Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos,
Kevin Murphy, Alan L. Yuille
ICLR 2015

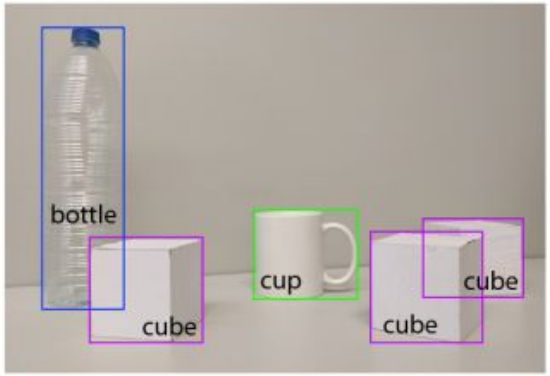
Carlos Feres & Mark Weber

Outline

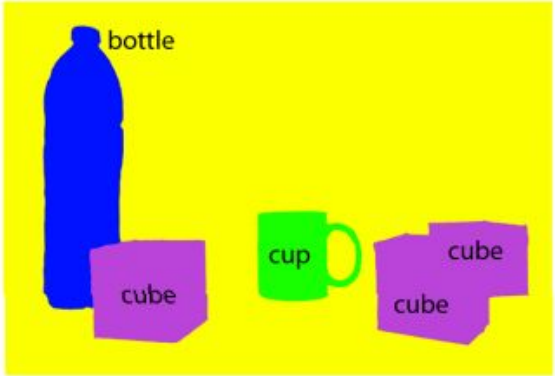
- Context
- Dense labeling challenge
  - Increasing resolution
  - Controlling the receptive field size
- Localization challenge
  - Fully-connected CRFs
  - Multi-scale prediction
- Experiments
- Comments

Context

Image classification



Object localization



Semantic segmentation

- Can we use a CNN for this task?

[1] Garcia-Garcia et al, *A Review on Deep Learning Techniques Applied to Semantic Segmentation*. arXiv:1704.06857v1

Context

- CNNs excel at classification/detection
  - Spatial invariance helps build hierarchical abstractions of data
- CNNs perform poorly on low-level tasks
  - Needs precise location and details
- Reduction of signal resolution
  - Information is not “dense”
- Spatial insensitivity (invariance)
  - Exact outline of objects is lost

Dense labeling challenge

- Want CNNs with **high spatial resolution & wide receptive field**

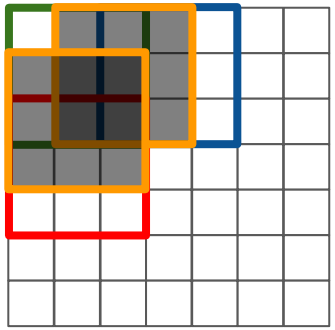
- Increase spatial resolution
  - *Hole algorithm*

- Adapt classification CNN for segmentation
  - Redefine the CNN architecture
  - Control the receptive field size

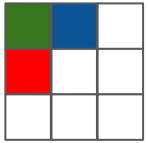
Increase spatial resolution

- How do CNNs lose resolution?

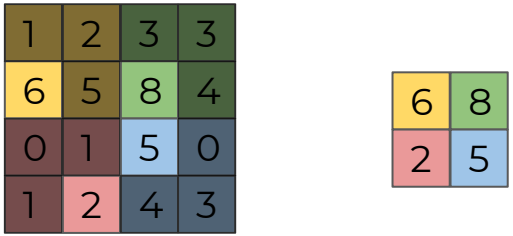
Input = 7x7



Output = 3x3



Convolution
3x3 filters, stride = 2
loses "in-between" features

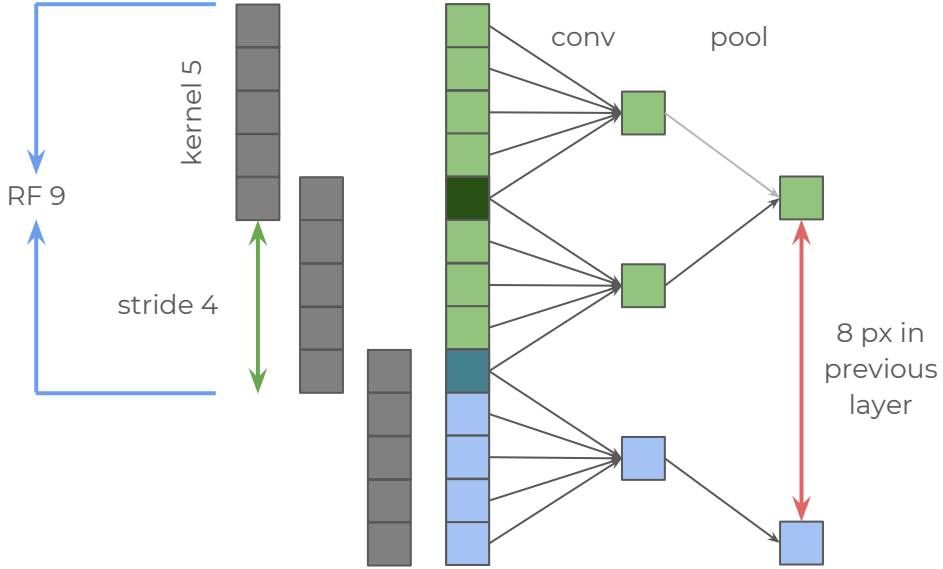


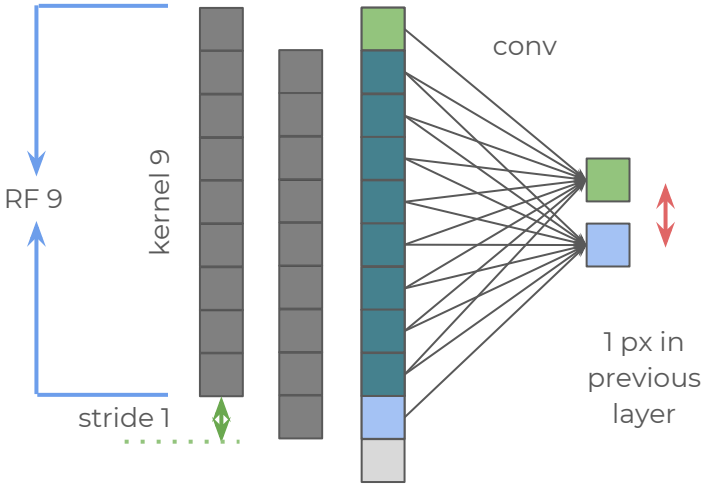
Max-pooling
2x2 filters, stride = 2
loses 75% of activations

- Striding & max-pooling !!

Increase spatial resolution

- Possible solution: stride = 1 & remove pooling
  - Requires larger kernels for same receptive field

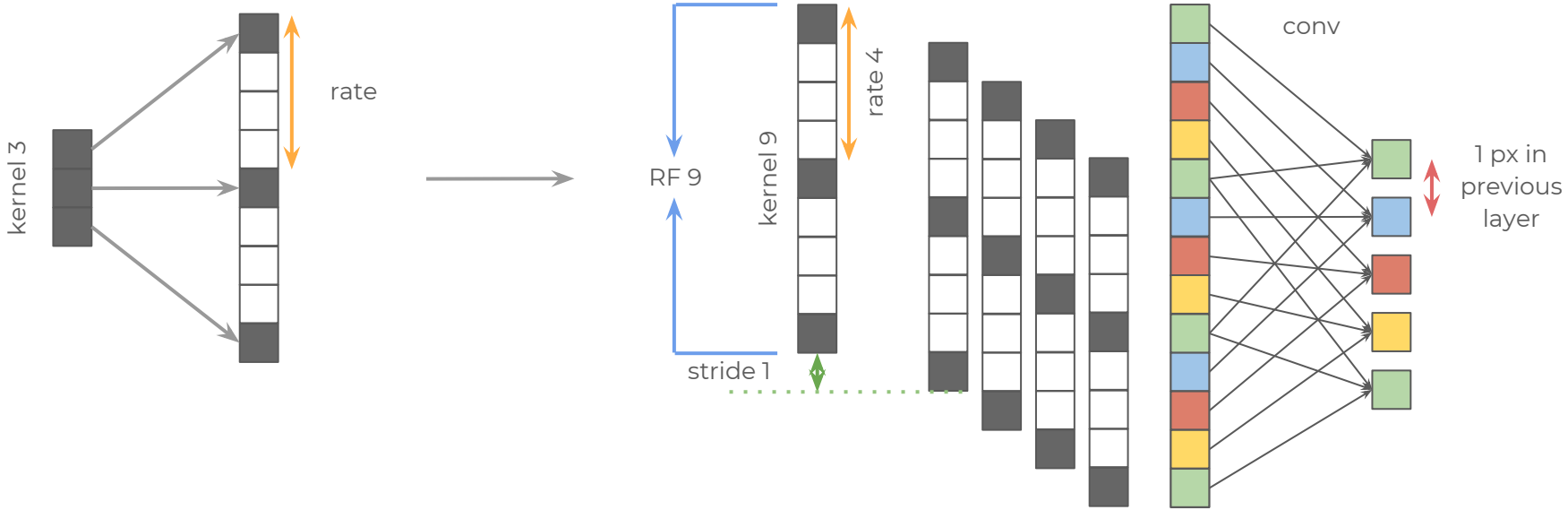




- Computationally expensive & possible overfitting

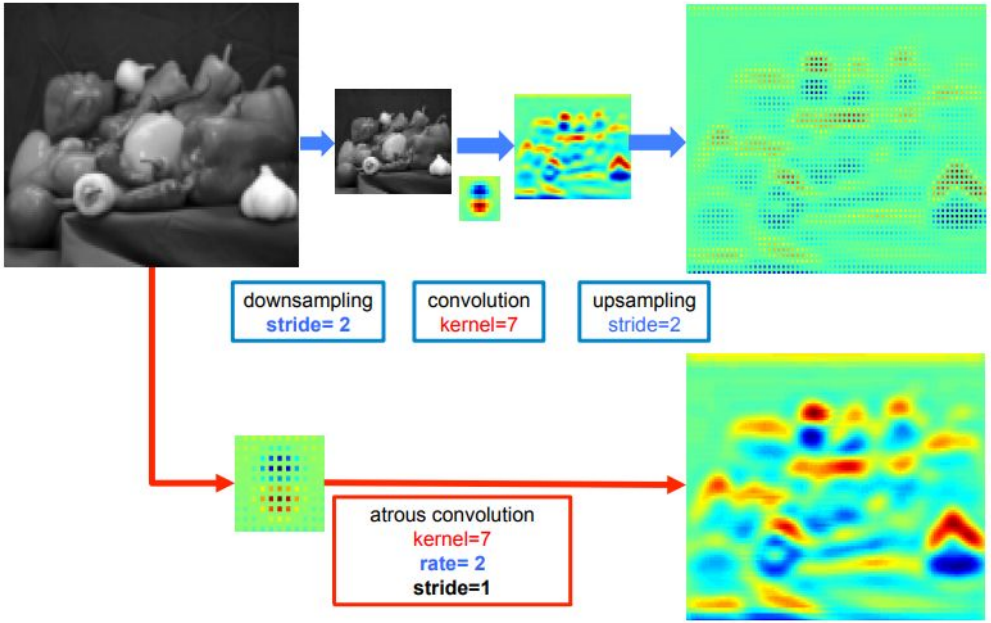
Increase spatial resolution

- Hole algorithm (atrous, dilated convolution)
  - Large, sparse kernel (zeros in between values) = upsampled small kernel
  - Complexity ~ # nonzero elements
  - Arbitrary size receptive field



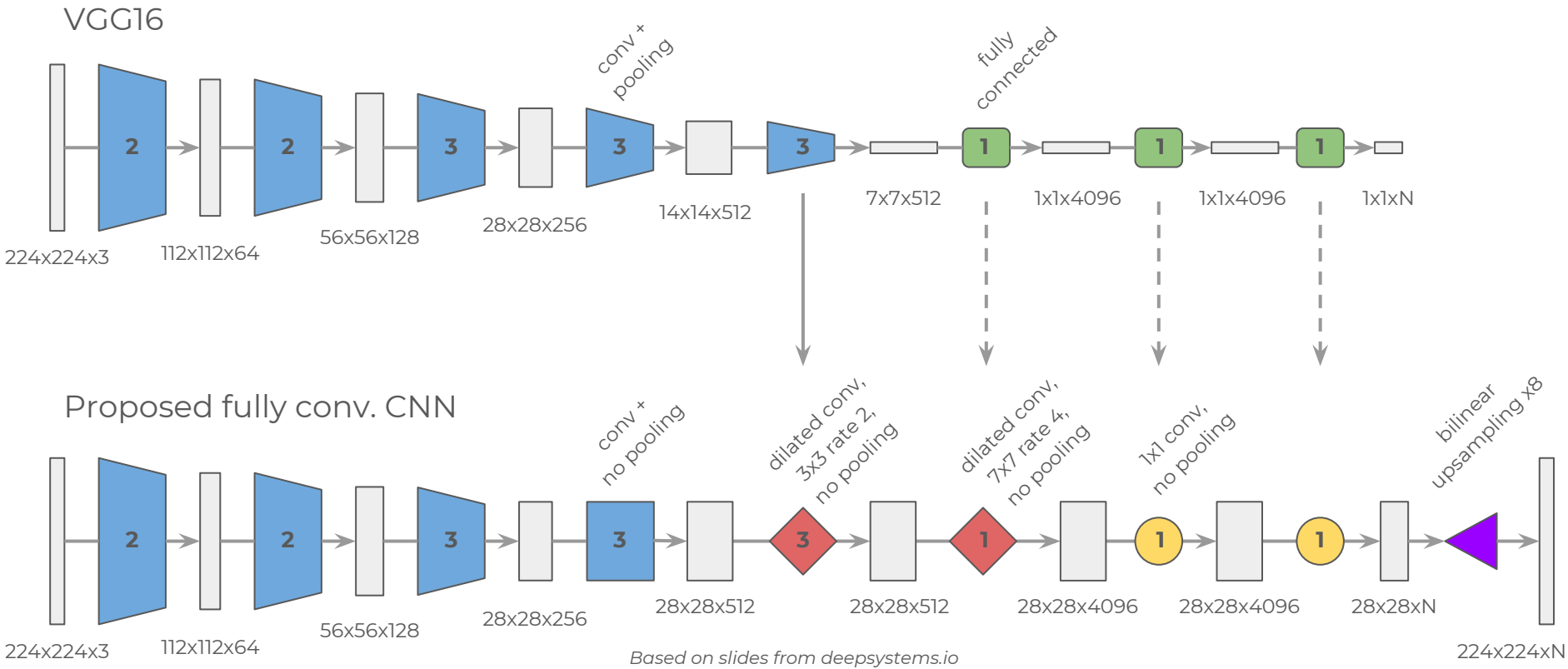
Increase spatial resolution

- Example in 2D [2]:



[2] Chen et al, *DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs*. arXiv:1606.00915v2

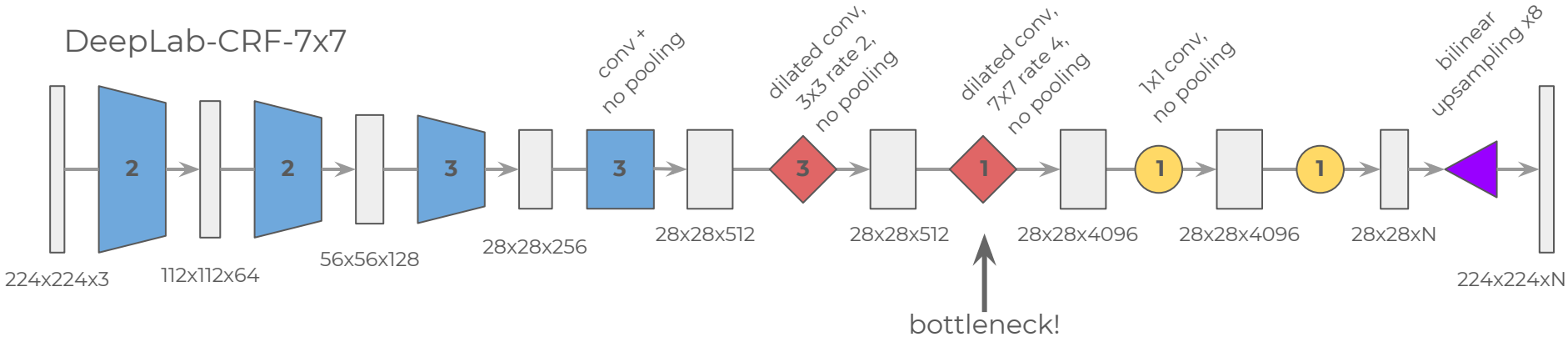
Adapting CNN for segmentation



Based on slides from deepsystems.io

Controlling receptive field size

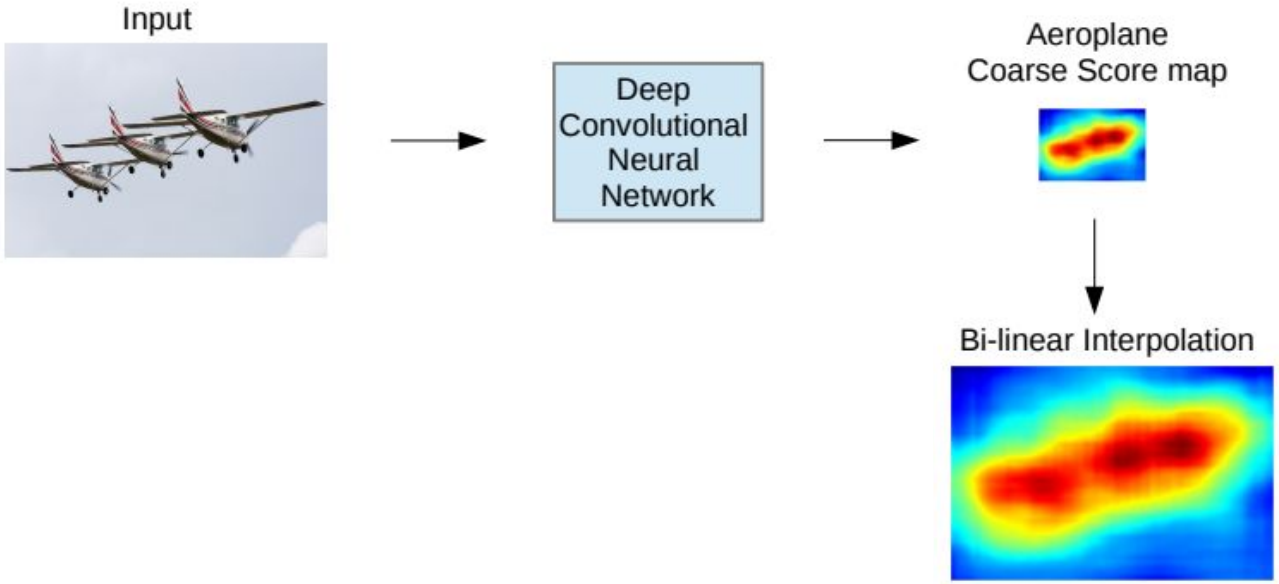
DeepLab-CRF-7x7



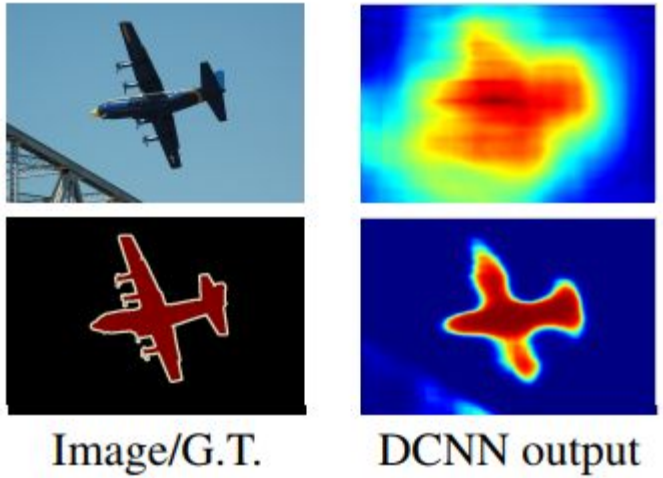
Further adjustments in 1st FC layer:

- DeepLab-CRF kernel = 4, rate = 4, 4096 filters
- DeepLab-CRF-4x4 kernel = 4, rate = 8, 4096 filters
- **DeepLab-CRF-LargeFOV** kernel = 3, rate = 12, 1024 filters same performance as 7x7!

Dense labeling summary



Localization challenge



Score map (before softmax)

Belief map (after softmax)

Object is detected
Outline is inaccurate

Conditional Random Fields

- A set of variables X (one per pixel)
- Image I
- We look for maximum a posteriori (MAP):

$$x^* = \arg\max_x P(x|I)$$

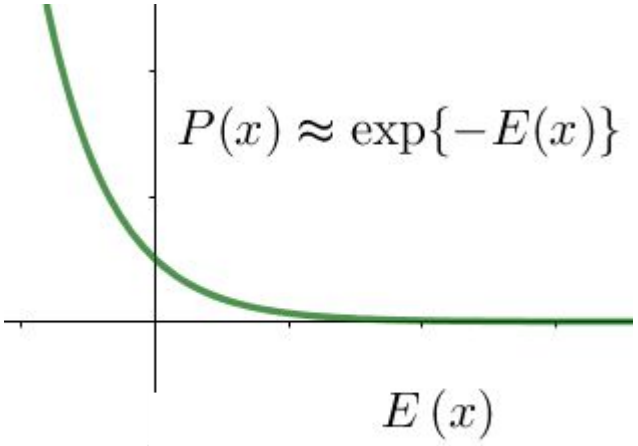
- Gibbs distribution:

$$P(X|I) \approx \exp\{-E(X|I)\}$$

[6] Krähenbühl and Koltun, *Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials*. arXiv:1210.5644v1

Fully-Connected CRFs

$$P(X|I) \approx \exp\{-E(X|I)\}$$



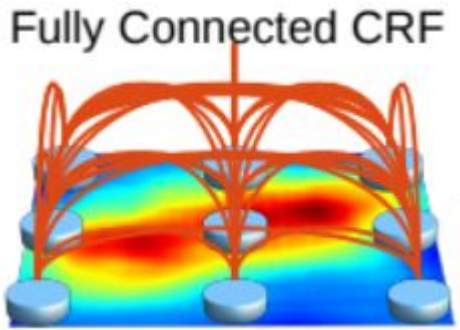
- (Gibbs) Energy: $E(x) = \sum_i \phi(x_i) + \sum_{i<j} \psi(x_i, x_j)$

Gibbs Energy

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

- Unary term $\phi(x_i)$
  - CNN output

- Pairwise term $\psi(x_i, x_j)$
  - Gaussian kernel for appearance $k^{(1)}(\mathbf{f}_i, \mathbf{f}_j)$
  - $+$
  - Gaussian kernel for smoothness $k^{(2)}(\mathbf{f}_i, \mathbf{f}_j)$



Mean Field Approximation

- Computing $P(X)$ is hard
- Choose family of distribution such that:

$$Q(X) = \prod_i Q_i(X_i)$$

- Find $Q(X)$ that approximates $P(X)$ best

  => Minimize KL-Divergence $\mathcal{KL}(Q||P)$

Update rule

- Derived from KL-Divergence:

$$Q_i(x_i = l) = \frac{1}{Z_i} \exp\{-\phi(x_i) - \sum_{l'} [l \neq l'] \sum_{m=1}^{2} w^{(m)} \underbrace{\sum_{i \neq j} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l')}_{[G^{(m)} \otimes Q(l')](f_i) - Q_i(l')}\}$$

Gaussian Approximation

$$Q_i = \ldots [G^{(m)} \otimes Q(l')](f_i) - Q_i(l')$$

Issue:

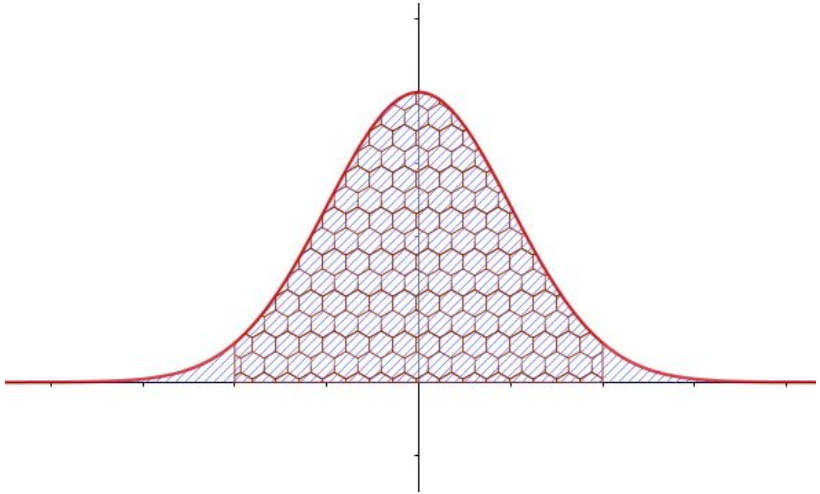
- Kernel size == Image size

$$\mathcal{O}(|I|^2)$$

New runtime:

- Kernel Size == constant

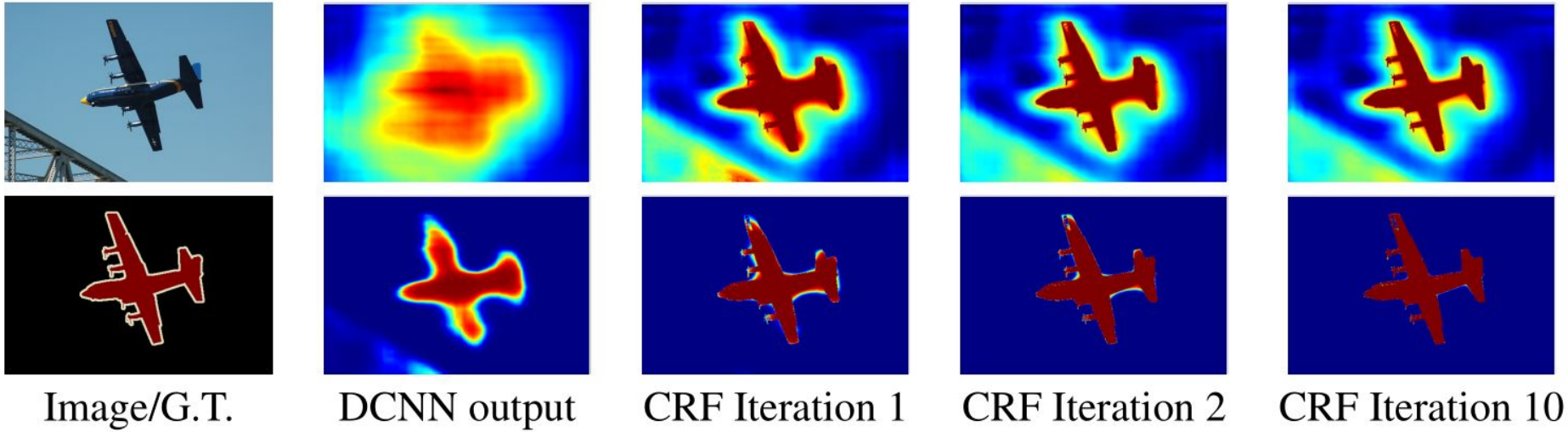
$$\mathcal{O}(|I|)$$



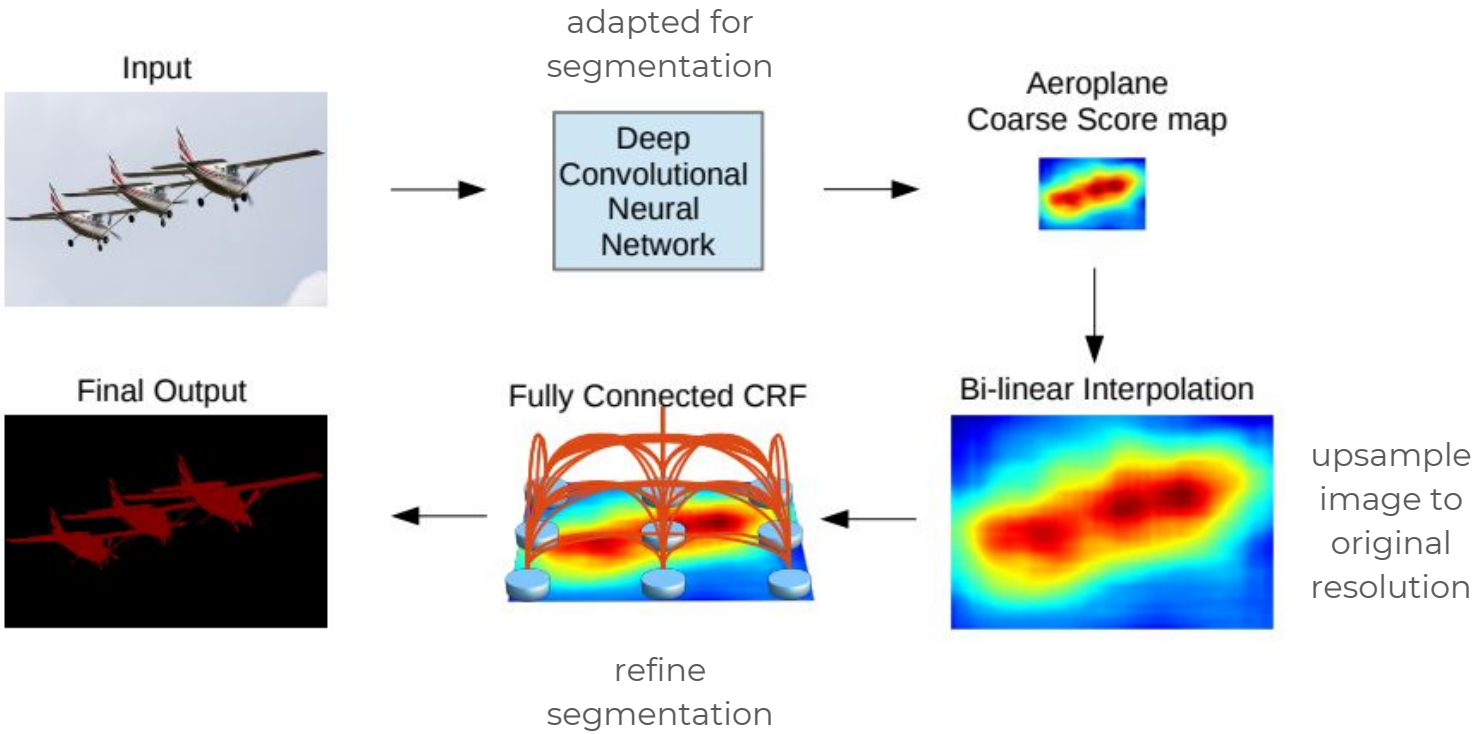
CRF Summary

- Post-processing step
- Fully connected CRF consists of
  - Unary terms (CNN outputs)
  - Pairwise terms (encode global context)
- Mean Field Approximation for $Q(X)$
- Iterate until convergence

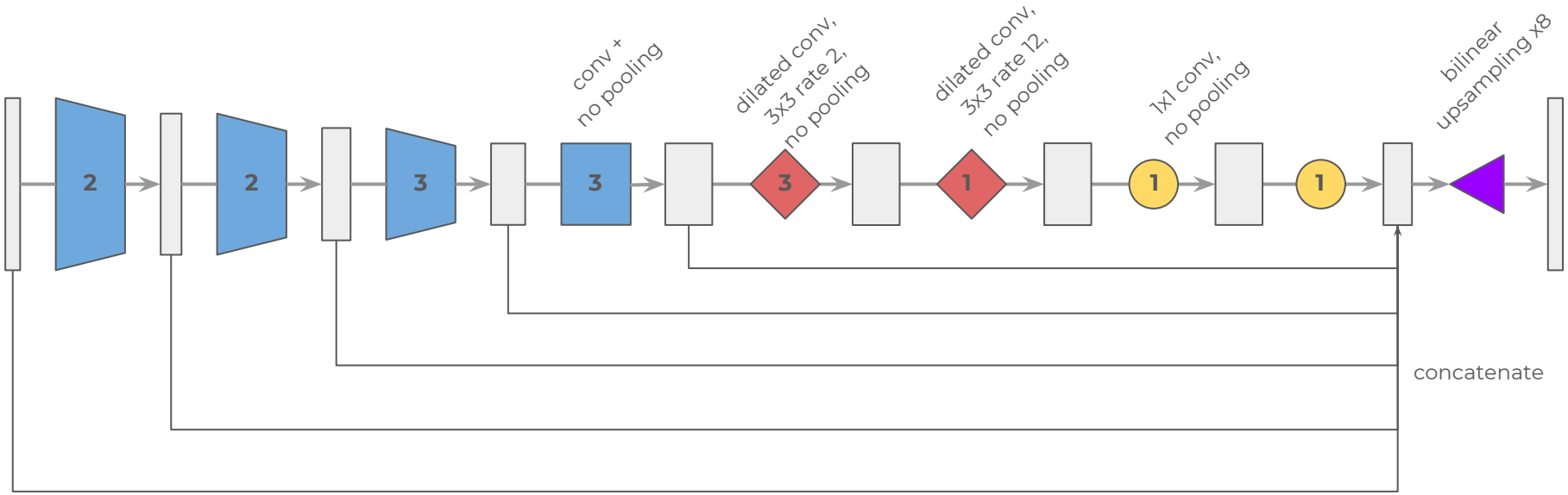
CRF Results


Image/G.T.
DCNN output
CRF Iteration 1
CRF Iteration 2
CRF Iteration 10

Complete model (DeepLab v1)



Extension: Multiscale-prediction



Experiments

Method	mean IOU (%)
DeepLab	59.80
DeepLab-CRF	63.74
DeepLab-MSc	61.30
DeepLab-MSc-CRF	65.21
DeepLab-7x7	64.38
DeepLab-CRF-7x7	67.64
DeepLab-LargeFOV	62.25
DeepLab-CRF-LargeFOV	67.64
DeepLab-MSc-LargeFOV	64.21
DeepLab-MSc-CRF-LargeFOV	68.70

PASCAL VOC 2012 'val'
(trained in augmented 'train')

Method	mean IOU (%)
MSRA-CFM [3]	61.8
FCN-8s [4]	62.2
TTI-Zoomout-16 [5]	64.4
DeepLab-CRF	66.4
DeepLab-MSc-CRF	67.1
DeepLab-CRF-7x7	70.3
DeepLab-CRF-LargeFOV	70.3
DeepLab-MSc-CRF-LargeFOV	71.6

PASCAL VOC 2012 'test'
(trained in augmented 'trainval')

[3] Dai, He, and Sun. *Convolutional feature masking for joint object and stuff segmentation*.arXiv:1412.1283v4
[4] Long, Shelhamer, and Darrel, *Fully convolutional networks for semantic segmentation*. arXiv:1411.4038v2
[5] Mostajabi, Yadollahpour and Shakhnarovich, *Feedforward semantic segmentation with zoom-out features*. arXiv:1412.0774v1

Experiments

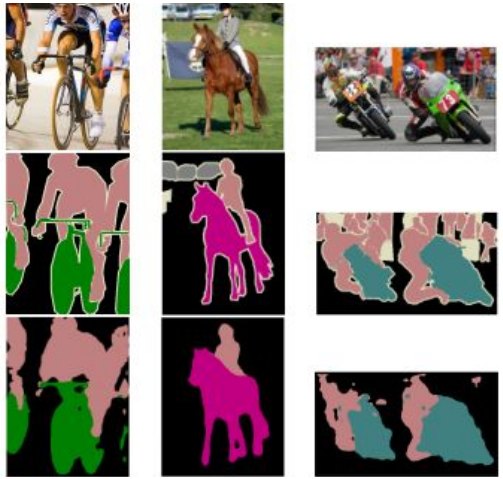
- Field of view
  - Modifications to first FC layer

Method	kernel size	input stride	receptive field	# parameters	mean IOU (%)	Training speed (img/sec)
DeepLab-CRF-7x7	7×7	4	224	134.3M	67.64	1.44
DeepLab-CRF	4×4	4	128	65.1M	63.74	2.90
DeepLab-CRF-4x4	4×4	8	224	65.1M	67.14	2.90
DeepLab-CRF-LargeFOV	3×3	12	224	20.5M	67.64	4.84

Training in PASCAL VOC 2012 'val'

Experiments

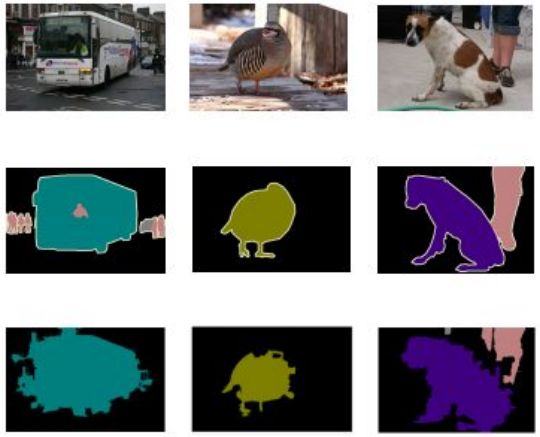
- CRF



image

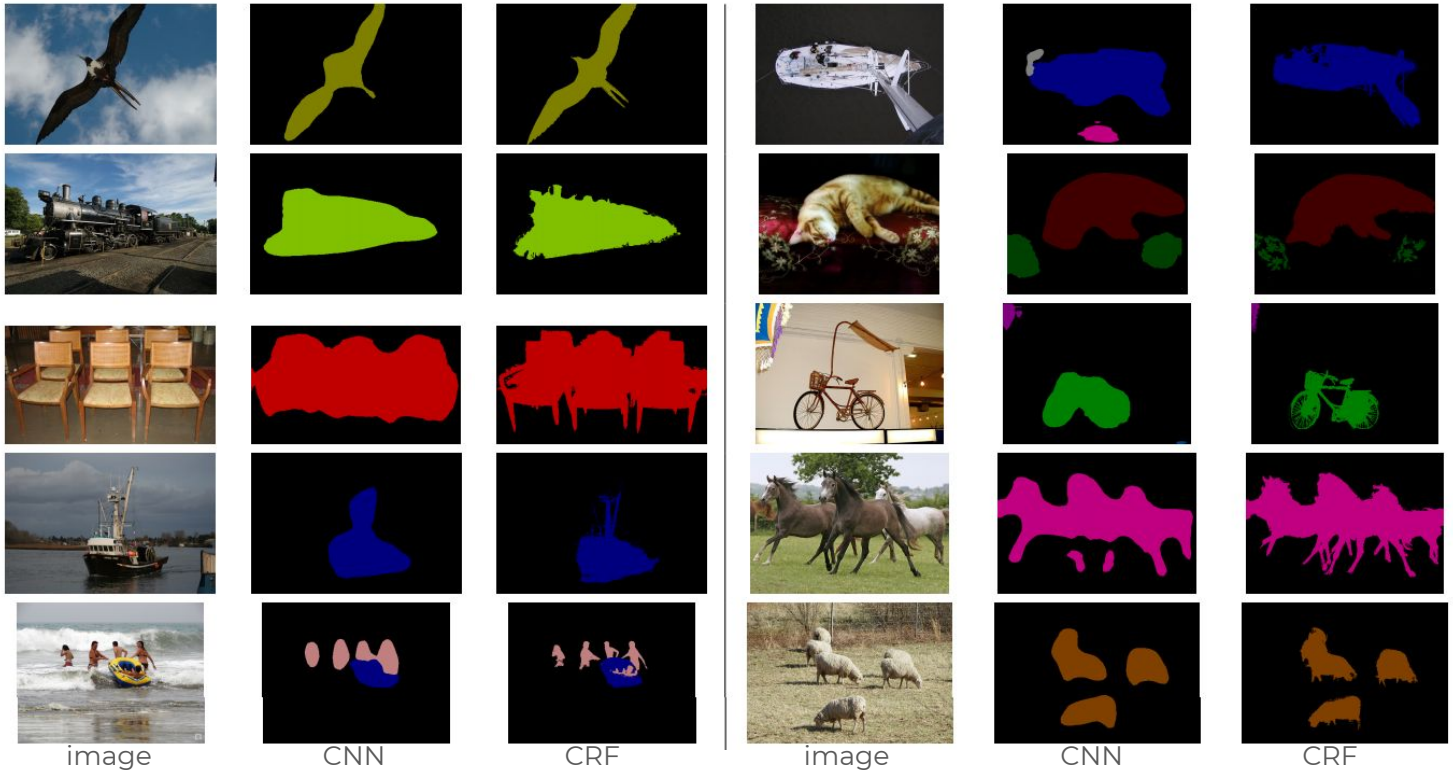
ground-truth

DeepLab-CRF



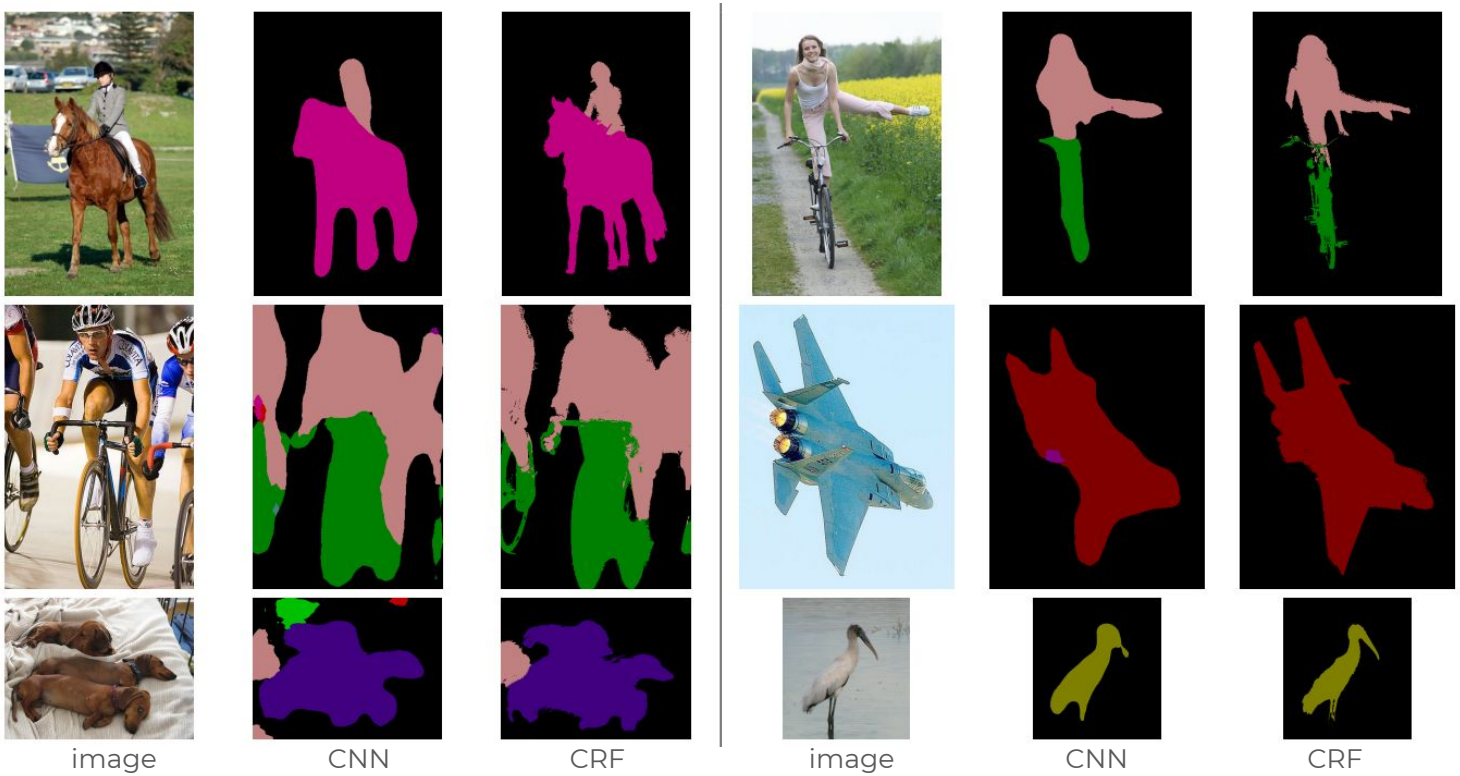
Experiments

- CRF: Success



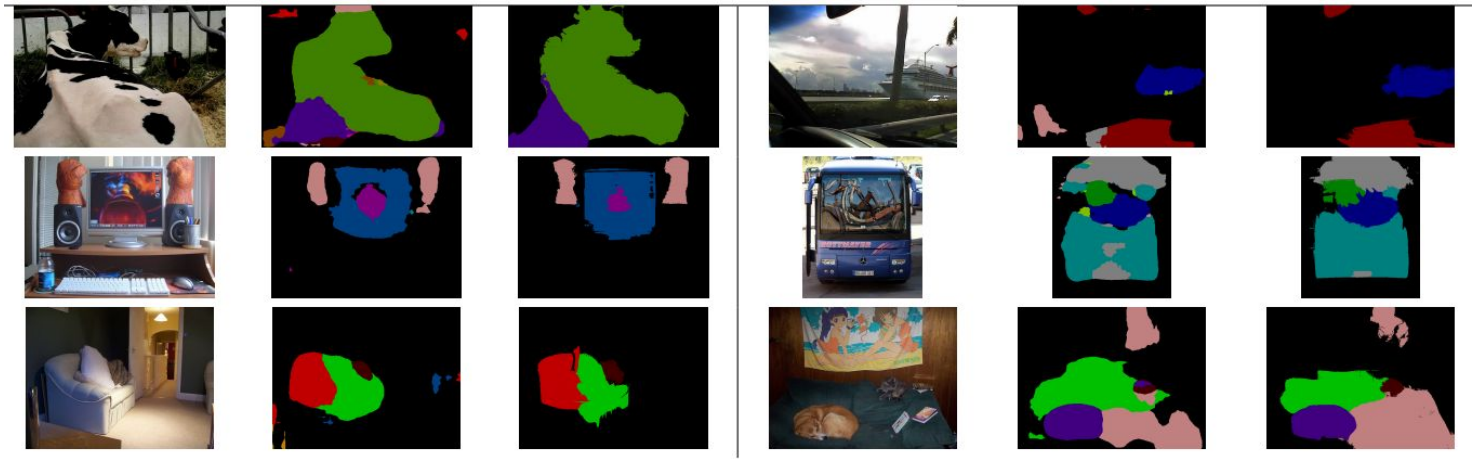
Experiments

- CRF: Success



Experiments

- CRF: Failures



Experiments

- Labeling IoU

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
MSRA-CFM	-	75.7	26.7	69.5	48.8	65.6	81.0	69.2	73.3	30.0	68.7	51.5	69.1	68.1	71.7	67.5	50.4	66.5	44.4	58.9	53.5	61.8
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
TTI-Zoomout-16	89.8	81.9	35.1	78.2	57.4	56.5	80.5	74.0	79.8	22.4	69.6	53.7	74.0	76.0	76.6	68.8	44.3	70.2	40.2	68.9	55.3	64.4
DeepLab-CRF	92.1	78.4	33.1	78.2	55.6	65.3	81.3	75.5	78.6	25.3	69.2	52.7	75.2	69.0	79.1	77.6	54.7	78.3	45.1	73.3	56.2	66.4
DeepLab-MSc-CRF	92.6	80.4	36.8	77.4	55.2	66.4	81.5	77.5	78.9	27.1	68.2	52.7	74.3	69.6	79.4	79.0	56.9	78.8	45.2	72.7	59.3	67.1
DeepLab-CRF-7x7	92.8	83.9	36.6	77.5	58.4	**68.0**	84.6	**79.7**	83.1	29.5	**74.6**	59.3	78.9	76.0	82.1	80.6	**60.3**	81.7	49.2	**78.0**	60.7	70.3
DeepLab-CRF-LargeFOV	92.6	83.5	36.6	**82.5**	62.3	66.5	**85.4**	78.5	**83.7**	30.4	72.9	**60.4**	78.5	75.5	82.1	79.7	58.2	82.0	48.8	73.7	63.3	70.3
DeepLab-MSc-CRF-LargeFOV	**93.1**	**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**

PASCAL VOC 2012 ‘test’ (trained with ‘trainval’)

Comments

- Strengths
  - Template to adapt classification CNNs for image segmentation
  - Successful incorporation of methods from other disciplines
  - Experiments with each individual modification allows to compare impact

- Weaknesses
  - Two processes: CNN + postprocessing
  - Paper is not self contained (need references *a lot*)
  - Key concepts are not well explained (better in Deeplab v2 [2])

[2] Chen et al, *DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs*. arXiv:1606.00915v2

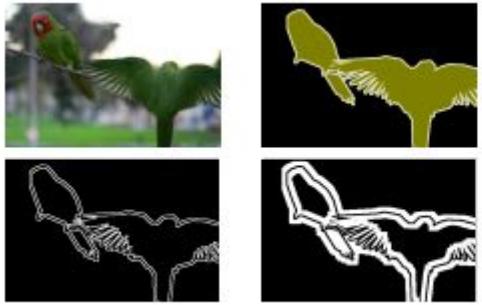
Thanks!

Questions?

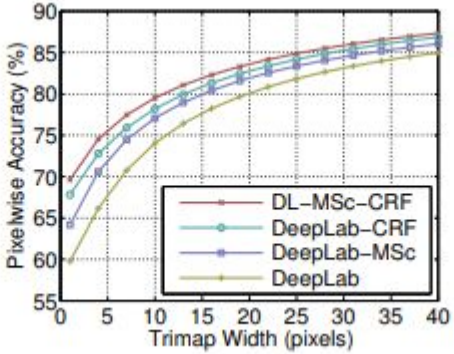
Appendix: Experiments

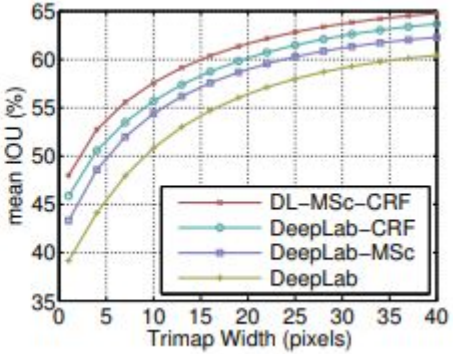
- Object boundaries
  - Void label (regularly seen in boundaries)
  - Mean IoU of pixels in narrow band (trimap)

image ground-truth



trimap 2px trimap 10px





Appendix: Experiments

- Multi-scale features
  - Modify kernel size of first FC layer and rate

DeepLab

DeepLab-MSc