Aligning 3D Models to RGB-D Images of Cluttered Scenes

Saurabh Gupta\textsuperscript{1}, Pablo Arbeláez\textsuperscript{2}, Ross Girshick\textsuperscript{3}, Jitendra Malik\textsuperscript{1}

\textsuperscript{1}UC Berkeley, \textsuperscript{2}Universidad de los Andes, Colombia, \textsuperscript{3}Microsoft Research

Improved

[Wu et al.] Z Wu, S Song, A Khosla, F Yu, L Zhang, X Tang, J Xiao

[Silberman et al.] N. Silberman, D. Hoiem, P. Kohli, R. Fergus


[Gupta et al.] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik

Titan GPUs used for this research.

MURI N00014-09-1-1051, ONR MURI N00014-10-10933, and a Berkeley Graduate Fellowship. We gratefully acknowledge NVIDIA corporation for the donation of Tesla and

Richer output

Segmentation

Starting with weak annotation (instance contours)

3D reasoning by initial 2D processing

Input

Output

Related Work

Object Detection and Instance Segmentation for RGB-D Images

Replacing in-place with a 3D model

Coarse Pose Estimation

Object Detection

Semantic Segm.

Instance Segmentation

Pose

3D Model

Sub-categorization

How to manipulate / grasp?

Input RGB / D Image Pair

Use a shallow 3 layer fully convolutional network (average pooling to predict)

Fine Pose Estimation

- Start with a model M, at scale \( s \), an initial pose estimate \( \hat{R}, \hat{t} \)
- Iterative Closed Point (ICP) to optimize for \( \hat{R}, \hat{t} \) (that aligns best to data)
- Render model, use visible points, run ICP between these points, and points in the segmentation mask, re-estimate \( \hat{R}, \hat{t} \), repeat

Pick best model \( M^* \), scale \( s^* \) and pose \( \hat{R}^*, \hat{t}^* \) based on fit to the data

Works reasonably well even though
- Inaccurate models
- Imperfect segmentation masks

Acknowledgments: The authors would like to thank Shubham Tulsiani and Bharath Hariharan for their valuable comments. This work was supported by ONR SMARTS MURI N00014-10-10933, and a Berkeley Graduate Fellowship. We gratefully acknowledge NVIDIA corporation for the donation of Tesla and

![Input](Image)

![Output](Image)

![Related Work](Image)

![Object Detection and Instance Segmentation for RGB-D Images](Image)