Hand Pose Estimation via 2.5D Latent Heatmap Regression
Umar Iqbal\textsuperscript{1,2}, Pavlo Molchanov\textsuperscript{2}, Thomas Breuel\textsuperscript{2}, Juergen Gall\textsuperscript{1} and Jan Kautz\textsuperscript{2}
\textsuperscript{1}Computer Vision Group, University of Bonn, Germany \hspace{1em} \textsuperscript{2}NVIDIA Research

Introduction

Motivation
- Sign-language recognition
- VR/AR
- Human-machine interactions
- Gaming

Problem:
- 2D and 3D hand pose estimation

Challenges
- Large amounts of appearance variation and self occlusions
- Occlusion due to interaction with objects
- Complex hand articulations
- 3D pose estimation is an ill-posed problem due to scale and depth ambiguities

Contributions
- A view-agnostic approach for monocular 2D and 3D hand pose estimation
- A 2.5D pose representation that can be estimated easily from an RGB image
- An exact approach to reconstruct 3D hand pose from 2.5D pose
- A 2.5D heatmap representation to enable accurate keypoint localization
- A CNN architecture to regress 2.5D heatmaps in a latent way

Overview

Latent 2.5D Heatmap Regression

2D Heatmaps
- Normalized relative depth
- Loss function

2D Coordinates
- \( p_k = \sum_{p \in \mathcal{P}} H(\mathcal{P}, \mathcal{L}_k)(p) \)

3D Pose Reconstruction

Given 2.5D pose, we need to find the depth \( \tilde{F}_{root} \) of the root keypoint to reconstruct the scale normalized 3D pose.

Given \( \tilde{F}_{2.5D} \) and \( \mathcal{K} \) there exists a unique 3D pose that satisfies:

\[
(\hat{x}_a - \tilde{x}_{root})^2 + (\hat{y}_a - \tilde{y}_{root})^2 + (\hat{z}_a - \tilde{z}_{root})^2 = C^2
\]

The equation can be rewritten in terms of the 2D projections \( (x_a, y_a, \tilde{x}_{root}, \tilde{y}_{root}) \) and relative depths \( (\tilde{x}_{root}, \tilde{y}_{root}) \) as follows:

\[
(x_a - \tilde{x}_{root})^2 + (y_a - \tilde{y}_{root})^2 = (\tilde{z}_{root})^2
\]

The coefficients of the quadratic equation:

\[
\begin{align*}
a &= (x_a - \tilde{x}_{root})^2 + (y_a - \tilde{y}_{root})^2 \\
b &= 2(x_a - \tilde{x}_{root}) + 2(y_a - \tilde{y}_{root}) - 2x_a - 2y_a \\
c &= (x_a - \tilde{x}_{root})^2 + (y_a - \tilde{y}_{root})^2 - (\tilde{z}_{root})^2
\end{align*}
\]

Scale Recovery

\[
\hat{z}_{root} = 0.5(-b + \sqrt{b^2 - 4ac})/a
\]

References