Multi-Grained Deep Feature Learning for Pedestrian Detection

Chunze Lin, Jiwen Lu, Jie Zhou
Tsinghua University, China
Outline

① Introduction
② Related Work
③ Proposed Approach
④ Experimental Results
What is Pedestrian Detection?

Complex problem for machine
- **Recognition**: What is a pedestrian?
- **Localization**: Where are pedestrians?
Why Pedestrian Detection?

Autonomous driving

Intelligent surveillance

Robotics
Challenges

- Large variances of scales
- Occlusions
- Blurry representation
- Noisy background
Outline

① Introduction

② Related Work

③ Proposed Approach

④ Experimental Results
Possible Solutions

- Learn multiple models for different scales
- Human parts detection
- Hard negative samples mining
Scale-aware Fast R-CNN for Pedestrian Detection[1]

Deep learning strong parts for pedestrian detection[2]

Is faster R-CNN doing well for pedestrian detection?[3]

(a) Small positive instances  (b) Hard negatives

Outline

① Introduction

② Related Work

③ Proposed Approach

④ Experimental Results
Motivation & Contributions

- Full-body and part-based methods: too coarse to localize small and occluded pedestrians

- **Fine-grained information** with pixel-wise classification to help detection

- **Multiple feature maps** of different resolutions to deal with scale variances issue
Flowchart

- Human Parsing Network
  - Generate human parsing mask and convert it into attention map

- Scale-Aware Network
  - Exploit intermediate feature maps for multiple scales detection
  - Attention map guides the detection to focus on pedestrians
Human Parsing Network

- **Network Architecture**
  - Truncated VGG16 with ‘atrous’ convolution
  - Deconvolution to up-sample to image size
  - Concatenate multiple layers to form hierarchical feature maps
Human Parsing Network

- **Weakly Supervised Training**
  - Only bounding box annotations available
  - Consider 80% pixels at the center area of the bounding box as foreground
    - Eliminate background noise
    - Focus on main parts of human

![HPN Diagram]
Scale-Aware Network

- Network Structure
  - Truncated VGG16 + extra convolutional layers

- Multiple scale detection
  - High resolution feature maps (shallower layers) for small targets detection
  - High-level semantic feature maps (deeper layers) for large pedestrians detection
  - Each detection layer followed by a detection module
Scale-Aware Network

Detection Module

- Encode attention map into feature maps
  \[ A_{s,c} = D_{s,c}(M) \odot F_{s,c} \]

- Context module: concatenate 2 layers of different receptive fields
  → Incorporate more context information

- Prediction module outputs the detection results
Visualization of feature maps

Image/Patch

Initial feature maps

Feature maps with attention
Implementation details

1. Separately train Scale-Aware Network and Human Parsing Network
2. Jointly optimize both networks
   \[ \rightarrow \text{Facilitate the convergence} \]

Multi-task loss

\[ L = L_{\text{box}} + \lambda_c L_{\text{conf}} + \lambda_s L_{\text{seg}} \]

- \( L_{\text{box}} \) for SAN
- \( L_{\text{conf}} \) for HPN
Outline

① Introduction

② Related Work

③ Proposed Approach

④ Experimental Results
Experimental Results

- **Caltech Pedestrian**
  - 42,782 training images
  - 4,024 test images
  - Evaluation metric: average miss rate

- **KITTI**
  - 7,481 training images
  - 7,518 test images
  - Evaluation metric: mean average precision (AP)
Experimental Results

- **Caltech Pedestrian**
  - **Heavy occluded**: taller than 50 pixels, visibility $\in [0.36, 0.80]$
  - **Medium**: pedestrian height $\in [30, 80]$ pixels, reasonable visibility
  - **Overall**: all pedestrian taller than 20 pixels, with or without occlusion

(a) Heavy Occluded

(b) Medium

(c) Overall
Experimental Results

- **KITTI**
  - Moderate setting: pedestrian taller than 25 pixels with or non occlusion

- **Computing Time**
  - Real time pedestrian detector
  - Our method is at least 2x faster
  - Great trade-off of performance and runtime

<table>
<thead>
<tr>
<th>Method</th>
<th>Caltech</th>
<th>KITTI</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPN+BF [4]</td>
<td>74.36</td>
<td>61.29</td>
<td>0.5s</td>
</tr>
<tr>
<td>SA-FastRCNN [25]</td>
<td>64.35</td>
<td>65.01</td>
<td>0.5s</td>
</tr>
<tr>
<td>DeepParts [16]</td>
<td>60.42</td>
<td>58.67</td>
<td>1s</td>
</tr>
<tr>
<td>MS-CNN [5]</td>
<td>59.94</td>
<td>73.70</td>
<td>0.14s</td>
</tr>
<tr>
<td>SDS-RCNN [11]</td>
<td>58.55</td>
<td>63.05</td>
<td>0.21s</td>
</tr>
<tr>
<td>F-DNN[13]</td>
<td>55.13</td>
<td>-</td>
<td>0.3s</td>
</tr>
<tr>
<td>F-DNN+SS [13]</td>
<td>53.76</td>
<td>-</td>
<td>2.48s</td>
</tr>
<tr>
<td>JL-Tops [10]</td>
<td>49.20</td>
<td>-</td>
<td>0.6s</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td>38.53</td>
<td>66.32</td>
<td>0.07s</td>
</tr>
</tbody>
</table>
Ablation Studies

Disable main components successively
- Segmentation mask: performance drops by ~2%
- Context module: performance drops by ~2.5%

<table>
<thead>
<tr>
<th>Component Disabled</th>
<th>Medium</th>
<th>Heavy</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context module</td>
<td>35.31</td>
<td>44.37</td>
<td>49.99</td>
</tr>
<tr>
<td>Segmentation mask</td>
<td>33.27</td>
<td>40.27</td>
<td>47.83</td>
</tr>
<tr>
<td>Our-MDFL</td>
<td>31.46</td>
<td>38.53</td>
<td>46.85</td>
</tr>
</tbody>
</table>

1. Segmentation mask
2. Context module
Conclusion and Future Works

- **Fine Grained Attention map**
  - Guide the detector to focus on pedestrians
  - Eliminate background interference

- **Future works**
  - Implement the proposed method into video based detector
  - Exploit temporal information
Thanks!