Fully Convolutional Networks

Jun Gao

Outline

- Movitation
- Semantic Segmentation
  - Fully Convolutional Networks
  - Deconvolution Networks
- Contour Segmentation
- Instance Segmentation
- Liver Segmentation

Motivation

- Fully connected layers can be viewed as convolutions
- Trained end-to-end, pixels-to-pixels on whole images
- Take input of arbitrary size

Fully Convolutional Networks for Semantic Segmentatoin

Evan Shelhamer , Jonathan Long, and Trevor DarrellIn
UC Berkely
In Computer Vision and Pattern Recognition(CVPR), 2015

Pixel-wise Prediction

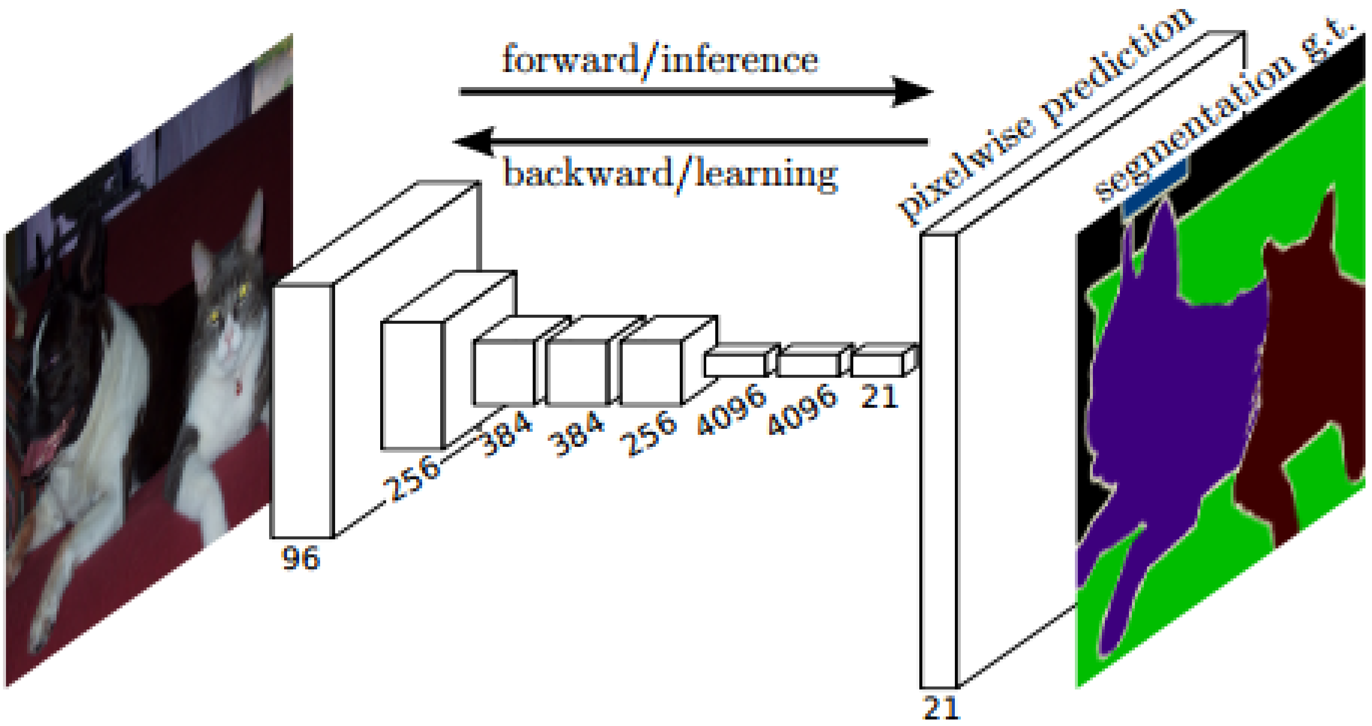
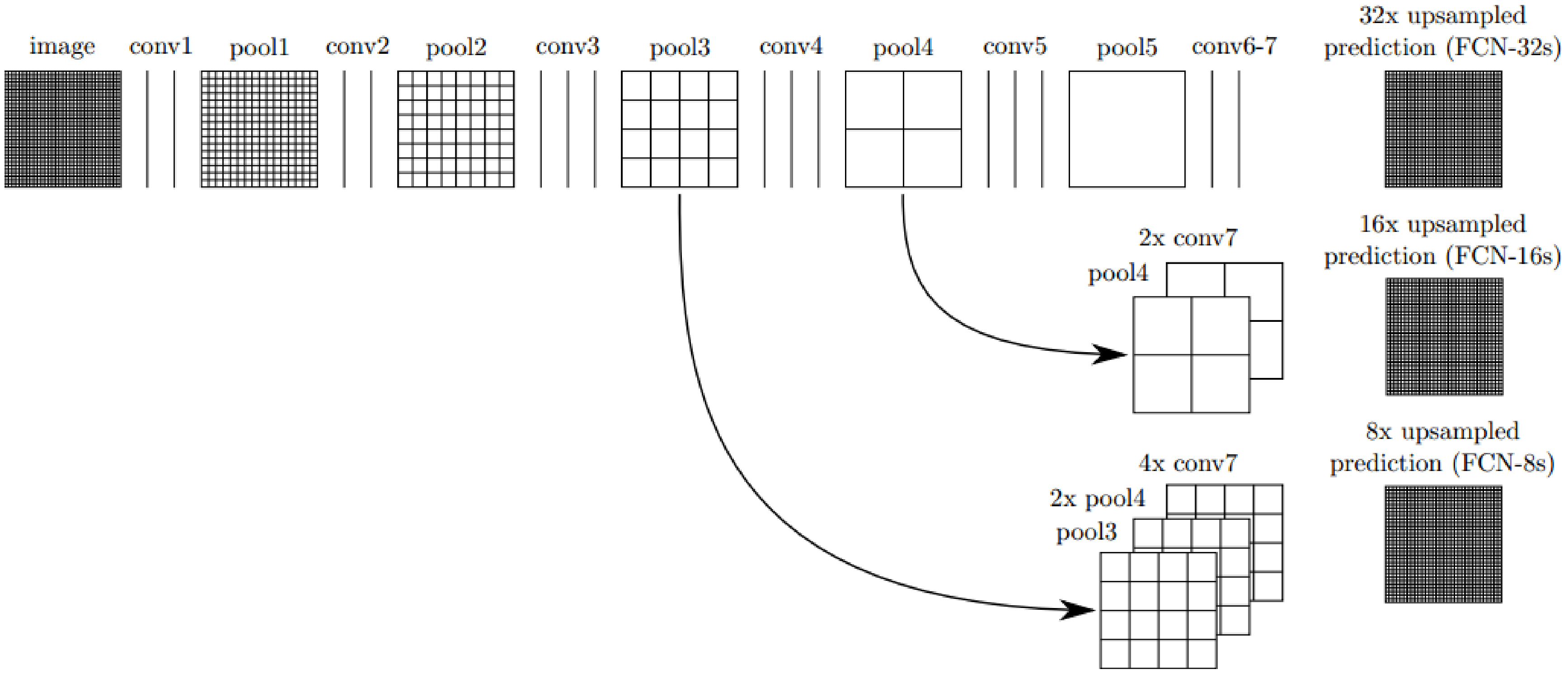


Fig. 1. Fully convolutional networks can efficiently learn to make dense predictions for per-pixel tasks like semantic segmentation.

Match resolutions by upsampling

$$y_{ij} = \sum_{\alpha,\beta=0}^{1} |1-\alpha-\{i/f\}|\,|1-\beta-\{i/j\}|\, x_{\lfloor i/f\rfloor+\alpha,\lfloor j/f\rfloor+\beta}$$



Results

	mean IU VOC2011 test	mean IU VOC2012 test	inference time
R-CNN [5]	47.9	-	-
SDS [14]	52.6	51.6	$\sim$ 50 s
FCN-8s	**67.5**	**67.2**	**$\sim$ 100 ms**

Limitations

- Fixed-size receptive filed

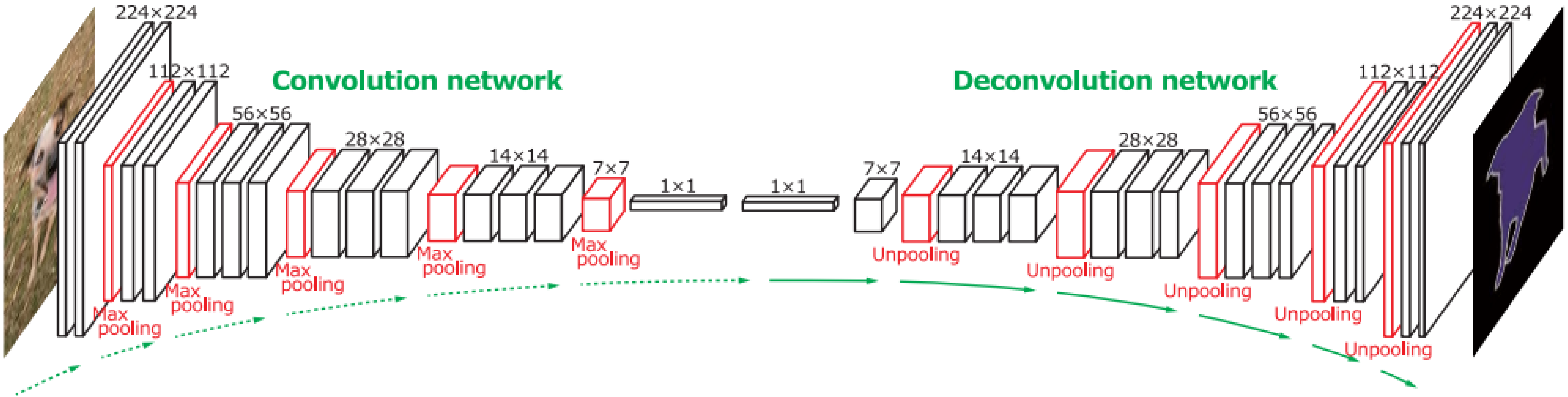
(a) Inconsistent labels due to large object size (b) Missing labels due to small object size

- Lost Detailed Structures

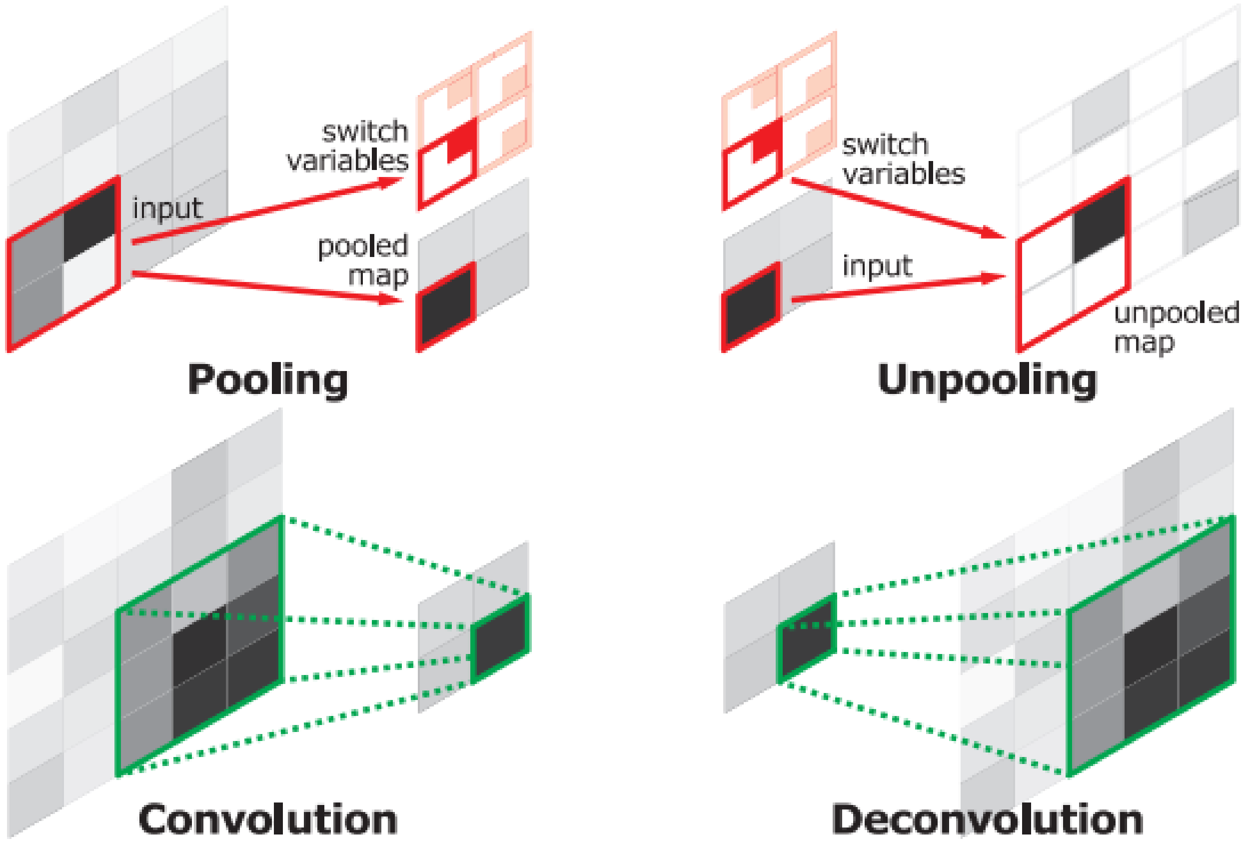
Learning Deconvolution Networks for Semantic Segmentation

Hyeonwoo Noh, Seunghoon Hong, Bohyung Han
POSTECH, Korea
In International Conference on Computer Vision(ICCV), 2015

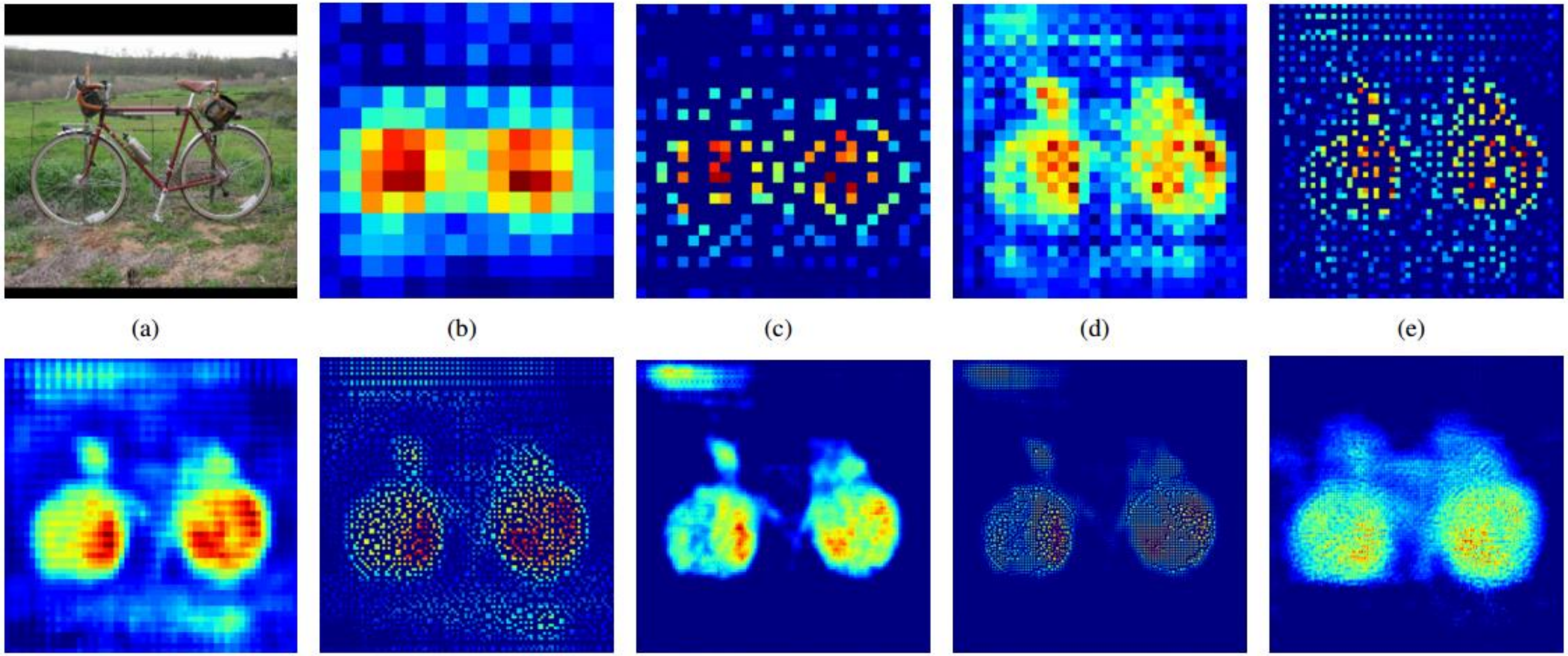
Network Architecture



Unpooling && Deconvolution

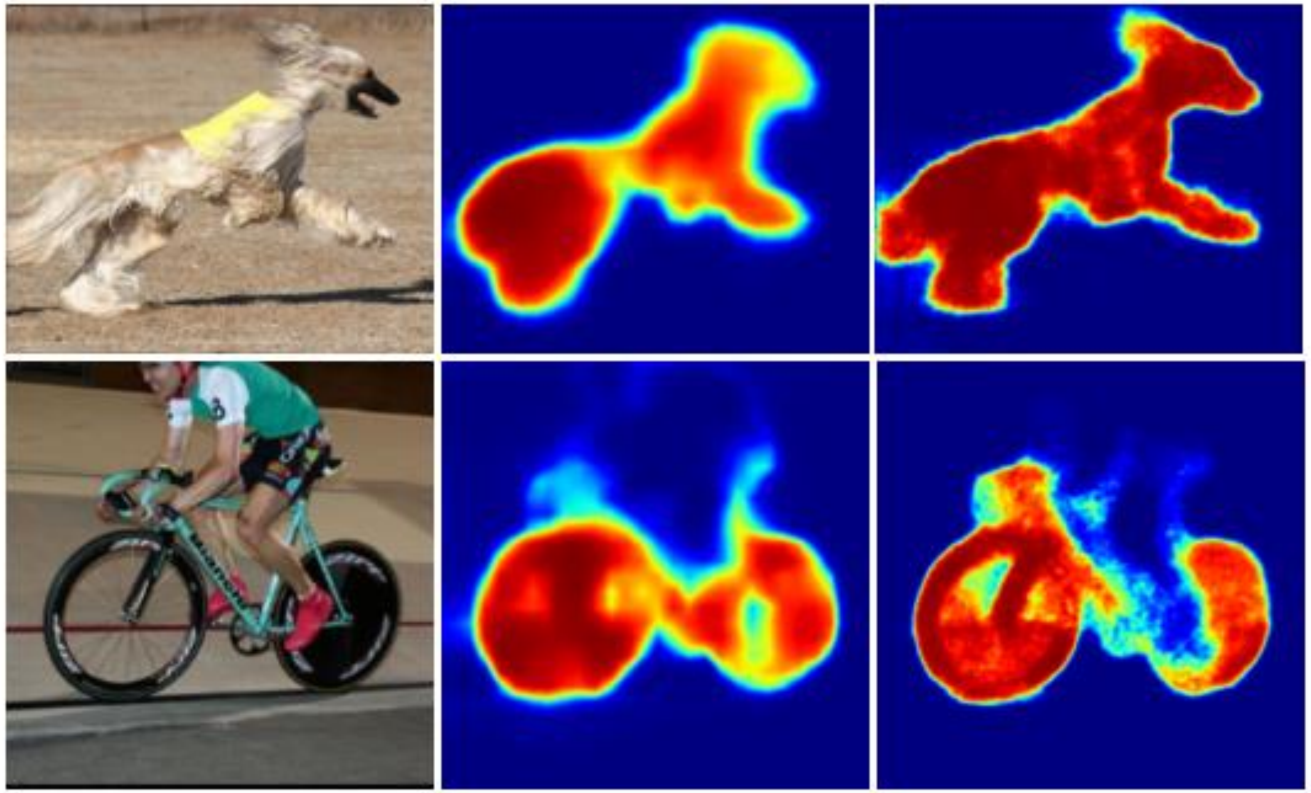


Visualization of deconvolution



(f) (g) (h) (i) (j)

Comparison with FCN



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

Training && inference

- Two stage training
  - Train the network with easy examples first and fine-tune the trained network with more challenging examples later.

- Aggregating instance-wise segmentation maps
  - Using edge-box to generate object proposals

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

or

$$P(x, y, c) = \sum_i G_i(x, y, c), \quad \forall i.$$

Results

Table 1. Evaluation results on PASCAL VOC 2012 test set. (Asterisk (∗) denotes the algorithms trained with additional data.)

Method	bkg	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mean
Hypercolumn [10]	88.9	68.4	27.2	68.2	47.6	61.7	76.9	72.1	71.1	24.3	59.3	44.8	62.7	59.4	73.5	70.6	52.0	63.0	38.1	60.0	54.1	59.2
MSRA-CFM [3]	87.7	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
FCN8s [17]	91.2	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 [18]	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 [1]	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
DeconvNet	92.7	85.9	42.6	78.9	62.5	66.6	87.4	77.8	79.5	26.3	73.4	60.2	70.8	76.5	79.6	77.7	58.2	77.4	52.9	75.2	59.8	69.6
DeconvNet+CRF	92.9	87.8	41.9	80.6	63.9	67.3	**88.1**	78.4	81.3	25.9	73.7	61.2	72.0	77.0	79.9	78.7	59.5	78.3	**55.0**	75.2	61.5	70.5
EDeconvNet	92.9	88.4	39.7	79.0	63.0	67.7	87.1	**81.5**	84.4	27.8	76.1	61.2	78.0	79.3	83.1	79.3	58.0	82.5	52.3	80.1	64.0	71.7
EDeconvNet+CRF	93.1	**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**
* WSSL [19]	93.2	85.3	36.2	**84.8**	61.2	67.5	84.7	81.4	81.0	**30.8**	73.8	53.8	77.5	76.5	82.3	**81.6**	56.3	78.9	52.3	76.6	63.3	70.4
* BoxSup [2]	**93.6**	86.4	35.5	79.7	**65.2**	65.2	84.3	78.5	83.7	30.5	76.2	**62.6**	**79.3**	76.1	82.1	81.3	57.0	78.2	**55.0**	72.5	**68.1**	71.0

Contributions

- Multi-layer deconvolution networks
- Combine instance-wise segmentations for the final semantic segmentation

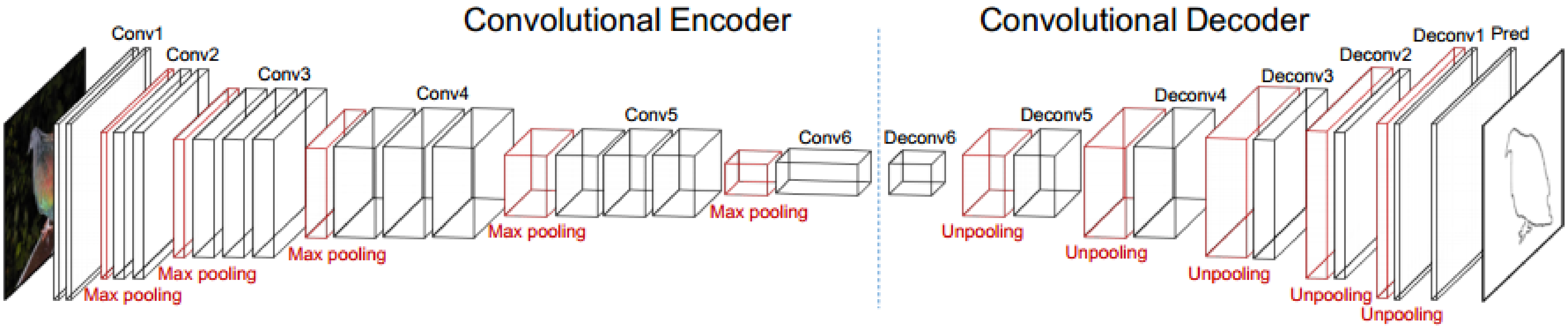
Object Contour Detection with a Fully Convolutional Encoder-Decoder Network

Jimei Yang, Brain Price, Scott Cohen, Honglak Lee, Ming-Hsuan Yang
Adobe Research, University of Michigan, UC Merced
In Computer Vision and Pattern Recognition(CVPR), 2016

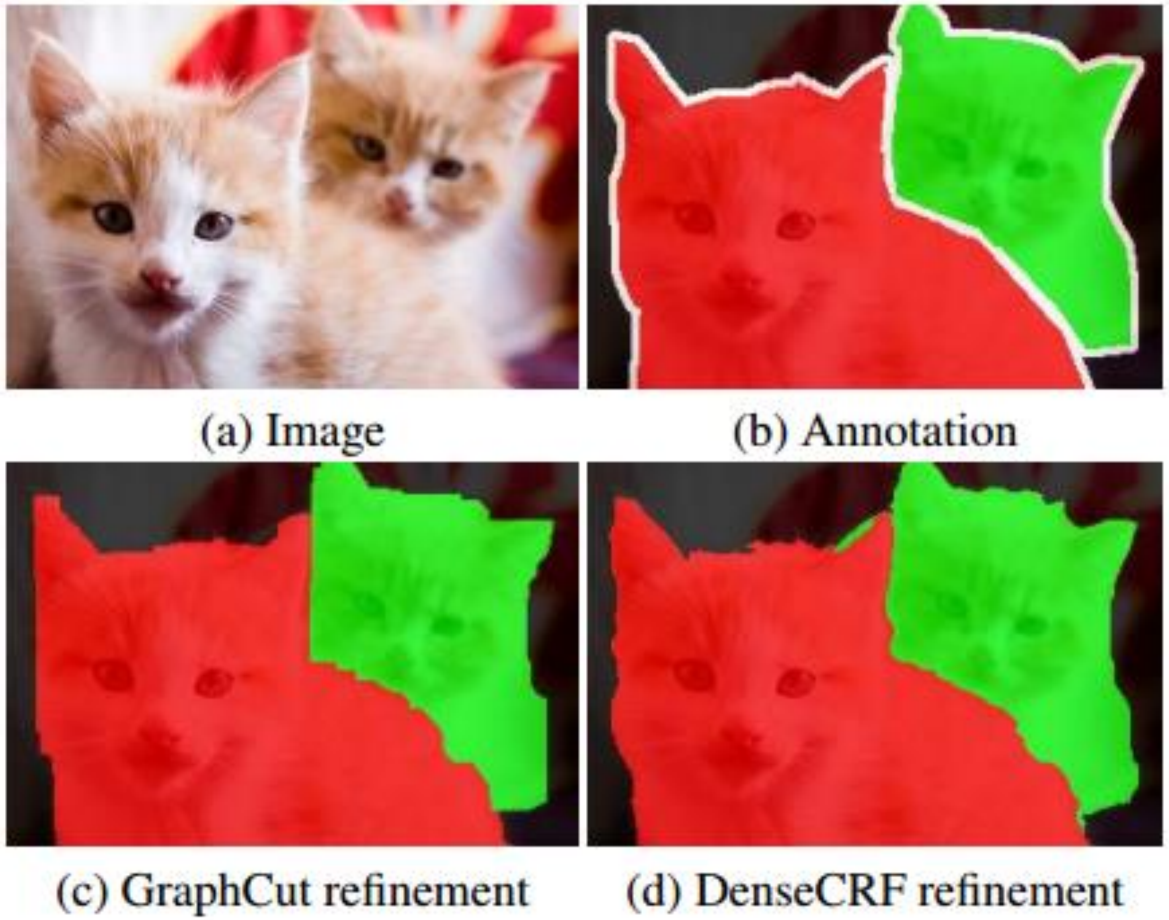
Motivation

- Detect high-level object contour, instead edge. (only foreground objects)
- Object contour detection is an image labeling problem
- Symmetric structures introduce a heavy decoder network which is hard to train with limited samples

Network Architecture



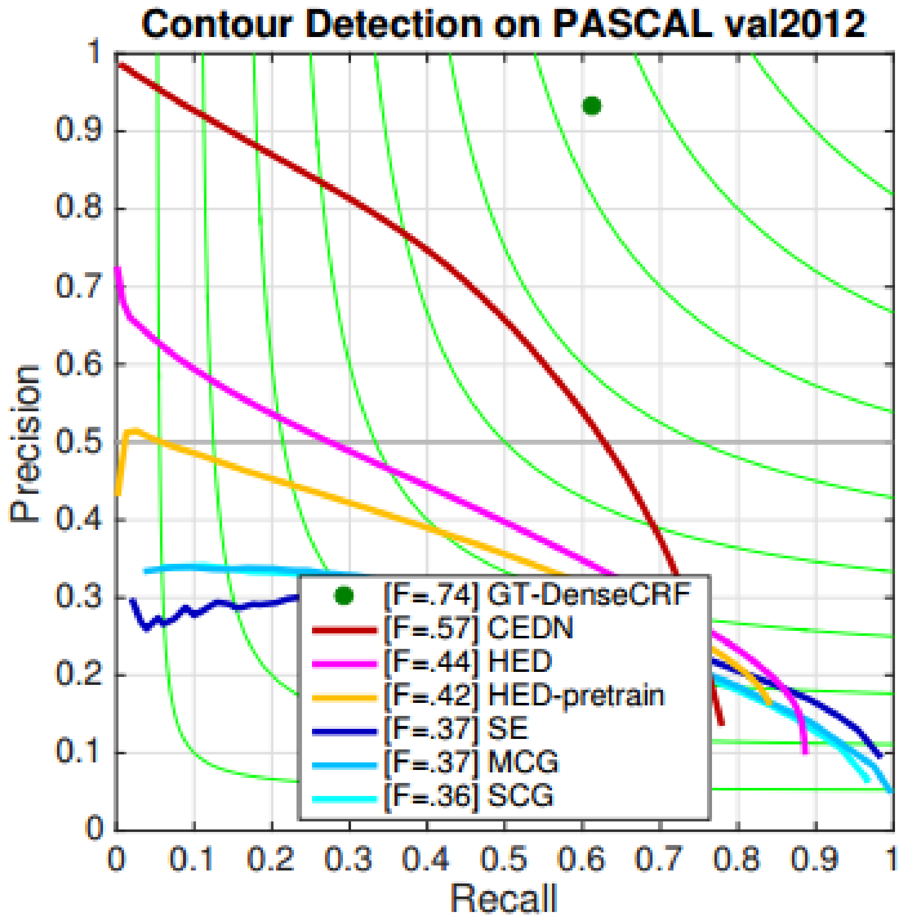
Contour refinement



Training

- Randomly crop 4 224*224*3 patches with their mirrored ones.
- For limited samples
  - Fix the encoder parameters.
  - Penalty for being "contour" is set to be 10 times the penalty for being "non-contour".
- Pixel-wise Logistic loss.

Results



F = 0.57
Upper bound = 0.74

Generalization

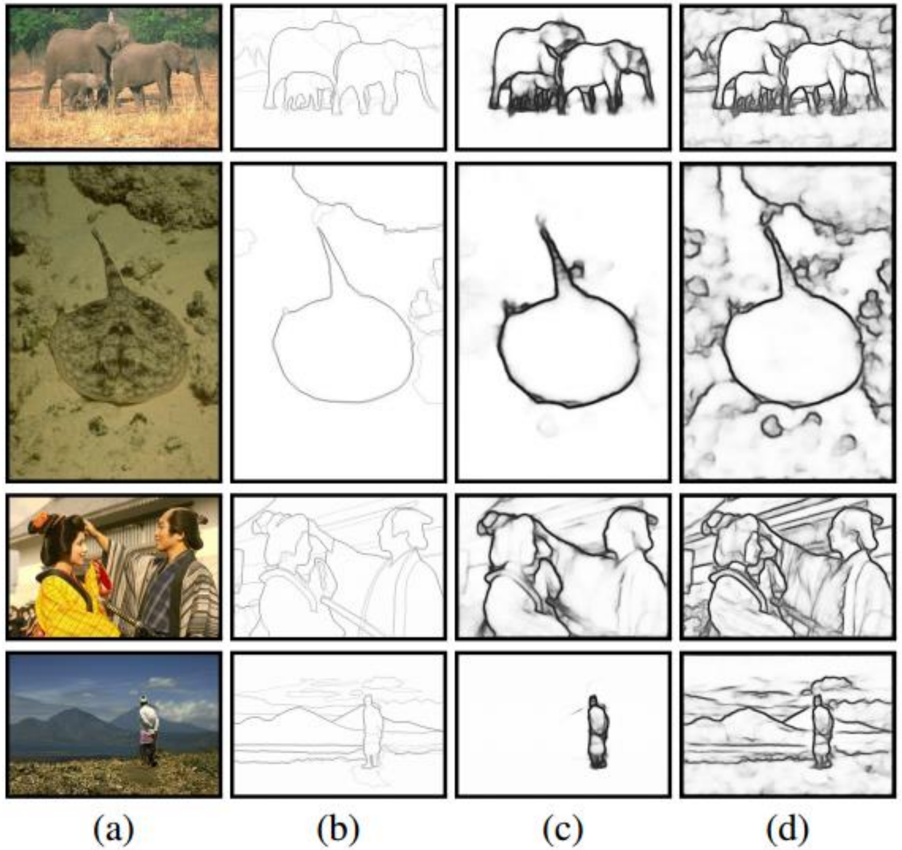


Figure 6. Example results on BSDS500 test set. In each row from left to right we present (a) input image, (b) ground truth contour, (c) contour detection with pretrained CEDN and (d) contour detection with fine-tuned CEDN.

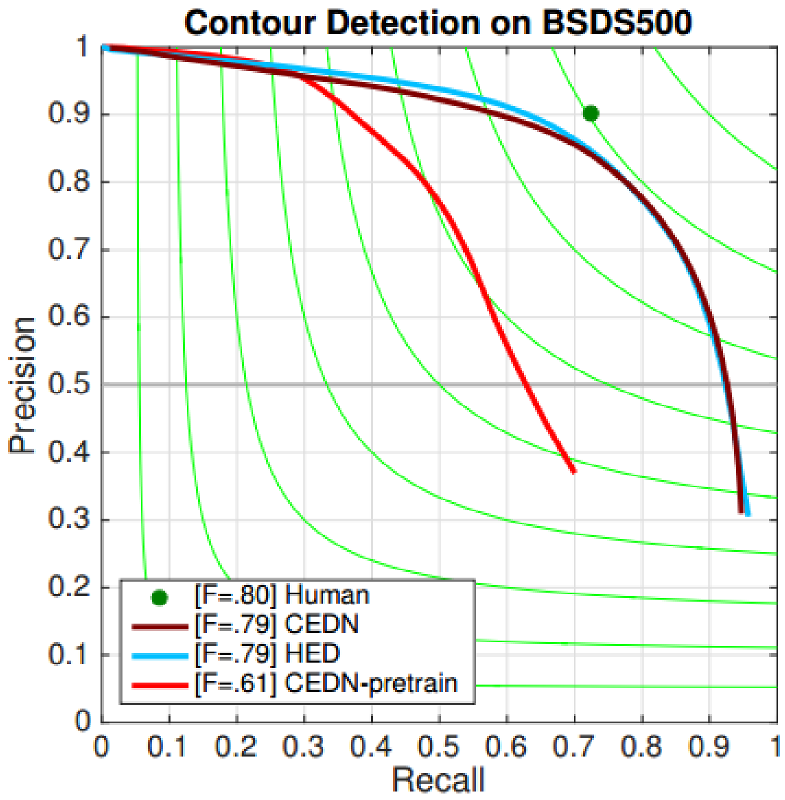
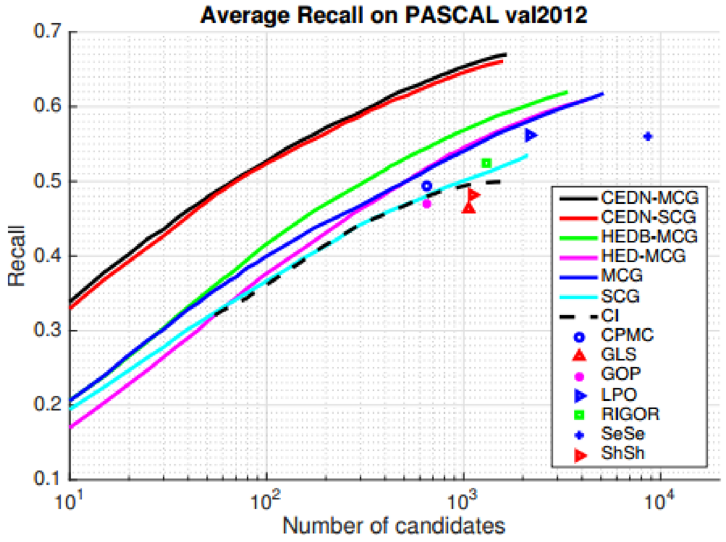
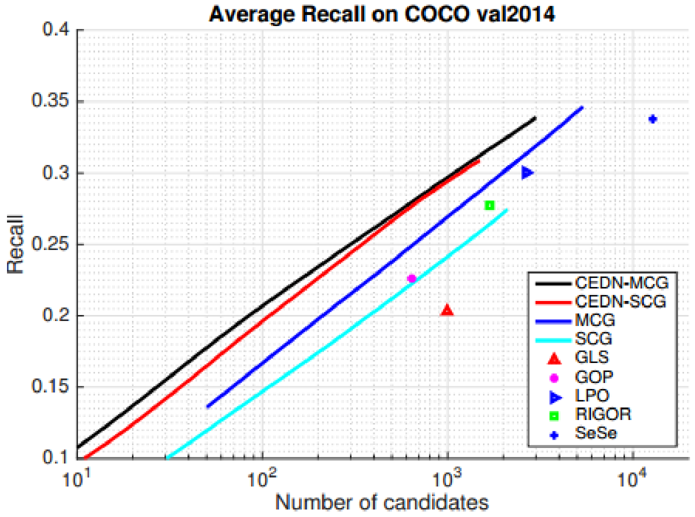
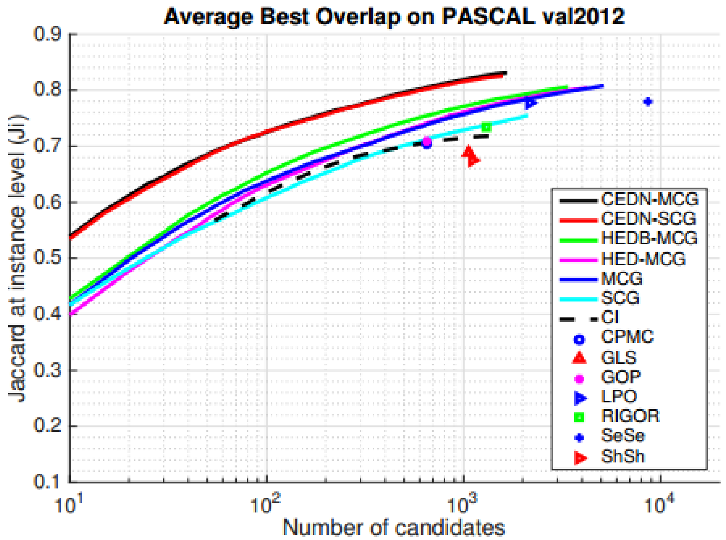


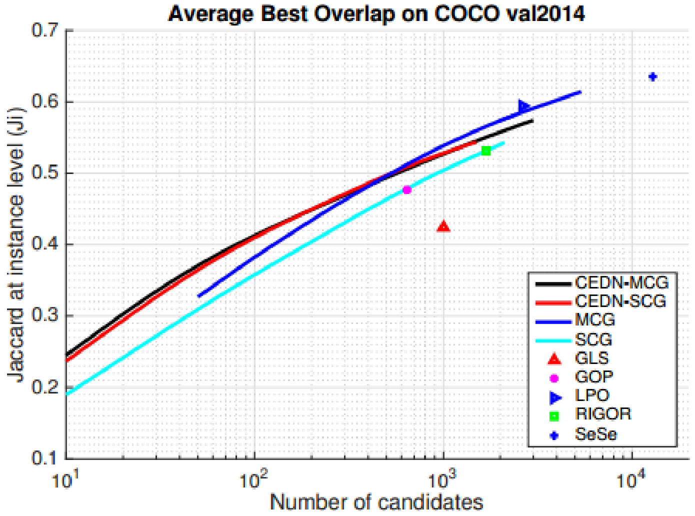
Figure 7. PR curve for contour detection on the BSDS500 set set.

Object proposal generation









Results

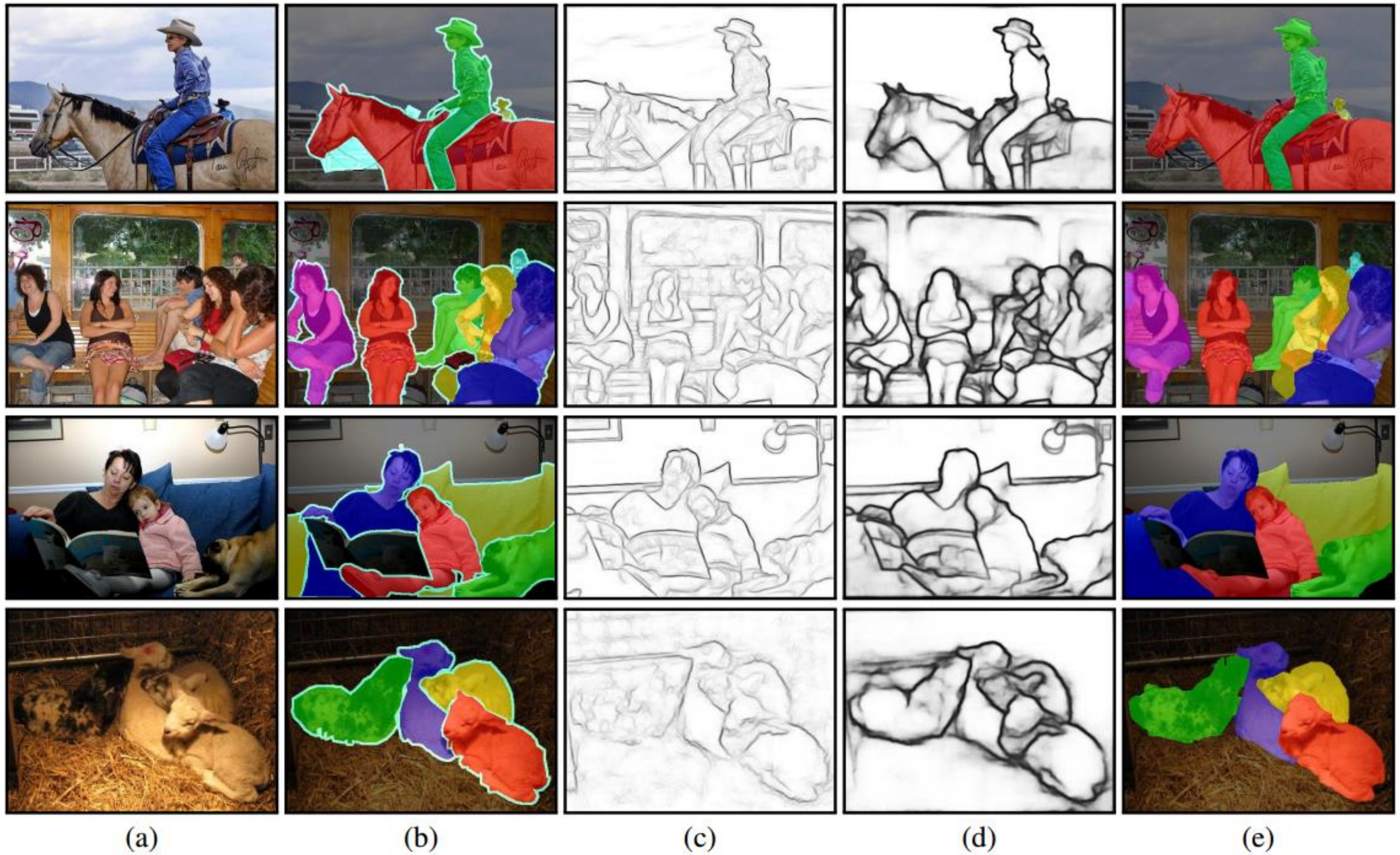


Figure 4. Example results on PASCAL VOC val2012. In each row from left to right we present (a) input image, (b) ground truth annotation, (c) edge detection [13], (d) our object contour detection and (e) our best object proposals.

Contributions

- A simple yet effective fully convolutional encoder-decoder network for object contour detection.
- Fine tune network for edge detection and obtain good result.
- A method to generate accurate object contours from imperfect polygon segmentation annotations.
- Improve results on segmented object proposals.

R-FCN: Object Detection via Region-based Fully Convolutional Networks

Jifeng Dai, Yi Li, Kaiming He, Jian Sun
Microsoft Research, Tsinghua University
In Arxiv.org. 21 June 2016.

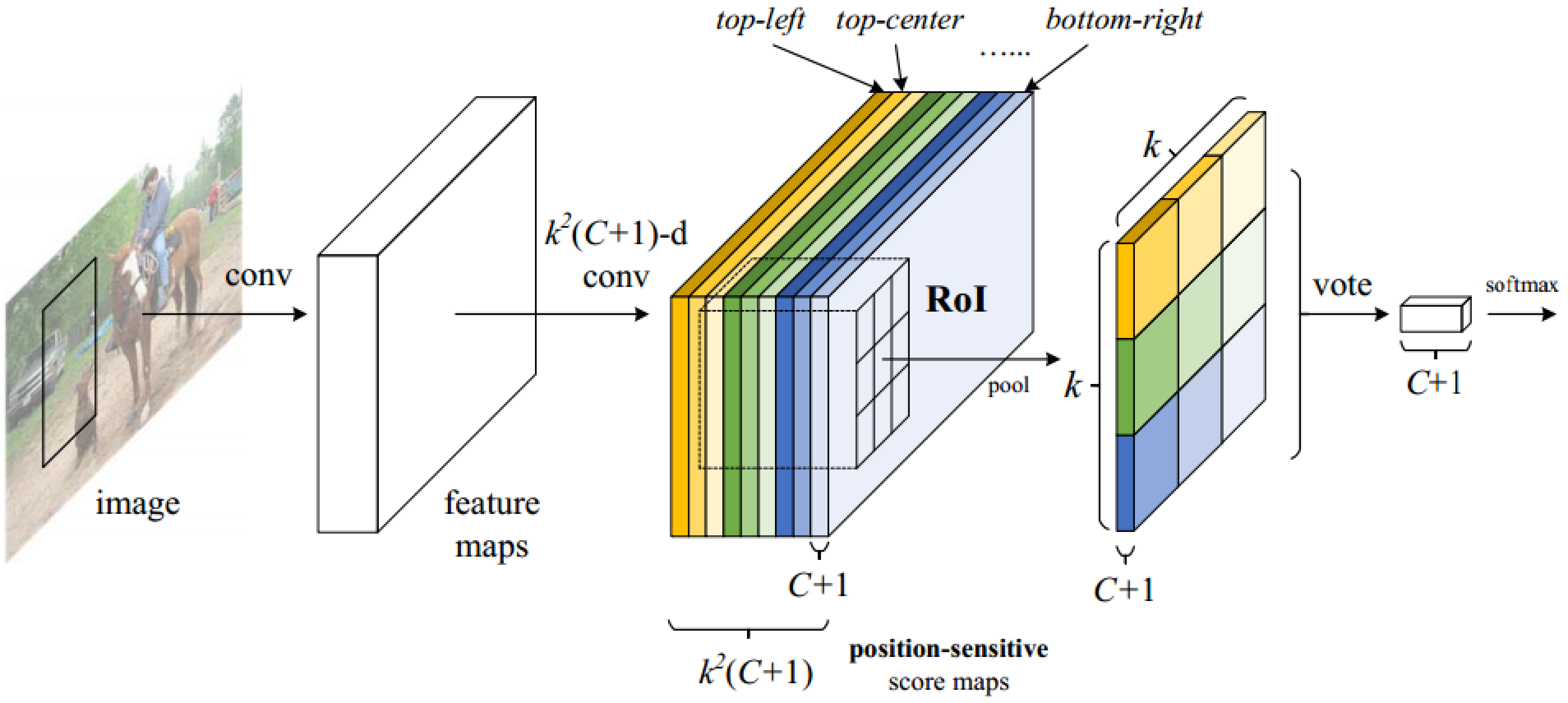
Motivation

- Translation invariance for classification V.S. translation variance for detection
- Unshared per-RoI computation (Fast R-CNN/Faster R-CNN)

Table 1: Methodologies of *region-based* detectors using **ResNet-101** [9].

	R-CNN [7]	Faster R-CNN [19, 9]	R-FCN [ours]
depth of shared convolutional subnetwork	0	91	101
depth of RoI-wise subnetwork	101	10	**0**

Key idea of R-FCN



$$r_c(i, j \mid \Theta) = \sum_{(x,y)\in \text{bin}(i,j)} z_{i,j,c}(x + x_0, y + y_0 \mid \Theta)/n.$$

Visualization of R-FCN

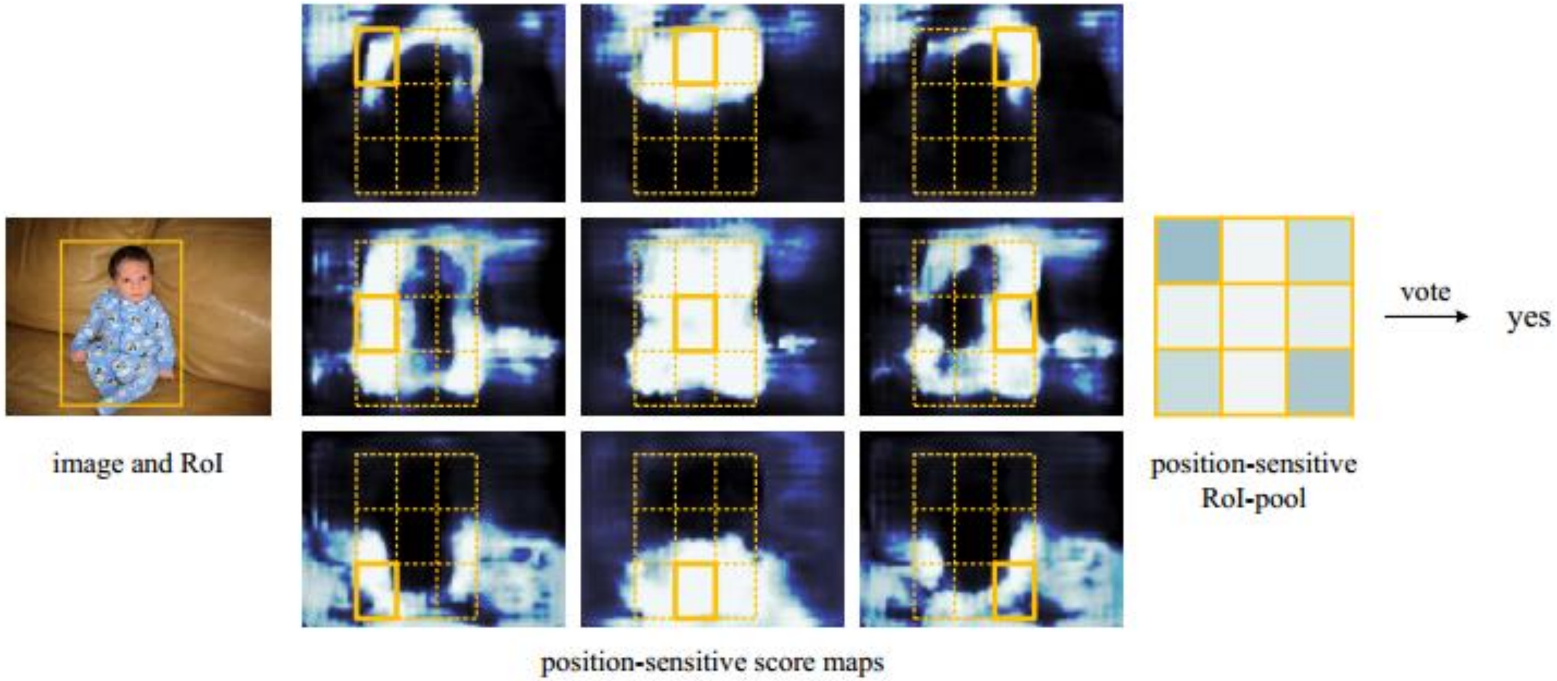


Figure 3: Visualization of R-FCN ($k \times k = 3 \times 3$) for the *person* category.

Visualization of R-FCN

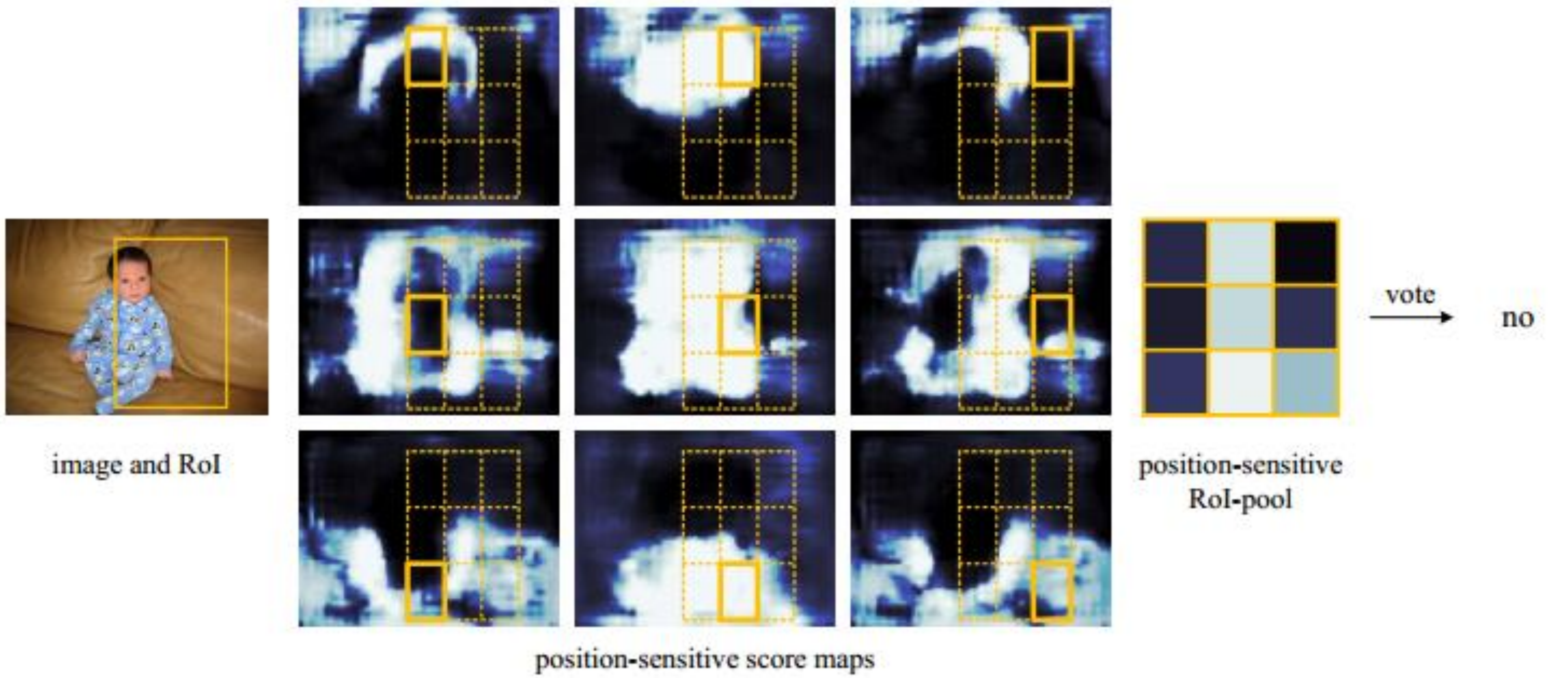
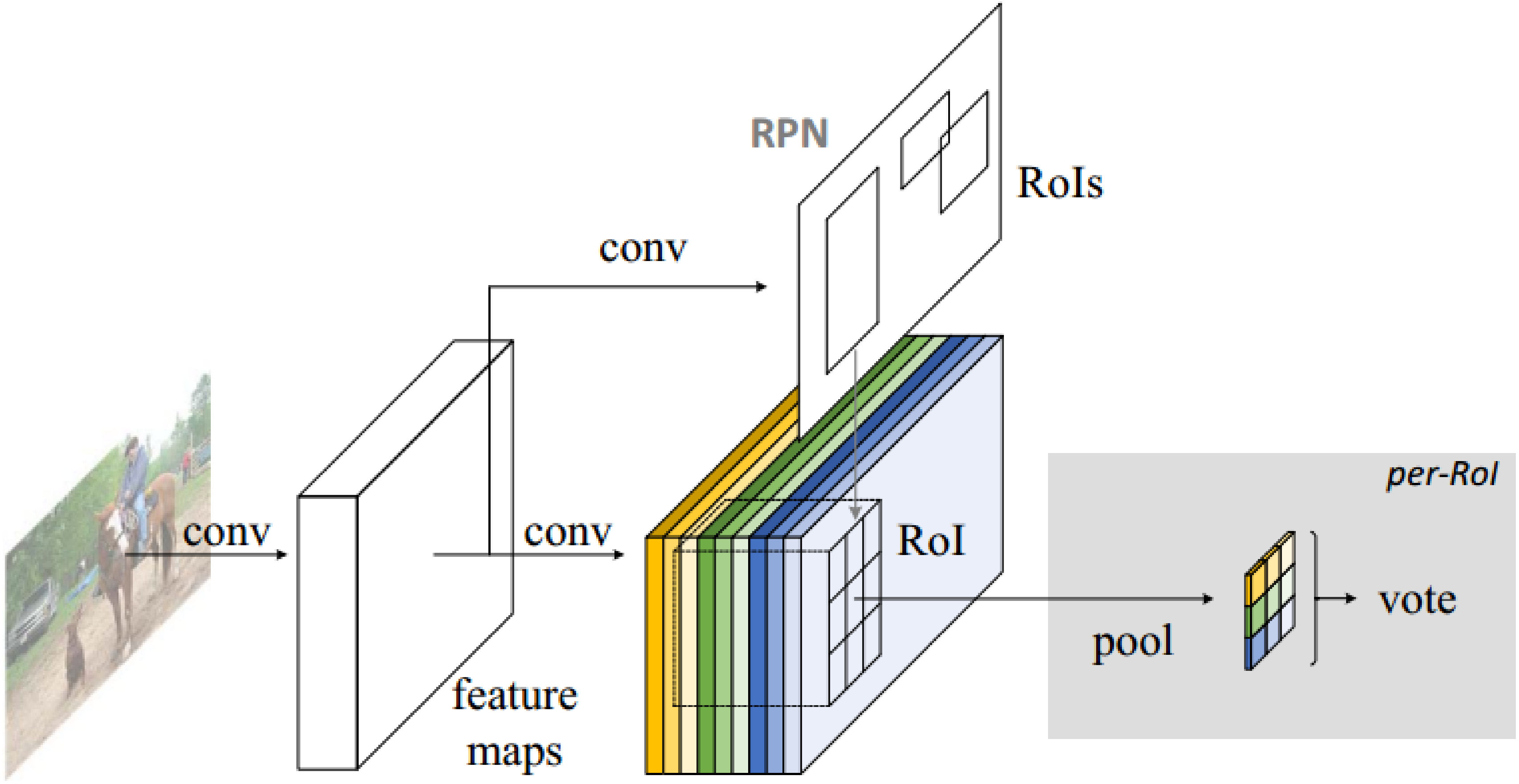


Figure 4: Visualization when an RoI does not correctly overlap the object.

Network Architecture



Results

- Test on PASCAL VOC 2007

	training data	mAP (%)	test time (sec/img)
Faster R-CNN [9]	07+12	76.4	0.42
Faster R-CNN +++ [9]	07+12+COCO	**85.6**	3.36
R-FCN	07+12	79.5	0.17
R-FCN multi-sc train	07+12	80.5	0.17
R-FCN multi-sc train	07+12+COCO	**83.6**	0.17

- Test on PASCAL VOC 2012

	training data	mAP (%)	test time (sec/img)
Faster R-CNN [9]	07++12	73.8	0.42
Faster R-CNN +++ [9]	07++12+COCO	**83.8**	3.36
R-FCN multi-sc train	07++12	77.6[†]	0.17
R-FCN multi-sc train	07++12+COCO	**82.0**[‡]	0.17

Contributions

- A simple but accurate and efficient framework for object detection
- Accuracy competitive with Faster R-CNN, but much faster.

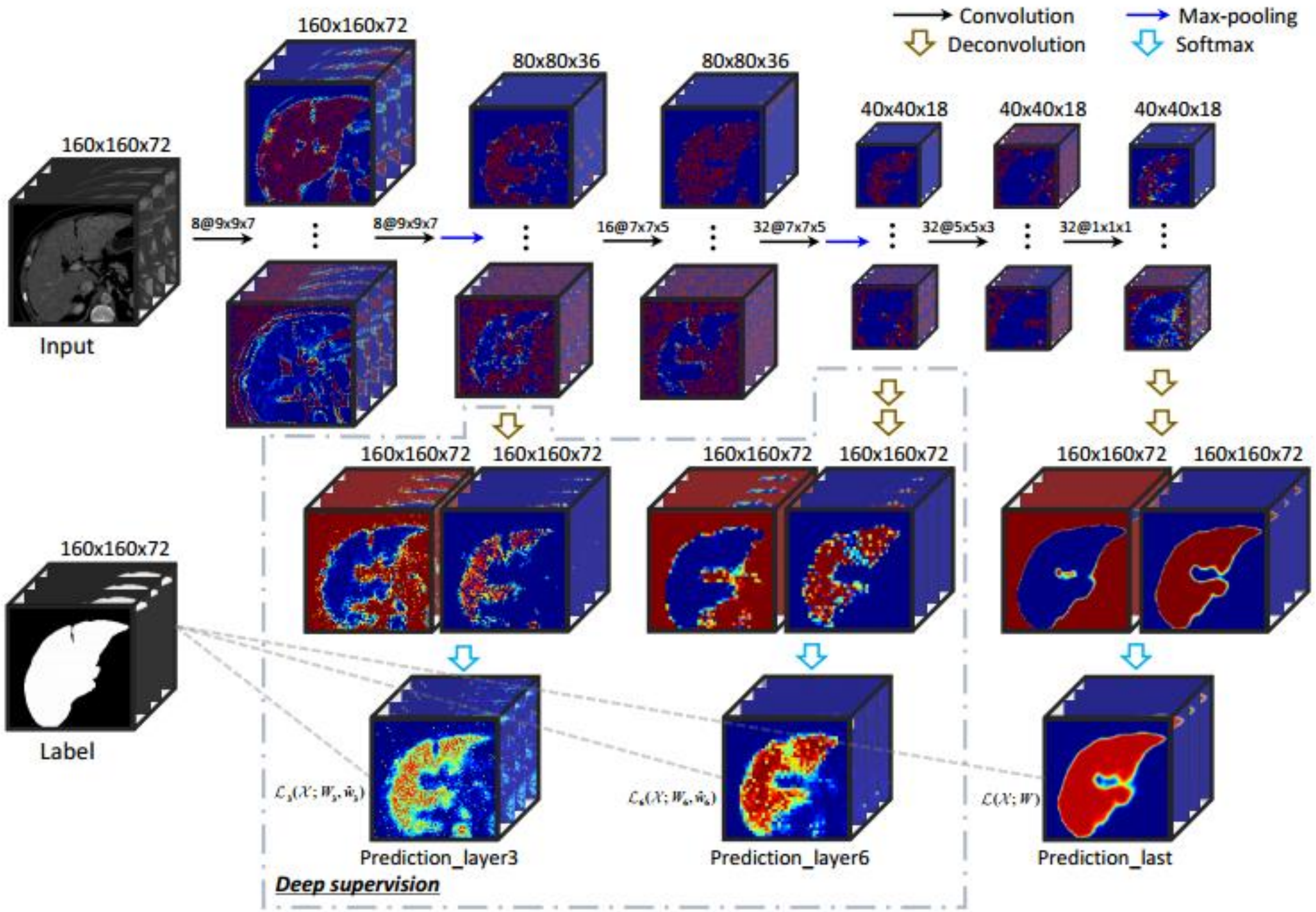
3D Deeply Supervised Network for Automatic Liver Segmentation from CT Volumes

Qi Dou, Hao Chen, Yueming Jin, Lequan Yu, Jing Qin and Pheng-Ann Heng
The Chinese University of HongKong, The HongKong Polytechnic University
In Arxiv.org. 3 July 2016.

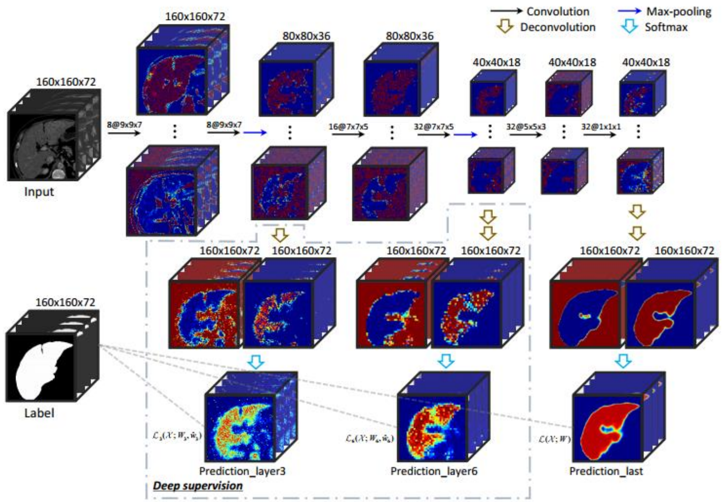
Motivation

- Accurate liver segmentation.
- Previous methods either relied on handcrafted features or did not take full advantage of 3D spatial information.
- Promising performance and efficiency of FCN.
- For small dataset: deep supervision.

Network Architecture



Deep supervision



- Prediction Loss:

$$\mathcal{L}(\mathcal{X}; W) = \sum_{x_i \in \mathcal{X}} -\log p\,(t_i \,|\, x_i; W),$$

- Auxiliary loss:

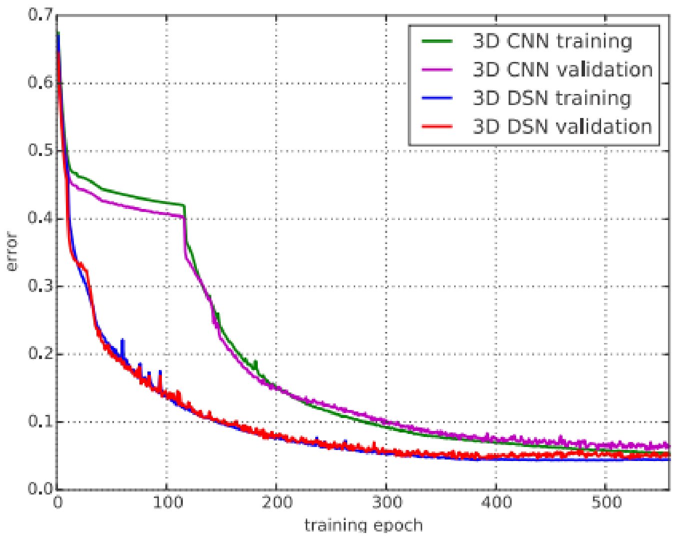
$$\mathcal{L}_d(\mathcal{X}; W_d, \hat{w}_d) = \sum_{x_i \in \mathcal{X}} -\log p\,(t_i \,|\, x_i; W_d, \hat{w}_d).$$

- Overall loss function:

$$\mathcal{L} = \mathcal{L}(\mathcal{X}; W) + \sum_{d \in \mathcal{D}} \eta_d \mathcal{L}_d(\mathcal{X}; W_d, \hat{w}_d) + \lambda(\|W\|^2 + \sum_{d \in \mathcal{D}} \|\hat{w}_d\|^2),$$

Train and inference

- MICCAI-SLiver07 dataset which contains 30 contrast-enhanced CT scans (20 training and 10 testing).
- 2 minutes per epoch on a GPU, and converges after 500 epochs.



- The output of the last deconv. layer is prediction result.

Results

Dataset	Methods	VOE	VD	AvgD	RMSD	MaxD
Training Set	3D-CNN	7.68	1.98	1.56	4.09	45.99
	3D-DSN	6.27	1.46	1.32	3.38	36.49
	3D-CNN+CRF	5.64	1.72	0.89	1.73	34.42
	3D-DSN+CRF	**5.37**	**1.32**	**0.67**	**1.48**	**29.63**

Dataset	Teams	VOE	VD	AvgD	RMSD	MaxD	Runtime
Testing Set	MBI@DKFZ [5]	7.73	1.66	1.39	3.25	30.07	7 mins
	ZIB-Charite [7]	6.09	-2.86	0.95	1.87	18.69	15 mins
	TNT-LUH [1]	6.44	1.53	0.95	**1.58**	**15.92**	-
	LME Erlangen [12]	6.47	**1.04**	1.02	2.00	18.32	-
	Ours(3D-DSN+CRF)	**5.42**	1.75	**0.79**	1.64	33.55	**1.5 mins**

VOE=$100(1-(|A\cap B|/|A\cup B|))$

VD=$100(|A|-|B|/|B|)$

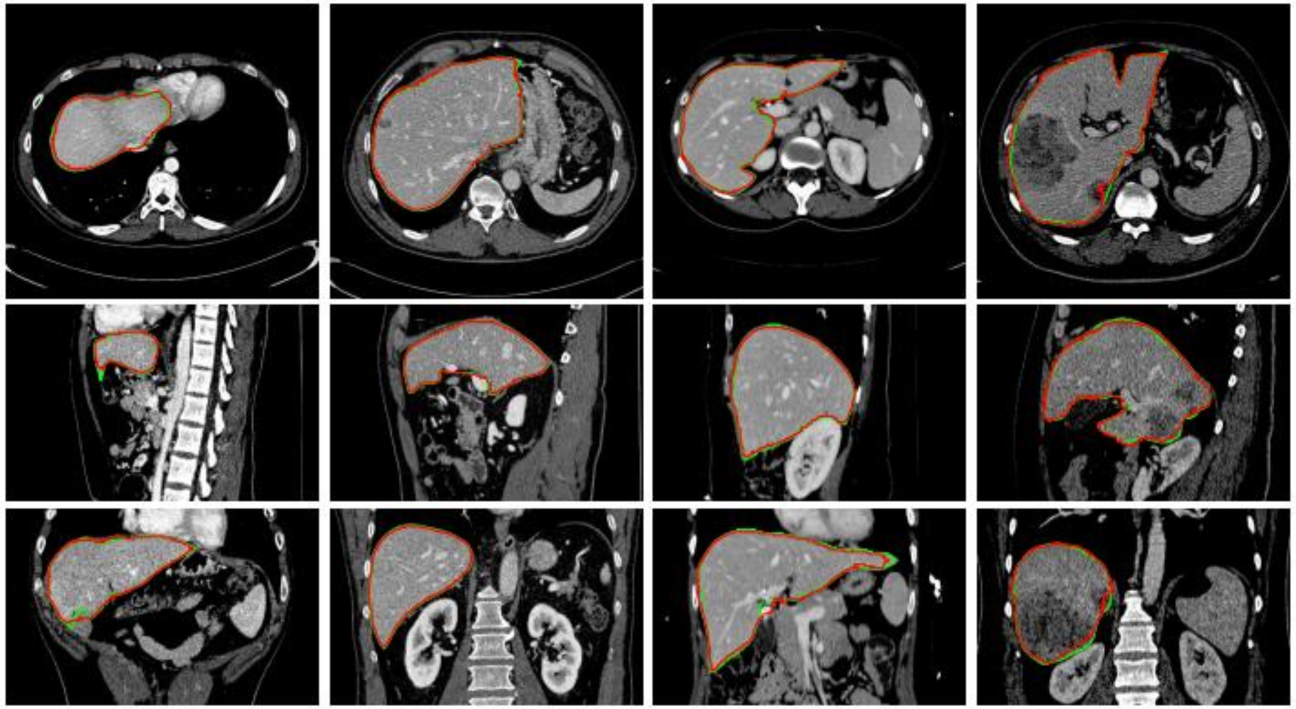
$$\mathrm{ASD}(A,B)=\frac{1}{|S(A)|+|S(B)|}\left(\sum_{s_A\in S(A)} d(s_A,S(B)) + \sum_{s_B\in S(B)} d(s_B,S(A))\right).$$

$$\mathrm{RMSD}(A,B)=\sqrt{\frac{1}{|S(A)|+|S(B)|}} \times\sqrt{\sum_{s_A\in S(A)} d^2(s_A,S(B))+\sum_{s_B\in S(B)} d^2(s_B,S(A))}.$$

$$\mathrm{MSD}(A,B)=\max\left\{\max_{s_A\in S(A)} d(s_A,S(B)),\ \max_{s_B\in S(B)} d(s_B,S(A))\right\}$$

Results

- Green is ground truth, red is results



(a) (b) (c) (d)

Summary

- FCN can be applied to multiple pixel-wise tasks.
- FCN can be trained end-to-end, pixels-to-pixels on whole image.
- The input of FCN can be arbitrary size.

Thank You