This WACV 2023 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Dense Prediction with Attentive Feature Aggregation

Yung-Hsu Yang[1] royyang@gapp.nthu.edu.tw
Thomas E. Huang[2] thomas.huang@vision.ee.ethz.ch
Min Sun[1] sunmin@ee.nthu.edu.tw
Samuel Rota Bulò[3] rotabulo@fb.com
Peter Kontschieder[3] pkontschieder@fb.com
Fisher Yu[2] i@yf.io
[1]National Tsing Hua University [2]ETH Zürich [3]Facebook Reality Labs

Abstract

Aggregating information from features across different layers is essential for dense prediction models. Despite its limited expressiveness, vanilla feature concatenation dominates the choice of aggregation operations. In this paper, we introduce Attentive Feature Aggregation (AFA) to fuse different network layers with more expressive non-linear operations. AFA exploits both spatial and channel attention to compute weighted averages of the layer activations. Inspired by neural volume rendering, we further extend AFA with Scale-Space Rendering (SSR) to perform a late fusion of multi-scale predictions. AFA is applicable to a wide range of existing network designs. Our experiments show consistent and significant improvements on challenging semantic segmentation benchmarks, including Cityscapes and BDD100K at negligible computational and parameter overhead. In particular, AFA improves the performance of the Deep Layer Aggregation (DLA) model by nearly 6% mIoU on Cityscapes. Our experimental analyses show that AFA learns to progressively refine segmentation maps and improve boundary details, leading to new state-of-the-art results on boundary detection benchmarks on NYUDv2 and BSDS500.

1. Introduction

Dense prediction tasks such as semantic segmentation [7, 26, 42] and boundary detection [28, 1] are fundamental enablers for many computer vision applications. Semantic segmentation requires predictors to absorb intra-class variability while establishing inter-class decision boundaries. Boundary detection also requires an understanding of fine-grained scene details and object-level boundaries. A popular solution is to exploit the multi-scale representations to balance preserving spatial details from shallower features and maintaining relevant semantic context in deeper ones.

There are two major approaches to obtaining effective multi-scale representations. Dilated convolutions [43] can aggregate context information while preserving spatial information. Most of the top performing segmentation methods adopt this approach [5, 50, 44, 34] to extract contextual pixel-wise information. The drawback is the extensive usage of layer memory for storing high-resolution feature maps. An alternative approach is to progressively downsample the layer resolution as in image classification networks and then upsample the resolution by aggregating information from different layer scales with layer concatenations [24, 45, 27]. Methods using this approach achieve state-of-the-art results with reduced computational efforts and fewer parameters [35]. Even though many works design new network architectures to effectively aggregate multi-scale information, the predominant aggregation operations are still feature concatenation or summation [27, 24, 35, 45, 49]. These linear operations do not consider feature interactions or selections between different levels or scales.

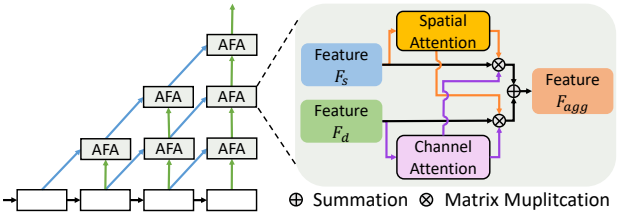

Figure 1. Attentive Feature Aggregation. F_s is the shallower input feature and F_d is the deeper one. We use attention to aggregate different scale or level information and obtain aggregated feature F_{agg} with rich representation.

We propose *Attentive Feature Aggregation* (AFA) as a nonlinear feature fusion operation to replace the prevailing tensor concatenation or summation strategies. Our attention module uses both spatial and channel attention to learn and predict the importance of each input signal during fusion. Aggregation is accomplished by computing a linear combination of the input features at each spatial location, weighted by their relevance. Compared to linear fusion op-

erations, our AFA module can attend to different feature levels depending on their importance. AFA introduces negligible computation and parameter overhead, and can be easily used to replace fusion operations in existing methods. Fig. 1 illustrates the concept of AFA.

Another challenge of dense prediction tasks is that fine details are better handled in higher resolutions but coarse information are better in lower resolutions. Multi-scale inference [3, 4, 32] has become a common approach to alleviate this trade-off, and using an attention mechanism is now a best practice. Inspired by neural volume rendering [11, 25], we extend AFA to *Scale-Space Rendering* (SSR) as a novel attention mechanism to fuse multi-scale predictions. We treat the prediction from each scale as sampled data in the scale-space and leverage the volume-rendering formulation to design a coarse-to-fine attention and render the final results. Our SSR is robust against the gradient vanishing problem and saves resources during training, thus achieving higher performance.

We demonstrate the effectiveness of AFA when applied to a wide range of existing networks on both semantic segmentation and boundary detection benchmarks. We plug our AFA module into various popular segmentation models: FCN [24], U-Net [27], HRNet [35], and Deep Layer Aggregation (DLA) [45]. Experiments on several challenging semantic segmentation datasets including Cityscapes [7] and BDD100K [42] show that AFA can significantly improve the segmentation performance of each representative model. Additionally, *AFA-DLA* has competitive results compared to the state-of-the-art models despite having fewer parameters and using less computation. Furthermore, AFA-DLA achieves the new state-of-the-art performances on boundary detection datasets NYUDv2 [28] and BSDS500 [1]. We conduct comprehensive ablation studies to validate the advantages of each component of our AFA module. Our source code will be released.

2. Related Work

Multi-Scale Context. To better handle fine details, segmentation models with convolutional trunks use low output strides. However, this limits the receptive field and the semantic information contained in the final feature. Some works utilize dilated backbones [43] and multi-scale context [39, 22, 13] to address this problem. PSPNet [50] uses a Pyramid Pooling Module to generate multi-scale context and fuse them as the final feature. The DeepLab models [5] use Atrous Spatial Pyramid Pooling to assemble context from multiple scales, yielding denser and wider features. In contrast, our AFA-DLA architecture extensively uses attention to conduct multi-scale feature fusion to increase the receptive field without using expensive dilated convolutions. Thus, our model can achieve comparable or even better performance with much less computation and parameters.

Feature Aggregation. Aggregation is widely used in the form of skip connections or feature fusion nodes in most deep learning models [24, 27, 35]. The Deep Layer Aggregation [45] network shows that higher connectivity inside the model can enable better performance with fewer parameters, but its aggregation is still limited to linear operators. Recently, some works have explored better aggregation of multi-scale features. [31, 20, 21, 48, 16, 18] improves the original FPN architecture with feature alignment and selection during fusion. CBAM [36] and SCA-CNN [2] uses channel and spacial self-attention to perform adaptive feature refinement and improves convolutional networks in image classification, object detection and image captioning. DANet [12] appends two separate branches with Transformer [33] self-attention as the proposed spatial and channel module on top of dilated FCN. In contrast, our AFA module leverages extracted spatial and channel information during aggregation to efficiently select the essential features with respect to the property of input features. With the efficient design, AFA can be directly adopted in popular architectures and extensively used at negligible computational and parameter overhead.

Multi-Scale Inference. Many computer vision tasks leverage multi-scale inference to get higher performance. The most common way to fuse the multi-scale results is to use average pooling [3, 45, 19], but it applies the same weighting to each scale. Some approaches use an explicit attention model [4, 40] to learn a suitable weighting for each scale. However, the main drawback is the increased computational requirement for evaluating multiple scales. To overcome this problem, HMA [32] proposes a hierarchical attention mechanism that only needs two scales during training but can utilize more scales during inference. In this work, we propose scale-space rendering (SSR), a more robust multi-scale attention mechanism that generalizes the aforementioned hierarchical approach and exploits feature relationships in scale-space to improve the performance further.

3. Method

In this section, we introduce our attentive feature aggregation (AFA) module and then extend AFA to scale-space rendering (SSR) attention for multi-scale inference. The overview of the complete architecture is shown in Fig. 2.

3.1. Attentive Feature Aggregation

Our attentive feature aggregation (AFA) module computes both spatial and channel attention based on the relation of the input feature maps. The attention values are then used to modulate the input activation scales and produce one merged feature map. The operation is nonlinear in contrast to standard feature concatenation or summation. We

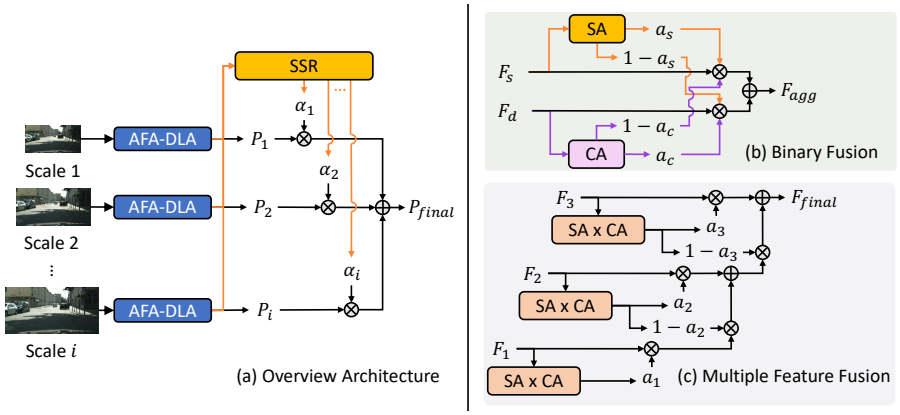

Figure 2. (a) Overview of AFA-DLA with SSR. We use two scales [0.5, 1.0] during training and more scales during inference to pursue higher performance. (b) Binary fusion module for input feature F_s and F_d. SA denotes our spatial attention module and generates spatial attention a_s. CA stands for our channel attention module and responsible for channel attention a_c. (c) Multiple feature fusion module for three input features F_1, F_2, and F_3. *SA* $\times$ *CA* represents computing spatial attention a_s and channel attention a_c first and then using element-wise multiplication to get the attention a_i for F_i.

use two different basic self-attention mechanisms to generate spatial and channel attention maps and reassemble them concerning the relation between the input features.

For the input feature $F_s \in \mathbb{R}^{C\times H\times W}$, the spatial attention uses a convolutional block ω_s consisting of two 3×3 convolutions to encode F_s. It is defined as

$$a_s \triangleq \sigma(\omega_s(F_s))\,, \tag{1}$$

where $a_s \in \mathbb{R}^{1\times H\times W}$ and σ is the sigmoid activation.

For computing the channel attention of input feature $F_d \in \mathbb{R}^{C\times H\times W}$, we first apply average pooling to get $F_d^{\texttt{avg}}$ and max pooling to get $F_d^{\texttt{max}}$. Then, we further transform the features to $F_c^{\texttt{avg}}$ and $F_c^{\texttt{max}}$ using another convolutional block ω_c, which consists of two 1×1 convolutions with a bottleneck input-output channel design. We sum them up with equal weighting and use sigmoid σ as the activation function to generate channel attention $a_c \in \mathbb{R}^{C\times 1\times 1}$ as

$$a_c \triangleq \sigma(\omega_c(\texttt{AvgPool}(F_d)) + \omega_c(\texttt{MaxPool}(F_d)))\,. \tag{2}$$

With the basis of the above attention mechanisms, we design two types of AFA for different aggregation scenarios and enable the network to model complex feature interactions and attend to different features.

Binary Fusion. We employ a simple attention-based aggregation mechanism using our spatial and channel attentions to replace standard binary fusion nodes. When merging two input feature maps, we apply channel and spatial attention separately to capture the relation of input features. As shown in Fig. 2 (b), when two features are aggregated, we denote the shallower feature map as F_s and the other as F_d. F_s is used to compute a_s and F_d is responsible for a_c, as the shallower layers will contain richer spatial information and the deeper ones will have more complex channel features. Then, we obtain the aggregated feature $F_{\texttt{agg}}$ as

$$F_{\texttt{agg}} \triangleq a_s \odot (1 - a_c) \odot F_s + (1 - a_s) \odot a_c \odot F_d\,, \tag{3}$$

where $\odot$ denotes element-wise multiplication (with broadcasted unit dimensions). By leveraging the input features properties, our binary fusion is simple yet effective.

Multiple Feature Fusion. We extend the binary fusion node to further fuse together multiple multi-scale features. Recent works [27, 35, 45] iteratively aggregate features across the model, but only exploit the final feature for downstream tasks, neglecting intermediate features computed during the aggregation process. By applying AFA on these intermediate features, we give the model more flexibility to select the most relevant features.

Given k multi-scale features F_i for $i \in \{1, \ldots, k\}$, we first order them based on the amount of aggregated information they contain, *i.e.*, a feature with higher priority will have gone through a higher number of aggregations. Then, we compute both spatial and channel attention for each feature and take the product as the new attention. The combined attention a_i is defined as

$$a_i \triangleq \texttt{SA}(F_i) \odot \texttt{CA}(F_i)\,, \tag{4}$$

where SA denotes our spatial attention function and CA our channel attention function. For fusing the multi-scale features, we perform hierarchical attentive fusion by progressively aggregating features starting from F_1 to F_k to obtain the final representation $F_{\texttt{final}}$ as

$$F_{\texttt{final}} \triangleq \sum_{i=1}^{k} \left[a_i \odot F_i \odot \prod_{j=i+1}^{k} (1 - a_j) \right]\,. \tag{5}$$

In Fig. 2 (c), we show an example of this process with $k = 3$. The new final representation $F_{\texttt{final}}$ is an aggregation of features at multiple scales, combining information from shallow to deep levels.

AFA is flexible and can be applied to widely used segmentation models, as shown in Fig. 3. In U-Net [27] and HRNet [35], we add our multiple feature fusion module to

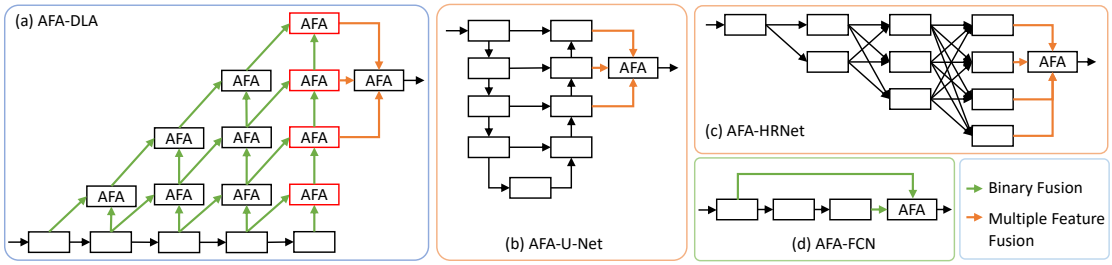

Figure 3. Segmentation models with our AFA module. We show parts of the original models related to feature aggregation and our modifications. Red blocks represent auxiliary segmentation heads added during training.

fully utilize the previously unused aggregated multi-scale features. In FCN [24], we replace the original linear aggregation node in the decoder with our attentive binary fusion. For DLA [45], we not only substitute the original aggregation nodes but also add our multiple feature fusion module. Due to higher connectivity of its nodes, the DLA network can benefit more from our improved feature aggregation scheme, and thus we use AFA-DLA as our final model.

Comparison with other Attention Modules. Unlike previous attention methods [2, 36, 12], AFA focuses on aggregating feature maps of different network layers to obtain more expressive representations with a lightweight module. Compared to GFF [21], AFA consumes $1/4$ FLOPs and model parameters for binary fusion and $1/2$ FLOPS and $1/5$ model parameters for multiple feature fusion. Without using heavy self-attention mechanism as DANet [12], AFA consumes only $1/2$ FLOPs and model parameters with $1/4$ GPU memory under the same input features. With a simple yet effective design, AFA can be extensively used in existing architectures without much additional overhead.

3.2. Scale-Space Rendering

Multi-scale attention [4, 32] is typically used to fuse multi-scale predictions and can alleviate the trade-off in performance on fine and coarse details in dense prediction tasks. However, repeated use of attention layers may lead to numerical instability or vanishing gradients, which hinders its performance. To resolve this issue, we extend the attention mechanism mentioned above using a volume rendering scheme applied to the scale space. By treating the multi-scale predictions as samples in a scale-space representation, this scheme provides a hierarchical, coarse-to-fine way of combining predictions using a scale-specific attention mechanism. We will also show that our approach generalizes the hierarchical multi-scale attention method [32].

Without loss of generality, we focus on a single pixel and assume that our model provides a dense prediction for the target pixel at k different scales. The prediction for the ith scale is denoted by $P_i \in \mathbb{R}^d$. Accordingly, $P \triangleq (P_1, \dots, P_k)$ denotes the feature representation of the target pixel in our scale-space. Furthermore, we assume that $i < j$ implies that scale i is coarser than scale j.

Our target pixel can be imagined as a ray moving through scale-space, starting from scale 1 towards scale k. We redesign the original hierarchical attention in the proposed multiple feature fusion mechanism to mimic the volume-rendering equation, where the volume is implicitly given by the scale-space. To this end, besides the feature representation P_i at scale i, we assume our model to predict for the target pixel also a scalar $y_i \in \mathbb{R}$ so that $e^{-\phi(y_i)}$ represents the probability that the particle will cross scale i, given some non-negative scalar function $\phi : \mathbb{R} \to \mathbb{R}_+$. We can then express the scale attention α_i as the probability of the particle to reach scale i and stop there, *i.e.*,

$$\alpha_i(y) \triangleq \left[1 - e^{-\phi(y_i)}\right] \prod_{j=1}^{i-1} e^{-\phi(y_j)} \tag{6}$$

where $y \triangleq (y_1, \dots, y_k)$. Finally, the fused multi-scale prediction for the target pixel can be regarded as the "rendered" pixel, where the pixel features at the different scales P_i are averaged by the attention coefficients α_i following the volume rendering equations. Accordingly, $P_{\texttt{final}} \triangleq \sum_{i=1}^{k} P_i \alpha_i(y)$ represents the feature for the target pixel that we obtain after fusing P across all scales with attention driven by y.

The proposed scale-space rendering (SSR) mechanism can be regarded as a generalization of the hierarchical multi-scale attention proposed in [32], for the latter can be obtained from our formulation by simply setting $\phi(y_i) \triangleq \log(1 + e^{y_i})$, *i.e.*, ϕ is the soft-plus function, and by fixing $\phi(y_k) \triangleq \infty$.

Choice of ϕ. In our experiments, we use the absolute value function as our ϕ, *i.e.*, $\phi(y_i) \triangleq |y_i|$. This is motivated by a better preservation of the gradient flow through the attention mechanism, as we found existing attention mechanisms to suffer from vanishing gradient issues. Consider the Jacobian of the attention coefficients, which takes the form:

$$J_{i\ell} \triangleq \frac{\partial \alpha_i(y)}{\partial y_\ell} = \begin{cases} \phi'(y_i) \prod_{j=1}^{i} e^{-\phi(y_j)} & \text{if } \ell = i \\ 0 & \text{if } \ell > i \\ -\phi'(y_\ell)\alpha_i(y) & \text{if } \ell < i\,. \end{cases} \tag{7}$$

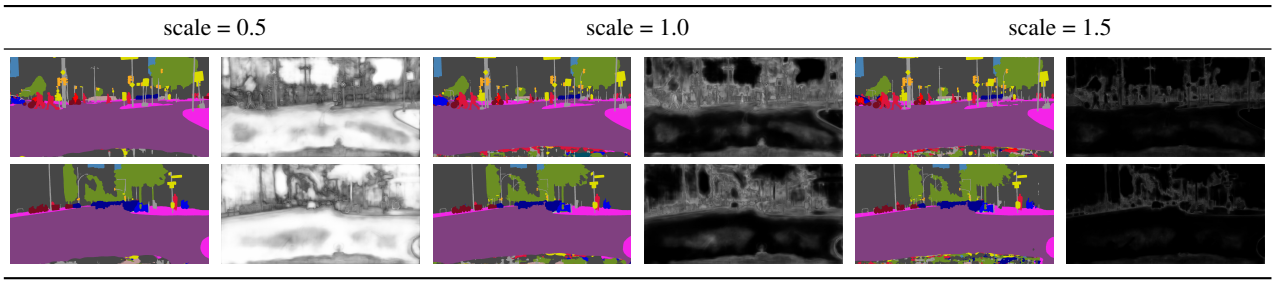

Figure 4. Visualization of attention maps generated by scale-space rendering (SSR) with the predictions. Whiter regions denote higher attention. SSR learns to focus on detailed regions in larger scale images and on lower frequency information in smaller scale ones.

In the presence of two scales, this becomes:

$$J = \begin{bmatrix} \phi'(y_1)a_1 & 0 \\ -\phi'(y_1)a_1(1-a_2) & \phi'(y_2)a_1a_2 \end{bmatrix}, \quad (8)$$

where $a_i \triangleq e^{-\phi(y_i)}$. As $a_1 \to 0$, the gradient vanishes, for J tends to a null matrix. Otherwise, irrespective of the value of a_2, the gradient will vanish only depending on the choice of ϕ. In particular, by taking the absolute value as ϕ we have that the Jacobian will not vanish for $a_1 > 0$ and $(y_1, y_2) \neq (0, 0)$, thus motivating our choice of using the absolute value as ϕ. If we consider instead the setting in HMA [32], we have that $a_2 = 0$ and $\phi'(y_i) = 1 - a_i$. It follows that the Jacobian vanishes also as $a_1 \to 1$. The conclusion is that the choice of ϕ plays a role in determining the amount of gradient that flows through the predicted attention and that the approach in HMA [32] is more subject to vanishing gradient issues than our proposed solution. We compare HMA and SSR quantitatively in Section 4.

To understand which parts of the image SSR attends to at each scale, we visualize generated attention maps in Fig. 4. Detailed regions are processed more effectively in larger scale images due to the higher resolution, while the prediction of lower frequency region is often better in smaller scales. SSR learns to focus on the right region for different scales and boosts the final performance.

We combine AFA-DLA with SSR to produce the final predictions. As shown in Fig. 2, AFA-DLA propagates information from different scales to the SSR module, which then generates attention masks α_i used to fuse the predictions P_i to get our final prediction $P_{\texttt{final}}$.

Training Details. For fair comparison with other methods [46, 32], we reduce the number of filters from 256 to 128 in the OCR [46] module and add it after AFA-DLA to refine our final predictions. Our final model can be trained at k different scales. Due to the limitation of computational resources, we use $k = 2$ for training and RMI [51] to be the primary loss function $L_{\texttt{primary}}$ for our final prediction $P_{\texttt{final}}$. We add three different types of auxiliary cross-entropy losses to stabilize the training. First, we use the generated SSR attention to fuse the auxiliary per-scale predictions from OCR, yielding $P^{\texttt{aux}}_{\texttt{ocr}}$ and the loss $L_{\texttt{ocr}}$. Second, we compute and sum up cross-entropy losses for each scale prediction P_i yielding $L_{\texttt{scale}}$. Lastly, we add auxiliary segmentation heads inside AFA-DLA as in Fig. 3 (a) and have predictions for each scale. We fuse them with SSR across scales and get $P^{\texttt{aux}}_j$, where $1 \leq j \leq 4$. We compute the auxiliary loss for each and sum them up as $L_{\texttt{aux}}$. Accordingly, the total loss function is the weighted sum as

$$L_{\texttt{all}} \triangleq L_{\texttt{primary}} + \beta_o L_{\texttt{ocr}} + \beta_s L_{\texttt{scale}} + \beta_a L_{\texttt{aux}}, \quad (9)$$

where we set $\beta_o \triangleq 0.4$, $\beta_s \triangleq 0.05$ and $\beta_a \triangleq 0.05$. We provide more details in the supplementary materials.

4. Experiments

We conduct experiments on several public datasets on both semantic segmentation and boundary detection tasks, and conduct a thorough analysis with a series of ablation studies. Due to the space limit, we leave additional implementation details to our supplementary materials.

4.1. Results on Cityscapes

The Cityscapes dataset [7] provides high resolution (2048 x 1024) urban street scene images and their corresponding segmentation maps. It contains 5K well annotated images for 19 classes and 20K coarsely labeled image as extra training data. Its finely annotated images are split into 2975, 500, and 1525 for training, validation and testing. We use DLA-X-102 as the backbone for AFA-DLA with a batch size of 8 and full crop size. Following [32], we train our model with auto-labeled coarse training data with 0.5 probability and otherwise use the fine labeled training set. During inference, we use multi-scale inference with [0.5, 1.0, 1.5, 1.75, 2.0] scales, image flipping, and SegFix [47] post-processing. We detail the effect of each post-processing technique in the supplementary material.

The results on the validation and test set are shown in Table 1. With only using ImageNet [8] pre-training and without using external segmentation datasets, AFA-DLA obtains 85.14 mean IoU on the Cityscapes validation set, achieving the best performance compared to other methods in the same setting. AFA-DLA outperforms the previous multi-scale attention methods and the recent methods using the Vision Transformer [10] architecture. On the Cityscapes test set, AFA-DLA also obtains competitive

Table 1. Segmentation results on Cityscapes validation and testing sets. We only compare to published methods without using extra segmentation datasets. AFA-DLA achieves the best performance on the validation set and competitive performance with the top performing method on the test set.

Method	mIoU (val)	mIoU (test)
DLA [45]	75.10	75.90
SFNet [20]	*N/A*	81.80
DeepLabV3+ [5]	79.55	82.10
DANet [12]	81.50	81.50
FPT [48]	81.70	82.20
Gated-SCNN [31]	81.80	82.80
GFF [21]	81.80	82.20
SETR [29]	82.20	81.60
SegFormer [37]	82.40	82.20
AlignSeg [18]	82.40	82.60
OCR [46]	82.40	83.00
DecoupleSegNets [19]	83.50	**83.70**
Mask2Former [6]	84.30	*N/A*
AFA-DLA (Ours)	**85.14**	*83.58*

Table 2. Resource usage of different models. AFA-DLA uses much fewer operations and parameters when compared to top performing methods.

Method	FLOPs (G)	Param. (M)
DLA-X-102 [45]	533	34.7
DeepLabV3+ [5]	2514	54.4
DecoupleSegNet [19]	6197	138.4
AFA-DLA-X-102 (Ours)	1333	36.3

performance with a top performing method, DecoupleSegNet [19], while using around 75% fewer operations and parameters as shown in Table 2.

We additionally evaluate the application of AFA to other widely used segmentation models, including FCN, U-Net, and HRNet. We build the baselines on our own and use the same shorter learning schedule and smaller training crop size for all models for fair comparison in Table 3. Since we only modify the aggregation operations of each model, we can still use the original ImageNet [8] pre-training weights.

Combined with AFA, the segmentation models can each obtain at least 2.5% improvement in mIoU, with only a small computational and parameter overhead. In particular, we even lighten HRNet by replacing its concatenation in the last layer with our multiple feature fusion and still achieve 2.5% improvement. This demonstrates AFA as a lightweight module that can be readily applied to existing models for segmentation.

4.2. Results on BDD100K

BDD100K [42] is a diverse driving video dataset for multitask learning. For the semantic segmentation task, it provides 10K images with same categories as Cityscapes at 1280 x 720 resolution. The dataset consists 7K, 1K, and 2K images for training, validation, and testing. Considering the amount of training data is twice as Cityscapes, we use DLA-169 as the backbone with full image crop and 16 training batch size for 200 epochs. During inference, we use multi-scale inference with [0.5, 1.0, 1.5, 2.0] scales and image flipping.

Table 3. Combining AFA with other widely used segmentation models on the Cityscapes validation set. With AFA, each model can obtain at least 2.5% improvement in mIoU, with only a small computational and parameter overhead. The baselines are implemented on our own and all experiments are under fair comparison.

Method	FLOP (G)	Param. (M)	mIoU	Δ (%)
FCN	1581.8	49.5	75.52	-
AFA-FCN	1659.2	51.9	**77.88**	3.1
U-Net-S5-D16	1622.8	29.1	62.73	-
AFA-U-Net	2146.7	29.4	**64.42**	2.7
HRNet-W48	748.7	65.9	78.48	-
AFA-HRNet	701.4	65.4	**80.41**	2.5

Table 4. Segmentation results on BDD100K validation and testing set. † denotes using Cityscapes data for pre-training. AFA-DLA achieves the new state-of-the-art performance on both sets.

Method	mIoU (val)	mIoU (test)
DLA [45]	57.84	*N/A*
CCNet [17]	64.03	55.93
DNL [41]	*N/A*	56.31
PSPNet [50]	*N/A*	56.32
Deeplabv3+ [5]	64.49	57.00
DecoupleSegNet† [19]	66.90	*N/A*
AFA-DLA (Ours)	**67.46**	**58.47**

The results on validation and test sets are shown in Table 4. AFA-DLA achieves new state-of-the-art performances on both sets despite using fewer operations and parameters when compared to the top performing methods as shown in Table 2. Our method achieves 67.46 mIoU on the validation set and is even higher than DecoupleSegNet [19], which uses Cityscapes pre-trained weights. Moreover, AFA-DLA obtains 58.47 mIoU on the test set, which outperforms all the strong official baselines.

4.3. Boundary Detection

We additionally conduct experiments on boundary detection, which involves predicting a binary segmentation mask indicating the existence of boundaries. We evaluate on two standard boundary detection datasets, NYU Depth Dataset V2 (NYUDv2) [28] and Berkeley Segmentation Data Set and Benchmarks 500 (BSDS500) [1]. For each dataset, we follow the standard data preprocessing and evaluation protocol in literature [38, 23]. Specifically, we augment each dataset by randomly flipping, scaling, and rotating each image. We evaluate using commonly used metrics, which are the F-measure at the Optimal Dataset Scale (ODS) and at

Table 5. Boundary detection results on NYUDv2 test set. AFA-DLA achieves the new state-of-the-art results.

Method	ODS	OIS
AMH-Net [23]	0.771	0.786
BDCN [15]	0.765	0.781
PiDiNet [30]	0.756	0.773
AFA-DLA (Ours)	**0.780**	**0.792**

Table 6. Boundary detection results on BSDS500 test set. AFA-DLA outperforms all other methods in ODS.

Method	ODS	OIS
DLA [45]	0.803	0.813
LPCB [9]	0.800	0.816
BDCN [15]	0.806	**0.826**
AFA-DLA (Ours)	**0.812**	**0.826**

the Optimal Image Scale (OIS). Following [45], we also scale the boundary labels by 10 to account for the label imbalance. For simplicity, we do not consider using multi-scale images during inference, so SSR is not used.

Results on NYUDv2. The NYUDv2 dataset contains both RGB and depth images. There are 381 training images, 414 validation images, and 654 testing images. We follow the same procedure as [38, 23, 15] and train a separate model on RGB and HHA [14] images. We evaluate using both RGB and HHA images as input by averaging each model's output during inference. The results are shown in Table 5. AFA-DLA outperforms all other methods by a large margin and achieves state-of-the-art performances. In particular, when using both RGB and HHA as input, AFA-DLA can achieve a high score of 0.780 in ODS and 0.792 in OIS.

Results on BSDS500. The BSDS500 dataset contains 200 training images, 100 validation images, and 200 testing images. We follow standard practice [38] and only use boundaries annotated by three or more annotators for supervision. We do not consider augmenting the training set with additional data, so we only utilize the available data in the BSDS500 dataset. As in Table 6, AFA-DLA achieves superior performance when compared to methods only trained on the BSDS500 dataset and obtains 0.812 in ODS.

4.4. Ablation Experiments

In this section, we conduct several ablation studies on the Cityscapes validation set to validate each component of AFA-DLA. The main baseline model we compare to is DLA [45] with DLA-34 as backbone. All the results are listed in Table 7. We also provide visualizations in order to qualitatively evaluate our model.

Binary Fusion. We first evaluate our attentive binary fusion module, which can learn the importance of each input signal during fusion. Compared to using standard linear fusion operators, introducing nonlinearity and using channel attention (denoted CA) during binary fusion achieves around 1.1 mean IoU improvement. This demonstrates that more expressive aggregation can drastically improve the results. When we additionally use spatial attention (denoted SA + CA), we observe 0.6 points further improvement.

Table 7. Ablation study on Cityscapes validation set with DLA-34 as backbone. Aux. Head denotes using auxiliary heads, MFF denotes multiple feature fusion, SA and CA denote using spatial and channel attention for feature fusion, Swap denotes switching the input of spatial and channel attention modules, and Both stands for using both input features to generate attention for fusion.

Binary Fusion	Aux. Head	MFF	MS Inference	mIoU
Original	-	-	Single Scale	74.43
Swap	-	-	Single Scale	75.37
CA	-	-	Single Scale	75.54
CBAM [36]	-	-	Single Scale	75.70
Both	-	-	Single Scale	75.77
SA + CA	-	-	Single Scale	**76.14**
SA + CA	✓	-	Single Scale	76.45
SA + CA	✓	✓	Single Scale	**77.08**
SA + CA	✓	✓	Avg. Pooling	78.56
SA + CA	✓	✓	HMA [32]	80.18
SA + CA	✓	✓	SSR	**80.74**

Table 8. Validation performance (mIoU) on Cityscapes between SSR and HMA across early training epochs. SSR achieves better performance over HMA across all epochs.

Method	epoch 1	epoch 50	epoch 100	epoch 150
HMA [32]	3.57	64.78	71.61	73.03
SSR	**5.49**	**68.16**	**72.76**	**74.48**

Attention Mechanism. We validate the design of AFA by evaluating various other strategies for computing attention. Switching the input of spatial and channel attention modules (denoted Swap) can lead to a minor improvement, but it is even worse than only using channel attention. We also apply the CBAM [36] module on top of the original DLA linear aggregation nodes to refine the aggregated features as another baseline. Finally, we concatenate both input features and use it to generate each attention (denoted Both), which requires much more computation. On the contrary, our attentive binary fusion design can achieve the best performance. This shows that the design of the aggregation node should consider the properties of the input features and AFA is the most effective.

Auxiliary Segmentation Head. We add several auxiliary heads into AFA-DLA to stabilize the training, which is common practice among other popular baseline models. The whole backbone can be supervised by the auxiliary losses efficiently. We see about 0.3 mIoU improvement.

Multiple Feature Fusion. We apply our multiple feature fusion to enable AFA-DLA to fully leverage intermediate features in the network. This gives the network more flexibility in selecting relevant features for computing the final

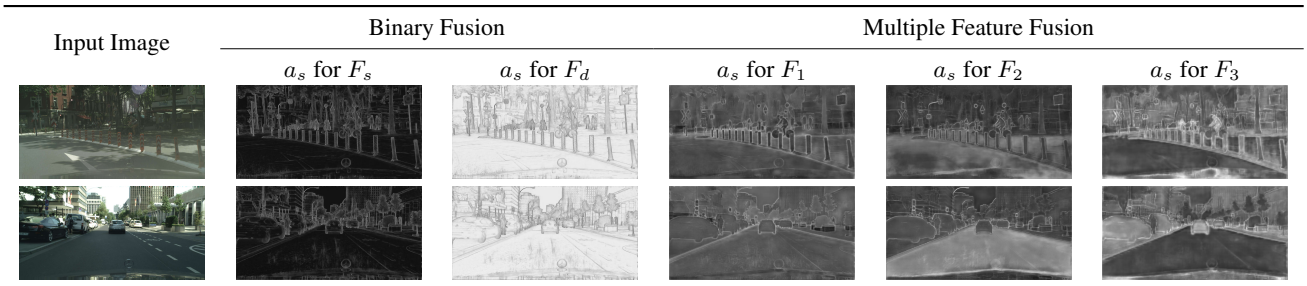

Figure 5. Visualization of spatial attention maps a_s generated by our attentive feature aggregation modules. Whiter regions denote higher attention. Compared to linear fusion operations, our AFA modules provide a more expressive way of combining features.

feature. By adding the multiple feature fusion module, we gain another 0.6 mIoU.

Scale-Space Rendering. We employ our SSR module to fuse multi-scale predictions. After applying SSR with [0.25, 0.5, 1.0, 2.0] inference scales, we gain an impressive improvement of nearly 3.7% mIoU over using only a single scale. We also compare with different multi-scale inference approaches under the same training setting. SSR gains 1.2 mIoU over standard average pooling and further outperforms hierarchical multi-scale attention [32] by nearly 0.6 mean IoU. In terms of FLOPs, HMA uses around 1433G and SSR consumes 1420G, so SSR does not require additional computational resources. Furthermore, we report validation performances of HMA and SSR at intermediate checkpoints in Table 8. The results suggest that our scale-space rendering attention can alleviate the gradient vanishing problem and boost the overall performance, while still retaining the flexibility for selecting different training and inference scales. With both AFA and SSR, we improve the DLA baseline model performance by over 6.3 mIoU.

Attention Visualization. To understand where our AFA fusion modules attends to, we visualize the generated attention maps for a set of input features in Fig. 5. AFA learns to attend to different regions of the input features depending on the information they contain. Binary fusion module focuses on object boundaries in shallower features F_s and attends to the rest on deeper features F_d. Our multiple feature fusion module can perform complex selection of features by attending to different regions for each feature level. F_1 aggregates shallower features and thus the module attends to the boundaries, while the rest attend to objects or the background. Compared to linear fusion operations, AFA provides a more expressive way of combining features.

Segmentation Visualization. We take a deeper look at the semantic segmentation results on the Cityscapes produced by AFA-DLA in Fig. 6 and compare them to those produced by DLA [45]. With our AFA module, the model can better leverage spatial and channel information to better distinguish object boundaries and classify object classes.

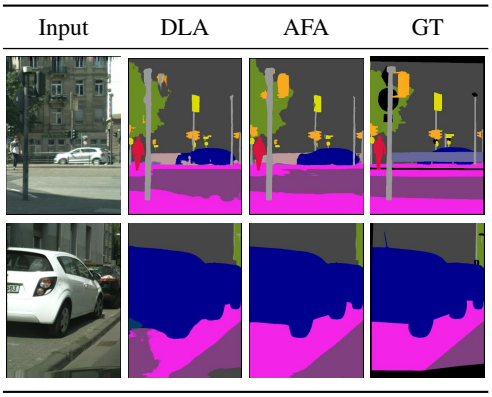

Figure 6. Comparison of predictions generated by DLA and AFA-DLA. The black pixels are ignored. AFA-DLA can better distinguish object boundaries and correctly classify object classes.

5. Conclusion

We propose a novel attention-based feature aggregation module combined with a new multi-scale inference mechanism to build the competitive AFA-DLA model. With spatial and channel attention mechanisms, AFA enlarges the receptive field and fuses different network layer features effectively. SSR improves existing multi-scale inference methods by being more robust towards the gradient vanishing problem. Applying all of our components, we improve the DLA baseline model performance by nearly 6.3 mean IoU on Cityscapes. When combining AFA with existing segmentation models, we found consistent improvements of at least 2.5% in mean IoU on Cityscapes, with only a small cost in computational and parameter overhead. AFA-DLA also establishes new state-of-the-art results on BDD100K and achieves the new best score on Cityscapes when not using external segmentation datasets. Moreover, for the boundary detection task, AFA-DLA obtains state-of-the-art results on NYUDv2 and BSDS500.

6. Acknowledgment

We gratefully acknowledge the support of computer time and facilities from Ministry of Science and Technology of Taiwan (MOST110-2634-F-002-051).

References

[1] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2010.

[2] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, 2017.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 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.

[4] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.

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

[6] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *arXiv preprint arXiv:2112.01527*, 2021.

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

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[9] Ruoxi Deng, Chunhua Shen, Shengjun Liu, Huibing Wang, and Xinru Liu. Learning to predict crisp boundaries. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 562–578, 2018.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[11] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM Siggraph Computer Graphics*, 22(4):65–74, 1988.

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

[13] Jun Fu, Jing Liu, Yuhang Wang, Yong Li, Yongjun Bao, Jinhui Tang, and Hanqing Lu. Adaptive context network for scene parsing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6748–6757, 2019.

[14] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European conference on computer vision*, pages 345–360. Springer, 2014.

[15] Jianzhong He, Shiliang Zhang, Ming Yang, Yanhu Shan, and Tiejun Huang. Bi-directional cascade network for perceptual edge detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3828–3837, 2019.

[16] Shihua Huang, Zhichao Lu, Ran Cheng, and Cheng He. Fapn: Feature-aligned pyramid network for dense image prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 864–873, 2021.

[17] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 603–612, 2019.

[18] Zilong Huang, Yunchao Wei, Xinggang Wang, Wenyu Liu, Thomas S Huang, and Humphrey Shi. Alignseg: Feature-aligned segmentation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):550–557, 2021.

[19] Xiangtai Li, Xia Li, Li Zhang, Guangliang Cheng, Jianping Shi, Zhouchen Lin, Shaohua Tan, and Yunhai Tong. Improving semantic segmentation via decoupled body and edge supervision. In *European Conference on Computer Vision*, pages 435–452. Springer, 2020.

[20] Xiangtai Li, Ansheng You, Zhen Zhu, Houlong Zhao, Maoke Yang, Kuiyuan Yang, Shaohua Tan, and Yunhai Tong. Semantic flow for fast and accurate scene parsing. In *European Conference on Computer Vision*, pages 775–793. Springer, 2020.

[21] Xiangtai Li, Houlong Zhao, Lei Han, Yunhai Tong, Shaohua Tan, and Kuiyuan Yang. Gated fully fusion for semantic segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34 (07), pages 11418–11425, 2020.

[22] Di Lin, Dingguo Shen, Siting Shen, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Zigzagnet: Fusing top-down and bottom-up context for object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7490–7499, 2019.

[23] Yun Liu, Ming-Ming Cheng, Xiaowei Hu, Kai Wang, and Xiang Bai. Richer convolutional features for edge detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3000–3009, 2017.

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

[25] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020.

[26] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 4990–4999, 2017.

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[28] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

[29] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7262–7272, 2021.

[30] Zhuo Su, Wenzhe Liu, Zitong Yu, Dewen Hu, Qing Liao, Qi Tian, Matti Pietikäinen, and Li Liu. Pixel difference networks for efficient edge detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5117–5127, 2021.

[31] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5229–5238, 2019.

[32] Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[34] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European Conference on Computer Vision*, pages 108–126. Springer, 2020.

[35] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.

[36] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[37] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021.

[38] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1395–1403, 2015.

[39] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3684–3692, 2018.

[40] Shiqi Yang and Gang Peng. Attention to refine through multi scales for semantic segmentation. In *Pacific Rim Conference on Multimedia*, pages 232–241. Springer, 2018.

[41] Minghao Yin, Zhuliang Yao, Yue Cao, Xiu Li, Zheng Zhang, Stephen Lin, and Han Hu. Disentangled non-local neural networks. In *European Conference on Computer Vision*, pages 191–207. Springer, 2020.

[42] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020.

[43] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[44] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 472–480, 2017.

[45] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412, 2018.

[46] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *European conference on computer vision*, pages 173–190. Springer, 2020.

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

[48] Dong Zhang, Hanwang Zhang, Jinhui Tang, Meng Wang, Xiansheng Hua, and Qianru Sun. Feature pyramid transformer. In *European Conference on Computer Vision*, pages 323–339. Springer, 2020.

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

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

[51] Shuai Zhao, Yang Wang, Zheng Yang, and Deng Cai. Region mutual information loss for semantic segmentation. *Advances in Neural Information Processing Systems*, 32, 2019.