after their heads. More details can be referred to supplementary. In Tab. 2(g), we verify two work including Deeplabv3[8] and CCNet [20] where we obtain significant gains over these baselines. This proves the generalization of our methods. Our method outperforms ResNet101-based models which indicates the improvement is not obtained by extra parameters introduced by PFM.

Effectiveness on Segmentation Boundaries: We further verify the boundary improvements using F1-score metric [37] with different pixel thresholds in Tab. 3. Appending boundary point flow leads to more significant improvements than salient point flow due to the explicit supervision and propagation on boundary pixels. Adopting both flows leads to the best results and it indicates the complemented property of our approach. Moreover, as shown in the last row of Tab. 3, our subtraction based edge prediction results better than direct prediction where it has better mask boundary. We include boundary prediction results in supplementary.

Balanced foreground-background Points: We analyze the ratio of sampled points on fore-ground parts over total sampled points by adding all three PFMs using validation images. Compared with baseline unbalanced points with 2.89% computed by ground truth mask, our method improves the ratio on foreground to 7.83% during the in-

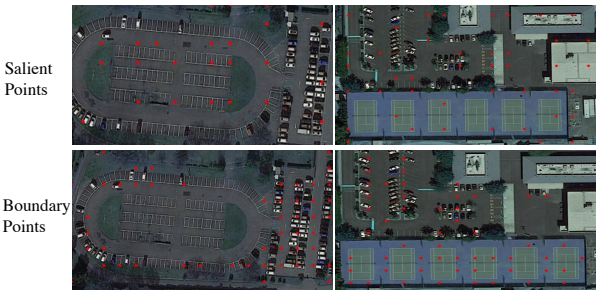


Figure 4: Visualization of sampled point for both point flows. Top: Salient Flow points. Bottom: Boundary Flow points. Best view it on screen.

ference which resolves the problems of imbalanced points aggregation. We use the best model found in ablation part.

Visualization of Sampled Points: In Fig. 3, we show several visual examples on sampled points on the original images. The first row gives the salient point results while the second row shows the boundary point results. We visualize the points from the PFM in the last stage. As shown in Fig. 4, the salient points uniformly locate around the foreground objects and several of them are on the background sparsely. The boundary points are mainly on the boundary of large foreground objects and the inner regions on small objects because the downsampled feature representation makes it hard to predict small object boundaries. More visual examples can be found in supplementary.

Benchmarking recent works on aerial images datasets: Recent work FarSeg [60] reports results of several segmentation methods [59, 22, 8] on iSAID datasets. We extend more representative work [20, 16, 26, 23] on iSAID, Vaihingen and Postdam datasets under the same experiment setting. Note that, for all methods, we use ResNet50 as backbone for fair comparison except for HRNet [42]. The work [36] also reports results on Vaihingen and Postdam using weak VGG-backbone [39]. Due to the lack of comparison with recent work, we re-implement this method using ResNet50 backbone and trained on larger cropped images and report mIoU as metric.[2] All the methods use the single scale inference on cropped images for testing.

Comparison with the state-of-the-arts on iSAID: We first benchmark more results on iSAID dataset in Tab. 4 and then compare our PFNet with previous work. Our PFNet achieves the state-of-the-art result among all previous work by a large margin. Our method outperforms previous state-of-the-art FarSeg [60] by 3.2%.

Experiments on Vaihingen and Potsdam: Rather than the previous work [36] cropping the images into small patches, we adopt large patches as the iSAID dataset and use more validation images for testing. That makes the segmentation

[2]We will opensource all the models and code for further research.

Method	Backbone	mIoU	OS
DenseASPP [48]	RenNet50	57.3	8
Deeplabv3 [8]	ResNet50	60.4	8
Deeplabv3+ [9]	ResNet50	61.2	8
RefineNet [29]	ResNet50	60.2	32
PSPNet [59]	ResNet50	60.3	8
OCNet-(ASP-OC) [52]	ResNet50	40.2	8
EMANet [26]	ResNet50	55.4	8
CCNet [20]	ResNet50	58.3	8
EncodingNet [56]	ResNet50	58.9	8
SemanticFPN [22]	ResNet50	62.1	32
UPerNet [22]	ResNet50	63.8	32
HRNet[46]	HRNetW18	61.5	4
SFNet[25]	ResNet50	64.3	32
GSCNN[40]	ResNe50	63.4	8
RANet[36]	ResNet50	62.1	8
FarSeg [60]	ResNet50	63.7	32
PFNet	ResNet50	**66.9**	32

Table 4: Comparison with the state-of-the-art results on iSAID dataset.

more challenging. The details of train and validation splitting can be found in the supplementary. For the Vaihingen dataset, we preprocess the images by cropping into 768× 768 patches. We adopt the same training setting with iSAID dataset except for 200 epochs and larger learning rate with 0.01. For the experiments on the Potsdam dataset, the images are cropped into 896×896 patches. The total training epoch is set to 80 with the initial learning rate of 0.01. As shown in Tab. 5, we benchmark recent segmentation methods with two metrics including mIoU and mean-F_1. Our PFNet achieves state-of-the-art results on two benchmarks.

Efficiency Comparison: In Fig. 5, we further benchmark the speed and parameters of our methods on above datasets. Compared with previous work, PFNet achieves the best speed and accuracy trade-off on those three benchmarks with fewer parameters without bells and whistles. Note that PFNet can also run in real-time setting and also achieves a significant margin compared with previous real-time methods[22, 25, 60].

Visual Results Comparison: In Fig. 6, we compare our method results with several state-of-the-art methods [9, 23, 22] on the iSAID validation set. Our PFNet has better segmentation results on handling false positives of small objects and has fine-grain object mask boundaries.

4.2. Results on general segmentation benchmarks:

We further verify our approach on general segmentation benchmarks including Cityscapes [12], ADE-20k [61] and BDD [49] for only verification purpose. We only report the results due to the limited space. More implementation details and visual results can be found in the supplementary file. We train both our baseline model and PFnet model on train datasets and report results on validation datasets under the same setting.

Comparison with the Baseline Methods: As shown in the last two rows of Tab. 6, our method improves the baseline model on various datasets about 1% mIoU with fewer pa-

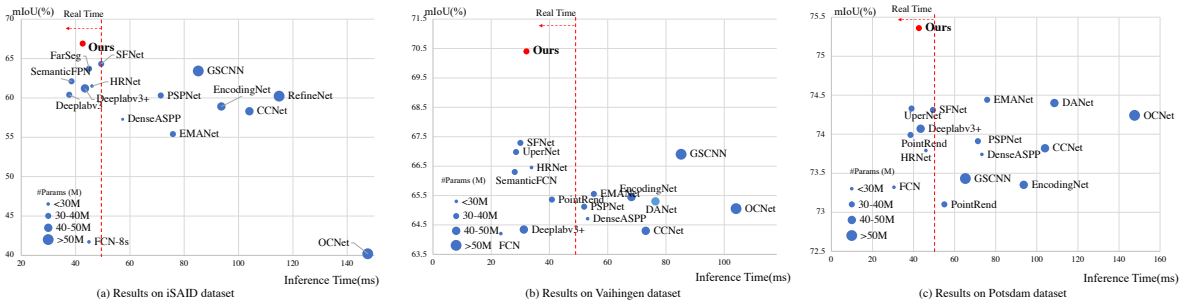


Figure 5: Speed (Inference Time) versus Accuracy (mIoU) on three aerial segmentation datasets. The radius of circles represents the number of parameters. All the methods are tested with one V-100 GPU card for fair comparison. Our PFNet achieves the best speed and accuracy trade-off on three benchmark. Real time is within 50ms. Best view it on screen and Zoom in.

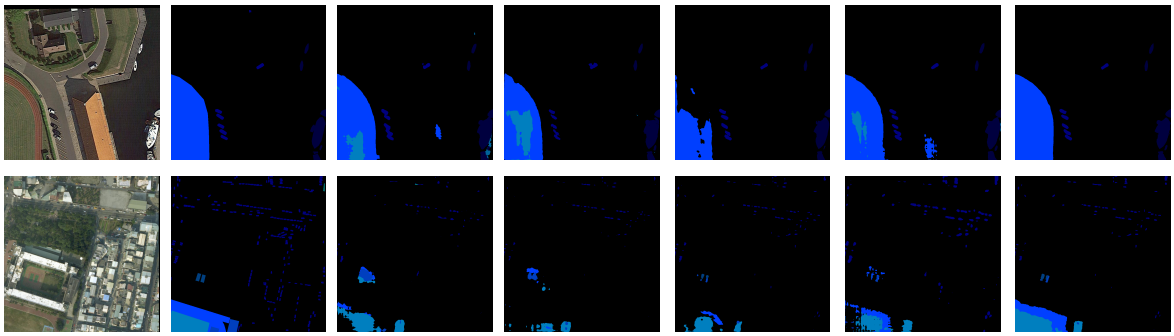


Figure 6: Visual results on iSAID validation set. Compared with previous works, our method obtains better segmentation results. Best view on the screen and zoom in. More visual results can be found in supplementary.

Method	mIoU	mean-F_1	mIoU	mean-F_1
PSPNet [59]	65.1	76.8	73.9	83.9
FCN [32]	64.2	75.9	73.1	83.1
OCnet(ASP-OC) [52]	65.7	77.4	74.2	84.1
Deeplabv3+ [9]	64.3	76.0	74.1	83.9
DAnet [16]	65.3	77.1	74.0	83.9
CCnet[20]	64.3	75.9	73.8	83.8
SemanticFPN [22]	66.3	77.6	74.3	84.0
UPerNet [46]	66.9	78.7	74.3	84.0
PointRend [23]	65.9	78.1	72.0	82.7
HRNet-W18 [42]	66.9	78.2	73.4	83.4
GSCNN [40]	67.7	79.5	73.4	84.1
SFNet [25]	67.6	78.6	74.3	84.0
EMANet [26]	65.6	77.7	72.9	83.1
RANet [36]	66.1	78.2	73.8	83.9
EncodingNet [56]	65.5	77.4	73.4	83.5
Denseaspp [48]	64.7	76.4	73.9	83.9
PFNet	**70.4**	**81.9**	**75.4**	**84.8**

Table 5: Comparison with the state-of-the-art results on Vahihigen(left) and Potsdam(right) datasets.

rameters and GFlops increase. Compared with the previous work [59, 52, 9], our method achieves better results with much less computation cost.

5. Conclusion

In this paper, we propose PointFlow Module to solve both imbalanced foreground-background objects and semantic gaps between feature pyramids problems for aerial image segmentation. We design a novel Dual Point Matcher to sampled the matched points from salient areas and boundaries accordingly. Extensive experiments have shown that our PF module can improve various baselines significantly on aerial benchmark. Based on the FPN framework, we build an efficient PFNet which achieves the best speed and accuracy trade-off on three public aerial benchmarks. Further experiments on three general segmentation datasets also prove the generality of our method.

Method	Cityscapes	ADE20k	BDD	Param(M)	GFlops(G)
PSPNet [59]	78.0	41.3	61.3	31.1	120.4
OCNet [52]	79.2	41.8	62.1	64.7	290.4
Deeplabv3+ [9]	79.4	42.0	61.0	40.5	189.8
baseline +PPM	78.8	40.9	61.1	32.9	83.1
Our PFnet	80.3	42.4	62.7	33.0	85.8

Table 6: Experiment results on general datasets including Cityscapes, ADE20k, BDD validation datasets. All the methods are trained under the same training setting and the results are reported with single scale inputs. The GFlops is calculated with 512×512 as input. All the methods use the ResNet50 backbone.

References

[1] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid methods in image processing. *RCA engineer*, 29(6), 1984. 4

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *PAMI*, 2017. 2

[3] F. Bastani, S. He, S. Abbar, M. Alizadeh, H. Balakrishnan, S. Chawla, S. Madden, and D. DeWitt. Roadtracer: Automatic extraction of road networks from aerial images. In *CVPR*, pages 4720–4728, 2018. 2

[4] P. Bilinski and V. Prisacariu. Dense decoder shortcut connections for single-pass semantic segmentation. In *CVPR*, 2018. 2

[5] P. J. Burt. Fast filter transform for image processing. *Computer graphics and image processing*, 16(1):20–51, 1981. 4

[6] H. Caesar, J. Uijlings, and V. Ferrari. Coco-stuff: Thing and stuff classes in context. In *CVPR*, 2018. 1

[7] 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. *PAMI*, 2018. 2

[8] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint*, 2017. 1, 2, 5, 6, 7

[9] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 2, 7, 8

[10] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, and Y. Kalantidis. Graph-based global reasoning networks. In *CVPR*, 2019. 2

[11] M. T. Chiu, X. Xu, Y. Wei, Z. Huang, A. G. Schwing, R. Brunner, H. Khachatrian, H. Karapetyan, I. Dozier, G. Rose, et al. Agriculture-vision: A large aerial image database for agricultural pattern analysis. In *CVPR*, 2020. 2

[12] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1, 2, 7

[13] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In *ICCV*, 2017. 5, 6

[14] M. Dickenson and L. Gueguen. Rotated rectangles for symbolized building footprint extraction. In *CVPR Workshops*, pages 225–228, 2018. 2

[15] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *CVPR*, 2018. 3

[16] J. Fu, J. Liu, H. Tian, Z. Fang, and H. Lu. Dual attention network for scene segmentation. In *CVPR*, 2019. 1, 3, 7, 8

[17] G. Ghiasi, T.-Y. Lin, and Q. V. Le. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *CVPR*, 2019. 3

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[19] B. Huang, B. Zhao, and Y. Song. Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. *Remote Sensing of Environment*, 214:73–86, 2018. 2

[20] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 2, 6, 7, 8

[21] P. Kaiser, J. D. Wegner, A. Lucchi, M. Jaggi, T. Hofmann, and K. Schindler. Learning aerial image segmentation from online maps. *IEEE Transactions on Geoscience and Remote Sensing*, 55(11):6054–6068, 2017. 2

[22] A. Kirillov, R. Girshick, K. He, and P. Dollár. Panoptic feature pyramid networks. In *CVPR*, pages 6399–6408, 2019. 2, 3, 5, 7, 8

[23] A. Kirillov, Y. Wu, K. He, and R. Girshick. PointRend: Image segmentation as rendering. In *CVPR*, 2020. 1, 2, 7, 8

[24] X. Li, Z. Houlong, H. Lei, T. Yunhai, and Y. Kuiyuan. Gff: Gated fully fusion for semantic segmentation. In *AAAI*, 2020. 3

[25] X. Li, A. You, Z. Zhu, H. Zhao, M. Yang, K. Yang, and Y. Tong. Semantic flow for fast and accurate scene parsing. *ECCV*, 2020. 3, 5, 6, 7, 8

[26] X. Li, Z. Zhong, J. Wu, Y. Yang, Z. Lin, and H. Liu. Expectation-maximization attention networks for semantic segmentation. In *ICCV*, 2019. 2, 7, 8

[27] Y. Li and A. Gupta. Beyond grids: Learning graph representations for visual recognition. In *NeurIPS*. 2018. 2

[28] J. Liang, N. Homayounfar, W.-C. Ma, S. Wang, and R. Urtasun. Convolutional recurrent network for road boundary extraction. In *CVPR*, pages 9512–9521, 2019. 2

[29] G. Lin, A. Milan, C. Shen, and I. D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017. 7

[30] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 2, 3

[31] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *CVPR*, pages 8759–8768, 2018. 3

[32] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2, 3, 8

[33] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia. Land cover mapping at very high resolution with rotation equivariant cnns: Towards small yet accurate models. *ISPRS journal of photogrammetry and remote sensing*, 145:96–107, 2018. 2

[34] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla. Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135:158–172, 2018. 2

[35] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014. 1

[36] L. Mou, Y. Hua, and X. X. Zhu. A relation-augmented fully convolutional network for semantic segmentation in aerial scenes. In *CVPR*, pages 12416–12425, 2019. 2, 7, 8

[37] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 6

[38] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *MICCAI*, 2015. 2

[39] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015. 7

[40] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. *ICCV*, 2019. 1, 2, 7, 8

[41] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3

[42] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2019. 2, 7, 8

[43] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *CVPR*, 2018. 2, 3

[44] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *CVPR*, 2018. 2, 6

[45] S. Waqas Zamir, A. Arora, A. Gupta, S. Khan, G. Sun, F. Shahbaz Khan, F. Zhu, L. Shao, G.-S. Xia, and X. Bai. isaid: A large-scale dataset for instance segmentation in aerial images. In *ICCV Workshops*, 2019. 1, 2, 5

[46] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018. 2, 7, 8

[47] 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 Sensing*, 10(1):144, 2018. 2

[48] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang. Denseaspp for semantic segmentation in street scenes. In *CVPR*, 2018. 2, 7, 8

[49] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 2, 7

[50] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *ICLR*, 2016. 2, 3

[51] Y. Yuan, X. Chen, and J. Wang. Object-contextual representations for semantic segmentation. *ECCV*, 2020. 2

[52] Y. Yuan and J. Wang. Ocnet: Object context network for scene parsing. *arXiv preprint*, 2018. 1, 2, 3, 7, 8

[53] Y. Yuan, J. Xie, X. Chen, and J. Wang. Segfix: Model-agnostic boundary refinement for segmentation. In *ECCV*, 2020. 2

[54] D. Zhang, H. Zhang, J. Tang, M. Wang, X. Hua, and Q. Sun. Feature pyramid transformer. *ECCV*, 2020. 3, 5

[55] F. Zhang, Y. Chen, Z. Li, Z. Hong, J. Liu, F. Ma, J. Han, and E. Ding. Acfnet: Attentional class feature network for semantic segmentation. In *ICCV*, 2019. 2

[56] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal. Context encoding for semantic segmentation. In *CVPR*, 2018. 7, 8

[57] H. Zhang, H. Zhang, C. Wang, and J. Xie. Co-occurrent features in semantic segmentation. In *CVPR*, 2019. 2

[58] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun. Exfuse: Enhancing feature fusion for semantic segmentation. In *ECCV*, 2018. 2

[59] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017. 1, 2, 3, 5, 7, 8

[60] Z. Zheng, Y. Zhong, J. Wang, and A. Ma. Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery. In *CVPR*, pages 4096–4105, 2020. 1, 2, 5, 7

[61] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ADE20K dataset. *arXiv preprint*, 2016. 1, 2, 7

[62] X. Zhu, H. Hu, S. Lin, and J. Dai. Deformable convnets v2: More deformable, better results. In *CVPR*, 2019. 5, 6

[63] Z. Zhu, M. Xu, S. Bai, T. Huang, and X. Bai. Asymmetric non-local neural networks for semantic segmentation. In *ICCV*, 2019. 2