Pyramid Scene Parsing Network

Hengshuang Zhao et al., 2016

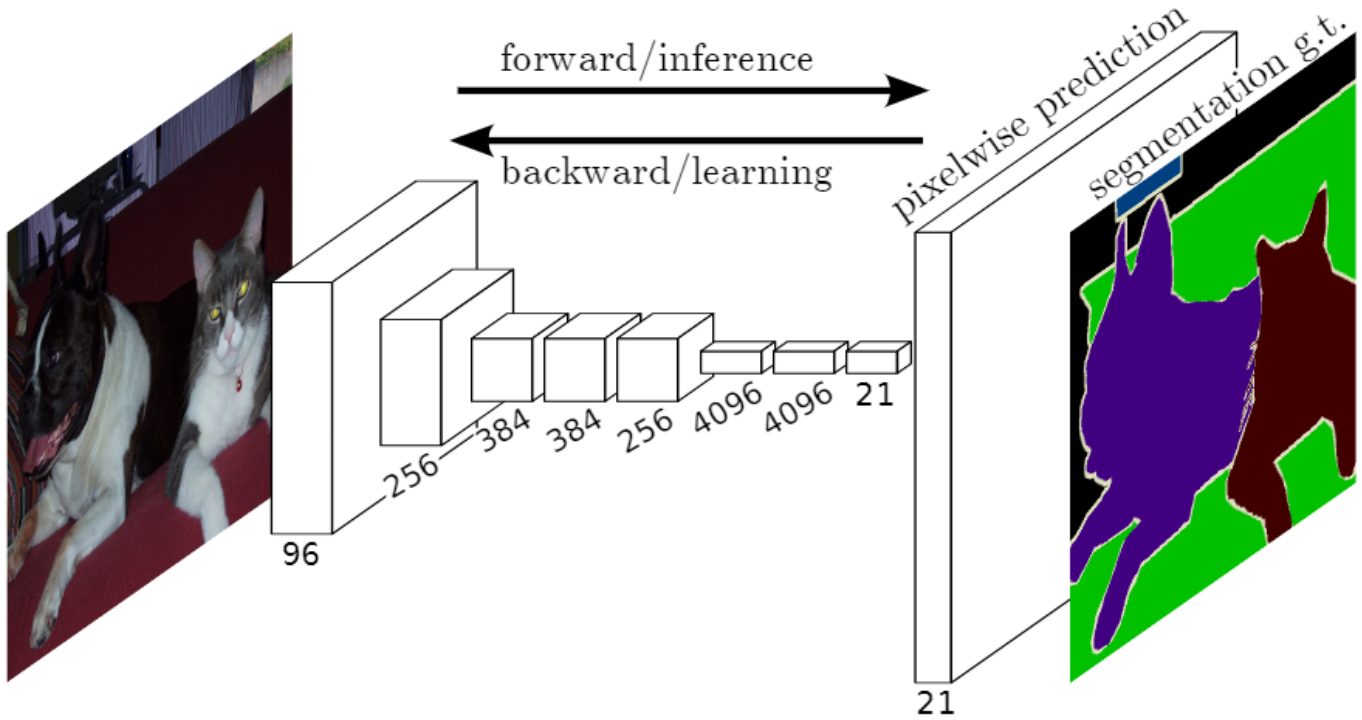
곽대훈

0. Overview

Related works & Data

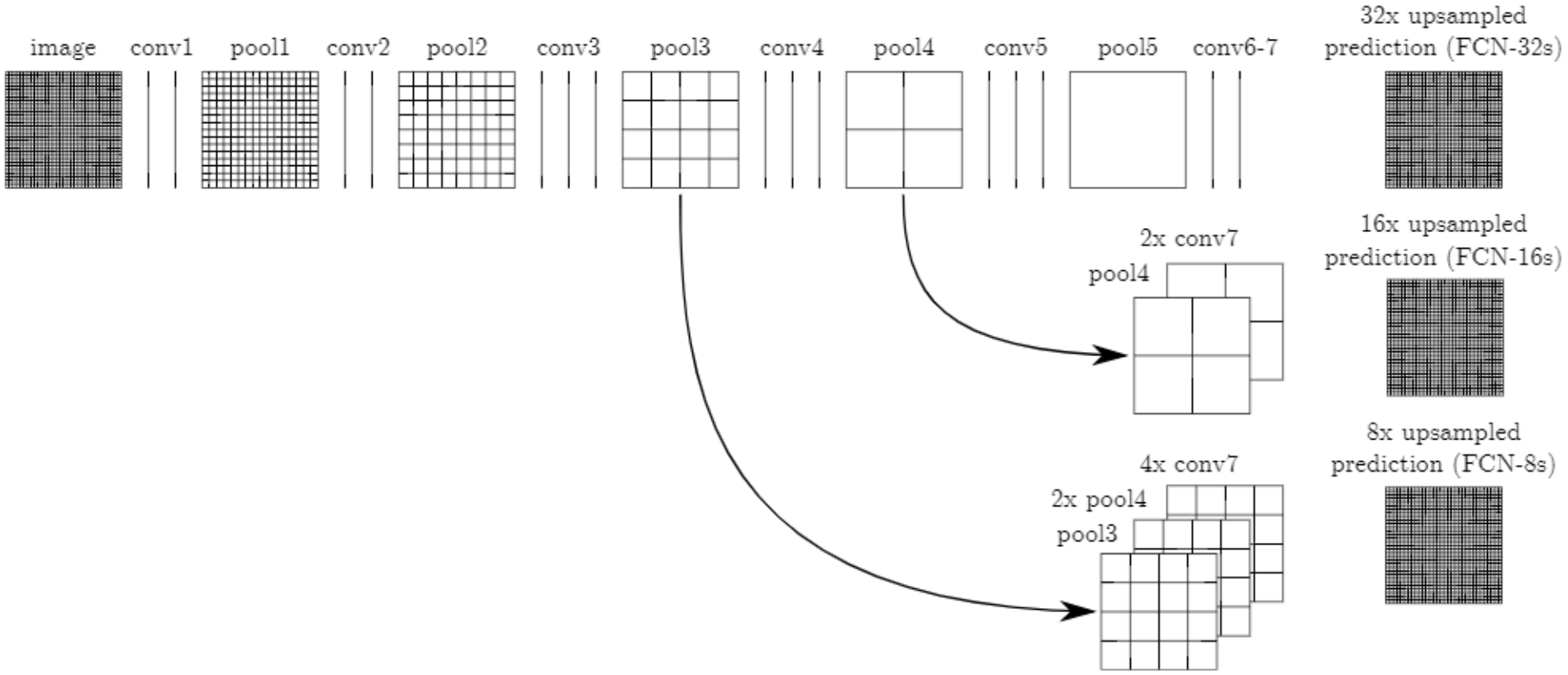
Related work

- FCN



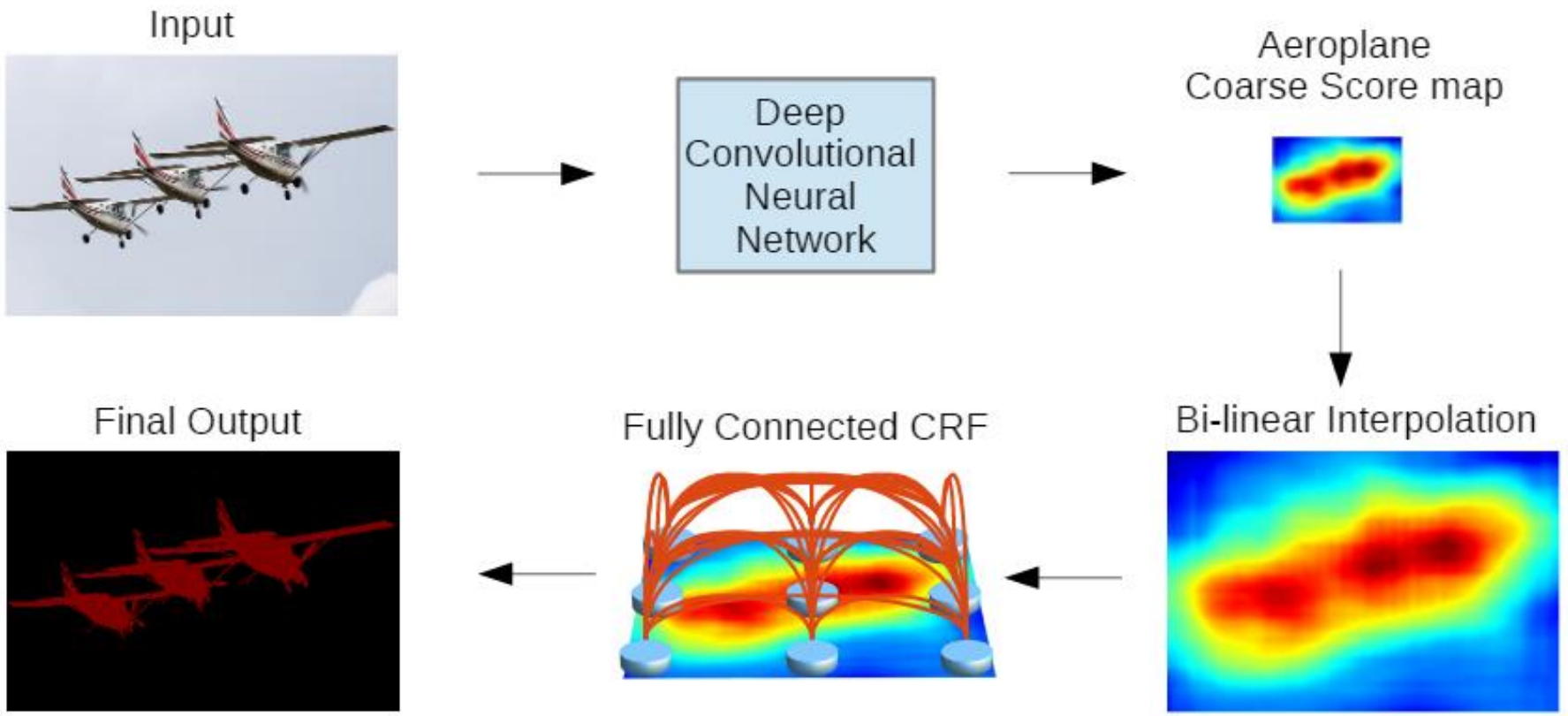
Related work

- FCN



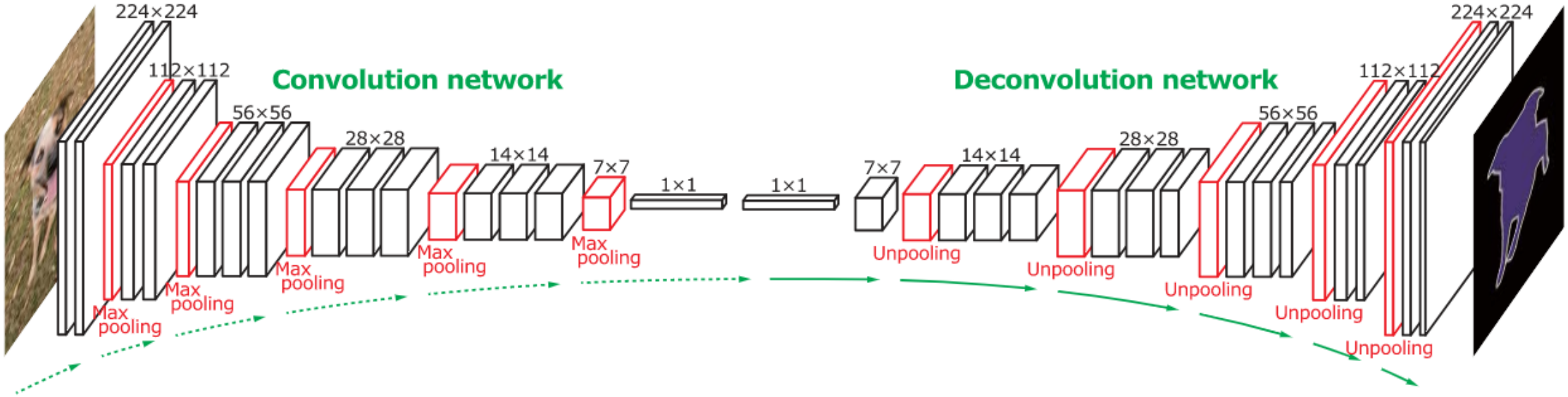
Related work

- Deeplab v1



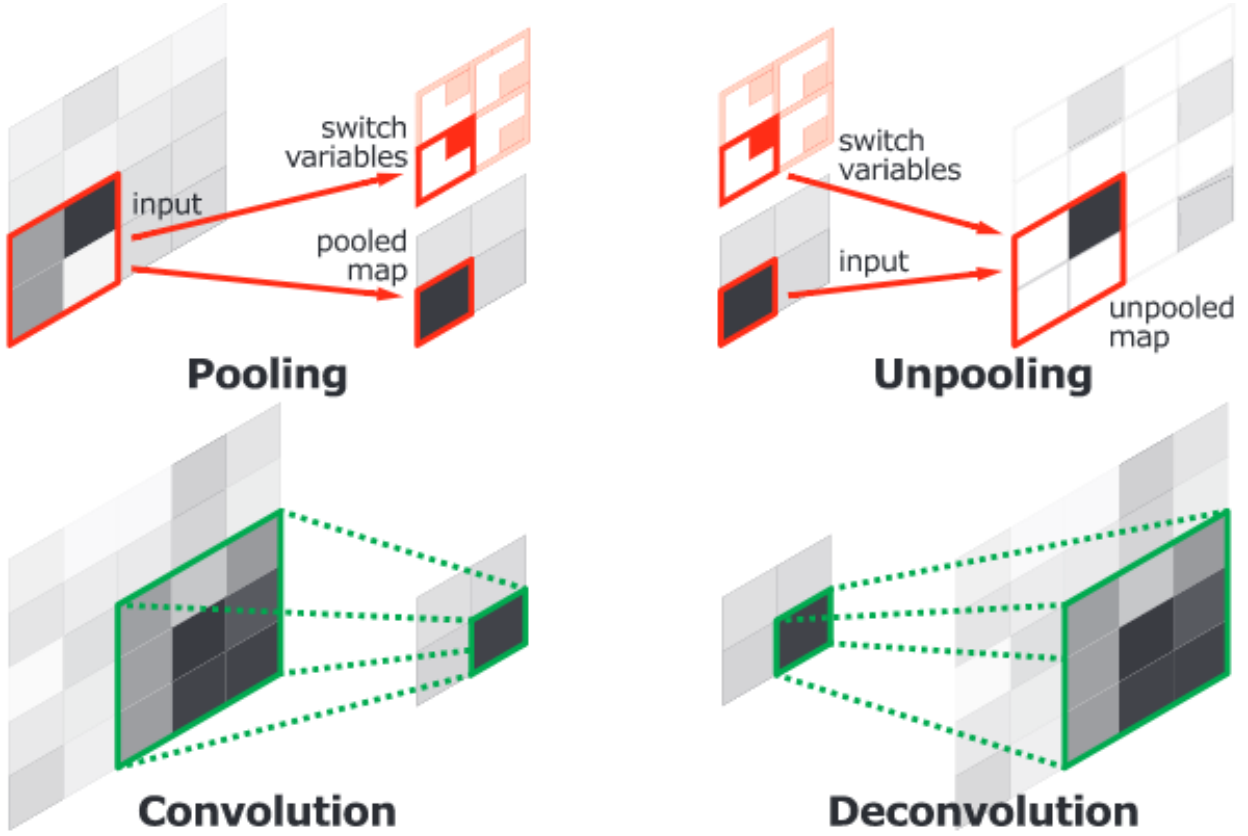
Related work

- Deconvolution networks



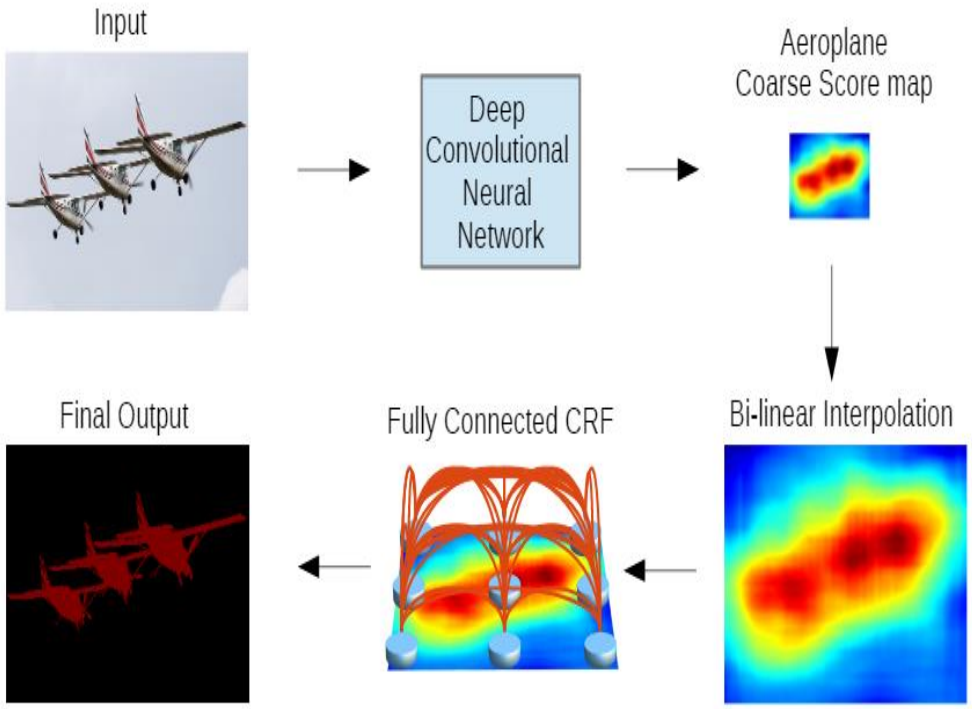
Related work

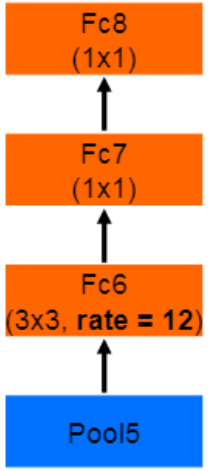
- Deconvolution networks

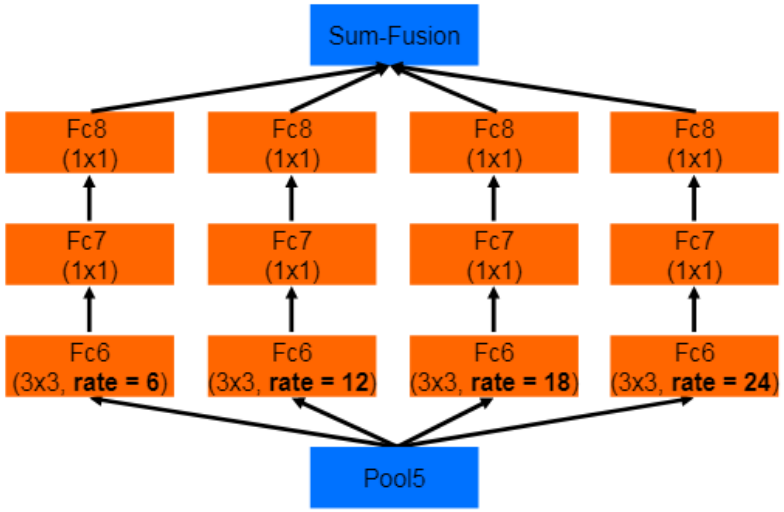


Related work

- Deeplab v2







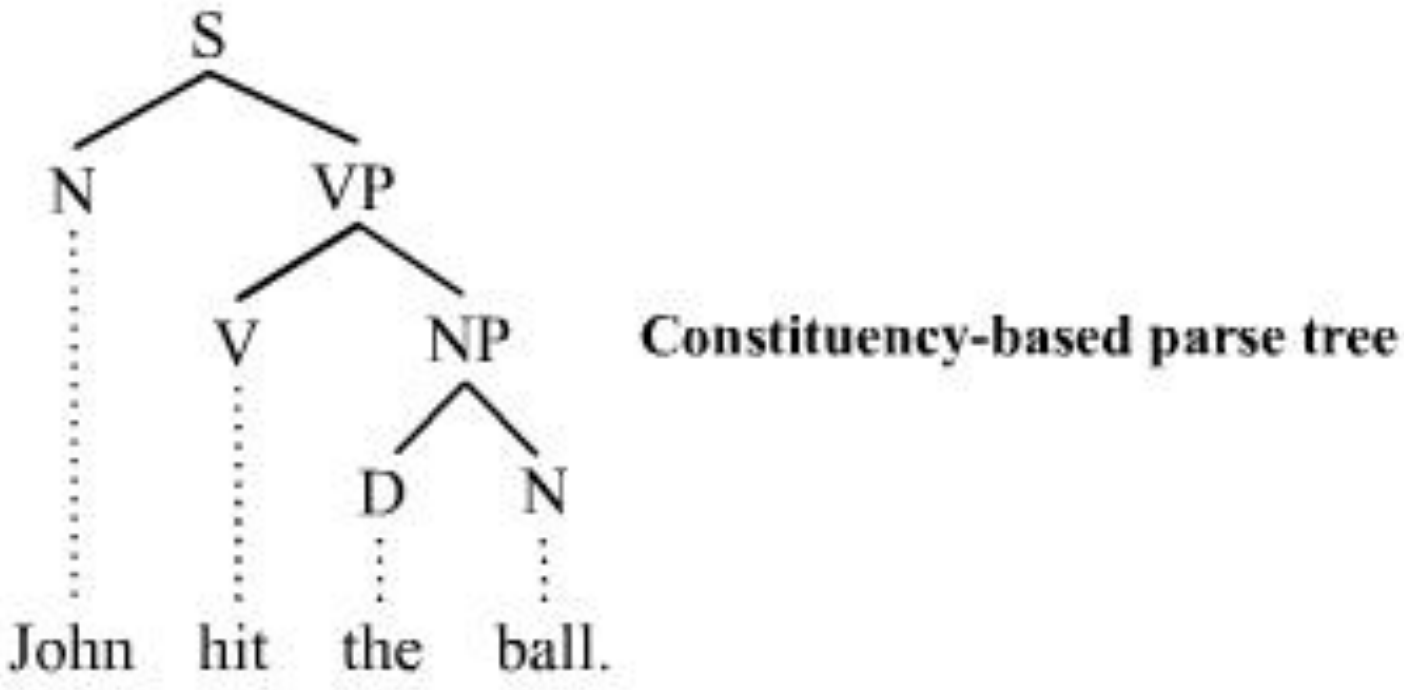
Related work

결국 지금까지 해결하려는 문제는

1. 어떻게 Spatial information loss를 줄일지
2. 어떻게 다양한 scale의 objects를 잡을지
3. 경계선을 잘 못잡는 문제

Data

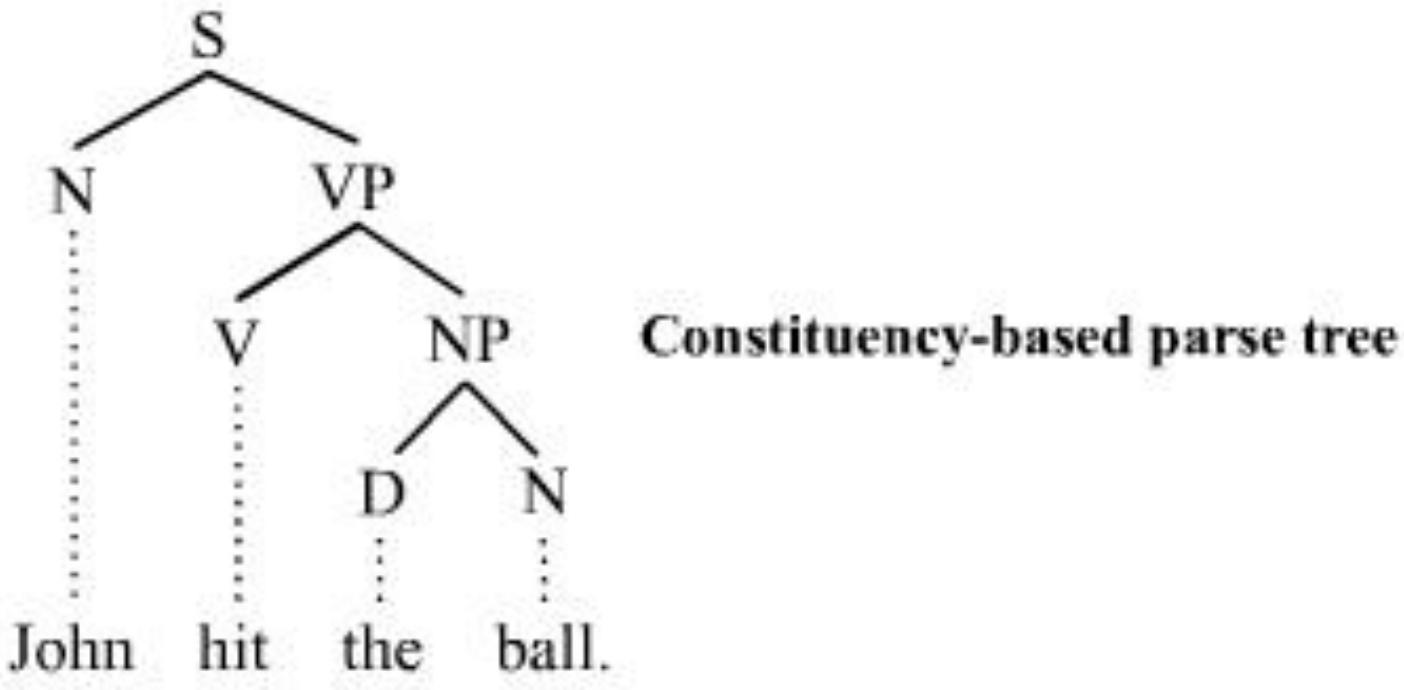
- ADE20K data



Parsing in NLP

Data

- ADE20K data



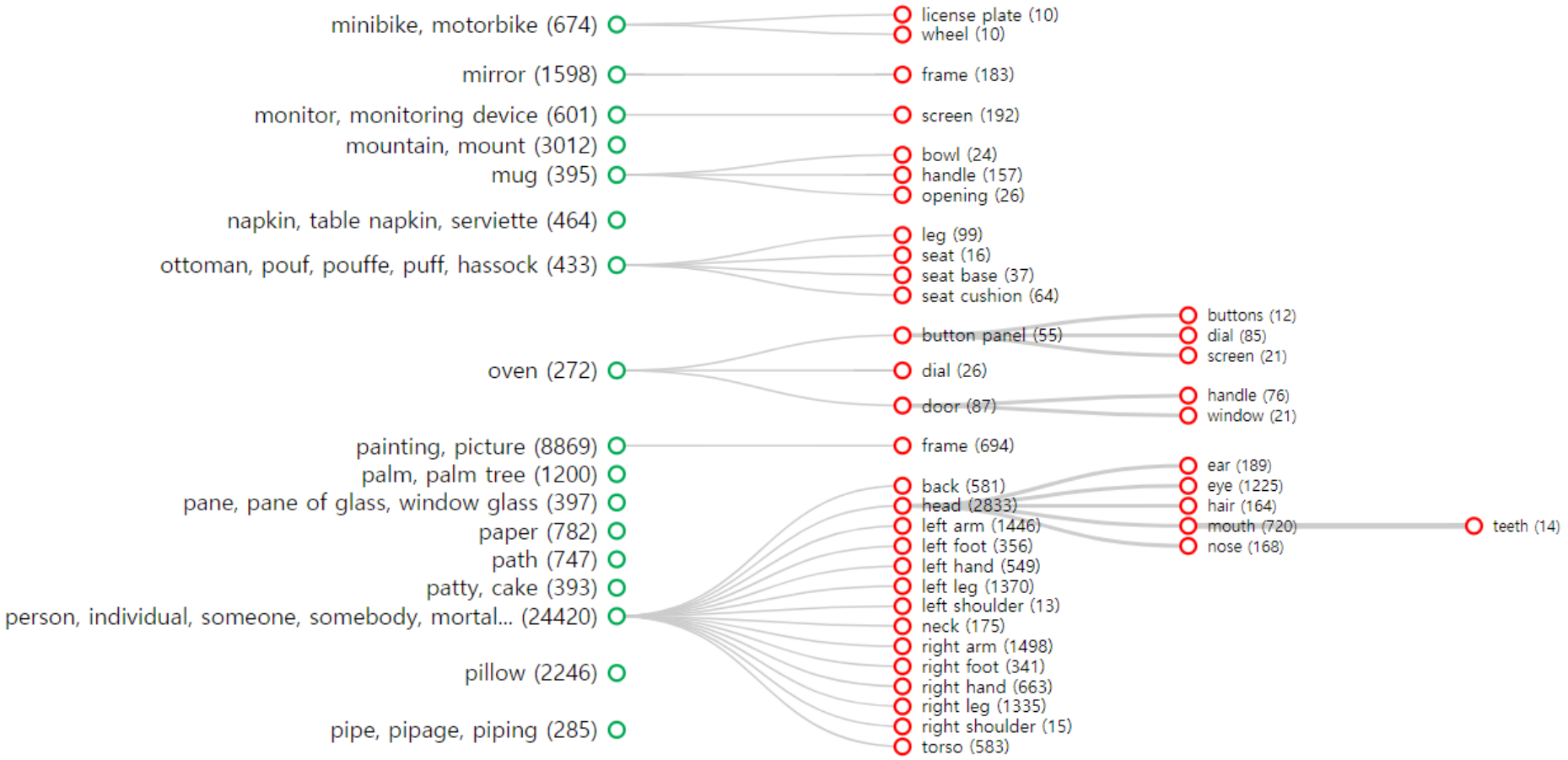
Constituency-based parse tree

Parsing in NLP

-> ADE20K data is Scene parsing data

Data

- ADE20K data



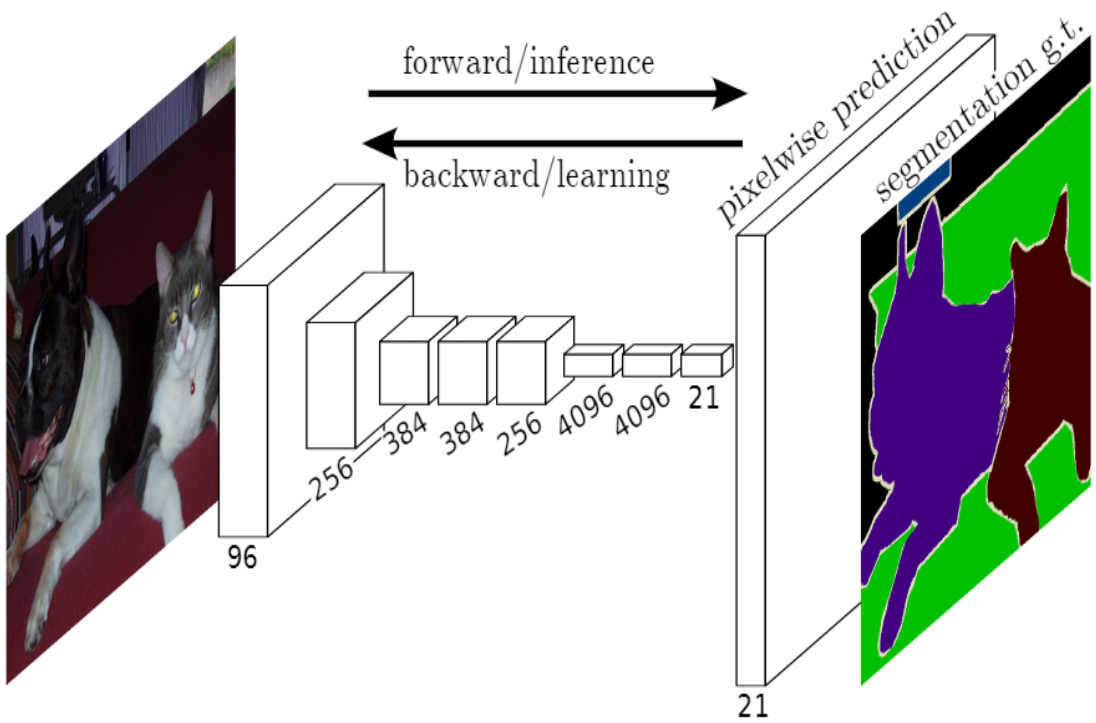
Data

- ADE20K data

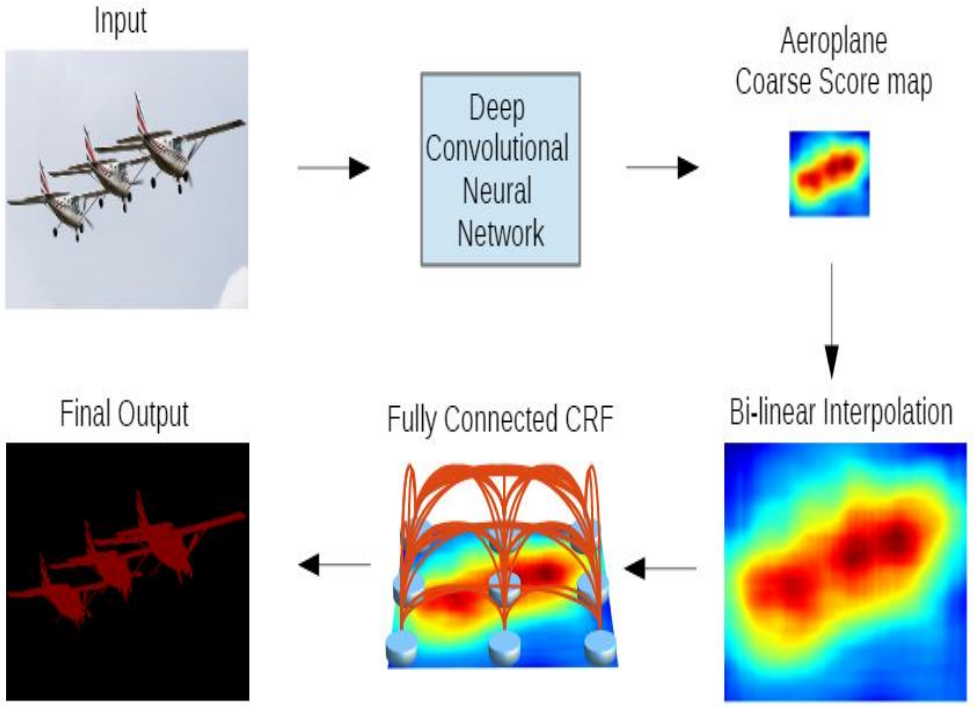
Objects and their pixel ratios:

Parts and their pixel ratios:

Baseline model

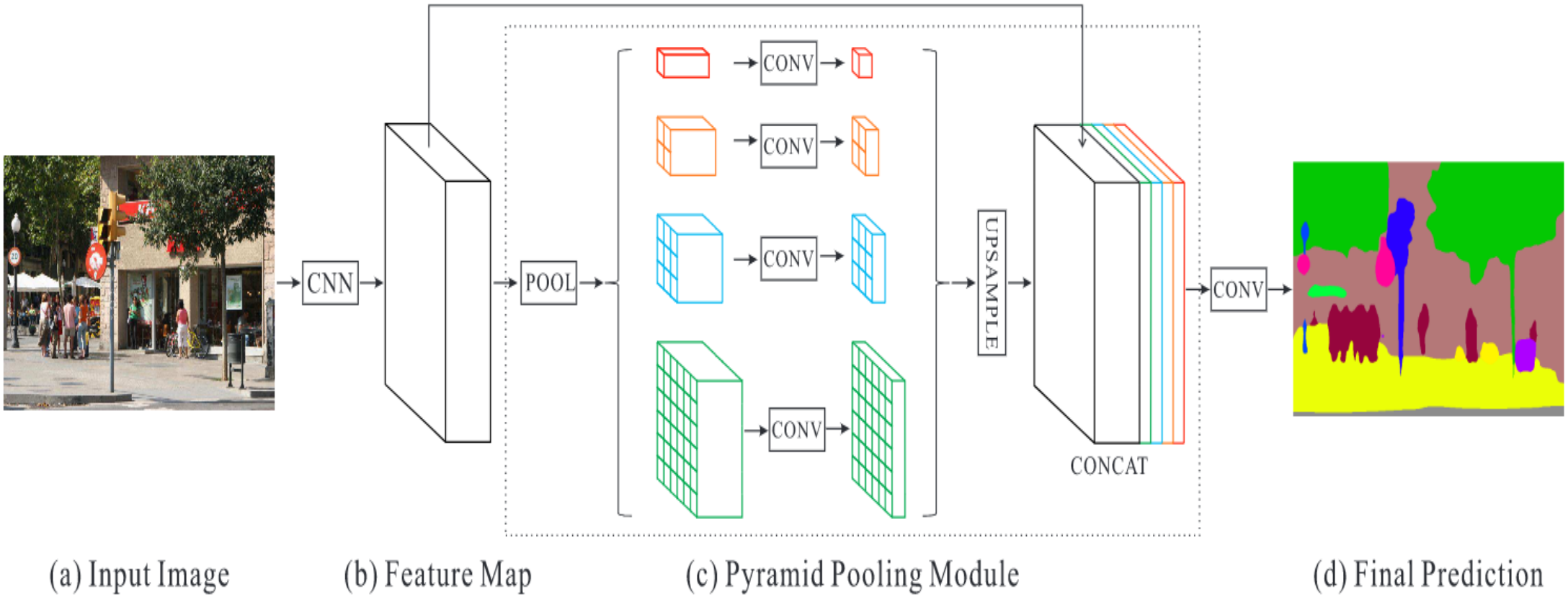


FCN



Deeplab v1

PSPNet



(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

PSPNet

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
FCN [26]	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
Zoom-out [28]	85.6	37.3	83.2	62.5	66.0	85.1	80.7	84.9	27.2	73.2	57.5	78.1	79.2	81.1	77.1	53.6	74.0	49.2	71.7	63.3	69.6
DeepLab [3]	84.4	54.5	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1	59.8	79.0	76.1	83.2	80.8	59.7	82.2	50.4	73.1	63.7	71.6
CRF-RNN [41]	87.5	39.0	79.7	64.2	68.3	87.6	80.8	84.4	30.4	78.2	60.4	80.5	77.8	83.1	80.6	59.5	82.8	47.8	78.3	67.1	72.0
DeconvNet [30]	89.9	39.3	79.7	63.9	68.2	87.4	81.2	86.1	28.5	77.0	62.0	79.0	80.3	83.6	80.2	58.8	83.4	54.3	80.7	65.0	72.5
GCRF [36]	85.2	43.9	83.3	65.2	68.3	89.0	82.7	85.3	31.1	79.5	63.3	80.5	79.3	85.5	81.0	60.5	85.5	52.0	77.3	65.1	73.2
DPN [25]	87.7	59.4	78.4	64.9	70.3	89.3	83.5	86.1	31.7	79.9	62.6	81.9	80.0	83.5	82.3	60.5	83.2	53.4	77.9	65.0	74.1
Piecewise [20]	90.6	37.6	80.0	67.8	74.4	92.0	85.2	86.2	39.1	81.2	58.9	83.8	83.9	84.3	84.8	62.1	83.2	58.2	80.8	72.3	75.3
PSPNet	**91.8**	**71.9**	**94.7**	**71.2**	**75.8**	**95.2**	**89.9**	**95.9**	**39.3**	**90.7**	**71.7**	**90.5**	**94.5**	**88.8**	**89.6**	**72.8**	**89.6**	**64.0**	**85.1**	**76.3**	**82.6**
CRF-RNN† [41]	90.4	55.3	88.7	68.4	69.8	88.3	82.4	85.1	32.6	78.5	64.4	79.6	81.9	86.4	81.8	58.6	82.4	53.5	77.4	70.1	74.7
BoxSup† [7]	89.8	38.0	89.2	68.9	68.0	89.6	83.0	87.7	34.4	83.6	67.1	81.5	83.7	85.2	83.5	58.6	84.9	55.8	81.2	70.7	75.2
Dilation8† [40]	91.7	39.6	87.8	63.1	71.8	89.7	82.9	89.8	37.2	84.0	63.0	83.3	89.0	83.8	85.1	56.8	87.6	56.0	80.2	64.7	75.3
DPN† [25]	89.0	61.6	87.7	66.8	74.7	91.2	84.3	87.6	36.5	86.3	66.1	84.4	87.8	85.6	85.4	63.6	87.3	61.3	79.4	66.4	77.5
Piecewise† [20]	94.1	40.7	84.1	67.8	75.9	93.4	84.3	88.4	42.5	86.4	64.7	85.4	89.0	85.8	86.0	67.5	90.2	63.8	80.9	73.0	78.0
FCRNs† [38]	91.9	48.1	93.4	69.3	75.5	94.2	87.5	92.8	36.7	86.9	65.2	89.1	90.2	86.5	87.2	64.6	90.1	59.7	85.5	72.7	79.1
LRR† [9]	92.4	45.1	94.6	65.2	75.8	**95.1**	89.1	92.3	39.0	85.7	70.4	88.6	89.4	88.6	86.6	65.8	86.2	57.4	85.7	77.3	79.3
DeepLab† [4]	92.6	60.4	91.6	63.4	76.3	95.0	88.4	92.6	32.7	88.5	67.6	89.6	92.1	87.0	87.4	63.3	88.3	60.0	86.8	74.5	79.7
PSPNet†	**95.8**	**72.7**	**95.0**	**78.9**	**84.4**	94.7	**92.0**	**95.7**	**43.1**	**91.0**	**80.3**	**91.3**	**96.3**	**92.3**	**90.1**	**71.5**	**94.4**	**66.9**	**88.8**	**82.0**	**85.4**

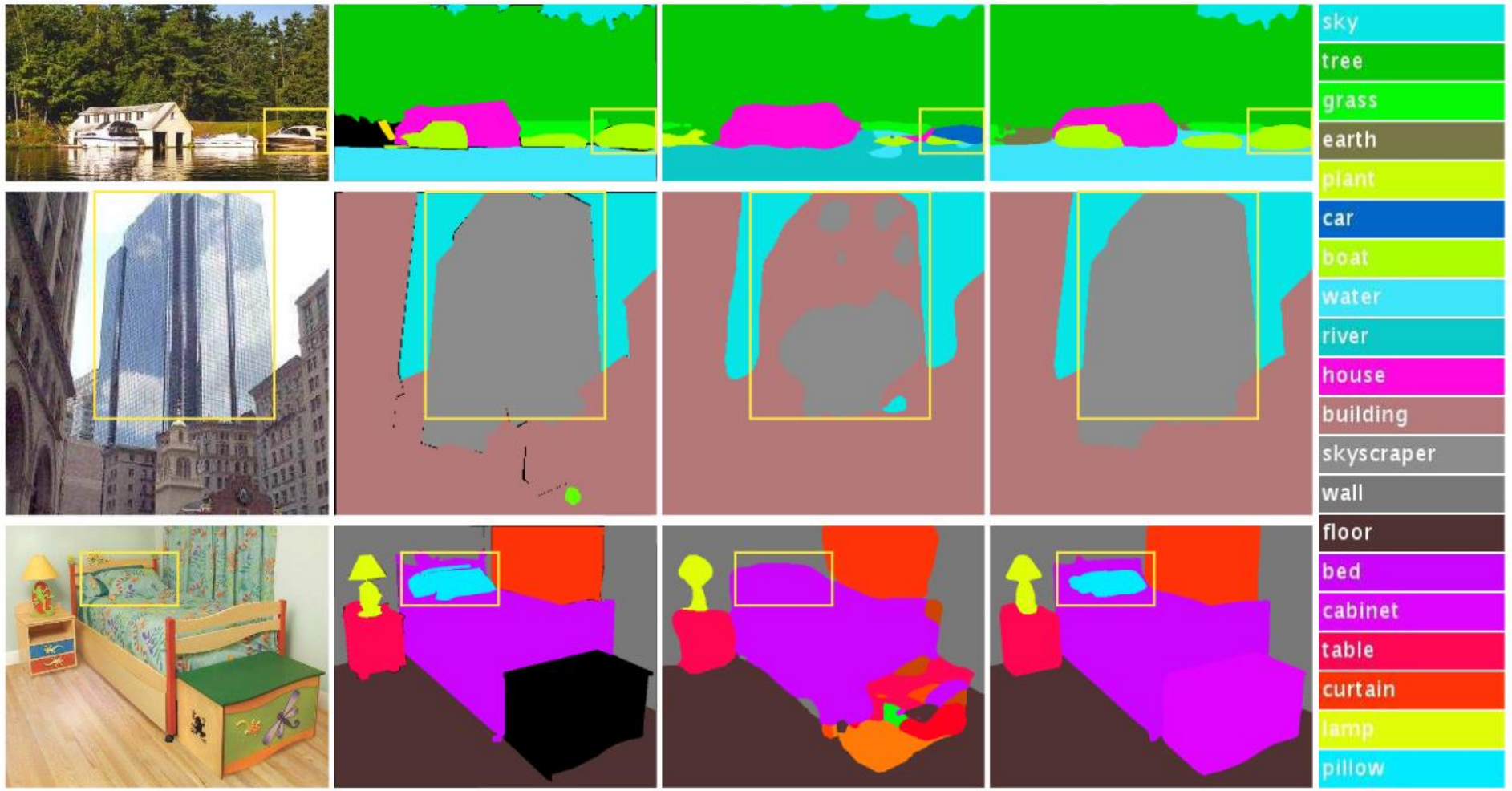
Method	IoU cla.	iIoU cla.	IoU cat.	iIoU cat.
CRF-RNN [41]	62.5	34.4	82.7	66.0
FCN [26]	65.3	41.7	85.7	70.1
SiCNN [16]	66.3	44.9	85.0	71.2
DPN [25]	66.8	39.1	86.0	69.1
Dilation10 [40]	67.1	42.0	86.5	71.1
LRR [9]	69.7	48.0	88.2	74.7
DeepLab [4]	70.4	42.6	86.4	67.7
Piecewise [20]	71.6	51.7	87.3	74.1
PSPNet	**78.4**	**56.7**	**90.6**	**78.6**
LRR‡ [9]	71.8	47.9	88.4	73.9
PSPNet‡	**80.2**	**58.1**	**90.6**	**78.2**

1. Introduction

Motivation & Contribution

PSPNet

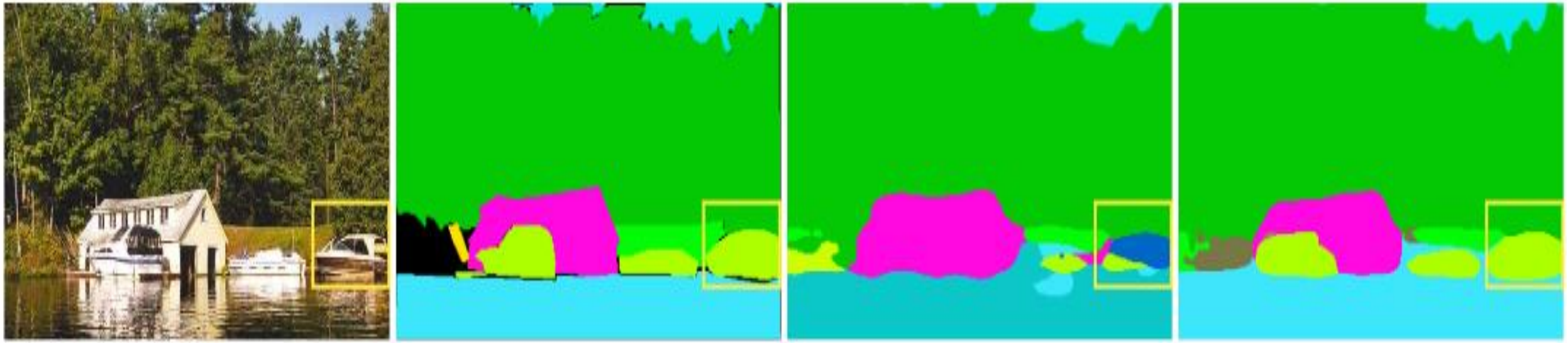
1. Introduction : motivation



(a) Image (b) Ground Truth (c) FCN (d) PSPNet (e) ColorMap

PSPNet

1. Introduction : motivation

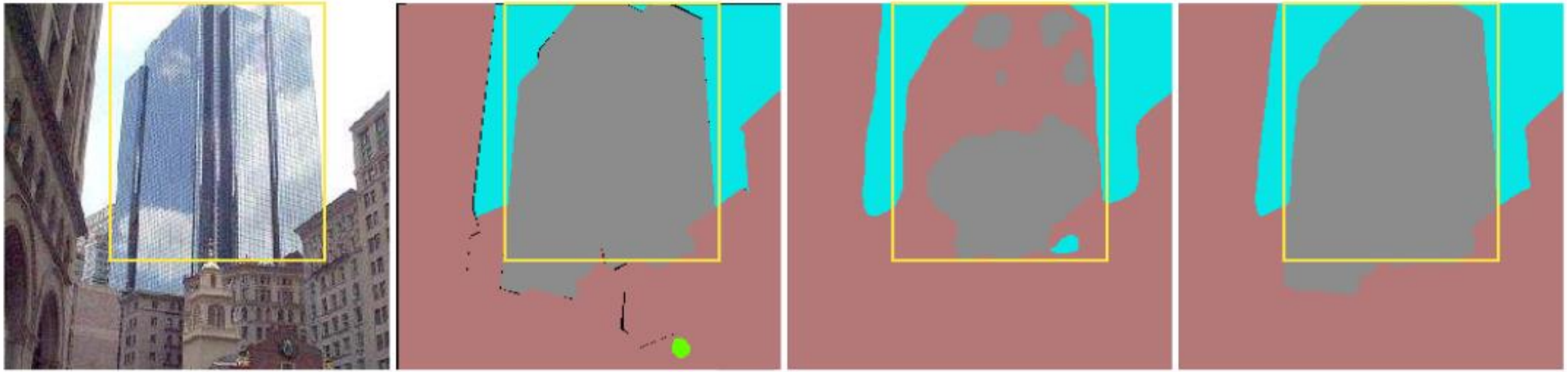


Mismatched Relationship

Lack of the ability to collect contextual information increases the chance of misclassification.

PSPNet

1. Introduction : motivation



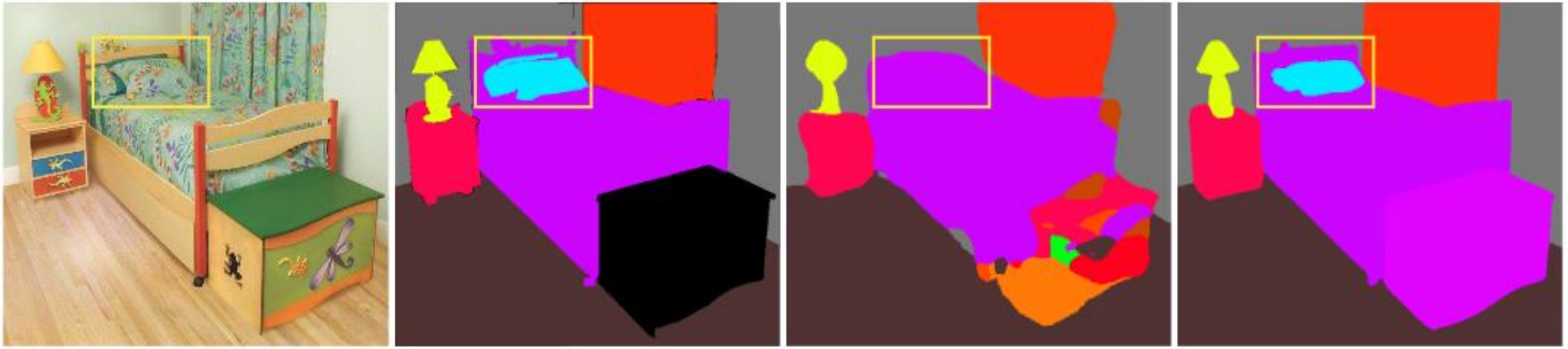
Confusion Categories

There are many class label pairs in the ADE20K dataset
that are confusing in classification. (field and earth; mountain and hill; wall, house, building and skyscraper)
FCN predicts the object in the box as part of skyscraper and part of building.

This problem can be remedied **by utilizing the relationship between categories.**

PSPNet

1. Introduction : motivation



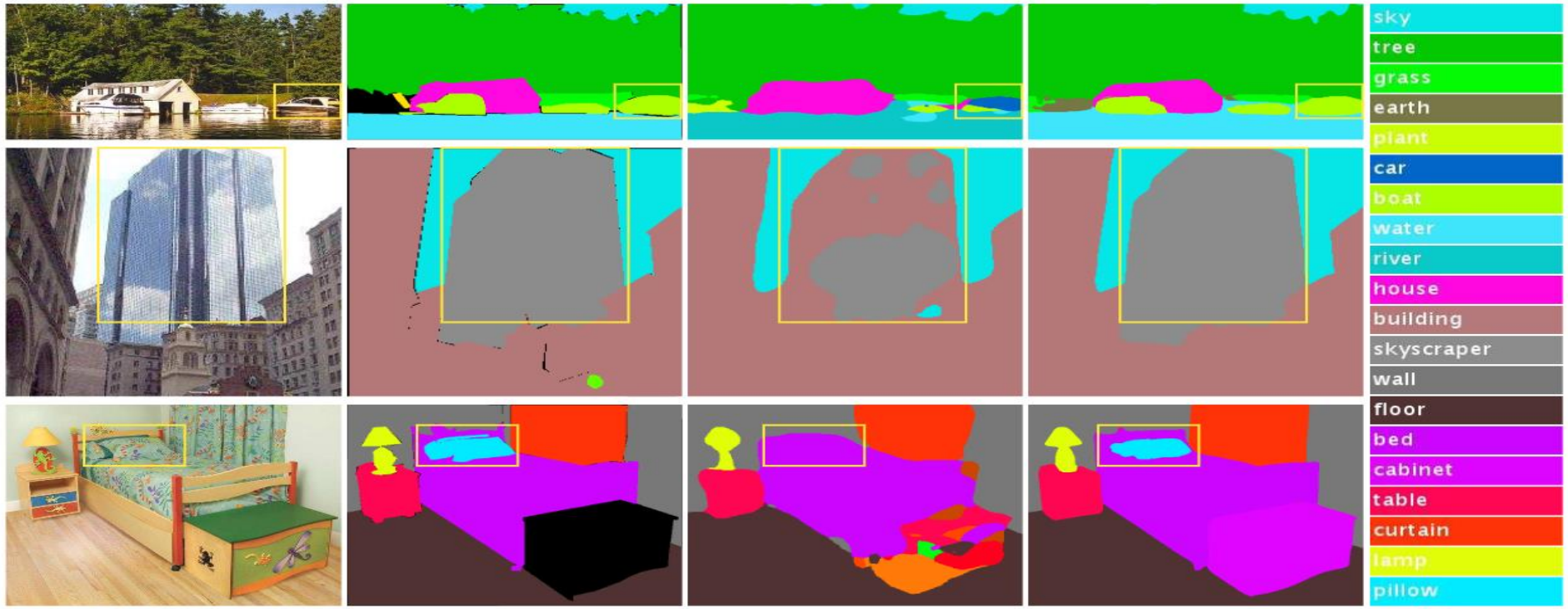
Inconspicuous Classes

Small size-things & big objects or stuff

To improve performance for remarkably small or large objects, one should pay much attention to **different sub-regions that contain inconspicuous-category stuff**.

PSPNet

1. Introduction : motivation



(a) Image (b) Ground Truth (c) FCN (d) PSPNet (e) ColorMap

Many errors are partially or completely related to **contextual relationship and global information for different receptive fields.**

PSPNet

1. Introduction : contribution

- We propose a **pyramid scene parsing network** to embed difficult scenery context features in an FCN based pixel prediction framework.
- We develop an **effective optimization strategy** for deep ResNet based on deeply supervised loss.
- We build a practical system for **state-of-the-art scene parsing and semantic segmentation** where **all crucial implementation details are included.**

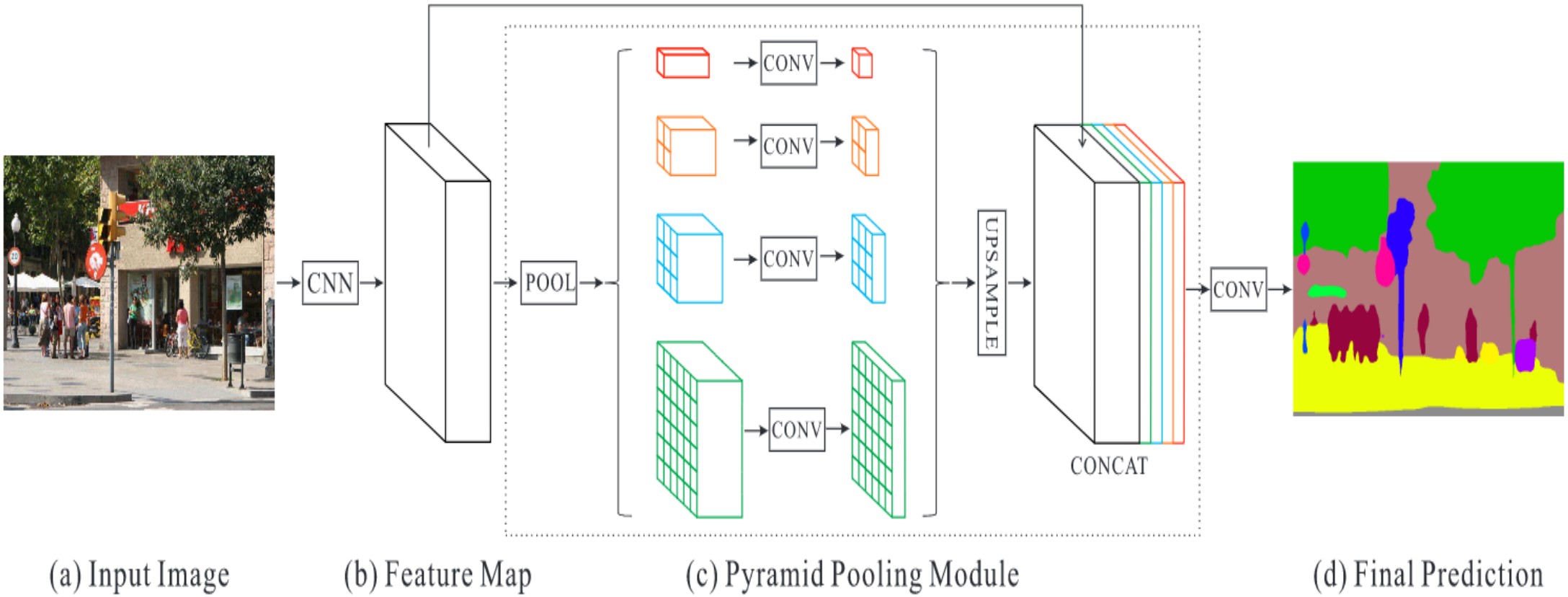
2. Model

(PSPNet – Pyramid Scene Parsing Network)

" With above analysis, we introduce the pyramid module "
which empirically proves to be an effective global contextual prior.

PSPNet

2. Model

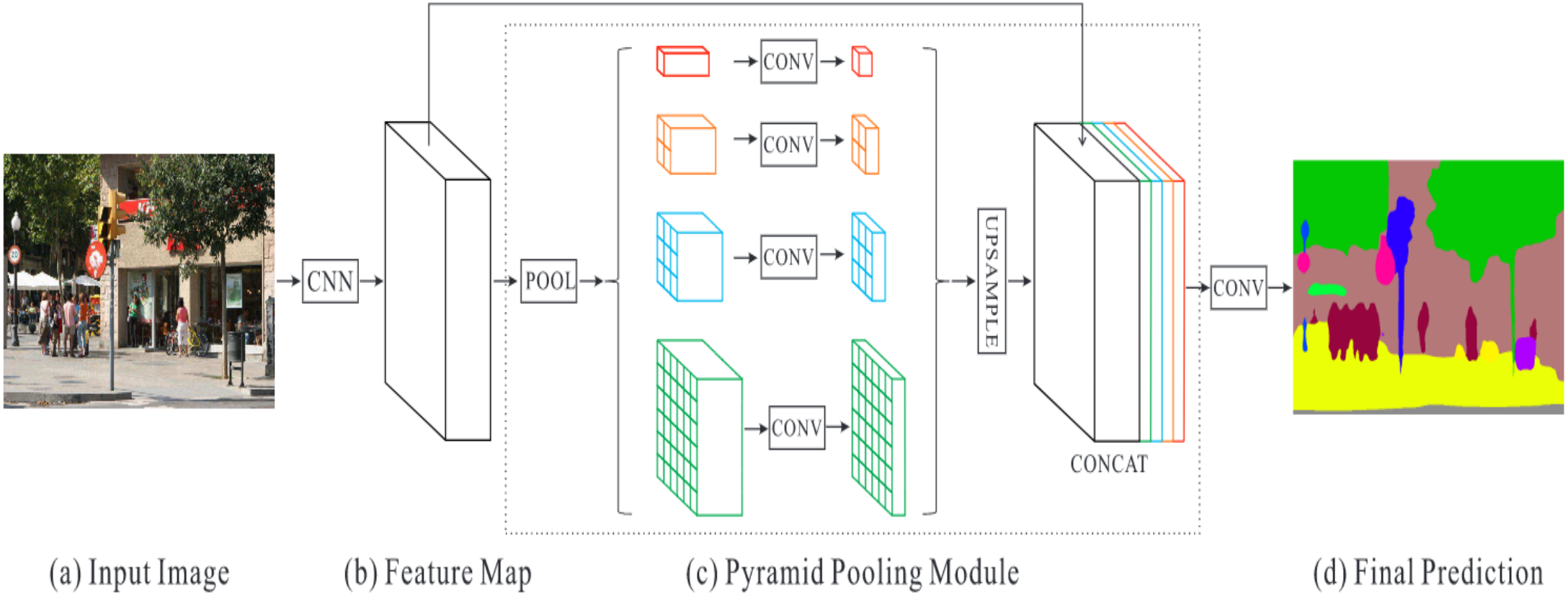


(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

PSPNet

2. Model

Pre-trained Resnet

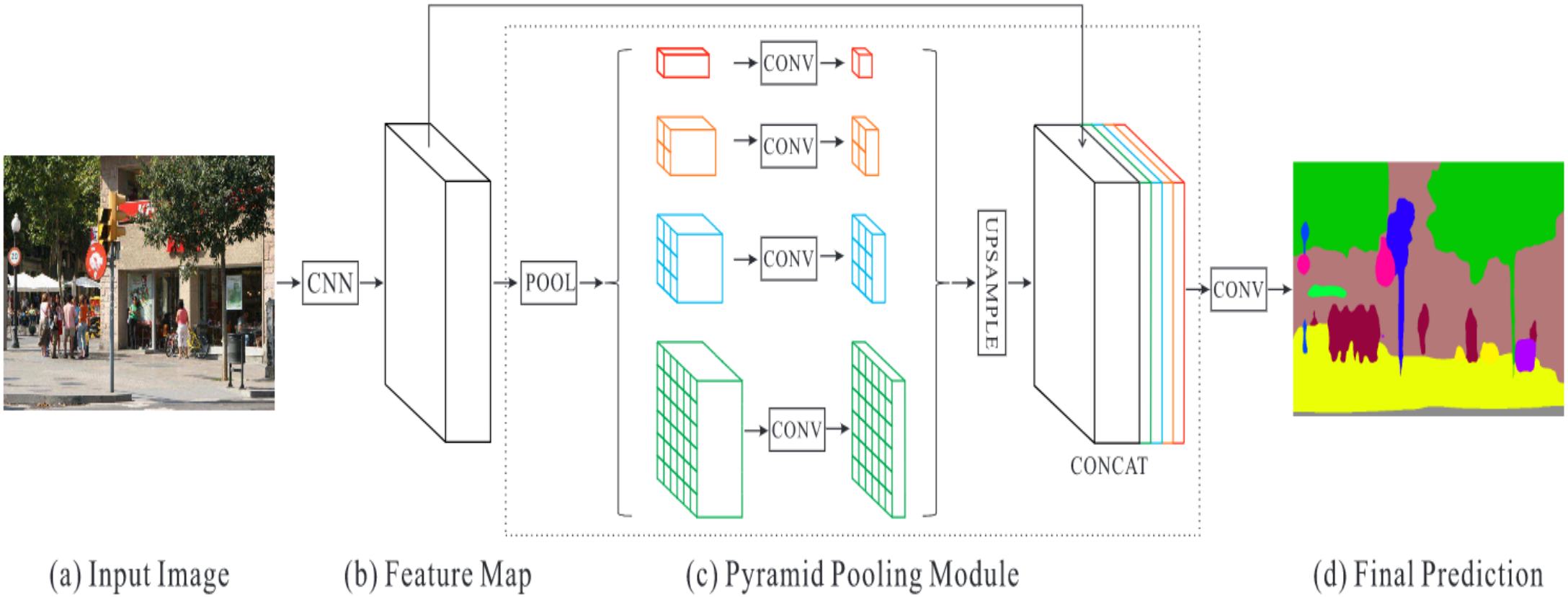


(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

PSPNet

2. Model

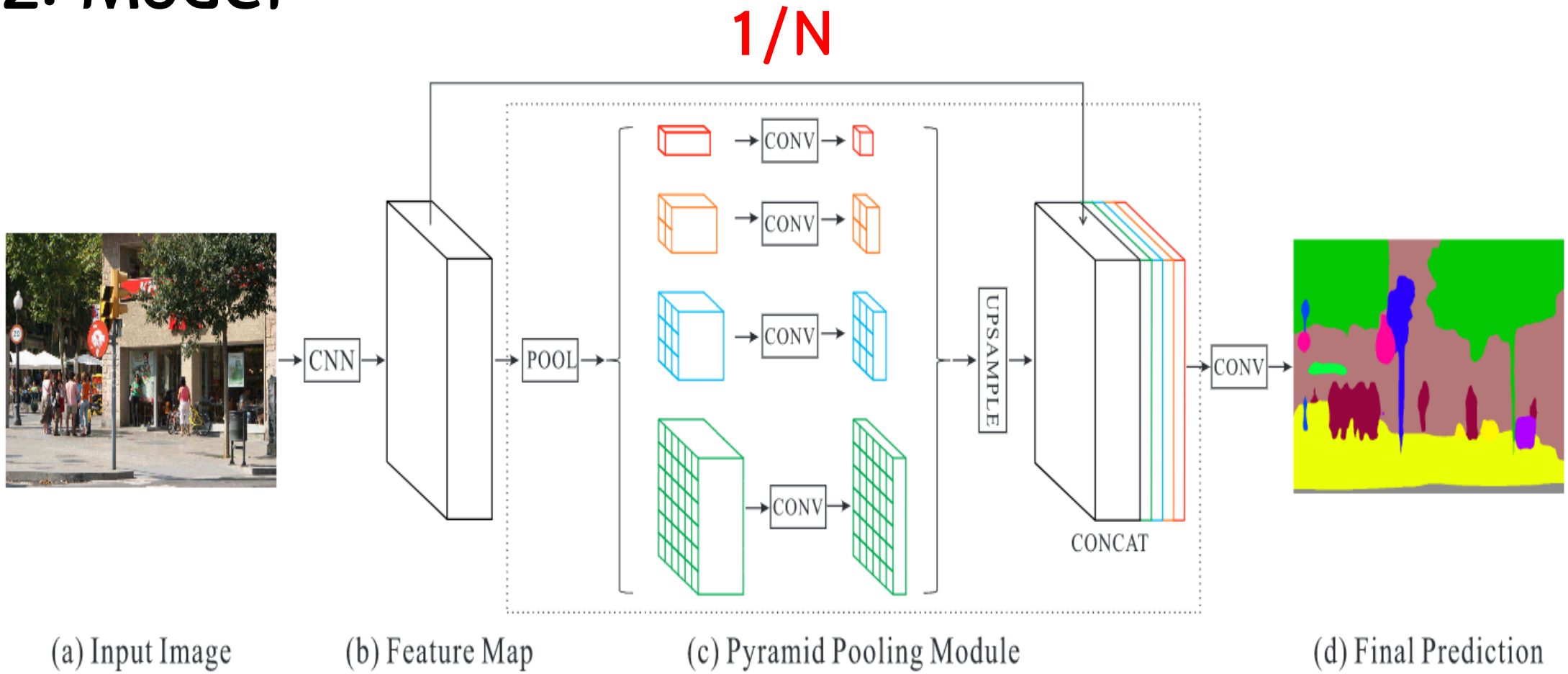
1, 2, 3, 6



(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

PSPNet

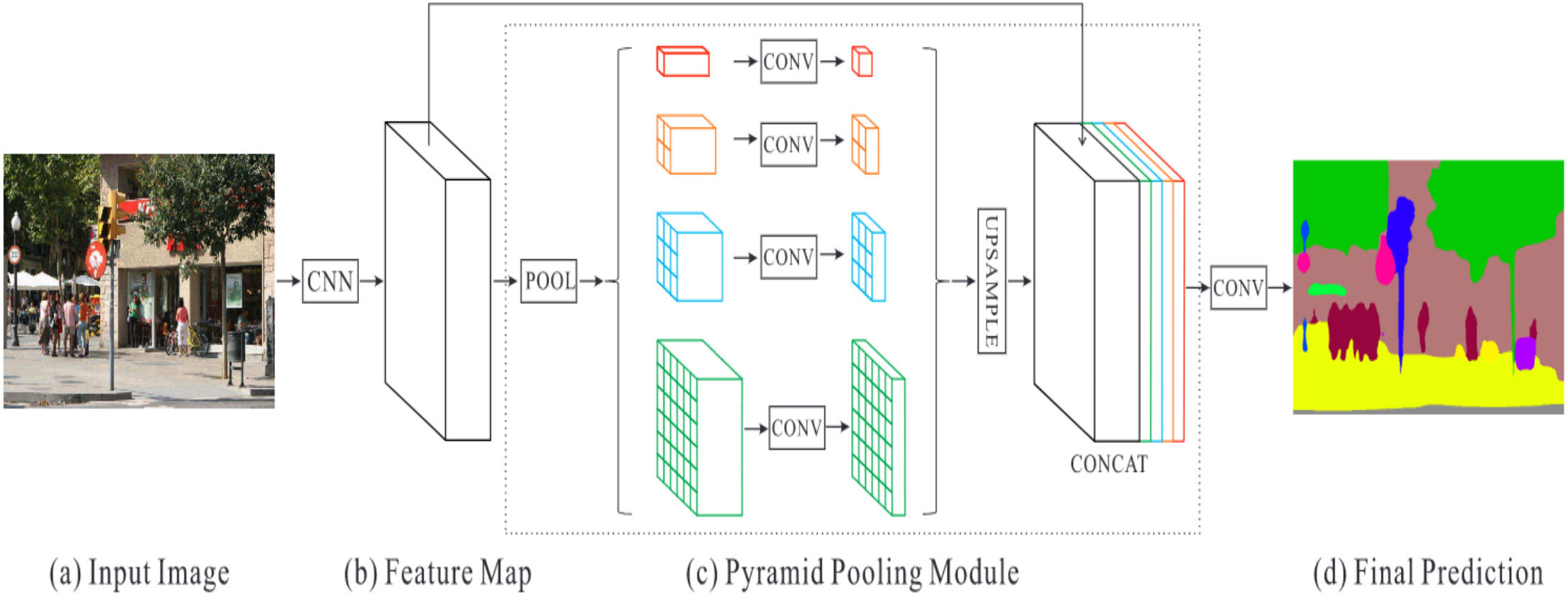
2. Model



PSPNet

2. Model

Bilinear Interpolation



(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

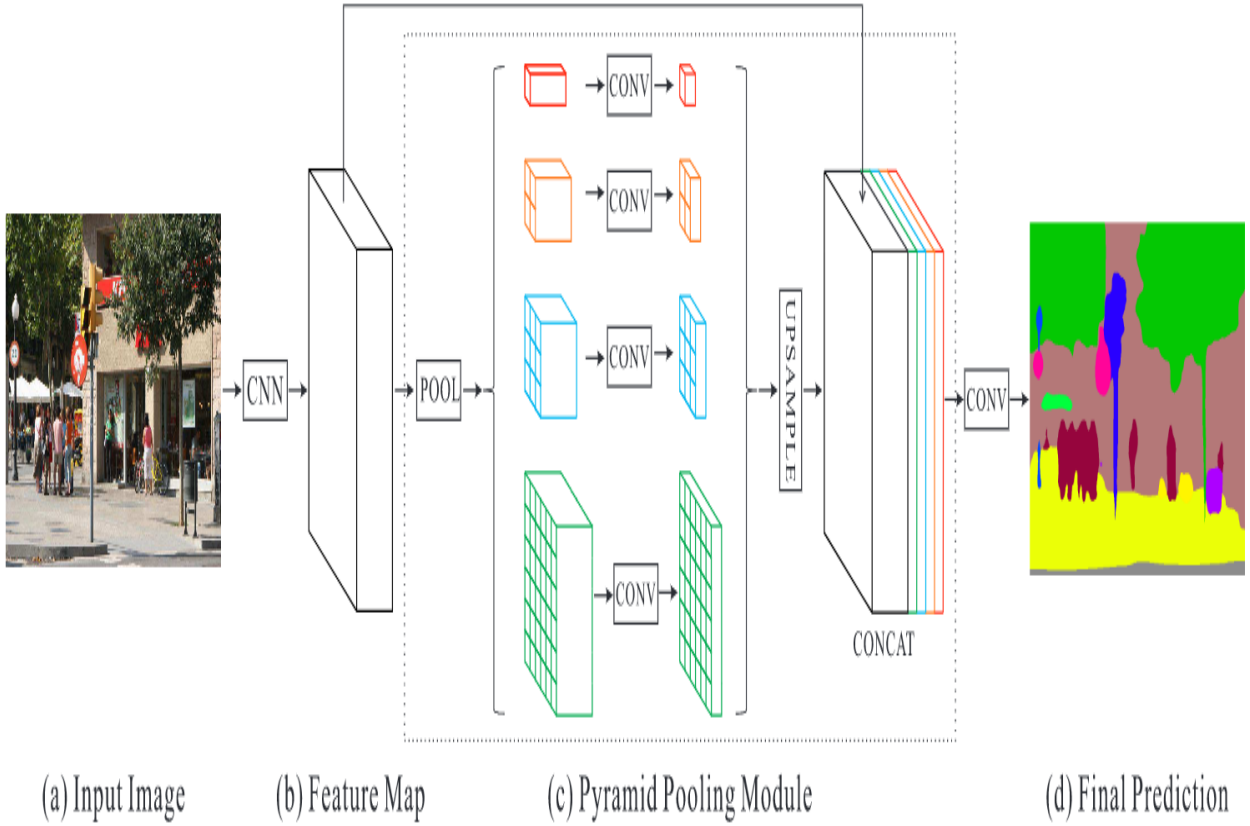
PSPNet

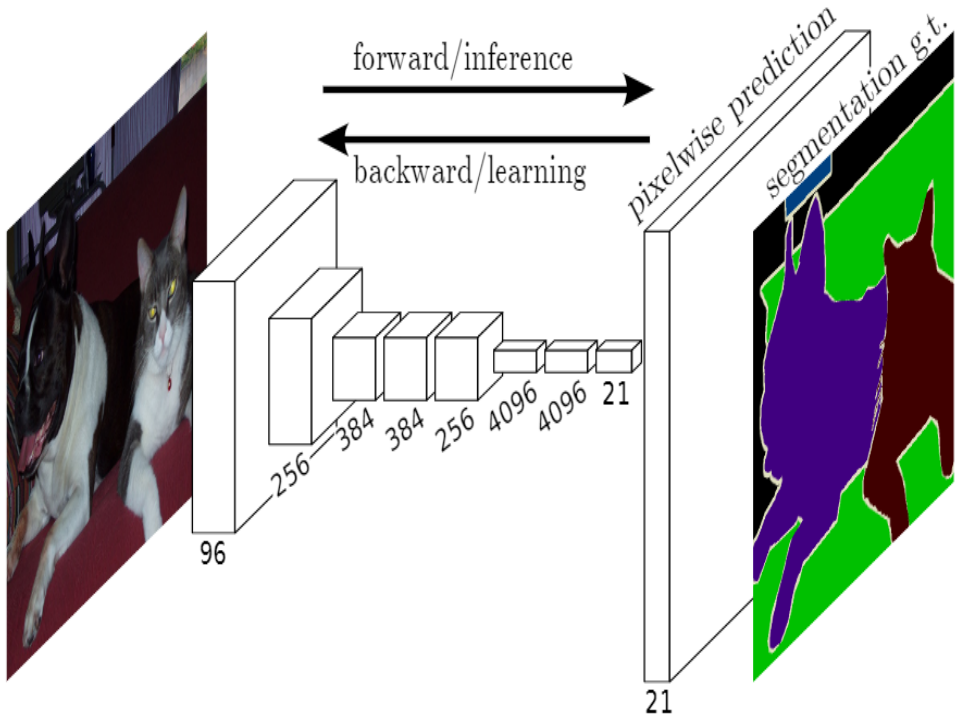
2. Model

- We use a pretrained ResNet model with the dilated network strategy [3, 40] to extract the feature map.
- https://github.com/hellochick/PSPNet-tensorflow/blob/master/model.py
- https://github.com/hellochick/PSPNet-tensorflow/blob/master/train.py

PSPNet

2. Model

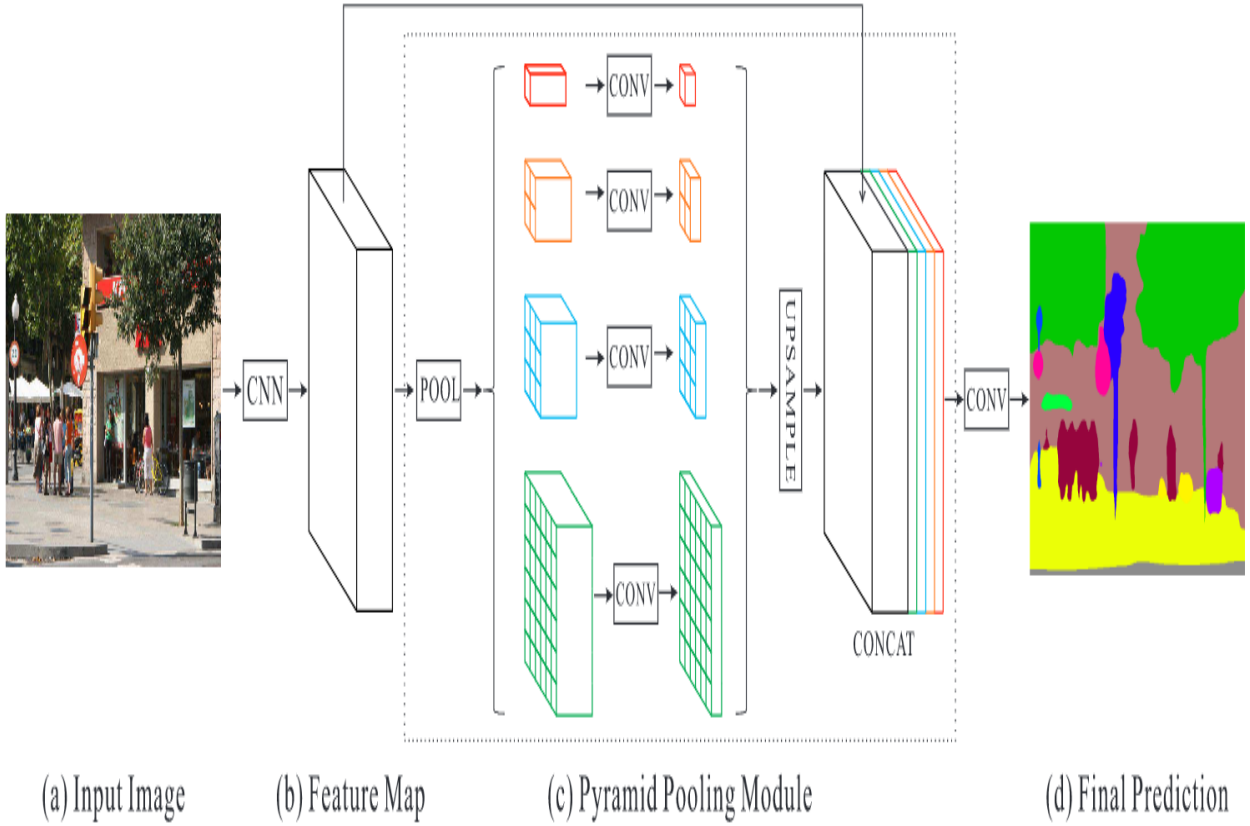


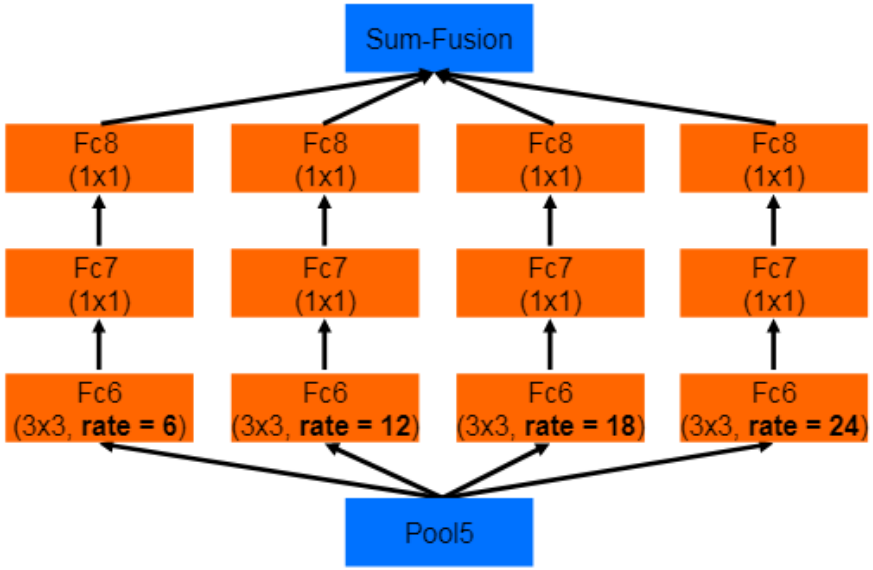


FCN

PSPNet

2. Model





ASPP

PSPNet

2. Model

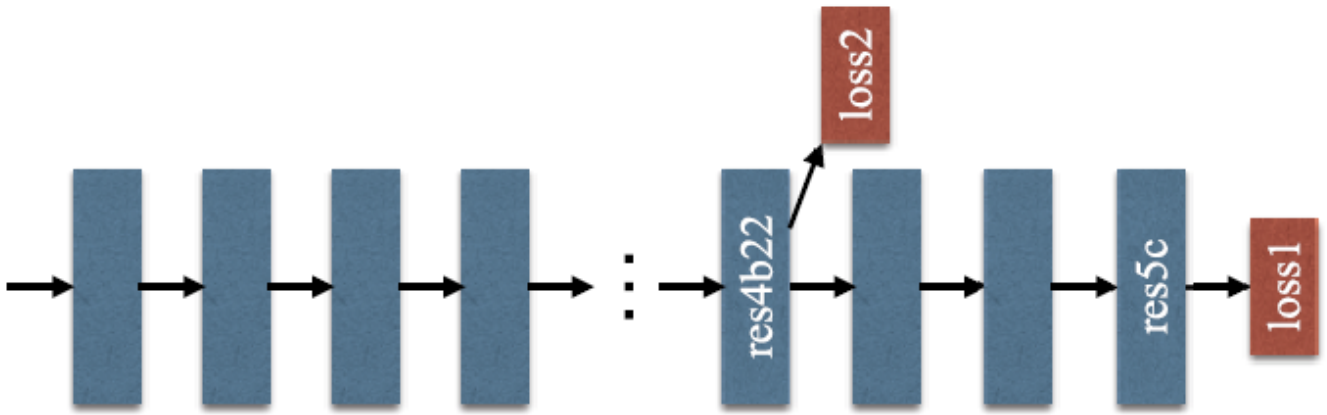


Figure 4. Illustration of auxiliary loss in ResNet101. Each blue box denotes a residue block. The auxiliary loss is added after the res4b22 residue block.

PSPNet

2. Model

Abandon this branch
in the testing phase

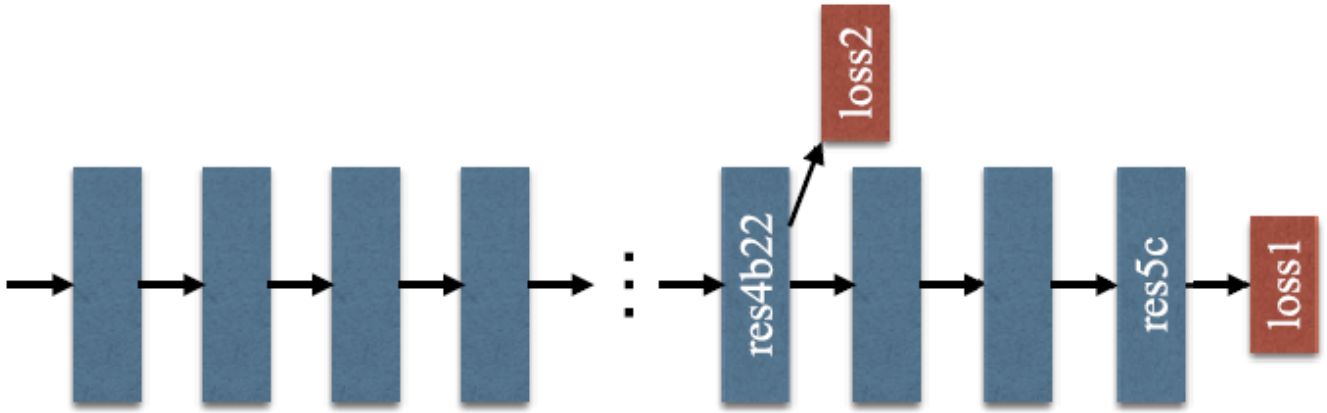


Figure 4. Illustration of auxiliary loss in ResNet101. Each blue box denotes a residue block. The auxiliary loss is added after the res4b22 residue block.

3. Results

Implementation details & Experiments
(with ADE20K dataset)

PSPNet

3. Results : Implementation Details

For a practical deep learning system, devil is always in the details. Our implementation is based on the public platform Caffe [15]. Inspired by [4], we use the "poly" learning rate policy where current learning rate equals to the base one multiplying $(1 - \frac{iter}{max\_iter})^{power}$. We set base learning rate to 0.01 and power to 0.9. The performance can be improved by increasing the iteration number, which is set to 150K for ImageNet experiment, 30K for PASCAL VOC and 90K for Cityscapes. Momentum and weight decay are set to 0.9 and 0.0001 respectively. For data augmentation, we adopt random mirror and random resize between 0.5 and 2 for all datasets, and additionally add random rotation between -10 and 10 degrees, and random Gaussian blur for ImageNet and PASCAL VOC. This comprehensive data augmentation scheme makes the network resist overfitting. Our network contains dilated convolution following [4].

PSPNet

3. Results : Implementation Details

During the course of experiments, we notice that an appropriately large "cropsize" can yield good performance and "batchsize" in the batch normalization [14] layer is of great importance. Due to limited physical memory on GPU cards, we set the "batchsize" to 16 during training. To achieve this, we modify Caffe from [37] together with branch [4] and make it support batch normalization on data gathered from multiple GPUs based on OpenMPI. For the auxiliary loss, we set the weight to 0.4 in experiments.

PSPNet

3. Results : Ablation Experiments

Method	Mean IoU(%)	Pixel Acc.(%)
ResNet50-Baseline	37.23	78.01
ResNet50+B1+MAX	39.94	79.46
ResNet50+B1+AVE	40.07	79.52
ResNet50+B1236+MAX	40.18	79.45
ResNet50+B1236+AVE	41.07	79.97
ResNet50+B1236+MAX+DR	40.87	79.61
ResNet50+B1236+AVE+DR	**41.68**	**80.04**

Table 1. Investigation of PSPNet with different settings. Baseline is ResNet50-based FCN with dilated network. 'B1' and 'B1236' denote pooled feature maps of bin sizes $\{1 \times 1\}$ and $\{1 \times 1, 2 \times 2, 3 \times 3, 6 \times 6\}$ respectively. 'MAX' and 'AVE' represent max pooling and average pooling operations individually. 'DR' means that dimension reduction is taken after pooling. The results are tested on the validation set with the single-scale input.

Loss Weight α	Mean IoU(%)	Pixel Acc.(%)
ResNet50 (without AL)	35.82	77.07
ResNet50 (with $\alpha = 0.3$)	37.01	77.87
ResNet50 (with $\alpha = 0.4$)	**37.23**	**78.01**
ResNet50 (with $\alpha = 0.6$)	37.09	77.84
ResNet50 (with $\alpha = 0.9$)	36.99	77.87

Table 2. Setting an appropriate loss weight α in the auxiliary branch is important. 'AL' denotes the auxiliary loss. Baseline is ResNet50-based FCN with dilated network. Empirically, $\alpha = 0.4$ yields the best performance. The results are tested on the validation set with the single-scale input.

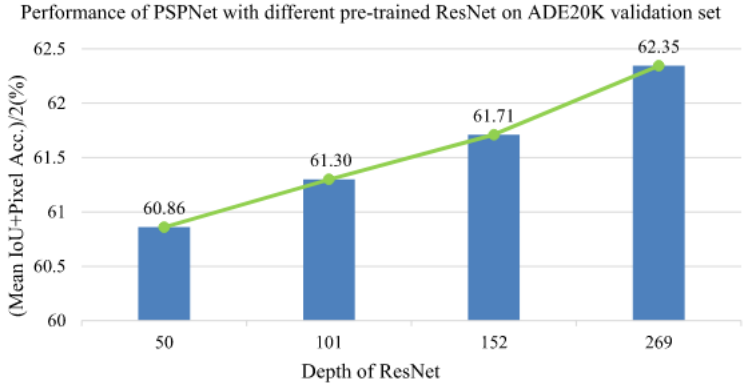


Figure 5. Performance grows with deeper networks. The results are obtained on the validation set with the single-scale input.

Method	Mean IoU(%)	Pixel Acc.(%)
PSPNet(50)	41.68	80.04
PSPNet(101)	41.96	80.64
PSPNet(152)	42.62	80.80
PSPNet(269)	**43.81**	**80.88**
PSPNet(50)+MS	42.78	80.76
PSPNet(101)+MS	43.29	81.39
PSPNet(152)+MS	43.51	81.38
PSPNet(269)+MS	**44.94**	**81.69**

Table 3. Deeper pre-trained model get higher performance. Number in the brackets refers to the depth of ResNet and 'MS' denotes multi-scale testing.

PSPNet

3. Results : Experiments

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
FCN [26]	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
Zoom-out [28]	85.6	37.3	83.2	62.5	66.0	85.1	80.7	84.9	27.2	73.2	57.5	78.1	79.2	81.1	77.1	53.6	74.0	49.2	71.7	63.3	69.6
DeepLab [3]	84.4	54.5	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1	59.8	79.0	76.1	83.2	80.8	59.7	82.2	50.4	73.1	63.7	71.6
CRF-RNN [41]	87.5	39.0	79.7	64.2	68.3	87.6	80.8	84.4	30.4	78.2	60.4	80.5	77.8	83.1	80.6	59.5	82.8	47.8	78.3	67.1	72.0
DeconvNet [30]	89.9	39.3	79.7	63.9	68.2	87.4	81.2	86.1	28.5	77.0	62.0	79.0	80.3	83.6	80.2	58.8	83.4	54.3	80.7	65.0	72.5
GCRF [36]	85.2	43.9	83.3	65.2	68.3	89.0	82.7	85.3	31.1	79.5	63.3	80.5	79.3	85.5	81.0	60.5	85.5	52.0	77.3	65.1	73.2
DPN [25]	87.7	59.4	78.4	64.9	70.3	89.3	83.5	86.1	31.7	79.9	62.6	81.9	80.0	83.5	82.3	60.5	83.2	53.4	77.9	65.0	74.1
Piecewise [20]	90.6	37.6	80.0	67.8	74.4	92.0	85.2	86.2	39.1	81.2	58.9	83.8	83.9	84.3	84.8	62.1	83.2	58.2	80.8	72.3	75.3
PSPNet	**91.8**	**71.9**	**94.7**	**71.2**	**75.8**	**95.2**	**89.9**	**95.9**	**39.3**	**90.7**	**71.7**	**90.5**	**94.5**	**88.8**	**89.6**	**72.8**	**89.6**	**64.0**	**85.1**	**76.3**	**82.6**
CRF-RNN† [41]	90.4	55.3	88.7	68.4	69.8	88.3	82.4	85.1	32.6	78.5	64.4	79.6	81.9	86.4	81.8	58.6	82.4	53.5	77.4	70.1	74.7
BoxSup† [7]	89.8	38.0	89.2	68.9	68.0	89.6	83.0	87.7	34.4	83.6	67.1	81.5	83.7	85.2	83.5	58.6	84.9	55.8	81.2	70.7	75.2
Dilation8† [40]	91.7	39.6	87.8	63.1	71.8	89.7	82.9	89.8	37.2	84.0	63.0	83.3	89.0	83.8	85.1	56.8	87.6	56.0	80.2	64.7	75.3
DPN† [25]	89.0	61.6	87.7	66.8	74.7	91.2	84.3	87.6	36.5	86.3	66.1	84.4	87.8	85.6	85.4	63.6	87.3	61.3	79.4	66.4	77.5
Piecewise† [20]	94.1	40.7	84.1	67.8	75.9	93.4	84.3	88.4	42.5	86.4	64.7	85.4	89.0	85.8	86.0	67.5	90.2	63.8	80.9	73.0	78.0
FCRNs† [38]	91.9	48.1	93.4	69.3	75.5	94.2	87.5	92.8	36.7	86.9	65.2	89.1	90.2	86.5	87.2	64.6	90.1	59.7	85.5	72.7	79.1
LRR† [9]	92.4	45.1	94.6	65.2	75.8	**95.1**	89.1	92.3	39.0	85.7	70.4	88.6	89.4	88.6	86.6	65.8	86.2	57.4	85.7	77.3	79.3
DeepLab† [4]	92.6	60.4	91.6	63.4	76.3	95.0	88.4	92.6	32.7	88.5	67.6	89.6	92.1	87.0	87.4	63.3	88.3	60.0	86.8	74.5	79.7
PSPNet†	**95.8**	**72.7**	**95.0**	**78.9**	**84.4**	94.7	**92.0**	**95.7**	**43.1**	**91.0**	**80.3**	**91.3**	**96.3**	**92.3**	**90.1**	**71.5**	**94.4**	**66.9**	**88.8**	**82.0**	**85.4**

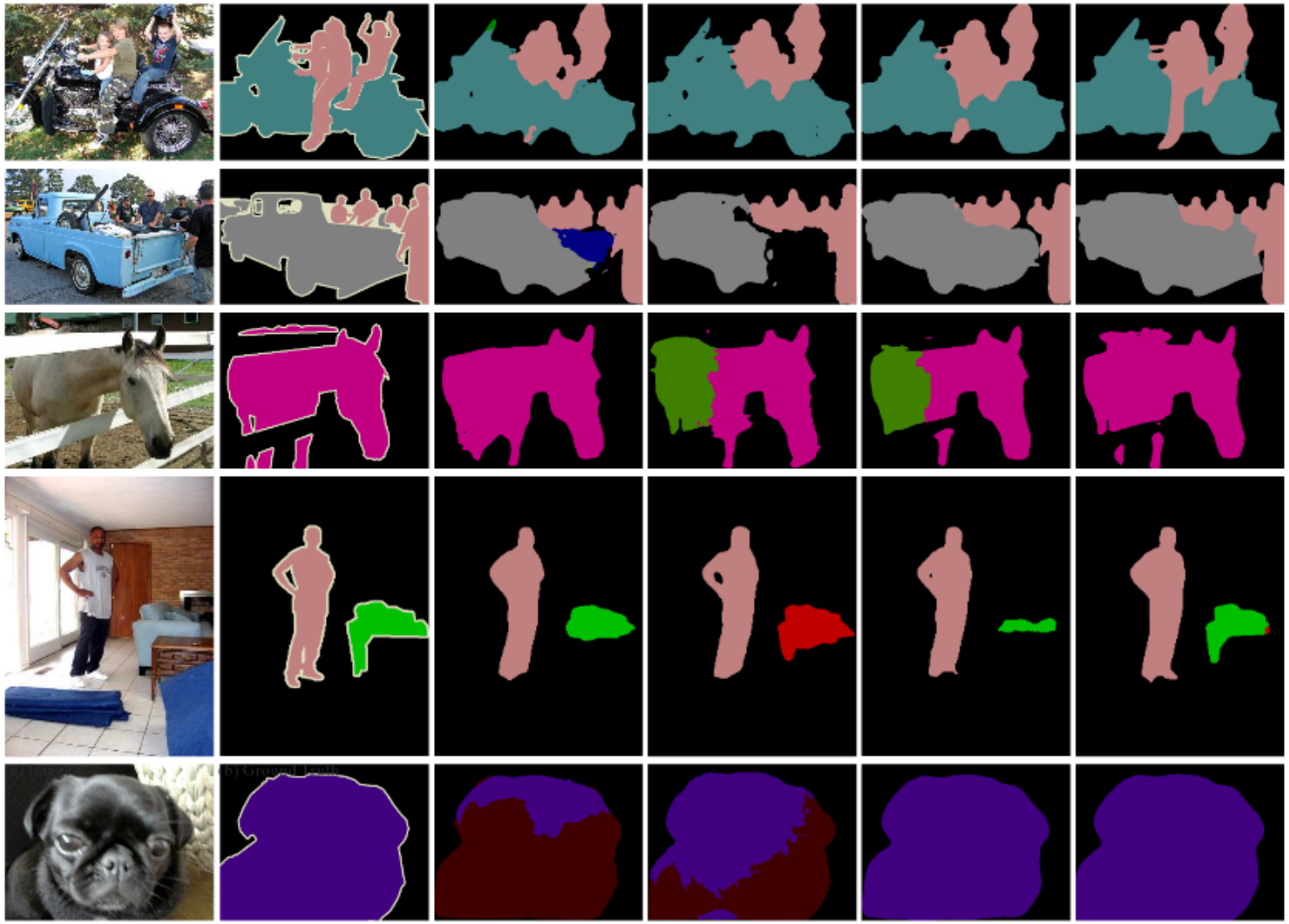
Method	IoU cla.	iIoU cla.	IoU cat.	iIoU cat.
CRF-RNN [41]	62.5	34.4	82.7	66.0
FCN [26]	65.3	41.7	85.7	70.1
SiCNN [16]	66.3	44.9	85.0	71.2
DPN [25]	66.8	39.1	86.0	69.1
Dilation10 [40]	67.1	42.0	86.5	71.1
LRR [9]	69.7	48.0	88.2	74.7
DeepLab [4]	70.4	42.6	86.4	67.7
Piecewise [20]	71.6	51.7	87.3	74.1
PSPNet	**78.4**	**56.7**	**90.6**	**78.6**
LRR‡ [9]	71.8	47.9	88.4	73.9
PSPNet‡	**80.2**	**58.1**	**90.6**	**78.2**

Method	Mean IoU(%)	Pixel Acc.(%)
FCN [26]	29.39	71.32
SegNet [2]	21.64	71.00
DilatedNet [40]	32.31	73.55
CascadeNet [43]	34.90	74.52
ResNet50-Baseline	34.28	76.35
ResNet50+DA	35.82	77.07
ResNet50+DA+AL	37.23	78.01
ResNet50+DA+AL+PSP	**41.68**	**80.04**
ResNet269+DA+AL+PSP	43.81	80.88
ResNet269+DA+AL+PSP+MS	**44.94**	**81.69**

Table 4. Detailed analysis of our proposed PSPNet with comparison with others. Our results are obtained on the validation set with the single-scale input except for the last row. Results of FCN, SegNet and DilatedNet are reported in [43]. 'DA' refers to data augmentation we performed, 'AL' denotes the auxiliary loss we added and 'PSP' represents the proposed PSPNet. 'MS' means that multi-scale testing is used.

PSPNet

3. Results : Experiments



(a) Image (b) Ground Truth (c) FCN (d) DPN (e) DeepLab (f) PSPNet

PSPNet

3. Results : Experiments

(a) Image (b) Ground Truth (c) PSPNet

Thank you!

PSPNet