Domain Randomization for Cuboid Pose Estimation

Jonathan Tremblay
jtremblay@nvidia.com
Real-time Multi-Person 2D Pose Estimation using Part Affinity Fields

Cao et al.
CVPR 2017
Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World

Tobin et al., 2017
arXiv:1703.06907
Cuboid Pose Estimation
Data Generation
Data Generation
Data Generation - Domain Randomization

- Unreal Engine 4
- Random distractor objects with random color
- Random lights
- Random solid color or imagenet sample as background
- Random camera position (depth, azimuth, elevation, yaw)

- Pixel location vertexes
- 2d bounding box
- Depth maps
- Labels
Architecture

Image 400x400x3

3x3:64 Conv 3x3:64 Conv Pool 2 3x3:128 Conv 3x3:128 Conv Pool 2 3x3:256 Conv 3x3:256 Conv 3x3:256 Conv Pool 2 3x3:512 Conv

3x3:512 Conv 3x3:512 Conv 3x3:256 Conv 3x3:128 Conv

3x3:128 Conv 3x3:128 Conv 3x3:128 Conv 1x1:512 Conv 1x1:7 Conv

Features Extraction (VGG-19)

Belief Maps
Model Output - Belief Maps
Find Point Locations

- Find local maximum using a sliding window
- Cluster local minimum using DBSCAN
- Select most probable point location
Baxter’s camera
Data - Contrast and Brightness
Regularizing Neural Networks by Penalizing Confident Output Distributions

\[ H(p_\theta(y|x)) = - \sum_i p_\theta(y_i|x) \log(p_\theta(y_i|x)). \]
Baxter’s camera
Surprising Result
Baxter’s camera
Let’s go further

- Using a neural network to output the points pixel location
- Deal with multiple objects
- Working with real world objects
- Why is domain randomization working?
Domain Randomization