InSeGAN: A Generative Approach to Segmenting Identical Instances in Depth Images





Anoop Cherian[1] Gonçalo Dias Pais[2] Siddarth Jain[1] Tim K. Marks[1] Alan Sullivan[1]

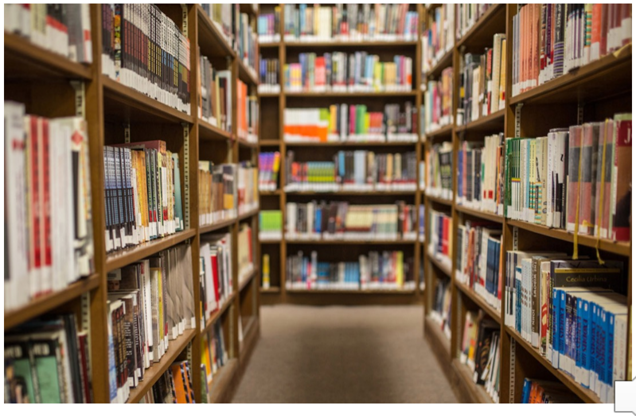
[1]Mitsubishi Electric Research Labs (MERL), Cambridge, MA
[2]Instituto Superior Tecnico, University of Lisbon, Portugal

ICCV Virtual, 2021

Pick an Instance?

Pick an Instance?

Pick an Instance?



Identical Instance Segmentation Problem

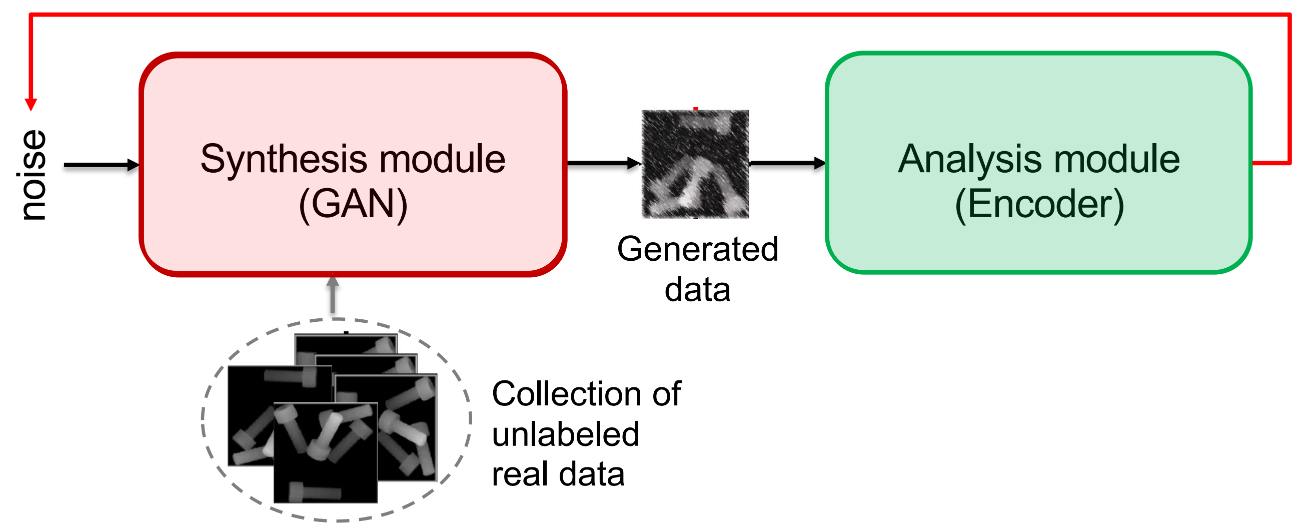
Input: depth image

Output: instance labels

Problem setup:

- A collection of depth images
  - Each with multiple instances of the same rigid object
- Unsupervised: No ground truth instance labels for learning

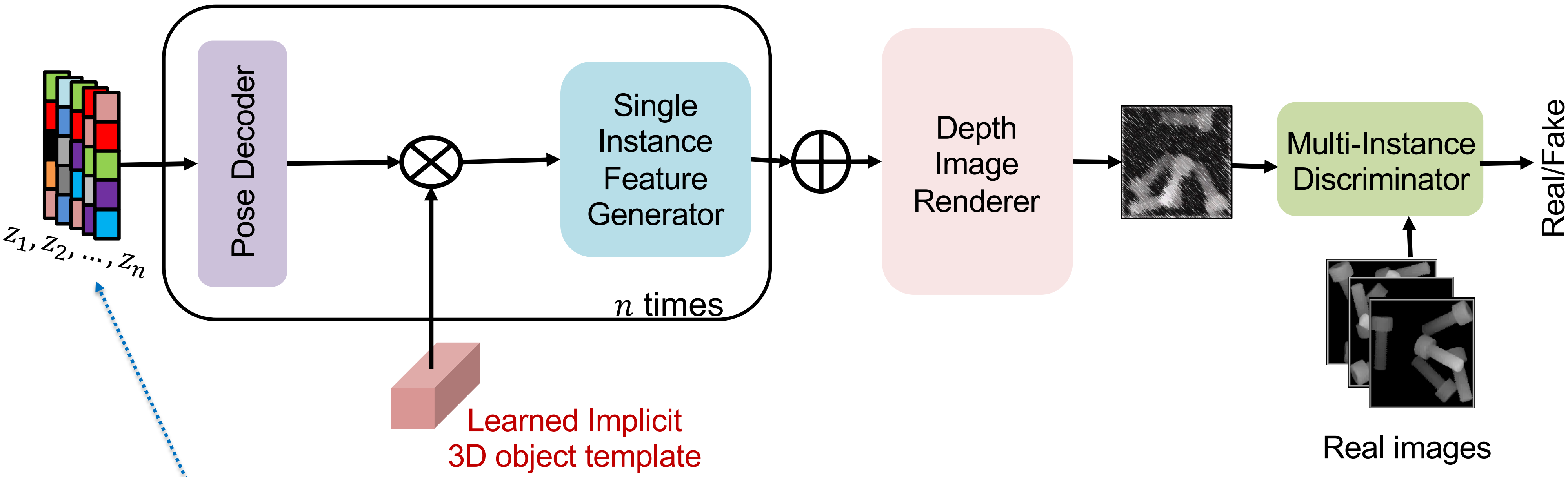
InSeGAN (**In**stance **Se**gmentation **GAN**) Approach: Analysis by Synthesis



Synthesis: GAN learns to produce outputs that look like real depth images.

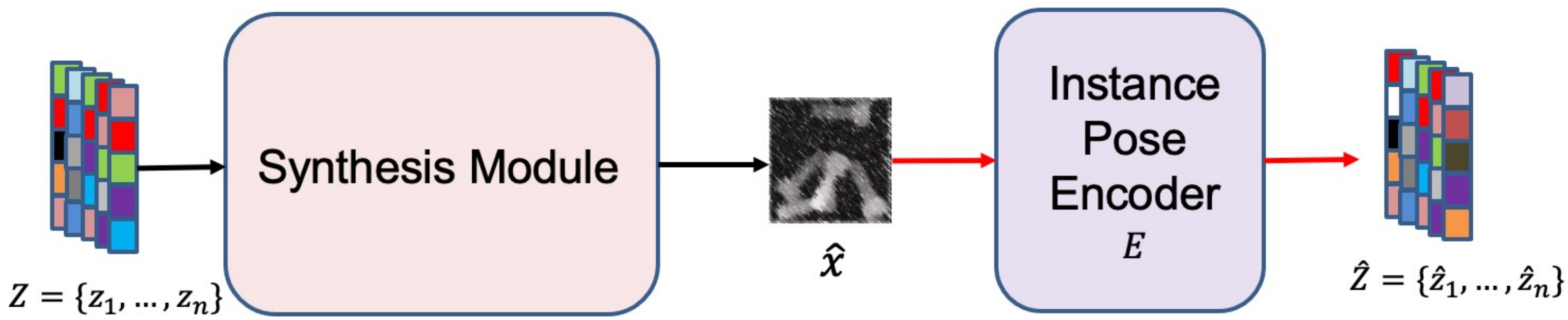
Analysis: Encoder learns to input a realistic depth image and output instance pose parameters.

InSeGAN Generator: Instance Poses to Depth Image



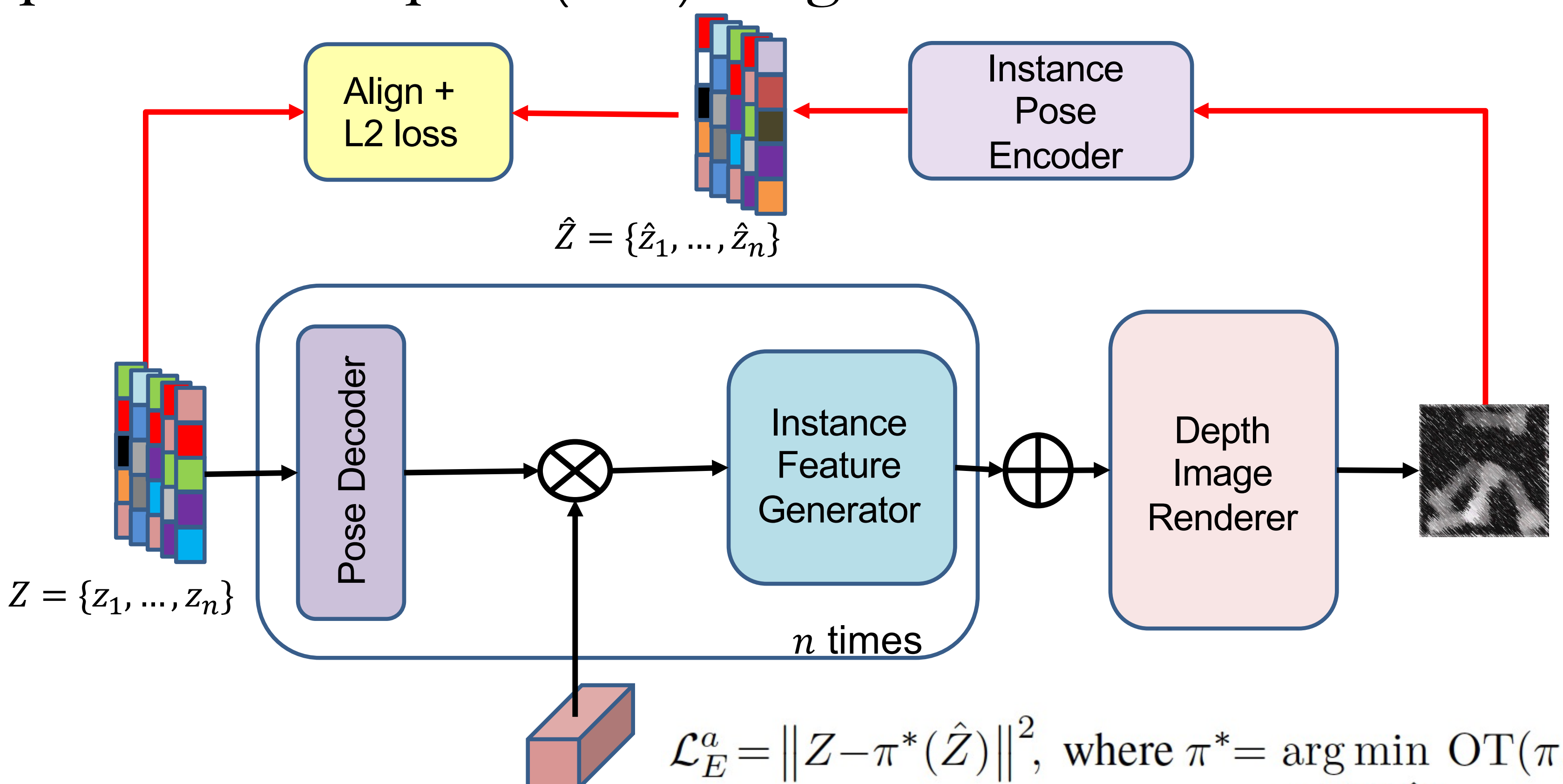
Instead of one noise vector, input is n noise vectors.
System learns to associate each vector with one instance.

InSeGAN Analysis: Depth Image to Instance Poses



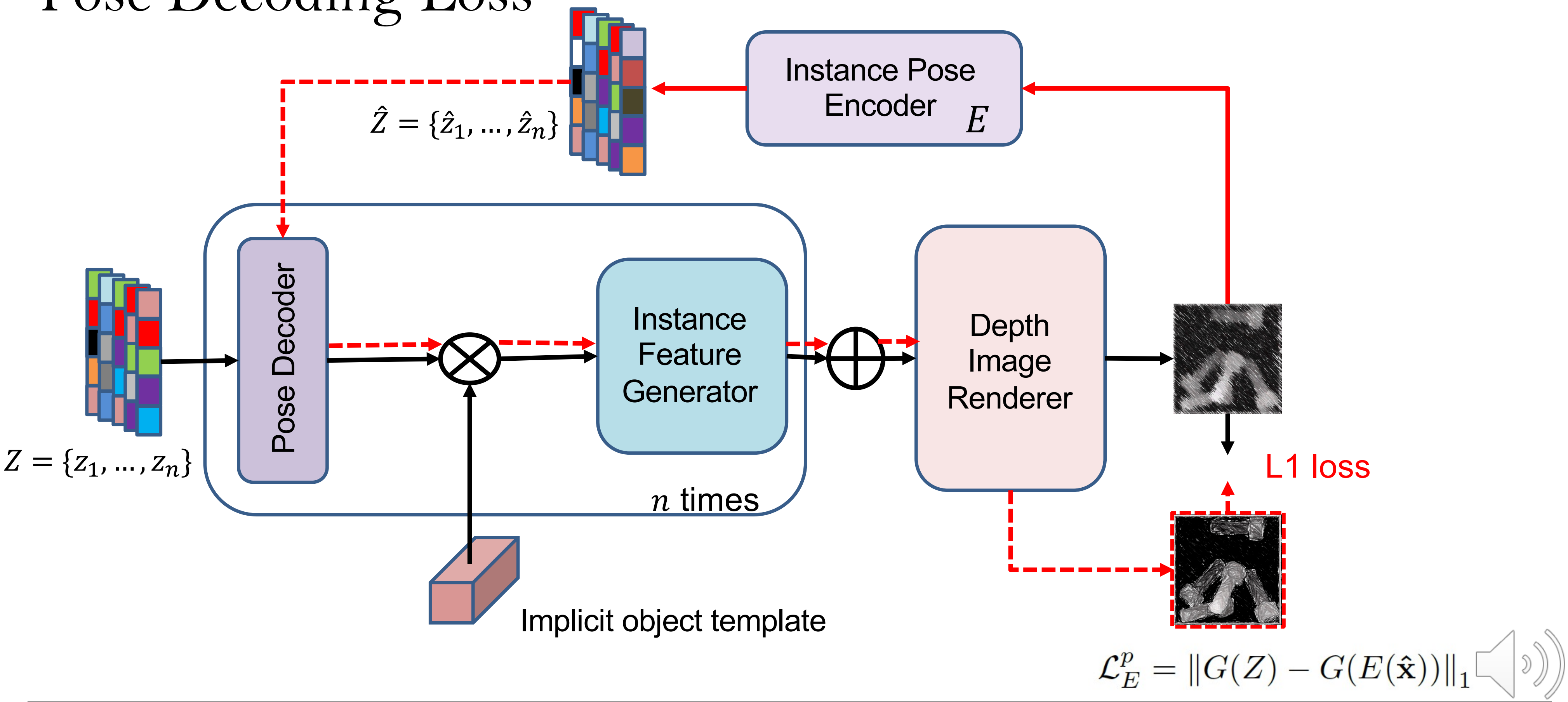
- Instance pose encoder E
  - takes in a synthesized multiple-instance depth image $\hat{x}$
  - produces pose vectors $\hat{Z}$ that could have produced $\hat{x}$.

Optimal Transport (OT) Alignment Loss

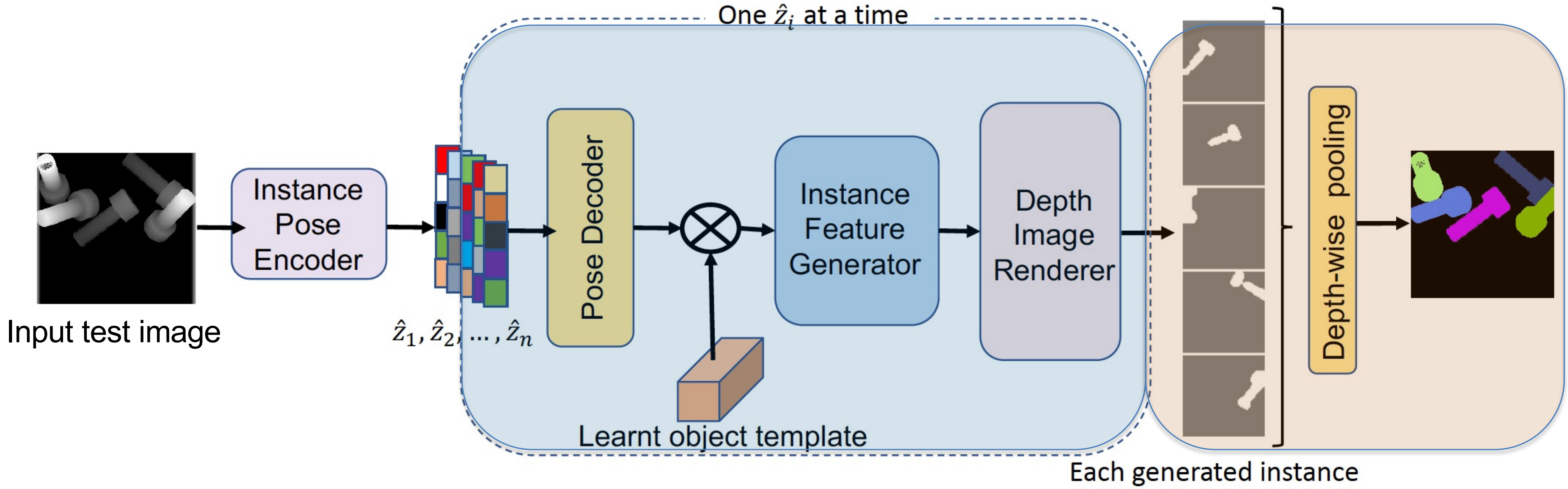


$$\mathcal{L}_E^a = \left\|Z - \pi^*(\hat{Z})\right\|^2, \text{ where } \pi^* = \underset{\pi \in \Pi(Z,\hat{Z})}{\arg\min}\ \mathrm{OT}(\pi, \mathbf{D}(Z, \hat{Z}))$$

Pose Decoding Loss



Inference: Instance Segmentation

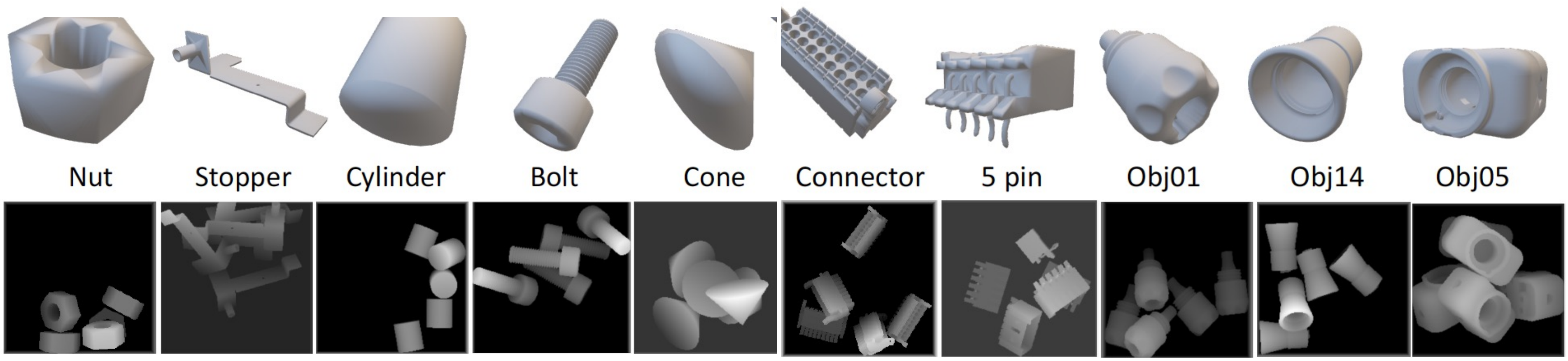


At test time:

- Input depth image is passed through the pose encoder to get the latent vectors
- Latent vectors are then decoded and rendered one at a time, each rendering a single instance
- Rendered instances are thresholded and transformed into segment masks

Experiments and Results

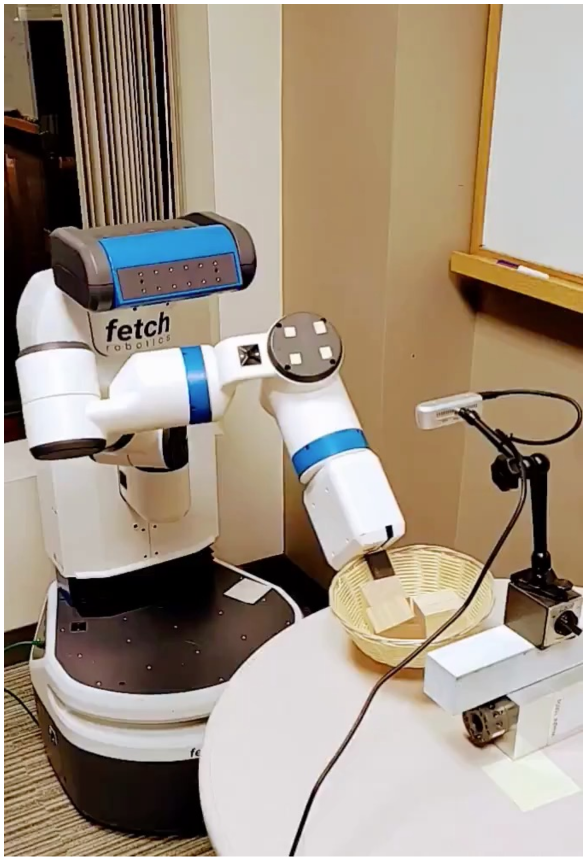
New Insta-10 Synthetic Dataset



Object CAD models (unavailable for training) and the depth images for each class

Insta-10 dataset has **10 object classes**, each defined by a 3D object CAD model
Each class has **10,000 depth images**
Each image has **5 object instances in varying poses, occlusions, etc.**

Real Robotic Data Collection



- We programmed a Fetch robot to shake a box containing 4 wooden blocks. The depth images were captured using a RealSense RGB-D camera.
- The evaluation test images are hand annotated using the LabelMe tool.
- Collected 3,000 depth images. Annotated 62 images for evaluation.

Comparisons to the State of the Art

Method	Nut	Stop.	Cyl.	Bolt	Cone	Conn.	5-pin	Obj01	Obj14	Obj05	Avg mIoU
Non-Deep Learning Methods											
K-Means	0.64	0.297	0.7	0.18	0.35	0.554	0.628	0.208	0.496	0.59	0.464
Spectral Clustering [31]	0.56	0.36	0.54	0.22	0.41	0.56	0.58	0.25	0.47	0.57	0.452
GrabCut [36]+KMeans	0.572	0.232	0.572	0.472	0.231	0.519	0.497	0.597	0.557	**0.605**	0.486
GraphCut [3]	0.569	0.1	0.589	0.447	0.12	0.476	0.12	0.597	0.540	0.511	0.373
Deep Learning Methods											
Wu et al. [41]	0.45	0.28	0.57	0.27	0.33	0.38	0.43	0.23	0.44	0.57	0.385
IODINE [9]	0.026	0.059	0.019	0.040	0.089	0.032	0.034	0.058	0.053	0.118	0.053
Slot Attn. [29]	0.375	0.276	0.535	0.43	**0.68**	**0.662**	0.628	0.655	**0.622**	0.481	0.535
InSeGAN (2D)*(ours)	0.215	**0.365**	0.258	0.524	0.435	0.585	0.628	0.365	0.286	0.532	0.419
InSeGAN (3D) (ours)	**0.773**	0.301	**0.760**	**0.539**	0.47	0.655	**0.642**	**0.686**	0.591	0.483	**0.590**

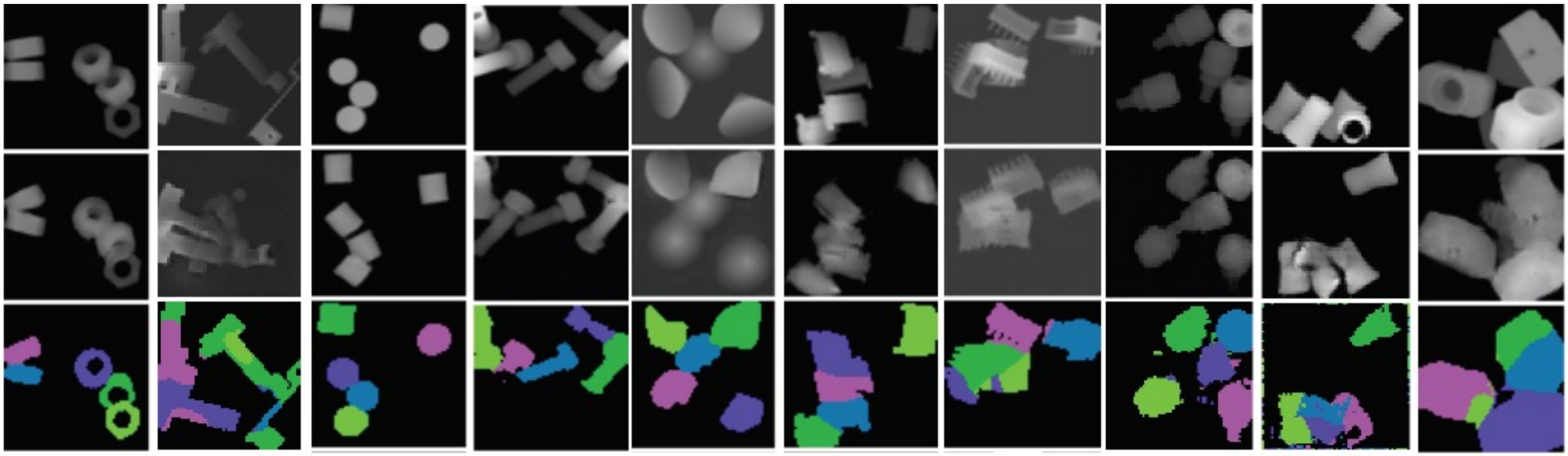
*InSeGAN2D baseline did not use the 3D template or the pose decoder modules, instead using the noise vector to directly produce a single instance feature vector

More Results

Method	mIoU
KMeans	0.797
Spectral Clustering	0.668
Graph Segmentation [7]	0.436
InSeGAN	**0.857**

Results on Real Data

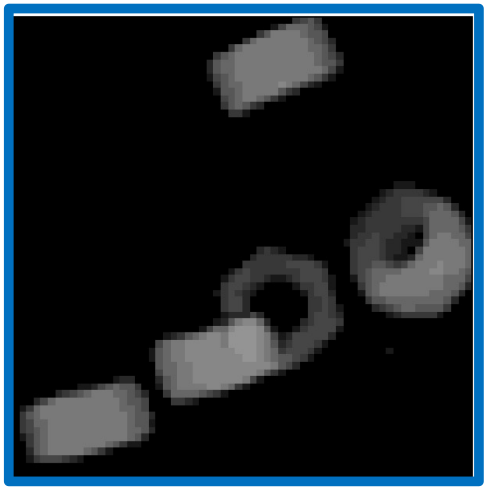
Qualitative Segmentation Results



See paper for more results

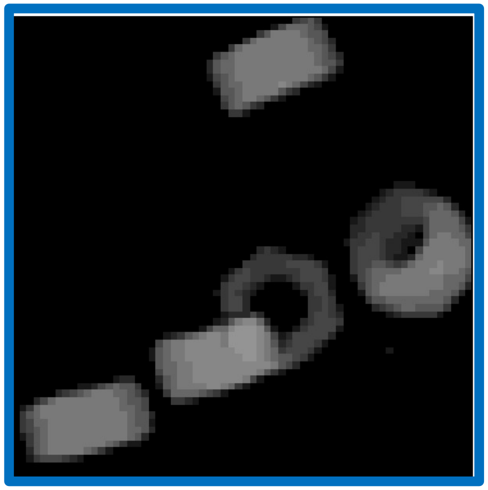
Top row: Input depth image. **Middle row:** Image rendered by InSeGAN. **Last row:** Segmentations produced by InSeGAN.

Instance Pose Disentanglement

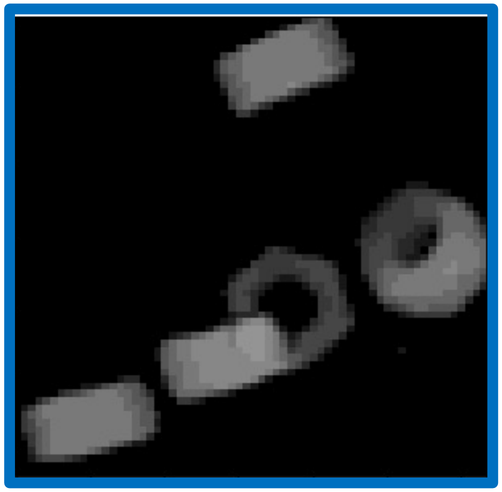


Depth image input

Instance Pose Disentanglement

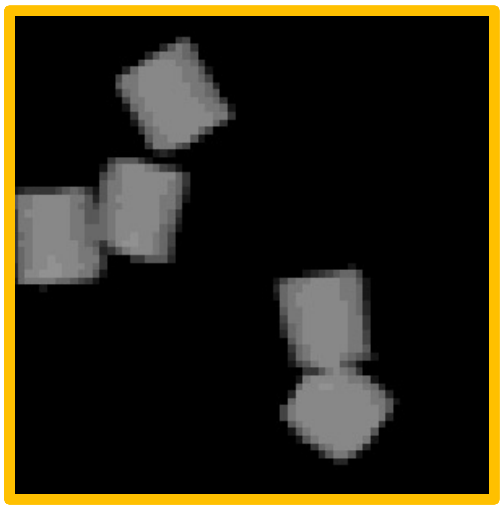


Depth image input

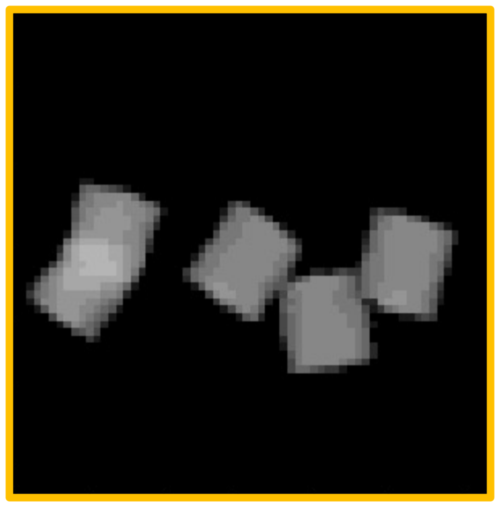


Rotating a single instance

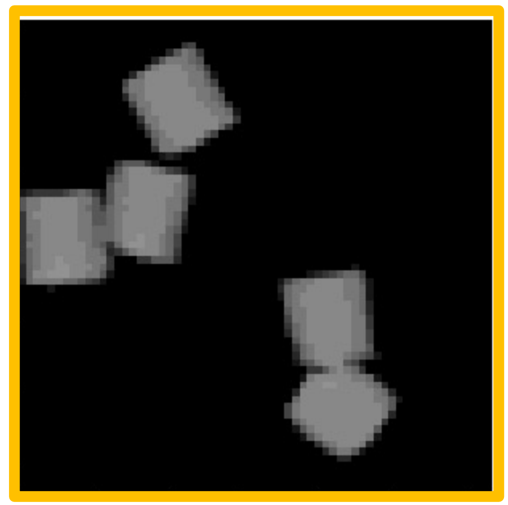
Instance Pose Disentanglement



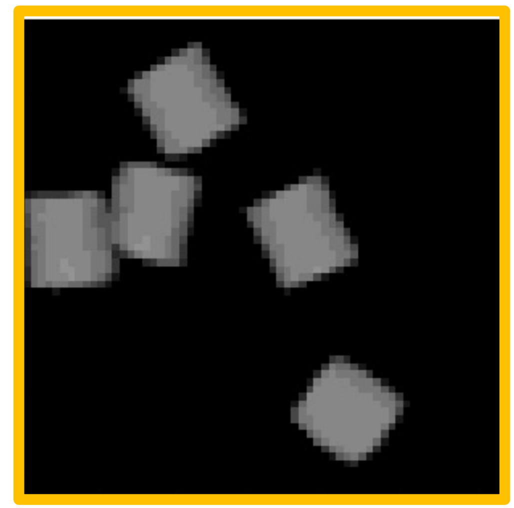
Depth image



Rotating all instances together



Rotating a single instance



Translating a single instance

Thank you!

For questions, please contact us at cherian@merl.com

PyTorch implementation of InSeGAN and the Insta-10 dataset are publicly available at https://www.merl.com/research/