CVPR 2021

Fully Convolutional Networks for Panoptic Segmentation

Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang
Zeming Li, Jian Sun, Jiaya Jia

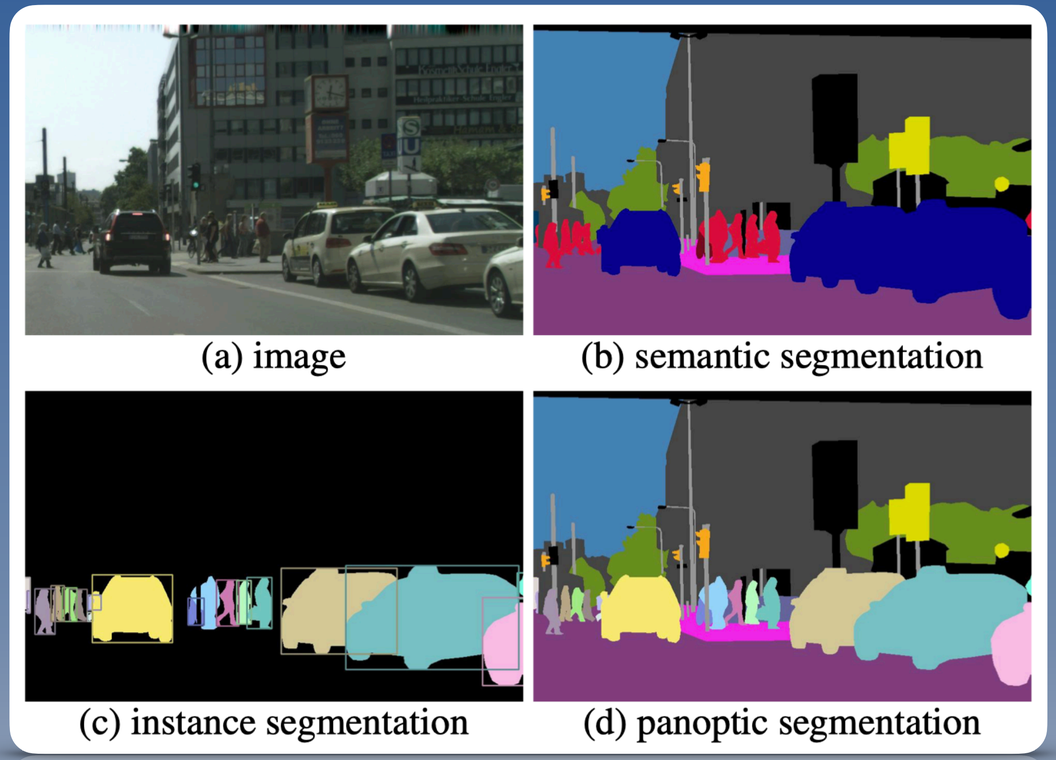
Introduction

Definition of Panoptic Segmentation

Assign each pixel with a semantic label and unique identity to Things and Stuff.

Difficulties in Panoptic Segmentation

- Conflicting properties of Things and Stuff. Things rely on instance-aware features, while Stuff need semantic-consistent characters.
- How to encode things and stuff in a unified representation?
- How to model the relationship among things, and between things and stuff?



Comparison among tasks. [1]

[1] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Doll´ar. Panoptic segmentation. In *CVPR*, 2019.

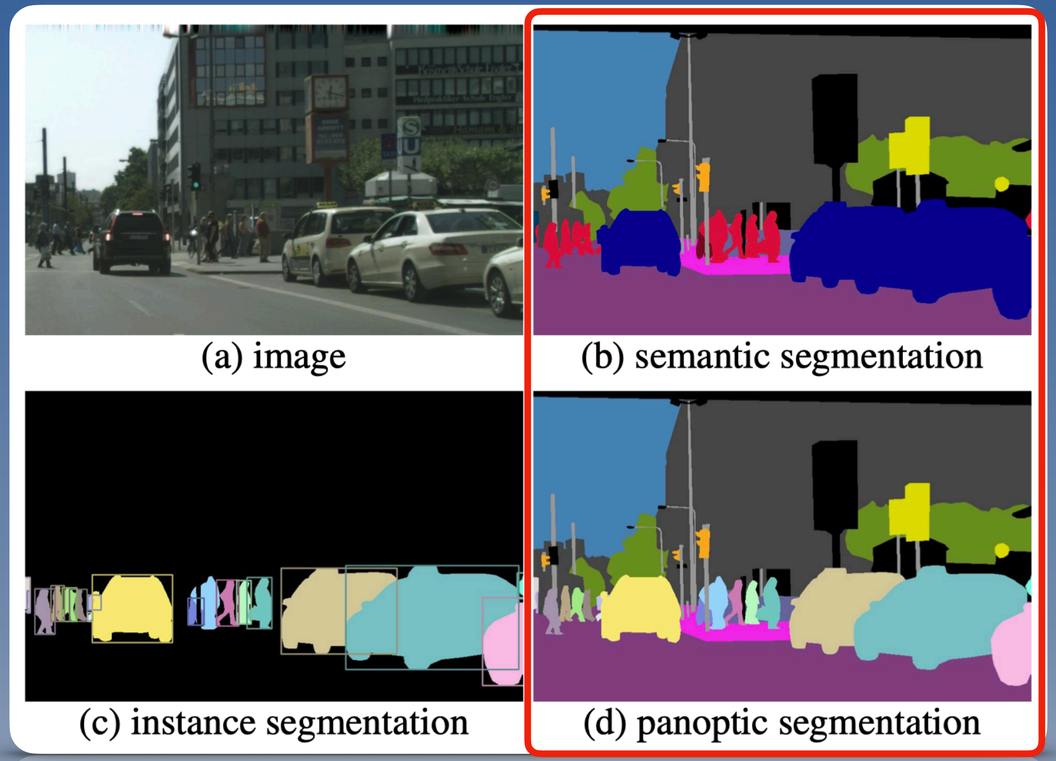
Introduction

Definition of Panoptic Segmentation

Assign each pixel with a semantic label and unique identity to Things and Stuff.

Difficulties in Panoptic Segmentation

- Conflicting properties of Things and Stuff. Things rely on instance-aware features, while Stuff need semantic-consistent characters.
- How to encode things and stuff in a unified representation?
- How to model the relationship among things, and between things and stuff?



Comparison among tasks. [1]

[1] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Doll´ar. Panoptic segmentation. In *CVPR*, 2019.

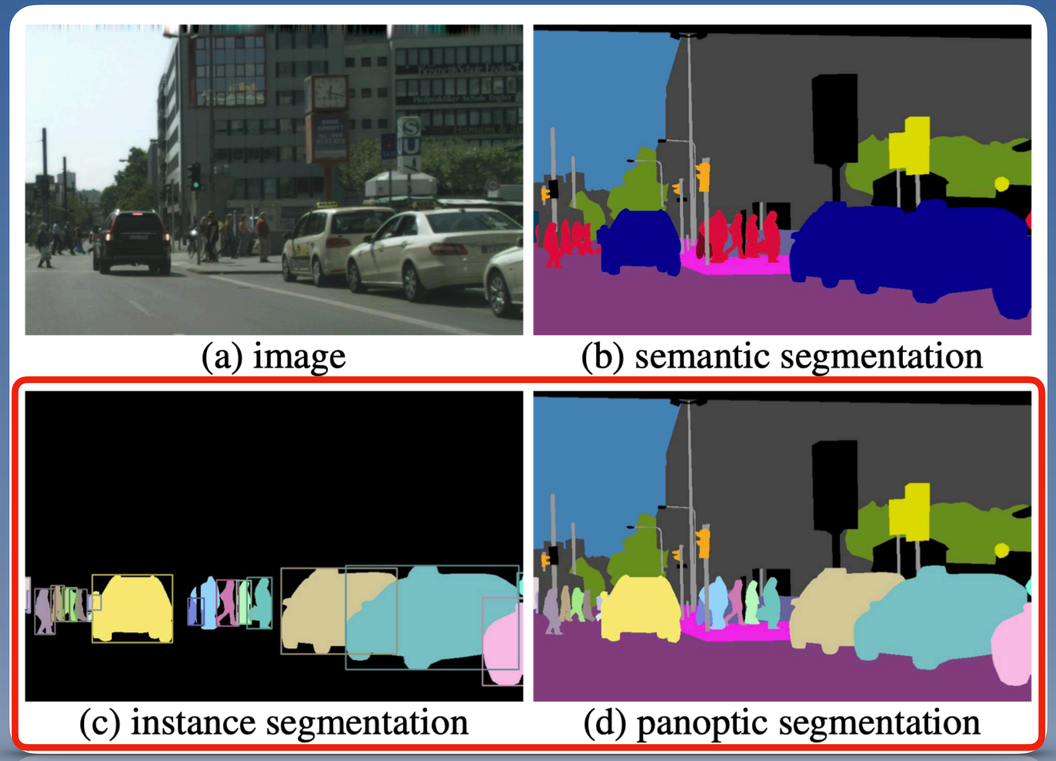
Introduction

Definition of Panoptic Segmentation

Assign each pixel with a semantic label and unique identity to Things and Stuff.

Difficulties in Panoptic Segmentation

- Conflicting properties of Things and Stuff. Things rely on instance-aware features, while Stuff need semantic-consistent characters.
- How to encode things and stuff in a unified representation?
- How to model the relationship among things, and between things and stuff?



Comparison among tasks. [1]

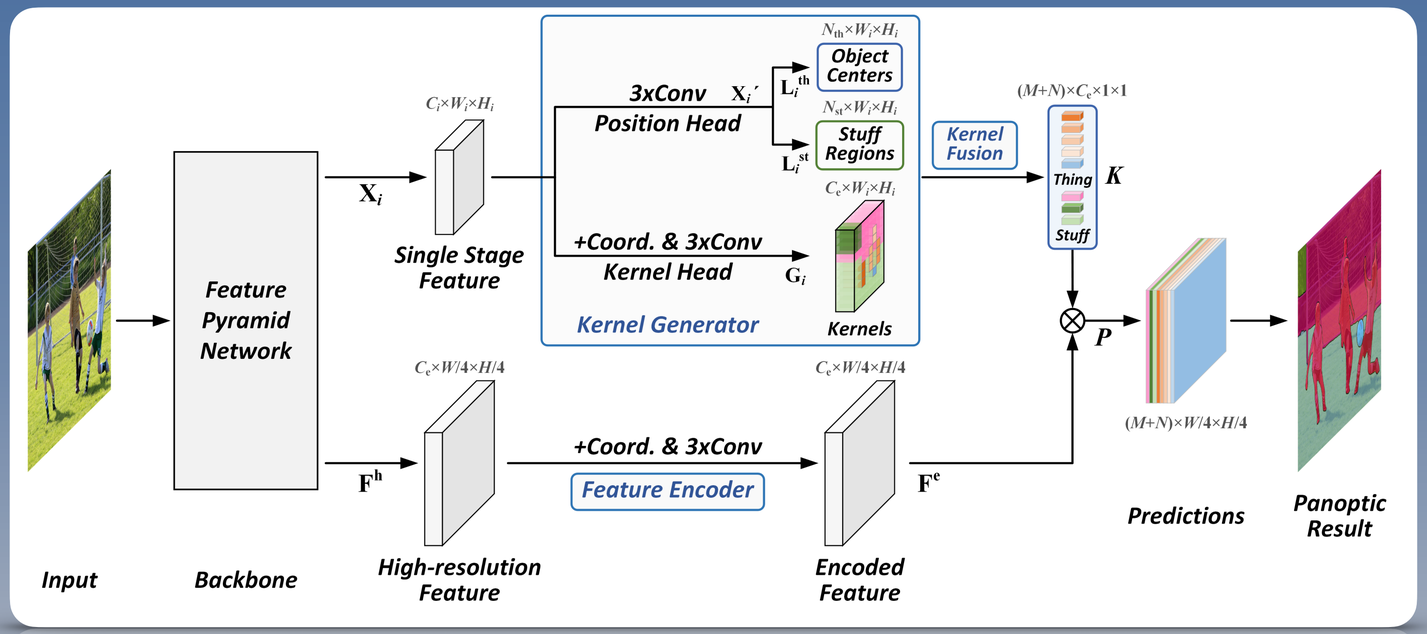
[1] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Doll´ar. Panoptic segmentation. In *CVPR*, 2019.

Panoptic FCN

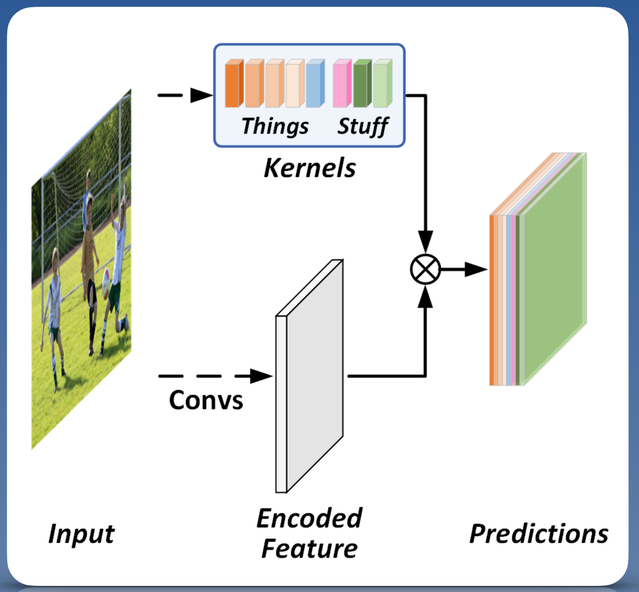
CVPR 2021

Panoptic FCN represent them uniformly

- It encodes each instance into a specific kernel and generates the prediction by convolutions directly.
- Instance-awareness for things: each thing has unique kernel.
- Semantic-consistency for stuff: identical stuff has same kernel.



Framework of Panoptic FCN.



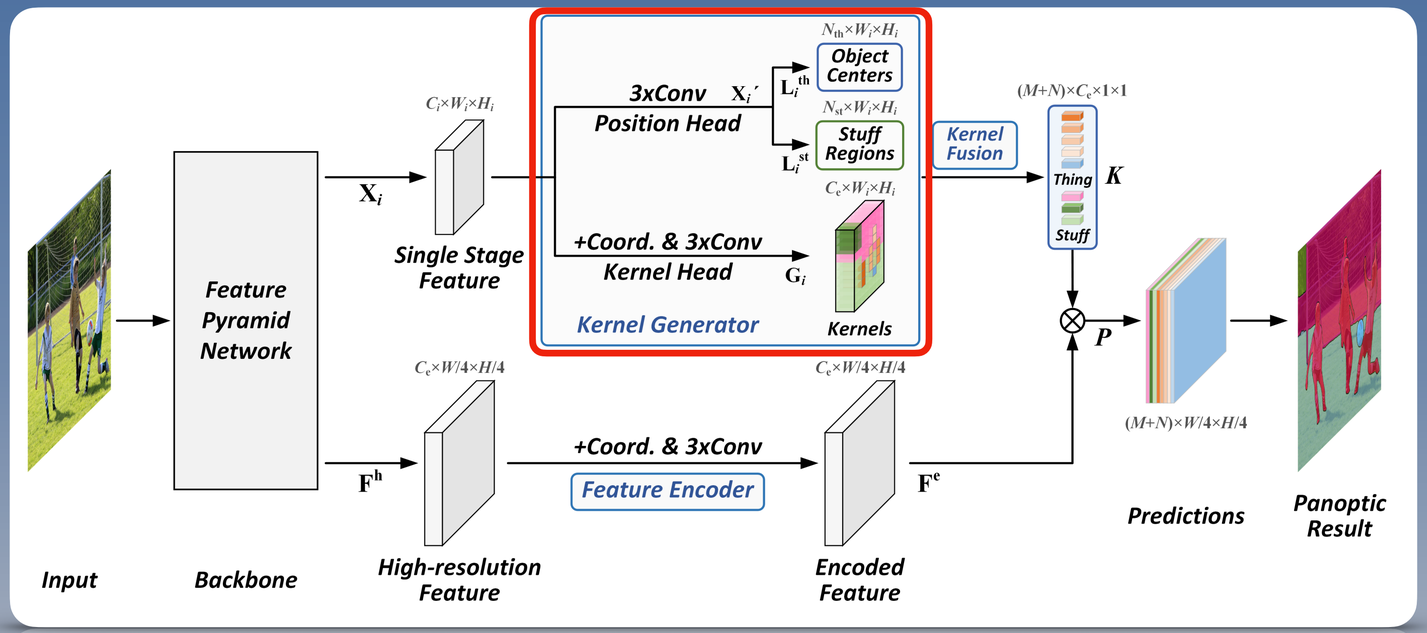
Unified representation.

Panoptic FCN

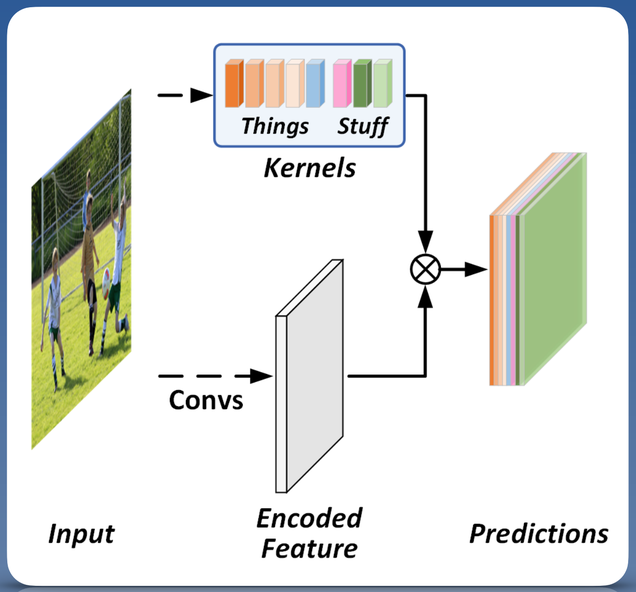
CVPR 2021

Panoptic FCN represent them uniformly

- It encodes each instance into a specific kernel and generates the prediction by convolutions directly.
- Instance-awareness for things: each thing has unique kernel.
- Semantic-consistency for stuff: identical stuff has same kernel.



Framework of Panoptic FCN.



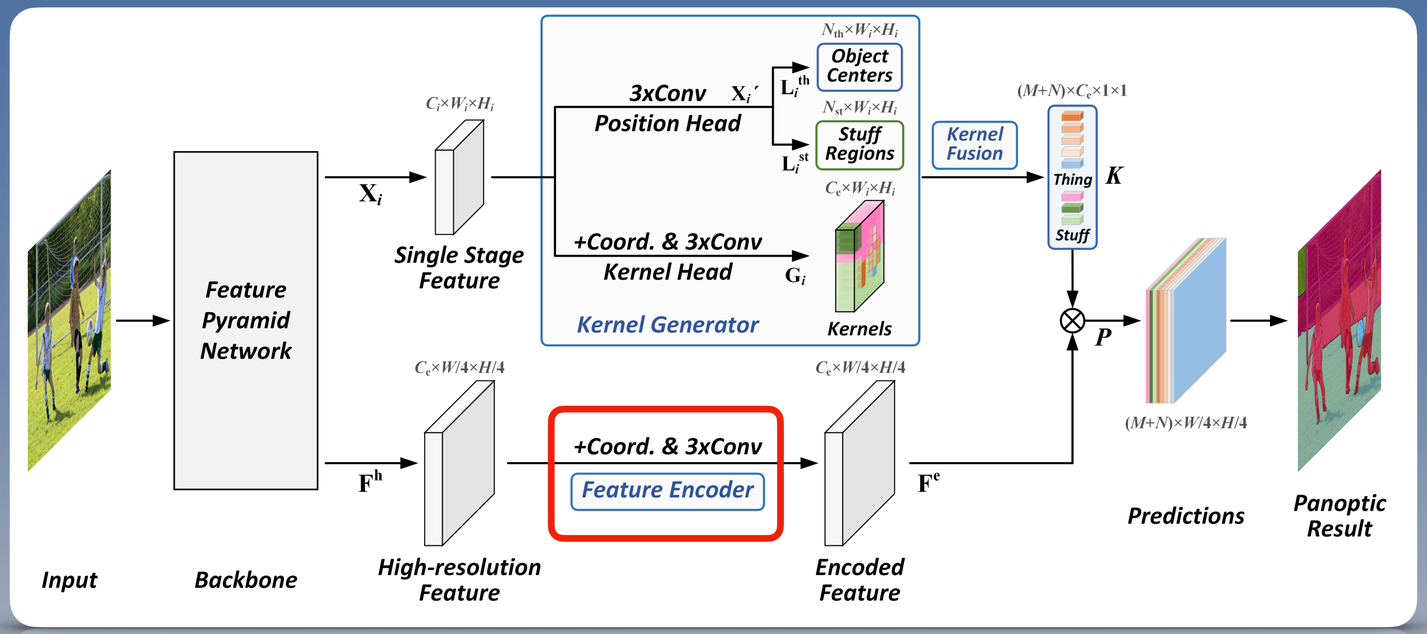
Unified representation.

Panoptic FCN

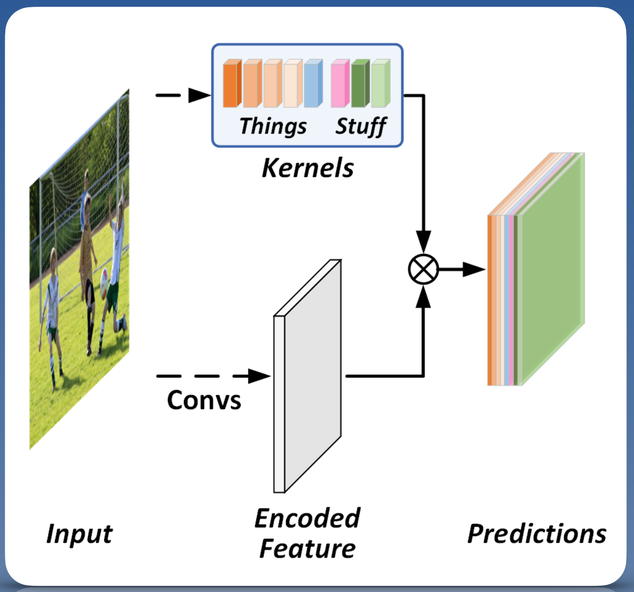
CVPR 2021

Panoptic FCN represent them uniformly

- It encodes each instance into a specific kernel and generates the prediction by convolutions directly.
- Instance-awareness for things: each thing has unique kernel.
- Semantic-consistency for stuff: identical stuff has same kernel.



Framework of Panoptic FCN.



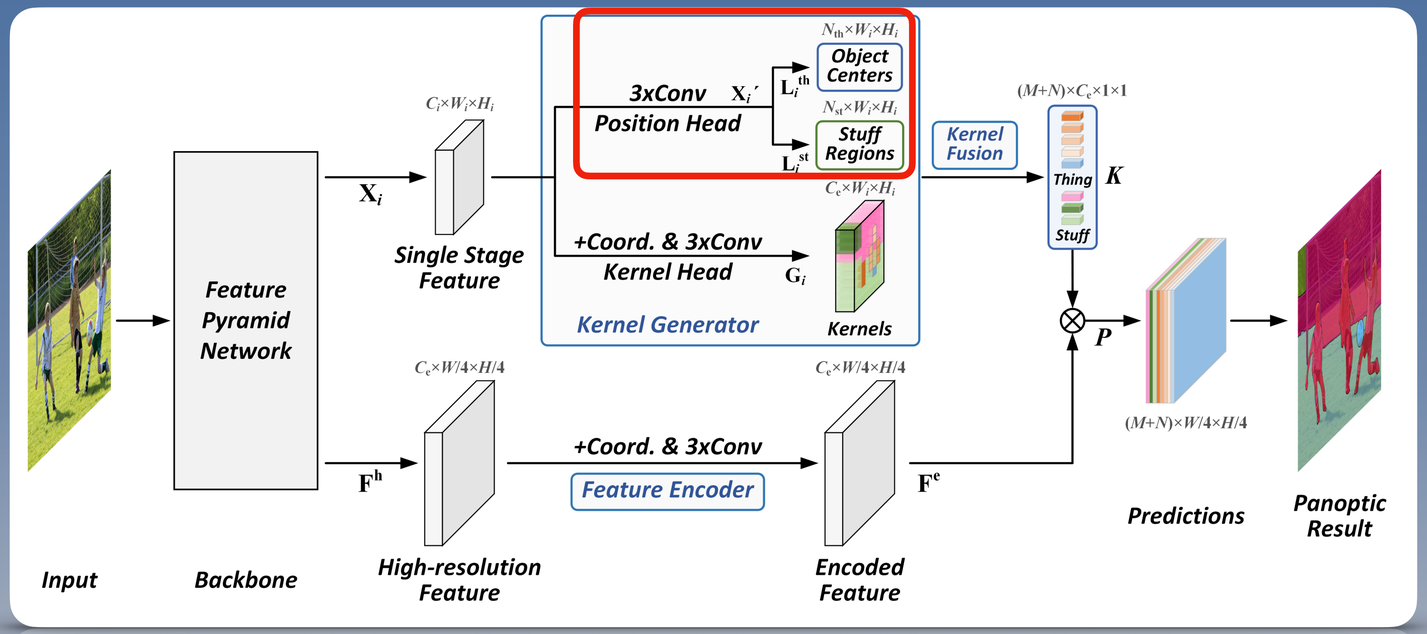
Unified representation.

Panoptic FCN

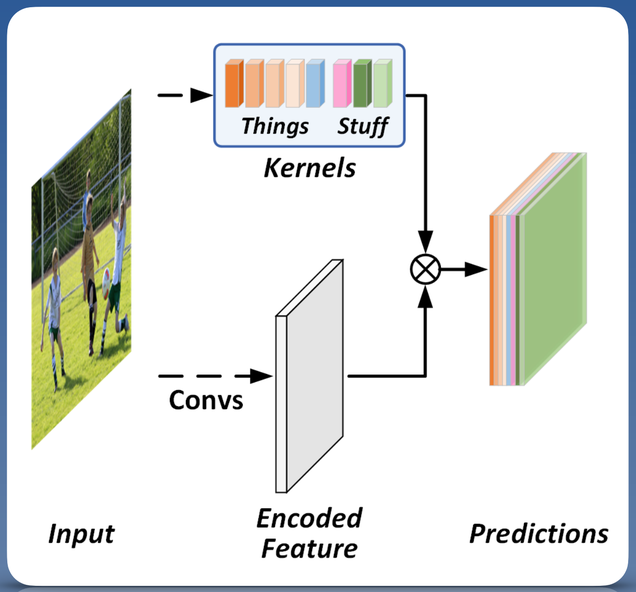
CVPR 2021

Panoptic FCN represent them uniformly

- It encodes each instance into a specific kernel and generates the prediction by convolutions directly.
- Instance-awareness for things: each thing has unique kernel.
- Semantic-consistency for stuff: identical stuff has same kernel.



Framework of Panoptic FCN.



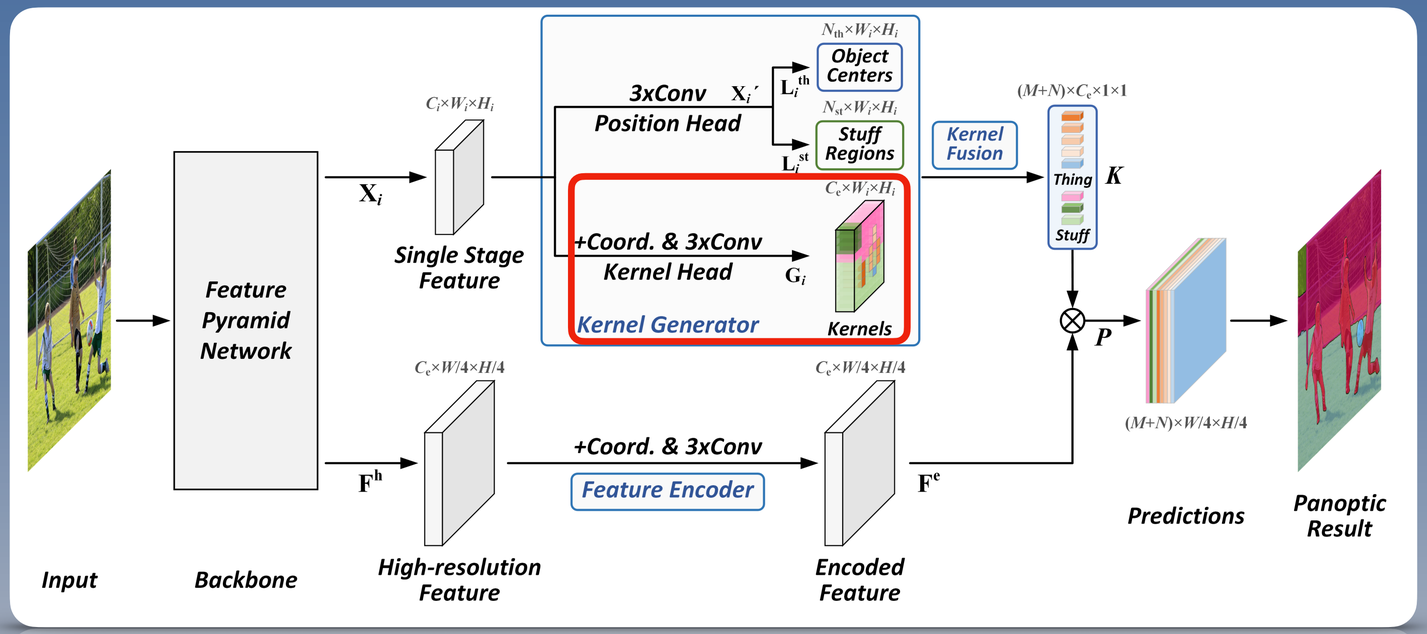
Unified representation.

Panoptic FCN

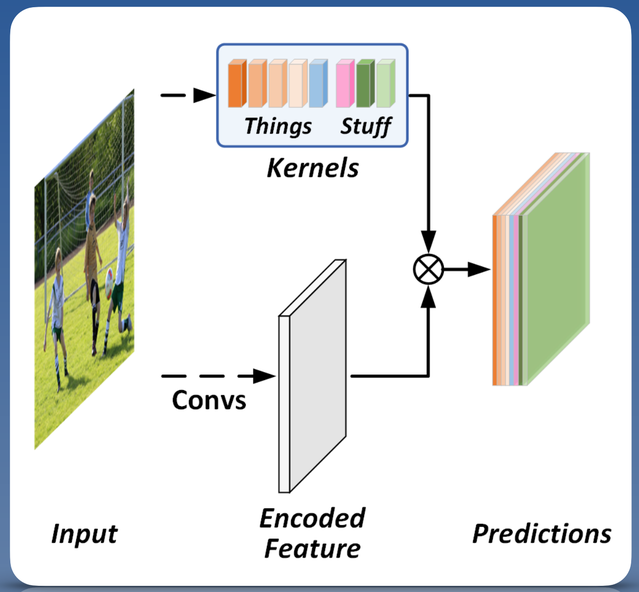
CVPR 2021

Panoptic FCN represent them uniformly

- It encodes each instance into a specific kernel and generates the prediction by convolutions directly.
- Instance-awareness for things: each thing has unique kernel.
- Semantic-consistency for stuff: identical stuff has same kernel.



Framework of Panoptic FCN.



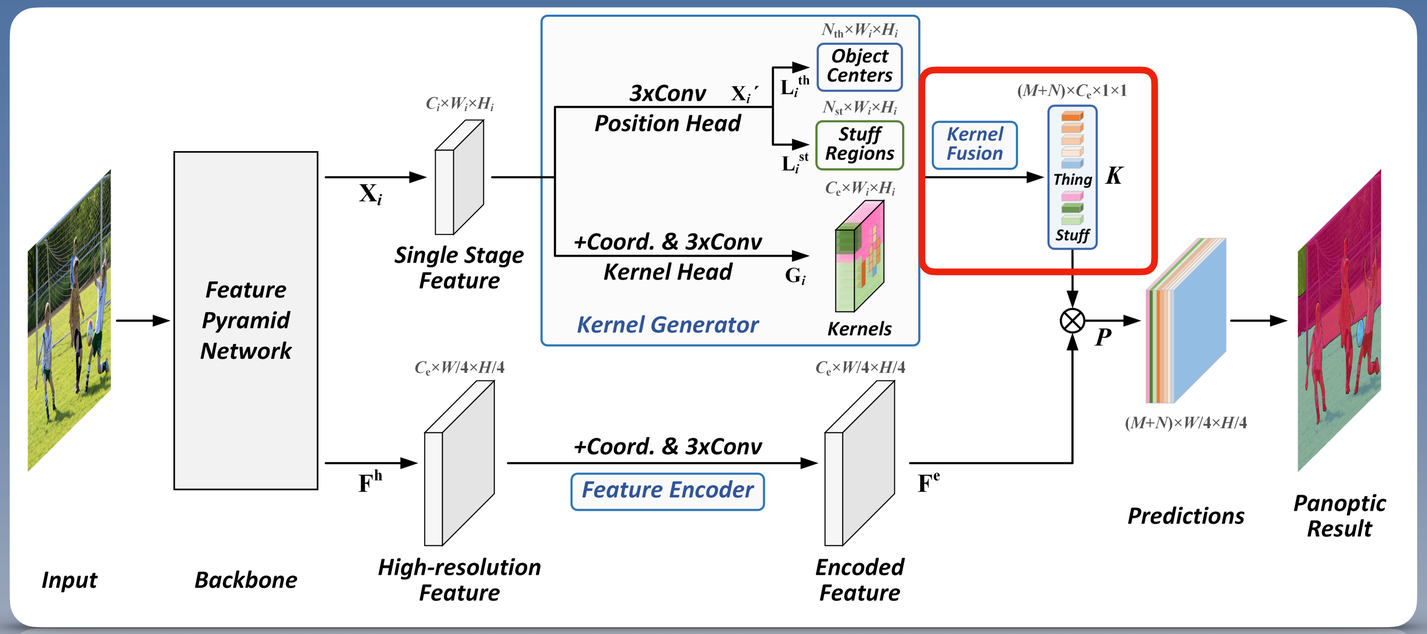
Unified representation.

Panoptic FCN

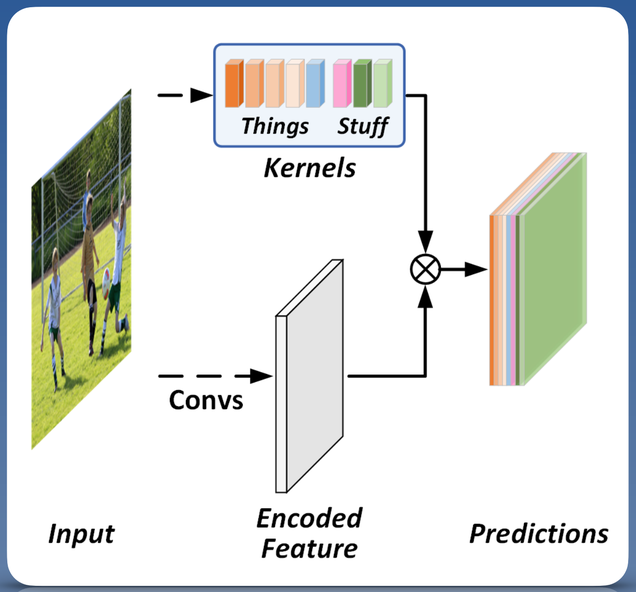
CVPR 2021

Panoptic FCN represent them uniformly

- It encodes each instance into a specific kernel and generates the prediction by convolutions directly.
- Instance-awareness for things: each thing has unique kernel.
- Semantic-consistency for stuff: identical stuff has same kernel.



Framework of Panoptic FCN.



Unified representation.

Unified loss function in Panoptic FCN

- Loss function for position localization

$$\mathscr{L}_{\text{pos}}^{\text{th}} = \sum_i \text{FL}(\mathbf{L}_i^{\text{th}}, \mathbf{Y}_i^{\text{th}})/N_{\text{th}}$$

$$\mathscr{L}_{\text{pos}}^{\text{st}} = \sum_i \text{FL}(\mathbf{L}_i^{\text{st}}, \mathbf{Y}_i^{\text{st}})/W_i H_i$$

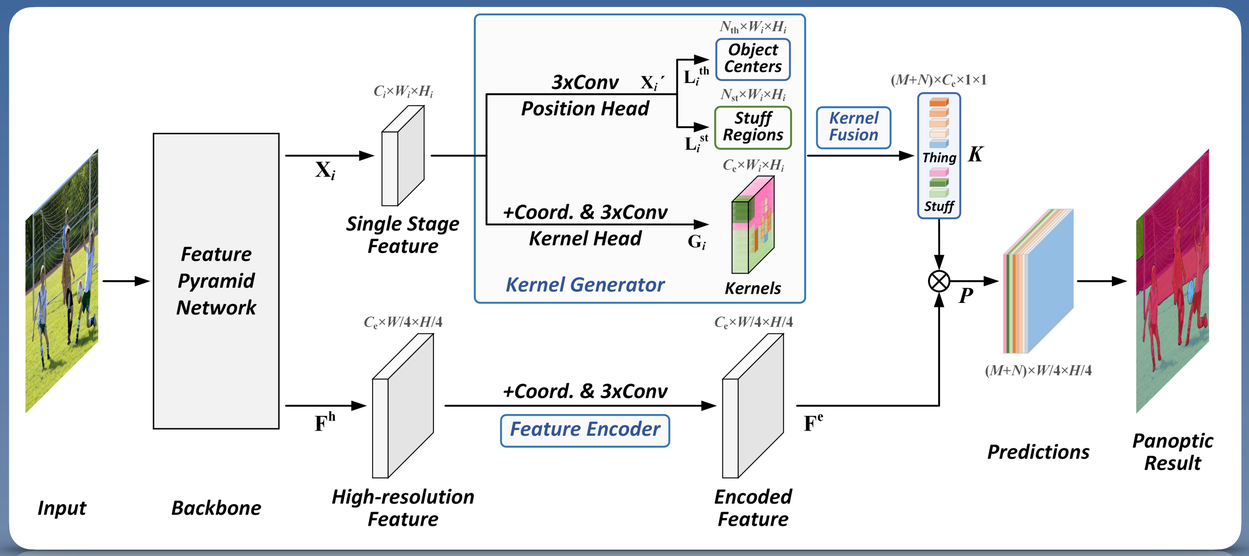
$$\mathscr{L}_{\text{pos}} = \mathscr{L}_{\text{pos}}^{\text{th}} + \mathscr{L}_{\text{pos}}^{\text{st}}$$

- Loss function for segmentation

$$\text{WDice}(\mathbf{P}_j, \mathbf{Y}_j^{\text{seg}}) = \sum_k w_k \text{Dice}(\mathbf{P}_{j,k}, \mathbf{Y}_j^{\text{seg}})$$

$$\mathscr{L}_{\text{seg}} = \sum_j \text{WDice}(\mathbf{P}_j, \mathbf{Y}_j^{\text{seg}})/(M+N)$$

$$\mathscr{L} = \lambda_{\text{pos}} \mathscr{L}_{\text{pos}} + \lambda_{\text{seg}} \mathscr{L}_{\text{seg}}$$



Framework of Panoptic FCN.

Unified loss function in Panoptic FCN

- Loss function for position localization

$$\mathscr{L}_{\text{pos}}^{\text{th}} = \sum_i \text{FL}(\mathbf{L}_i^{\text{th}}, \mathbf{Y}_i^{\text{th}})/N_{\text{th}}$$

$$\mathscr{L}_{\text{pos}}^{\text{st}} = \sum_i \text{FL}(\mathbf{L}_i^{\text{st}}, \mathbf{Y}_i^{\text{st}})/W_i H_i$$

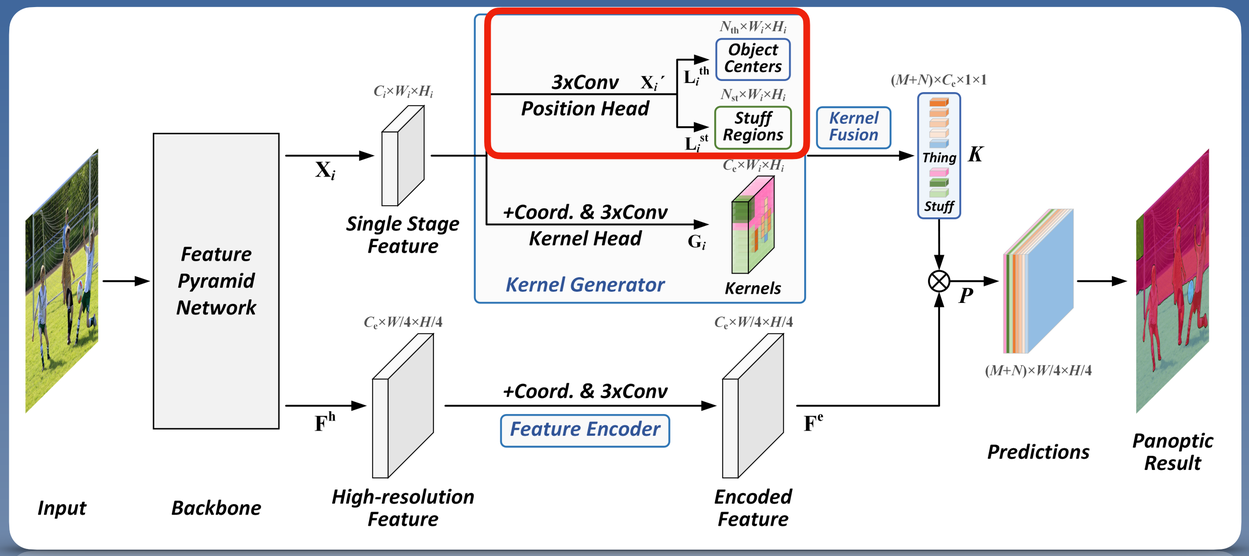
$$\mathscr{L}_{\text{pos}} = \mathscr{L}_{\text{pos}}^{\text{th}} + \mathscr{L}_{\text{pos}}^{\text{st}}$$

- Loss function for segmentation

$$\text{WDice}(\mathbf{P}_j, \mathbf{Y}_j^{\text{seg}}) = \sum_k w_k \text{Dice}(\mathbf{P}_{j,k}, \mathbf{Y}_j^{\text{seg}})$$

$$\mathscr{L}_{\text{seg}} = \sum_j \text{WDice}(\mathbf{P}_j, \mathbf{Y}_j^{\text{seg}})/(M+N)$$

$$\mathscr{L} = \lambda_{\text{pos}} \mathscr{L}_{\text{pos}} + \lambda_{\text{seg}} \mathscr{L}_{\text{seg}}$$



Framework of Panoptic FCN.

Unified loss function in Panoptic FCN

- Loss function for position localization

$$\mathscr{L}_{\text{pos}}^{\text{th}} = \sum_i \text{FL}(\mathbf{L}_i^{\text{th}}, \mathbf{Y}_i^{\text{th}})/N_{\text{th}}$$

$$\mathscr{L}_{\text{pos}}^{\text{st}} = \sum_i \text{FL}(\mathbf{L}_i^{\text{st}}, \mathbf{Y}_i^{\text{st}})/W_i H_i$$

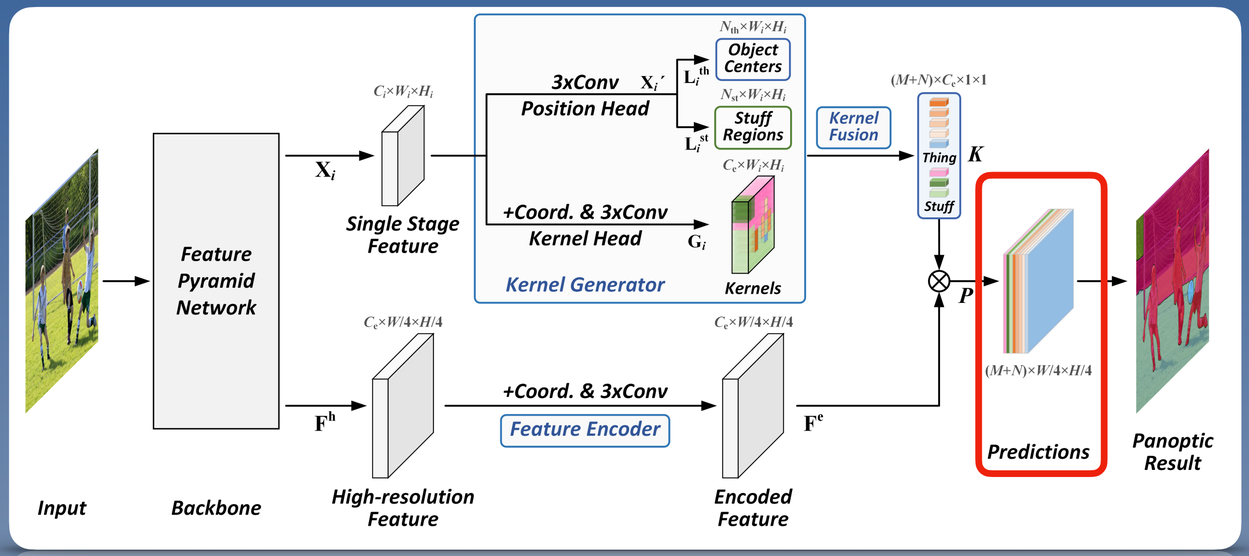
$$\mathscr{L}_{\text{pos}} = \mathscr{L}_{\text{pos}}^{\text{th}} + \mathscr{L}_{\text{pos}}^{\text{st}}$$

- Loss function for segmentation

$$\text{WDice}(\mathbf{P}_j, \mathbf{Y}_j^{\text{seg}}) = \sum_k w_k \text{Dice}(\mathbf{P}_{j,k}, \mathbf{Y}_j^{\text{seg}})$$

$$\mathscr{L}_{\text{seg}} = \sum_j \text{WDice}(\mathbf{P}_j, \mathbf{Y}_j^{\text{seg}})/(M+N)$$

$$\mathscr{L} = \lambda_{\text{pos}} \mathscr{L}_{\text{pos}} + \lambda_{\text{seg}} \mathscr{L}_{\text{seg}}$$



Framework of Panoptic FCN.

Results & Analysis

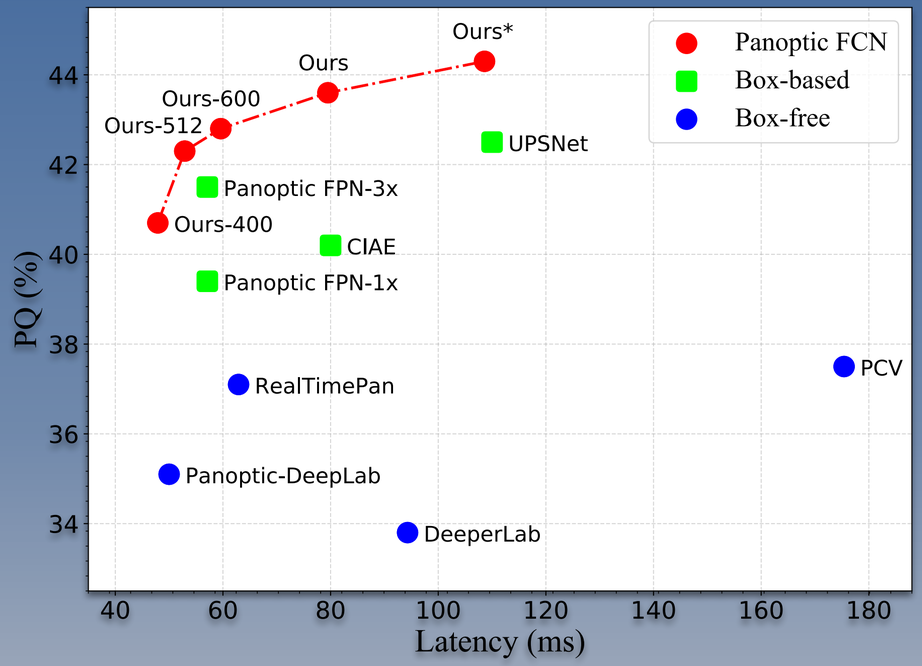
Component-wise Analysis in Panoptic FCN

Ablation studies on upper-bound and speed-accuracy.
More detailed ablations please refer to the paper.

Upper-bound analysis on the COCO val set. gt position and gt class denote utilizing ground-truth position and class for kernel generation, respectively.

gt position	gt class	PQ	PQ^{th}	PQ^{st}	AP	mIoU
○	✓	43.6	49.3	35.0	34.5	43.8
✓	○	49.8	52.2	46.1	38.2	54.6
✓	✓	**65.9**	**64.1**	**68.7**	**45.4**	**86.6**
		+22.3	*+14.8*	*+33.7*	*+11.0*	*+42.8*

Speed-accuracy trade-off curve on the COCO val set. The latency is measured end-to-end from single input to panoptic result.



Results of Panoptic FCN

It surpasses previous box-based and box-free methods with efficiency.

Comparisons with previous methods on the COCO val set. Panoptic FCN-400, 512, and 600 denotes utilizing smaller input instead of the default setting. All of our results are achieved with single input and no flipping.

Method	*Backbone*	PQ	SQ	RQ	PQ^{th}	SQ^{th}	RQ^{th}	PQ^{st}	SQ^{st}	RQ^{st}	Device	FPS
box-based												
Panoptic FPN-1x	Res50-FPN	39.4	77.8	48.3	45.9	80.9	55.3	29.6	73.3	37.7	V100	17.5
Panoptic FPN-3x	Res50-FPN	41.5	79.1	50.5	48.3	82.2	57.9	31.2	74.4	39.5	V100	17.5
CIAE	Res50-FPN	40.2	-	-	45.3	-	-	32.3	-	-	2080Ti	12.5
UPSNet	Res50-FPN	42.5	78.0	52.5	48.6	79.4	**59.6**	33.4	75.9	41.7	V100	9.1
Unifying	Res50-FPN	43.4	79.6	53.0	48.6	-	-	35.5	-	-	-	-
box-free												
DeeperLab	Xception-71	33.8	-	-	-	-	-	-	-	-	V100	10.6
Panoptic-DeepLab	Res50	35.1	-	-	-	-	-	-	-	-	V100	20
AdaptIS	Res50	35.9	-	-	40.3	-	-	29.3	-	-	-	-
PCV	Res50-FPN	37.5	77.7	47.2	40.0	78.4	50.0	33.7	76.5	42.9	1080Ti	5.7
SOLO V2	Res50-FPN	42.1	-	-	49.6	-	-	30.7	-	-	-	-
ours												
Panoptic FCN-400	Res50-FPN	40.7	80.5	49.3	44.9	82.0	54	34.3	78.1	42.1	V100	**20.9**
Panoptic FCN-512	Res50-FPN	42.3	**80.9**	51.2	47.4	82.1	56.9	34.7	**79.1**	42.7	V100	18.9
Panoptic FCN-600	Res50-FPN	42.8	80.6	51.6	47.9	82.6	57.2	35.1	77.4	43.1	V100	16.8
Panoptic FCN	Res50-FPN	43.6	80.6	52.6	49.3	82.6	58.9	35.0	77.6	42.9	V100	12.5
Panoptic FCN*	Res50-FPN	**44.3**	80.7	**53.0**	**50.0**	**83.4**	59.3	**35.6**	76.7	**43.5**	V100	9.2

Results & Analysis

Results of Panoptic FCN

It surpasses previous box-based and box-free methods with efficiency.

Experiments on the COCO test-dev set.

Method	Backbone	PQ	PQ^{th}	PQ^{st}
box-based				
Panoptic FPN	Res101-FPN	40.9	48.3	29.7
CIAE	DCN101-FPN	44.5	49.7	36.8
AUNet	ResNeXt152-FPN	46.5	**55.8**	32.5
UPSNet	DCN101-FPN	46.6	53.2	36.7
Unifying^	DCN101-FPN	47.2	53.5	37.7
box-free				
DeeperLab	Xception-71	34.3	37.5	29.6
SSAP	Res101-FPN	36.9	40.1	32.0
Panoptic-DeepLab	Xception-71	39.7	43.9	33.2
AdaptIS	ResNeXt101	42.8	53.2	36.7
Axial-DeepLab	Axial-ResNet-L	43.6	48.9	35.6
ours				
Panoptic FCN	Res101-FPN	45.5	51.4	36.4
Panoptic FCN	DCN101-FPN	47.0	53.0	37.8
Panoptic FCN*	DCN101-FPN	47.1	53.2	37.8
Panoptic FCN*^	DCN101-FPN	**47.5**	53.7	**38.2**

Experiments on the Cityscapes val set.

Method	Backbone	PQ	PQ^{th}	PQ^{st}
box-based				
Panoptic FPN	Res101-FPN	58.1	52.0	62.5
AUNet	Res101-FPN	59.0	54.8	62.1
UPSNet	Res50-FPN	59.3	54.6	62.7
Seamless	Res50-FPN	60.2	55.6	63.6
Unifying	Res50-FPN	61.4	54.7	66.3
box-free				
PCV	Res50-FPN	54.2	47.8	58.9
DeeperLab	Xception-71	56.5	-	-
SSAP	Res50-FPN	58.4	50.6	-
AdaptIS	Res50	59.0	**55.8**	61.3
Panoptic-DeepLab	Res50	59.7	-	-
ours				
Panoptic FCN	Res50-FPN	59.6	52.1	65.1
Panoptic FCN*	Res50-FPN	**61.4**	54.8	**66.6**

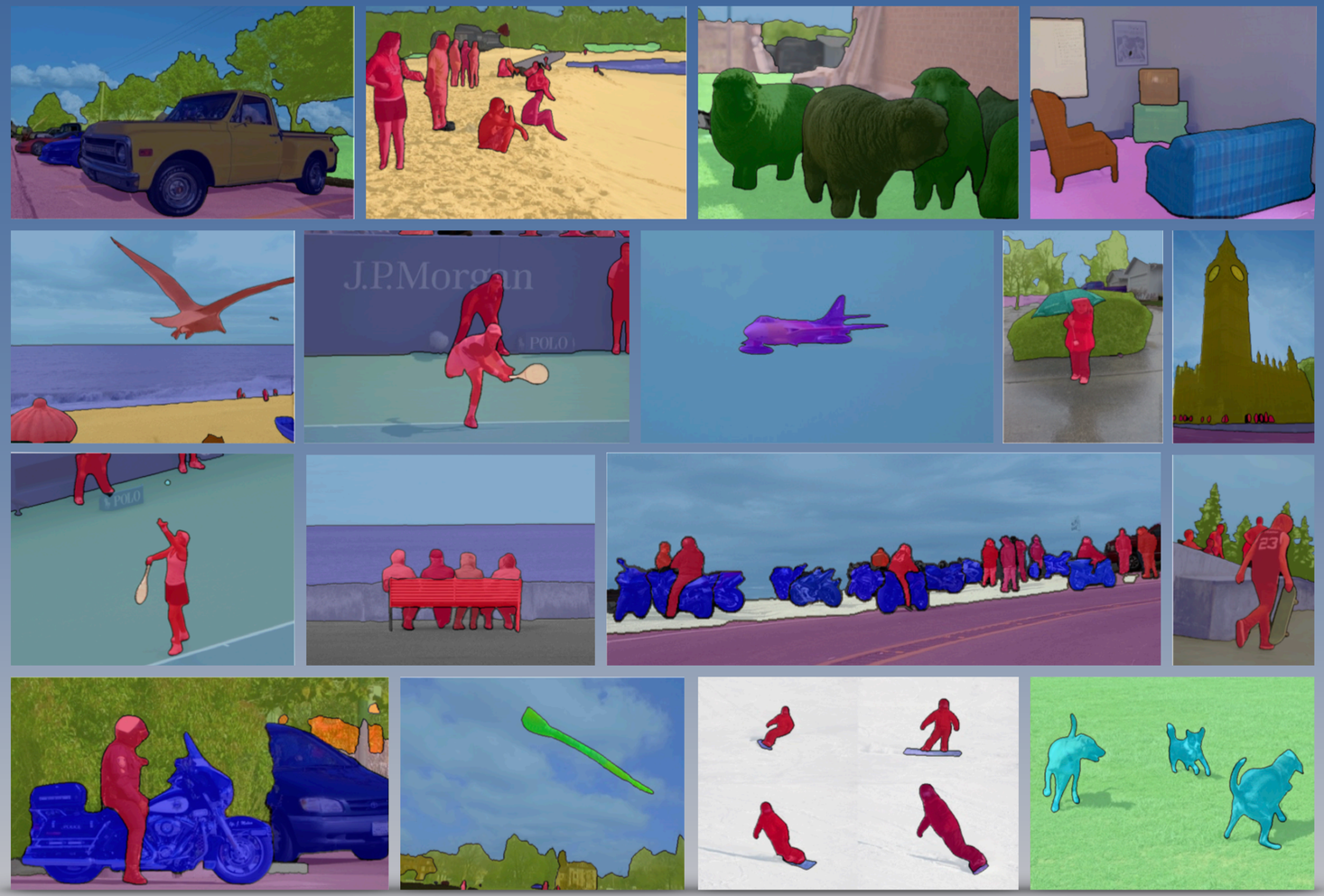
Experiments on the Mapillary Vistas val set.

Method	Backbone	PQ	PQ^{th}	PQ^{st}
box-based				
BGRNet	Res50-FPN	31.8	**34.1**	27.3
TASCNet	Res50-FPN	32.6	31.1	34.4
Seamless	Res50-FPN	36.2	33.6	40.0
box-free				
DeeperLab	Xception-71	32.0	-	-
AdaptIS	Res50	32.0	26.6	39.1
Panoptic-DeepLab	Res50	33.3	-	-
ours				
Panoptic FCN	Res50-FPN	34.8	30.6	40.5
Panoptic FCN*	Res50-FPN	**36.9**	32.9	**42.3**

Visualization of Panoptic FCN

It achieve fine results on common context and traffic-related scenarios.

Visualization of panoptic results on the COCO val set.

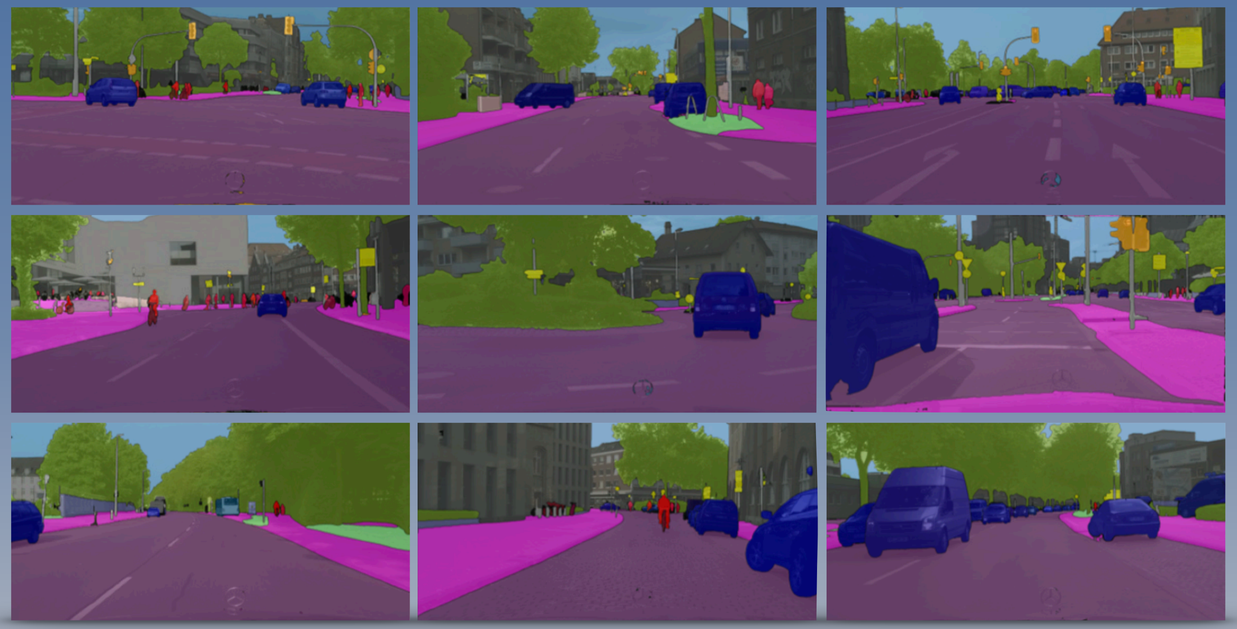


Results & Analysis

Visualization of Panoptic FCN

It achieve fine results on common context and traffic-related scenarios.

Visualization of panoptic results on the Cityscapes val set.



Visualization of panoptic results on the Mapillary Vistas val set.

Thanks

For more questions, please contact

www.yanwei-li.com
ywli@cse.cuhk.edu.hk

Paper

Code