Synergies Between Affordance and Geometry: 6-DoF Grasp Detection via Implicit Representations

Zhenyu Jiang[1] Yifeng Zhu[1] Maxwell Svetlik[1] Kuan Fang[2] Yuke Zhu[1]
[1]The University of Texas at Austin [2]Stanford University

Abstract

Grasp detection in clutter requires the robot to reason about the 3D scene from incomplete and noisy perception. In this work, we draw insight that 3D reconstruction and grasp learning are two intimately connected tasks, both of which require a fine-grained understanding of local geometry details. We thus propose to utilize the synergies between grasp affordance and 3D reconstruction through multi-task learning of a shared representation. Our model takes advantage of deep implicit functions, a continuous and memory-efficient representation, to enable differentiable training of both tasks. We train the model on self-supervised grasp trials data in simulation. Evaluation is conducted on a clutter removal task, where the robot clears cluttered objects by grasping them one at a time. The experimental results in simulation and on the real robot have demonstrated that the use of implicit neural representations and joint learning of grasp affordance and 3D reconstruction have led to state-of-the-art grasping results. Our method outperforms baselines by over 10% in terms of grasp success rate. Additional results and videos can be found at https://sites.google.com/view/giga2021

1. Introduction

Generating robust grasps from raw perception is an essential task for robots to physically interact with objects in unstructured environments. Here we consider the problem of 6-DoF grasp detection in clutter from 3D point cloud of the robot's on-board depth camera. Our goal is to predict a set of candidate grasps on a clutter of objects from partial point cloud for grasping and decluttering.

Robot grasping is a long-standing challenge with decades of research. Pioneer work [6, 17, 20] has cast it as a *geometry-centric* task, typically assuming access to the full 3D model of the objects. In practice, the requirement of ground-truth models has impeded their applicability in unstructured scenes. Motivated by the new development of machine learning, in particular deep learning, recent work on grasping has shifted focus towards a *data-driven* paradigm [1, 9, 11, 19, 21], where deep networks are trained end-to-end on large-scale grasping datasets, either through manual labeling [8] or self-exploration [9, 16]. Data-driven methods have enabled direct grasp prediction from noisy perception. However, end-to-end deep learning for grasping often suffers from limited generalization within the training domains.

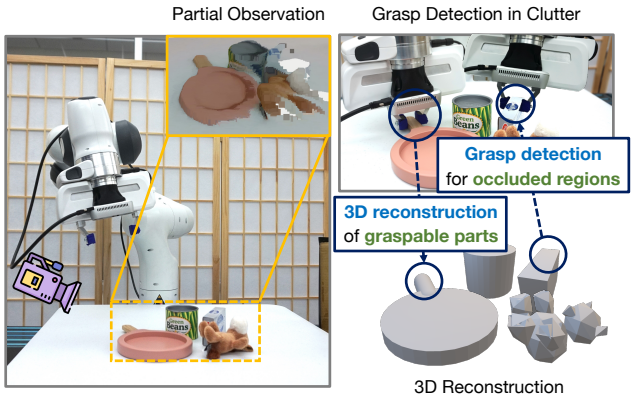

Figure 1: We harness the synergies between affordance and geometry for 6-DoF grasp detection in clutter. Our model jointly learns grasp affordance prediction and 3D reconstruction. Supervision from reconstruction facilitates our model to learn geometrically-aware features for accurate grasps in occluded regions from partial observation. Supervision from grasp, in turn, produce better 3D reconstruction in graspable regions.

Inspired by the two threads of research on geometry-centric and data-driven approaches to grasping, we investigate the synergistic relations between geometry reasoning and grasp learning. Our key intuition is that *a learned representation capable of reconstructing the 3D scene encodes relevant geometry information for predicting grasp points and vice versa*. In this work, we develop a unified learning framework that enables a shared scene representation for both tasks of grasp prediction and 3D reconstruction, where grasps are represented by a landscape of grasp affordance over the scene. By grasp affordance, we refer to

the likelihood of grasp success and the corresponding grasp parameters at each location.

The primary challenge here is to develop a shared representation that effectively encodes 3D geometry and grasp affordance information. Recent work from the 3D vision and graphics communities has shed light on the merits of implicit representations for geometry reasoning tasks [12, 14, 3, 15]. Deep implicit functions define a scene by a deep network that maps each spatial location to corresponding geometry quantities, such as occupancy [12], signed distance functions [14], probability density, and emitted color [10, 13]. Implicit neural representations are able to represent smooth surfaces in high resolution. They are differentiable and continuous.And the representations parameterized by deep networks can adaptively allocate computational budgets to regions of importance.

To this end, we introduce our model: Grasp detection via Implicit Geometry and Affordance (GIGA). We develop a structured implicit neural representation for 6-DoF grasp detection. Our method extracts structured feature grids from the Truncated Signed Distance Function (TSDF) voxel grid fused from the input depth image. A local feature can be computed from the feature grids given a query 3D coordinate. This local feature is used by the implicit functions for estimating the grasp affordance (in the form of grasp quality, grasp orientation, and gripper width of a parallel jaw) and the 3D geometry (in the form of binary occupancy) at the query location. The model is jointly trained in simulation with known 3D geometry and self-supervised grasp trials. With multi-task supervision of affordance and geometry, our model takes advantage of the synergies between them for more robust grasp detection.

We conduct experiments on a clutter removal task [2] in simulation and on physical hardware. The ability to reconstruct the 3D scene from a single view enables GIGA to achieve grasp performance on par with multi-view input as employed in prior work [2]. Empirical results have confirmed the benefits of implicit neural representations and the exploitation of the synergies between affordance and geometry. Our model achieved 87.9% and 69.2% grasp success rates on the packed and pile scenes, outperforming 74.5% and 60.7% reported by the state-of-the-art VGN model.

2. Method

We now present GIGA, a learning framework that exploits synergies between affordance and geometry for 6-DoF grasp detection from partial observation. We learn grasp affordance prediction and 3D occupancy prediction jointly with shared feature grids and a unified implicit neural representation. Figure 2 illustrates the overall model architecture.

2.1. Structured Feature Grids

We adopt the encoder architecture from ConvONets [15] and learn to extract structured feature grids from partial observation. Our encoder takes as input a TSDF voxel field and processes it with a 3D CNN layer to obtain a feature embedding for every voxel. Given these features, we construct planar feature representations by performing an orthographic projection onto a canonical plane for each input voxel. Then we aggregate the features of voxels projected onto the same pixel cell using average pooling, which gives us a 2D feature grid. We process each of the feature plane with a 2D U-Net [18] which is composed of a series of down-sampling and up-sampling convolutions with skip connections. The U-Net integrates both local and global information and acts as a feature inpainting network. The output feature grids denoted as $\mathbf{c}$, are shared for affordance and geometry learning.

2.2. Implicit Neural Representations

We use implicit neural representations to encode both affordance and geometry. As both require reasoning about local geometry details, we condition the deep implicit functions on local features. We query the local feature from the shared feature planes $\mathbf{c}$. Given a query position $\mathbf{p}$, we project it to each feature plane and query the features at the projected locations using bilinear interpolation. The local feature $\psi_{\mathbf{p}}$ is concatenation of the query results of three planes.

Affordance Implicit Functions The affordance implicit functions represent the grasp affordance field of grasp parameters and grasp quality. They map the grasp center $\mathbf{t}$ to grasp parameters ($\mathbf{r}$ and w) and the grasp quality metric q. These implicit neural representations enable learning directly from data with continuous grasp centers.

We implement implicit neural representations as functions that map a pair of point and corresponding local feature $(\mathbf{t}, \psi_{\mathbf{t}})$ to target values. We parametrize the functions with small fully-connected occupancy networks with multiple ResNet blocks [7]. Grasp center, grasp orientation, and gripper width are output from three separate implicit functions.

Geometry Implicit Function Our geometry implicit function maps from an arbitrary query point $\mathbf{p}$ inside the bounded volume to the occupancy probability $o(\mathbf{p})$ at the point. Similar to affordance implicit functions, we learn a function that predicts occupancy based on input point coordinate and the corresponding local feature. Notice that the query points of occupancy $\mathbf{p}$ can be different from the grasp center points $\mathbf{t}$.

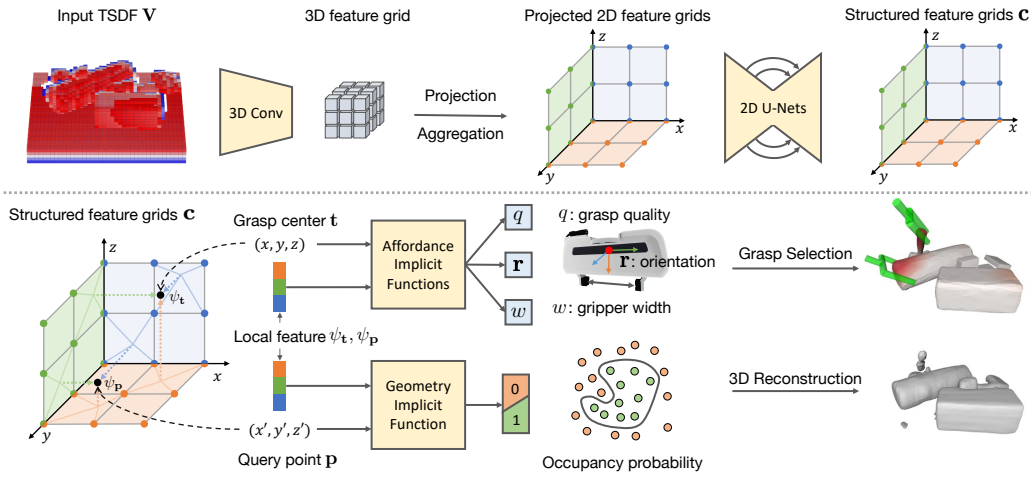

Figure 2: Model architecture of GIGA.

2.3. Grasp Detection

GIGA takes as input a TSDF voxel grid, a grasp center, and multiple occupancy query points and predicts grasp parameters corresponding to the grasp center and occupancy probabilities at the query points.

Given the trained GIGA model, we use a sampling procedure to select the final grasp pose. Grasp affordance is implicitly defined by the learned neural networks, so we need to query it from the learned implicit functions. To cover all possible graspable regions, we discretize the volume of the workspace into voxel grids and use the position of all the voxel cells as grasp centers. Then we query the grasp quality and grasp parameters corresponding to these grasp centers in parallel. Next, we mask out impractical grasps and apply non-maxima suppression as done in VGN [2]. Finally, we select a grasp with the highest quality if the quality is beyond a threshold. If no grasp has the quality above the threshold, we don't make grasp predictions and give up the current scene.

2.4. Training

The loss for training consists of two parts: the affordance loss and the geometry loss. For the affordance loss, we adopt the same training objective as VGN [2]:

$$\mathcal{L}_A(\hat{g}, g) = \mathcal{L}_q(\hat{q}, q) + q(\mathcal{L}_r(\hat{\mathbf{r}}, \mathbf{r}) + \mathcal{L}_w(\hat{w}, w)). \quad (1)$$

Here $\hat{g}$ denotes predicted grasp parameters and g denotes ground-truth parameters. $q \in \{0, 1\}$ is the ground-truth grasp quality (0 for failure, 1 for success) and $\hat{q} \in [0, 1]$ is the predicted grasp quality. $\mathcal{L}_q$ is a binary cross-entropy loss between the predicted and ground-truth grasp quality. $\mathcal{L}_w$ is the ℓ_2-distance between predicted gripper width $\hat{w}$ and ground-truth one w. For orientation, $\mathcal{L}_{quat}$ between predicted quaternion $\hat{\mathbf{r}}$ and target quaternion $\mathbf{r}$ is given by $\mathcal{L}_{quat}(\hat{\mathbf{r}}, \mathbf{r}) = 1 - |\hat{\mathbf{r}} \cdot \mathbf{r}|$. However, the parallel-jaw gripper is symmetric, which means a grasp configuration corresponds to itself after rotated by 180° about the gripper's wrist axis. To handle this symmetry during training, both mirrored rotations $\mathbf{r}$ and $\mathbf{r}_\pi$ are deemed as ground-truth. Thus the orientation loss is defined as:

$$\mathcal{L}_r(\hat{\mathbf{r}}, \mathbf{r}) = \min(\mathcal{L}_{quat}(\hat{\mathbf{r}}, \mathbf{r}), \mathcal{L}_{quat}(\hat{\mathbf{r}}, \mathbf{r}_\pi)). \quad (2)$$

We only supervise the grasp orientation and gripper width when a grasp is successful ($q = 1$).

For the geometry loss, we apply the standard binary cross-entropy loss between the predicted occupancy $\hat{o} \in [0, 1]$ and the ground-truth occupancy label $o \in \{0, 1\}$. The loss is denoted as $\mathcal{L}_G$. The final loss is simply the direct sum of the affordance loss and geometry loss.

Table 1: Quantitative results of clutter removal. We report mean of grasp success rates (GSR) and declutter rates (DR). HR denotes high resolution.

Method	Packed		Pile	
	GSR (%)	DR (%)	GSR (%)	DR (%)
VGN [2]	74.5	79.2	60.7	44.0
GIGA-Aff	77.2	78.9	67.8	49.7
GIGA	83.5	84.3	69.3	49.8
GIGA (HR)	**87.9**	**86.0**	**69.8**	**51.1**

3. Experiments

We study the efficacy of synergies between affordance geometry on grasp detection in clutter.

3.1. Experimental Setup

Our model is trained in a self-supervised manner with ground-truth grasp labels collected from physical trials in simulation and occupancy data obtained from the object

meshes. The use of TSDF enables zero-shot transfer of our model from simulation to a real Panda arm from Franka Emika.

Simulation Environment Our simulated environment is built on PyBullet [4]. We use a free gripper to sample grasps in a $30 \times 30 \times 30\text{cm}^3$ tabletop workspace.

We collect grasp data in a self-supervised fashion in two type of simulated scenes, *pile* and *packed* as in VGN [2]. In the pile scenario, objects are randomly dropped to a box of the same size as the workspace. Removing the box leaves a cluttered pile of objects. In the packed scenario, a subset of taller objects is placed at random locations on the table at their canonical pose. We store grasp parameters and the corresponding outcomes of random grasp trials and balance the dataset by discarding redundant negative samples.

We collect the occupancy training data in the same scenes where grasp trials are performed. Upon the creation of a simulation scene, we query the binary occupancy of a large number of points uniformly distributed in the cubic workspace as the training data.

Camera Observations To evaluate the model's robustness against noise and occlusion in real-world clutter, we assume that the robot perceives the workspace by a single depth image from a fixed side view. To expedite sim-to-real transfer, we add noise to the rendered images in simulation using the additive noise model in [11]. The input to the our algorithm is a $40 \times 40 \times 40$ TSDF [5] fused from the noisy single-view depth image using the Open3D library [22].

Grasp Execution We select top grasps to execute by querying grasp parameters from the learned implicit functions with a set of grasp centers. For a fair comparison with VGN [2], our **GIGA** model samples $40 \times 40 \times 40$ uniformly distributed grasp centers in the workspace and query the grasp parameters. However, our implicit representations are continuous, so we can query grasp samples in arbitrary resolutions. In **GIGA (HR)**, we query at a higher resolution of $60 \times 60 \times 60$.

We use a set of clutter removal scenarios to evaluate GIGA and other baselines. Each round, a pile or packed scene with 5 objects is generated. We take a depth image from the same viewpoint as training. The grasp detection algorithm generates a grasp proposal given the input TSDF. We execute the grasp and remove the grasped object from the workspace. If all objects are cleared, two consecutive failures happen, or no grasp is detected, we terminate the current scene. Otherwise, we collect the new observation and predict the next grasp. In our experiments, grasp proposals with a predicted grasp quality below 0.5 are discarded.

Performance is measured using the following metrics averaged over 100 simulation rounds: 1) Grasp success rate (GSR), the ratio of success grasp executions; and 2) Declutter rate (DR), the average ratio of objects removed.

Table 2: Quantitative results of clutter removal in real world. We report GSR, DR, the number of successful grasps, and the number of total grasp trials (in bracket).

Method	Packed		Pile	
	GSR (%)	DR (%)	GSR (%)	DR (%)
VGN [2]	77.2	81.3	79.0	85.3
GIGA	**83.3**	**86.6**	**86.9**	**97.3**

3.2. Grasp Detection Results

We report grasp success rate and declutter rate for different scenarios in Table 1. GIGA-Aff is an ablated version of our method with only affordance implicit function branch. We can see that GIGA and GIGA-Aff outperform other baselines in almost all scenarios and metrics. Even though GIGA-Aff does not utilize the synergies between affordance and geometry and is trained without geometry supervision, it still outperforms the state-of-the-art VGN baseline. We attribute this to the high expressiveness of our implicit neural representations. Next, we compare the results of GIGA-Aff with GIGA. In the pile scenario, the gain from geometry supervision is relatively small (around 2% grasp success rate). However, in the packed scenario, GIGA outperforms GIGA-Aff by a large margin of around 5%. We believe this is due to the different characteristics of these two scenarios. In the packed scene, some tall objects standing in the workspace would occlude the objects behind them and the occluded objects are partially visible. We hypothesize that in this case, the geometrically-aware feature representation learned via geometry supervision facilitates the model to predict grasps on partially visible objects. Such occlusion is, however, less frequent in pile scenarios. These results demonstrate that synergies between affordance and geometry improve grasp detection, especially in the presence of occlusion.

3.3. Real Robot Experiments

Finally, we test our method in the clutter removal experiments on the real hardware. Table 2 reports the real-world evaluations. GIGA achieves higher success rates and clears more objects.

4. Conclusion

We introduced GIGA, a method for 6-DoF grasp detection in a clutter removal task. Our model learns grasp detection and 3D reconstruction simultaneously using implicit neural representations, exploiting the synergies between affordance and geometry. The experimental results demonstrate that utilizing the synergies between affordance and geometry can improve 6-DoF grasp detection, especially in the case of large occlusion.

References

[1] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesis—a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2013. 1

[2] Michel Breyer, Jen Jen Chung, Lionel Ott, Siegwart Roland, and Nieto Juan. Volumetric grasping network: Real-time 6 dof grasp detection in clutter. In *Conference on Robot Learning*, 2020. 2, 3, 4

[3] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2

[4] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2019. 4

[5] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Annual Conference on Computer Graphics and Interactive Techniques*, pages 303–312, 1996. 4

[6] Carlo Ferrari and John F Canny. Planning optimal grasps. In *ICRA*, volume 3, pages 2290–2295, 1992. 1

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[8] Yun Jiang, Stephen Moseson, and Ashutosh Saxena. Efficient grasping from rgbd images: Learning using a new rectangle representation. In *IEEE International Conference on Robotics and Automation*, pages 3304–3311, 2011. 1

[9] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293*, 2018. 1

[10] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *arXiv preprint arXiv:2007.11571*, 2020. 2

[11] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv preprint arXiv:1703.09312*, 2017. 1, 4

[12] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 2

[13] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421. Springer, 2020. 2

[14] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019. 2

[15] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. *arXiv preprint arXiv:2003.04618*, 2020. 2

[16] Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 3406–3413. IEEE, 2016. 1

[17] Alberto Rodriguez, Matthew T Mason, and Steve Ferry. From caging to grasping. *The International Journal of Robotics Research*, 31(7), 2012. 1

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and Computer-Assisted Intervention*, pages 234–241, 2015. 2

[19] Andreas ten Pas, Marcus Gualtieri, Kate Saenko, and Robert Platt. Grasp pose detection in point clouds. *The International Journal of Robotics Research*, 36(13-14):1455–1473, 2017. 1

[20] Andreas Ten Pas and Robert Platt. Localizing handle-like grasp affordances in 3d point clouds. In *Experimental Robotics*, pages 623–638. Springer, 2016. 1

[21] Andreas ten Pas and Robert Platt. Using geometry to detect grasp poses in 3d point clouds. In *Robotics Research*, pages 307–324. Springer, 2018. 1

[22] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018. 4