Efficient Convolutional Patch Networks for Scene Understanding

Clemens-Alexander Brust, Sven Sickert, Marcel Simon, Erik Rodner and Joachim Denzler
Computer Vision Group, Friedrich Schiller University Jena, Germany
http://www.inf-cv.uni-jena.de

Code Available

CN24 Open Source Deep Learning Framework

CNN Framework especially designed for scene understanding [1]

• Graph-based network structure can be freely specified
• OpenCL, MKL, ACML and dependency-free reference backends
• Hybrid patch-wise and fully convolutional design for fast prediction and efficient training
• 3-clause BSD license suitable for research and commercial applications

Visit our project page!
www.inf-cv.uni-jena.de/semseg

Method Details

• Fully convolutional networks (FCN) [6] for prediction speed-up (up to 100x) over patch-wise approach
• Patch-wise training leads to faster optimization
• Real-time segmentation of VGA-sized inputs
• Incorporating position information as spatial priors
• Post-processing: quality enhancement of network outputs using unsupervised segmentation [5]
• Weighting: optimization with inverse class frequency weights accounts for imbalanced training set

cvjena.github.io/cn24/

Incorporating Spatial Priors

• Adding the location of a patch to the network's input increases classification performance significantly
• Simple solution: We add additional input layers with pixel coordinates

Qualitative Results

Visit our project page!
www.inf-cv.uni-jena.de/semseg

Quantitative Results

<table>
<thead>
<tr>
<th>Method</th>
<th>ORR</th>
<th>ARR</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDF+SIFT [2]</td>
<td>49.06%</td>
<td>44.08%</td>
</tr>
<tr>
<td>ICF [3]</td>
<td>67.33%</td>
<td>56.61%</td>
</tr>
<tr>
<td>RDF-MAP [4]</td>
<td>71.28%</td>
<td>-</td>
</tr>
<tr>
<td>Our Approach</td>
<td>ORR</td>
<td>ARR</td>
</tr>
<tr>
<td>CNN outputs (FCN training)</td>
<td>58.17%</td>
<td>29.48%</td>
</tr>
<tr>
<td>CNN outputs (patch-wise training)</td>
<td>67.87%</td>
<td>42.89%</td>
</tr>
<tr>
<td>+ spatial prior</td>
<td>72.21%</td>
<td>47.74%</td>
</tr>
<tr>
<td>+ post processing</td>
<td>74.33%</td>
<td>47.77%</td>
</tr>
<tr>
<td>+ weighting</td>
<td>63.41%</td>
<td>58.98%</td>
</tr>
</tbody>
</table>