Deep Learning for Fluoroscopic Feature Detection

Background Presentation

Liujiang Yan

Mentor: Robert Grupp, Professor Russell Taylor

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING
Motivation

• Feature Detection for 2D-3D Registration
• Anatomical Landmark Detection
• Contour Detection

Papers Discussed Today

• Human Pose Detection
  

• Edge Detection
  
Convolutional Pose Machine

• Designed for human pose estimation task
• Sequential architecture composed of convolution layers to implicitly modeling long-range dependencies
• Perform belief map regression instead of pixel position
• Alleviate gradient vanishing by intermediate supervision
Network Architecture

- Feature extractor at each stage (x)
  - Weight sharing among stages
  - Representation for the original image for following classification use
Network Architecture

- Classifier at each stage (gt)
  - Takes previous stage output and feature extracted as input
  - Receptive field grows larger with deeper network
Results and Discussions

- State of the art performance on single human pose estimation task.
- Weight sharing for feature extractor
- Inexplicit long range dependencies modeling
- Intermediate supervision (MSE loss for each stage)

\[ F = \sum_{t}^{T} \sum_{p}^{P+1} \sum_{z} \| b^p_t(z) - b^p_* (z) \|^2 \]
Results and Discussions

**Figure 8:** Quantitative results on the MPII dataset using the PCKh metric. We achieve state of the art performance and outperform significantly on difficult parts such as the ankle.

**Figure 9:** Quantitative results on the LSP dataset using the PCK metric. Our method again achieves state of the art performance and has a significant advantage on challenging parts.
Application

• Fixed number key point regression problem
• Belief map as uncertainly measure
  • Threshold poor predictions to make robust registration
  • Embed uncertainty into optimization objective function
Holistically-Nested Edge Detection

- Holistic image training and prediction
- Multi scale feature learning through deep neural networks
- Completely differentiate framework
- Implicitly long range dependencies capture without graphical model
Network Architecture

- Utilize VGG (convolutional parts) as basic block for feature extraction
  - With network going deeper, the features are extracted locally to globally
Network Architecture - Training

- Side output at each stage
  - Utilize conv layer as predictor here
- Loss at each stage
  - Weighted binary cross entropy loss

\[
L = -\beta \sum_{Y_+} \log P(y = 1|X; w) \\
-(1 - \beta) \sum_{Y_-} \log P(y = 0|X; w)
\]
Network Architecture - Prediction

• Obtain side outputs and fused output from the network
  \[ Y^{fuse}, Y^{(1)}, \ldots, Y^{(M)} = HED(X. (W. w, h)) \]

• Aggregate these predictions for final result
  \[ Y^{HED} = \text{average}(Y^{fuse}, Y^{(1)}, \ldots, Y^{(M)}) \]

• Perform non maximum suppression for thinned edges result.
  \[ Y^{Final} = \text{non max suppression}(Y^{HED}) \]
Results

• State of the art performance
  • fixed contour threshold (ODS)
  • Per-image best threshold (OIS)
  • Average precision (AP)

<table>
<thead>
<tr>
<th></th>
<th>ODS</th>
<th>OIS</th>
<th>AP</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human</td>
<td>.80</td>
<td>.80</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Canny</td>
<td>.600</td>
<td>.640</td>
<td>.580</td>
<td>15</td>
</tr>
<tr>
<td>BEL [5]</td>
<td>.660*</td>
<td>-</td>
<td>-</td>
<td>1/10</td>
</tr>
<tr>
<td>gPb-owt-ucm [1]</td>
<td>.726</td>
<td>.757</td>
<td>.696</td>
<td>1/240</td>
</tr>
<tr>
<td>Sketch Tokens [24]</td>
<td>.727</td>
<td>.746</td>
<td>.780</td>
<td>1</td>
</tr>
<tr>
<td>SCG [31]</td>
<td>.739</td>
<td>.758</td>
<td>.773</td>
<td>1/280</td>
</tr>
<tr>
<td>SE-Var [6]</td>
<td>.746</td>
<td>.767</td>
<td>.803</td>
<td>2.5</td>
</tr>
<tr>
<td>OEF [13]</td>
<td>.749</td>
<td>.772</td>
<td>.817</td>
<td>-</td>
</tr>
<tr>
<td>DeepNets [21]</td>
<td>.738</td>
<td>.759</td>
<td>.758</td>
<td>1/5†</td>
</tr>
<tr>
<td>N4-Fields [10]</td>
<td>.753</td>
<td>.769</td>
<td>.784</td>
<td>1/6†</td>
</tr>
<tr>
<td>DeepEdge [2]</td>
<td>.753</td>
<td>.772</td>
<td>.807</td>
<td>1/103†</td>
</tr>
<tr>
<td>CSCNN [19]</td>
<td>.756</td>
<td>.775</td>
<td>.798</td>
<td>-</td>
</tr>
<tr>
<td>DeepContour [34]</td>
<td>.756</td>
<td>.773</td>
<td>.797</td>
<td>1/30†</td>
</tr>
<tr>
<td><strong>HED (ours)</strong></td>
<td>.782</td>
<td>.804</td>
<td>.833</td>
<td>2.5†, 1/12</td>
</tr>
</tbody>
</table>
Application

• Contour detection in X-Ray images
• Flexibility of basic blocks of HED
  • e.g. ResNet, Fully Convolutional Net, U-Net
• Predicted feature map has uncertainty measure about edges
  • Threshold weak edges
  • Embed into registration method
Conclusion

• Utilize deep neural network as feature extractor and classifier.
• Implicitly model long range dependencies by composing conv layers.
• Alleviate gradient vanishing by intermediate supervision.
• Uncertainty measure as extra information for registration.