Amodal 3D Reconstruction for Robotic Manipulation via Stability and Connectivity

William Agnew [1] **Christopher Xie** [1] **Aaron Walsman** [1] **Octavian Murad** [1] **Caelen Wang** [1] **Pedro Domingos** [1] **Siddhartha Srinivasa** [1]

Abstract

Learning-based 3D object reconstruction enables single- or few-shot estimation of 3D object models. For robotics this holds the potential to allow model-based methods to rapidly adapt to novel objects and scenes. Existing 3D reconstruction techniques optimize for visual reconstruction fidelity, typically measured by chamfer distance or voxel IOU. We find that when applied to realistic, cluttered robotics environments these systems produce reconstructions with low physical realism, resulting in poor task performance when used for model-based control. We propose ARM an amodal 3D reconstruction system that introduces (1) an object stability prior over the shapes of groups of objects, (2) an object connectivity prior over object shapes, and (3) a multi-channel input representation and reconstruction objective that allows for reasoning over relationships between groups of objects. By using these priors over the physical properties of objects, our system improves reconstruction quality not just by standard visual metrics, but also improves performance of model-based control on a variety of robotics manipulation tasks in challenging, cluttered environments.

1. Introduction

There has been a surge of interest in learning and using object representations for reinforcement learning and control. By learning to predict future observations or using encoder-decoder architectures with object priors, several recent works learn object masks or 2D keypoints (Goel et al., 2018; Zhu et al., 2018; Greff et al., 2019; Kulkarni et al., 2019; Anand et al., 2019; Lin et al., 2020; Agnew & Domingos, 2020; Van Steenkiste et al., 2018; Kosiorek et al., 2018), enabling significant improvements in downstream task sample efficiency. However, in 3D environments, 2D object representations are insufficient for understanding important features such as relative distances and contacts between objects. In addition, all of these methods require many environment observations to learn object representations, limiting their ability to rapidly adapt to novel objects. Recent years have seen major advances in reconstructing 3D object models from images (Wu et al., 2017; Richter & Roth, 2018; Wang et al., 2018; Kato et al., 2018; Smith et al., 2019; Gkioxari et al., 2019; Tian et al., 2019; Kanazawa et al., 2018; Zhang et al., 2018; Tatarchenko et al., 2017; Yingze Bao et al., 2013; Smith et al., 2018; Mescheder et al., 2019). We find that directly applying state-of-the-art unseen object reconstruction techniques (Zhang et al., 2018) to cluttered environments frequently fails to reconstruct objects in regions occluded by distractor objects, leading to physically unstable models. These low-quality reconstructions often lead to poor performance in downstream manipulation problems. In this paper we propose ARM, an object reconstruction system which incorporates priors over the physical properties of objects to produces 3D reconstructions with high physical fidelity from a single RGBD image. Through experiments we show that ARM can produce models of sufficient physical quality for robotic planning and manipulation even in cluttered scenes of objects unseen during training. In summary, our contributions are: **(1)** Objects are generally stable unless being manipulated. We introduce

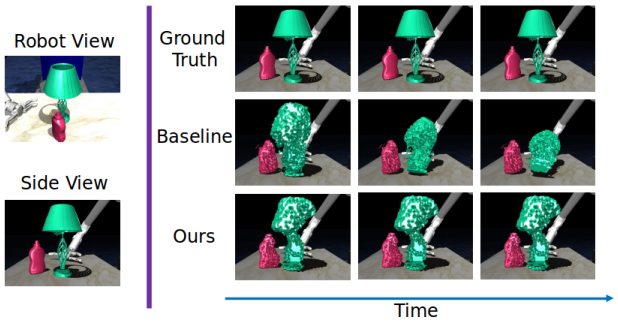

Figure 1. Comparison of physical behaviors of reconstructions from different algorithms.

[1]School of Computer Science and Engineering, University of Washington. Correspondence to: William Agnew <wagnew3@cs.washington.edu>.

Workshop on Object-Oriented Learning at ICML 2020.

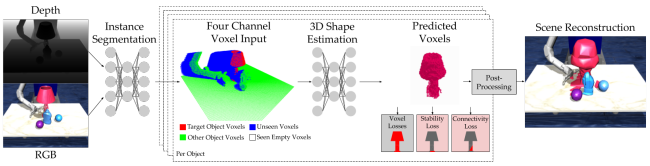

Figure 2. Reconstruction system overview

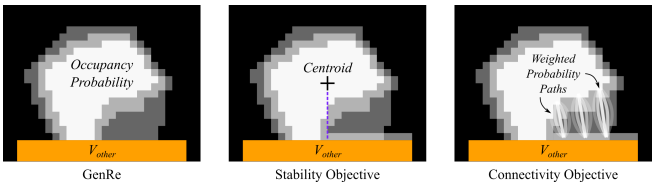

Figure 3. Impact of stability and connectivity objectives. Left: occupancy probabilities of an estimated shape, in greyscale. Adding the stability objective makes the object stable, and adding the connectivity objective fills in the gap between the shape and inferred base.

a novel differentiable loss function that penalizes unstable reconstructions, thus encouraging stable scenes. **(2)** Objects are connected. We introduce a novel differentiable loss function that penalizes disconnectedness to reconstruct connected objects. **(3)** Reconstruction requires reasoning not only about the object being reconstructed, but also about how it interacts with other objects in the scene. Thus, we introduce a multi-channel scene representation that allows reconstruction in the context of the spatial extent of other objects in the scene.

2. Amodal 3D Reconstruction

2.1. System Architecture

In this section we describe the architecture of ARM, our 3D reconstruction architecture, which consists of four stages. First, we apply an instance segmentation network to the input RGB-D image. Second, for each object we detect, we compute its *four channel representation*, defined below. In the third stage, ARM uses this representation to perform 3D reconstruction with a deep network. In the final stage, we apply post-processing steps to obtain mesh representations suitable for physics simulations.

Instance Segmentation ARM takes as input an RGB image, $I \in R^{h \times w \times 3}$, and an organized point cloud, $P \in R^{h \times w \times 3}$ computed by backprojecting a depth image with camera intrinsics. This is passed to an instance segmentation network $\mathcal{S}$ which outputs instance masks $L = \mathcal{S}(I, D) \in \mathcal{L}^{h \times w}$, where $\mathcal{L} = \{0, \ldots, K\}$ and K is the number of detected object instances. We use UOIS-Net (Xie et al., 2019) as $\mathcal{S}$ which produces high quality segmentations for unseen objects.

Four Channel Representation We introduce a four-channel voxel representation to enable ARM to reconstruct shape in the context of the spatial extent of other objects in the scene. For each object $o \in \mathcal{L}$, we compute its voxel occupancy grid $F_o \in \{0, 1\}^{d^3 \times 4}$ augmented with the surrounding objects, as well as with voxel visibilities with respect to the camera. The first channel of F_o is the voxel grid of object o alone, which is computed by voxelizing P_o, where P_o is the point cloud segmented with the instance mask for o. Channel two is the voxel grid of all other objects in L except for o. The third channel consists of empty voxels, and the final channel contains a voxel grid of occluded voxels. Note that the third and fourth channels are computed using the camera extrinsics and intrinsics. The bounding box volume of F_o is centered at the center of mass of object o and has side length $k\delta_o$, where δ_o is the maximum distance between points in P_o. In our implementation, $k = 4$. Finally, we normalize our bounding box volume so the table occupies the $z = 0$ plane in our voxel grid by finding the z plane of height z_{table} in $P \setminus P_o$ with the most set voxels and shift F_o downwards by z_{table}.

3D Reconstruction For each object $o \in \mathcal{L}$, we use F_o as input to a 3D reconstruction network $\mathcal{C}$ which outputs the probability of o's presence at each voxel as $\mathcal{C}(F_o) = V_o \in [0, 1]^{d^3}$. We use GenRe-Oracle (Zhang et al., 2018) as our 3D reconstruction network, increasing the channels on the input convolutions to handle the additional channels in our input representation. Finally, we use marching cubes (Lorensen & Cline, 1987) after thresholding V_o to transform the output voxel probabilities into a mesh.

Post-Processing We apply post-processing steps to meshes in order to make them suitable for physics simulation. First, we remove intersections between object meshes by re-voxelizing the meshes and removing any intersecting voxels. Finally, we compute an approximate convex decomposition of each mesh using V-HACD (Mamou et al., 2016).

2.2. Loss Functions

GenRE (Zhang et al., 2018) uses a weighted combination of cross entropy and a surface loss between reconstructed and ground truth voxels to train their 3D reconstruction network. However, in robotic settings, optimizing these losses alone are not sufficient to solve the downstream task of robotic manipulation, as we show in Section 3.3. This results in reconstructions that often fail at reconstructing portions of objects in occluded regions, leading to poor physical fidelity during the planning phase. We tackle this issue by designing auxiliary loss functions based on two reasonable assumptions: 1) objects and scenes are stable

prior to manipulation, and 2) objects are a single connected component. This motivates us to design loss functions that encourage stability and connectivity of the reconstructed objects.

2.2.1. Stability Loss

Objects are almost always stable unless being actively manipulated. This provides a prior over object shape, even in occluded regions, by allowing inference of hidden supports objects may have. An object is in static equilibrium if the net forces acting upon it are equal to zero (Urone et al.). This means that the center of mass is within the base of support of an object: along every direction s perpendicular to the force of gravity, the center of mass is behind a pivot point, or point where the object rests on another object.

Let $V_o \in [0,1]^{d^3}$ be the parameters of a multivariate Bernoulli distribution $\mathbf{V}$ over binarized voxel grids $\{0,1\}^{d^3}$. Let $v \sim \mathbf{V}$ be a sample from $\mathbf{V}$. Let $i \in d^3$ be a voxel index. Let $M(v)$ be the center of mass of v. Let S be the set of all vectors perpendicular to the force of gravity g and in the ground plane. For each $s \in S$, let i^s and $M^s(v)$ be the projections of i and $M(v)$ onto the plane defined by s and g. Let V_o be the parameters of $\mathbf{V_{\bar{o}}}$, the distribution of voxels belonging to objects other than o, and $v_{\bar{o}} \sim \mathbf{V_{\bar{o}}}$ be a sample from this distribution. For each direction s and voxel index i, i may be supported by several other voxels, either by being directly above those voxels or by leaning against those voxels in direction s. Let $H_s(i)$ be the set of voxels belonging to other objects supporting i in direction s. Let $E(v) = 1$ if v is stable, and 0 otherwise. Then the probability that v is stable is

$$P(E(v)) = \prod_{s \in S} (1 - u_s)(1)$$

$$u_s = \prod_{i \in d^3} 1 - V(i) P(i^s > M^s(v)) h_s(i)$$

$$h_s(i) = 1 - \prod_{i' \in H_s(i)} 1 - V_{\bar{o}(i')}$$

u_s is the probability that v is unstable in direction s, and is the chance that every voxel i is unstable; that is i either doesn't exist, doesn't support v along direction s, or isn't supported itself. By deriving the probability that a voxel grid sampled from a distribution over voxels grids is stable, we may apply this loss to standard 3D reconstruction networks such as GenRE to encourage learning to output stable meshes. Equation (1) is intractable, so in order to take the gradient we introduce independence assumptions and other approximations to derive an efficiently computable derivative of object stability with respect to each object voxel:

$$\frac{d \log P(E(v))}{dV(i)} = \sum_{s \in S} \frac{-u'_s}{1 - u_s}$$

$$u'_s = -P(i^s > M^s(v)) \hat{h}_s(i) \prod_{i_o \in d^3, i_o \neq i} 1 - P(i_o^s > M^s(v))(V(i_o) \geq \frac{1}{2}) \hat{h}_s(i_o)$$

$$\hat{h}_s(i_o) = 1 - \prod_{i_b \in H_s(i)} 1 - (V_{\bar{o}(i_b)} \geq \tfrac{1}{2})$$

This gradient captures several intuitive properties of stability. If an object has even a single voxel supporting it in a particular direction then it is stable. If a single supporting voxel i is present, then u'_s is close to zero, and the denominator of the derivative will be large compared to when no supporting voxel is present and u'_s is close to 1. This captures the relationship that when no supporting voxels are present, the effect on stability of adding a voxel is large, but when supporting voxels are present, the effect is small.

2.2.2. Connectivity Loss

Objects are generally connected wholes. This imposes a prior on object shape even in occluded regions by allowing us to infer connections between disjoint parts of observed objects. This prior complements the stability objective which frequently infers occluded bases of objects. A voxel grid v is connected if for every pair of voxels $a, b \in d^3$, there exists a path $t = \{i_0, i_1, \dots\}$ between a and b. The probability that a path t exists in v is $P(t) = \prod_{i \in t} V(i)$. Let $T(a,b)$ be the set of all possible paths between a and b. Let $C(v) = 1$ if v is connected, and 0 otherwise, and $C(a,b) = 1$ if there is a path between a and b, and 0 otherwise. Then the probability that v is connected is:

$$P(C(v)) = \prod_{a,b \in d^3, a \neq b} V(a)V(b)(1 - \prod_{t \in T(a,b)} 1 - P(t)) + 1 - V(a)V(b)$$

The derivative of this equation is intractable because it requires considering every path t between every vertex pair (a,b). To resolve this we note that relative to the most likely path t^* between a, b most paths have vanishingly small probability. Thus, for any other point c, we may ignore low probability paths passing through c when calculating their contribution to the connectivity of a and b and only consider the most likely path from a to b passing through c. We derive the following efficiently comutable per-voxel derivative of connectivity:

$$\frac{d \log P(\mathbb{C}(\mathbb{V}))}{dV(c)} = \sum_{a,b \in d^3, a \neq b \neq c} \frac{V(a)V(b) \frac{d}{dV(c)} P(C(a,b))}{V(a)V(b)P(C(a,b)) + 1 - V(a)V_{\bar{o}(b)}}$$

Where $P(C(a,b)) = \bigvee_{t \in T(a,b)} P(t) \approx P(t^* \vee t^c)$, t^s is the path from a to b with the highest probability of existing, and t^c is the path from a to b that includes c with the highest probability of existing.

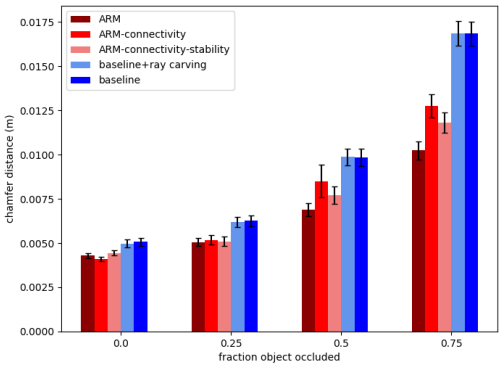

Figure 4. Chamfer distances on held-out models, broken down by observation occlusion. Error bars are a 90% confidence interval.

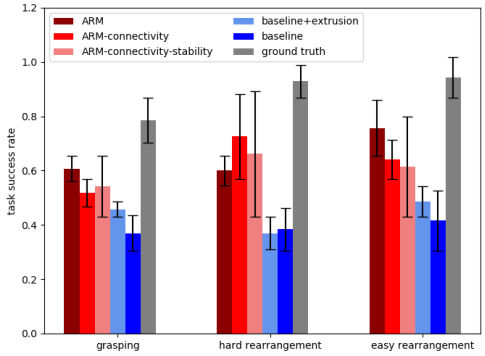

Figure 5. Robot manipulation success rates. Error bars are a 90% confidence interval.

3. Experiments

Implementation and Training We implement ARM using UOIS (Xie et al., 2019) for instance segmentation, and the GenRE depth backbone (Zhang et al., 2018) for shape prediction. We use MuJoCo (Todorov et al., 2012) as a physics simulator for our reconstructed environment. To train ARM, we create a large dataset of cluttered tabletop scenes in MuJoCo using shapenet tables and objects. For each scene we select a random table and drop between 5 and 20 randomly selected objects onto a random point on the table. We then render several views with randomized camera positions and targets. We trained each network on 80,000 reconstruction instances drawn from this dataset.

Reconstruction Quality We compare the visual reconstruction quality of ARM to several baselines on reconstruction of cluttered scenes generated with held-out test objects in Figure 4. The first baseline is GenRE given ground truth depth (Zhang et al., 2018). We also examine GenRE given ground truth depth with voxel carving, or removing all empty voxels after shape prediction. In addition, we conduct ablations on the stability and connectivity priors. We find that ARM performs 22% better overall, and even better on highly occluded objects.

Robot Manipulation We created a suite of robotics manipulation tasks across an range of challenging objects in cluttered scenes. We consider three robot tasks: grasping, pushing, and rearrangement. We conduct each task on 13 different objects from the YCB dataset (Calli et al., 2015) and from a set of challenging, highly non-convex objects downloaded from 3D repositories. For each task and object, we add distractor objects to occlude the target manipulation object. We solve each task by using ARM to reconstruct the scene and MPPI to generate a plan in the reconstructed environment which we then execute in the ground truth environment. Figure 5 shows average success rates of the manipulation tasks. In addition to GenRE, we introduce an extrusion baseline: we take the shape predicted by GenRE and extrude it to the table surface, guaranteeing stability. All of our proposed models outperform both baselines by 20-40% across all three tasks. To highlight the effects of the different 3D reconstruction networks, we use ground truth segmentations for these experiments.

4. Conclusion

Directly applying 3D object reconstruction methods to robot scenes often produces poor reconstructions, especially in clutter. When used for 3D manipulation tasks, the resulting plans are often unsuccessful. In this paper, we use a multiobject stability prior, a connectivity prior, and a novel input representation which allows for multiobject reasoning to solve this problem. The 3D reconstructions our system generates are not only better by standard visual loss metrics, but most importantly they allow for significantly better robot task performance in challenging cluttered scenes. While much recent work in learning-based 3D reconstruction has focused on network architecture, our input representation and loss functions are agnostic to architecture, allowing our ideas to be used in any 3D reconstruction system. We presented a system for producing complete and physically faithful 3D reconstructions of scenes from single images. These reconstructions enable one-shot planning with MPPI for robotics tasks, and we hope to extend these results to one-shot learning of policies. Unlike other approaches to obtaining object representations which used unsupervised techniques to learn the representation in parallel to action execution, we pretrain ARM on a large synthetically generated dataset to obtain a system that immediately produces accurate and detailed object representations. We hope this will spur interest in pretraining-based approaches, and we are interested in future work in combining the ability of pretraining approaches to immediate produce good representations with that of unsupervised approaches to fine tune representations to specific tasks.

References

Agnew, W. and Domingos, P. Self-supervised object-level deep reinforcement learning. *arXiv preprint arXiv:2003.01384*, 2020.

Anand, A., Racah, E., Ozair, S., Bengio, Y., Côté, M.-A., and Hjelm, R. D. Unsupervised state representation learning in atari. In *Advances in Neural Information Processing Systems*, pp. 8766–8779, 2019.

Calli, B., Singh, A., Walsman, A., Srinivasa, S., Abbeel, P., and Dollar, A. M. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 international conference on advanced robotics (ICAR)*, pp. 510–517. IEEE, 2015.

Gkioxari, G., Malik, J., and Johnson, J. Mesh r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9785–9795, 2019.

Goel, V., Weng, J., and Poupart, P. Unsupervised video object segmentation for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 5688–5699, 2018.

Greff, K., Kaufmann, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., and Lerchner, A. Multi-object representation learning with iterative variational inference. *arXiv preprint arXiv:1903.00450*, 2019.

Kanazawa, A., Tulsiani, S., Efros, A. A., and Malik, J. Learning category-specific mesh reconstruction from image collections. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 371–386, 2018.

Kato, H., Ushiku, Y., and Harada, T. Neural 3d mesh renderer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3907–3916, 2018.

Kosiorek, A., Kim, H., Teh, Y. W., and Posner, I. Sequential attend, infer, repeat: Generative modelling of moving objects. In *Advances in Neural Information Processing Systems*, pp. 8606–8616, 2018.

Kulkarni, T. D., Gupta, A., Ionescu, C., Borgeaud, S., Reynolds, M., Zisserman, A., and Mnih, V. Unsupervised learning of object keypoints for perception and control. In *Advances in Neural Information Processing Systems*, pp. 10723–10733, 2019.

Lin, Z., Wu, Y.-F., Peri, S. V., Sun, W., Singh, G., Deng, F., Jiang, J., and Ahn, S. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition. *arXiv preprint arXiv:2001.02407*, 2020.

Lorensen, W. E. and Cline, H. E. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987.

Mamou, K., Lengyel, E., and Peters, A. Volumetric hierarchical approximate convex decomposition. *Game Engine Gems 3*, pp. 141–158, 2016.

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., and Geiger, A. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4460–4470, 2019.

Richter, S. R. and Roth, S. Matryoshka networks: Predicting 3d geometry via nested shape layers. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1936–1944, 2018.

Smith, E., Fujimoto, S., and Meger, D. Multi-view silhouette and depth decomposition for high resolution 3d object representation. In *Advances in Neural Information Processing Systems*, pp. 6478–6488, 2018.

Smith, E. J., Fujimoto, S., Romero, A., and Meger, D. Geometrics: Exploiting geometric structure for graph-encoded objects. *arXiv preprint arXiv:1901.11461*, 2019.

Tatarchenko, M., Dosovitskiy, A., and Brox, T. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2088–2096, 2017.

Tian, Y., Luo, A., Sun, X., Ellis, K., Freeman, W. T., Tenenbaum, J. B., and Wu, J. Learning to infer and execute 3d shape programs. *arXiv preprint arXiv:1901.02875*, 2019.

Todorov, E., Erez, T., and Tassa, Y. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012.

Urone, P. P., Dirks, K., and Sharma, M. *Statics and Torque*, pp. 289–316. OpenStax.

Van Steenkiste, S., Chang, M., Greff, K., and Schmidhuber, J. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. *arXiv preprint arXiv:1802.10353*, 2018.

Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., and Jiang, Y.-G. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 52–67, 2018.

Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, W. T., and Tenenbaum, J. B. MarrNet: 3D Shape Reconstruction

via 2.5D Sketches. In *Advances In Neural Information Processing Systems*, 2017.

Xie, C., Xiang, Y., Mousavian, A., and Fox, D. The best of both modes: Separately leveraging rgb and depth for unseen object instance segmentation. *arXiv preprint arXiv:1907.13236*, 2019.

Yingze Bao, S., Chandraker, M., Lin, Y., and Savarese, S. Dense object reconstruction with semantic priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1264–1271, 2013.

Zhang, X., Zhang, Z., Zhang, C., Tenenbaum, J. B., Freeman, W. T., and Wu, J. Learning to Reconstruct Shapes from Unseen Classes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Zhu, G., Huang, Z., and Zhang, C. Object-oriented dynamics predictor. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 9804–9815. Curran Associates, Inc., 2018.