# Learning Implicit Feature Alignment Function for Semantic Segmentation

Hanzhe Hu[1]*, Yinbo Chen[2]*, Jiarui Xu[2], Shubhankar Borse[3], Hong Cai[3], Fatih Porikli[3], and Xiaolong Wang[2]

[1] Peking University
[2] University of California, San Diego
[3] Qualcomm AI Research

**Abstract.** Integrating high-level context information with low-level details is of central importance in semantic segmentation. Towards this end, most existing segmentation models apply bilinear up-sampling and convolutions to feature maps of different scales, and then align them at the same resolution. However, bilinear up-sampling blurs the precise information learned in these feature maps and convolutions incur extra computation costs. To address these issues, we propose the Implicit Feature Alignment function (IFA). Our method is inspired by the rapidly expanding topic of implicit neural representations, where coordinate-based neural networks are used to designate fields of signals. In IFA, feature vectors are viewed as representing a 2D field of information. Given a query coordinate, nearby feature vectors with their relative coordinates are taken from the multi-level feature maps and then fed into an MLP to generate the corresponding output. As such, IFA implicitly aligns the feature maps at different levels and is capable of producing segmentation maps in arbitrary resolutions. We demonstrate the efficacy of IFA on multiple datasets, including Cityscapes, PASCAL Context, and ADE20K. Our method can be combined with improvement on various architectures, and it achieves state-of-the-art computation-accuracy trade-off on common benchmarks. Code is available at https://github.com/hzhupku/IFA.

**Keywords:** Semantic Segmentation, Implicit Neural Representation, Feature Alignment

## 1 Introduction

Semantic Segmentation is one of the most fundamental and challenging tasks in computer vision. It aims at classifying each pixel in the image into a semantic category. Its wide applications include scene understanding, image editing, augmented reality, and autonomous driving. Most of these applications not only require the segmentation model to predict high resolution and high-quality masks but also demand high efficiency in speed and memory cost, especially when running online or on edge devices.

---

* Equal contribution.
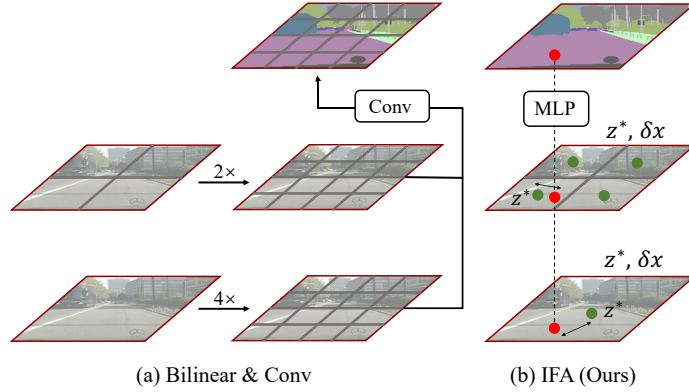
Fig. 1: **Implicit Feature Alignment function (IFA).** (a) Prior works transform feature maps to the same resolution for alignment, where bilinear upsampling blurs the precise information and convolutions can be inefficient. (b) IFA decodes directly from the original feature maps for arbitrary coordinates. An MLP takes as input the multi-level features around the query coordinate with their relative coordinates and outputs the aggregated information.

Most current approaches for semantic segmentation are built upon the Fully Convolutional Network [27]. At the heart of these approaches, is to integrate high-level context information with low-level details during segmentation. Empirically, deeper features with coarse resolution correspond to higher semantic information while shallow features in lower layers contain more local details. To aggregate different levels of information, state-of-the-art approaches such as Feature Pyramid Network (FPN) [20] and DeepLab V3+ [3] utilize bilinear upsampling followed by convolutions to align the low resolution deep features with the high resolution shallow features. However, the bilinear up-sampling can blur the precise context learned in deep features and the convolutions are not optimal for speed and memory efficiency, especially for high resolution segmentation where the resolution difference between high-level context and low-level details can be large.

To perform efficient and precise feature alignment, we will require a representation that can flexibly query a location in any resolution and output the corresponding feature values. This formulation corresponds to the implicit neural representation [32,35,39,18] proposed for high-quality 3D shape reconstruction, where a 3D object is represented as a neural network that maps a 3D coordinate to its occupancy in the object [28] or its signed distance to the object surface [32]. This idea is also migrated to the 2D domain [39,4], where neural functions are proposed as continuous image representations that allow the image to be decoded in an arbitrary resolution. Instead of decoding RGB values, can we use implicit neural representation to perform feature alignment?

In this paper, we propose a novel Implicit Feature Alignment function (IFA) to efficiently and precisely aggregate features from different levels for semantic

segmentation. By forwarding an image to a ConvNet, the features can be viewed as latent codes evenly distributed in spatial dimensions (shown as green dots in Figure 1). Intuitively, each latent code will represent a field of information. IFA will then decode the output segmentation map at every coordinate independently and parallelly. It takes as inputs the latent codes around the queried coordinate from different levels and the relative coordinates to the latent codes, then outputs the aggregated feature at the queried coordinate for classification, as illustrated in Figure 1. Take the FPN [20] model as an example, the original design is to use bilinear up-sampling and convolutions to align multi-layer features. IFA can be a replacement here to align and aggregate the multi-level features. Instead of bilinear up-sampling the features and aligning them in a fixed resolution, IFA allows the features to be learned as precisely representing continuous fields of information. The information in different levels are functions of continuous coordinates, which leads to a precise feature alignment in a resolution-free manner, and we can query the coordinate in arbitrary output resolutions with IFA for semantic segmentation.

We demonstrate the effectiveness of IFA on multiple semantic segmentation architectures and multiple datasets including Cityscapes [8], PASCAL Context [31] and ADE20K [59]. We replace feature alignment approaches with IFA in different methods, and IFA outperforms the original methods in all cases. IFA shows state-of-the-art computation-accuracy trade-off on all experimented datasets. For performing high resolution image segmentation, we also experiment with reducing the high-level feature map size while maintaining the low-level feature resolution, which improves efficiency and reduces memory cost. IFA has shown a much larger gain on aligning features with larger resolution differences. This not only shows the effectiveness of IFA on precise feature alignment, but also reveals its potential on efficient high resolution semantic segmentation.

To sum up, our contributions are summarized as follows:

- We propose a novel implicit neural representation IFA for efficient and precise alignment among multi-level feature maps for semantic segmentation.
- Our IFA can be incorporated with multiple state-of-the-art semantic segmentation models and show improvement in all cases.
- We achieve state-of-the-art computation-accracy trade-off on benchmarks including Cityscapes, PASCAL Context and ADE20K.

## 2   Related Work

**Semantic Segmentation.**  With the success of deep neural networks [21,38,12], semantic segmentation has achieved great progress. Based on Fully Convolutional Network (FCN) [27], many works have been proposed. To produce high-resolution semantic segmentation map, spatial and semantic information are both indispensable. There are mainly two lines of research for learning the two kinds of information in semantic segmentation.

The first stream lies in that the final output of the network contains both spatial and semantic information. Many state-of-the-art methods follow this line

to design segmentation head to capture contextual information. From the local perspective, DeepLabV3 [2] employs multiple atrous convolutions with different dilation rates to capture contextual information, while PSPNet [56] utilizes pyramid pooling over sub-regions to harvest information. While from the global perspective, Wang *et al.* [45] apply the idea of self-attention from transformer [42] into vision problems and propose the non-local module where correlations between all pixels are calculated to guide the dense contextual information aggregation. Recently, Transformer based models [58,40,7,46] have achieved great progress in semantic segmentation, while they also suffer from heavy parameters and computation cost.

The other stream dissipates information along outputs of different layers of the network. Hence, the success relies on the feature alignment among the outputs. Our method focuses on this direction.

**Feature Alignment.** A common knowledge in semantic segmentation is that outputs from shallower network layers contain more low-level spatial details, while outputs from deeper network layers possess more high-level semantic information. How to effectively align those features has become a vital problem in this stream of study. Many methods focus on fusing multi-level feature maps for high-resolution spatial details and rich semantics. U-Net [34] adds several expanding paths to the contracting path to enable precise localization with the context. CARAFE [43] introduces a context-aware feature upsampling method, where features inside a predefined region centered at each location are reassembled via a weighted combination. Semantic FPN [20] applies FPN[26] structure to semantic segmentation where multi-level features are aligned by several upsampling stages consisted of convolution layers and bilinear up-sampling. On top of FPN, SFNet [24] proposes the flow alignment module to broadcast high-level context to high-resolution details. AlignSeg [17] explicitly learns the transformation offsets and adaptively aggregate contextual information for better alignment. However, aforementioned methods for alignment usually take up expensive computations and tend to be inefficient for real-time applications.

**Implicit Neural Representation.** In recent methods in 3D reconstruction, shape, object and scene can be represented by multi-layer perceptron (MLP) that maps coordinates to signals, known as implicit neural representations (INR) [32,10,29,30] since the parameters of the 3D representation are not explicitly encoded by point cloud, mesh or voxel. For example, DeepSDF [32] learns a set of continuous signed distance functions for shape representation. Later, NeRF [30] provides a more flexible way for synthesizing novel views of complex scenes.

Although implicit nerual representation has achieved great progress in 3D tasks, it is relatively under-explored for 2D tasks. [5] performs 2D shape generation from latent space for simple digits. [39] replaces ReLU with periodic activation functions inside MLP of implicit neural representation to model natural images in high quality. Recently, LIIF [4] applies implicit neural representation to model continuous image representation and UltraSR [47] further improves the accuracy by adding spatial encoding for implicit function on 2D images. CRM [36] performs image segmentation refinement by using implicit neural rep-
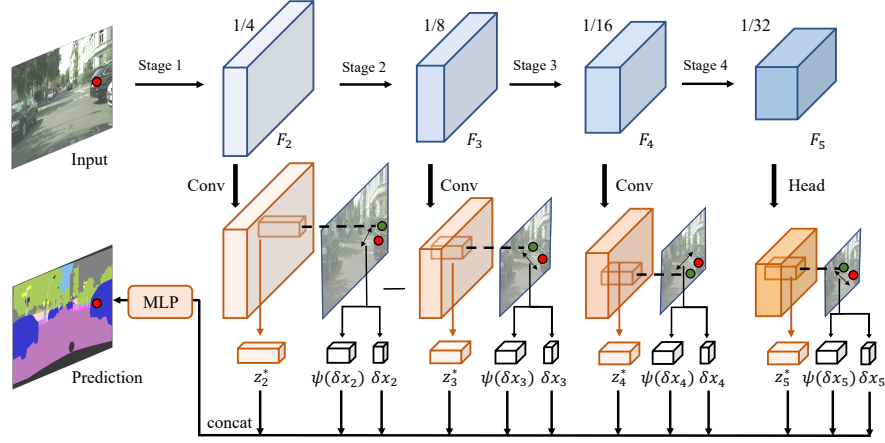
Fig. 2: **Overview of our proposed Implicit Feature Alignment function (IFA).** The general architecture consists of an encoder part (in blue) and a decoder part (in orange). IFA aligns multi-level feature maps from different stages of the encoder. Each feature map is projected to the same dimension via a convolution layer. We could also project the last feature with a segmentation head like ASPP [2]. We view the features in feature maps as latent codes evenly distributed in the 2D space. Given a query coordinate $x_q$, we first find its nearest latent codes $\{z_i^*\}_{i=2}^5$ for each feature map $i$ and use $x_i^*$ to denote the coordinate of $z_i^*$. We then concatenate these latent codes $\{z_i^*\}_{i=2}^5$ and relative coordinates $\{\delta x_i = (x_q - x_i^*)\}_{i=2}^5$, and pass the concatenated vector into an MLP that directly predicts the segmentation label of point $x_q$. The red point refers to the query coordinate $x_q$, while the green point denotes the nearest coordinate $x_i^*$ from $x_q$ on feature $F_i$.

resentations. Implicit PointRend [6] focuses on instance segmentation with point supervision, where implicit function is used to generate different parameters of the point head for each object. Different from these works, we focus on utilizing INR to perform implicit alignment of multi-level features.

## 3  Method

In this section, we first introduce the preliminary knowledge about semantic segmentation and reveal the feature alignment within the structures in Section 3.1. Then we present an overview of the network architectures with the proposed Implicit Feature Alignment Function (IFA) as the alignment method in Section 3.2. Finally, the details of IFA are introduced in Section 3.3.

### 3.1   Preliminary

We will first revisit the basic background of semantic segmentation. Given a RGB image $I \in \mathbb{R}^{3 \times H \times W}$, the network aims to produce the segmentation prediction $P \in \mathbb{R}^{N \times H \times W}$, where $H$, $W$ denote the height and width of the input image and $N$ denotes the number of classes. An encoder-decoder paradigm is often adopted, where the encoder applies several down-sampling operations and the decoder employs up-sampling modules to recover the original size. To capture rich information, state-of-the-art methods propose to aggregate features from different levels to capture both local details and high-level semantic information. Following the setting in FPN [20], different levels of features $\{F_i\}_{i=2}^5$ are extracted from different network stages, where a larger $i$ denotes a deeper stage. $F_i \in \mathbb{R}^{C_i \times H_i \times W_i}$ is a $C_i$ dimensional feature map defined on a spatial grid with size of $H_i \times W_i$ ($H_i = \frac{H}{2^i}$, $W_i = \frac{W}{2^i}$). To compensate the information loss during consecutive down-sampling operations, FPN aims to fuse different levels of feature for better representations. Originally proposed for object detection [26], FPN fuses high-level feature maps with low-level features in a top-down strategy step by step. At each step, FPN fuses high-level feature map with low-level one through several $2\times$ bilinear up-sampling operations with convolution layers.

### 3.2   Overall Framework

**Network Architecture.** Figure 2 demonstrates an overview of our network architecture, the general architecture can be described as an encoder part (in blue) and a decoder part (in orange). For a given input image, a bottom-up encoder will encode the image to the feature maps in different levels. A typical example of the encoder is ResNet [12], which generates four feature maps $\{F_i\}_{i=2}^5$ from different stages. For the decoder part, while FPN builds a top-down pathway with bilinear up-sampling, our method instead takes features $\{F_i\}_{i=2}^5$ from the encoder as inputs and decodes the output at every coordinate independently and parallelly, forming a point-independent prediction manner.

**Supervision.** Following standard practice in previous state-of-the-art works [56,53,14,24,23,13,15], we add the auxiliary supervision for improving the performance, as well as making the network easier to optimize. Specifically, the output of the third stage of the backbone is further fed into an auxiliary layer to produce an auxiliary prediction, which is supervised with the auxiliary loss. We apply standard cross entropy loss to supervise the auxiliary output and employ OHEM loss [37] to supervise the main output.

### 3.3   Implicit Feature Alignment Function

In this subsection, we will first introduce an implicit feature function defined on a single feature map. Then we present the details of the position encoding used in our method. Finally, we extend the implicit feature function to the IFA for multi-level features.

**Implicit Feature Function.** One of the main challenges of aggregating information from multi-level feature maps comes from their different resolutions. Up-sampling modules are usually necessary to align them within the same resolution. Our key idea is to define continuous feature maps (i.e. fields of features), which can be decoded at arbitrary coordinates, so that they are aligned in a continuous field and no up-sampling is required.

To define a continuous feature map $M$, we introduce the implicit feature function. It is inspired by recent works of implicit neural representations [18,4] for 3D reconstruction and image super-resolution. Implicit feature function defines a decoding function $f_\theta$ (typically an MLP) over a discrete feature map to get the continuous feature map $M$. Given the discrete feature map, feature vectors are viewed as latent codes evenly distributed in the 2D space, each of them is assigned with a 2D coordinate. The feature value of $M$ at $x_q$ is defined by

$$M(x_q) = f_\theta(z^*, x_q - x^*), \tag{1}$$

where $z^*$ is the nearest latent code from $x_q$ and $x^*$ is the coordinate of latent code $z^*$. To summarize, with the decoding function $f_\theta$, we can define a continuous feature map $M$ over a discrete feature map. In practice, $f_\theta$ is jointly learned with the feature encoder so that the features are learned to precisely represent continuous fields of information.

**Position Encoding.** As discussed in previous works [30,47], although neural networks can be treated as universal function approximators, the learning power gets limited when directly operated on $xy$ coordinates due to its inferiority at representing high-frequency signals. This is consistent with the discovery of a recent work [33] that neural networks are biased towards low-frequency signals and are insensitive to high-frequency signals. Therefore, instead of directly feeding the coordinates to the network, we first encode them with the position encoding function. Formally, the encoding function we use is:

$$\psi(x) = (\sin(\omega_1 x), \cos(\omega_1 x), ..., \\ \sin(\omega_L x), \cos(\omega_L x)), \tag{2}$$

where the frequency $\omega_l$ are initialized as $\omega_l = 2e^l, l \in \{1, ..., L\}$ and can be fine-tuned during training, and the encoding function expands the 2D coordinates into the $2L$-dimensional encoding. We also perform experiments on position encoding functions using only sin or cos function, and Eq. 2 performs the best (see Section 4.3 for details). The final definition of implicit feature function is:

$$M(x_q) = f_\theta(z^*, \psi(x_q - x^*), x_q - x^*), \tag{3}$$

where the relative coordinates together with their position encodings are fed into the implicit function.

**Feature Alignment.** A direct way to perform feature alignment is to define implicit feature functions and convert each feature map in different levels to a continuous feature map, so that their features can be queried at arbitrary

coordinates for alignment. In this subsection, we show that this can be simplified to a more efficient method.

Take aligning the feature maps $\{F_i\}_{i=2}^5$ as an example, we extend the implicit feature function to implicit feature alignment function (IFA), which directly defines a continuous feature map $M$ over multi-level discrete feature maps in different resolutions. Specifically, we define the the value of $M$ at $x_q$ as

$$M(x_q) = f_\theta\big(\{z_i^*\}_{i=2}^5, \{\psi_i(\delta x_i), \delta x_i\}_{i=2}^5\big), \qquad (4)$$
$$\delta x_i = x_q - x_i^*,$$

where $i$ denotes the index of feature level, $z_i^*$ is the nearest latent code from $x_q$ at level $i$ and $x_i^*$ is the coordinate of $z_i^*$. We implement $f_\theta$ as concatenating all its input vectors and passing it into an MLP. Intuitively, each latent code still represents a field of feature that can be decoded by relative coordinate, $f_\theta$ can decode the field for each level and model the interaction across different levels at the same time.

The alignment among features produced from different stages is also shown in the lower half of Figure 2. For a query coordinate, we obtain the nearest latent codes from each level of features, noted as $\{z_i^*\}_{i=2}^5$, relative coordinates $\delta x_i$ together with the corresponding encoded ones, noted as $\{\psi(\delta x_i)\}_{i=2}^5$. After concatenating the latent codes, relative coordinates and the encoded relative coordinates, we feed them into the MLP of the decoding function $f_\theta$. Given a output resolution, we decode the segmentation map by querying every pixel location independently and parallelly. Therefore, IFA aligns the features in a resolution-free manner and allows decoding to arbitrary resolutions.

Besides FPN, the proposed IFA can be easily applied into other semantic segmentation models that require multi-level feature aggregation, such as DeepLab V3+ [3] and HRNet [44].

## 4    Experiments

### 4.1    Datasets and Evaluation Metrics

**Cityscapes.** The Cityscapes dataset [8] is tasked for urban scene understanding, which contains 30 classes and only 19 classes of them are used for scene parsing evaluation. The dataset contains 5000 finely annotated images and 20000 coarsely annotated images. The size of the images is 2048×1024 pixels. The finely annotated 5,000 images are split into 2975, 500 and 1525 images for training, validation and testing respectively. We only use finely annotated part in our experiments.
**PASCAL Context.** The PASCAL Context is a dataset [31] is a challenging scene parsing dataset which contains 59 semantic classes and 1 background class. The training set and test set consist of 4,998 and 5,105 images respectively.
**ADE20K.** The ADE20K dataset [59] is a large scale scene parsing benchmark which contains dense labels of 150 stuff/object categories. The annotated images are divided into 20K, 2K and 3K for training, validation and testing, respectively.

| Method | mIoU(%) | #Params | GFLOPs |
|---|---|---|---|
| Bilinear Up-sampling | 76.52 | 27.7M | 183.4 |
| Nearest Neighbor | 76.32 | 27.7M | 183.4 |
| Deconvolution | 72.89 | 29.5M | 304.4 |
| Up-sampling Module | 77.19 | 31.0M | 219.1 |
| CARAFE [43] | 76.80 | 29.0M | 190.5 |
| AlignSeg [17] | 78.50 | 49.7M | 348.6 |
| IFA (Ours) | 78.02 | 27.8M | 186.9 |

Table 1: Performance comparisons of different aligning methods within the FPN structure on Cityscapes `val` set. GFLOPs calculations adopt $1024 \times 1024$ images as input.

| Method | mIoU(%) |
|---|---|
| DeepLab V3+ | 76.69 |
| DeepLab V3+ (IFA) | **77.57** |
| PSPNet | 73.64 |
| PSPNet (IFA) | **74.42** |
| HRNet-W18 | 77.60 |
| HRNet-W18 (IFA) | **78.10** |
| HRNet-W48-OCR | 85.80 |
| HRNet-W48-OCR (IFA) | **86.10** |

Table 2: Performance of IFA on different segmentation models on Cityscapes `val` set.

**Evaluation Metric.** The mean of class-wise Intersection over Union (mIoU) is used as the evaluation metric. Number of float-point operations (FLOPs) and number of parameters are also adopted for efficiency evaluations.

## 4.2   Implementation Details

We use ResNet pretrained on ImageNet [21] as our backbone. For Cityscapes dataset, we use stochastic gradient descent (SGD) optimizer with initial learning rate 0.01, weight decay 0.0005 and momentum 0.9. We adopt the 'poly' learning rate policy, where the initial learning rate is multiplied by $(1 - \frac{\text{iter}}{\text{max\_iter}})^{0.9}$. We adopt the crop size as $769 \times 769$, batch size as 16 and training iterations as 18k. For PASCAL Context dataset, we set the initial learning rate as 0.001, weight decay as 0.0001, crop size as $513 \times 513$, batch size as 16 and training iterations as 30K. For ADE20K dataset, we set the initial learning rate as 0.004, weight decay as 0.0001, crop size as $480 \times 480$, batch size as 16 and training iterations as 150K.

### 4.3   Results and Ablations

In this subsection, we conduct extensive experiments on the `val` set of Cityscapes dataset with different settings for our proposed IFA. For all the experiments in this subsection, we use ResNet-50 as the backbone and down-sampling rate as 32 if not specified. All compared methods are evaluated by single-scale inference.

**Aligning Method.**   We first compare IFA against commonly used aligning methods, i.e. bilinear up-sampling, nearest up-sampling, deconvolution and up-sampling module (bilinear+convolution), and state-of-the-art methods including CARAFE [43] and AlignSeg [17]. In particular, we use FPN as the decoder, where the original aligning method is the up-sampling module. We then replace it with other aligning methods. As shown in Table 1, our proposed IFA performs the best over other baseline methods. While up-sampling module also achieves high performance, its overhead is much higher than the proposed IFA. IFA achieves better results than up-sampling module with 85% of its computation. Moreover, although AlginSeg obtains slightly better results, it takes up almost twice as many parameters and computation cost as ours. Hence, IFA achieves a better trade-off between computational cost and accuracy.

**Extension to Other Models.**   Since our proposed IFA targets at aligning features from different levels, we can directly apply it into other segmentation models involving feature alignment such as DeepLab V3+ [3], PSPNet [56] and HRNet [44]. In particular, DeepLab V3+ aggregates low-level feature $F_2$ and high-level feature $F_5$ by simple bilinear up-sampling, which can be replaced by IFA. PSPNet aggregate features of different scales produced with different pooling strides by bilinear up-sampling as well. And HRNet also aggregates features of four different scales by bilinear up-sampling. Hence, we plug IFA into these models to replace bilinear up-sampling and perform alignment. The results are presented in Table 2. IFA improves DeepLab V3+ by 0.9%, PSPNet by 0.8%, and HRNet-W18 by 0.5%. Furthermore, IFA also boost the performance of HRNet-W48-OCR [52] by 0.3%, indicating the strong generalization ability of IFA for different segmentation models.

**Resolution Difference.** Commonly in FPN based methods, the largest scale difference between two feature maps is 8 times (between $F_2$ and $F_5$). The larger the scale difference is, the harder it is to align feature maps. Moreover, as the improvement of high-resolution image collection tools, we will obtain a large amount of super high resolution data for training and testing. However, with higher-resolution images as input, current methods could be unable to fit into training machines due to limited memory capacity.

   To demonstrate that our proposed IFA can better align feature maps with large scale difference, we alternate the stride of FPN and DeepLab V3+ models and apply IFA. In particular, we add average pooling operation after the first stage of the backbone to further downsample the feature maps, which simultaneously increase both the down-sampling rate of the network and the scale difference between the highest resolution feature ($F_2$) and the lowest resolution feature ($F_5$). The results of experiments on FPN is shown in Table 3. As the scale difference gets larger, the performance gain over the baseline becomes larger as

| Method | Stride | Diff | IFA | mIoU(%) | Gain(%) |
|--------|--------|------|-----|---------|---------|
| FPN | 32 | 8 | ✓ | 77.19<br>**78.02** | 0.9 |
| | 64 | 16 | ✓ | 76.52<br>**77.69** | 1.1 |
| | 128 | 32 | ✓ | 74.88<br>**76.40** | 1.6 |
| DeepLab<br>V3+ | 32 | 8 | ✓ | 76.69<br>**77.57** | 0.9 |
| | 64 | 16 | ✓ | 75.20<br>**76.23** | 1.1 |
| | 128 | 32 | ✓ | 70.01<br>**72.18** | 2.1 |

Table 3: Effect of resolution difference on the feature maps of the FPN model. 'Stride' denotes the down-sampling rate of the network and 'Diff' denotes the scale different between $F_2$ and $F_5$. Results are reported on Cityscapes `val` set.
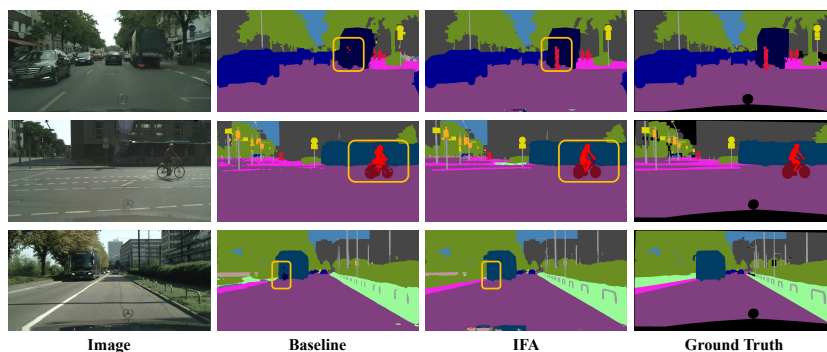


Fig. 3: Visualization results on Cityscapes `val` set. From left to right: input image, predictions made by the FPN baseline, predictions made by the FPN with the proposed IFA and groundtruth map. Yellow squares denote the challenging regions that can be resolved by our proposed IFA.

well, demonstrating the capability of IFA to align feature map with large scale difference. And the results of experiments on DeepLab V3+ is also shown in Table 3. Similar conclusion can be obtained.

**Visualizations of the effect of IFA.** We further provide comparisons of visualization results on `val` set of Cityscapes dataset in Figure 3. IFA considerably resolves category ambiguities within large objects and produces more precise boundaries of small objects, by effectively aggregating low-level and high-level

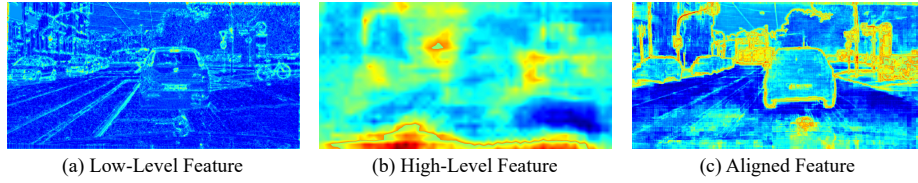(a) Low-Level Feature          (b) High-Level Feature          (c) Aligned Feature

Fig. 4: Visualizations of feature maps. (a) Feature map from the first stage of the encoder. (b) Feature map from the last stage of the encoder. (c) Aligned feature from our proposed IFA.

| Method | Backbone | mIoU(%) | #Params | GFLOPs |
|---|---|---|---|---|
| RefineNet [25] | Dilated ResNet-101 | 73.6 | - | - |
| PSPNet [56] | Dilated ResNet-101 | 78.4 | 68.1M | 1104.4 |
| AAF [19] | Dilated ResNet-101 | 79.1 | - | - |
| DFN [51] | Dilated ResNet-101 | 79.3 | 96.7M | 1185.6 |
| PSANet [57] | Dilated ResNet-101 | 80.1 | 89.2M | 1205.7 |
| DenseASPP [48] | DenseNet-161 | 80.6 | 39.9M | 640.1 |
| ANNet [61] | Dilated ResNet-101 | 81.3 | 66.5M | 1120.5 |
| CPNet [49] | ResNet-101 | 81.3 | | |
| CCNet [16] | Dilated ResNet-101 | 81.4 | 69.8M | 1190.0 |
| DANet [9] | Dilated ResNet-101 | 81.5 | 69.7M | 1335.9 |
| STLNet [60] | Dilated ResNet-101 | **82.3** | 81.39M | 535.9 |
| SETR [58] | ViT-Large | 81.1 | 318.3M | 2352.0 |
| SegFormer [46] | MiT-B5 | 82.2 | 84.7M | 730.5 |
| BiSeNet [50] | ResNet-18 | 77.7 | 15.6M | 130.2 |
| BiSeNet [50] | ResNet-101 | 78.9 | 54.2M | 255.1 |
| SFNet [24] | ResNet-18 | 79.5 | 15.9M | 165.4 |
| SFNet [24] | ResNet-101 | 81.8 | 55.0M | 459.9 |
| IFA (Ours) | ResNet-18 | 79.3 | 12.3M | 93.3 |
| IFA* (Ours) | ResNet-18 | 79.8 | 16.8M | 98.0 |
| IFA (Ours) | ResNet-101 | 81.2 | 46.7M | 262.8 |
| IFA* (Ours) | ResNet-101 | **82.0** | 64.3M | 281.4 |

Table 4: Comparisons with state-of-art on the Cityscapes `test` set. 'IFA*' denotes IFA with ASPP module. All methods use multi-scale inference. The GFLOPs is calculated with a image size of $1024 \times 1024$.

feature maps. Hence, low-level spatial details and high-level semantic information can be precisely aligned to produce a more accurate prediction.

**Visualizations of the feature maps.** To better demonstrate the effect of IFA, we visualize feature maps from the first stage and final stage of the encoder and the decoded feature from IFA. Bright areas denote the existence of objects. The visualizations are generated by averaging the features along the channel dimension. As shown in Figure 4, IFA can effectively leverage spatial details from the low-level feature and semantic information from the high-level feature, thus output a comprehensive feature representation.

| Method | Backbone | mIoU(%) | GFLOPs |
|---|---|---|---|
| FCN-8s [27] | VGG-16 | 37.8 | - |
| DeepLab V2 [1] | D-ResNet-101 | 45.7 | - |
| RefineNet [25] | ResNet-152 | 47.3 | - |
| EncNet [54] | D-ResNet-101 | 51.7 | - |
| DANet [9] | D-ResNet-101 | 52.6 | 296.4 |
| ANNet [61] | D-ResNet-101 | 52.8 | 248.3 |
| EMANet [22] | D-ResNet-101 | 53.1 | 212.3 |
| SETR [58] | ViT-Large | **55.8** | 519.5 |
| SFNet [24] | ResNet-101 | 53.8 | 103.0 |
| IFA (Ours) | ResNet-101 | 53.0 | 59.1 |
| IFA* (Ours) | ResNet-101 | **53.8** | 63.5 |

Table 5: Comparisons with state-of-art on the PASCAL Context `test` set. The results are reported under 60 classes (w/o background). 'D-' denotes the dilated version of the backbone and 'IFA*' denotes IFA with ASPP module. All methods use multi-scale inference. GFLOPs is calculated with a image size of $480 \times 480$.

### 4.4    Comparisons with State-of-the-Arts

In this subsection, we compare our method with other state-of-the-art methods on three benchmark datasets including Cityscapes, PASCAL Context and ADE20K. Specifically, we choose ResNet-101 as the encoder, FPN as the decoder and replace the up-sampling module in FPN with our proposed IFA. Moreover, to further improve the performance, we add an ASPP [2] module at the end of the encoder, only performing contextual learning on the last feature map of the pyramid ($F_5$). Similar to other state-of-the-art methods, we also use the multi-scale and flipping strategies for testing to achieve better results. For convenience, we use IFA to represent FPN with IFA.

**Cityscapes.** We train the proposed method using both training and validation set of Cityscapes dataset and make the evaluation on the `test` set by submitting our test results to the official evaluation server. Model parameters and computation FLOPs are also listed for comparison. From Table 4, it can be observed that our proposed IFA achieves competitive performance on Cityscapes `test` set with less computation cost. In particular, our proposed IFA achieves competitive result (81.2) compared with SFNet (81.8) with only 57% of its computational overhead. And with ASPP module bringing extra 7% computation, our method (82.0) surpasses SFNet (81.8) while only requiring 61% of its computation. Moreover, although recent Transformer based model SegFormer achieves slightly better results (82.2) than ours (82.0), it takes extra 32% parameters and 160% computation cost. The results demonstrate that our method achieved a better computation-accuracy trade-off compared with other state-of-the-art methods. It's also worth noting that SegFormer backbone is pre-trained with stronger data augmentation from the recipe of DeiT [41], while our backbone is pre-trained with standard data augmentation.

| Method | Backbone | mIoU(%) | GFLOPs |
|---|---|---|---|
| PSPNet [56] | D-ResNet-101 | 43.29 | 280.3 |
| PSANet [57] | D-ResNet-101 | 43.77 | 305.4 |
| CFNet [55] | D-ResNet-101 | 44.89 | - |
| CCNet [16] | D-ResNet-101 | 45.22 | 301.2 |
| APCNet [11] | D-ResNet-101 | 45.38 | - |
| CPNet [49] | D-ResNet-101 | 46.27 | 314.3 |
| SETR [58] | ViT-Large | 50.28 | 591.0 |
| SegFormer [46] | MiT-B5 | **51.80** | 184.6 |
| SFNet [24] | ResNet-101 | 44.67 | 119.4 |
| IFA (Ours) | ResNet-101 | 45.23 | 67.1 |
| IFA* (Ours) | ResNet-101 | **45.98** | 72.0 |

Table 6: Comparisons with state-of-art on the ADE20K `val` set. 'D-' denotes the dilated version of the backbone, 'IFA*' denotes IFA with ASPP module. All methods use multi-scale inference. GFLOPs is calculated with a image size of $512 \times 512$.

**PASCAL Context.** We also conduct experiments on the PASCAL Context dataset. We report the results under 60 classes without the background. Table 5 presents the results on the PASCAL Context `test` set. Our method achieves competitive performance compared with state-of-the-art methods, with less computation cost. In particular, our method surpasses most of the previous methods with much less computation cost and achieves competitive results with SFNet, with only 61% of its overhead.

**ADE20K.** We also carry out experiments on the ADE20K dataset. Performance results are reported in Table 6. Our method achieves competitive result compared with CNN based CPNet, with only 22% of its computational cost.

## 5   Conclusion

In this paper, we focus on the feature alignment problem in popular semantic segmentation models involving feature aggregation operations. Hence, we present the Implicit Feature Alignment function (IFA) to perform precise feature alignment among multi-level features. IFA let multi-level features be learned as representing a continuous field of feature and represents the context from different levels as a function of continuous coordinates, which leads to a precise feature alignment in a resolution-free manner. Extensive experiments demonstrate the effectiveness of each component of IFA. Our IFA achieves competitive results on three benchmark datasets, *i.e.,* Cityscapes, PASCAL Context and ADE20K. Importantly, our method obtains a better trade-off between segmentation accuracy and computational cost than previous methods.

# References

1. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017) 13
2. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017) 4, 5, 13
3. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018) 2, 8, 10
4. Chen, Y., Liu, S., Wang, X.: Learning continuous image representation with local implicit image function. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8628–8638 (2021) 2, 4, 7
5. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5939–5948 (2019) 4
6. Cheng, B., Parkhi, O., Kirillov, A.: Pointly-supervised instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2617–2626 (2022) 5
7. Cheng, B., Schwing, A., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. Advances in Neural Information Processing Systems **34** (2021) 4
8. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3223 (2016) 3, 8
9. Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H.: Dual attention network for scene segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3146–3154 (2019) 12, 13
10. Genova, K., Cole, F., Vlasic, D., Sarna, A., Freeman, W.T., Funkhouser, T.: Learning shape templates with structured implicit functions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7154–7164 (2019) 4
11. He, J., Deng, Z., Zhou, L., Wang, Y., Qiao, Y.: Adaptive pyramid context network for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7519–7528 (2019) 14
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 3, 6
13. Hu, H., Cui, J., Wang, L.: Region-aware contrastive learning for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16291–16301 (2021) 6
14. Hu, H., Ji, D., Gan, W., Bai, S., Wu, W., Yan, J.: Class-wise dynamic graph convolution for semantic segmentation. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XVII. vol. 12362, pp. 1–17. Springer (2020) 6
15. Hu, H., Wei, F., Hu, H., Ye, Q., Cui, J., Wang, L.: Semi-supervised semantic segmentation via adaptive equalization learning. Advances in Neural Information Processing Systems **34**, 22106–22118 (2021) 6

16. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: Ccnet: Criss-cross attention for semantic segmentation. arXiv preprint arXiv:1811.11721 (2018) 12, 14

17. Huang, Z., Wei, Y., Wang, X., Shi, H., Liu, W., Huang, T.S.: Alignseg: Feature-aligned segmentation networks. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021) 4, 9, 10

18. Jiang, C., Sud, A., Makadia, A., Huang, J., Nießner, M., Funkhouser, T., et al.: Local implicit grid representations for 3d scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6001–6010 (2020) 2, 7

19. Ke, T.W., Hwang, J.J., Liu, Z., Yu, S.X.: Adaptive affinity fields for semantic segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 587–602 (2018) 12

20. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6399–6408 (2019) 2, 3, 4, 6

21. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012) 3, 9

22. Li, X., Zhong, Z., Wu, J., Yang, Y., Lin, Z., Liu, H.: Expectation-maximization attention networks for semantic segmentation. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019) 13

23. Li, X., Li, X., Zhang, L., Cheng, G., Shi, J., Lin, Z., Tan, S., Tong, Y.: Improving semantic segmentation via decoupled body and edge supervision. In: European Conference on Computer Vision. pp. 435–452. Springer (2020) 6

24. Li, X., You, A., Zhu, Z., Zhao, H., Yang, M., Yang, K., Tan, S., Tong, Y.: Semantic flow for fast and accurate scene parsing. In: European Conference on Computer Vision. pp. 775–793. Springer (2020) 4, 6, 12, 13, 14

25. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1925–1934 (2017) 12, 13

26. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017) 4, 6

27. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015) 2, 3, 13

28. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4460–4470 (2019) 2

29. Michalkiewicz, M., Pontes, J.K., Jack, D., Baktashmotlagh, M., Eriksson, A.: Implicit surface representations as layers in neural networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4743–4752 (2019) 4

30. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I. vol. 12346, pp. 405–421. Springer (2020) 4, 7

31. Mottaghi, R., Chen, X., Liu, X., Cho, N.G., Lee, S.W., Fidler, S., Urtasun, R., Yuille, A.: The role of context for object detection and semantic segmentation in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 891–898 (2014) 3, 8

32. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 165–174 (2019) 2, 4

33. Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., Bengio, Y., Courville, A.: On the spectral bias of neural networks. In: International Conference on Machine Learning. pp. 5301–5310. PMLR (2019) 7

34. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015) 4

35. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2304–2314 (2019) 2

36. Shen, T., Zhang, Y., Qi, L., Kuen, J., Xie, X., Wu, J., Lin, Z., Jia, J.: High quality segmentation for ultra high-resolution images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1310–1319 (2022) 4

37. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 761–769 (2016) 6

38. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 3

39. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. Advances in Neural Information Processing Systems **33** (2020) 2, 4

40. Strudel, R., Garcia, R., Laptev, I., Schmid, C.: Segmenter: Transformer for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7262–7272 (2021) 4

41. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning. pp. 10347–10357. PMLR (2021) 13

42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017) 4

43. Wang, J., Chen, K., Xu, R., Liu, Z., Loy, C.C., Lin, D.: CARAFE: content-aware reassembly of features. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. pp. 3007–3016. IEEE (2019) 4, 9, 10

44. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., et al.: Deep high-resolution representation learning for visual recognition. IEEE transactions on pattern analysis and machine intelligence (2020) 8, 10

45. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7794–7803 (2018) 4

46. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in Neural Information Processing Systems **34** (2021) 4, 12, 14
47. Xu, X., Wang, Z., Shi, H.: Ultrasr: Spatial encoding is a missing key for implicit image function-based arbitrary-scale super-resolution. arXiv preprint arXiv:2103.12716 (2021) 4, 7
48. Yang, M., Yu, K., Zhang, C., Li, Z., Yang, K.: Denseaspp for semantic segmentation in street scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3684–3692 (2018) 12
49. Yu, C., Wang, J., Gao, C., Yu, G., Shen, C., Sang, N.: Context prior for scene segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12416–12425 (2020) 12, 14
50. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 325–341 (2018) 12
51. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Learning a discriminative feature network for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1857–1866 (2018) 12
52. Yuan, Y., Chen, X., Wang, J.: Object-contextual representations for semantic segmentation. In: Proceedings of the European Conference on Computer Vision. pp. 173–190 (2020) 10
53. Zhang, F., Chen, Y., Li, Z., Hong, Z., Liu, J., Ma, F., Han, J., Ding, E.: Acfnet: Attentional class feature network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6798–6807 (2019) 6
54. Zhang, H., Dana, K., Shi, J., Zhang, Z., Wang, X., Tyagi, A., Agrawal, A.: Context encoding for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7151–7160 (2018) 13
55. Zhang, H., Zhang, H., Wang, C., Xie, J.: Co-occurrent features in semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 548–557 (2019) 14
56. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881–2890 (2017) 4, 6, 10, 12, 14
57. Zhao, H., Zhang, Y., Liu, S., Shi, J., Change Loy, C., Lin, D., Jia, J.: Psanet: Pointwise spatial attention network for scene parsing. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 267–283 (2018) 12, 14
58. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H.S., Zhang, L.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021. pp. 6881–6890. Computer Vision Foundation / IEEE (2021) 4, 12, 13, 14
59. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017) 3, 8
60. Zhu, L., Ji, D., Zhu, S., Gan, W., Wu, W., Yan, J.: Learning statistical texture for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12537–12546 (2021) 12
61. Zhu, Z., Xu, M., Bai, S., Huang, T., Bai, X.: Asymmetric non-local neural networks for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 593–602 (2019) 12, 13