SRN: Stacked Regression Network for Real-time 3D Hand Pose Estimation
Supplementary Material

Pengfei Ren
rpf@bupt.edu.cn
Haifeng Sun
sunhaifeng_1@ebupt.com
Qi Qi
qiqi@ebupt.com
Jingyu Wang
wangjingyu@bupt.edu.cn
Weiting Huang
huangweiting@ebupt.com

State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications,
Beijing, China

1 Qualitative Results

Some qualitative results for NYU [6], ICVL [5] and MSRA [4] datasets are shown in Fig. 1. Specifically, in the last three columns, we show some error annotations in ICVL [5] and MSRA [4] datasets. On Hands17 [8], we chose some special cases, including low-quality images, extreme views and self-occlusion, to demonstrate the robustness of our method. As is shown in Fig. 2, for the complex cases, our method can still obtain accurate and reasonable pose, which indicates that our method can capture global constraints and correlations among different joint well.

2 Impact of Refinement

We use a 3 stacked network to further evaluate the impact of the refining stage. Fig. 3 and Fig. 4 shows some examples of the iterative process on Hands17 [8] dataset with inaccurate initialization and partial error initialization, respectively. The first row shows the results of the initialize hand pose, the second to third rows show the refined results on stage 1–2. As is shown in Fig. 3, for some pose and view, initial regression results are inaccurate, which is mainly manifested by the fact that some joints fall outside the hand area. In this case, the inaccurate joints will be fine-tuned in subsequent stages. Furthermore, for some initial estimates with obvious errors in Fig. 4, our method can still obtain satisfying results by re-predicting the joints that are obviously unreasonable.

© 2019. The copyright of this document resides with its authors. It may be distributed unchanged freely in print or electronic forms.
Table 1: Comparison of runtime with state-of-the-art methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Ours</th>
<th>V2V-PoseNet</th>
<th>DenseReg</th>
<th>Point-to-Point</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPU</td>
<td>2080 Ti</td>
<td>Titan X</td>
<td>Titan X</td>
<td>Titan Xp</td>
</tr>
<tr>
<td>Runtime</td>
<td>3.8 ms</td>
<td>285.7 ms</td>
<td>36 ms</td>
<td>23.9 ms</td>
</tr>
<tr>
<td>FPS</td>
<td>263.1</td>
<td>3.5</td>
<td>27.8</td>
<td>41.8</td>
</tr>
</tbody>
</table>

3 Additional Comparison on Runtime

During the testing stage, similar to the [1, 2, 3], we did not take into account the time of intercepting the hand area. Our 2-stack model takes 3.8 ms for one frame in average (263.1 FPS) on a single GeForce RTX 2080 Ti GPU with a batch size 1. Table 1 shows a comparison of runtime to the state-of-the-art methods [1, 2, 3, 7]. Our method outperforms all previous state-of-the-art approaches and the inference time is much less than these methods.

Figure 1: Qualitative results for NYU [6], ICVL [5] and MSRA [4] datasets. We show hand joint locations on depth images. Different hand joints and bones are visualized with different colors. The ground truth hand joint locations are presented in the second row.

References


[2] Liuhao Ge, Zhou Ren, and Junsong Yuan. Point-to-point regression pointnet for 3d hand
Poor quality
Extreme viewpoint
self-occlusion

Figure 2: Qualitative results on Hands17 [8] for complex cases.

Initial pose
Stage 1
Stage 2

Figure 3: Qualitative results on Hands17 [8] of different stages with inaccurate initialization.


Figure 4: Qualitative results on Hands17 [8] of different stages with error initialization.