2022 IEEE International Conference on Robotics and Automation (ICRA)
May 23-27, 2022. Philadelphia, PA, USA

HGC-Net: Deep Anthropomorphic Hand Grasping in Clutter

Yiming Li[1, 2, *], Wei Wei[1, 2, *], Daheng Li[1, 2], Peng Wang[1, 2, 3, 4, ✉], Wanyi Li[1], Jun Zhong[1]

2022 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9681-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICRA46639.2022.9811756

***Abstract*— Grasping in cluttered environments is one of the most fundamental skills in robotic manipulation. Most of the current works focus on estimating grasp poses for parallel-jaw or suction-cup end effectors. However, the study for dexterous anthropomorphic hand grasping in clutter remains a great challenge. In this paper, we propose HGC-Net, a single-shot network that learns to predict dense hand grasp configurations in clutter from single-view point cloud input. Our end-to-end neural network can predict hand grasp proposals efficiently and effectively. To enhance generalization, we built a large-scale synthetic grasping dataset with 179 household objects, 5K cluttered scenes and over 10M hand annotations. Experiments in simulation show that our model can predict dense and robust hand grasps and clear over 78% of unseen objects in clutter without any post-processing and outperform baseline methods by a large margin. Experiments on the real robot platform also demonstrate that the model trained on synthetic data performs well in natural environments. Code is available at https://github.com/yimingli1998/hgc_net.**

I. Introduction

Grasping is one of the most fundamental manipulation skills for robots to interact with objects and has been widely applied in industry and household service. Recent works mainly focus on grasping objects with parallel-jaw grippers in structured scenes [1], [2]. However, the study of anthropomorphic hand grasping in cluttered scenes remains a challenge.

Previous work on multi-finger hand grasping can be categorized into two groups: model-based methods and data-driven methods. Model-based methods assume that both geometry and pose of the object are known in prior and sample grasp configurations with commonly used grasping metrics [3], [4]. However, the high-dimensional degree-of-freedom of the anthropomorphic hand leads to inefficient sampling due to the enormous searching space. The requirement of complete models also limits these methods to generalize to novel objects.

To handle the time-consuming problem and improve the generalization ability, data-driven methods aim to address the generic problem and have attached more research attention. Some of the works focus on human grasp synthesis achieving promising results on both in-domain and out-of-domain objects [5], [6], [7], [8], while these methods require complete object models as input to model precise hand-objects interactions based on contacts. Human hands are more flexible than robotic hands and enjoy soft-touch characteristics for grasping. It is challenging to transfer their models and datasets to robotic hands directly with the physical properties gap. [9] propose to synthesize functional grasps in simulation with contact affordance of object models, while it relies on completed objects model and reliable contact affordance. [10] generate an anthropomorphic hand grasp dataset and train a generative-evaluative model to grasp a single object, which requires complex handcraft rules to sample grasp proposals. [11] propose a coarse-to-fine generative model to predict hand configurations from single RGB-D images with a time-consuming shape completion module.

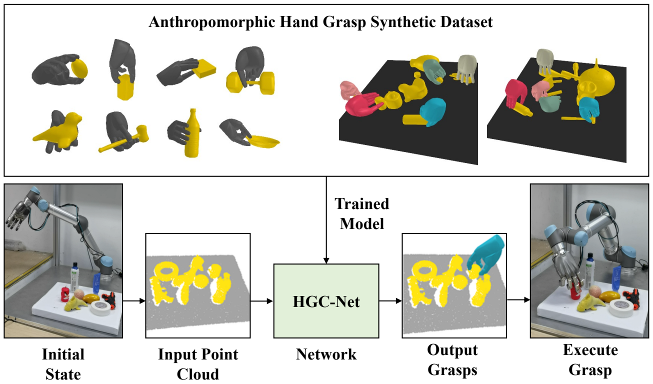

Fig. 1: Overview of the proposed hand grasping method. HGC-Net is trained offline to predict robustness hand configurations from the point cloud using a synthetic dataset with 5K clutters and over 10M grasp annotations. When a scene is presented to the robot, the network takes a point cloud captured by a depth camera as input and predicts dense grasp configurations efficiently. Grasp with the highest quality score is selected for the robot to execute.

Although these approaches achieve significant performance for a single object, it is still an open problem for a robot to grasp generic objects in cluttered scenes with single-view observation with anthropomorphic hands. To this end, we propose HGC-Net, an end-to-end efficient hand grasp generation network that directly predicts grasp proposals from cluttered scenes, illustrated in Fig. 1. Our method involves (1) a single-shot neural network that predicts collision-free hand grasps from a single-view point cloud; and (2) a scene-level training data synthesis pipeline leveraging an innovative anthropomorphic hand grasp model.

Our HGC-Net enjoys effective and efficient grasp performance compared with existing methods [10], [11] in

This work was supported in part by the InnoHK, in part by the National Natural Science Foundation of China under Grants (91748131, 62006229 and 61771471), and in part by the Strategic Priority Research Program of Chinese Academy of Science under Grant XDB32050100.

[1] Institute of Automation, Chinese Academy of Sciences, Beijing, China.
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3] CAS Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China
[4] Centre for Artificial Intelligence and Robotics, Hong Kong Institute of Science and Innovation, Chinese Academy of Sciences, Hong Kong, China.
[*] Authors contributed equally.
✉ Corresponding author: peng_wang@ia.ac.cn

978-1-7281-9680-0/22/$31.00 ©2022 IEEE

anthropomorphic hand grasping literature. Previous methods, like [11], formulate the hand grasping problem as a multi-stage task with shape reconstruction, grasp generation, and refinement. The long-term pipeline makes the whole procedure time-consuming. Besides, grasp proposals generated from a single object may not be applicable in scenes due to the potential collision between the hand and around objects. Unlike these methods, we propose to directly regress the 26-DOF hand configurations from the raw input in one pass. To train our model, we build a large-scale grasping dataset in the simulator [12] automatically. It consists of 5K scenes in structured clutter with 179 household objects. Point clouds for each scene are sparsely annotated in terms of: graspable or not, corresponding grasp types and configurations.

We evaluate our approach on MuJoCo simulator [12] and real-world robot platform with UR5 arm and HIT-DLR II anthropomorphic hand. Experiments show that the model trained on synthetic data performs well on real-world, with diverse, robust grasp proposals and generalizes well on unseen objects. Our method shows significant improvement in both time efficiency and success rate compared with the baseline method.

In summary, our primary contributions are:

- A large-scale synthetic scene dataset with 179 objects, 5K cluttered scenes, and over 10M five-finger hand grasp annotations.
- An end-to-end hand grasp proposal network that predicts robust grasp configurations from single-view point cloud effectively and efficiently.
- Significant improvement in terms of grasping success rate and time cost in cluttered scenes compared with baseline method.

II. Related work

Model-based vs. data-driven grasp approaches. Traditional model-based grasp approaches assume that both object model and environment are accessible [13], [14]. These methods sample a large number of grasp proposals at first and measure the grasp quality based on certain metrics [4], [3]. However, the search policy in high-dimensional space is time-consuming, and it is difficult to apply to real-world scenarios because of the lack of complete object models and unstructured experiments. In recent years, data-driven grasp methods have attached more attention. Some approaches propose to detect 2D grasp rectangle to achieve top-down grasping [15], [16], [17], while a lot of recent works [18], [1], [2], [19], [20] directly predict 6-DoF grasp poses from a single-view input. [11], [21] present a coarse-to-fine grasp method to generate collision-free grasps through a multi-finger hand.

Parallel-jaw grasping vs. hand grasping. Most literature focuses on estimating grasp poses of the parallel-jaw gripper, either take 2D image [15], [16] or point cloud [18], [2], [22] as input. Although the simplified end effector is suited for pick-and-place tasks, it is difficult to achieve dexterous grasping since most household objects are designed for human hands [9]. Recently, hand-object interactions have been attached much more attention [6], [23], [5], [24], [7]. Some works provide human hand datasets that label grasp postures from images or videos [23], [24], or propose to jointly optimize hand-object contacts and interpenetration by predicting hand-object affordance map [6], [25], [5].

Anthropomorphic robotic hand vs. human hand. Many related works on hand grasping focus on human grasp synthesis [6], [24], [7], [5], [25] based on the MANO hand model [26]. Although these approaches can generate realistic hand grasps, the large gap between the human hand and robotic hand makes it difficult to transfer to robotic grasping. [9] propose to generate functional grasps based on the contact map on the object surface and is successfully applied to different multi-finger hands. [11], [10], [21] propose to plan dexterous grasps for the specific anthropomorphic robotic hand and achieve good performance on real-world robotic grasping.

Grasping in clutter vs. isolated objects. Among learning-based methods for grasping, most studies focus on dealing with isolated objects placed on planar surface [17], [27], especially for high dimensional grippers [11], [28], [10]. However, grasps planned on single objects may not be accessible in scenes due to the potential collision between the hand and the environment. Grasping in clutters with multiple objects is significantly harder because of the limited space to access the object and occlusions [1], [2]. Some recent works propose to predict collision-free 6-DoF grasping in clutter with parallel-jaw grippers [2], [19], [22], [29]. However, it remains an open problem to grasp generic objects in clutters with the anthropomorphic hand. To our knowledge, [21] is the most similar work to ours, which proposes to grasp scene objects through a multi-stage method, while our model jointly optimizes grasp poses and hand joints in one pass to generate collision-free grasps.

III. Problem statement

In this work, we focus on the problem of predicting robust grasp poses from the single-view point cloud in cluttered scenes. HGC-Net tasks single-view point cloud and outputs point-wise grasp poses with high quality and different types. Several key definitions are presented here:

Object states: Let $\mathcal{O}_i$ and $\mathcal{T}_i$ denote object $\mathcal{O}_i$ with 6D pose $\mathcal{T}_i$ in a specific scene.

Point clouds: Let $\mathcal{P}_k \in \mathbb{R}^{N\times 3}$ denotes the point cloud of N points in the k^{th} scene captured by depth camera.

Hand grasps: Let $\mathcal{H} = \{\mathcal{H}_1, \mathcal{H}_2, \cdots, \mathcal{H}_m\}$ denotes the hand configurations for m objects in a cluttered scene, where each hand configuration $h = (\boldsymbol{t}, \boldsymbol{q}, \boldsymbol{\theta}, \boldsymbol{c}) \in \mathcal{H}_{1,2,\cdots,m}$. Specifically, $\boldsymbol{t} = (t_x, t_y, t_z)$ represents the translation while $\boldsymbol{q} = (q_w, q_x, q_y, q_z)$ is the orientation quaternion of the hand palm. $\boldsymbol{\theta} \in \mathbb{R}^{20}$ denotes 20-DoF hand joints of our HIT-DLR II hand. $\boldsymbol{c}$ is the grasp type defined by $[\boldsymbol{\theta}_{\boldsymbol{c}}^{init}, \boldsymbol{\theta}_{\boldsymbol{c}}^{end}]$, which respectively represents the initial and final hand joint configurations.

IV. Dataset generation

In this section, we introduce the pipeline for our hand grasp dataset generation. Illustrated in Fig. 2, we label hand

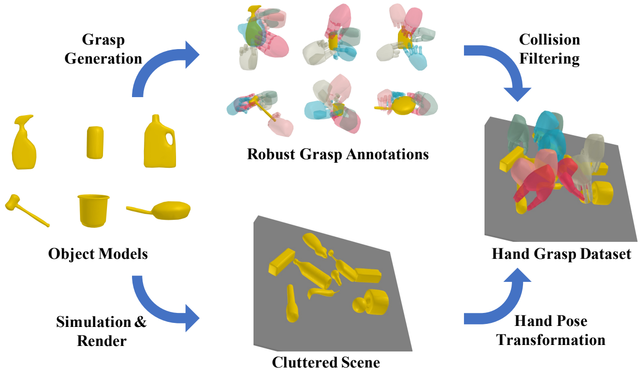

Fig. 2: Pipeline of our synthetic dataset generation procedure. We collect single object at first and generate robust hand grasps for each object in simulation. Objects are randomly selected and placed on a table to construct clutter scenes. Grasps without collision are aligned to the scene.

grasp annotations for a single object at first and then match them to scenes according to the known 6D object pose. Finally, we adopt a collision detection module to filter invalid hand grasps.

A. Single Object Grasp Generation

We collect 179 common objects in total with various shapes and categories from existing datasets and the internet. We apply the approach-based grasp sampler method[30], which assumes that the hand approaches the object along a line that defined by the point normal. We sample 512 points for each object. For each point $\boldsymbol{p}_k \in \mathbb{R}^3$ with normal $\boldsymbol{n}_k \in \mathbb{R}^3$, we sample uniform depths $\boldsymbol{d} \in \mathcal{D}$, in-plane rotation angles $\boldsymbol{a} \in \mathcal{A}$ and grasp type $\boldsymbol{c} \in \mathcal{C}$ to generate dense grasp candidates, where $\mathcal{D}, \mathcal{A}, \mathcal{C}$ are predefined depth, angle and type sets. So that the ith hand configuration $h_k^i \in \mathcal{H}$ for point $\boldsymbol{p}_k$ can be represented as:

$$\begin{aligned} h_k^i &= (\boldsymbol{p}_k - N(\boldsymbol{n}_i) \cdot \boldsymbol{d}_i, \boldsymbol{R}_i, \boldsymbol{\theta}_i, \boldsymbol{c}_i) \\ \boldsymbol{R}_i &= [N(\boldsymbol{n}_i), N(\boldsymbol{r}_i), N(\boldsymbol{n}_i \times \boldsymbol{r}_i)] \end{aligned} \tag{1}$$

where $\boldsymbol{R}_i \in \mathbb{R}^{3\times3}$ denotes a rotation matrix equal to $\boldsymbol{q}_i$ and $\boldsymbol{r}_i \in \mathbb{R}^3$ is the hand axis defined by $\boldsymbol{n}_i$ and $\boldsymbol{a}_i$. $\boldsymbol{N}(*)$ represents the normalization function.

We evaluate grasp quality on the MoJuCo [12] physics simulator. During the simulation, hand joints are closed from $\boldsymbol{\theta}_{\boldsymbol{c}_i}^{init}$ to $\boldsymbol{\theta}_{\boldsymbol{c}_i}^{end}$ until contact with the object and recorded as $\boldsymbol{\theta}_i$. Then fingers keep a minimum grasp force to resist the gravity on the object to prevent the object from falling. A slight shaking is also added to filter unstable grasps, and only successful grasps which keep objects in hand are reserved. The generated grasps for each grasp type are illustrated in Fig. 3. For each object, we simulate over 100 thousand grasp experiments to generate hand grasp configuration labels.

B. Scene Grasp Dataset Generation

The schedule of generating scene grasp dataset consists of three substeps. At First, we adopt BlenderProc [31] to simulate structured cluttered scenes. For each scene, m objects sampled with random poses and then made to freely fall on a table. RGB-D images are captured by a simulated camera from random views when the objects reach stable states.

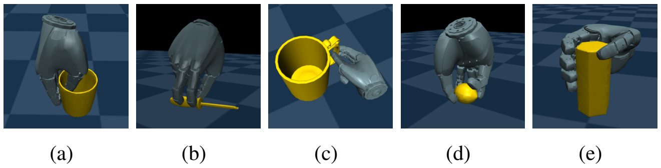
(a) (b) (c) (d) (e)

Fig. 3: Five types of HIT-DLR II hand we selected for the single object grasp generation. (a) Parallel extension. (b) Pen pinch. (c) Palmar pinch. (d) Precision sphere. (e) Medium wrap.

After that, we record object poses$\{\mathcal{T}_1, \mathcal{T}_2, \cdots, \mathcal{T}_m\}$ for each scene and transform corresponding hand grasps $\{\mathcal{H}_1, \mathcal{H}_2, \cdots, \mathcal{H}_m\}$ to object coordinates. A collision checker is applied to filter collided grasps between each grasp $h_k^i \in \mathcal{H}_{1,2,\cdots,m}$ and the scene. For each grasp type, sampled points $p_{1,2,\cdots,k}^m$ of object m with at least one collision-free grasps are annotated as positive grasp points with hand configurations, and others are annotated with negative labels.

The last step is to match the point cloud captured by the camera with sampled grasp point candidates. We apply a KD-Tree algorithm for each sampled point to search the nearby points among the point cloud with query radius $r = 0.005$m. Scene points in a group will be broadcast the same label as the sampled point.

V. Method

In this section, we present our HGC-Net for dexterous grasp pose detection in clutter. Given the partially observed point clouds, HGC-Net predicts point-wise mask and corresponding grasp configurations for each predefined grasp type. The overall pipeline is shown in Fig. 4.

A. Feature Representation

Similar to several previous works which detect parallel-jaw grasps directly in point clouds, we address the hand grasp detection problem in a learning-based framework. To avoid exhaustive searching in $\mathbb{SE}(3)$, we propose to directly predict the 6-DoF grasp pose in a bottom-up manner through a single-shot neural network.

The proposed hand grasp network is developed based on PointNet++ [32], a robust model to encode 3D geometry features for point clouds. To extract point-wise features, we utilize PointNet++ with a multi-scale grouping strategy as our backbone network.

B. Grasp Points Segmentation

Given the encoded features, we first utilize a grasp points segmentation head to classify graspable points for each predefined grasp type. To decrease the interference caused by the table, we randomly annotate table points as negative labels with a sample rate of γ =0.05. We use a weighted cross-entropy loss to handle the unbalanced distribution of positive and negative point samples:

$$\mathcal{L}_{seg}^{p,c} = \mathcal{F}_{cls}(y^{p,c}, \widehat{y^{p,c}}), \tag{2}$$

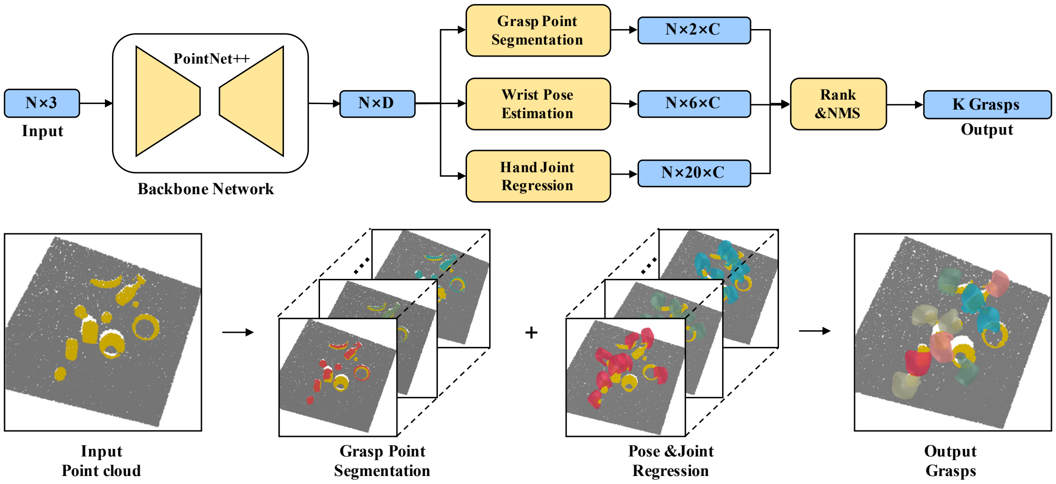

Fig. 4: Overview of the proposed method. Given a $N \times 3$ point cloud, our model first extracts hierarchical point features $N \times D$ based on PointNet++ multi-scale grouping module where D is the dimension of feature vector at each point. For each predefined grasp type, the network predicts point-wise graspable masks, wrist poses, and hand joints. Finally, the generated grasps are ranked by grasp mask score and filtered by non-maximum suppression. C, K is the number of predefined grasp types and output grasps, respectively.

where $y^{p,c}, \widehat{y^{p,c}}$ respectively denote graspable label and predicted mask for each labeled point $p \in \mathcal{P}_{pos} \cup \mathcal{P}_{neg}$ with type c. $\mathcal{P}_{pos}$ and $\mathcal{P}_{neg}$ are positive and negative point sets. The weights, w_1 and w_2 for $p \in \{\mathcal{P}_{pos}, \mathcal{P}_{neg}\}$ are set to 1.0 and 10.0.

C. Wrist Pose Estimation

It is difficult to directly regress high-dimensional hand pose in $\mathbb{SE}(3)$ [33], [34]. We observe that since the point mask and position are both known, the 6-DoF hand pose $(\boldsymbol{t}, \boldsymbol{q})$ of a graspable point $p \in \mathcal{P}_{pos}$ with type c can be simplified by approach depth $\boldsymbol{d}^{p,c}$, approach direction $\boldsymbol{n}^{p,c}$, and hand grasp axis $\boldsymbol{r}^{p,c}$ (Eq. 1).

To predict the 6-DoF wrist pose, we utilize a bin-based regress method for pose generation inspired by [33], [22]. We divide $\boldsymbol{n}$ and $\boldsymbol{r}$ into three angles, shown in Fig. 5(a). Azimuth angle $\boldsymbol{\delta}_1 \in [0, 2\pi]$ and elevation angle $\boldsymbol{\delta}_2 \in [0, \pi/2]$ jointly define the approach direction $\boldsymbol{n}$. Rotation angle $\boldsymbol{\delta}_3 \in [-\pi, \pi]$ denotes the projected axis $\boldsymbol{r}$ onto X-Y plane.

The target angle $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \boldsymbol{\delta}_3$ are divided into n_1, n_2, n_3 bins with uniform angle $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3 = \frac{2\pi}{n_1}, \frac{\pi/2}{n_2}, \frac{2\pi}{n_3}$. For each point p, we calculate bin classification label and residual label as follows:

$$\begin{aligned} \underset{i=1,2,3}{\operatorname{bin}^{p,c}_{\boldsymbol{\delta}_i}} &= \lfloor \frac{\boldsymbol{\delta}_i^{p,c} - \boldsymbol{\delta}_i^{init}}{\boldsymbol{\phi}_i} \rfloor \\ \underset{i=1,2,3}{\operatorname{res}^{p,c}_{\boldsymbol{\delta}_i}} &= \frac{1}{\boldsymbol{\phi}_i}(\boldsymbol{\delta}_i^{p,c} - \boldsymbol{\delta}_i^{init} - (\operatorname{bin}^{p,c}_{\boldsymbol{\delta}_i} \cdot \boldsymbol{\phi}_i + \frac{\boldsymbol{\phi}_i}{2})), \end{aligned} \tag{3}$$

where $\operatorname{bin}_{\boldsymbol{\delta}_i}$, $\operatorname{res}_{\boldsymbol{\delta}_i}$ are classification and regression labels, and $\boldsymbol{\delta}_i^{init}$ is the start angle for $\boldsymbol{\delta}_i (i = 1, 2, 3)$(Fig. 5(b)). Similarity, the grasp depth can be defined by bin_d and res_d, and the grasp pose loss is formulated as:

$$\mathcal{L}^{p,c}_{pose} = \sum_{w \in \{\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \boldsymbol{\delta}_3, \boldsymbol{d}\}} \mathcal{F}_{cls}(\operatorname{bin}^{p,c}_w, \widehat{\operatorname{bin}^{p,c}_w}) + \mathcal{F}_{reg}(\operatorname{res}^{p,c}_w, \widehat{\operatorname{res}^{p,c}_w}), \tag{4}$$

where $\operatorname{bin}^{p,c}_w$ and $\operatorname{res}^{p,c}_w$ are ground-truth bin assignment and residual of point p with type c, $\widehat{\operatorname{bin}^{p,c}_w}$ and $\widehat{\operatorname{res}^{p,c}_w}$ are predicted corresponding values. $\mathcal{F}_{cls}$ and $\mathcal{F}_{reg}$ represent cross entropy loss and smooth L1 loss, respectively.

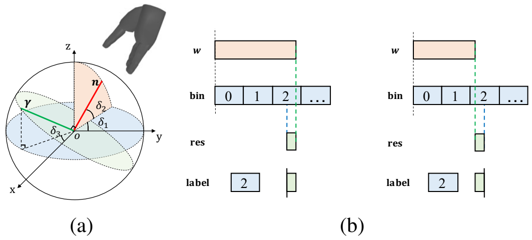

Fig. 5: illustrations of hand rotation representation and bin-based loss. (a) Approach vector $\boldsymbol{n}$ is defined by azimuth angle $\boldsymbol{\delta}_1$ and elevation angle $\boldsymbol{\delta}_2$. Hand grasp axis $\boldsymbol{r}$ is located on the plane orthogonal to the approach vector and its projection on X-Y plane is $\boldsymbol{\delta}_3$. (b) Two examples show the wrist pose label $w \in \{\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \boldsymbol{\delta}_3, \boldsymbol{d}\}$ is divide into the classification label bin_w and regression label res_w. For more details about the loss function, we refer readers to [33].

D. Hand Joint Regression

To improve the grasp quality and generate realistic hand configurations, we adopt a hand joint prediction layer to predict precise hand joint $\boldsymbol{\theta}$. For each graspable point $p \in \mathcal{P}_{pos}$ and type $\boldsymbol{c}$, we first normalize the joint label $\boldsymbol{\theta}^{p,c}$ to $[0, 1]$, and supervise the predicted hand joint by mean square error (MSE) loss:

$$\begin{aligned} N(\boldsymbol{\theta}^{p,c}) &= \frac{\boldsymbol{\theta}^{p,c} - \boldsymbol{\theta}^{init}_c}{\boldsymbol{\theta}^{end}_c - \boldsymbol{\theta}^{init}_c}, \\ \mathcal{L}^{p,c}_{joint} &= ||N(\boldsymbol{\theta}^{p,c}) - \hat{N}(\boldsymbol{\theta}^{p,c})||^2_2, \end{aligned} \tag{5}$$

where $N(\boldsymbol{\theta}^{p,c})$ denotes the normalized hand joint label, while $\hat{\boldsymbol{N}}(\boldsymbol{\theta}^{p,c})$ is the corresponding prediction. $\boldsymbol{\theta}^{init}_c$ and $\boldsymbol{\theta}^{end}_c$ denote the initial and final hand joint of type $\boldsymbol{c}$.

E. Total Loss

The overall loss of HGC-Net could be formulated as follows:

$$\mathcal{L}_{total} = \sum_{p \in \mathcal{P}_{pos} \cup \mathcal{P}_{neg}, c} \mathcal{L}^{p,c}_{seg} + \sum_{p \in \mathcal{P}_{pos}, c} (\mathcal{L}^{p,c}_{pose} + \mathcal{L}^{p,c}_{joint}). \tag{6}$$

The total loss includes two terms, graspable points segmentation and grasp configurations prediction. During inference, each input point predicts a graspable mask, and grasp configurations belonging to positive points are regarded as predicted grasps.

To measure the quality of generated grasps, we rank the output grasps by grasp points segmentation score with the softmax function:

$$s_{p,c} = \frac{e^{\boldsymbol{y}_{p,c}^{pos}}}{e^{\boldsymbol{y}_{p,c}^{pos}} + e^{\boldsymbol{y}_{p,c}^{neg}}}, \tag{7}$$

where $\boldsymbol{y}_{p,c}^{pos}$ and $\boldsymbol{y}_{p,c}^{neg}$ are the probability of point p as a positive or negative grasp point with type $\boldsymbol{c}$. A pose-NMS algorithm [19] is also applied to select local maximum within $0.03m$ and 30°.

VI. Experiments

In this section, we introduce the experimental setup at first, including the dataset, evaluation metrics, and implementation details. Then we conduct experiments both in simulation and on the real robot platform. Experiments demonstrate that our proposed method can generate dense and robust hand grasp proposals and achieve a high success rate and completion rate compared with the baseline method.

A. Experimental Setup

Dataset. Our synthetic hand grasp dataset contains 179 objects for training with over 10M hand grasp annotations. During training, we generate 5K cluttered scenes, and each scene contains 10 objects randomly placed on a table with 4 images captured by a virtual camera from different views.

Evaluate metrics. To evaluate our proposed method, we introduce four quantitative metrics regarding previous works both in robotic grasping [18], [2] and dexterous hand manipulation [11], [21]:

- **Interpenetration** to measure the penetration between hand mesh and object.
- **Grasp success rate (SR)** to measure the quality of generated hand grasps by robotic experiments.
- **Grasp completion rate (CR)** to measure that how many objects are successfully grasped in a scene.
- **Time cost** to measure the grasp efficiency.

Implementation details. For each scene point cloud, we sample 20,000 points as input. The network is trained for 80 epochs with Adam optimizer. The learning rate is set to 0.01 and decreased by a factor 2 for every 10 epochs with the batch size of 32.

B. Simulation Experiments

We first conduct experiments based on the MuJoCo simulator [12]. The grasping pipeline is described as follows: At first, we create 100 cluttered scenes with 10 novel objects randomly placed on a table. For each scene, a depth camera takes 4 photos from a random viewpoint above the table. The network takes the single-view point cloud as input and outputs point-wise hand configurations for each grasp type. For each object, we select two grasps with the highest scores to execute. The interpenetration between object and scene is calculated through [35]. The maximum simulation step is set to 1000. We classify successful grasp attempts by judging whether the anthropomorphic hand can lift the target object over $20cm$.

Wrist pose and hand joint prediction module. We conduct ablation studies to investigate the effectiveness of the proposed wrist pose estimation and hand joint regression module. Specifically, we compare the bin-based loss L_{bin} with directly regressing quaternions in $\mathbb{SE}(3)$ by calculating the relative angle:

$$\begin{aligned} A_{p,c}(q_{p,c}, \hat{q}_{p,c}) &= \arccos(0.5 \times (\mathbf{Tr}\ (R_{p,c} \cdot \hat{R}_{p,c}^{T}) - 1) \\ \mathcal{L}_{quat}^{p,c} &= A_{p,c}(q_{p,c}, \hat{q}_{p,c}), \end{aligned} \tag{8}$$

where $A_{p,c}(q_{p,c}, \hat{q}_{p,c})$ is the relative angle between predicted $\hat{q}_{p,c}$ and ground truth $q_{p,c}$ of point p with type c, and $\hat{R}_{p,c}$ and $R_{p,c}$ are corresponding rotation matrices. $\mathcal{L}_{quat}$ represents the quaternion loss. To evaluate the influence of the joint angles, we remove the hand joint regression module $\mathcal{M}_{joint}$ during inference, and utilize initial and final joints $[\boldsymbol{\theta}_c^{init}, \boldsymbol{\theta}_c^{end}]$ as an alternative: During grasping, the hand closes from $\boldsymbol{\theta}_c^{init}$ to $\boldsymbol{\theta}_c^{end}$ until capturing the object.

TABLE I: Ablation Studies in Simulation.

L_{bin}	L_{quat}	M_{joint}	Interpenetration (cm^3)↓	SR (%)↑	CR (%)↑
	✓		**2.72**	58.4	66.7
✓			4.14	65.6	74.1
	✓	✓	5.33	66.2	75.3
✓		✓	7.83	**71.9**	**78.8**

Experimental results are shown in Tab. I. The proposed bin-based loss boosts grasping performance in terms of success rate and completion rate compared with quaternion loss. Since bin-based loss and hand joint regression module help to predict more precise hand wrist pose and joint angles for tight hand-object contacts, they inevitably cause additional interpenetration between hand model and objects.

Grasp quality. To demonstrate the quality and efficiency of predicted grasps, we evaluate the grasping performance in terms of SR, CR and compare it with GraspIt! [13]. The experimental settings of our approach are the same as mentioned above. For GraspIt!, we first sample 360 hand grasp proposals based on the simulated annealing planner within 75K steps, and for each object, we choose 2 grasps with the highest quality according to the ϵ-metric [3]. Note that the GraspIt! needs the complete object model to generate grasp configurations while our approach only takes single-view point cloud as input.

TABLE II: Grasping Results in Simulation.

	Input	SR	CR
GraspIt! [13]	Complete scene model	58.5%	54.3%
HGC-Net	Partial point cloud	**71.9%**	**78.8%**

As is illustrated in Tab. II, our model outperforms the baseline method by a large margin in terms of both success rate and completion rate with 13.4% and 24.5% improvement

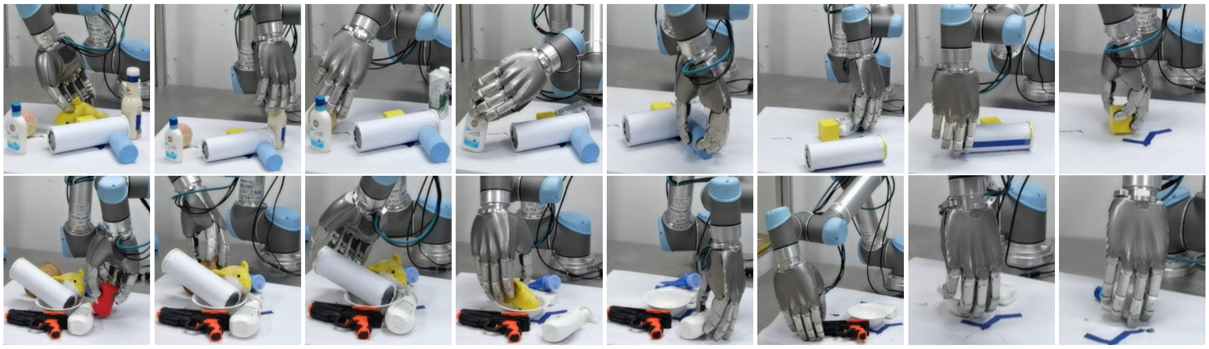

Fig. 6: Qualitative experimental results for robotic grasping. Top: Grasping in structured clutter. Bottom: Grasping in dense clutter.

respectively. Furthermore, we observe that because of the severe stacking in clutter, GraspIt! is not able to plan reasonable grasps for the target object within 75K steps, which causes the lower completion rate.

TABLE III: Time Efficiency.

Method	GraspIt! [13]	DDGC [21]	HGC-Net
Time Cost	40s	9.4s	**0.25**s

Time efficiency. We compare the time efficiency with GraspIt! [13] and DDGC [21], as shown in Tab. III. Due to the large searching space, GraspIt! runs over 100 times slower compared with our method. DDGC is a multi-stage method for a three-finger hand grasping in structured clutters with a serial of modules: scene completion, image encoding, grasp generation and finger refinement, while the fine-grained procedures especially for shape completion lead to high time consumption. Our method shows significant improvement in terms of time efficiency by the single-shot network design.

C. Robot Experiments

In order to validate the performance of our model in the real world, we set up real-world robot experiments on a UR5 robot arm with a HIT-DLR II hand. An Ensenso N35 camera is mounted on the top of the robot and captures scenes from backward at a 60-degree viewpoint (shown in Fig. 7). We prepare 30 objects absent in the training dataset with various shapes and sizes. Points within a $45cm \times 45cm$ square are cropped out for input, and the grasp with the highest score is selected for executing. We utilize ROS to control the robot arm and adopt MoveIt! [36] for motion planning.

To demonstrate the effectiveness and efficiency of the proposed HGC-Net, we conduct experiments on both structured clutter and dense clutter [1]. For structured clutter, we randomly place 4,8,12 objects to build structured cluttered scenes with varying difficulty levels. For dense clutter, we randomly sample 12 objects and pile them together (Fig. 7). The robot attempts multiple grasps until all objects are grasped or 6, 12, 18, 18 grasps have been attempted. Each experiment is conducted for 5 times.

As shown in Tab IV, our method achieves 66.7% success rate and 78.1% completion rate on average in structured clutter, and 55.4% success rate and 68.3% completion rate in dense clutter. Qualitative results shown in Fig. 6 demonstrate that our model can generate robust collision-free hand grasps and generalize well on novel objects. Furthermore, we observe that our model tend to grasp objects from near-vertical direction to avoid collisions with surrounding objects.

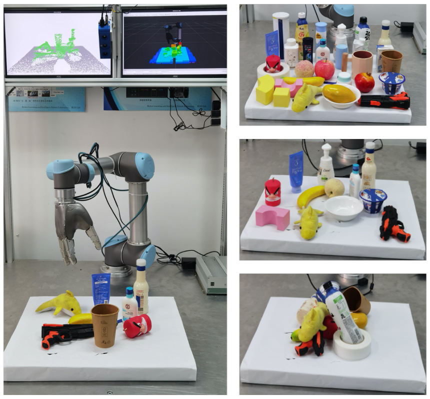

Fig. 7: Setting of real-world robot experiments. Left: The whole work space of our UR5 with HIT-DLR II robotic arm-hand system. Top right: 30 objects absent in the synthetic dataset for grasping. Middle right: Structured clutter with 12 objects placed. Bottom right: Dense clutter scene with 12 objects placed.

TABLE IV: Robot Experiments in Cluttered Scenes.

	Structure clutter			Dense clutter
objects	4	8	12	12
SR	73.9%	67.4%	58.9%	55.4%
CR	85.0%	77.5%	71.7%	68.3%

VII. Conclusion

In this work, we propose HGC-Net, an end-to-end hand grasp proposal network for generating robust hand grasps efficiently. Given a single-view point cloud, our model can directly output point-wise collision-free grasp configurations in cluttered scenes without post-processing. To train the network, we built a large-scale synthetic dataset with 5K cluttered scenes and over 10M grasp annotations. Experiments show that the model trained on synthetic dataset performs well both in simulation and real-world scenarios and outperforms the baseline method by a large margin. In future work, we will (1) generalize our method to universal hand grasping with the different anthropomorphic hands; (2) improve the precision of hand joint to achieve more dexterous hand grasping.

References

[1] A. Murali, A. Mousavian, C. Eppner, C. Paxton, and D. Fox, "6-dof grasping for target-driven object manipulation in clutter," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6232–6238.

[2] Y. Qin, R. Chen, H. Zhu, M. Song, J. Xu, and H. Su, "S4g: Amodal single-view single-shot se (3) grasp detection in cluttered scenes." PMLR, 2020, pp. 53–65.

[3] C. Ferrari and J. F. Canny, "Planning optimal grasps," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1992.

[4] V. Nguyen, "Constructing force-closure grasps," *The International Journal of Robotics Research (IJRR)*, 1988.

[5] H. Jiang, S. Liu, J. Wang, and X. Wang, "Hand-object contact consistency reasoning for human grasps generation," in *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[6] E. Corona, A. Pumarola, G. Alenya, F. Moreno-Noguer, and G. Rogez, "Ganhand: Predicting human grasp affordances in multi-object scenes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR*, 2020.

[7] S. Brahmbhatt, C. Tang, C. D. Twigg, C. C. Kemp, and J. Hays, "Contactpose: A dataset of grasps with object contact and hand pose," in *European Conference on Computer Vision (ECCV)*, 2020.

[8] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *CVPR*, 2019.

[9] S. Brahmbhatt, A. Handa, J. Hays, and D. Fox, "Contactgrasp: Functional multi-finger grasp synthesis from contact," in *IEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[10] U. R. Aktas, C. Zhao, M. Kopicki, A. Leonardis, and J. L. Wyatt, "Deep dexterous grasping of novel objects from a single view," *arXiv preprint arXiv:1908.04293*, 2019.

[11] J. Lundell, E. Corona, T. N. Le, F. Verdoja, P. Weinzaepfel, G. Rogez, F. Moreno-Noguer, and V. Kyrki, "Multi-fingan: Generative coarse-to-fine sampling of multi-finger grasps," in *IEEE International conference on robotics and automation (ICRA)*, 2021.

[12] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[13] A. T. Miller and P. K. Allen, "Graspit! A versatile simulator for robotic grasping," *IEEE Robotics & Automation Magazine*, 2004.

[14] M. Ciocarlie, C. Goldfeder, and P. Allen, "Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem," in *Robotics: Science and systems manipulation workshop-sensing and adapting to the real world (RSS)*, 2007.

[15] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1316–1322.

[16] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," in *Robotics: Science and Systems (RSS)*, 2017.

[17] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[18] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp pose detection in point clouds," *The International Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1455–1473, 2017.

[19] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: a large-scale benchmark for general object grasping," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 444–11 453.

[20] H. Duan, P. Wang, Y. Huang, G. Xu, W. Wei, and X. Shen, "Robotics dexterous grasping: The methods based on point cloud and deep learning," *Frontiers in Neurorobotics*, 2021.

[21] J. Lundell, F. Verdoja, and V. Kyrki, "Ddgc: Generative deep dexterous grasping in clutter," *arXiv preprint arXiv:2103.04783*, 2021.

[22] W. Wei, Y. Luo, F. Li, G. Xu, J. Zhong, W. Li, and P. Wang, "Gpr: Grasp pose refinement network for cluttered scenes," *arXiv preprint arXiv:2105.08502*, 2021.

[23] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "Grab: A dataset of whole-body human grasping of objects," in *European Conference on Computer Vision (ECCV)*, 2020.

[24] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield *et al.*, "Dexycb: A benchmark for capturing hand grasping of objects," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[25] P. Grady, C. Tang, C. D. Twigg, M. Vo, S. Brahmbhatt, and C. C. Kemp, "Contactopt: Optimizing contact to improve grasps," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[26] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 2017.

[27] A. Mousavian, C. Eppner, and D. Fox, "6-dof graspnet: Variational grasp generation for object manipulation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2901–2910.

[28] M. Liu, Z. Pan, K. Xu, K. Ganguly, and D. Manocha, "Deep differentiable grasp planner for high-dof grippers," in *Robotics: Science and Systems (RSS)*, 2020.

[29] Y. Li, T. Kong, R. Chu, Y. Li, P. Wang, and L. Li, "Simultaneous semantic and collision learning for 6-dof grasp pose estimation," *arXiv preprint arXiv:2108.02425*, 2021.

[30] C. Eppner, A. Mousavian, and D. Fox, "A billion ways to grasp: An evaluation of grasp sampling schemes on a dense, physics-based grasp data set," *arXiv preprint arXiv:1912.05604*, 2019.

[31] M. Denninger, M. Sundermeyer, D. Winkelbauer, Y. Zidan, D. Olefir, M. Elbadrawy, A. Lodhi, and H. Katam, "Blenderproc," *arXiv preprint arXiv:1911.01911*, 2019.

[32] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *arXiv preprint arXiv:1706.02413*, 2017.

[33] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[34] C. R. Qi, O. Litany, K. He, and L. J. Guibas, "Deep hough voting for 3d object detection in point clouds," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[35] D.-H. et al., "trimesh," https://trimsh.org/, 2019.

[36] D. Coleman, I. Sucan, S. Chitta, and N. Correll, "Reducing the barrier to entry of complex robotic software: a moveit! case study," *Journal of Software Engineering in Robotics, Special issue on Best Practice in Robot Software Development*, 2014.