Segmentation Transformer: Object-Contextual Representations for Semantic Segmentation

Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang

ECCV 2020 Spotlight

https://arxiv.org/abs/1909.11065

Presenter: Minho Park

Contribution

- Study the context aggregation problem in semantic segmentation.
- Present a simple yet effective approach, object-contextual representations, characterizing a pixel by exploiting the representation of the corresponding object class.
- "HRNet+OCR+SegFix" achieves 1st place on the Cityscapes leaderboard (2020)

Overview

- Object-Contextual Representation and Augmented Representation

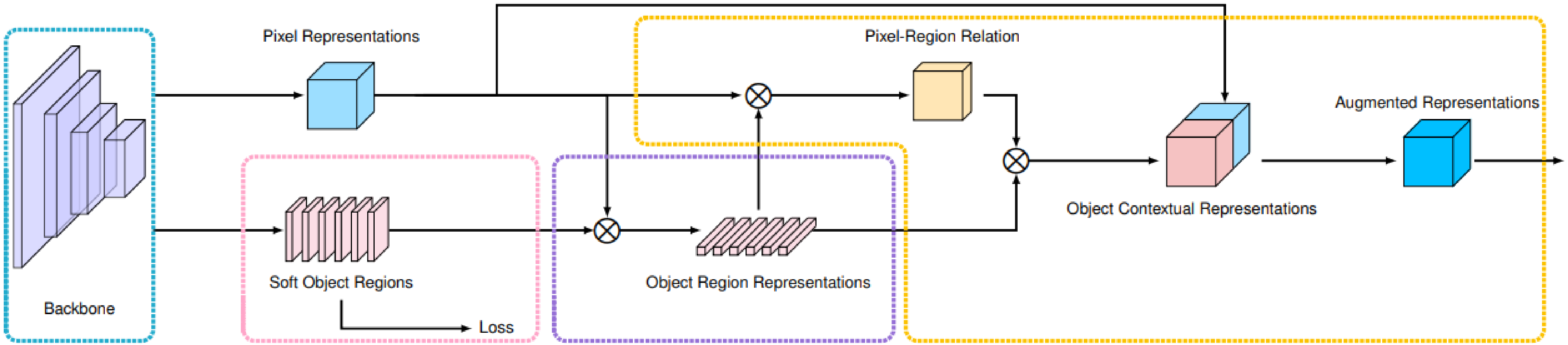

Fig. 3: Illustrating the pipeline of OCR.

Two main streams

1. Exploit multi-scale contexts
    - ASPP

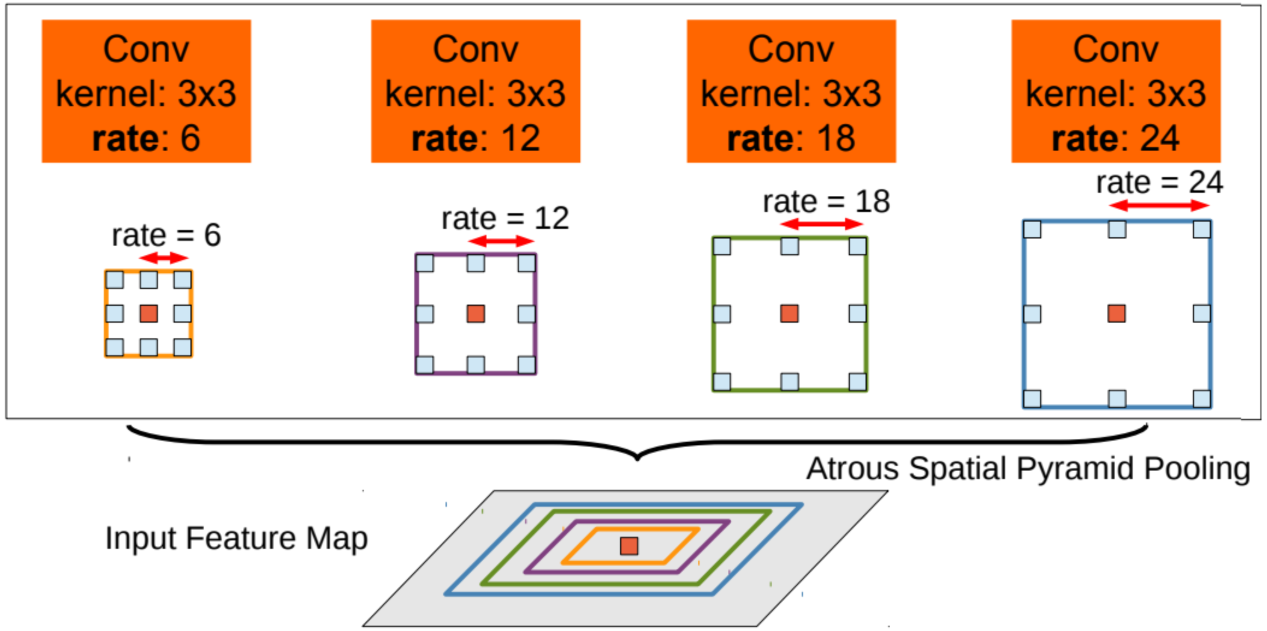

Fig. 4: Atrous Spatial Pyramid Pooling (ASPP).

Chen, Liang-Chieh, et al. "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2017): 834-848.

Two main streams

2. Attention mechanism
   - OCNet

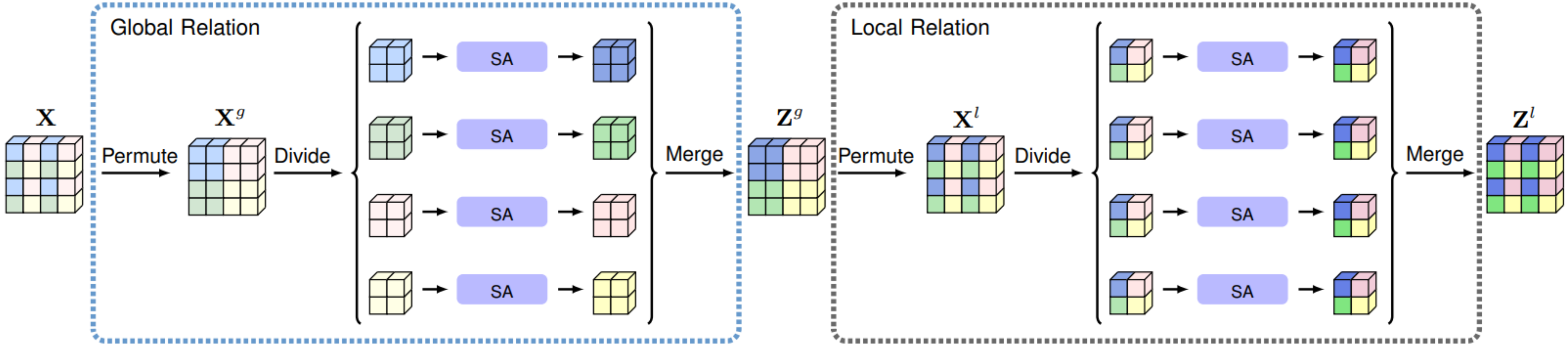

Fig. 2: Illustrating the Interlaced Sparse Self-Attention.

Yuan, Y., Huang, L., Guo, J. et al. OCNet: Object Context for Semantic Segmentation. Int J Comput Vis (2021). https://doi.org/10.1007/s11263-021-01465-9

Two main streams

2. Attention mechanism

- ACFNet

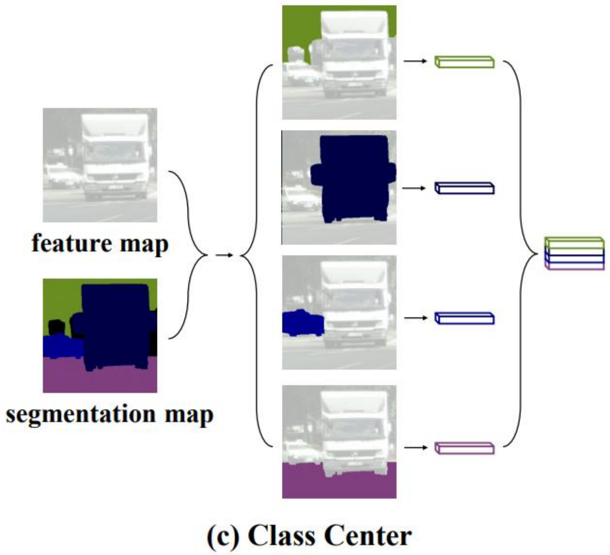

Figure 1. Class Center (c) captures the context via a categorical strategy

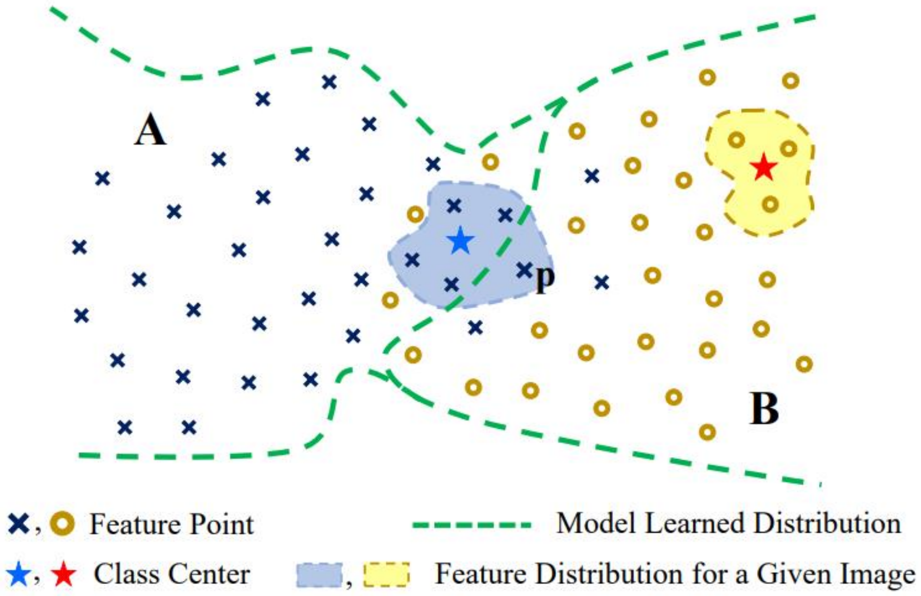

Figure 4. An illustration of the role of class center.

Zhang, Fan, et al. "Acfnet: Attentional class feature network for semantic segmentation." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019.

Motivation

(a) ASPP (b) OCR

Fig. 2: Illustrating the multi-scale context with the ASPP as an example and the OCR context for the pixel marked with .

Architecture

- Object Contextual Representation

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

$$\mathbf{f}_k = \sum_{i\in\mathcal{I}} \tilde{m}_{ki}\mathbf{x}_i$$

$\tilde{m}_{ki}$: spatial softmax to mormalize object region $\mathbf{M}_k$

$\rho(\cdot), \delta(\cdot)$: 1×1 conv $\rightarrow$ BN $\rightarrow$ ReLU

$$w_k = \frac{e^{\kappa(\mathbf{x}_i,\mathbf{f}_k)}}{\sum_{j=1}^{K} e^{\kappa(\mathbf{x}_i,\mathbf{f}_k)}}, \kappa(\mathbf{x}_i,\mathbf{f}_k) = \phi(\mathbf{x})^T\psi(\mathbf{f})$$

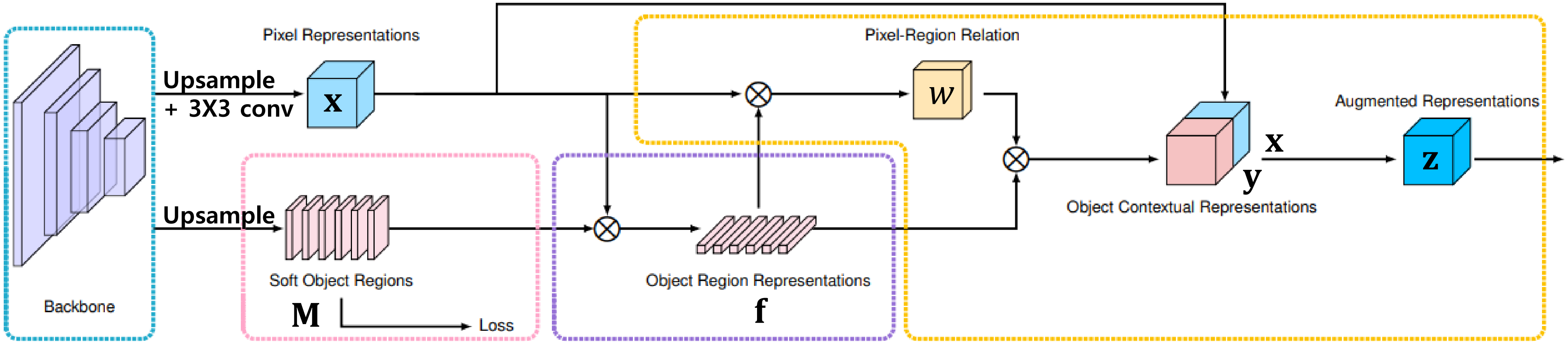

Fig. 3: Illustrating the pipeline of OCR.

Architecture

- Augmented representations

$$\mathbf{z}_i = g\left(\left[\mathbf{x}_i^T\ \mathbf{y}_i^T\right]^T\right)$$

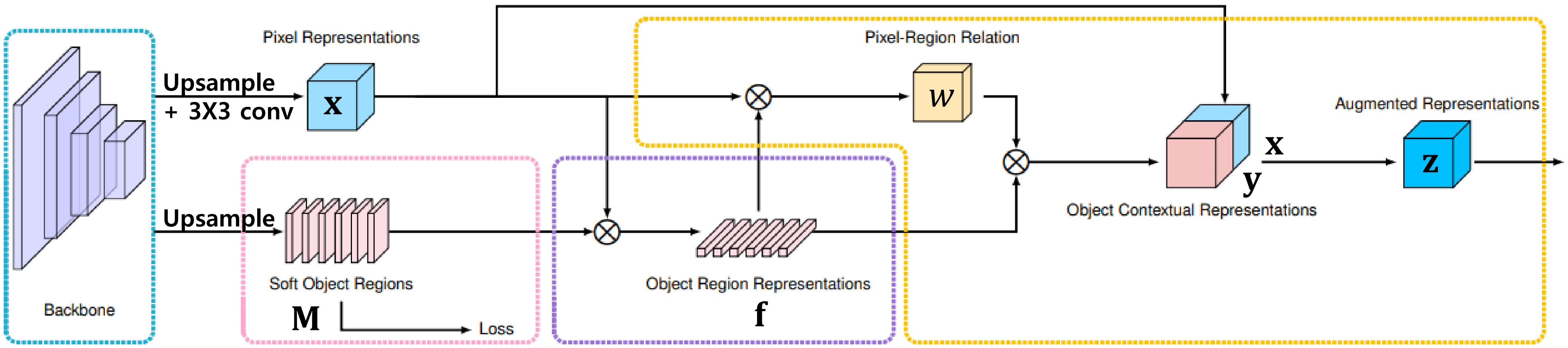

Fig. 3: Illustrating the pipeline of OCR.

Architecture

- Segmentation Transformer: Rephrasing the OCR Method

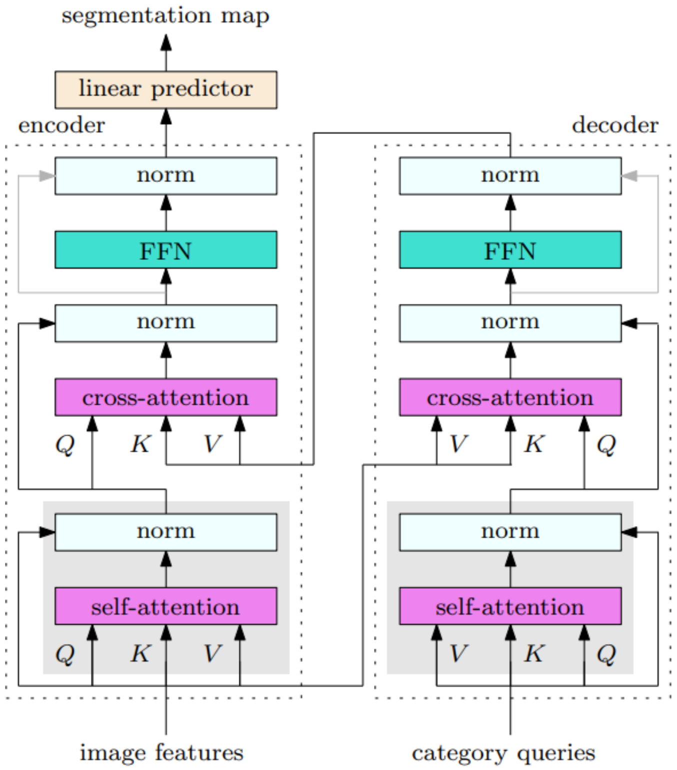

Fig. 4: Segmentation transformer.

$$a_{ij} = \frac{e^{\frac{1}{\sqrt{d}}\mathbf{q}_i^T\mathbf{k}_j}}{Z_i}, \text{where } Z_i = \sum_{j=1}^{N_{kv}} e^{\frac{1}{\sqrt{d}}\mathbf{q}_i^T\mathbf{k}_j}$$

$$\mathrm{Attn}(\mathbf{q}_i, K, V) = \sum_{j=1}^{N_{kv}} a_{ij}\mathbf{v}_j$$

Architecture

- An alternative of segmentation transformer

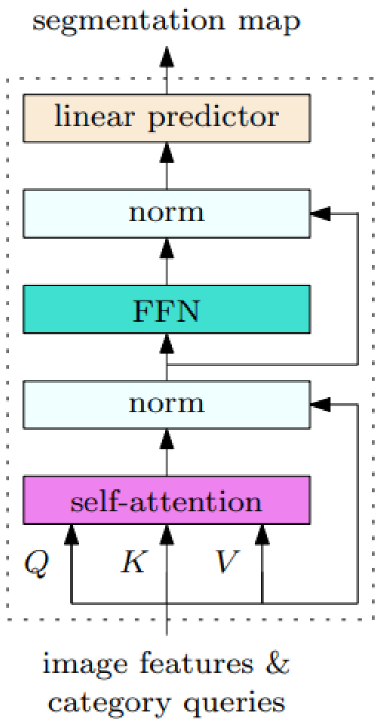

Fig. 5: An alternative of segmentation transformer

Empirical Analysis

- Object region supervision
  - Existence of **soft object regions**
- Pixel-region relations
  - Existence of **object region representations**

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%**

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

Empirical Analysis

- Ground-truth OCR
  - $m_{ki} = \begin{cases} 1, & if\ l_i = k \\ 0, & otherwise \end{cases}, \; w_{ki} = \begin{cases} 1, & if\ l_i = k \\ 0, & otherwise \end{cases}$

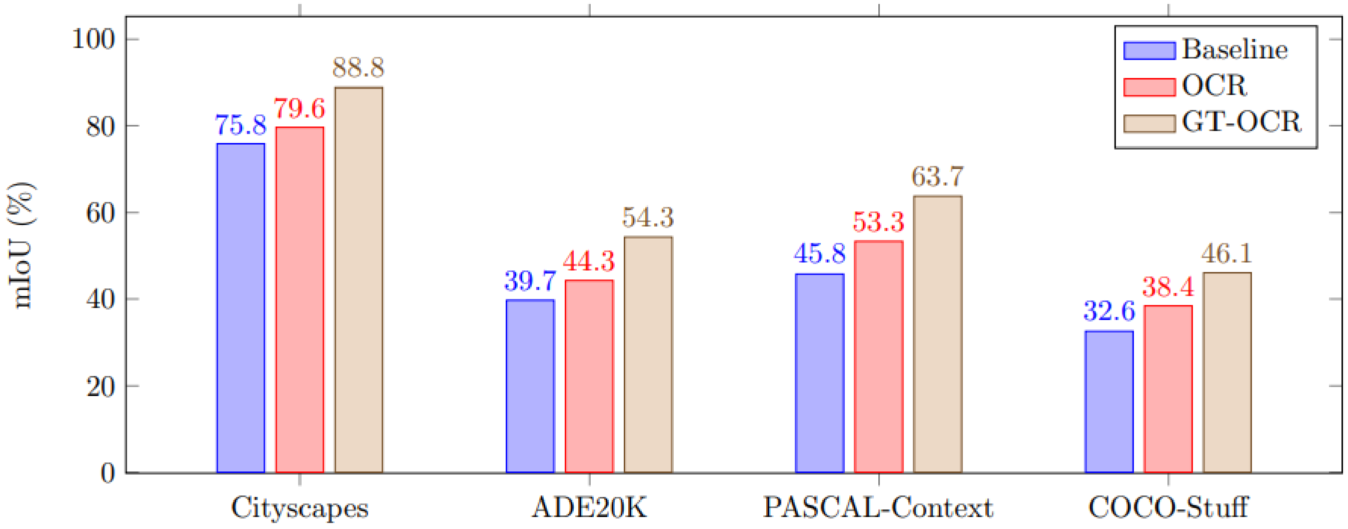

Fig. 1: Illustrating the effectiveness of our OCR scheme.

Experiments

Method	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP
PPM [80]	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 2: Comparison with multi-scale context scheme.

Method	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP
CC-Attention [27]	81.4%	-	45.22%	-
DANet [18]	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%**

Table 3: Comparison with relational context scheme.

Experiments

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.)	**10.2**M	209M	**338**G	46ms
OCR	10.5M	**202**M	340G	**45**ms

Table 4: Complexity comparison.

Experiments

Method	Baseline	Stride	Context schemes	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP	PASCAL Context	COCO-Stuff
Simple baselines									
PSPNet [80]	ResNet-101	8×	M	78.4♭	81.2	43.29	-	47.8	-
DeepLabv3 [6]	ResNet-101	8×	M	-	81.3	-	-	-	-
PSANet [81]	ResNet-101	8×	R	80.1	81.4	43.77	-	-	-
SAC [79]	ResNet-101	8×	M	78.1	-	44.30	-	-	-
AAF [29]	ResNet-101	8×	R	79.1♭	-	-	-	-	-
DSSPN [41]	ResNet-101	8×	-	77.8	-	43.68	-	-	38.9
DepthSeg [32]	ResNet-101	8×	-	78.2	-	-	-	-	-
MMAN [48]	ResNet-101	8×	-	-	-	-	46.81	-	-
JPPNet [39]	ResNet-101	8×	M	-	-	-	51.37	-	-
EncNet [76]	ResNet-101	8×	-	-	-	44.65	-	51.7	-
GCU [38]	ResNet-101	8×	R	-	-	44.81	-	-	-
APCNet [24]	ResNet-101	8×	M,R	-	-	45.38	-	54.7	-
CFNet [77]	ResNet-101	8×	R	79.6	-	44.89	-	54.0	-
BFP [12]	ResNet-101	8×	R	81.4	-	-	-	53.6	-
CCNet [27]	ResNet-101	8×	R	81.4	-	45.22	-	-	-
ANNet [84]	ResNet-101	8×	M,R	81.3	-	45.24	-	52.8	-
OCR (Seg. transformer)	ResNet-101	8×	R	81.8	82.4	45.28	55.60	54.8	39.5

Table 5: Comparison with the state-of-the-art. We use M to represent multiscale context and R to represent relational context. Red, Green, Blue represent the top-3 results.

Experiments

Method	Baseline	Stride	Context schemes	Cityscapes (w/o coarse)	Cityscapes (w/ coarse)	ADE20K	LIP	PASCAL Context	COCO-Stuff
Advanced baselines									
DenseASPP [68]	DenseNet-161	8×	M	80.6	-	-	-	-	-
DANet [18]	ResNet-101 + MG	8×	R	81.5	-	45.22	-	52.6	39.7
DGCNet [78]	ResNet-101 + MG	8×	R	82.0	-	-	-	53.7	-
EMANet [36]	ResNet-101 + MG	8×	R	-	-	-	-	53.1	39.9
SeENet [51]	ResNet-101 + ASPP	8×	M	81.2	-	-	-	-	-
SGR [40]	ResNet-101 + ASPP	8×	R	-	-	44.32	-	52.5	39.1
OCNet [72]	ResNet-101 + ASPP	8×	M,R	81.7	-	45.45	54.72	-	-
ACFNet [75]	ResNet-101 + ASPP	8×	M,R	81.8	-	-	-	-	-
CNIF [63]	ResNet-101 + ASPP	8×	M	-	-	-	56.93	-	
GALD [37]	ResNet-101 + ASPP	8×	M,R	81.8	82.9	-	-	-	-
GALD† [37]	ResNet-101 + CGNL + MG	8×	M,R	-	83.3	-	-	-	-
Mapillary [52]	WideResNet-38 + ASPP	8×	M	-	82.0	-	-	-	-
GSCNN† [55]	WideResNet-38 + ASPP	8×	M	82.8	-	-	-	-	
SPGNet [10]	2× ResNet-50	4×	-	81.1	-	-	-	-	-
ZigZagNet [42]	ResNet-101	4×	M	-	-	-	-	52.1	-
SVCNet [13]	ResNet-101	4×	R	81.0	-	-	-	53.2	39.6
ACNet [19]	ResNet-101 + MG	4×	M,R	82.3	-	45.90	-	54.1	40.1
CE2P [45]	ResNet-101 + PPM	4×	M	-	-	-	53.10	-	
VPLR†‡ [83]	WideResNet-38 + ASPP	4×	M	-	83.5	-	-	-	
DeepLabv3+ [7]	Xception-71	4×	M	-	82.1	-	-	-	-
DPC [4]	Xception-71	4×	M	82.7	-	-	-	-	
DUpsampling [57]	Xception-71	4×	M	-	-	-	-	52.5	-
HRNet [54]	HRNetV2-W48	4×	-	81.6	-	-	55.90	54.0	
OCR (Seg. transformer)	HRNetV2-W48	4×	R	82.4	83.0	45.66	56.65	56.2	40.5
OCR† (Seg. transformer)	HRNetV2-W48	4×	R	83.6	84.2	-	-	-	-

Table 5: Comparison with the state-of-the-art. We use M to represent multiscale context and R to represent relational context. Red, Green, Blue represent the top-3 results.