Semantic Segmentation, Dense Labeling

Liwei Wang
CS@UIUC
Semantic segmentation
Why Semantic Segmentation?

- Road Scene Understanding
- Useful for Self-Driving Car and autonomous drones

From *cityscape* dataset
Why Semantic Segmentation?

- Medical Image Analysis
Very Challenging Problem
History

• Problem: label each pixel by one of C classes
• Define an energy function where unaries correspond to local classifier responses and smoothing potentials correspond to contextual terms
• Solve a multi-class graph cut problem

\[
\log P(c|x, \theta) = \sum_i \psi_i(c_i, x; \theta_\psi) + \pi(c_i, x_i; \theta_\pi) + \lambda(c_i, i; \theta_\lambda) + \sum_{(i,j) \in E} \phi(c_i, c_j, g_{ij}(x); \theta_\phi) - \log Z(\theta, x) \quad (1)
\]

From TextonBoost, ECCV 2006
CRF energy function is defined on super-pixels:

- Unaries are based on nearest neighbor retrieval
- Pairwise potentials capture class co-occurrence statistics

J. Tighe and S. Lazebnik, *SuperParsing, ECCV 2010*
Now, what is happening

• How deep neural networks can be used for Semantic Segmentation?

• How to model local and contextual information with Deep Nets?

• Differences and Similarities among methods?
Zoom-out Features

- A simple Feed-Forward Network
- Feature Concatenation from Different Scales
- Strong features + Softmax Classification
Zoom-out Features

• **Sub-scene Level Features**
  • Bounding box of superpixels within radius three from the superpixel at hand
  • Warp bounding box to 256 x 256 pixels
  • Activations of the last fully connected layer

• **Scene Level Features**
  • Warp image to 256 x 256 pixels
  • Activations of the last fully connected layer
Zoom-out Features
Zoom-out Features

Feedforward Semantic Segmentation With Zoom-Out Features, CVPR 2015
Hypercolumns Representation

Hypercolumns for object segmentation and fine-grained localization CVPR 2015
Hypercolumn Representation

Hypercolumns for object segmentation and fine-grained localization CVPR 2015
Hypercolumn Representation

![Diagram of Hypercolumn Representation]

\[ p_i = \sum_k \alpha_{ik} g_k(f_i) = \sum_k \alpha_{ik} p_{ik} \]

*Hypercolumns for object segmentation and fine-grained localization* CVPR 2015
Evaluation

• Mean IoU

Per-class evaluation: an intersection of the predicted and true sets of pixels for a given class, divided by their union (IoU)

\[
\text{seg. accuracy} = \frac{\text{true pos.}}{\text{true pos.} + \text{false pos.} + \text{false neg.}}
\]

<table>
<thead>
<tr>
<th></th>
<th>VOC 2012</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zoom-out</td>
<td>69.6</td>
</tr>
<tr>
<td>Hypercolumn</td>
<td>62.6</td>
</tr>
</tbody>
</table>
Evaluation

**Zoom-out Method on Pascal VOC2012**

| class | mean | bg |鸟 |自行车 |汽车 |狗 |桌子 |椅子 |汽车 |狗 |自行车 |汽车 |桌子 |椅子 |汽车 |狗 |自行车 |汽车 |桌子 |椅子 |汽车 |狗 |自行车 |汽车 |桌子 |椅子 |汽车 |狗 |自行车 |汽车 |桌子 |椅子 |汽车 |狗 |自行车 |汽车 |桌子 |椅子 |汽车 |
|-------|------|----|----|--------|------|----|-----|------|------|----|--------|------|-----|------|------|----|--------|------|-----|------|------|----|--------|------|-----|------|------|----|--------|------|-----|------|------|----|--------|------|-----|------|------|----|
| acc   | 69.6 | 91.9 | 85.6 | 37.3   | 83.2 | 66  | 85.1 | 80.7 | 84.9 | 27.2| 73.3   | 57.5 | 78.1 | 79.2 | 81.1 | 77.1 | 53.6   | 74   | 49.2 | 71.7 | 63.3 |
Hypercolumn and Zoom out

• Both uses the *multi-scale* features from intermediate layers in CNN

• Both use *upsampling* operations for each scale

Any Pixel to Pixel ways ?
Can upsampling be learned ?
FCN for Semantic Segmentation

Fully convolutional Networks for Semantic Segmentation, CVPR 2015
Convnets for Classification

Fully convolutional Networks for Semantic Segmentation, CVPR 2015
Convnets for Segmentation?

Fully convolutional Networks for Semantic Segmentation, CVPR 2015
From Convnets to FCN

Fully convolutional Networks for Semantic Segmentation, CVPR 2015
Pixel in, Pixel out

Upsampling is backwards strided convolution
How does convolution works?

Kernel/filter:

\[
\begin{pmatrix}
0 & 1 & 2 \\
2 & 2 & 0 \\
0 & 1 & 2 \\
\end{pmatrix}
\]
How does convolution works?

Kernel/filter:

```
<table>
<thead>
<tr>
<th>0 2</th>
<th>0 0</th>
<th>0 1</th>
<th>0 0</th>
<th>0 0</th>
<th>0 0</th>
<th>0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 2</td>
<td>2 0</td>
<td>2 0</td>
<td>3 3</td>
<td>3 0</td>
<td>0 0</td>
<td>0 0</td>
</tr>
<tr>
<td>0 1</td>
<td>0 1</td>
<td>1 1</td>
<td>3 0</td>
<td>3 0</td>
<td>0 0</td>
<td>0 0</td>
</tr>
<tr>
<td>0 0</td>
<td>2 3</td>
<td>0 1</td>
<td>3 0</td>
<td>0 0</td>
<td>0 0</td>
<td>0 0</td>
</tr>
<tr>
<td>0 3</td>
<td>3 2</td>
<td>1 2</td>
<td>0 0</td>
<td>0 0</td>
<td>0 0</td>
<td>0 0</td>
</tr>
<tr>
<td>0 0</td>
<td>2 3</td>
<td>0 2</td>
<td>3 0</td>
<td>0 0</td>
<td>0 0</td>
<td>0 0</td>
</tr>
</tbody>
</table>
```

```plaintext
1 6 5
1 7 10 9
7 10 8
```
How does convolution work?
How does convolution work?

Convolution as a matrix operation

From Theano Document Website
Upsampling is backwards strided convolution

Transposed convolution

From Theano Document Website
Actually, we can build deconv networks...
Deconv Layers for generating image!
Framework of FCN
Framework of FCN
More than one upsampling layer

FCN is still not good?

Ground Truth  Image  FCN-8s

Very coarse feature maps --- FCN-8s is still very coarse
Dilated Convolution

Discrete Convolution Operation:

\[(F \ast k)(p) = \sum_{s+t=p} F(s) k(t)\]

Dilated Convolution Operation:

\[(F \ast_t k)(p) = \sum_{s+lt=p} F(s) k(t)\]
Dilated Convolution

(a) 
(b) 
(c)
Dilated Convolution
Network with Dilated Convolution

• Following FCN structure

• Using VGG-16 networks with modifications

• Called Front End and greatly improve performance
Network with Dilated Convolution

From arunmallya
Network with Dilated Convolution
Network with Dilated Convolution

<table>
<thead>
<tr>
<th></th>
<th>aero</th>
<th>bike</th>
<th>bird</th>
<th>boat</th>
<th>bottle</th>
<th>bus</th>
<th>car</th>
<th>cat</th>
<th>chair</th>
<th>cow</th>
<th>table</th>
<th>dog</th>
<th>horse</th>
<th>mbike</th>
<th>person</th>
<th>plant</th>
<th>sheep</th>
<th>sofa</th>
<th>train</th>
<th>tv</th>
<th>mean IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCN-8s</td>
<td>76.8</td>
<td>34.2</td>
<td>68.9</td>
<td>49.4</td>
<td>60.3</td>
<td>75.3</td>
<td>74.7</td>
<td>77.6</td>
<td>21.4</td>
<td>62.5</td>
<td>46.8</td>
<td>71.8</td>
<td>63.9</td>
<td>76.5</td>
<td>73.9</td>
<td>73.6</td>
<td>75.0</td>
<td>79.1</td>
<td>75.3</td>
<td>60.9</td>
<td>62.2</td>
</tr>
<tr>
<td>DeepLab</td>
<td>72</td>
<td>31</td>
<td>71.2</td>
<td>53.7</td>
<td>60.5</td>
<td>77</td>
<td>71.9</td>
<td>73.1</td>
<td>25.2</td>
<td>62.6</td>
<td>49.1</td>
<td>68.7</td>
<td>63.3</td>
<td>73.9</td>
<td>73.6</td>
<td>50.8</td>
<td>72.3</td>
<td>42.1</td>
<td>67.9</td>
<td>52.6</td>
<td>62.1</td>
</tr>
<tr>
<td>DeepLab-Msc</td>
<td>74.9</td>
<td>34.1</td>
<td>72.6</td>
<td>52.9</td>
<td>61.0</td>
<td>77.9</td>
<td>73.0</td>
<td>73.7</td>
<td>26.4</td>
<td>62.2</td>
<td>49.3</td>
<td>68.4</td>
<td>64.1</td>
<td>74.0</td>
<td>75.0</td>
<td>51.7</td>
<td>72.7</td>
<td>42.5</td>
<td>67.2</td>
<td>55.7</td>
<td>62.9</td>
</tr>
<tr>
<td>Our front end</td>
<td>82.2</td>
<td>37.4</td>
<td>72.7</td>
<td>57.1</td>
<td>62.7</td>
<td>82.8</td>
<td>77.8</td>
<td>78.9</td>
<td>28</td>
<td>70</td>
<td>51.6</td>
<td>73.1</td>
<td>72.8</td>
<td>81.5</td>
<td>79.1</td>
<td>56.6</td>
<td>77.1</td>
<td>49.9</td>
<td>75.3</td>
<td>60.9</td>
<td>67.6</td>
</tr>
</tbody>
</table>
Multi-scale Context Aggregation

<table>
<thead>
<tr>
<th>Layer</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convolution</td>
<td>$3 \times 3$</td>
<td>$3 \times 3$</td>
<td>$3 \times 3$</td>
<td>$3 \times 3$</td>
<td>$3 \times 3$</td>
<td>$3 \times 3$</td>
<td>$3 \times 3$</td>
<td>$1 \times 1$</td>
</tr>
<tr>
<td>Dilation</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Truncation</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Receptive field</td>
<td>$3 \times 3$</td>
<td>$5 \times 5$</td>
<td>$9 \times 9$</td>
<td>$17 \times 17$</td>
<td>$33 \times 33$</td>
<td>$65 \times 65$</td>
<td>$67 \times 67$</td>
<td>$67 \times 67$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Output channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic</td>
</tr>
<tr>
<td>Large</td>
</tr>
</tbody>
</table>
Multi-scale Context Aggregation

<table>
<thead>
<tr>
<th>Layer Name</th>
<th>Receptive Field</th>
<th>Effective Stride</th>
<th>Output Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>--</td>
<td>--</td>
<td>66</td>
</tr>
<tr>
<td>ct_conv1_1</td>
<td>3</td>
<td>1</td>
<td>130</td>
</tr>
<tr>
<td>ct_conv1_2</td>
<td>5</td>
<td>1</td>
<td>128</td>
</tr>
<tr>
<td>ct_conv2_1</td>
<td>9</td>
<td>1</td>
<td>124</td>
</tr>
<tr>
<td>ct_conv3_1</td>
<td>17</td>
<td>1</td>
<td>116</td>
</tr>
<tr>
<td>ct_conv4_1</td>
<td>33</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>ct_conv5_1</td>
<td>65</td>
<td>1</td>
<td>68</td>
</tr>
<tr>
<td>ct_fcl</td>
<td>67</td>
<td>1</td>
<td>66</td>
</tr>
<tr>
<td>ct_final</td>
<td>67</td>
<td>1</td>
<td>66</td>
</tr>
</tbody>
</table>

From arunmallya
Front End + Context Net

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

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

Learning to refine object segments, ECCV 2016
Refine Deep Mask

(a) DeepMask Output  (b) SharpMask Output

Learning to refine object segments, ECCV 2016
Dense Labeling Task: Learning Optical Flow

FlowNet: Learning Optical Flow with Convolutional Networks
FlowNetSimple

FlowNet: Learning Optical Flow with Convolutional Networks
FlowNetCorr

FlowNet: Learning Optical Flow with Convolutional Networks
Refinement of the coarse feature maps

FlowNet: Learning Optical Flow with Convolutional Networks
Examples of Data Pairs
## Results

<table>
<thead>
<tr>
<th>Images</th>
<th>Ground truth</th>
<th>EpicFlow</th>
<th>FlowNetS</th>
<th>FlowNetC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>EPE: 0.27</td>
<td>EPE: 1.06</td>
<td>EPE: 0.91</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EPE: 11.62</td>
<td>EPE: 7.17</td>
<td>EPE: 11.18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EPE: 32.56</td>
<td>EPE: 20.82</td>
<td>EPE: 26.63</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EPE: 24.98</td>
<td>EPE: 35.33</td>
<td>EPE: 46.68</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EPE: 0.33</td>
<td>EPE: 0.89</td>
<td>EPE: 0.71</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EPE: 1.56</td>
<td>EPE: 1.41</td>
<td>EPE: 1.07</td>
</tr>
</tbody>
</table>

FlowNet: Learning Optical Flow with Convolutional Networks