

Object-Contextual Representations for Semantic Segmentation

Yuhui Yuan^{1,2,3}, Xilin Chen^{2,3}, Jingdong Wang¹

¹Microsoft Research Asia

²Institute of Computing Technology, CAS

³University of Chinese Academy of Sciences

{yuhui.yuan, jingdw}@microsoft.com, xlchen@ict.ac.cn

Abstract

In this paper, we address the semantic segmentation problem with a focus on the context aggregation strategy. Our motivation is that the label of a pixel is the category of the object that the pixel belongs to. We present a simple yet effective approach, object-contextual representations, characterizing a pixel by exploiting the representation of the corresponding object class. First, we learn object regions under the supervision of the ground-truth segmentation. Second, we compute the object region representation by aggregating the representations of the pixels lying in the object region. Last, we compute the relation between each pixel and each object region, and augment the representation of each pixel with the object-contextual representation which is a weighted aggregation of all the object region representations according to their relations with the pixel. We empirically demonstrate that the proposed approach achieves competitive performance on various challenging semantic segmentation benchmarks: Cityscapes, ADE20K, LIP, PASCAL-Context, and COCO-Stuff.

1. Introduction

Semantic segmentation is a problem of assigning a class label to each pixel for an image. It is a fundamental topic in computer vision and is critical for various practical tasks such as autonomous driving. Deep convolutional networks since FCN [41] have been the dominant solutions. Various studies have been conducted, including high-resolution representation learning [7, 48], contextual aggregation [66, 6] that is the interest of this paper, and so on.

The context of one position typically refers to a set of positions, e.g., the surrounding pixels. The early study is mainly about the spatial scale of contexts, i.e., the spatial scope. Representative works, such as ASPP [6] and PPM [66], exploit multi-scale contexts. Recently, several works, such as DANet [15], CFNet [63] and OCNNet [60], consider the relations between a position and its contextual

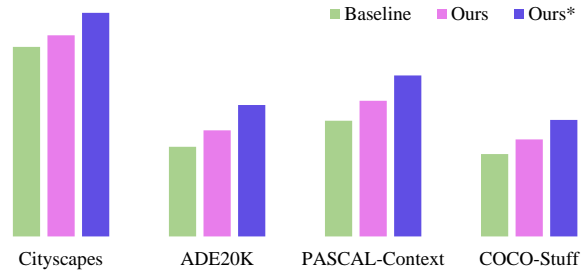


Figure 1: Illustrating the effectiveness of our OCR method. Ours* estimates the ideal object-contextual representations through exploiting the ground-truth. Ours is the performance of our proposed object-contextual representations. The three methods, ours*, ours and the baseline, use the dilated ResNet-101 with output stride 8 as the backbone. The (single-scale) segmentation results are: Cityscapes val: 75.8%, 79.6%, 88.8%; ADE20K val: 39.7%, 44.3%, 54.3%; PASCAL-Context test: 45.8%, 53.3%, 63.7%; COCO-Stuff test: 32.6%, 38.4%, 46.1%.

positions, and aggregate the representations of the contextual positions with higher weights for similar representations.

We propose to investigate the contextual representation scheme along the line of exploring the relation between a position and its context. The motivation is that *the class label assigned to one pixel is the category of the object that the pixel belongs to*. We aim to augment the representation of one pixel by exploiting the representation of the object region of the corresponding class. The empirical study, shown in Figure 1, verifies that such a representation augmentation scheme, when the ground-truth object region is given, dramatically improves the segmentation quality¹.

Our approach consists of three main steps. First, we divide the contextual pixels into a set of soft object regions with each corresponding to a class, i.e., a coarse soft segmentation computed from a deep network (e.g., ResNet [22] or HRNet [48]). Such division is learned under the supervision of the ground-truth segmentation. Second, we estimate

¹See Section 3.4 for more details.

the representation for each object region by aggregating the representations of the pixels in the corresponding object region. Last, we augment the representation of each pixel with the object-contextual representation (OCR). The OCR is the weighted aggregation of all the object region representations with the weights calculated according to the relations between pixels and object regions.

The proposed OCR approach differs from the conventional multi-scale context schemes. Our OCR differentiates the same-object-class contextual pixels from the different-object-class contextual pixels, while the multi-scale context schemes, such as ASPP [6] and PPM [66], do not, and only differentiate the pixels with different spatial positions. Figure 2 provides an example to illustrate the differences between our OCR context and the multi-scale context. On the other hand, our OCR approach is also different from the previous relational context schemes [55, 15, 60, 61, 63]. Our approach structures the contextual pixels into object regions and exploits the relations between pixels and object regions. In contrast, the previous relational context schemes consider the contextual pixels separately and only exploit the relations between pixels and contextual pixels [15, 60, 63] or predict the relations only from pixels without considering the regions [61].

We evaluate our approach on various challenging semantic segmentation benchmarks. Our approach outperforms the multi-scale context schemes, e.g., PSPNet, DeepLabv3, and the recent relational context schemes, e.g., DANet, and the efficiency is also improved. Our approach achieves competitive performance on five benchmarks: 83.7% on Cityscapes test, 45.66% on ADE20K val, 56.65% on LIP val, 56.2% on PASCAL-Context test and 40.5% on COCO-Stuff test.

2. Related Work

Multi-scale context. PSPNet [66] performs regular convolutions on pyramid pooling representations to capture the multi-scale context. The DeepLab series [5, 6] adopt parallel dilated convolutions with different dilation rates (each rate captures the context of a different scale). The recent works [21, 57, 71, 60] propose various extensions, e.g., DenseASPP [57] densifies the dilated rates to cover larger scale ranges. Some other studies [7, 38, 16] construct the encoder-decoder structures to exploit the multi-resolution features as the multi-scale context.

Relational context. DANet [15], CFNet [63] and OCNet [60] augment the representation for each pixel by aggregating the representations of the contextual pixels, where the context consists of all the pixels. Different from the global context [40], these works consider the relation (or similarity) between the pixels, which is based on the self-attention scheme [55, 53], and perform a weighted aggrega-



(a) ASPP

(b) OCR

Figure 2: Illustrating the multi-scale context with the ASPP as an example and the OCR context for the pixel marked with ■. (a) ASPP: The context is a set of sparsely sampled pixels marked with ■, ■, ■. The pixels with different colors correspond to different dilation rates. Those pixels are distributed in both the object region and the background region. (b) Our OCR: The context is expected to be a set of pixels lying in the object (marked with color ■). The image is chosen from ADE20K.

tion with the similarities as the weights.

Double Attention and its related work [8, 61, 9, 36, 34, 59, 32] and ACFNet [61] group the pixels into a set of regions, and then augment the pixel representations by aggregating the region representations with the consideration of their context relations predicted by using the pixel representation.

Our approach is a relational context approach and is related to Double Attention and ACFNet. The differences lie in the region formation and the pixel-region relation computation. Our approach learns the regions with the supervision of the ground-truth segmentation. In contrast, the regions in previous approaches except ACFNet are formed unsupervisedly. On the other hand, the relation between a pixel and a region is computed by considering both the pixel and region representations, while the relation in previous works is only computed from the pixel representation.

Coarse-to-fine segmentation. Various coarse-to-fine segmentation schemes have been developed [14, 17, 29, 51, 25, 28, 70] to gradually refine the segmentation maps from coarse to fine. For example, [29] regards the coarse segmentation map as an additional representation and combines it with the original image or other representations for computing a fine segmentation map.

Our approach in some sense can also be regarded as a coarse-to-fine scheme. The difference lies in that we use the coarse segmentation map for generating a contextual representation instead of directly used as an extra representation. We compare our approach with the conventional coarse-to-fine schemes in the supplementary material.

Region-wise segmentation. The region-wise segmentation scheme [1, 2, 20, 19, 56, 44, 2, 52] organizes the pixels into a set of regions (usually super-pixels), and then classifies each region to get the image segmentation result. Our approach does not classify each region for segmentation and instead uses the region to learn a better representation for the pixel, which leads to better pixel labeling.

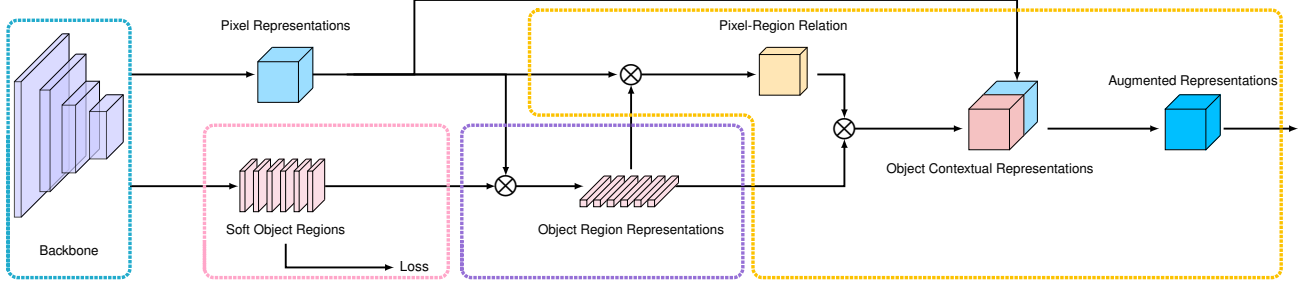


Figure 3: The pipeline of our approach. (i) form the soft object regions in the *pink dashed box*. (ii) estimate the object region representations in the *purple dashed box*; (iii) compute the object contextual representations and the augmented representations in the *orange dashed box*. See Section 3.2 and 3.3 for more details.

3. Approach

Semantic segmentation is a problem of assigning one label l_i to each pixel p_i of an image I , where l_i is one of K different classes.

3.1. Background

Multi-scale context. The ASPP [5] module captures the multi-scale context information by performing several parallel dilated convolutions with different dilation rates [5, 6, 58]:

$$\mathbf{y}_i^d = \sum_{\mathbf{p}_s = \mathbf{p}_i + d\Delta_t} \mathbf{K}_t^d \mathbf{x}_s. \quad (1)$$

Here, $\mathbf{p}_s = \mathbf{p}_i + d\Delta_t$ is the s th sampled position for the dilation convolution with the dilation rate d (e.g., $d = 12, 24, 36$ in DeepLabv3 [6]) at the position \mathbf{p}_i . t is the position index for a convolution, e.g., $\{\Delta_t = (\Delta_w, \Delta_h) | \Delta_w = -1, 0, 1, \Delta_h = -1, 0, 1\}$ for a 3×3 convolution. \mathbf{x}_s is the representation at \mathbf{p}_s . \mathbf{y}_i^d is the output representation at \mathbf{p}_i for the d th dilated convolution. \mathbf{K}_t^d is the kernel parameter at position t for the d th dilated convolution. The output multi-scale contextual representation is the concatenation of the representations output by the parallel dilated convolutions.

The multi-scale context scheme based on dilated convolutions captures the contexts of multiple scales without losing the resolution. The pyramid pooling module in PSP-Net [66] performs regular convolutions on representations of different scales, and also captures the contexts of multiple scales but loses the resolution for large scale contexts.

Relational context. The relational context scheme [15, 60, 63] computes the context for each pixel by considering the relations:

$$\mathbf{y}_i = \rho\left(\sum_{s \in \mathcal{I}} w_{is} \delta(\mathbf{x}_s)\right), \quad (2)$$

where \mathcal{I} refers to the set of pixels in the image, w_{is} is the relation between \mathbf{x}_i and \mathbf{x}_s , and may be predicted only from

\mathbf{x}_i or computed from \mathbf{x}_i and \mathbf{x}_s . $\delta(\cdot)$ and $\rho(\cdot)$ are two different transform functions as done in self-attention [53]. The global context scheme [40] is a special case of relational context with $w_{is} = \frac{1}{|\mathcal{I}|}$.

3.2. Formulation

The class label l_i for pixel p_i is essentially the label of the object that pixel p_i lies in. Motivated by this, we present an object-contextual representation approach, characterizing each pixel by exploiting the corresponding object representation.

The proposed object-contextual representation scheme (1) structurizes all the pixels in image I into K soft object regions, (2) represents each object region as \mathbf{f}_k by aggregating the representations of all the pixels in the k th object region, and (3) augments the representation for each pixel by aggregating the K object region representations with consideration of its relations with all the object regions:

$$\mathbf{y}_i = \rho\left(\sum_{k=1}^K w_{ik} \delta(\mathbf{f}_k)\right), \quad (3)$$

where \mathbf{f}_k is the representation of the k th object region, w_{ik} is the relation between the i th pixel and the k th object region. $\delta(\cdot)$ and $\rho(\cdot)$ are transformation functions.

Soft object regions. We partition the image I into K soft object regions $\{\mathbf{M}_1, \mathbf{M}_2, \dots, \mathbf{M}_K\}$. Each object region \mathbf{M}_k corresponds to the class k , and is represented by a 2D map (or coarse segmentation map), where each entry indicates the degree that the corresponding pixel belongs to the class k .

We compute the K object regions from an intermediate representation output from a backbone (e.g., ResNet or HR-Net). During training, we learn the object region generator under the supervision from the ground-truth segmentation using the cross-entropy loss.

Object region representations. We aggregate the representations of all the pixels weighted by their degrees be-

longing to the k th object region, forming the k th object region representation:

$$\mathbf{f}_k = \sum_{i \in \mathcal{I}} \tilde{m}_{ki} \mathbf{x}_i. \quad (4)$$

Here, \mathbf{x}_i is the representation of pixel p_i . \tilde{m}_{ki} is the normalized degree for pixel p_i belonging to the k th object region. We use spatial softmax to normalize each object region \mathbf{M}_k .

Object contextual representations. We compute the relation between each pixel and each object region as below:

$$w_{ik} = \frac{e^{\kappa(\mathbf{x}_i, \mathbf{f}_k)}}{\sum_{j=1}^K e^{\kappa(\mathbf{x}_i, \mathbf{f}_j)}}. \quad (5)$$

Here, $\kappa(\mathbf{x}, \mathbf{f}) = \phi(\mathbf{x})^\top \psi(\mathbf{f})$ is the unnormalized relation function, $\phi(\cdot)$ and $\psi(\cdot)$ are two transformation functions implemented by 1×1 conv \rightarrow BN \rightarrow ReLU. This is inspired by self-attention [53] for a better relation estimation.

The object contextual representation \mathbf{y}_i for pixel p_i is computed according to Equation 3. In this equation, $\delta(\cdot)$ and $\rho(\cdot)$ are both transformation functions implemented by 1×1 conv \rightarrow BN \rightarrow ReLU, and this follows non-local networks [55].

Augmented representations. The final representation for pixel p_i is updated as the aggregation of two parts, (1) the original representation \mathbf{x}_i , and (2) the object contextual representation \mathbf{y}_i :

$$\mathbf{z}_i = g([\mathbf{x}_i^\top \mathbf{y}_i^\top]^\top). \quad (6)$$

where $g(\cdot)$ is a transform function used to fuse the original representation and the object contextual representation, implemented by 1×1 conv \rightarrow BN \rightarrow ReLU. The whole pipeline of our approach is illustrated in Figure 3.

Comments: Some recent works, e.g., Double Attention [8] and ACFNet [61], can be formulated similarly to Equation 3, but differ from our approach in some aspects. For example, the region formed in Double Attention do not correspond to an object class, and the relation in ACFNet [61] is computed only from the pixel representation w/o using the object region representation.

3.3. Architecture

Backbone. We use the dilated ResNet-101 [22] (with output stride 8) or HRNet-W48 [48] (with output stride 4) as the backbone. For dilated ResNet-101, there are two representations input to the OCR module. The first representation from Stage 3 is for predicting coarse segmentation (object regions). The other representation from Stage 4 goes through a 3×3 convolution (512 output channels), and then is fed into the OCR module. For HRNet-W48, we only use the final representation as the input to the OCR module.

Table 1: Influence of the object region supervision and the pixel-region relation estimation scheme.

Object region supervision		Pixel-region relations		
w/o supervision	w/ supervision	DA scheme	ACF scheme	Ours
77.31%	79.58%	79.01%	78.02%	79.58%

OCR module. We implement the above formulation of our approach as the OCR module, as illustrated in Figure 3. We use a linear function (a 1×1 convolution) to predict the coarse segmentation (soft object region) supervised with a pixel-wise cross-entropy loss. All the transform functions, $\psi(\cdot)$, $\phi(\cdot)$, $\delta(\cdot)$, $\rho(\cdot)$, and $g(\cdot)$, are implemented as 1×1 conv \rightarrow BN \rightarrow ReLU, and the first three output 256 channels and the last two output 512 channels. We predict the final segmentation from the final representation using a linear function and we also apply a pixel-wise cross-entropy loss on the final segmentation prediction.

3.4. Empirical Analysis

We conduct the empirical analysis experiments using the dilated ResNet-101 as the backbone on Cityscapes val.

Object region supervision. We study the influence of the object region supervision. We modify our approach through removing the supervision (i.e., loss) on the soft object regions (within the pink dashed box in Figure 3), and adding another auxiliary loss in the stage-3 of ResNet-101. We keep all the other settings the same and report the results in the left-most 2 columns of Table 1. We can see that the supervision for forming the object regions is crucial for the performance.

Pixel-region relations. We compare our approach with other two mechanisms that do not use the region representation for estimating the pixel-region relations: (i) Double-Attention [8] uses the pixel representation to predict the relation; (ii) ACFNet [61] directly uses one intermediate segmentation map to indicate the relations. We use DA scheme and ACF scheme to represent the above two mechanisms. We implement both methods by ourselves and only use the dilated ResNet-101 as the backbone without using multi-scale contexts (the results of ACFNet is improved by using ASPP [61]). The comparison in Table 1 shows that our approach gets superior performance. The reason is that we exploit the pixel representation as well as the region representation for computing the relations. The region representation is able to characterize the object in the specific image, and thus the relation is more accurate for the specific image than that only using the pixel representation.

Ground-truth OCR. We study the segmentation performance using the ground-truth segmentation to form the object regions and the pixel-region relations, called ours*, to justify our motivation. (i) Object region formation using the ground-truth: set the confidence of pixel i belonging to

k th object region $m_{ki} = 1$ if the ground-truth label $l_i \equiv k$ and $m_{ki} = 0$ otherwise. (ii) Pixel-region relation computation using the ground-truth: set the pixel-region relation $w_{ik} = 1$ if the ground-truth label $l_i \equiv k$ and $w_{ik} = 0$ otherwise. We have illustrated the detailed results of ours* on four different benchmarks in Figure 1.

4. Experiments

4.1. Datasets

Cityscapes. The Cityscapes dataset [11] is tasked for urban scene understanding. There are totally 30 classes and only 19 classes are used for parsing evaluation. The dataset contains 5K high quality pixel-level finely annotated images and 20K coarsely annotated images. The finely annotated 5K images are divided into 2, 975/500/1, 525 images for training, validation and testing.

ADE20K. The ADE20K dataset [68] is used in ImageNet scene parsing challenge 2016. There are 150 classes and diverse scenes with 1, 038 image-level labels. The dataset is divided into 20K/2K/3K images for training, validation and testing.

LIP. The LIP dataset [18] is used in the LIP challenge 2016 for single human parsing task. There are about 50K images with 20 classes (19 semantic human part classes and 1 background class). The training, validation, and test sets consist of 30K, 10K, 10K images respectively.

PASCAL-Context. The PASCAL-Context dataset [43] is a challenging scene parsing dataset that contains 59 semantic classes and 1 background class. The training set and test set consist of 4, 998 and 5, 105 images respectively.

COCO-Stuff. The COCO-Stuff dataset [3] is a challenging scene parsing dataset that contains 171 semantic classes. The training set and test set consist of 9K and 1K images respectively.

4.2. Implementation Details

Training setting. We initialize the backbones using the model pretrained on ImageNet and the OCR module randomly. We perform the polynomial learning rate policy with factor $(1 - (\frac{iter}{iter_{max}})^{0.9})$, the weight on the final loss as 1, the weight on the loss used to supervise the object region estimation (or auxiliary loss) as 0.4. We use INPLACE-ABN^{sync} [47] to synchronize the mean and standard-deviation of BN across multiple GPUs. For the data augmentation, we perform random flipping horizontally, random scaling in the range of $[0.5, 2]$ and random brightness jittering within the range of $[-10, 10]$. We perform the same training settings for the reproduced approaches, e.g., PPM, ASPP, to ensure the fairness. We follow the previous works [6, 62, 66] for setting up the training for the benchmark datasets.

Table 2: Comparison with multi-scale context scheme.

Method	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP
PPM [66]	78.4%	81.2%	43.29%	—
ASPP [6]	—	81.3%	—	—
PPM (Our impl.)	80.3%	81.6%	44.50%	54.76%
ASPP (Our impl.)	81.0%	81.7%	44.60%	55.01%
OCR	81.8%	82.4%	45.28%	55.60%

Table 3: Comparison with relational context scheme.

Method	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP
CC-Attention [24]	81.4%	-	45.22%	-
DANet [15]	81.5%	-	-	-
Self Attention (Our impl.)	81.1%	82.0%	44.75%	55.15%
Double Attention (Our impl.)	81.2%	82.0%	44.81%	55.12%
OCR	81.8%	82.4%	45.28%	55.60%

■ **Cityscapes:** We set the initial learning rate as 0.01, weight decay as 0.0005, crop size as 769×769 and batch size as 8. For the experiments evaluated on val/test, we set training iterations as 40K/100K on train/train+val separately. For the experiments augmented with extra data, we finetune our model on coarse/Mapillary for 50K iterations and continue finetune our model on train+val for 20K iterations following [31].

■ **ADE20K:** We set the initial learning rate as 0.02, weight decay as 0.0001, crop size as 520×520 , batch size as 16 and training iterations as 150K if not specified.

■ **LIP:** We set the initial learning rate as 0.007, weight decay as 0.0005, crop size as 473×473 , batch size as 32 and training iterations as 100K if not specified.

■ **PASCAL-Context:** We set the initial learning rate as 0.001, weight decay as 0.0001, crop size as 520×520 , batch size as 16 and training iterations as 30K if not specified.

■ **COCO-Stuff:** We set the initial learning rate as 0.001, weight decay as 0.0001, crop size as 520×520 , batch size as 16 and training iterations as 60K if not specified.

4.3. Comparison with Existing Context Schemes

We conduct the experiments using the dilated ResNet-101 as the backbone and use the same training/testing settings to ensure the fairness.

Multi-scale contexts. We compare our OCR with the multi-scale context schemes including PPM [66] and ASPP [6] on three benchmarks including Cityscapes test, ADE20K val and LIP val. We present the results in Table 2. Our reproduced PPM/ASPP outperforms the originally reported numbers in [66, 6]. From Table 2, it can be seen that our OCR outperforms both multi-scale context schemes by a large margin.

Relational contexts. We compare our OCR with various relational context schemes including Self-Attention [53, 55],

Table 4: Complexity comparison when processing input feature map of size $[1 \times 2048 \times 128 \times 128]$ during inference. The numbers are obtained on a single P40 GPU with CUDA 10.0. The numbers are the smaller the better.

Method	Parameters▲	Memory▲	FLOPs ▲	Time▲
PPM (Our impl.)	23.1M	792M	619G	99ms
ASPP (Our impl.)	15.5M	284M	492G	97ms
DANet (Our impl.)	10.6M	2339M	1110G	121ms
CC-Attention (Our impl.)	10.6M	427M	804G	131ms
Self-Attention (Our impl.)	10.5M	2168M	619G	96ms
Double Attention (Our impl.)	10.2M	209M	338G	46ms
OCR	10.5M	202M	340G	45ms

Criss-Cross attention [24] (CC-Attention), DANet [15] and Double Attention [8] on the same three benchmarks including Cityscapes *test*, ADE20K *val* and LIP *val*. For the reproduced Double Attention, we fine-tune the number of the regions and choose 64 with the best performance. More detailed analysis and comparisons are illustrated in the supplementary material. According to the results in Table 3, it can be seen that our OCR outperforms these relational context schemes.

Complexity. We compare the efficiency of our OCR with the efficiencies of the multi-scale context schemes and the relational context schemes. We measure the increased parameters, GPU memory, computation complexity (measured by the number of FLOPs) and inference time that are introduced by the context modules, and do not count the complexity from the backbones. The comparison in Table 4 shows the superiority of the proposed OCR scheme.

- *Parameters:* Most relational context schemes require less parameters compared with the multi-scale context schemes. For example, our OCR only requires less than 50% of the parameters of PPM.
- *Memory:* Both our OCR and Double Attention perform much better compared with the other approaches (e.g., DANet, PPM). For example, DANet requires nearly $10 \times$ larger GPU memory than our OCR. Besides, our OCR only requires 50% GPU memory of the recent CC-Attention.
- *FLOPs:* Our OCR only requires 50%/30% of the PPM’s FLOPs/DANet’s FLOPs separately.
- *Running time:* Our OCR is more than $2 \times$ faster than all the other approaches except Double Attention.

4.4. Comparison with State-of-the-Art

Considering that different approaches perform improvements on different baselines to achieve the best performance, we categorize the existing works to two groups according to the baselines that they apply: (i) *simple baseline*: dilated ResNet-101 with stride 8; (ii) *advanced baseline*: PSPNet, DeepLabv3, multi-grid (MG), encoder-decoder structures that achieve higher resolution outputs with stride 4 or stronger backbones such as WideResNet-38, Xception-71 and HRNet.

For fair comparison with the two groups fairly, we perform our OCR on a simple baseline (dilated ResNet-101 with stride 8) and an advanced baseline (HRNet-W48 with stride 4). We present all the results in Table 5 and illustrate the comparison details on each benchmark separately as follows.

Cityscapes. Compared with the methods based on the simple baseline on Cityscape *test* w/o using the coarse data, our approach achieves the best performance 81.8%, which is already comparable with some methods based on the advanced baselines, e.g. DANet, ACFNet. Our approach achieves better performance 82.4% through exploiting the coarsely annotated images for training.

For comparison with the approaches based on the advanced baselines, we perform our OCR on the HRNet-W48, combine our OCR with ASPP and fine-tune our model on the Mapillary dataset [44]. Our approach achieves 83.7% on Cityscapes *test* with a single model entry. Besides, we perform PPM and ASPP on HRNet-W48 separately and empirically find that directly applying either PPM or ASPP does not improve the performance and even degrades the performance, while our OCR consistently improves the performance.

ADE20K. From Table 5, it can be seen that our OCR achieves competitive performance (45.28% and 45.66%) compared with most of the previous approaches based on both simple baselines and advanced baselines. For example, the ACFNet [21] exploits both the multi-scale context and relational context to achieve higher performance. The very recent ACNet [16] achieves the best performance through combining richer local and global contexts.

LIP. Our approach achieves the best performance 55.60% on LIP *val* based on the simple baselines. Applying the stronger backbone HRNetV2-W48 further improves the performance to 56.65%, which outperforms the previous approaches. The very recent work CNIF [54] achieves the best performance (56.93%) through injecting the hierarchical structure knowledge of human parts. Our approach potentially benefit from such hierarchical structural knowledge. All the results are based on only flip testing without multi-scale testing².

PASCAL-Context. We evaluate the performance over 59 categories following [48]. It can be seen that our approach outperforms both the previous best methods based on simple baselines and the previous best methods based on advanced baselines. The HRNet-W48 + OCR approach achieves the best performance 56.2%, significantly outperforming the second best, e.g., ACPNet (54.7%) and ACNet (54.1%).

COCO-Stuff. It can be seen that our approach achieves the

²Only few methods adopt multi-scale testing. For example, CNIF [54] gets the improved performance from 56.93% to 57.74%.

Table 5: Comparison with state-of-the-art on Cityscapes test, ADE20K val, LIP val, PASCAL-Context test, COCO-Stuff test. We use M to represent multi-scale context and R to represent relational context. Red, Green, Blue represent the top-3 results.

Method	Baseline	Stride	Context schemes	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP	PASCAL-Context	COCO-Stuff
Simple baselines									
PSPNet [66]	ResNet-101	8×	M	78.4	81.2	43.29	-	47.8	-
DeepLabv3 [6]	ResNet-101	8×	M	-	81.3	-	-	-	-
PSANet [67]	ResNet-101	8×	R	80.1	81.4	43.77	-	-	-
SAC [65]	ResNet-101	8×	M	78.1	-	44.30	-	-	-
AAF [26]	ResNet-101	8×	R	79.1	-	-	-	-	-
DSSPN [37]	ResNet-101	8×	-	77.8	-	43.68	-	-	38.9
DepthSeg [27]	ResNet-101	8×	-	78.2	-	-	-	-	-
MMAN [42]	ResNet-101	8×	-	-	-	-	46.81	-	-
JPPNet [35]	ResNet-101	8×	M	-	-	-	51.37	-	-
EncNet [62]	ResNet-101	8×	-	-	-	44.65	-	51.7	-
GCU [34]	ResNet-101	8×	R	-	-	44.81	-	-	-
APCNet [21]	ResNet-101	8×	M,R	-	-	45.38	-	54.7	-
CFNet [63]	ResNet-101	8×	R	79.6	-	44.89	-	54.0	-
BFP [12]	ResNet-101	8×	R	81.4	-	-	-	53.6	-
CCNet [24]	ResNet-101	8×	R	81.4	-	45.22	-	-	-
Asymmetric NL [71]	ResNet-101	8×	M,R	81.3	-	45.24	-	52.8	-
OCR	ResNet-101	8×	R	81.8	82.4	45.28	55.60	54.8	39.5
Advanced baselines									
DenseASPP [57]	DenseNet-161	8×	M	80.6	-	-	-	-	-
DANet [15]	ResNet-101 + MG	8×	R	81.5	-	45.22	-	52.6	39.7
DGCNet [64]	ResNet-101 + MG	8×	R	82.0	-	-	-	53.7	-
EMANet [33]	ResNet-101 + MG	8×	R	-	-	-	-	53.1	39.9
SeENet [46]	ResNet-101 + ASPP	8×	M	81.2	-	-	-	-	-
SGR [36]	ResNet-101 + ASPP	8×	R	-	-	44.32	-	52.5	39.1
OCNet [60]	ResNet-101 + ASPP	8×	M,R	81.7	-	45.45	54.72	-	-
ACFNet [61]	ResNet-101 + ASPP	8×	M,R	81.8	-	-	-	-	-
CNIF [54]	ResNet-101 + ASPP	8×	M	-	-	-	56.93	-	-
GALD [31]	ResNet-101 + ASPP	8×	M,R	81.8	82.9	-	-	-	-
GALD (w/ Mapillary) [31]	ResNet-101 + CGNL + MG	8×	M,R	-	83.3	-	-	-	-
Mapillary Research [47]	WideResNet-38 + ASPP	8×	M	-	82.0	-	-	-	-
GSCNN (w/ Mapillary) [49]	WideResNet-38 + ASPP	8×	M	82.8	-	-	-	-	-
SPGNet [10]	2× ResNet-50	4×	-	81.1	-	-	-	-	-
ZigZagNet [38]	ResNet-101	4×	M	-	-	-	-	52.1	-
SVCNet [13]	ResNet-101	4×	R	81.0	-	-	-	53.2	39.6
ACNet [16]	ResNet-101 + MG	4×	M,R	82.3	-	45.90	-	54.1	40.1
CE2P [39]	ResNet-101 + PPM	4×	M	-	-	-	53.10	-	-
VPLR (w/ Mapillary, w/ Video) [69]	WideResNet-38 + ASPP	4×	M	-	83.5	-	-	-	-
DeepLabv3+ [7]	Xception-71	4×	M	-	82.1	-	-	-	-
DPC [4]	Xception-71	4×	M	82.7	-	-	-	-	-
DUpsampling [50]	Xception-71	4×	M	-	-	-	-	52.5	-
HRNet [48]	HRNetV2-W48	4×	-	81.6	-	-	55.90	54.0	-
OCR	HRNetV2-W48	4×	R	82.4	83.0	45.66	56.65	56.2	40.5
OCR (w/ Mapillary)	HRNetV2-W48 + ASPP	4×	M,R	83.2	83.7	-	-	-	-

best performance, 39.5% based ResNet-101 and 40.5 based on HRNetV2-48.

Qualitative Results. We illustrate the qualitative results in Figure 4 on the 5 benchmarks. We use white dashed boxes to mark the hard regions that are well-classified by our approach but mis-classified by the baseline.

5. Conclusions

In this work, we present an object-contextual representation approach for semantic segmentation. The main reason for the success is that the label of a pixel is the label

of the object that the pixel lies in and the pixel representation is strengthened by characterizing each pixel with the corresponding object region representation. We empirically show that our approach brings consistent improvements on various benchmarks.

References

- [1] Pablo Arbeláez, Bharath Hariharan, Chunhui Gu, Saurabh Gupta, Lubomir Bourdev, and Jitendra Malik. Semantic segmentation using regions and parts. 2012. 2
- [2] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Region-

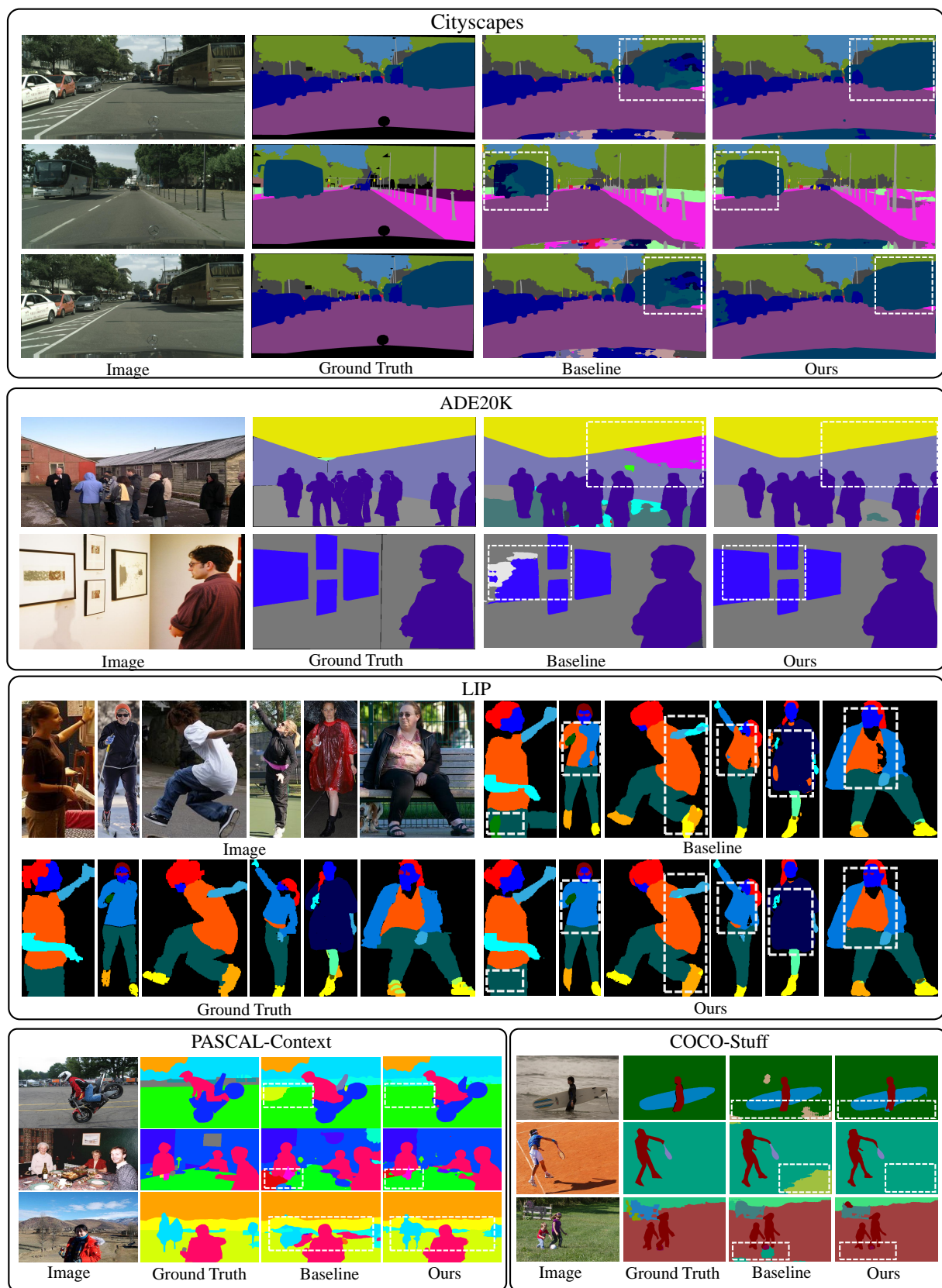


Figure 4: Qualitative comparison between dilated ResNet-101 (baseline) and dilated ResNet-101 + OCR (ours) on the 5 benchmarks. We mark the improved regions with white dashed boxes.

- based semantic segmentation with end-to-end training. In *ECCV*, 2016. 2
- [3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *CVPR*, 2018. 5
- [4] Liang-Chieh Chen, Maxwell Collins, Yukun Zhu, George Papandreou, Barret Zoph, Florian Schroff, Hartwig Adam, and Jon Shlens. Searching for efficient multi-scale architectures for dense image prediction. In *NIPS*, 2018. 7
- [5] 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. *PAMI*, 2018. 2, 3
- [6] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv:1706.05587*, 2017. 1, 2, 3, 5, 7
- [7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 1, 2, 7
- [8] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. A²-nets: Double attention networks. In *NIPS*, 2018. 2, 4, 6, 11
- [9] Yunpeng Chen, Marcus Rohrbach, Zhicheng Yan, Shuicheng Yan, Jiashi Feng, and Yannis Kalantidis. Graph-based global reasoning networks. *arXiv:1811.12814*, 2018. 2
- [10] Bowen Cheng, Liang-Chieh Chen, Yunchao Wei, Yukun Zhu, Zilong Huang, Jinjun Xiong, Thomas S. Huang, Wen-Mei Hwu, and Honghui Shi. Spynet: Semantic prediction guidance for scene parsing. In *ICCV*, 2019. 7
- [11] 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 *CVPR*, 2016. 5
- [12] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-aware feature propagation for scene segmentation. *ICCV*, 2019. 7
- [13] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic correlation promoted shape-variant context for segmentation. In *CVPR*, 2019. 7
- [14] Mihai Fieraru, Anna Khoreva, Leonid Pishchulin, and Bernt Schiele. Learning to refine human pose estimation. In *CVPRW*, 2018. 2, 11
- [15] Jun Fu, Jing Liu, Haijie Tian, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. *arXiv:1809.02983*, 2018. 1, 2, 3, 5, 6, 7
- [16] Jun Fu, Jing Liu, Yuhang Wang, Yong Li, Yongjun Bao, Jinhui Tang, and Hanqing Lu. Adaptive context network for scene parsing. In *ICCV*, 2019. 2, 6, 7
- [17] Spyros Gidaris and Nikos Komodakis. Detect, replace, refine: Deep structured prediction for pixel wise labeling. In *CVPR*, 2017. 2, 11
- [18] Ke Gong, Xiaodan Liang, Dongyu Zhang, Xiaohui Shen, and Liang Lin. Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing. In *CVPR*, 2017. 5
- [19] Stephen Gould, Richard Fulton, and Daphne Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009. 2
- [20] Chunhui Gu, Joseph J Lim, Pablo Arbelaez, and Jitendra Malik. Recognition using regions. In *CVPR*, 2009. 2
- [21] Junjun He, Zhongying Deng, Lei Zhou, Yali Wang, and Yu Qiao. Adaptive pyramid context network for semantic segmentation. In *CVPR*, 2019. 2, 6, 7
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 4
- [23] Yu-Hui Huang, Xu Jia, Stamatis Georgoulis, Tinne Tuytelaars, and Luc Van Gool. Error correction for dense semantic image labeling. In *CVPRW*, 2018. 11
- [24] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 5, 6, 7
- [25] Md Amirul Islam, Shujon Naha, Mrigank Rochan, Neil Bruce, and Yang Wang. Label refinement network for coarse-to-fine semantic segmentation. *arXiv:1703.00551*, 2017. 2
- [26] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X. Yu. Adaptive affinity fields for semantic segmentation. In *ECCV*, 2018. 7
- [27] Shu Kong and Charles C. Fowlkes. Recurrent scene parsing with perspective understanding in the loop. In *CVPR*, 2018. 7
- [28] Weicheng Kuo, Anelia Angelova, Jitendra Malik, and Tsung-Yi Lin. Shapemask: Learning to segment novel objects by refining shape priors. *ICCV*, 2019. 2
- [29] Ke Li, Bharath Hariharan, and Jitendra Malik. Iterative instance segmentation. In *CVPR*, 2016. 2, 11
- [30] Xiaoxiao Li, Ziwei Liu, Ping Luo, Chen Change Loy, and Xiaoou Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *CVPR*, 2017. 11
- [31] Xiangtai Li, Li Zhang, Ansheng You, Maoke Yang, Kuiyuan Yang, and Yunhai Tong. Global aggregation then local distribution in fully convolutional networks. *BMVC*, 2019. 5, 7
- [32] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu. Expectation-maximization attention networks for semantic segmentation. In *ICCV*, 2019. 2
- [33] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu. Expectation-maximization attention networks for semantic segmentation. In *ICCV*, 2019. 7
- [34] Yin Li and Abhinav Gupta. Beyond grids: Learning graph representations for visual recognition. In *NIPS*, 2018. 2, 7
- [35] Xiaodan Liang, Ke Gong, Xiaohui Shen, and Liang Lin. Look into person: Joint body parsing & pose estimation network and a new benchmark. *PAMI*, 2018. 7
- [36] Xiaodan Liang, Zhiting Hu, Hao Zhang, Liang Lin, and Eric P Xing. Symbolic graph reasoning meets convolutions. In *NIPS*, 2018. 2, 7
- [37] Xiaodan Liang, Hongfei Zhou, and Eric Xing. Dynamic-structured semantic propagation network. In *CVPR*, 2018. 7

- [38] 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 *CVPR*, 2019. 2, 7
- [39] Ting Liu, Tao Ruan, Zilong Huang, Yunchao Wei, Shikui Wei, Yao Zhao, and Thomas Huang. Devil in the details: Towards accurate single and multiple human parsing. *arXiv:1809.05996*, 2018. 7
- [40] Wei Liu, Andrew Rabinovich, and Alexander C Berg. Paraset: Looking wider to see better. *arXiv:1506.04579*, 2015. 2, 3
- [41] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1
- [42] Yawei Luo, Zhedong Zheng, Liang Zheng, G Tao, Y Junqing, and Yi Yang. Macro-micro adversarial network for human parsing. In *ECCV*, 2018. 7
- [43] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014. 5
- [44] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Buló, and Peter Kotschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *CVPR*, 2017. 2, 6
- [45] Ishan Nigam, Chen Huang, and Deva Ramanan. Ensemble knowledge transfer for semantic segmentation. In *WACV*, 2018. 11
- [46] Yanwei Pang, Yazhao Li, Jianbing Shen, and Ling Shao. Towards bridging semantic gap to improve semantic segmentation. In *ICCV*, 2019. 7
- [47] Samuel Rota Buló, Lorenzo Porzi, and Peter Kotschieder. In-place activated batchnorm for memory-optimized training of dnns. In *CVPR*, 2018. 5, 7
- [48] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. *arXiv:1904.04514*, 2019. 1, 4, 6, 7
- [49] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. *ICCV*, 2019. 7
- [50] Zhi Tian, Tong He, Chunhua Shen, and Youliang Yan. Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation. In *CVPR*, 2019. 7
- [51] Zhuowen Tu and Xiang Bai. Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *PAMI*, 2010. 2, 11
- [52] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 2013. 2
- [53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 2, 3, 4, 5
- [54] Wenguan Wang, Zhijie Zhang, Siyuan Qi, Jianbing Shen, Yanwei Pang, and Ling Shao. Learning compositional neural information fusion for human parsing. In *ICCV*, 2019. 6, 7
- [55] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 2, 4, 5
- [56] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *CVPR*, 2017. 2
- [57] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *CVPR*, 2018. 2, 7
- [58] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *ICLR*, 2016. 3
- [59] Kaiyu Yue, Ming Sun, Yuchen Yuan, Feng Zhou, Errui Ding, and Fuxin Xu. Compact generalized non-local network. In *NIPS*, 2018. 2
- [60] Yuan Yuhui and Wang Jingdong. Ocnet: Object context network for scene parsing. *arXiv:1809.00916*, 2018. 1, 2, 3, 7
- [61] Fan Zhang, Yanqin Chen, Zhihang Li, Zhibin Hong, Jingtuo Liu, Feifei Ma, Junyu Han, and Errui Ding. Acfnnet: Attentional class feature network for semantic segmentation. In *ICCV*, 2019. 2, 4, 7
- [62] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Amrith Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *CVPR*, 2018. 5, 7
- [63] Hang Zhang, Han Zhang, Chenguang Wang, and Junyuan Xie. Co-occurrent features in semantic segmentation. In *CVPR*, 2019. 1, 2, 3, 7
- [64] Li Zhang, Xiangtai Li, Anurag Arnab, Kuiyuan Yang, Yunhai Tong, and Philip HS Torr. Dual graph convolutional network for semantic segmentation. *BMVC*, 2019. 7
- [65] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Scale-adaptive convolutions for scene parsing. In *ICCV*, 2017. 7
- [66] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017. 1, 2, 3, 5, 7
- [67] Hengshuang Zhao, Zhang Yi, Liu Shu, Shi Jianping, Chen Change Loy, Lin Dahua, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. *ECCV*, 2018. 7
- [68] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 5
- [69] Yi Zhu, Karan Sapra, Fitsum A Reda, Kevin J Shih, Shawn Newsam, Andrew Tao, and Bryan Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *CVPR*, 2019. 7
- [70] Zhuotun Zhu, Yingda Xia, Wei Shen, Elliot Fishman, and Alan Yuille. A 3d coarse-to-fine framework for volumetric medical image segmentation. In *3DV*, 2018. 2
- [71] Zhen Zhu, Mengde Xu, Song Bai, Tengpeng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In *ICCV*, 2019. 2, 7

6. Supplementary Material

In Section 6.1, we compare our approach to the existing coarse-to-fine approaches. In Section 6.2, we study the influence of the region numbers and illustrate the qualitative results with Double Attention. In Section 6.3, we apply our OCR on MobileNetV2 to verify the effectiveness of our approach.

6.1. Comparison with Coarse-to-fine Schemes

Many existing studies [14, 17, 29, 51] have exploited various coarse-to-fine schemes to use the coarse segmentation results to boost the final segmentation results. We mainly compare OCR with two popular mechanisms including:

- label-refinement [23, 17]: combine the input image or feature map with a coarse prediction to predict the refined label map. We concatenate the coarse segmentation maps with the feature map output from ResNet-101 Stage 4 and apply the final classifier on the concatenated feature map to predict the refined segmentation maps.
- label-ensemble [30, 45]: ensemble the coarse segmentation maps with the fine segmentation maps directly. We directly use the weighted sum of the coarse segmentation map and the fine segmentation map as the final refined prediction.

Besides, we also report the performance with only the coarse segmentation map (prediction from the ResNet Stage 3) and with only the fine segmentation map (prediction from the ResNet Stage 4). We choose the dilated ResNet-101 as our baseline. According to the results in Table 6, it can be seen that our OCR outperforms all the other coarse-to-fine approaches by a large margin.

6.2. Ablation Study of Double Attention

6.2.1 Number of Regions

We fine-tune the number of regions within Double Attention [8] method and report the results on Cityscapes val in Table 7. We choose $K=64$ if not specified. Besides, it can be seen that our approach (with fixed number of regions) consistently outperforms the Double Attention with different region numbers.

6.2.2 Qualitative Results

We visualize the predicted regions with Double Attention and the object regions predicted with OCR in Figure 5. It can be seen that the predicted object regions with OCR all correspond to explicit semantic meaning, e.g., road, sidewalk and car category separately, while the predicted regions with Double Attention mainly highlight the contour pixels without specific semantic meaning, which might be the main advantages of our approach.

Table 6: Comparison with other coarse-to-fine mechanisms. All the results are evaluated on Cityscapes val.

Method	Coarse. seg	Fine. seg	mIoU (%)
baseline	✓	✗	73.90
baseline	✗	✓	75.80
label-ensemble	✓	✓	76.20
label-refinement	✓	✓	77.10
OCR	✓	✓	79.58

Table 7: Influence of K within Double Attention. K is the number of regions. K is exact the number of categories for our OCR.

	Double Attention					OCR
# of regions	K=8	K=16	K=32	K=64	K=128	K=19
mIoU	78.52	78.49	78.53	<u>78.65</u>	77.43	79.58

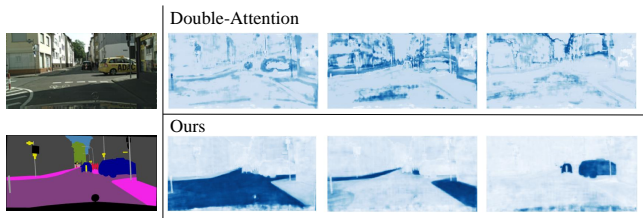


Figure 5: We randomly choose an image and its ground-truth segmentation map from Cityscapes val. The first row illustrates 3 regions predicted with Double Attention and the second row illustrates 3 object regions generated with our OCR. It can be seen that OCR based object regions are more reliable compared to the Double Attention.

Table 8: MobileNetV2 + OCR: Speed (measured by FPS) is tested on P40 GPU with input image of size 1024×512 .

Method	FPS	Cityscapes val mIoU
MobileNetV2	31	69.50%
MobileNetV2 + OCR	28	74.18%

6.3. Application to MobileNetV2

We apply the OCR on MobileNetV2 and report the performance in Table 8. Specifically, we train the MobileNetV2 following the same training settings except changing the batch size as 16 and the training iterations as 100K. It can be seen that our OCR significantly improves the segmentation performance on the Cityscapes val while slightly increases the FPS.