Computer Vision; Image Transformation; Semantic Segmentation

Maziar Raissi

Assistant Professor

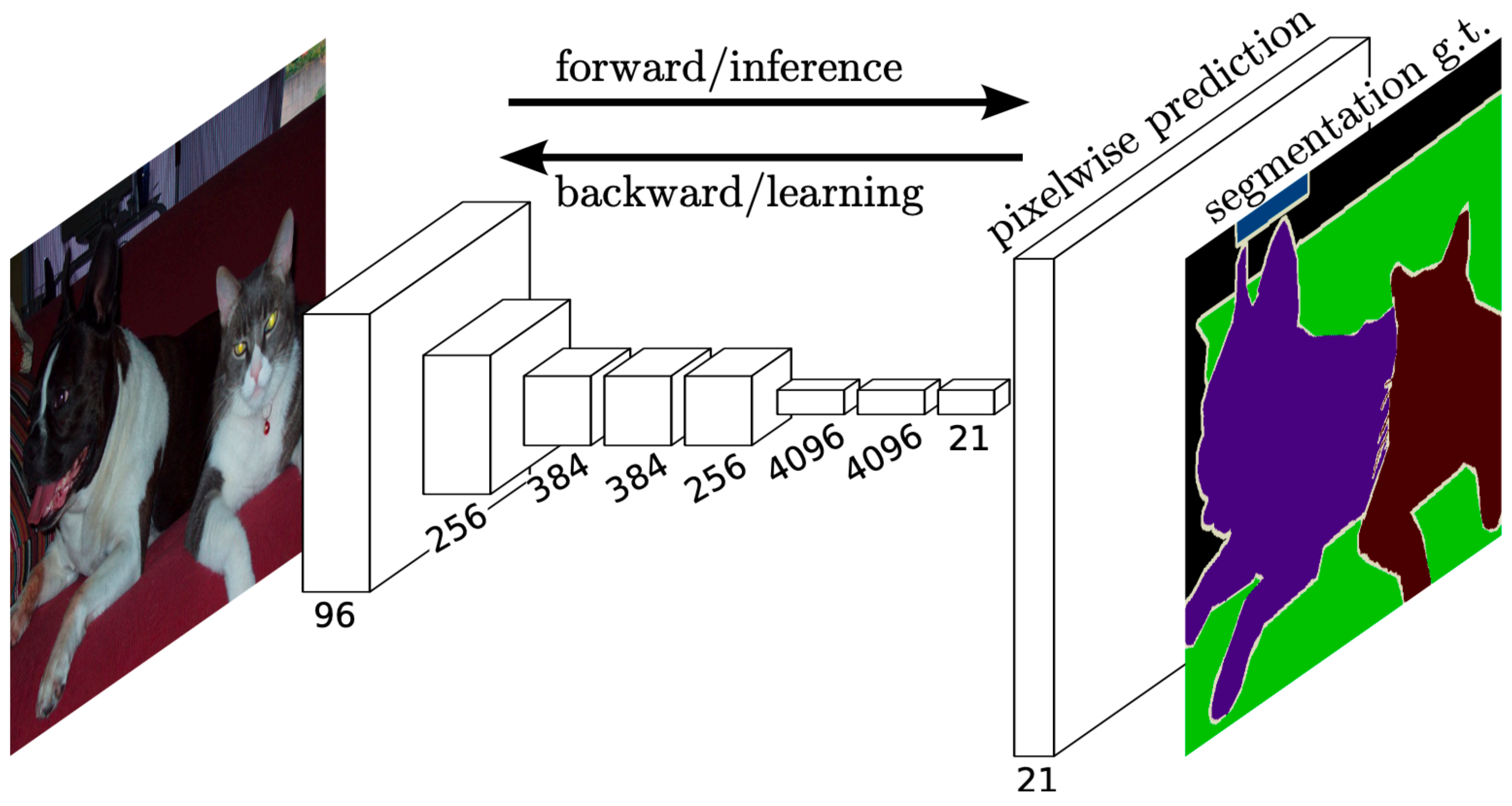
Department of Applied Mathematics

University of Colorado Boulder

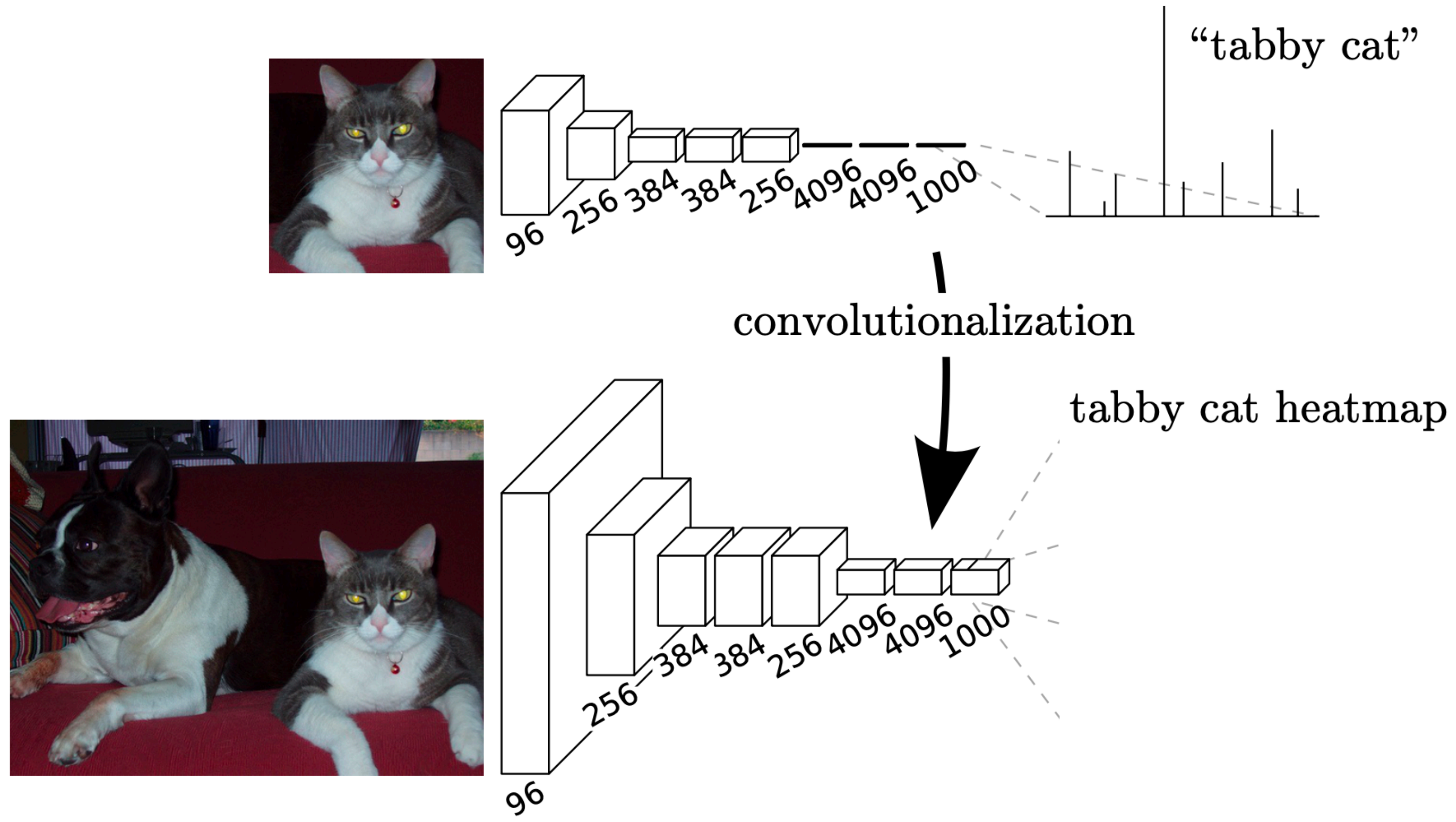
maziar.raissi@colorado.edu

Fully Convolutional Networks for Semantic Segmentation

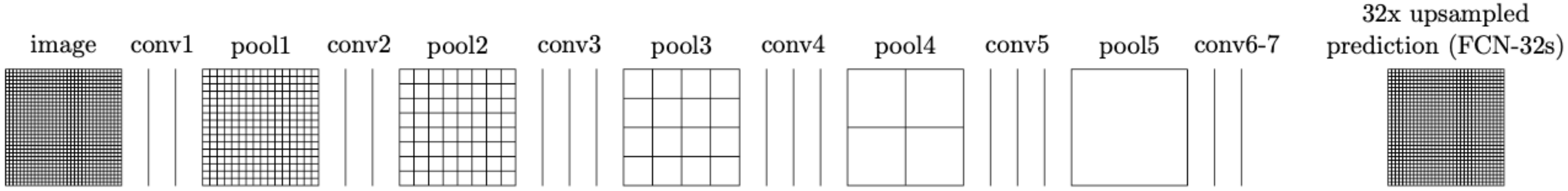
[YouTube Playlist](#)



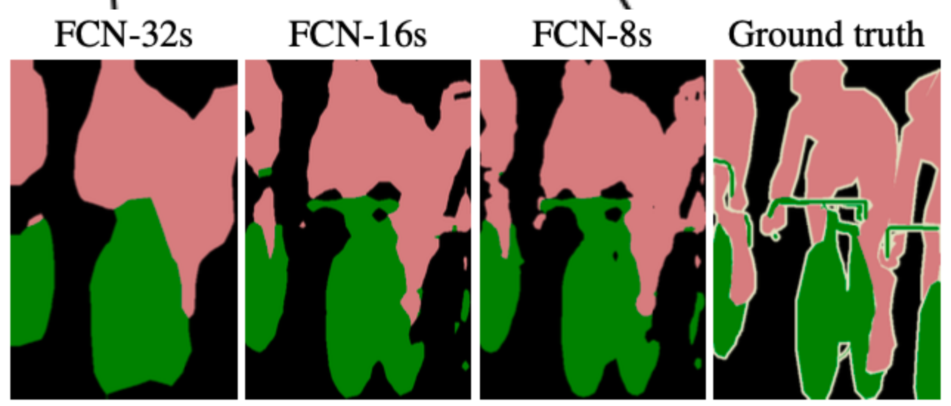
global information resolves **what**
local information resolves **where**

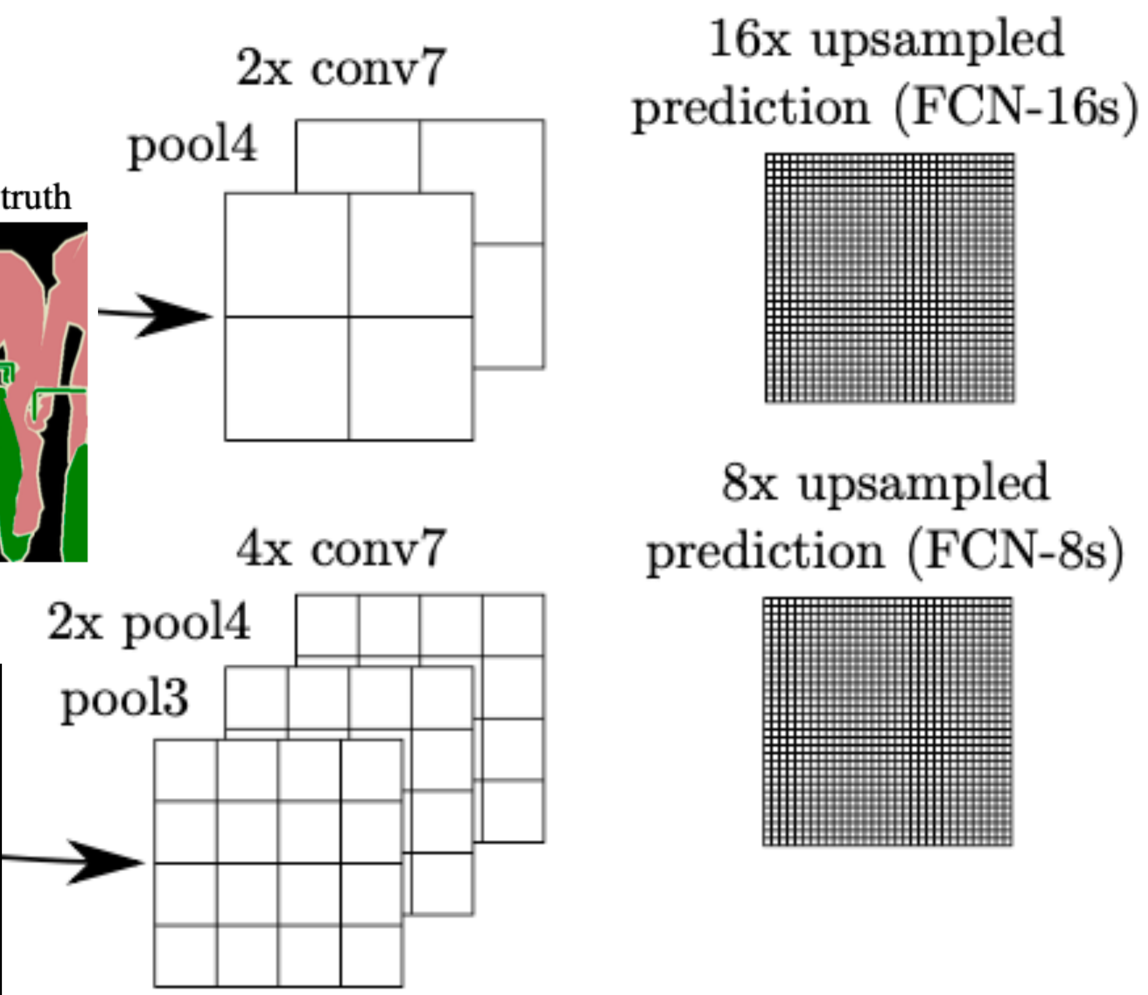



```
The fully connected layers can also be
viewed as convolutions with kernels that
cover their entire input regions.
```



Upsampling with factor f is convolution with a fractional input stride of $1/f$.
Backwards convolution (deconvolution) with an output stride of f
Reverse the forward and backward passes of convolution





Evaluation Metrics

- **pixel accuracy:** $\sum_i n_{ii} / \sum_i t_i$
- **mean accuraccy:** $(1/n_{\text{cl}}) \sum_i n_{ii}/t_i$
- **mean IU:** $(1/n_{\text{cl}}) \sum_i n_{ii} / \left(t_i + \sum_j n_{ji} - n_{ii}\right)$
- **frequency weighted IU:** $\left(\sum_k t_k\right)^{-1} \sum_i t_i n_{ii} / \left(t_i + \sum_j n_{ji} - n_{ii}\right)$

$n_{ij} \rightarrow$ number of pixels of class i predicted to belong to class j

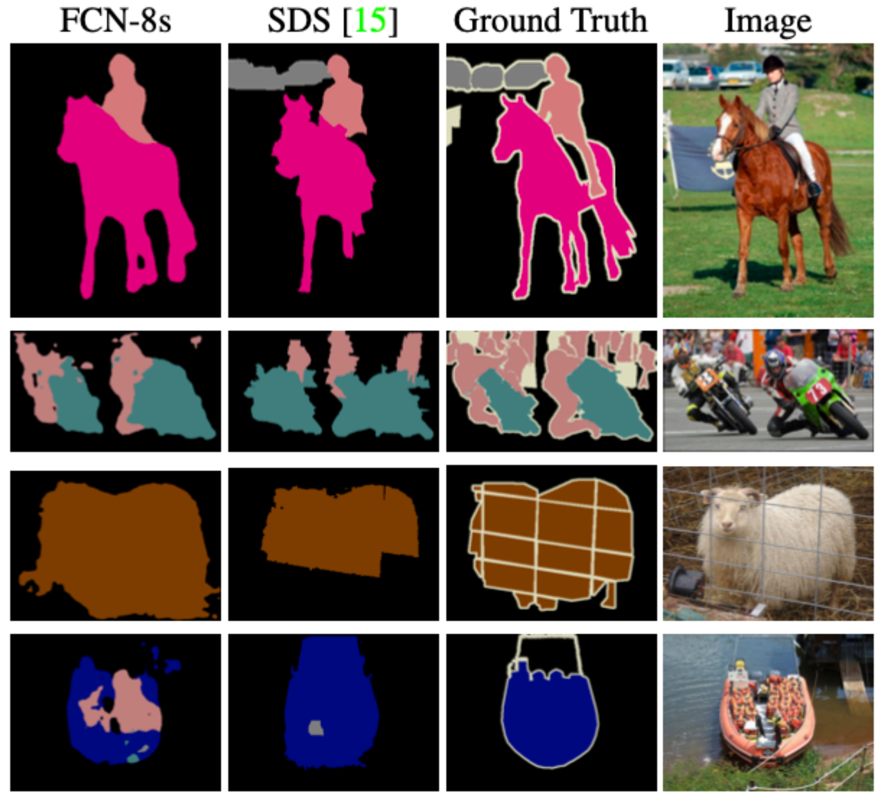
$n_{\text{cl}} \rightarrow$ number of different classes

$t_i = \sum_j n_{ij} \rightarrow$ total number of pixels of class i

	pixel acc.	mean acc.	mean IU	f.w. IU
FCN-32s-fixed	83.0	59.7	45.4	72.0
FCN-32s	89.1	73.3	59.4	81.4
FCN-16s	90.0	75.7	62.4	83.0
FCN-8s	**90.3**	**75.9**	**62.7**	**83.2**

	FCN-AlexNet	FCN-VGG16	FCN-GoogLeNet[4]
mean IU	39.8	**56.0**	42.5
forward time	50 ms	210 ms	59 ms
conv. layers	8	16	22
parameters	57M	134M	6M
rf size	355	404	907
max stride	32	32	32

	mean IU VOC2011 test	mean IU VOC2012 test	inference time
R-CNN [10]	47.9	-	-
SDS [15]	52.6	51.6	$\sim$ 50 s
FCN-8s	**62.7**	**62.2**	$\sim$ **175 ms**



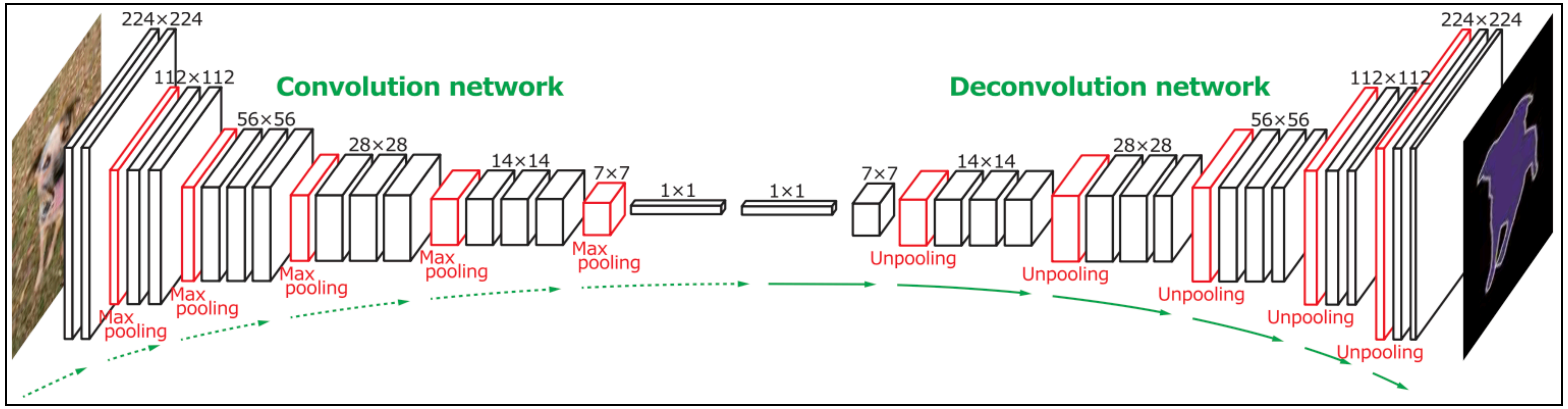
Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Learning Deconvolution Network for Semantic Segmentation

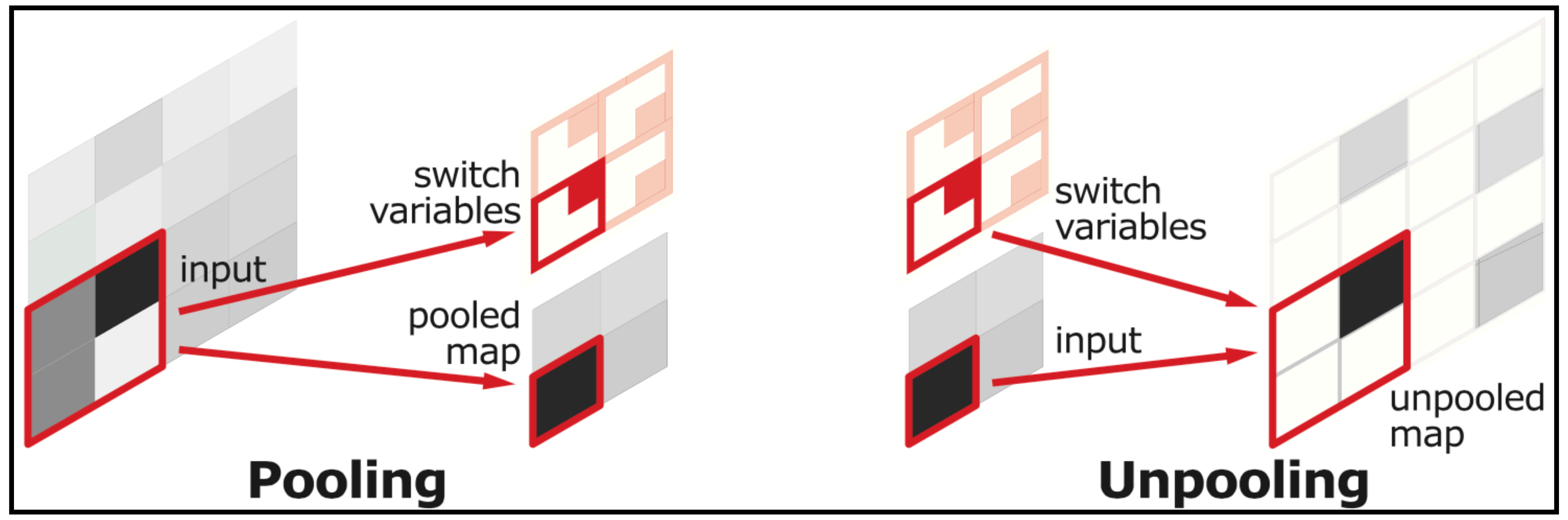
Pre-defined fixed-size receptive field!

(a) Inconsistent labels due to large object size

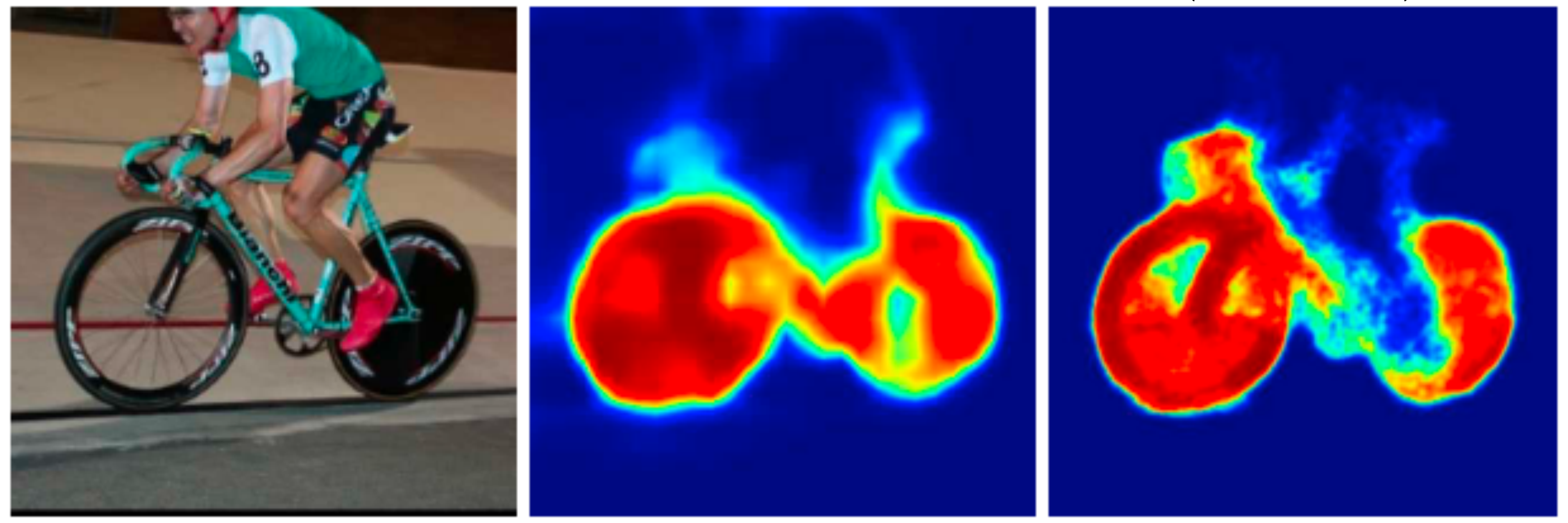
(b) Missing labels due to small object size



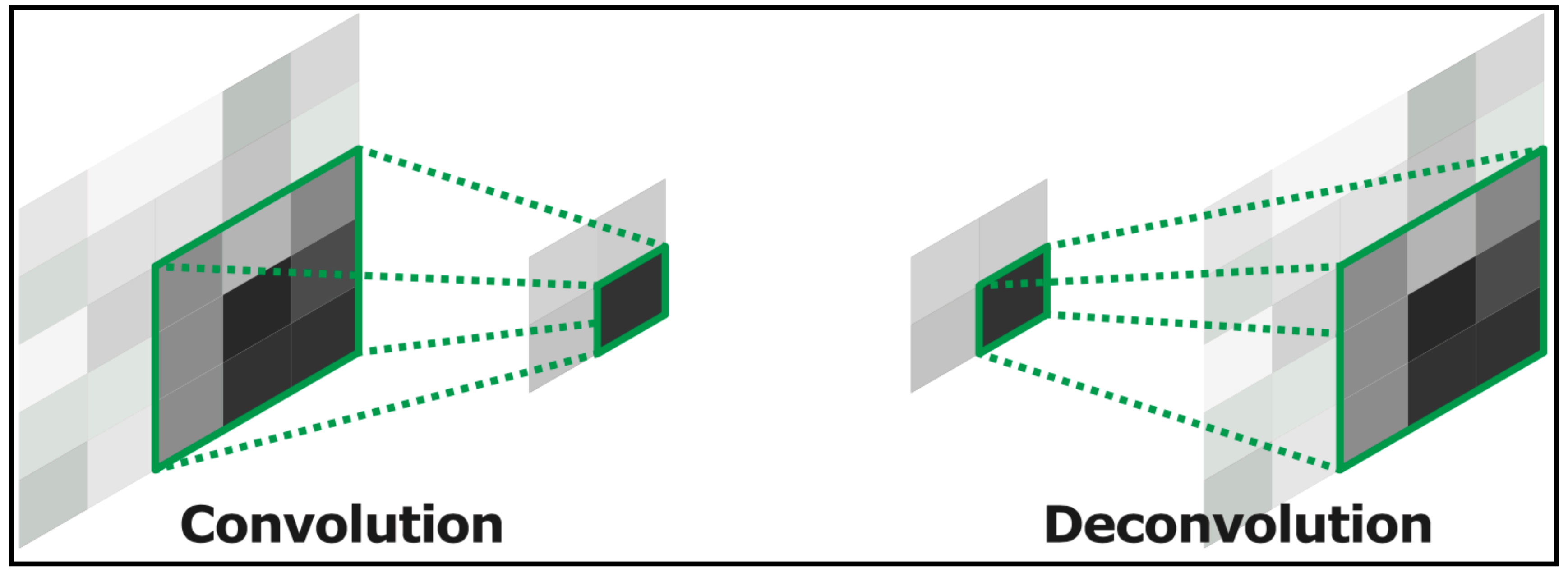
The detailed structures of an object are often lost or smoothed because the label map, input to the deconvolutional layer, is too coarse and deconvolution procedure is overly simple.



Class conditional probability map (bicycle)



(a) Input image (b) FCN-8s (c) Ours



Instance-wise prediction!

$g_i \in \mathbb{R}^{W \times H \times C} \rightarrow$ output score maps of the i-th proposal $\boxed{\text{edge-box}}$

$G_i \rightarrow$ zero padded outside g_i

$$P(x, y, c) = \max_i G_i(x, y, c)$$

$$P(x, y, c) = \sum_i G_i(x, y, c)$$

pixel-wise class score map (before softmax)

Batch Normalization

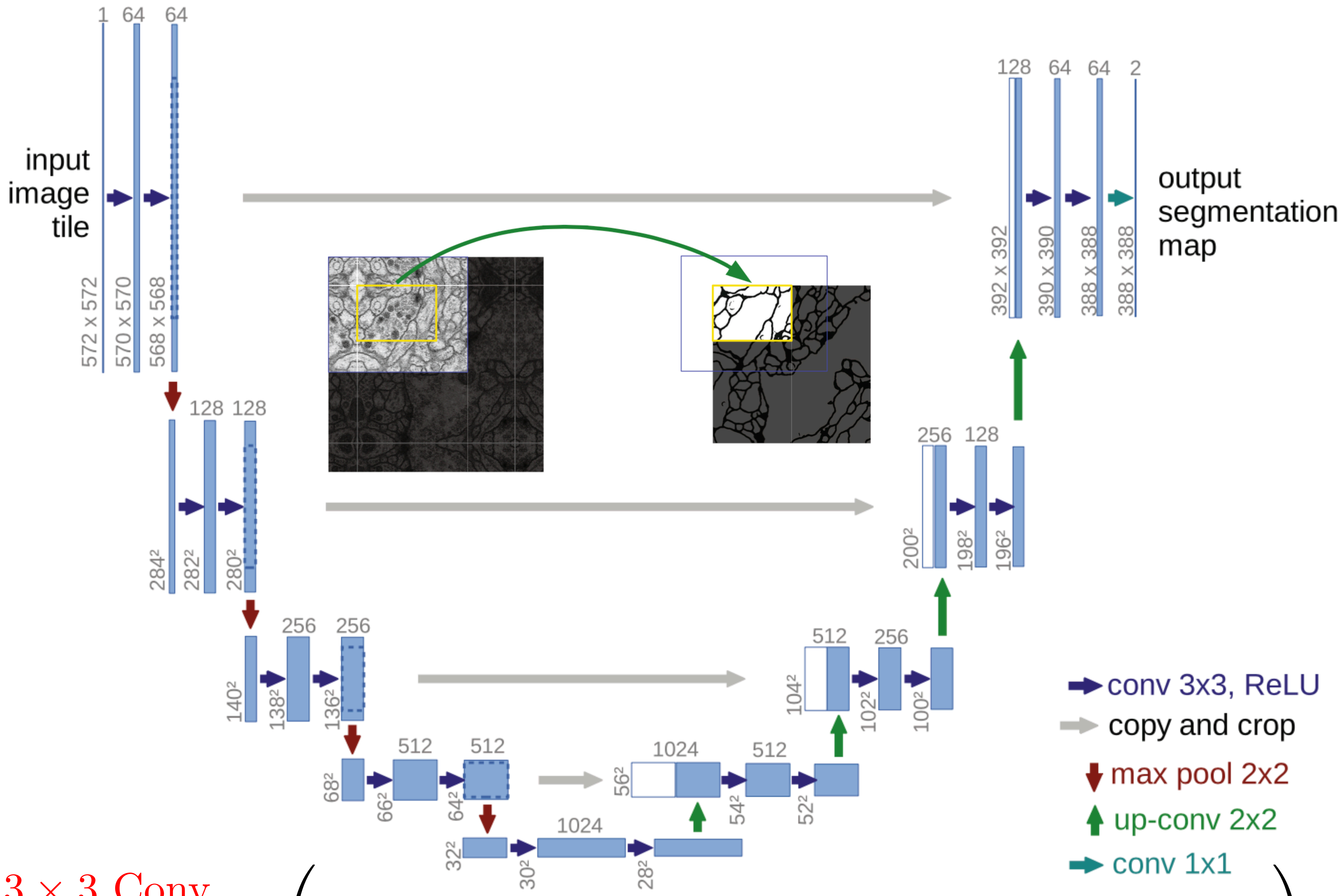
Two-stage Training:

1) ground-truth bounding boxes and 2) object proposals (≥ 0.5 in IoU)

Noh, Hyeonwoo, Seunghoon Hong, and Bohyung Han. "Learning deconvolution network for semantic segmentation." *Proceedings of the IEEE international conference on computer vision*. 2015.

U-Net: Convolutional Networks for Biomedical Image Segmentation

[YouTube Video](#)



3 × 3 Conv

$$b_{x,y,\ell} = \mathrm{ReLU}\left(\sum_{i\in\{-1,0,1\}}\sum_{j\in\{-1,0,1\}}\sum_{k\in\{1,\dots,K\}} w_{i,j,k,\ell}\, a_{x+i,y+j,k} + c_\ell\right)$$

2 × 2 maxpooling

$$b_{x,y,k} = \max_{i,j\in\{0,1\}} a_{2x+i,2y+j,k} \rightarrow \text{stride} = 2$$

2 × 2 up-conv

$$b_{2x+i,2y+j,\ell} = \mathrm{ReLU}\left(\sum_{k\in\{1,\dots,K\}} w_{i,j,k,\ell}\, a_{x,y,k} + c_\ell\right) \quad \text{for } i,j\in\{0,1\}$$

$$\mathcal{L} = -\sum_{(x,y)\in\Omega} w(x,y)\log p_{\ell(x,y)}(x,y)$$

$$\ell:\ \Omega \rightarrow \{1,\dots,K\}$$

$$(x,y) \mapsto \ell(x,y)$$

true label of each pixel

$$p_k(x,y) = \exp(a_k(x,y)) \Big/ \left(\sum_{k'=1}^{K}\exp(a_{k'}(x,y))\right)$$

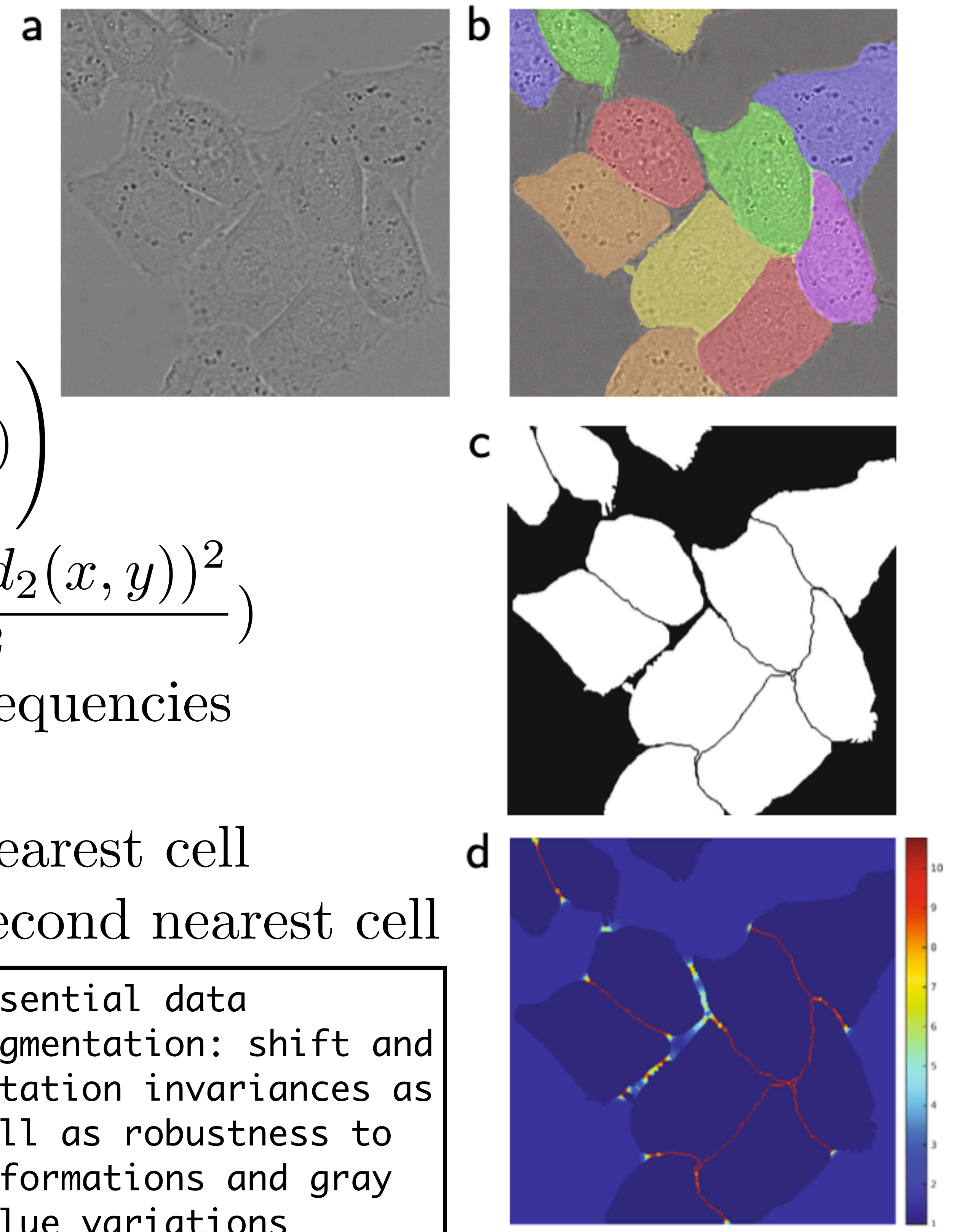
$$w(x,y) = w_c(x,y) + w_0\exp\left(-\frac{(d_1(x,y)+d_2(x,y))^2}{2\sigma^2}\right)$$

$w_c(x,y) \rightarrow$ weight map to balance class frequencies

$w_0 = 10$ & $\sigma \approx 5$ pixels

$d_1(x,y) \rightarrow$ distance to the border of the nearest cell

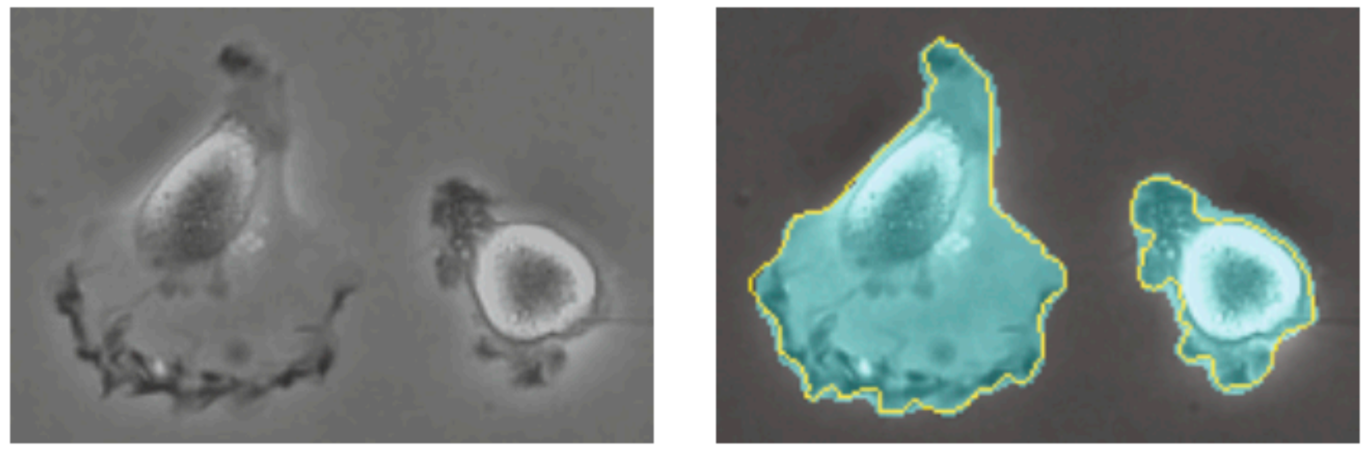
$d_2(x,y) \rightarrow$ distance to the border of the second nearest cell

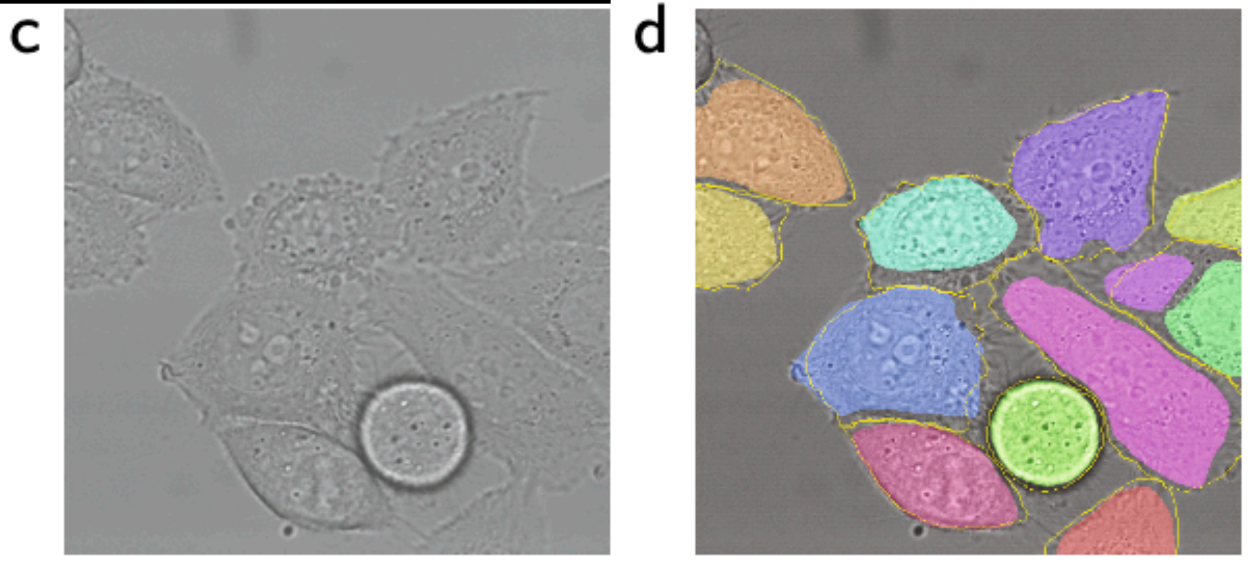


Name	PhC-U373	DIC-HeLa
IMCB-SG (2014)	0.2669	0.2935
KTH-SE (2014)	0.7953	0.4607
HOUS-US (2014)	0.5323	-
second-best 2015	0.83	0.46
u-net (2015)	**0.9203**	**0.7756**

Essential data augmentation: shift and rotation invariances as well as robustness to deformations and gray value variations

a





Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs

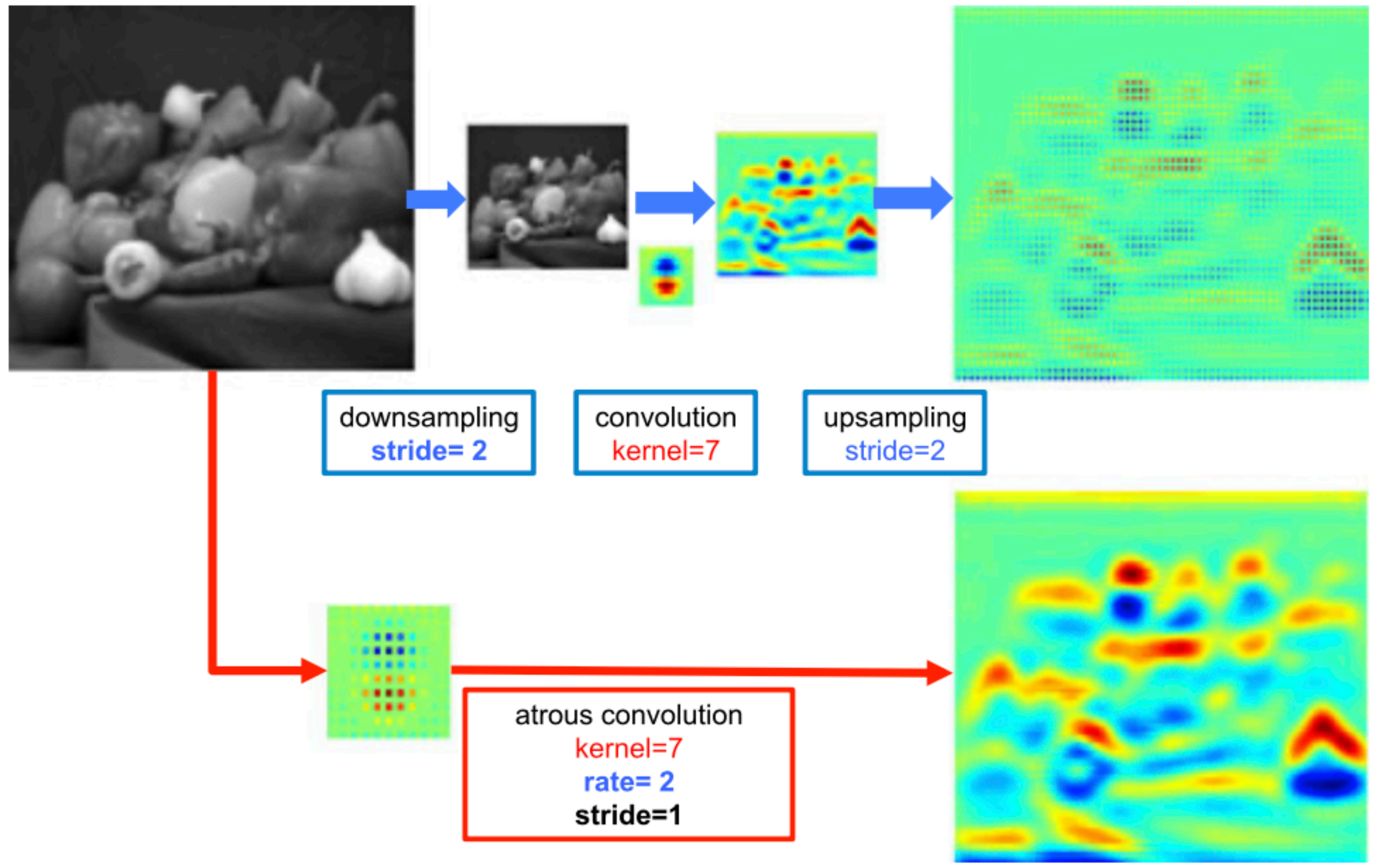
Boulder

YouTube Playlist

Three challenges in the application of DCNNs to semantic image segmentation: (1) reduced feature resolution, (2) existence of objects at multiple scales, and (3) reduced localization accuracy due to DCNN invariance.

Atrous Convolution

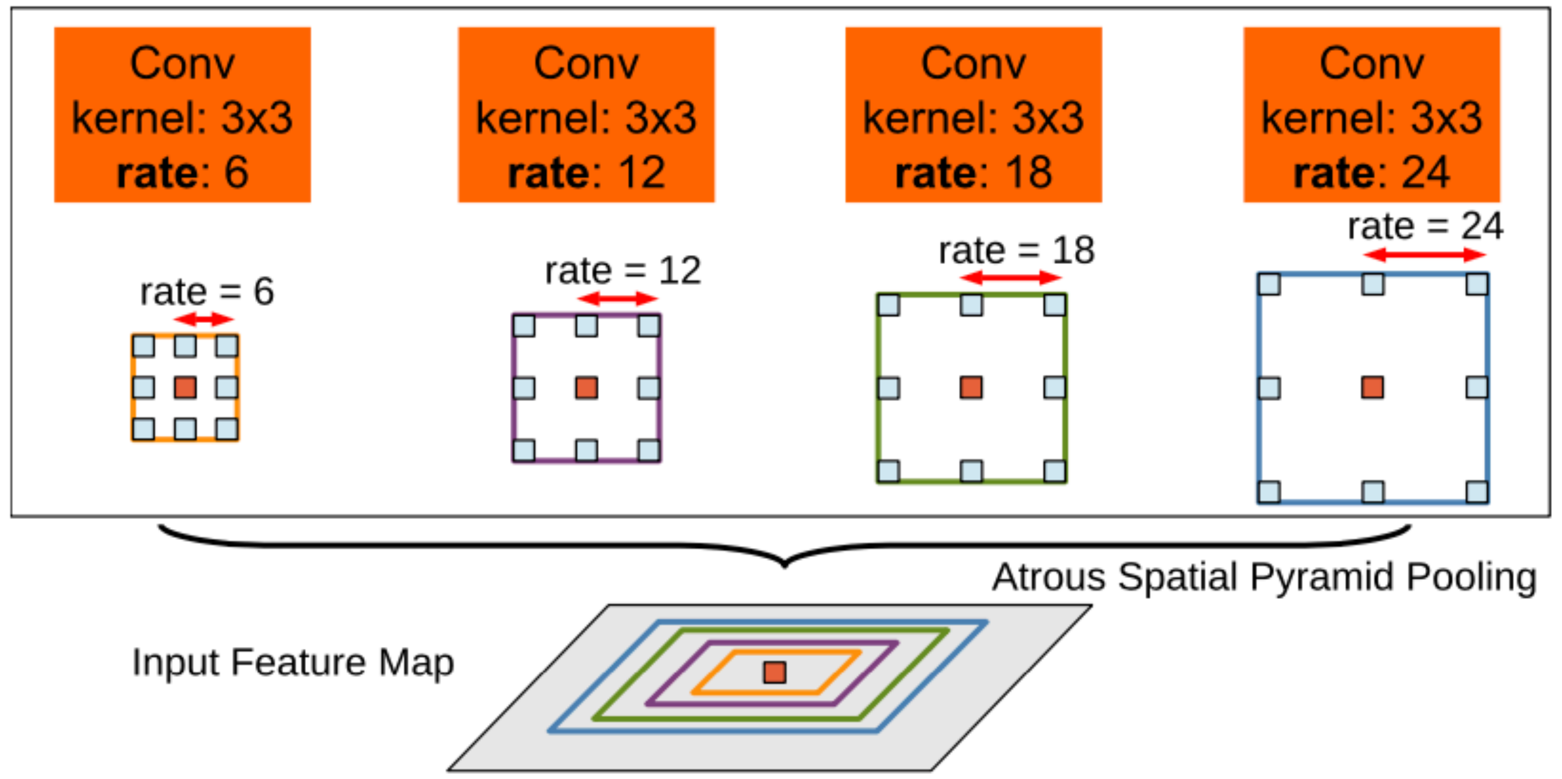
Reduce the degree of signal downsampling due to max-pooling and striding (from 32x down to 8x).



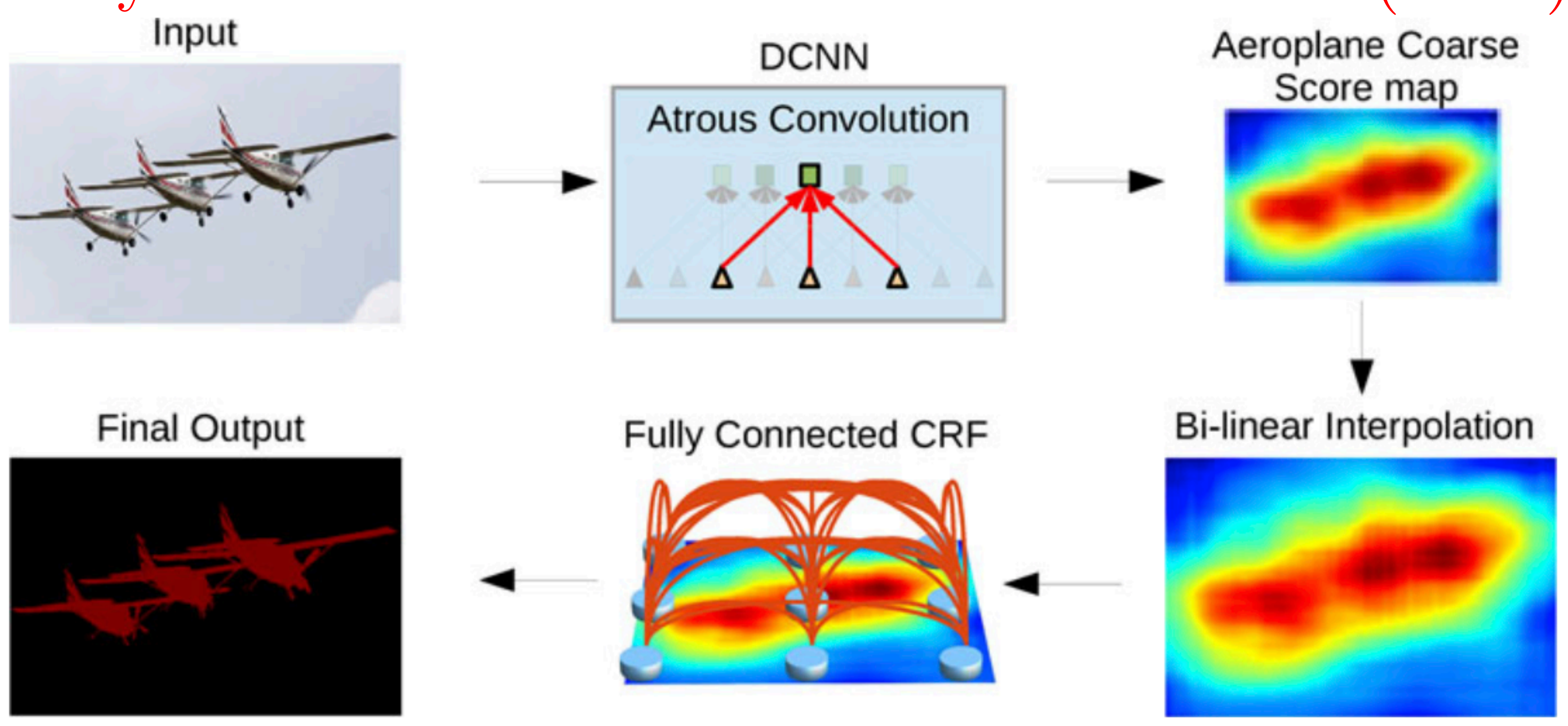
$$y[i] = \sum_{k=1}^{K} x[i + \underbrace{r}_{\text{rate parameter}} \cdot k] w[k]$$

Same number of parameters and amount of computation

Atrous Spatial Pyramid Pooling (ASPP)



Fully-Connected Conditional Random Fields (CRF)

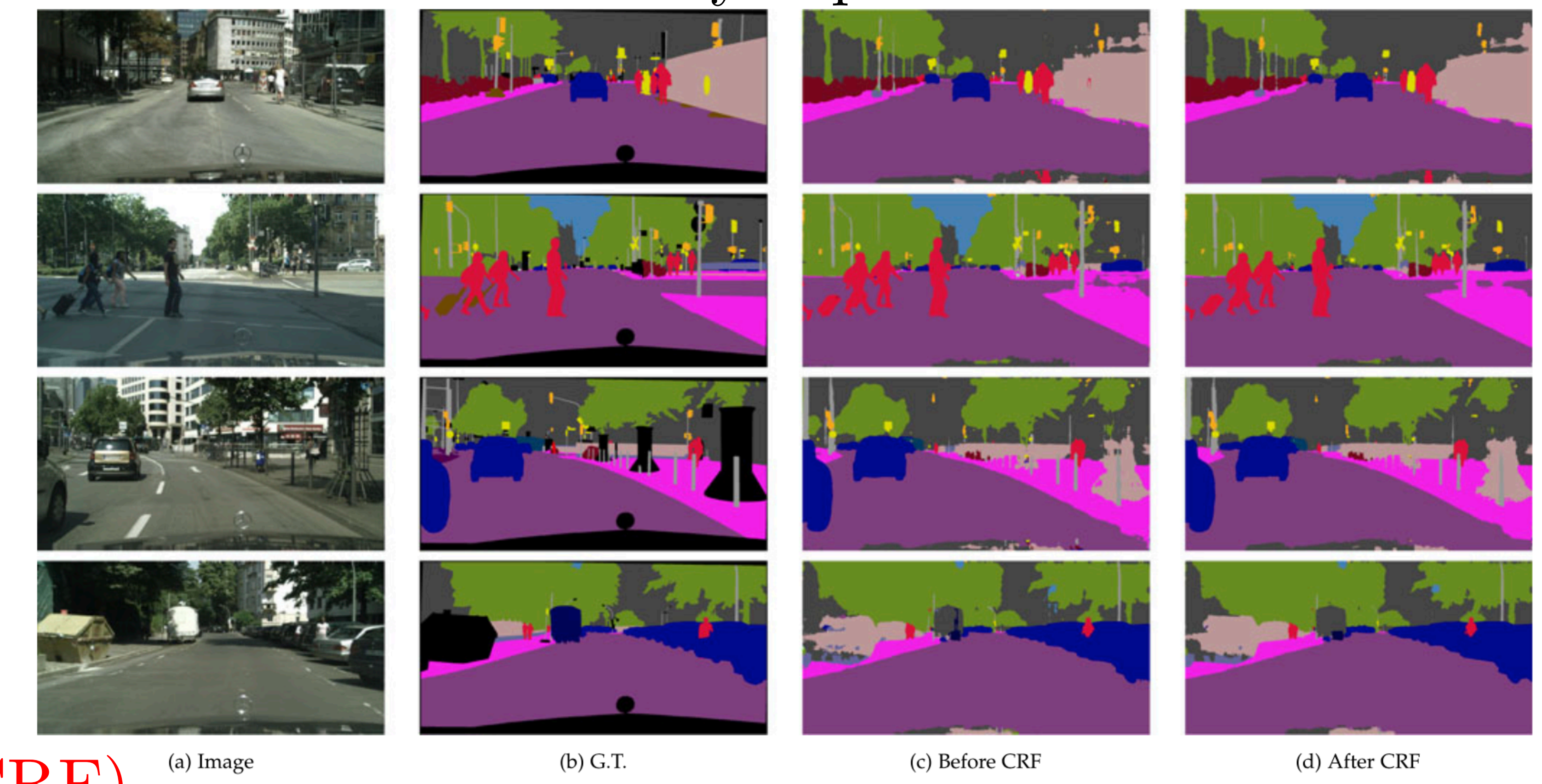


$\mathbf{x} \rightarrow$ label assignment for pixels

$$\underbrace{E(\boldsymbol{x})}_{\text{energy function}} = \sum_i \theta_i(x_i) + \sum_{ij} \theta_{ij}(x_i, x_j) \qquad \theta_i(x_i) = -\log P(x_i)$$

$$\theta_{ij}(x_i, x_j) = \mu(x_i, x_j)\left[w_1 \exp\left(-\frac{||p_i - p_j||^2}{2\sigma_\alpha^2} - \frac{||I_i - I_j||^2}{2\sigma_\beta^2}\right) + w_2 \exp\left(-\frac{||p_i - p_j||^2}{2\sigma_\gamma^2}\right)\right]$$

Cityscapes



Method	before CRF	after CRF
LargeFOV	65.76	69.84
ASPP-S	66.98	69.73
ASPP-L	68.96	71.57

PASCAL VOC VGG-16

MSC	COCO	Aug	LargeFOV	ASPP	CRF	mIOU
						68.72
✓						71.27
✓	✓					73.28
✓	✓	✓				74.87
✓	✓	✓	✓			75.54
✓	✓	✓		✓		76.35
✓	✓	✓		✓	✓	77.69

PASCAL VOC ResNet-101

$P(x_i) \rightarrow$ label assignment prob. at pixel i (DCNN)

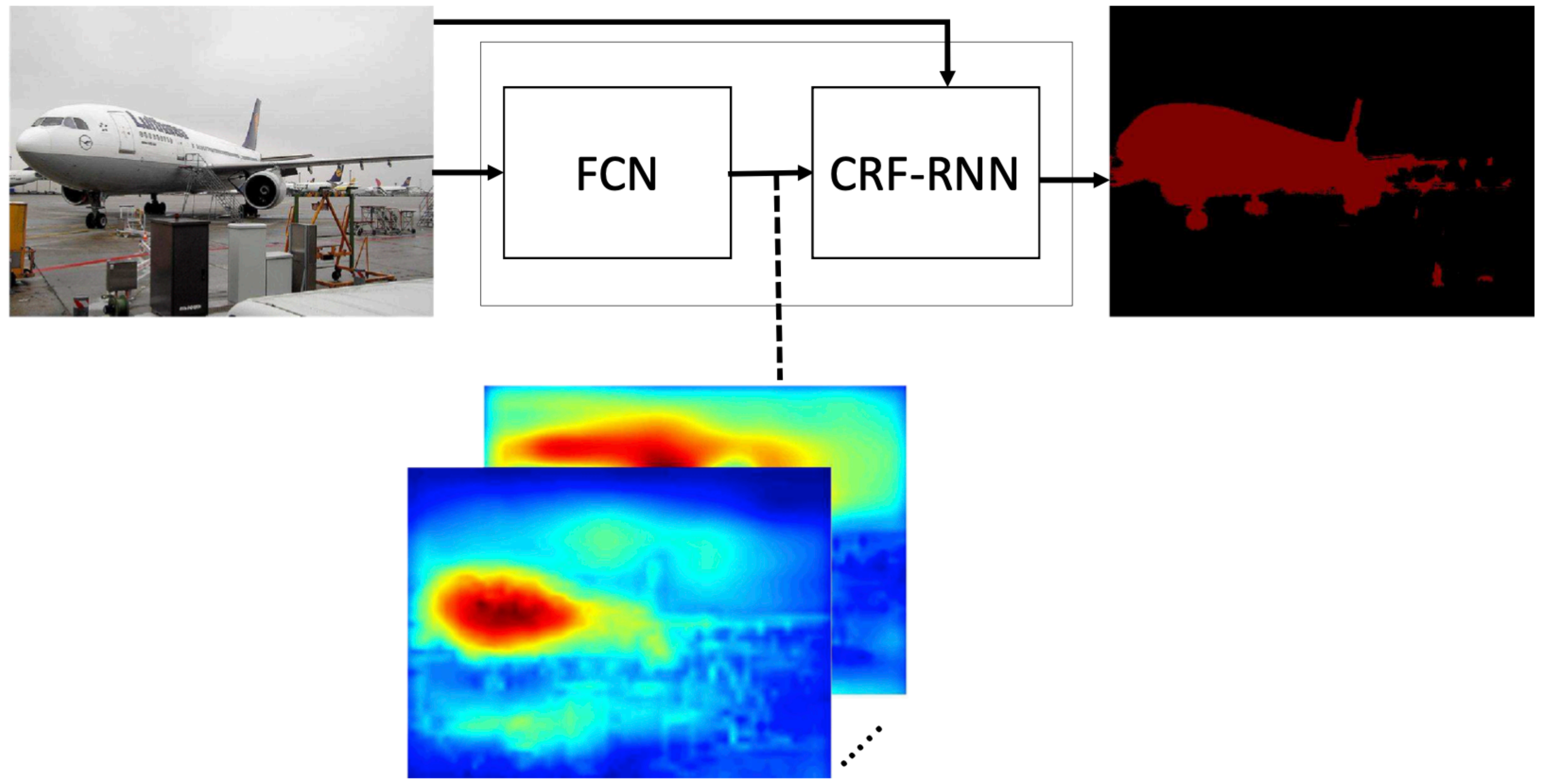
$p_i \rightarrow$ pixel position

$I_i \rightarrow$ RGB color

$\mu(x_i, x_j) = 1$ iff $x_i \neq x_j$

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.

Conditional Random Fields as Recurrent Neural Networks



Conditional Random Fields (CRFs)

Model pixel labels as random variables that form a Markov Random Field (MRF) when conditioned upon a global observation (i.e., image)

$X_i \rightarrow$ random variable associated to pixel i (represents the label assigned to pixel i)

$\mathcal{L} = \{l_1, l_2, \ldots, l_L\} \rightarrow$ set of labels

$X = (X_1, X_2, \ldots, X_N) \rightarrow$ vector

$N \rightarrow$ number of pixels in the image

$I \rightarrow$ global observation (image)

$$P(X = x|I) = \frac{1}{Z(I)} \exp(-E(x|I))$$

$Z(I) \rightarrow$ partition function

$E(I|x) \rightarrow$ energy of the configuration $x \in \mathcal{L}^N$

Drop conditioning on I for convenience!

Fully Connected Pairwise CRF

$$E(x) = \sum_i \psi_u(x_i) + \sum_{i<j} \psi_p(x_i, x_j)$$

$E(x) \rightarrow$ energy of a label assignment x

$\psi_u(x_i) \rightarrow$ unary energy components
inverse likelihood (i.e., cost) of pixel i taking label x_i (obtained from a CNN)

$\psi_p(x_i, x_j) \rightarrow$ pairwise energy component
cost of assigning labels x_i, x_j to pixels i, j, simultaneously.
encourage assigning similar labels to pixels with similar properties

$$\psi_p(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^{M} w^{(m)} k_G^{(m)}(\mathbf{f}_i, \mathbf{f}_j)$$

Gaussian kernel applied on feature vectors

$\mathbf{f}_i \rightarrow$ derived from image features such as spatial location and RGB values

$\mu \rightarrow$ label compatibility

$\arg\min_x E(x) \rightarrow$ most probable label assignment

Mean-field approximation to the CRF distribution

$$P(x) \approx Q(x) = \prod_i Q_i(x_i)$$

approximate maximum posterior marginal inference

CRF-RNN 5 iterations in training and 10 in testing!

$U_i(l) = -\psi_u(X_i = l)$

$\boldsymbol{\theta} = \left\{w^{(m)}, \mu(l, l')\right\} \rightarrow$ CRF parameters

Algorithm 1 Mean-field in dense CRFs [27], broken down to common CNN operations.

$Q_i(l) \leftarrow \frac{1}{Z_i} \exp(U_i(l))$ for all i softmax ▷ Initialization

while not converged **do**

$\tilde{Q}_i^{(m)}(l) \leftarrow \sum_{j \neq i} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l)$ for all m

permutohedral lattice implementation ($O(N)$ time) ▷ Message Passing

$\check{Q}_i(l) \leftarrow \sum_m w^{(m)} \tilde{Q}_i^{(m)}(l)$

1×1 conv ▷ Weighting Filter Outputs

$\hat{Q}_i(l) \leftarrow \sum_{l' \in \mathcal{L}} \mu(l, l') \check{Q}_i(l)$

1×1 conv ▷ Compatibility Transform

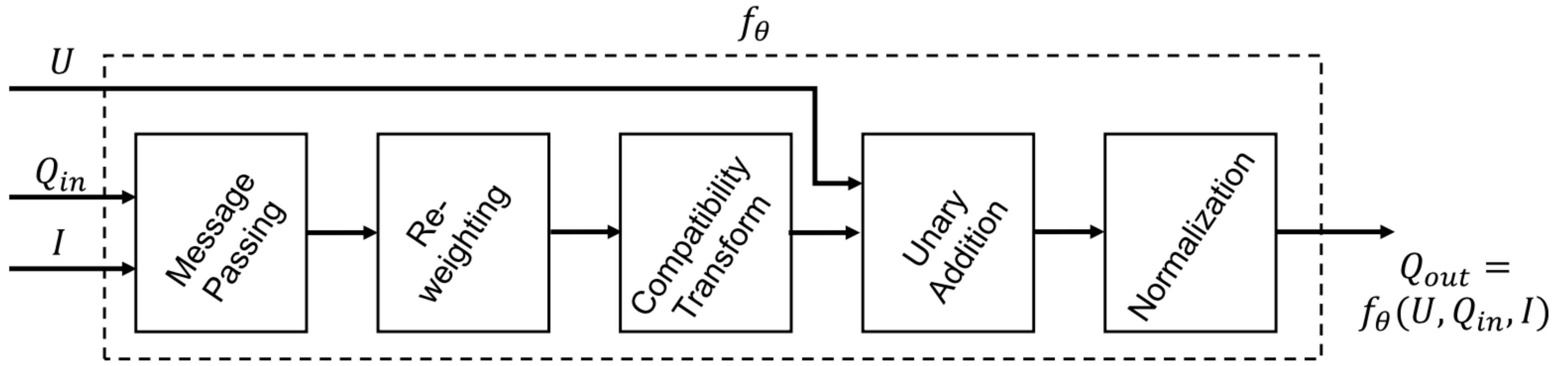
$\breve{Q}_i(l) \leftarrow U_i(l) - \hat{Q}_i(l)$

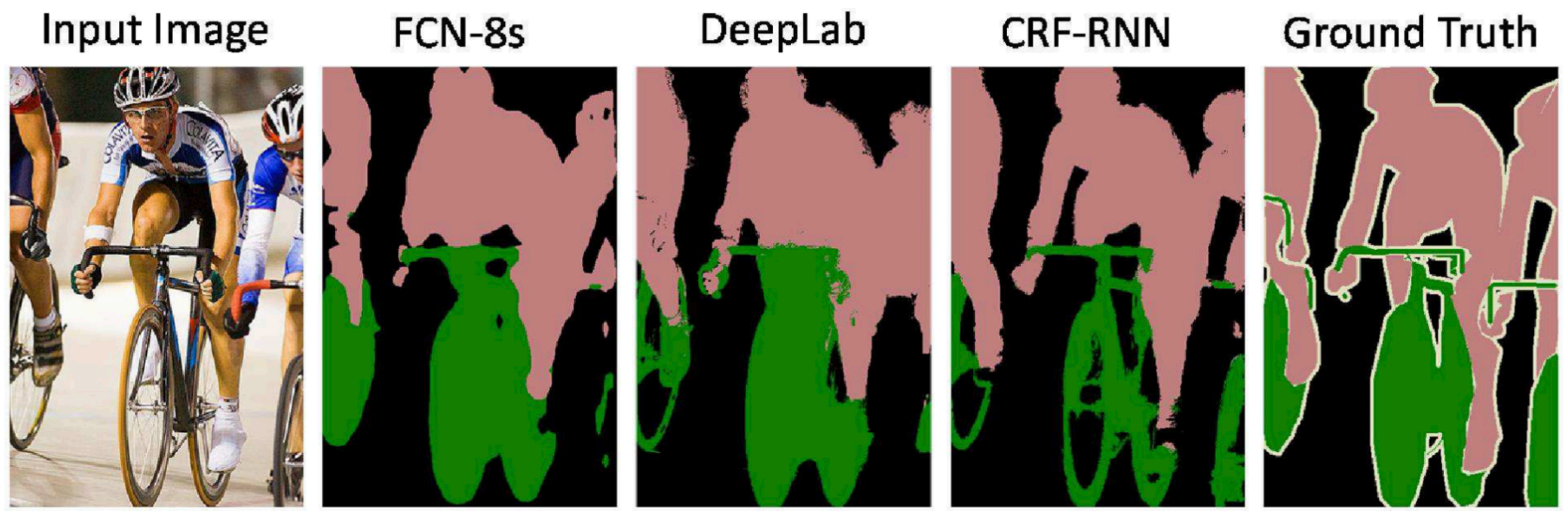
▷ Adding Unary Potentials

$Q_i \leftarrow \frac{1}{Z_i} \exp\left(\breve{Q}_i(l)\right)$

softmax ▷ Normalizing

end while





Zheng, Shuai, et al. "Conditional random fields as recurrent neural networks." *Proceedings of the IEEE international conference on computer vision*. 2015.

Boulder

Multi-scale Context Aggregation by Dilated Convolutions

YouTube Playlist

$F : \mathbb{Z}^2 \to \mathbb{R}$
discrete function

$\Omega_r = [-r, r]^2 \cap \mathbb{Z}^2$

$k : \Omega_r \to \mathbb{R}$
discrete filter of size $(2r+1)^2$

$$(F * k)(p) = \sum_{t \in \Omega_r} F(p-t)k(t)$$

discrete convolution operator

$$(F *_\ell k)(p) = \sum_{t \in \Omega_r} F(p-\ell t)k(t)$$

ℓ-dilated convolution

algorithme à trous
(an algorithm for
wavelet decomposition)
uses dilated convolutions

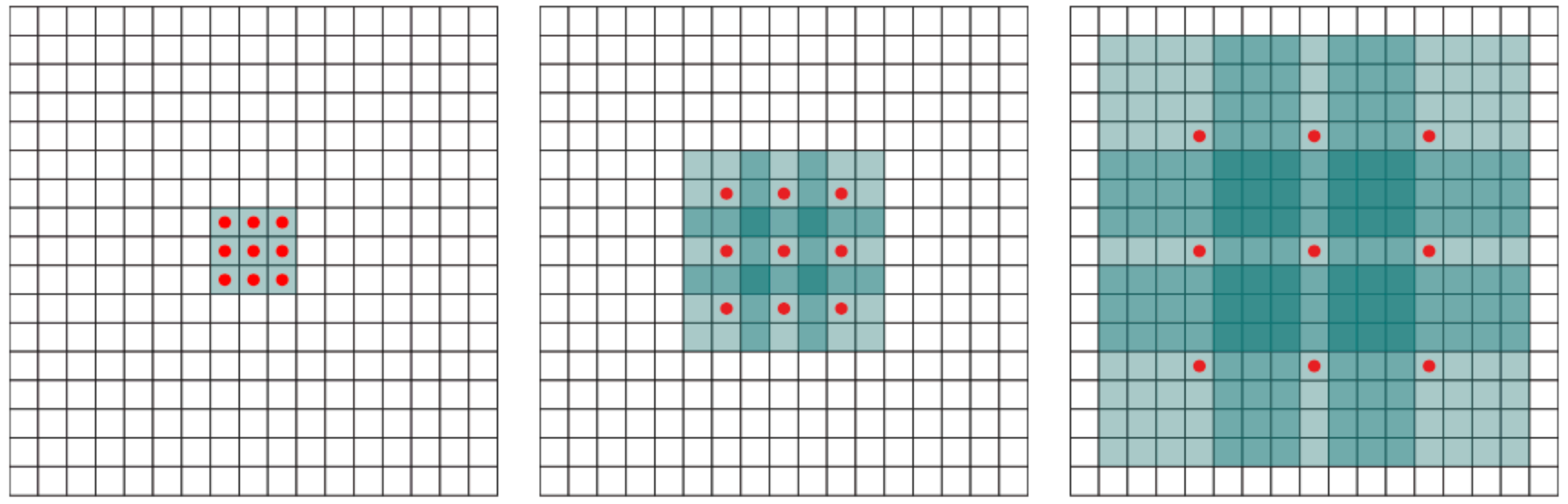
```
Dilated convolutions support
exponentially expanding
receptive fields without losing
resolution or coverage.
```

$$F_{i+1} = F_i *_{2^i} k_i \quad \text{for } i = 0, 1, \ldots, n-2.$$

3×3 filters

Receptive field of an element p in F_{i+1} is the set of elements in F_0 that modify the value of $F_{i+1}(p)$

Size of the receptive field of each element in F_{i+1} is $(2^{i+2}-1) \times (2^{i+2}-1)$.



(a) (b) (c)

Context network architecture

Layer	1	2	3	4	5	6	7	8
Convolution	3×3	3×3	3×3	3×3	3×3	3×3	3×3	1×1
Dilation	1	1	2	4	8	16	1	1
ReLU	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Receptive field	3×3	5×5	9×9	17×17	33×33	65×65	67×67	67×67
Output channels								
Basic	C	C	C	C	C	C	C	C
Large	$2C$	$2C$	$4C$	$8C$	$16C$	$32C$	$32C$	C

$k^b(t, a) = 1_{[t=0]} 1_{[a=b]}$ → identity initialization

a → index of the input feature map

b → index of the output feature map

$$k^b(\mathbf{t}, a) = \begin{cases} \dfrac{C}{c_{i+1}} & \mathbf{t} = 0 \text{ and } \left\lfloor \dfrac{aC}{c_i} \right\rfloor = \left\lfloor \dfrac{bC}{c_{i+1}} \right\rfloor \\ \varepsilon & \text{otherwise} \quad \varepsilon \sim \mathcal{N}(0, \sigma^2) \text{ and } \sigma \ll C/c_{i+1} \end{cases}$$

→ identity initialization (Large)

c_i and c_{i+1} → number of feature maps in two consecutive layers

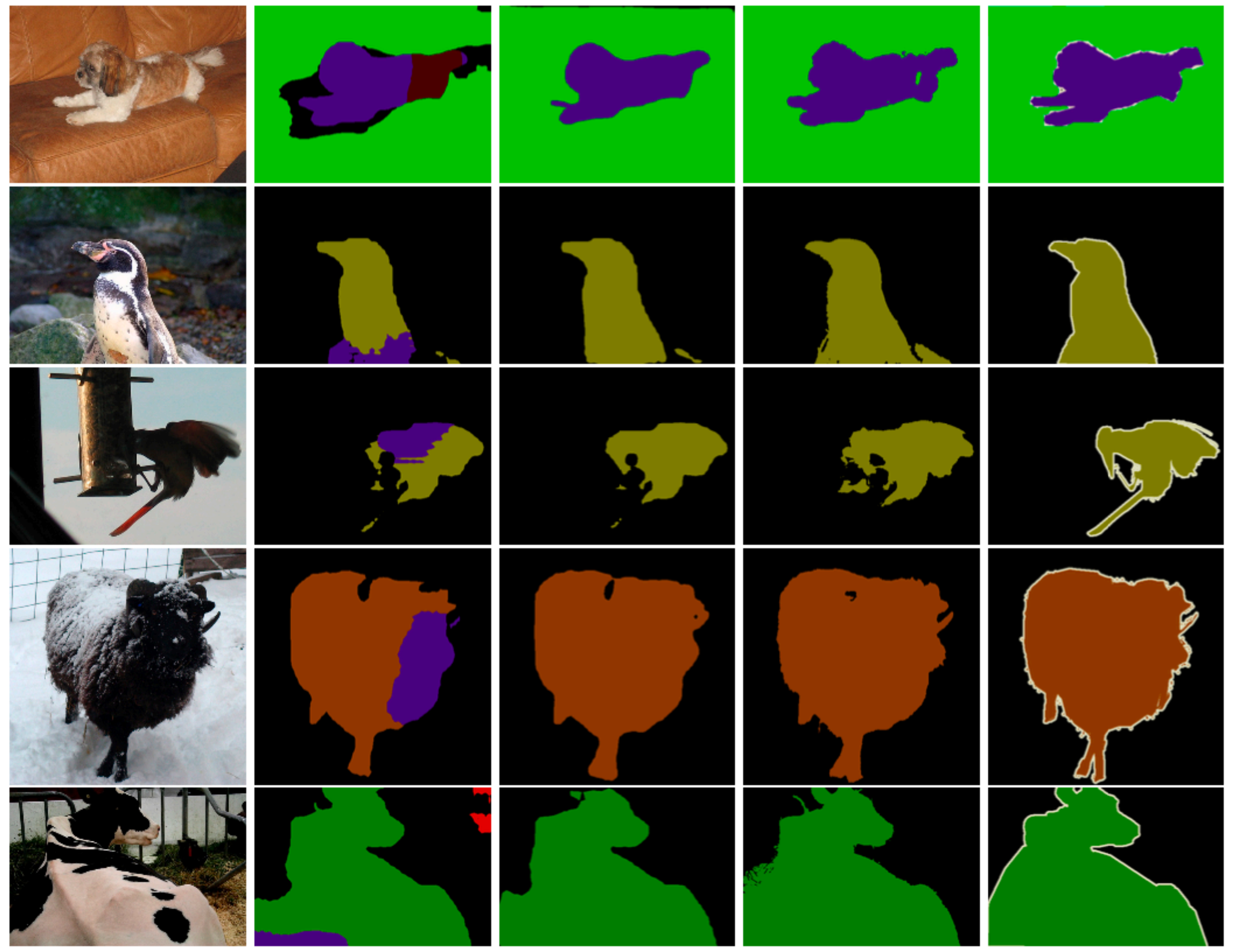
C → divides both c_i & c_{i+1}

Front End VGG-16

input: padded images (reflection padding)

output: $64 \times 64 \times 21$ feature maps $C = 21$

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean IoU
FCN-8s	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
DeepLab	72	31	71.2	53.7	60.5	77	71.9	73.1	25.2	62.6	49.1	68.7	63.3	73.9	73.6	50.8	72.3	42.1	67.9	52.6	62.1
DeepLab-Msc	74.9	34.1	72.6	52.9	61.0	77.9	73.0	73.7	26.4	62.2	49.3	68.4	64.1	74.0	75.0	51.7	72.7	42.5	67.2	55.7	62.9
Our front end	**82.2**	**37.4**	**72.7**	**57.1**	**62.7**	**82.8**	**77.8**	**78.9**	**28**	**70**	**51.6**	**73.1**	**72.8**	**81.5**	**79.1**	**56.6**	**77.1**	**49.9**	**75.3**	**60.9**	**67.6**



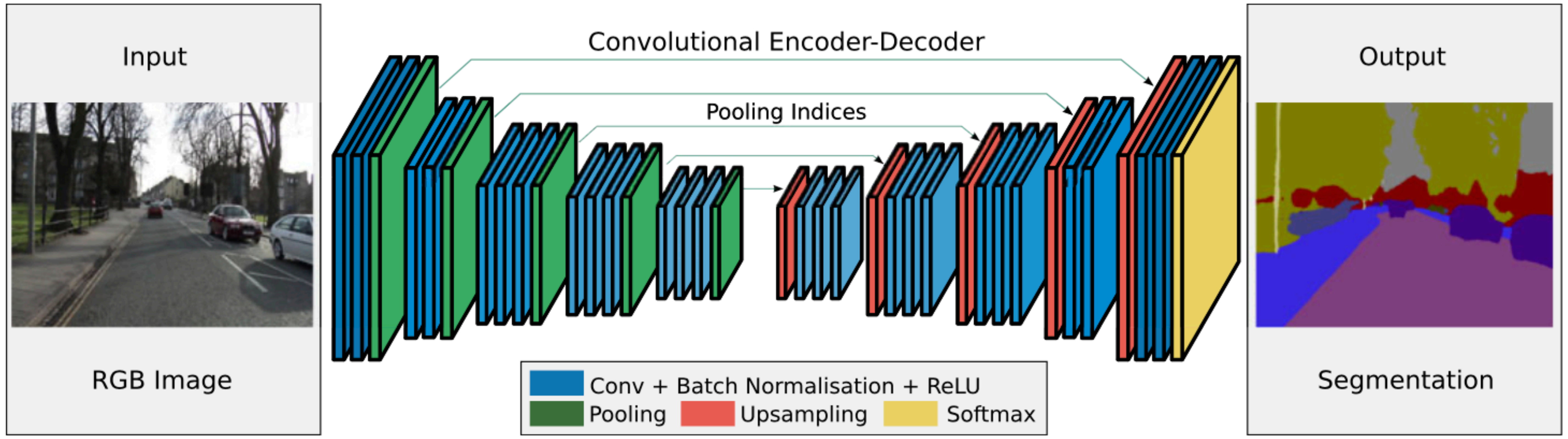
(a) Image (b) Front end (c) +Context (d) +CRF-RNN (e) Ground truth

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean IoU
Front end	86.3	38.2	76.8	**66.8**	63.2	87.3	78.7	82	33.7	76.7	53.5	73.7	76	76.6	83	**51.9**	77.8	44	79.9	**66.3**	69.8
Front + Basic	86.4	37.6	78.5	66.3	64.1	89.9	79.9	84.9	**36.1**	79.4	**55.8**	77.6	81.6	79	83.1	51.2	81.3	43.7	82.3	65.7	71.3
Front + Large	**87.3**	**39.2**	**80.3**	65.6	**66.4**	**90.2**	**82.6**	**85.8**	34.8	**81.9**	51.7	**79**	**84.1**	**80.9**	**83.2**	51.2	**83.2**	**44.7**	**83.4**	65.6	**72.1**
Front end + CRF	89.2	38.8	80	**69.8**	63.2	88.8	80	85.2	33.8	80.6	55.5	77.1	80.8	77.3	84.3	**53.1**	80.4	45	80.7	**67.9**	71.6
Front + Basic + CRF	89.1	38.7	81.4	67.4	65	91	81	86.7	**37.5**	81	**57**	79.6	83.6	79.9	**84.6**	52.7	83.3	44.3	82.6	67.2	72.7
Front + Large + CRF	**89.6**	**39.9**	**82.7**	66.7	**67.5**	**91.1**	**83.3**	**87.4**	36	**83.3**	52.5	**80.7**	**85.7**	**81.8**	84.4	52.6	**84.4**	**45.3**	**83.7**	66.7	**73.3**
Front end + RNN	88.8	38.1	80.8	**69.1**	65.6	89.9	79.6	85.7	36.3	83.6	57.3	77.9	83.2	77	**84.6**	**54.7**	82.1	**46.9**	80.9	66.7	72.5
Front + Basic + RNN	89	38.4	82.3	67.9	65.2	91.5	80.4	87.2	**38.4**	82.1	**57.7**	79.9	85	79.6	84.5	53.5	84	45	82.8	66.2	73.1
Front + Large + RNN	**89.3**	**39.2**	**83.6**	67.2	**69**	**92.1**	**83.1**	**88**	**38.4**	**84.8**	55.3	**81.2**	**86.7**	**81.3**	84.3	53.6	**84.4**	45.8	**83.8**	**67**	**73.9**

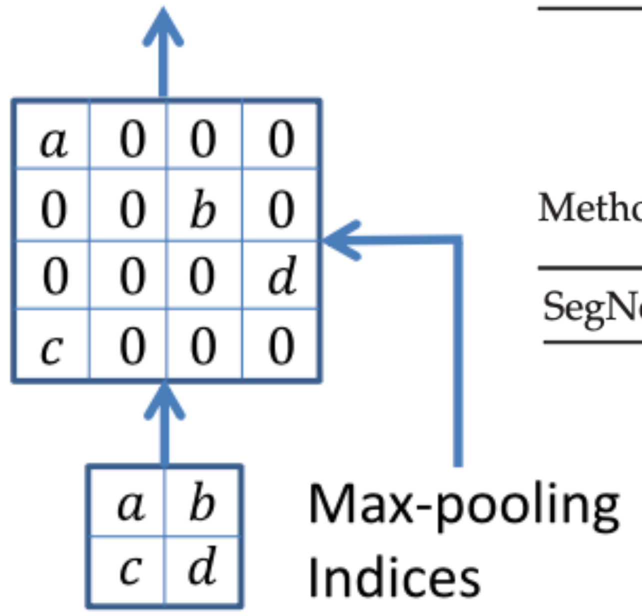
Yu, Fisher, and Vladlen Koltun. "Multi-scale context aggregation by dilated convolutions." *arXiv preprint arXiv:1511.07122* (2015).

SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation

YouTube Video

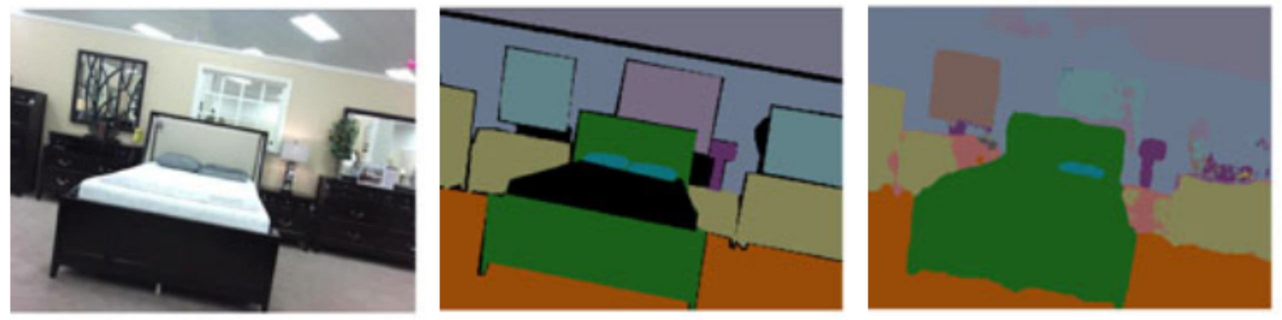


Convolution with trainable decoder filters



Method	Building	Tree	Sky	Car	Sign-Symbol	Road	Pedestrian	Fence	Column-Pole	Side-walk	Bicyclist	mIoU
SegNet (3.5K dataset training - 140K)	**89.6**	**83.4**	96.1	**87.7**	52.7	96.4	**62.2**	**53.45**	**32.1**	**93.3**	**36.5**	**90.40**

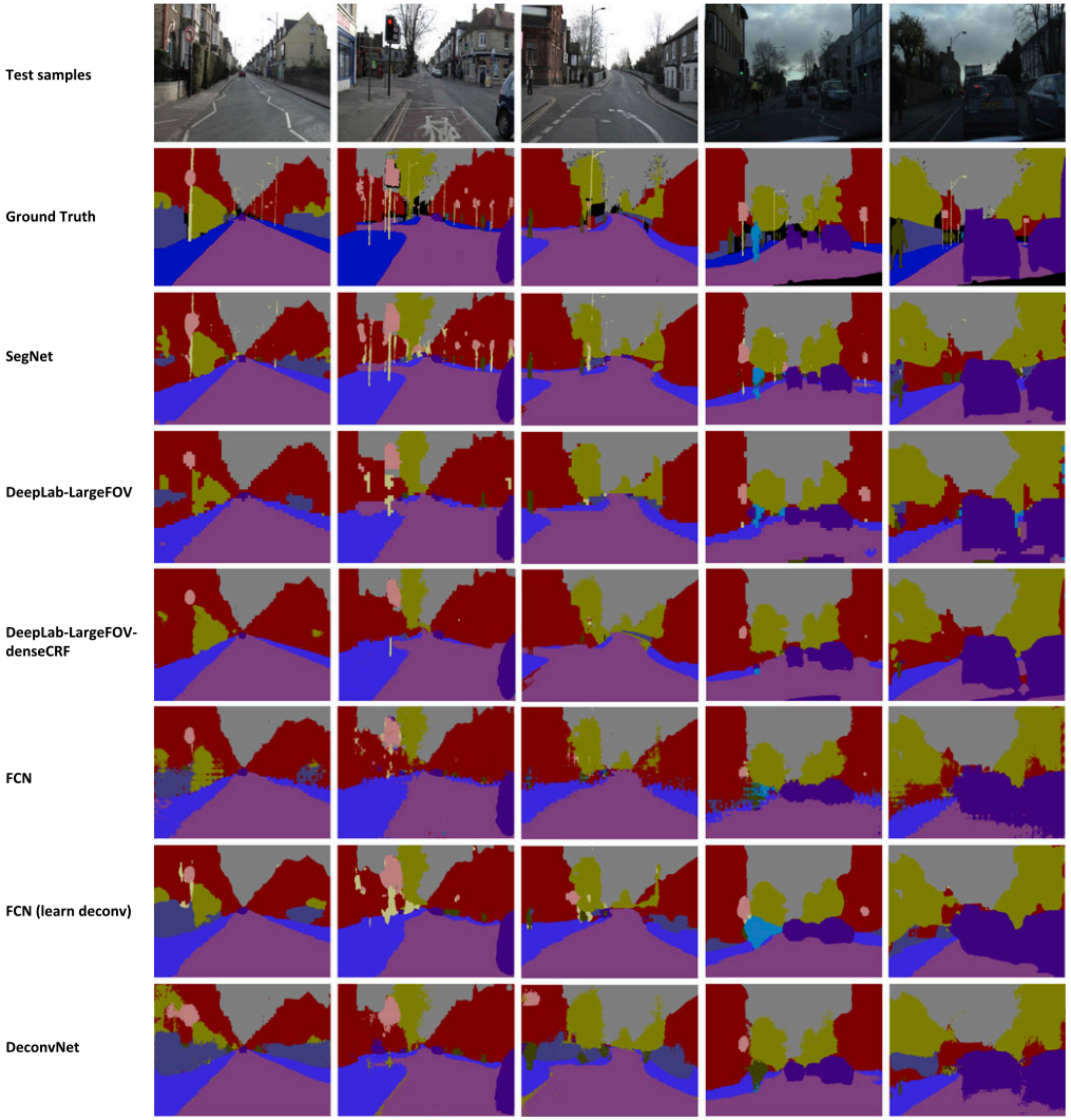
SUN RGB-D indoor scenes dataset



A Comparison of Computational Time and Hardware Resources Required for Various Deep Architectures

Network	Forward pass(ms)	Backward pass(ms)	GPU training memory (MB)	GPU inference memory (MB)	Model size (MB)
SegNet	422.50	488.71	6803	**1,052**	117
DeepLab-LargeFOV [3]	**110.06**	**160.73**	**5618**	1,993	83
FCN (learnt deconv) [2]	317.09	484.11	9735	1,806	539
DeconvNet [4]	474.65	602.15	9731	1,872	877

CamVid road scenes dataset

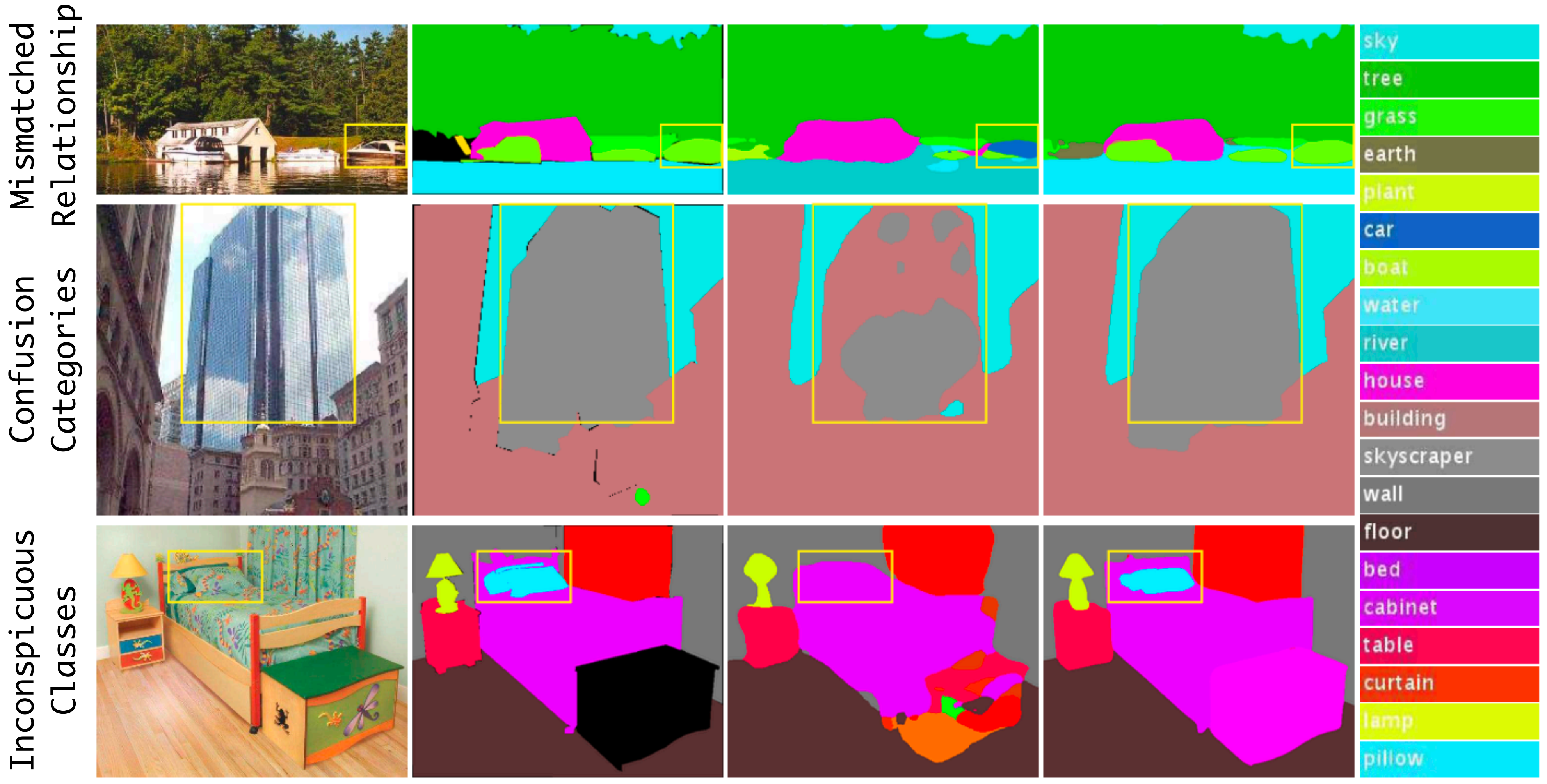


Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017): 2481-2495.

Pyramid Scene Parsing Network

YouTube Video

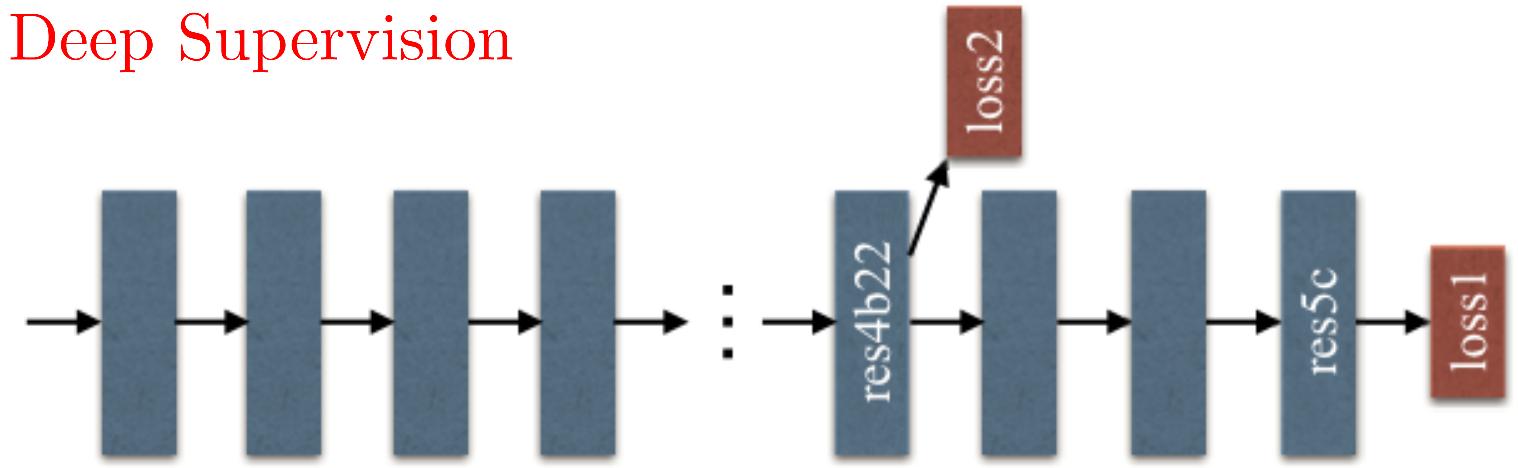
Scene parsing issues on ADE20K dataset



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

ADE20K dataset contains 150 stuff/object category labels (e.g., wall, sky, and tree) and 1,038 image-level scene descriptors (e.g., airport terminal, bedroom, and street).

Deep Supervision



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

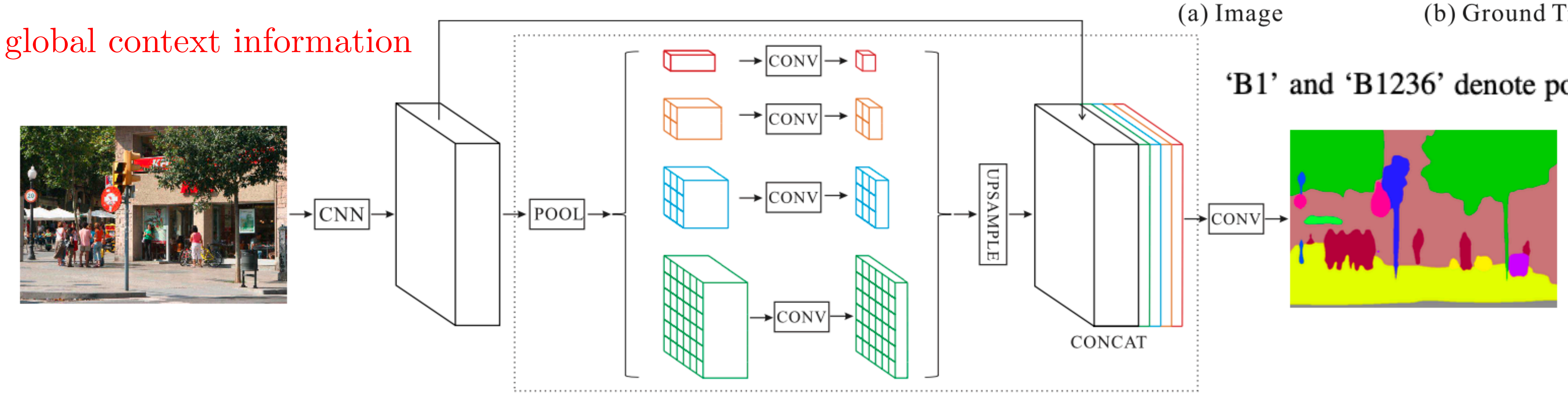
Cityscapes dataset

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

PASCAL VOC 2012 data

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

global context information



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

'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.

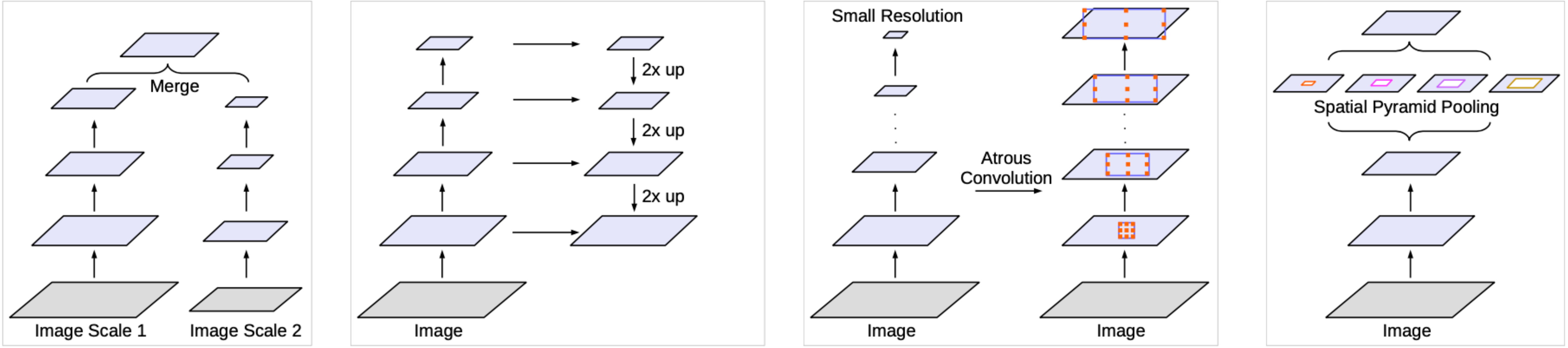
Method	Mean IoU(%)	Pixel Acc.(%)
ResNet50-Baseline	37.23	78.01
ResNet50+B1+MAX	39.94	79.46
ResNet50+B1+AVE	40.07	79.52
ResNet50+B1236+MAX	40.18	79.45
ResNet50+B1236+AVE	41.07	79.97
ResNet50+B1236+MAX+DR	40.87	79.61
ResNet50+B1236+AVE+DR	**41.68**	**80.04**

Zhao, Hengshuang, et al. "Pyramid scene parsing network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Rethinking Atrous Convolution for Semantic Image Segmentation

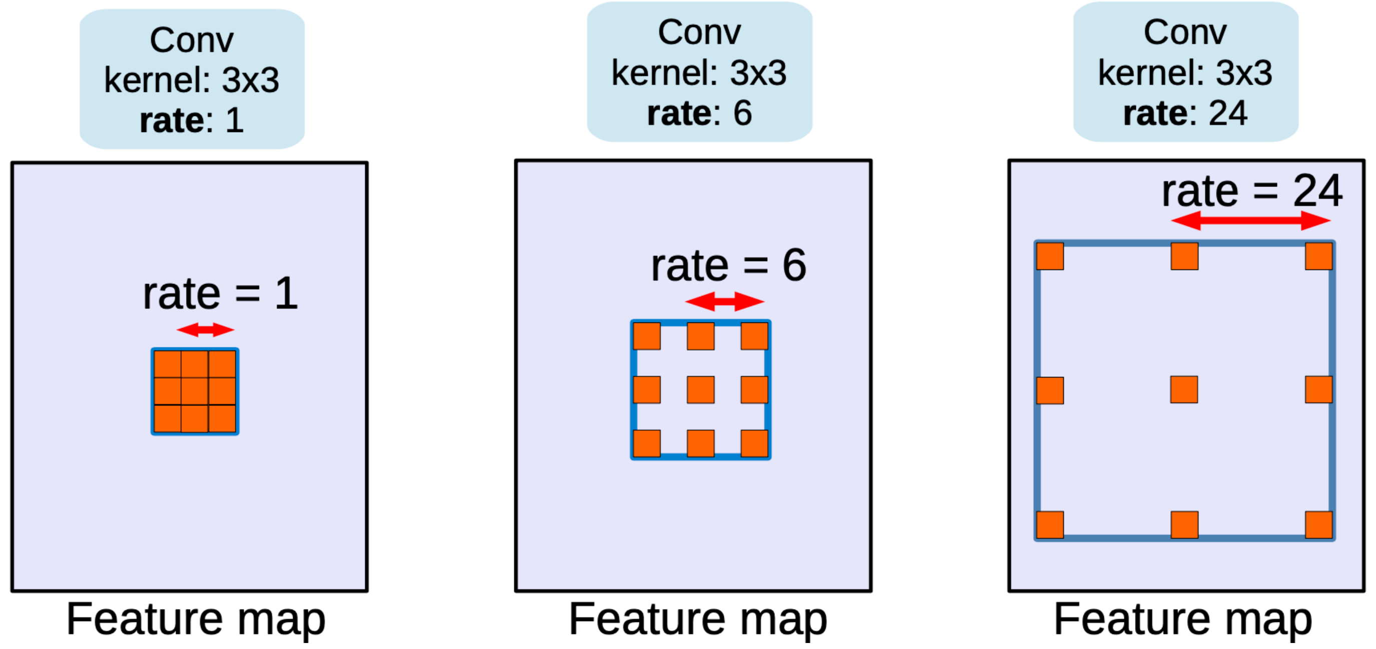
Boulder

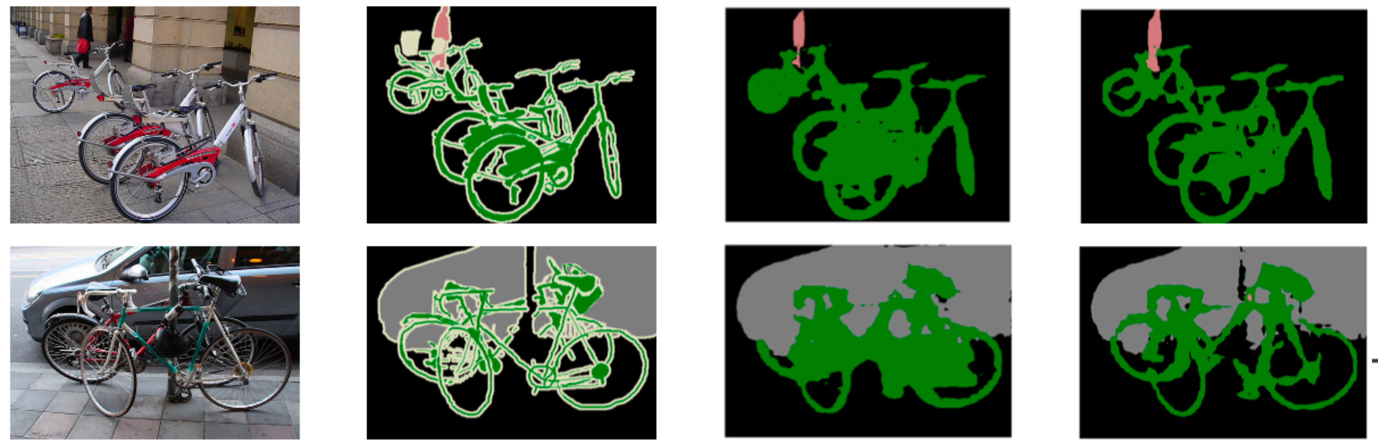
Two challenges in applying Deep Convolutional Neural Networks (DCNNs):
– reduced feature resolution
– objects at multiple scales



(a) Image Pyramid (b) Encoder-Decoder (c) Deeper w. Atrous Convolution (d) Spatial Pyramid Pooling

$$\boldsymbol{y}[\boldsymbol{i}] = \sum_{\boldsymbol{k}} \boldsymbol{x}[\boldsymbol{i} + r \cdot \boldsymbol{k}]\boldsymbol{w}[\boldsymbol{k}]$$



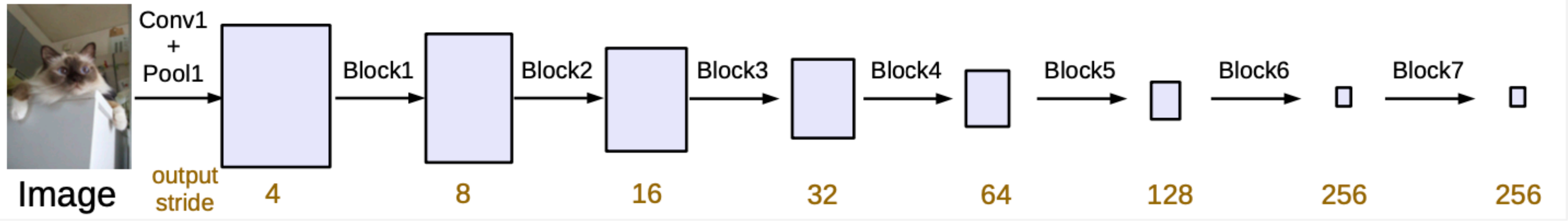


(a) Image (b) G.T. (c) w/o bootstrapping (d) w/ bootstrapping

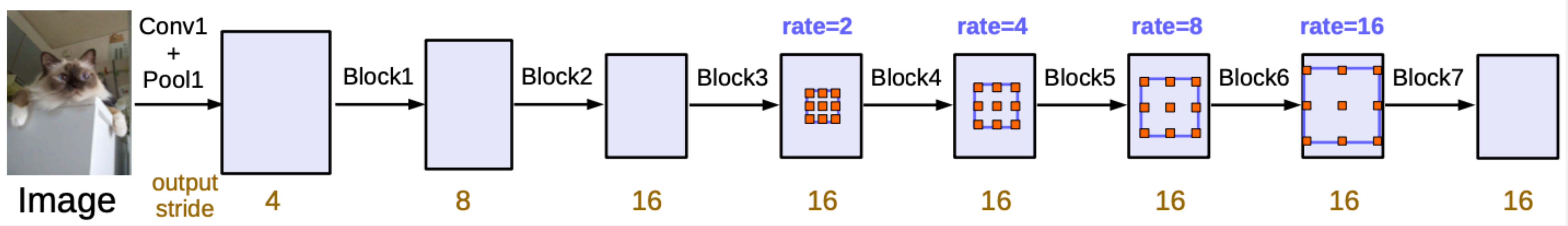
Bootstrapping on hard images improves segmentation

Method	mIOU
Adelaide_VeryDeep_FCN_VOC [85]	79.1
LRR_4x_ResNet-CRF [25]	79.3
DeepLabv2-CRF [11]	79.7
CentraleSupelec Deep G-CRF [8]	80.2
HikSeg_COCO [80]	81.4
SegModel [75]	81.8
Deep Layer Cascade (LC) [52]	82.7
TuSimple [84]	83.1
Large_Kernel_Matters [68]	83.6
Multipath-RefineNet [54]	84.2
ResNet-38_MS_COCO [86]	84.9
PSPNet [95]	85.4
IDW-CNN [83]	86.3
CASIA_IVA_SDN [23]	86.6
DIS [61]	86.8
DeepLabv3	85.7
DeepLabv3-JFT	86.9

Performance on PASCAL VOC 2012 *test* set.



(a) Going deeper without atrous convolution.



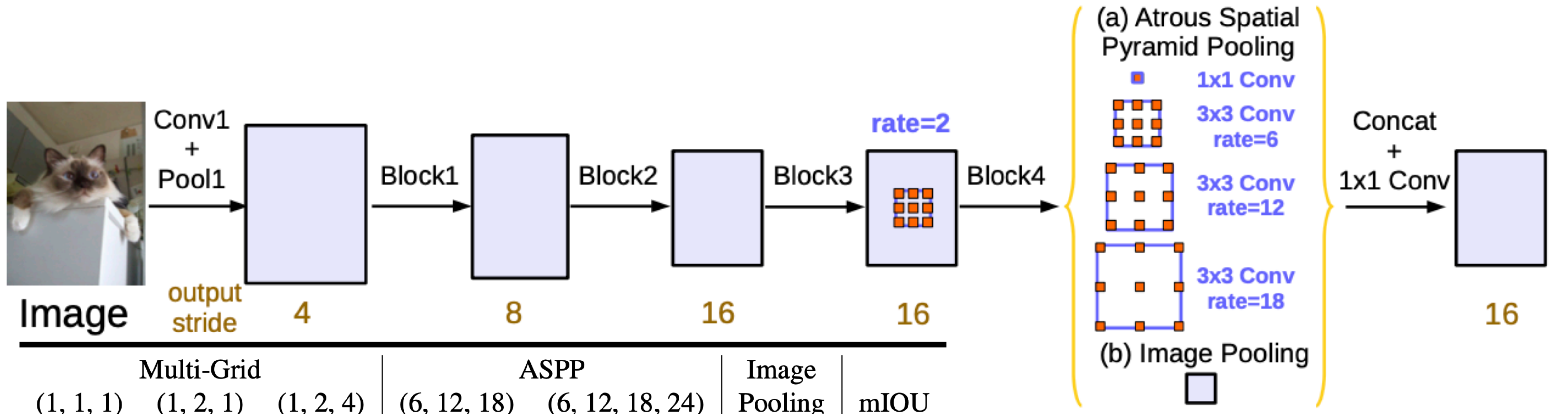
(b) Going deeper with atrous convolution. Atrous convolution with $rate > 1$ is applied after block3 when $output\_stride = 16$.

output_stride	8	16	32	64	128	256
mIOU	75.18	73.88	70.06	59.99	42.34	20.29

Network	block4	block5	block6	block7
ResNet-50	64.81	72.14	74.29	73.88
ResNet-101	68.39	73.21	75.34	75.76

Multi-Grid	block4	block5	block6	block7
(1, 1, 1)	68.39	73.21	75.34	75.76
(1, 2, 1)	70.23	75.67	76.09	**76.66**
(1, 2, 3)	73.14	75.78	75.96	76.11
(1, 2, 4)	73.45	75.74	75.85	76.02
(2, 2, 2)	71.45	74.30	74.70	74.62

Multi_Grid $= (r_1, r_2, r_3)$ the unit rates for the three convolutional layers within block4 to block7.

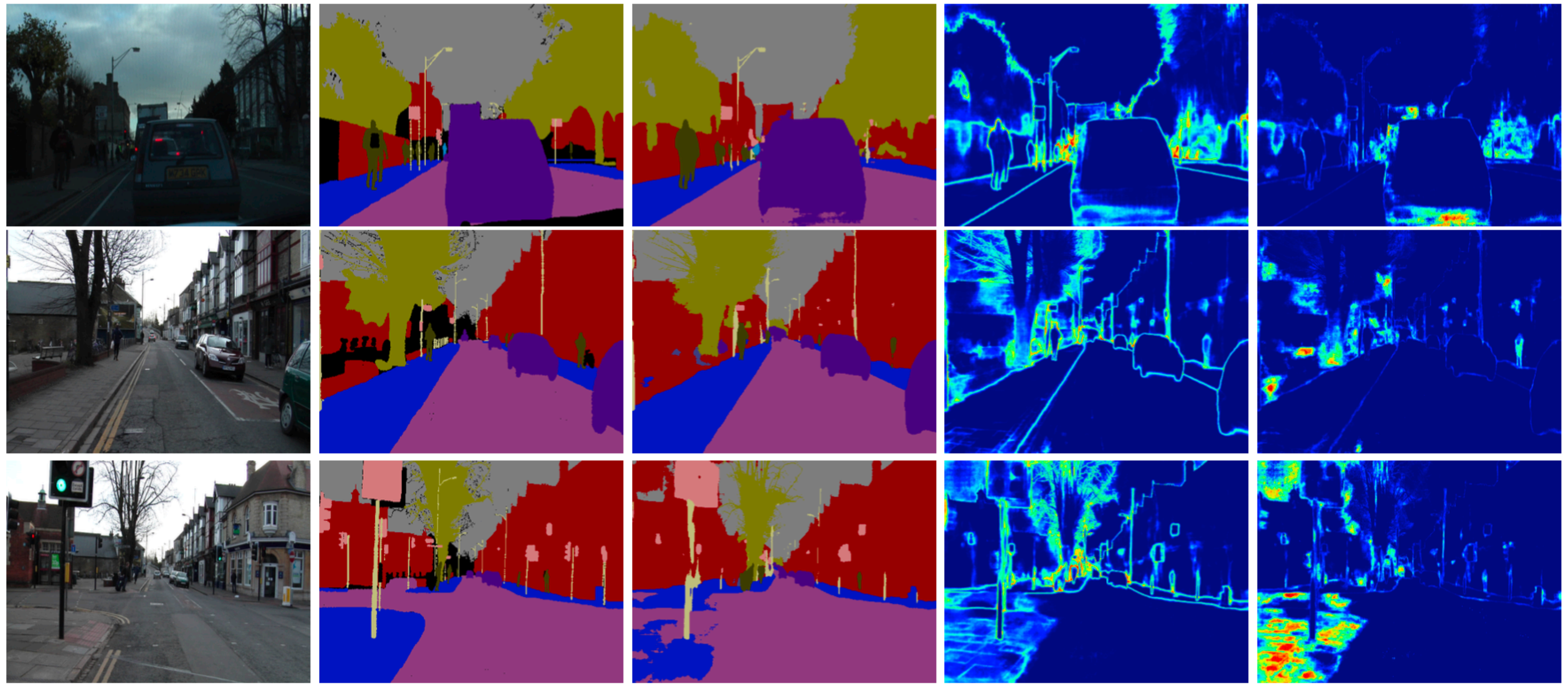


Multi-Grid			ASPP		Image Pooling	mIOU
(1, 1, 1)	(1, 2, 1)	(1, 2, 4)	(6, 12, 18)	(6, 12, 18, 24)		
✓			✓			75.36
	✓		✓			75.93
		✓	✓			76.58
		✓		✓		76.46
		✓	✓		✓	77.21

Chen, Liang-Chieh, et al. "Rethinking atrous convolution for semantic image segmentation." *arXiv preprint arXiv:1706.05587* (2017).

Boulder

What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?



(a) Input Image (b) Ground Truth (c) Semantic Segmentation (d) Aleatoric Uncertainty (e) Epistemic Uncertainty

Aleatoric Uncertainty: noise inherent in the obsevations
– homoscedastic: constant for different inputs
– heteroscedastic: depends on the inputs to the model

Epistemic (Model) Uncertainty: can be explained away given enough data

Epistemic Uncertainty in Bayesian Deep Learning

$\mathbf{W} \sim \mathcal{N}(0, I) \rightarrow$ prior distribution over the weights of neural network

$\mathbf{f}^{\mathbf{W}}(\mathbf{x}) \rightarrow$ random output of a Bayesian Neural Network

$p(\mathbf{y}|\mathbf{f}^{\mathbf{W}}(\mathbf{x})) \rightarrow$ model likelihood

$\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}, \mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_N\} \rightarrow$ dataset

$p(\mathbf{W}|\mathbf{X}, \mathbf{Y}) \rightarrow$ posterior over the weights (Bayesian inference)

$$p(\mathbf{W}|\mathbf{X}, \mathbf{Y}) = p(\mathbf{Y}|\mathbf{X}, \mathbf{W})p(\mathbf{W})/p(\mathbf{Y}|\mathbf{X})$$

$p(\mathbf{Y}|\mathbf{X}) \rightarrow$ marginal probability (cannot be evaluated analytically)

$q^*_\theta(\mathbf{W}) \rightarrow$ a simple distribution approximating the posterior

$$\mathcal{L}(\theta, p) = -\frac{1}{N}\sum_{i=1}^{N} \log p(\mathbf{y}_i|\mathbf{f}^{\widehat{\mathbf{W}}_i}(\mathbf{x}_i)) + \frac{1-p}{2N}||\theta||^2$$

$\widehat{\mathbf{W}}_i \sim q^*_\theta(\mathbf{W}) \rightarrow$ dropout distribution

$p \rightarrow$ dropout probability

$\theta \rightarrow$ parameters of the simple distribution (weight matrices)

Heteroscedastic Aleatoric Uncertainty (Regression)

$$-\log p(\mathbf{y}_i|\mathbf{f}^{\widehat{\mathbf{W}}_i}(\mathbf{x}_i)) \propto \underbrace{\frac{1}{2\hat{\sigma}_i^2}}_{\text{learned loss attenuation}}\|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2 + \frac{1}{2}\log \hat{\sigma}_i^2$$

$$[\hat{\mathbf{y}}_i, \hat{\sigma}_i^2] = \mathbf{f}^{\widehat{\mathbf{W}}_i}(\mathbf{x}_i)$$

$$\frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{y}}_t^2 - \Bigg(\underbrace{\frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{y}}_t}_{\text{predictive mean}}\Bigg)^2 + \frac{1}{T}\sum_{t=1}^{T}\hat{\sigma}_t^2 \rightarrow \text{predictive variance}$$

$$\hat{\mathbf{y}}_t, \hat{\sigma}_t^2 = \mathbf{f}^{\widehat{\mathbf{W}}_t}(\mathbf{x})$$

CamVid dataset for road scene segmentation

	IoU
DenseNet (Our Implementation)	67.1
+ Aleatoric Uncertainty	67.4
+ Epistemic Uncertainty	67.2
+ Aleatoric & Epistemic	**67.5**

Heteroscedastic Aleatoric Uncertainty (Classification)

$$p(\mathbf{y}_i|\mathbf{f}^{\widehat{\mathbf{W}}_i}(\mathbf{x}_i)) = \mathbf{y}_i^T \text{softmax}(\hat{\mathbf{y}}_i + \hat{\sigma}_i\epsilon_i), \epsilon_i \sim \mathcal{N}(0, I)$$

$$\mathbf{p} = \frac{1}{T}\sum_{t=1}^{T} \text{softmax}(\hat{\mathbf{y}}_t + \hat{\sigma}_t\epsilon_t), \epsilon_t \sim \mathcal{N}(0, I)$$

$H(\mathbf{p}) = -\sum_{c=1}^{C} p_c \log p_c \rightarrow$ uncertainty of probability vector $\mathbf{p}$

Each datapoint and each pixel will have its own prediction and uncertainty!

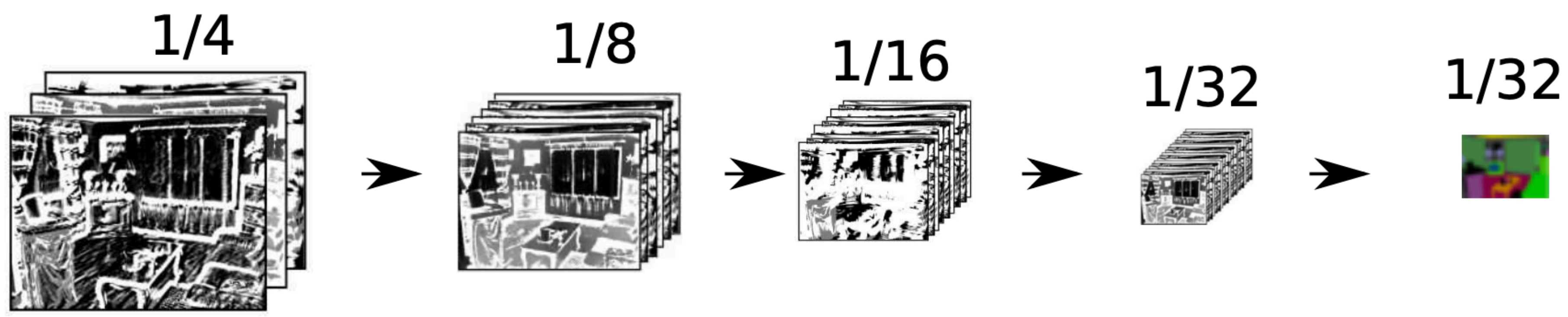
Kendall, Alex, and Yarin Gal. "What uncertainties do we need in bayesian deep learning for computer vision?." *arXiv preprint arXiv:1703.04977* (2017).

RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation

Boulder

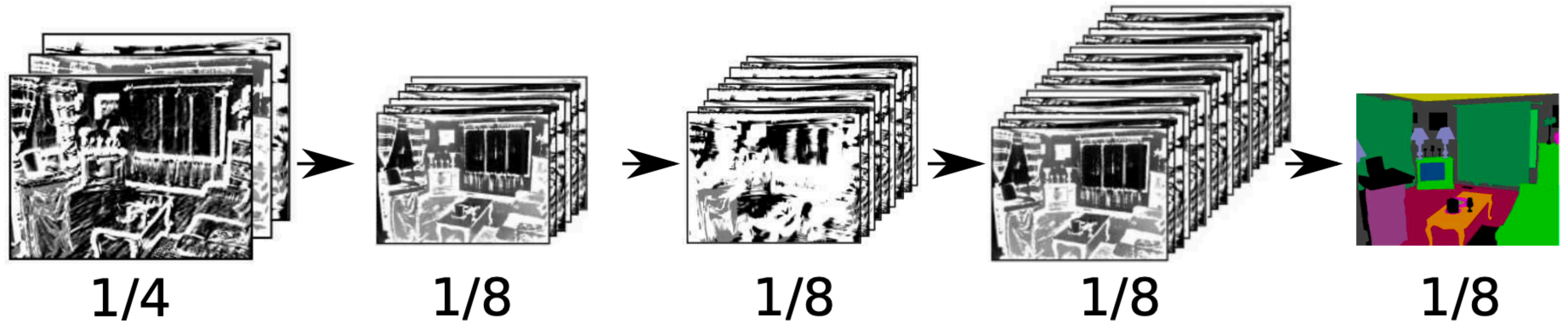
object parsing (left) and semantic segmentation (right)

(a) ResNet



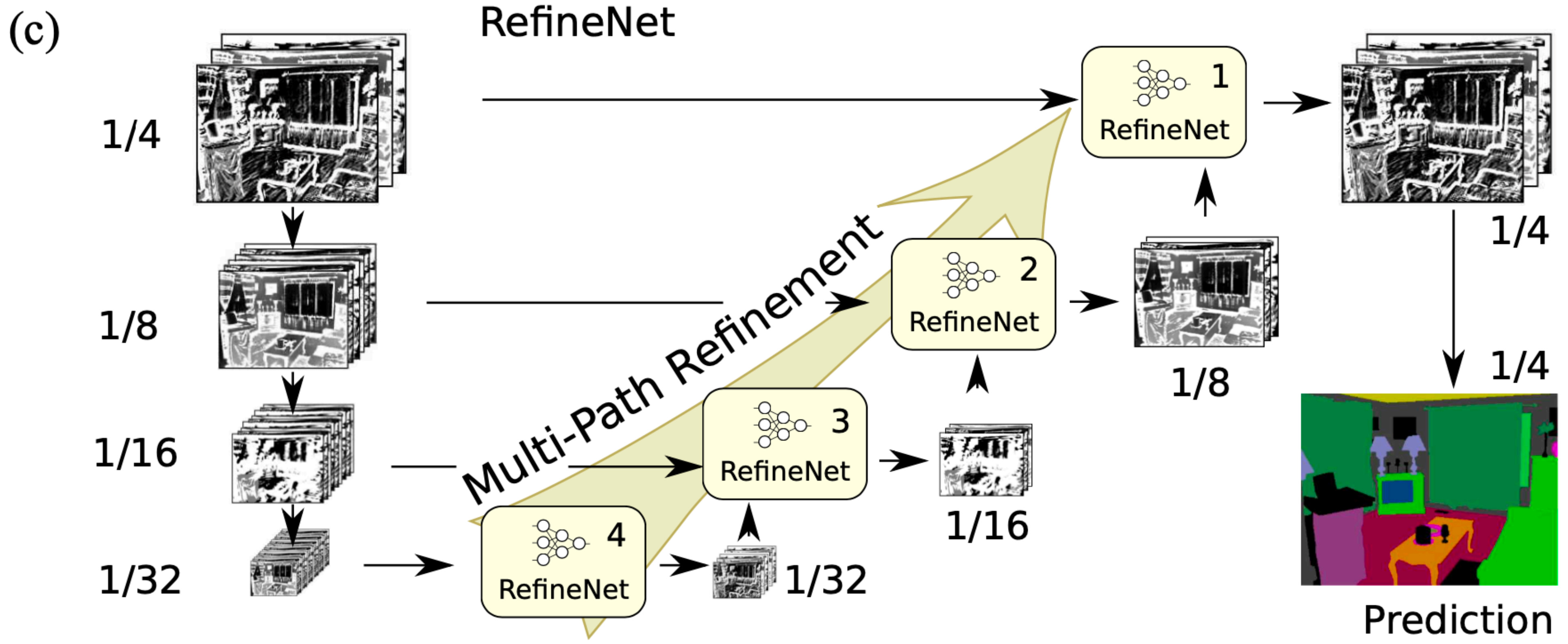
suffers from downscaling of the feature maps

(b) Dilated convolutions



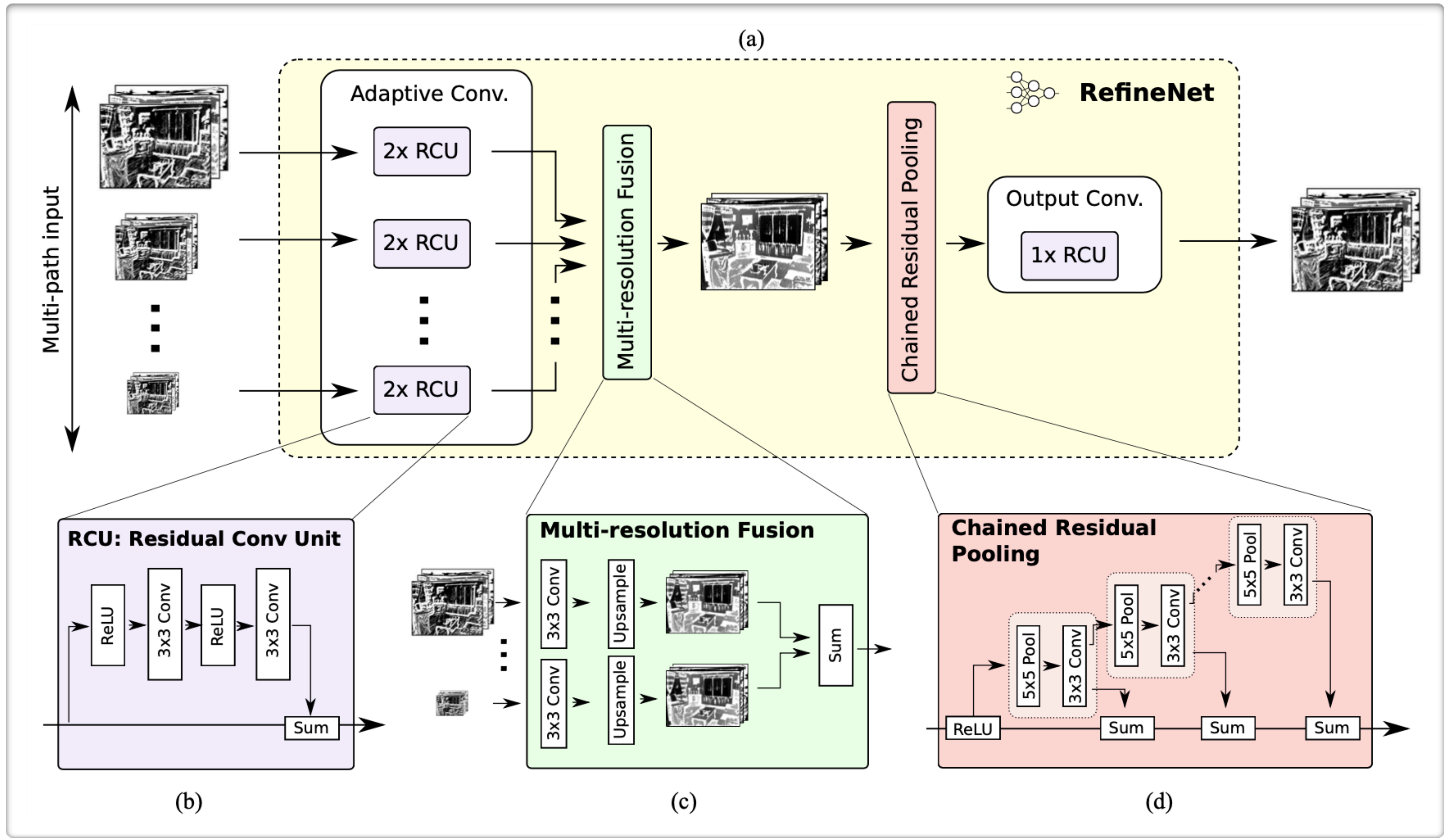
computationally expensive to train and quickly reaches memory limits

Effectively combine high-level semantics and low-level features to produce high-resolution segmentation maps.



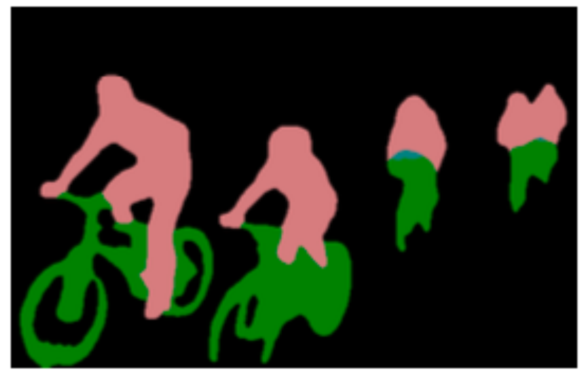
Method	mean
FCN-8s [36]	62.2
DeconvNet [38]	72.5
CRF-RNN [47]	74.7
BoxSup [10]	75.2
DPN [35]	77.5
Context [30]	78.0
DeepLab [5]	72.7
DeepLab2-Res101 [6]	79.7
CSupelec-Res101 [4]	80.2
RefineNet-Res101	**82.4**
RefineNet-Res152	**83.4**

PASCAL VOC 2012 test set (IoU scores).



(a) Test Image

(b) Ground Truth

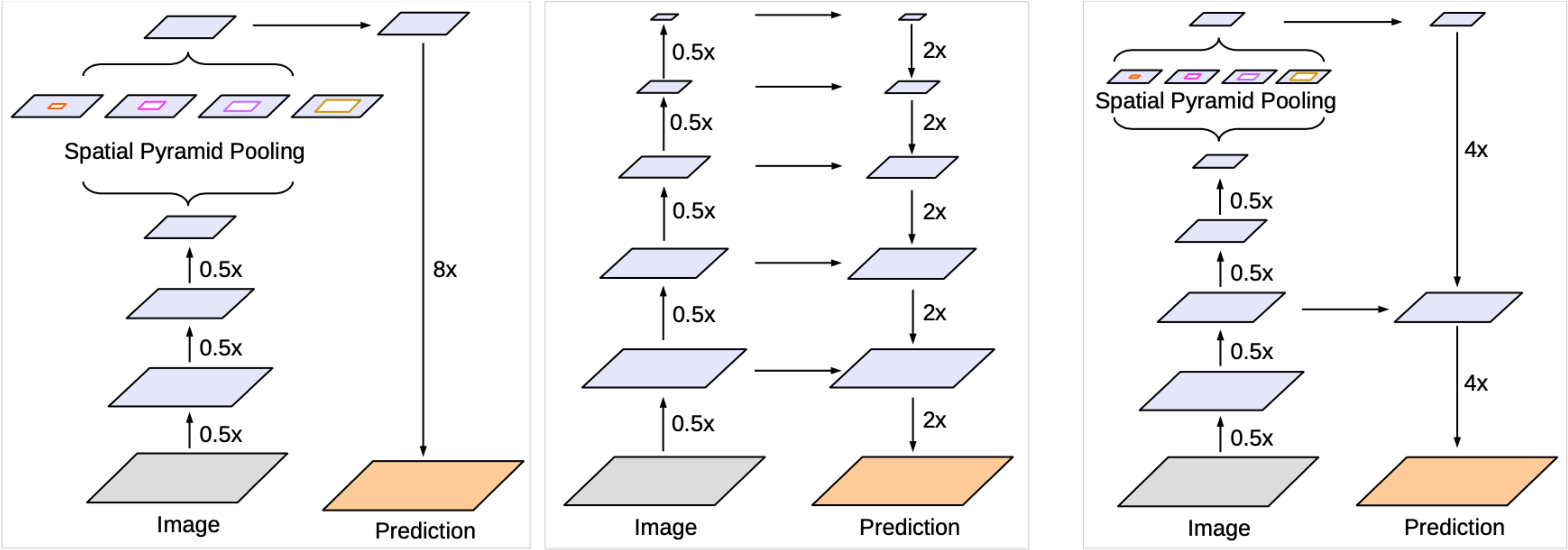


(c) Prediction

Lin, Guosheng, et al. "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

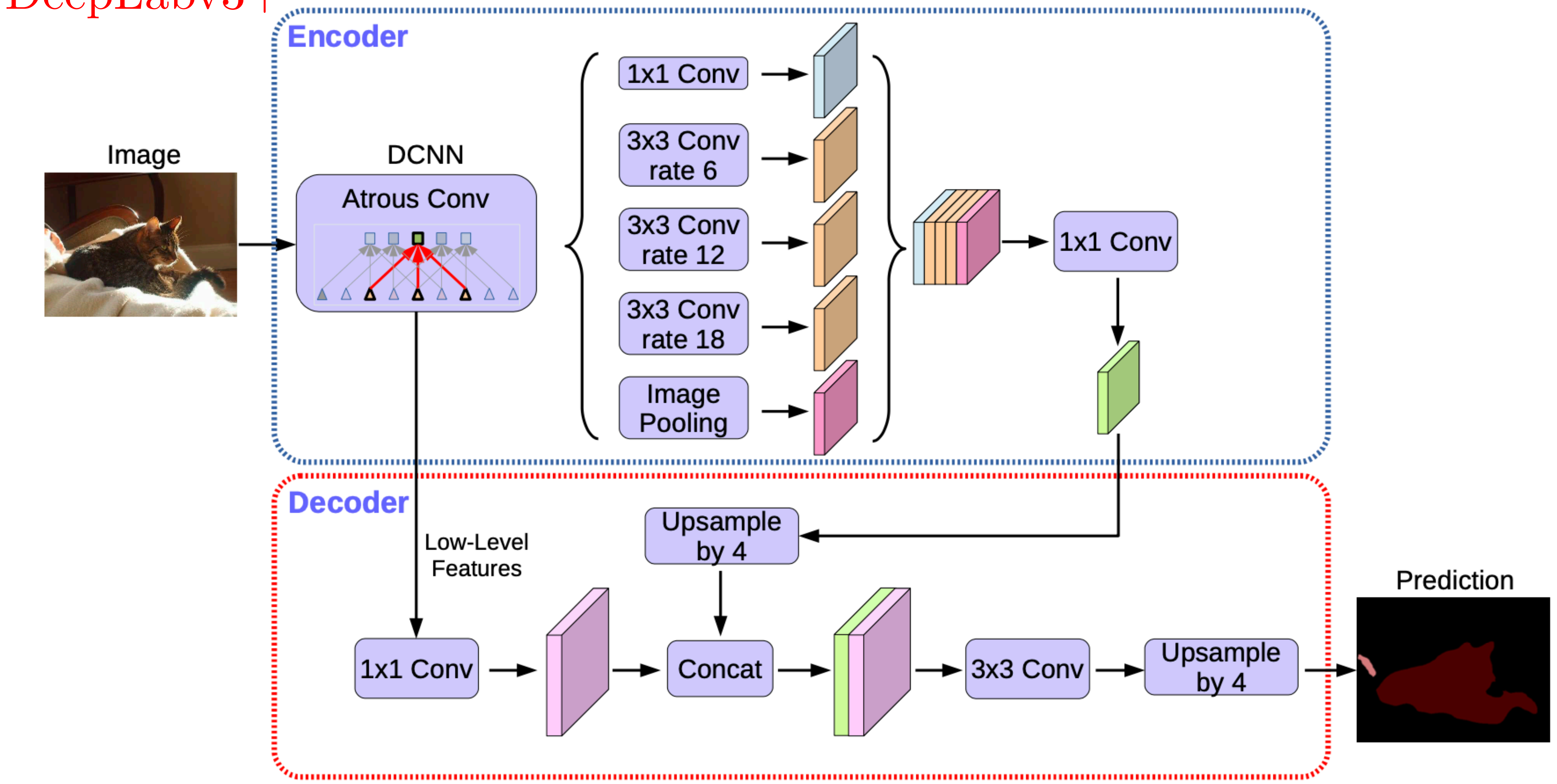
Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation

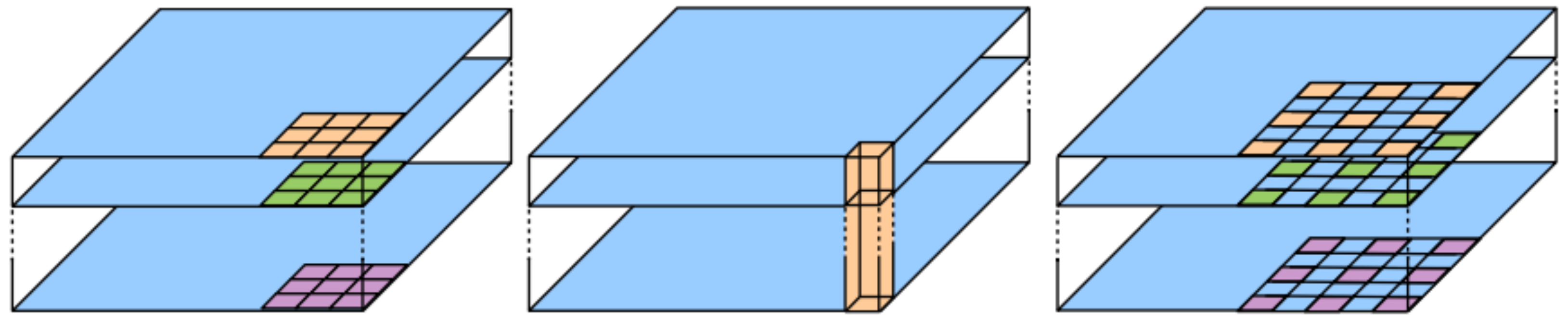
YouTube Video



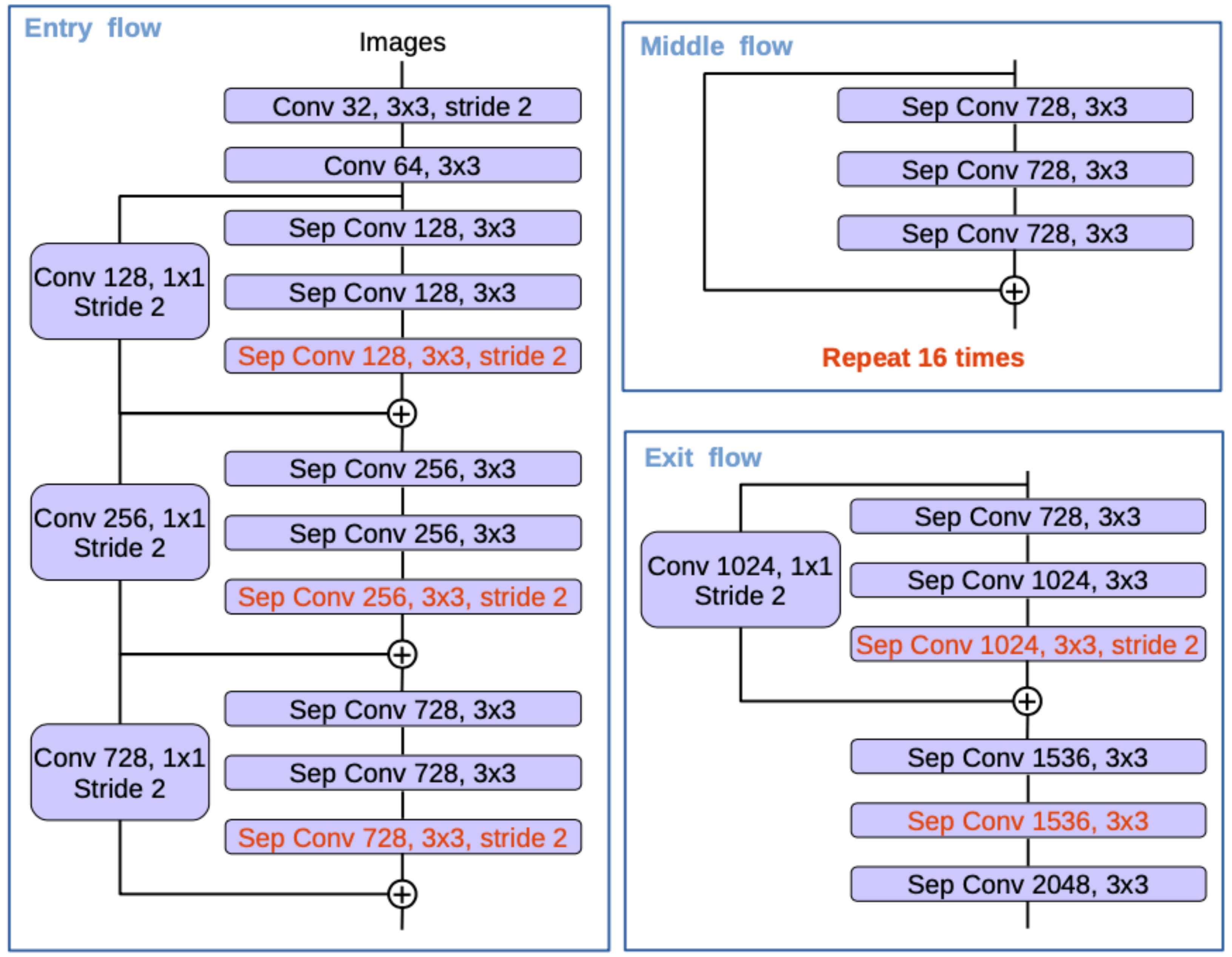
(a) Spatial Pyramid Pooling (b) Encoder-Decoder (c) Encoder-Decoder with Atrous Conv

DeepLabv3+





(a) Depthwise conv. (b) Pointwise conv. (c) Atrous depthwise conv.



PASCAL VOC 2012 test set

Method	mIOU
Deep Layer Cascade (LC) [82]	82.7
TuSimple [77]	83.1
Large_Kernel_Matters [60]	83.6
Multipath-RefineNet [58]	84.2
ResNet-38_MS_COCO [83]	84.9
PSPNet [24]	85.4
IDW-CNN [84]	86.3
CASIA_IVA_SDN [63]	86.6
DIS [85]	86.8
DeepLabv3 [23]	85.7
DeepLabv3-JFT [23]	86.9
DeepLabv3+ (Xception)	87.8
DeepLabv3+ (Xception-JFT)	89.0

Effect of decoder 1×1 convolution

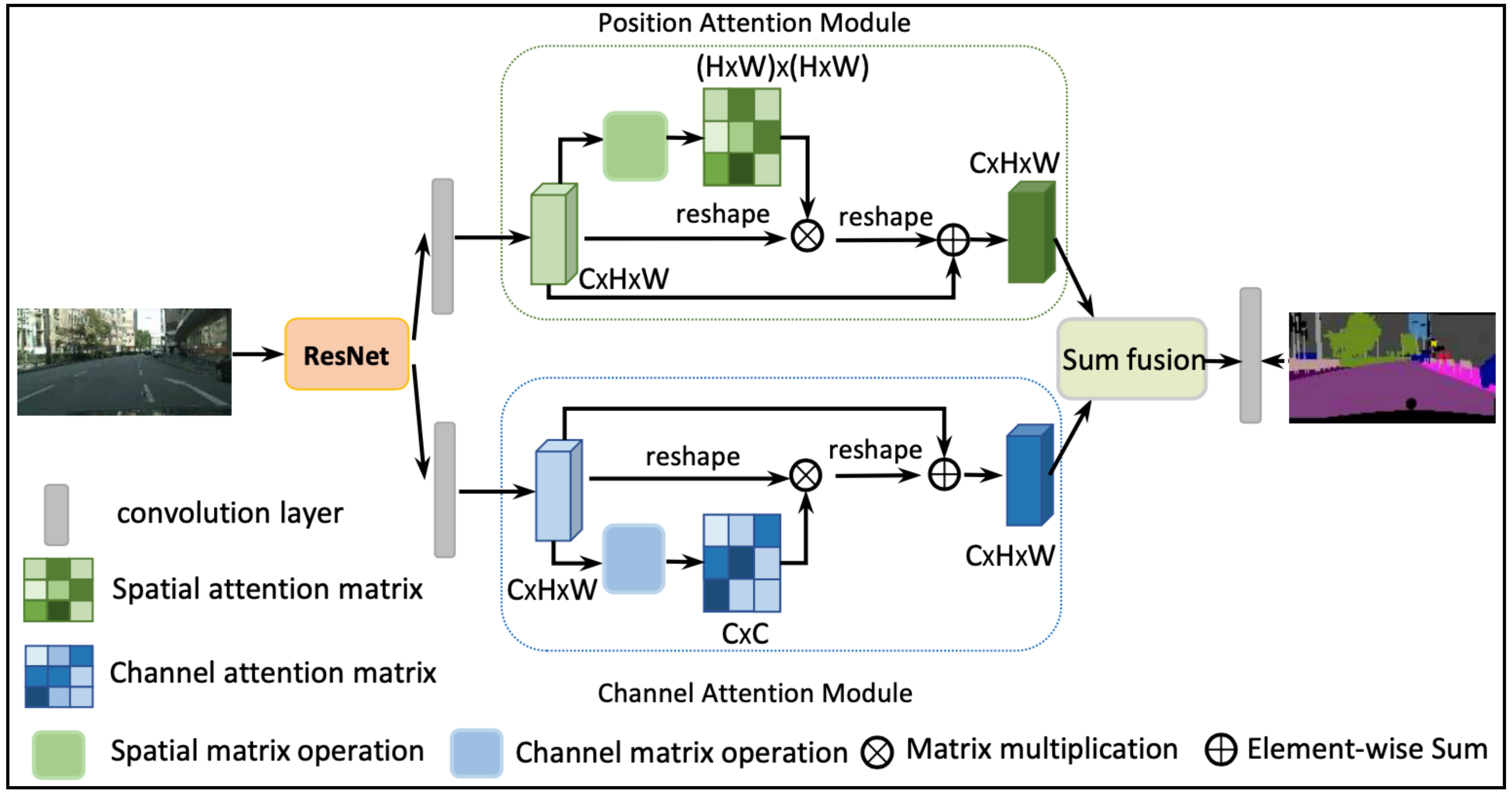
Channels	8	16	32	48	64
mIOU	77.61%	77.92%	78.16%	**78.21%**	77.94%

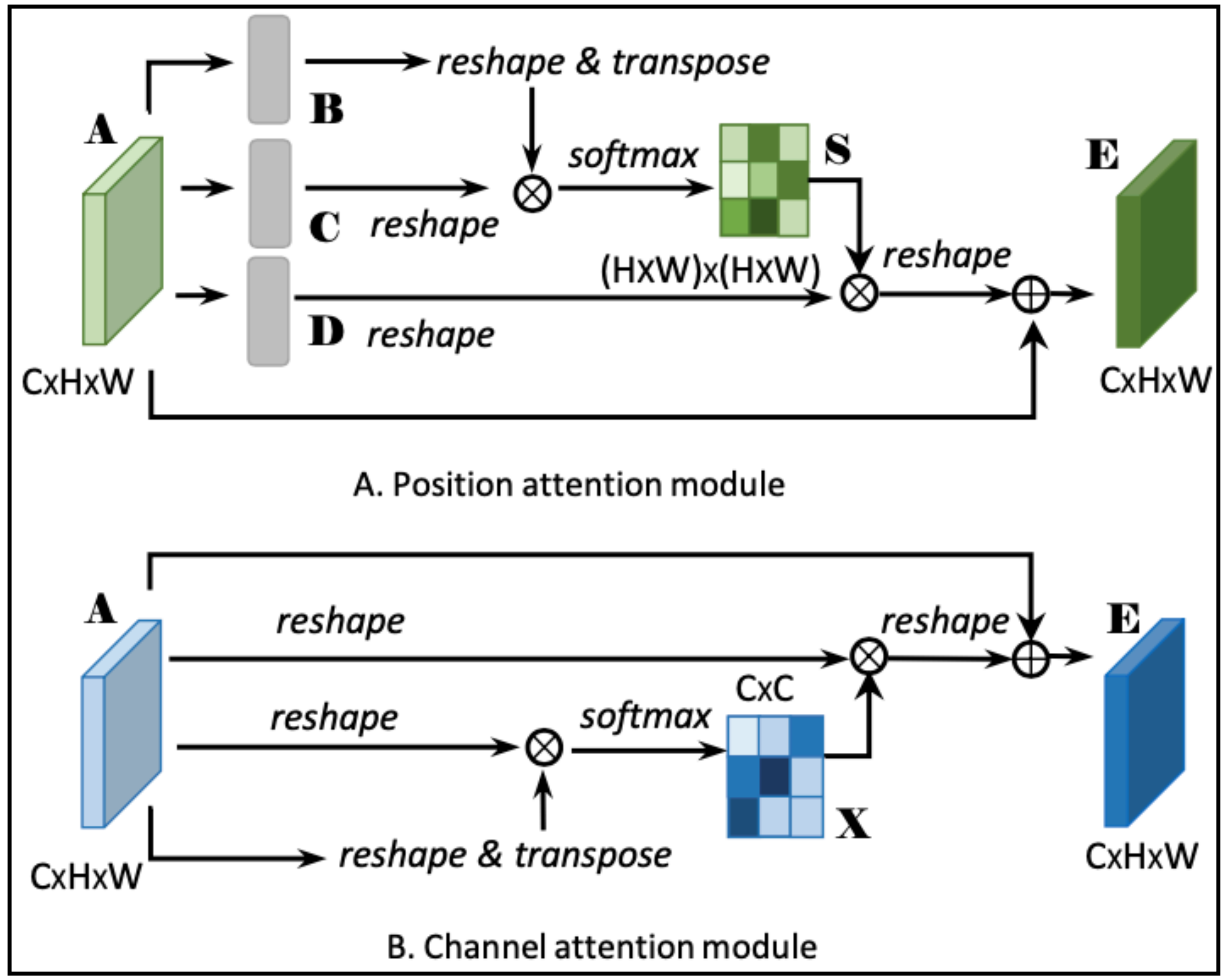
Effect of decoder 3×3 convolution

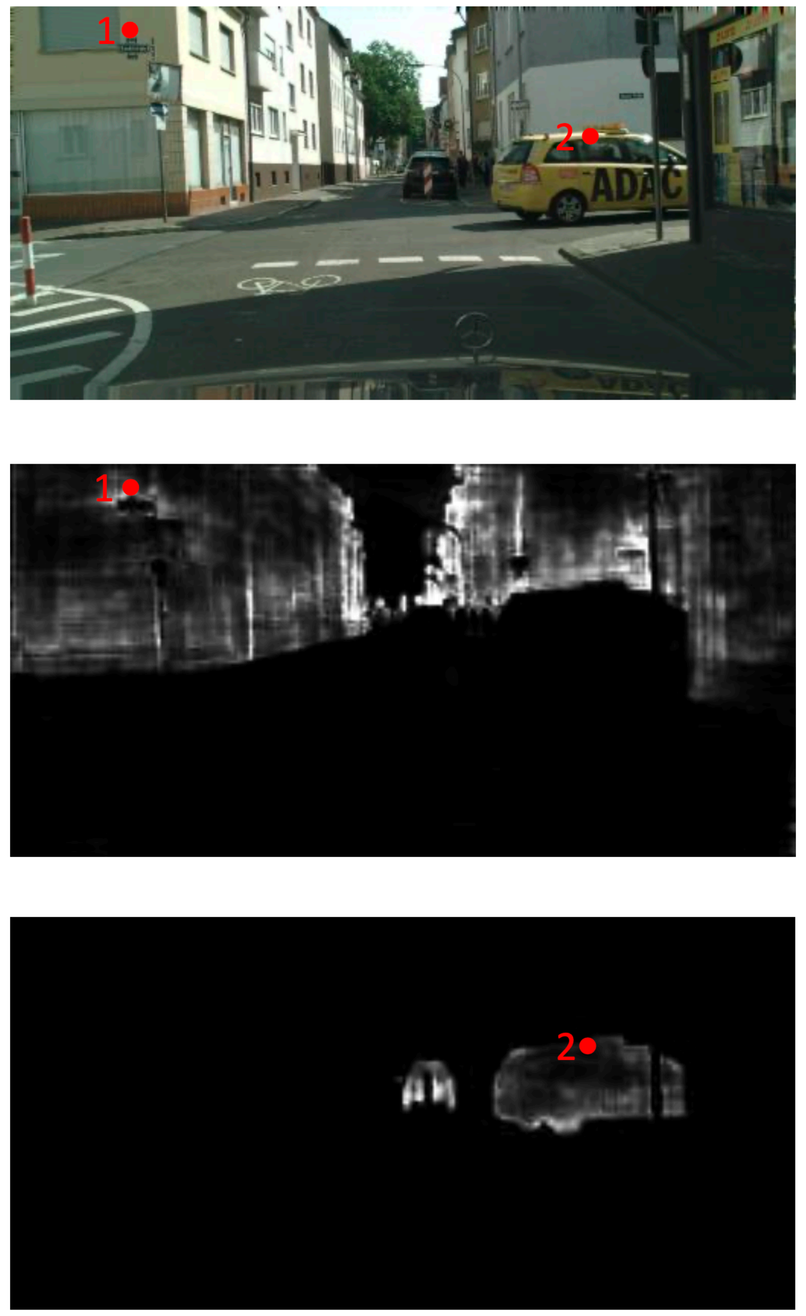
Features Conv2	Features Conv3	3×3 Conv Structure	mIOU
✓		$[3 \times 3, 256]$	78.21%
✓		$[3 \times 3, 256] \times 2$	**78.85%**
✓		$[3 \times 3, 256] \times 3$	78.02%
✓		$[3 \times 3, 128]$	77.25%
✓		$[1 \times 1, 256]$	78.07%
✓	✓	$[3 \times 3, 256]$	78.61%

Chen, Liang-Chieh, et al. "Encoder-decoder with atrous separable convolution for semantic image segmentation." *Proceedings of the European conference on computer vision (ECCV)*. 2018.

Dual Attention Network for Scene Segmentation







Dual Attention Network (DANet)
– Cityscapes – PASCAL VOC2012
– PASCAL Context – COCO Stuff
Stuff: sky, road, grass, etc.
Objects: person, car, bicycle, etc.

Position Attention Module

$A \in \mathbb{R}^{C\times H\times W} \rightarrow$ local feature
$B, C \in \mathbb{R}^{C\times H\times W} \rightarrow$ after conv on A
$B, C \in \mathbb{R}^{C\times N} \rightarrow$ reshape ($N = HW$)
$S \in \mathbb{R}^{N\times N} \rightarrow$ spatial attention map
$S = \mathrm{softmax}(B^T C)$

$$s_{ji} = \frac{\exp(B_i \cdot C_j)}{\sum_{i=1}^{N} \exp(B_i \cdot C_j)}$$

$\sum_{i=1}^{N} s_{ji} = 1$

$D \in \mathbb{R}^{C\times H\times W} \rightarrow$ after conv on A
$D \in \mathbb{R}^{C\times N} \rightarrow$ reshape
$DS^T \in \mathbb{R}^{C\times H\times W} \rightarrow$ after reshape
$E \in \mathbb{R}^{C\times H\times W}$

$$E_j = \alpha \sum_{i=1}^{N} s_{ji} D_i + A_j$$

Channel Attention Module

$A \in \mathbb{R}^{C\times H\times W} \rightarrow$ local feature
$A \in \mathbb{R}^{C\times N} \rightarrow$ reshape
$X = \mathrm{softmax}(AA^T) \in \mathbb{R}^{C\times C}$

$$x_{ji} = \frac{\exp(A_i \cdot A_j)}{\sum_{i=1}^{C} \exp(A_i \cdot A_j)}$$

$\sum_{i=1}^{C} x_{ji} = 1$

$X^T A \in \mathbb{R}^{C\times H\times W} \rightarrow$ after reshape

$$E_j = \beta \sum_{i=1}^{C} x_{ji} A_i + A_j$$

Method	BaseNet	PAM	CAM	Mean IoU%
Dilated FCN	Res50			70.03
DANet	Res50	✓		75.74
DANet	Res50		✓	74.28
DANet	Res50	✓	✓	76.34
Dilated FCN	Res101			72.54
DANet	Res101	✓		77.03
DANet	Res101		✓	76.55
DANet	Res101	✓	✓	77.57

Without CAM　With CAM　Groundtruth

Fu, Jun, et al. "Dual attention network for scene segmentation." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

Boulder

Questions?

YouTube Playlist