Team OKS Keypoints Detection

Joint Workshop of the COCO and Places Challenges at ICCV 2017

Speaker: Cheng Li

Yujie Wang\(^1\)*, Changbao Wang\(^1\)*, Quanquan Li\(^2\)*, Biao Leng\(^1\), Zhoujun Li\(^1\), Junjie Yan\(^2\)

\(^1\)Beihang University \(^2\)SenseTime Group Limited

(*Equal contribution. This work was done when Yujie Wang and Changbao Wang were interns at SenseTime Group Limited)
Outline

• Top-Down method
  • Person detection
  • Pose estimation

• Inference
  • Box Proposal Rescoring
  • OKS-NMS

• AP of our submission
  • 72.0 (test-dev)
  • 71.4 (test-challenge)
Person Detection

- Re-implement FPN + Mask-RCNN
  - Backbone: ResNet-50
  - Data: COCO only
  - Top 20 boxes
- Performance
  - COCO keypoint validation set
  - Box AP (person) 52.1
  - Box AR (person) 61.3

Pose Estimation Network

- Stacked Hourglass (v1) 8 stacks
  - Input size: 256x256
  - Supervision: Gaussian with std 1
  - Only backpropagate the loss of annotated keypoints

Can we make the pose network better?
Explore new architecture

• Hourglass is good, but is it the best?

<table>
<thead>
<tr>
<th>Method</th>
<th>AP (validation, ground truth box)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hourglass 8 stacks</td>
<td>73.4</td>
</tr>
<tr>
<td>Inception ResNet V2*</td>
<td>69.4</td>
</tr>
<tr>
<td>ResNet-269*</td>
<td>69.7</td>
</tr>
</tbody>
</table>

• Can we design more effective and efficient architecture?

*These two networks have same stride with hourglass*
Explore new architecture

- We use automatic neural network design approach BlockQNN to generate optimal model on keypoints task.
- We search the best model on MPII dataset and transfer it to coco challenge.
- It costs 5 days to complete the searching process with only 32 GPUs.

Design Network Blocks by Q-learning

Accuracy of Q-Learning in Exploration and Exploitation

Comparison with state-of-the-art methods on CIFAR-10 and CIFAR-100

Explore new architecture

- Due to time limited, we only verify the result of hourglass 2 stacks and the generated network
- The generated network has less number of parameter
- We evaluate them on validation set with ground truth box

<table>
<thead>
<tr>
<th>Method</th>
<th>AP (validation)</th>
<th>Parameter Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hourglass 2 stacks</td>
<td>70.1</td>
<td>19M</td>
</tr>
<tr>
<td>The generated network</td>
<td>70.5</td>
<td>17M</td>
</tr>
</tbody>
</table>
Box Candidates Rescoring

- **Traditional method**
  - Sort box candidates by box score
  - Select top k boxes as the result

- **Our method**
  - Sort box candidates by the product of box score and keypoint score
  - Select top k boxes as the result
Box Candidates Rescoring

- Comparison among different rescoring criterion

<table>
<thead>
<tr>
<th>method</th>
<th>AP (validation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Box score</td>
<td>70.3</td>
</tr>
<tr>
<td>Keypoint score</td>
<td>56.1</td>
</tr>
<tr>
<td>Rescoring</td>
<td>71.5</td>
</tr>
</tbody>
</table>

![Box score](image1.png)

![Rescoring](image2.png)
OKS-NMS

• Object Keypoint Similarity (OKS)

\[
OKS_p = \frac{\sum_i \exp\left\{-\frac{d^2_{pi}/2s^2_p\sigma_i^2}{s^2_p\sigma_i^2}\right\} \delta (v_{pi} = 1)}{\sum_i \delta (v_{pi} = 1)}
\]

• Can be seen as “IoU” in keypoint detection to perform NMS
• OKS-NMS fails to suppress proposals with high IoU
• Combine IoU-NMS and OKS-NMS:
  • Apply 0.6 IoU-NMS first, then perform 0.5 OKS-NMS (best practice)

Data Selection

False annotations in COCO dataset
Data Selection

• We statistic the joint distribution of keypoint similarity (KS) (between box center and keypoint center) and keypoint number of an instance

\[ ks(c_{box}, c_{kps}) = e^{-\frac{||c_{box} - c_{kps}||^2}{2*area_{box}}} \]

• We only keep the data right of the line (0.45,1) – (0.65,17)
External data

- We use the **AI Challenge Keypoint Dataset (AICKD)** for joint training
  1. Train a hourglass 8 stacks with COCO only data
  2. Use the model above to select hard examples in AICKD
  3. Joint train with COCO data and hard examples of AICKD
- We only backpropagate the loss of common annotations with COCO for AICKD data
## Experiment Results

<table>
<thead>
<tr>
<th>Method</th>
<th>AP (validation set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hourglass 8 stacks naïve</td>
<td>70.3</td>
</tr>
<tr>
<td>++ data selection</td>
<td>70.8</td>
</tr>
<tr>
<td>++ proposal rescoring</td>
<td>71.5</td>
</tr>
<tr>
<td>++ OKS-NMS</td>
<td>71.7</td>
</tr>
<tr>
<td>++ external data</td>
<td>73.0</td>
</tr>
<tr>
<td>++ ground truth box</td>
<td>75.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Final submission</th>
<th>AP (test-dev / test-challenge)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (single model, COCO + external data)</td>
<td>72.0 / 71.4</td>
</tr>
<tr>
<td>GRMI (COCO + external data)</td>
<td>68.5 / NA</td>
</tr>
</tbody>
</table>

Results Visualization
What we learned?

- For performance improvement of top-down methods, single person pose estimation module is much more important than detection module.
- A direct simple CNN regression model can solve complicated pose estimation problems in COCO dataset, including heavily occlusion, large variance and crowding cases.
- Hourglass shows great performance for single pose estimation task, but it is not the only choice. We expect better results from automatic designed networks in the future.
Our team

Yujie Wang
Changbao Wang
Quanquan Li
Biao Leng
Zhoujun Li
Junjie Yan

Email: liquanquan@sensetime.com
Thank you!