2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

ASGrasp: Generalizable Transparent Object Reconstruction and 6-DoF Grasp Detection from RGB-D Active Stereo Camera

Jun Shi[1], Yong A[1], Yixiang Jin[1], Dingzhe Li[1], Haoyu Niu[1], Zhezhu Jin[1], He Wang[2,3,4†]

***Abstract*— In this paper, we tackle the problem of grasping transparent and specular objects. This issue holds importance, yet it remains unsolved within the field of robotics due to failure of recover their accurate geometry by depth cameras. For the first time, we propose ASGrasp, a 6-DoF grasp detection network that uses an RGB-D active stereo camera. ASGrasp utilizes a two-layer learning-based stereo network for the purpose of transparent object reconstruction, enabling material-agnostic object grasping in cluttered environments. In contrast to existing RGB-D based grasp detection methods, which heavily depend on depth restoration networks and the quality of depth maps generated by depth cameras, our system distinguishes itself by its ability to directly utilize raw IR and RGB images for transparent object geometry reconstruction. We create an extensive synthetic dataset through domain randomization, which is based on GraspNet-1Billion. Our experiments demonstrate that ASGrasp can achieve over 90% success rate for generalizable transparent object grasping in both simulation and the real via seamless sim-to-real transfer. Our method significantly outperforms SOTA networks and even surpasses the performance upper bound set by perfect visible point cloud inputs. Project page: https://pku-epic.github.io/ASGrasp**

I. INTRODUCTION

Recent years have witnessed enormous progress [1], [2], [3], [4], [5], [6] in the field of learning-based diffuse object grasping from depth observations. However, commercial depth sensors fail to accurately sense transparent and specular objects, therefore grasping these kind of objects has become a major bottleneck in developing full 3D sensor solution to general grasping tasks.

Several works [7], [8], [9] have tackle this challenging problem by learning to estimate or restore the depth of the transparent and specular objects. ClearGrasp[9] directly removes the transparent area from the raw depth map and then uses an optimization technique to restore the depth. We note that this method fails to fully utilize the original depth observation and relies on precise segmentation of the transparent objects. Later on, DREDS[7] and TransCG[8] propose to learn a mapping from the raw depth along with its RGB observation to ground-truth depth. However, the correct depth information of the transparent objects may already be gone in the original depth, due to errors or failures in stereo matching, and therefore taking the complete raw depth as input wouldn't help much, either. Recently, GraspNeRF [10] and EvoNeRF [11] get rid of depth sensors and instead reconstruct the shape of transparent objects from multi-view RGB images, at the cost of extra time cost for multi-view image capture on a mobile robot. Overall, none of these methods have gained informative cues of the transparent area from the depth observations, thus leaving a huge space to improve further reconstruction and grasping of the transparent objects.

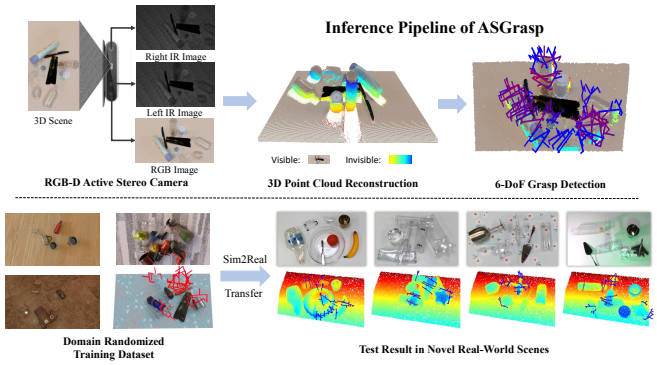

Fig. 1: Overview of proposed method ASGrasp and the dataset. Our approach takes one RGB image and left-right infrared (IR) pair as inputs, predicts visible and invisible point cloud to generate grasp pose. We train the model using the DREDS dataset and the rendered dataset based on GraspNet-1Billion in complex scenes.

In this work, we propose to leverage the raw IR observations from an active stereo camera that are commonly found in commercial products(e.g., Intel R200 and D400 family [12]) to improve transparent object depth estimation and grasping detection. Our insight is that the left and right IR observations, before going through error-prone stereo matching, carry the original information of transparent object depth while RGB information can provide extra shape priors. We thus devise an RGB-aware learning-based stereo matching network that take inputs both RGB and two IR images, based upon a popular flow estimation network[13].

Our experiments show that the predicted depth is of very high quality and passing its backprojected point cloud to the SOTA point cloud based grasping detection network, GSNet[6] can already be on par with using perfect depth point cloud input. We suspect that, limited by the single-view nature of the depth point cloud input,

[1]Samsung R&D Institute China-Beijing
[2]CFCS, School of Computer Science, Peking University
[3]Galbot
[4]Beijing Academy of Artificial Intelligence (BAAI)
†Corresponding to hewang@pku.edu.cn.

GSNet may still lack sufficient geometric cue of the invisible region, especially when lateral grasping is needed. To surpass the performance of perfect depth point cloud, we further propose to predict the second-layer depth to approximate the invisible point cloud, inspired by [14]. Via combining the visible and invisible predicted point clouds and using them as inputs to GSNet, the grasping performance even outperforms when taking input the perfect depth point cloud.

To train our networks, we extend the GraspNet-1Billion[1] dataset to a large-scale photorealistic synthetic dataset of objects with diffuse, transparent, or specular material. Comprising 115,000 sets of RGB and IR images, this dataset is generated through physics-based rendering and enhanced with diverse domain randomization. Our network exhibits the ability to readily generalize to novel real-world scenes after trained on this extensive dataset.

In summary, our main contributions are as follows:

1) We propose a novel RGB-aware two-layer stereo network for generalizable transparent object reconstruction from RGB-D active stereo camera.
2) To the best of our knowledge, we, for the first time, achieve over 90% success rate for generalizable transparent object grasping in both simulation and the real, without seeing any real training data.

II. RELATED WORK

A. Active Stereo Based Depth Estimation

Active stereo camera projects a texture into the scene using an IR projector, effectively addressing the depth estimation challenge in textureless scenes when compared to passive stereo sensors[15]. However, it still grapples with common stereo matching issues. ActiveStereoNet[12] is a fully self-supervised method that produces precise depth with subpixel precision, preserves edges, handles occlusions, and avoids over-smoothing issues. ActiveZero[16] combining supervised disparity loss and self-supervised losses to train active stereo vision systems without the need for real-world depth annotation. Active stereo cameras such as Intel RealSense include stereo cameras and RGB cameras[17]. Following the MVS framework, a triview stereo system was created in this paper by combining RGB with left IR image and right IR image, aiming to explore its effectiveness in transparent and specular scenes.

B. Transparent And Specular Scene Depth Completion

Current commercial depth sensors fail to capture depth images for transparent and specular objects. ClearGrasp[9] makes key modifications to the two-stage depth completion pipline, trying to optimize the depth image by using RGB. DepthGrasp[18] modifies the global optimization, which partially improved the performance. [19] uses the local implicit function for depth completion of transparent objects. DREDS[7] and TransCG[8] attempt to restore depth utilizing RGB information and can be considered as a one-stage depth completion. Compared to the above depth completion methods, the two-layer depth estimation proposed in this paper can complete invisible depth from the current viewpoint.

C. Grasp Network

Learning-based approaches play a significant role in the field of robot grasping[20], generating 6 DoF grasps from a point cloud or TSDF[2]. GSNet[6] takes a dense scene point cloud as input and uses a graspness-based sampling strategy to select points with high graspness. AnyGrasp[21] generates grasp poses from a partial point cloud using a geometry processing module. VGN[2] directly outputs the predicted grasp quality, gripper orientation, and opening width for each voxel in the queried TSDF volume. GIGA[3] takes the input TSDF volume and generates grasp proposals based on the learned implicit functions. Point cloud can represent irregular and non-uniformly sampled data, making them suitable for complex, unstructured environments. In this paper, we choose GSNet as grasp network.

III. METHOD

A. Problem Statement and Method Overview

In the context of an active stereo system comprising an RGB camera and two infrared (IR) cameras capturing a single viewpoint of a cluttered tabletop scene containing transparent, specular, and diffuse objects, the objective of the proposed robotic system is to detect 6-DoF grasping poses and subsequently execute grasping-to-removal operations for all the objects. The intrinsic and extrinsic parameters of the active stereo system are acquired from the factory settings. Next, we define the 6D grasp detection learning task as the mapping of a set of one RGB image $I_c \in R^{H\times W\times 3}$ and two IR images $I_{ir}^l, I_{ir}^r \in R^{H\times W}$ to a set of 6-DoF grasp poses $\{g_j | g_j = (q_j, t_j, R_j, \omega_j)\}$. Each detected grasp pose includes grasp score $q_j \in [0,1]$, grasp center $t_j \in R^3$, rotation$R_j \in SO(3)$, and opening width ω_j.

Our proposed framework consists of two main components: the scene reconstruction module, denoted as F_d, and the grasp detection module based on explicit point cloud, denoted as F_g, as illustrated in the Fig. 2. In F_d, we choose the RGB image I_c as the reference image and differentiably warp the left I_{ir}^l and right IR images I_{ir}^r to the RGB reference coordinate system, constructing a cost volume $C \in R^{H\times W\times D_h}$. This allows us to leverage the classical GRU-based stereo matching method[20]. Additionally, we introduce a second-layer depth branch that enables the network to predict not only the visible depth, refer to it as the first-layer depth $D_1 \in R^{H\times W}$ but also attempts to recover second-layer depth D_2 for objects in the scene, which help to capture the complete 3D shape of unoccluded obejcts. In F_g, we employ the two-stage grasping network F_{gsnet}[6], and thanks to the richer input point cloud information, the F_{gsnet} can predict more accurate grasp poses in the second stage.

B. Scene Reconstruction

For single-view grasping networks, one crucial aspect is the accurate recovery of scene depth, especially for transparent and specular objects. Additionally, the single-view depth only record the first intersection point of rays with the surfaces of objects in the scene. With only partial object shape information available, predicting optimal grasp poses becomes a more challenging task for the grasping network, particularly when dealing with lateral grasping orientations. To address this challenge, we propose a framework based on RAFT-like stereo matching that simultaneously recovers first layer depth for transparent and specular objects as well as occluded surface information for objects by second-layer depth prediction.

Image Features Following [13], given the left and right IR images I_{ir}^{l}, I_{ir}^{r} and RGB image I_c, we use a feature encoder to extract IR image features F_{ir}^{l}, F_{ir}^{r} at 1/4 of the $I_{ir}^{l(r)}$ image resolution, and a context encoder to extract multi-scale context features F_c at 1/4,1/8,1/16 of the RGB image resolution with 128 channels.

Epipolar Cost Volume Following [22], [23], after extacting feature maps from IR image pairs, we firstly use differentiable bilinear sampling to warp the IR features w.r.t reference RGB view at the given depth hypotheses and then construct a 3D cost volume by computing the correlation between them. Specifically, with known camera intrinsics $\{K_c, K_{ir}\}$ and relative transformations$\{[R_{c\to j}|t_{c\to j}]\}$ between RGB and IR cameras ($j = l/r$ represents left or right IR, respectively), for each pixel p_c in the RGB view, and the depth hypothesis $d := d(p)$, we can compute the corresponding pixel features in IR images as:

$$p_j = K_j \cdot (R_{c\to j} \cdot (K_c^{-1} \cdot p_c \cdot d) + t_{c\to j}) \tag{1}$$

and the cost volume $C \in R^{H\times W\times D_h}$can be computed as $c(p) = \langle F_{ir}^{l}(p_l), F_{ir}^{r}(p_r)\rangle$, where $\langle\cdot,\cdot\rangle$ denotes the dot product, and D_h is the number of the depth hypotheses. As all depths hypotheses sampled uniformly over the inverse depth range, we proceed to represent the iterative outputs in the disparity field.

Iterative Update Besides original first-layer disparity d_1, we introduce an additional branch to update a second-layer disparity d_2 to predict invisible shapes of objects. Both d_1 and d_2 are initialized to zero. In each iteration i, per-pixel first and second-layer cost features are extracted from the cost volumes using the current disparity $d_1^{(i)}, d_2^{(i)}$ respectively. These features include a hidden state $h^{(i)}$, the context features $F_c^{(i)}$ from the RGB image, which are inputted to the GRU-based update operator. The update operator outputs all increments Δd_k of the first and second disparity fields as well as a new hidden state. Then we update both the first and second-layer disparities as $\Delta d_{k+1} = d_k + \Delta d_k$, where k denotes the number of iterations.

C. Point Cloud Based Grasp Detection

Our point cloud based grasp detection network is built upon the SOTA work, GSNet [6]. The original GSNet is a two-stage grasp detector to predict dense grasp poses from a single-view point cloud. To extract comprehensive geometric features of the scene, our network F_g takes not only visible but also invisible point clouds as inputs, which are obtained by sampling from the first and second-layer depths, respectively. The visible point cloud serves as the primary resource for generating augmented grasp points for the subsequent neural network, while the invisible point cloud is used as a reference to provide additional contextual features.

D. Synthetic Dataset Generation

Existing grasping datasets like [1] don't include transparent and specular objects, while current datasets for these materials lack dense grasping annotations[8], [9], [19], [24]. We introduce a synthetic grasping dataset featuring transparent and specular objects which comprises 115k sets of RGB and IR images, along with 1 billion grasp poses.

We create this dataset using a data generation pipeline based on [7]. Initially, we modify the object materials in [1] to include diffuse, specular, or transparent properties, so we name it STD-GraspNet. Subsequently, we use Blender[25] to generate photorealistic RGB and IR images, utilizing the settings of the Realsense D415 camera. To address the challenge of sim-to-real gap, we apply domain randomization in [7]. In terms of camera poses, we not only use the original poses from [1], but also add 3k additional random camera poses to enhance diversity and mimic various distances to objects. We train our networks with STD-GraspNet, treating real data as a training data variation, thus improving real-world performance.

E. Network Training

Our training objectives consist of two main components and we train them separately:

Geometry Loss We follow [13] to calculate the L1 loss on all predicted first and second-layer disparities $\{d_1^{(i)}, d_2^{(i)}\}_{i=1}^{N}$. Same to [7], we add higher weight to the loss with in the transparent and specular objects, to concentrate more on the depth completion:

$$L_{geo} = \sum_{i=1}^{N} \gamma^{N-i}(||d_1^* - d_1^{(i)}||_1 + ||d_2^* - d_2^{(i)}||_1) \tag{2}$$

where γ=0.9, d_1^* and d_2^* are respresented as ground truth disparties.

Grasping Loss For grasping learning, we supervise the point-wise graspable landscape, view-wise graspable landscape, grasp scores and gripper widths as in GSNet[6]. Although we predict a complete point cloud, we still calculate the loss based on the visible point cloud. The whole objective can be formulated as:

$$L_{grasp} = L_o + \alpha(L_p + \lambda L_v) + \beta(L_s + L_w) \tag{3}$$

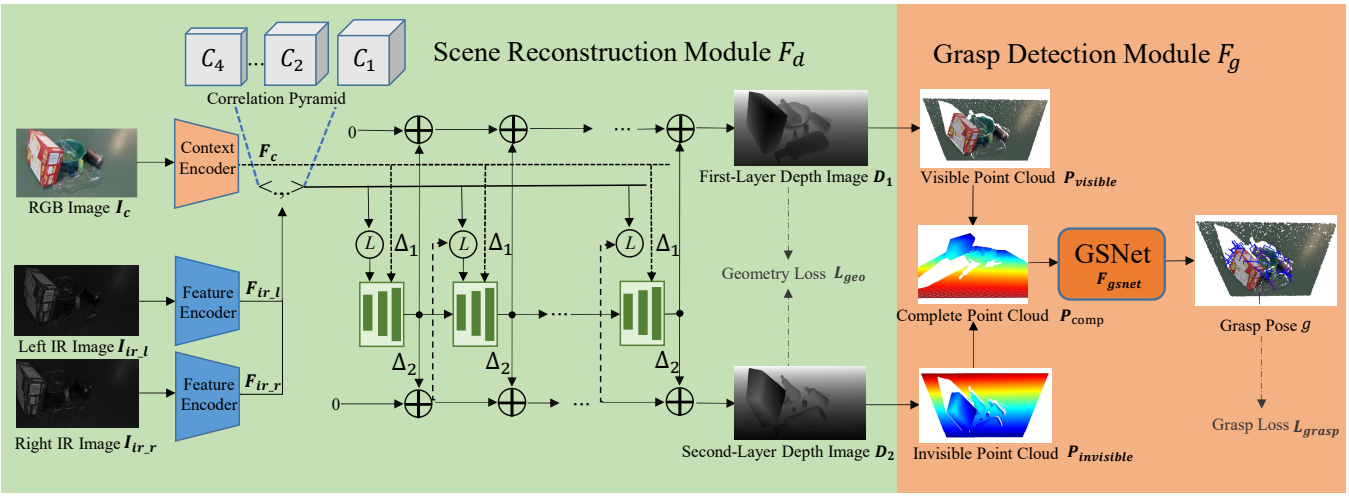

Fig. 2: The architecture of our proposed approach involves the extraction of features F_c, F_{ir}^l, and F_{ir}^r from the RGB image I_c, the left IR image I_{ir}^l, and the right IR image I_{ir}^r. F_{ir}^l and F_{ir}^r are used to construct a correlation pyramid. This correlation pyramid, along with F_c, is fed into a GRU network for the prediction of a complete point cloud, which includes the first-layer depth (visible point cloud) and the second-layer depth (invisible point cloud). Subsequently, GSNet generates grasp poses based on the complete point cloud.

where L_o is for objectness classification, L_p, L_v, L_s and L_w are for regressions of point-wise graspable landscape, view-wise graspable landscape, grasp scores and gripper widths respectively. L_p and L_s are calculated when the related points are on objects, L_v is calculated for views on seed points and L_w is calculated for grasp poses with ground truth scores > 0. We use softmax for classification task and smooth-L1 loss for regression tasks.

IV. EXPERIMENTS

A. Implementation Details

Scene reconstruction network. We implement the network with PyTorch. For all training, we use the AdamW optimizer and clip gradients to the range $[-1, 1]$. On DREDS-CatKnown, we train the model with the first-layer depth loss for 100k steps with a batch size of 4. On STD-GraspNet training split, we finetune the pre-trained model for another 100k steps. For all experiments, we use 12 update iterations during training.

Grasp detection network. The input point cloud is cropped with a depth range of [0.25m, 1.0m]. The visible point cloud is down-sampled to 15000 for training and 25000 for inference, and invisible point cloud is always sampled to 10000. The graspness threshold is 0 for both training and inference. The model is trained on four NVIDIA RTX 3090 GPU that takes about 1 day for 10 epochs with a batch size of 4. We utilize the Adam optimizer with an initial learning rate of 0.001 and decay 5% every epoch.

B. Experiment Setup

Hardware Setup. We use a 7-DoF Franka Panda arm with its default two-finger gripper, on which we mount an active stereo RGB-D camera, Intel Realsense D415.

Simulation Setup. We use PyBullet [26] for physical simulation of grasping. We use an active stereo sensor simulation [7] built upon Blender[25] to render realistic RGB and IR images.

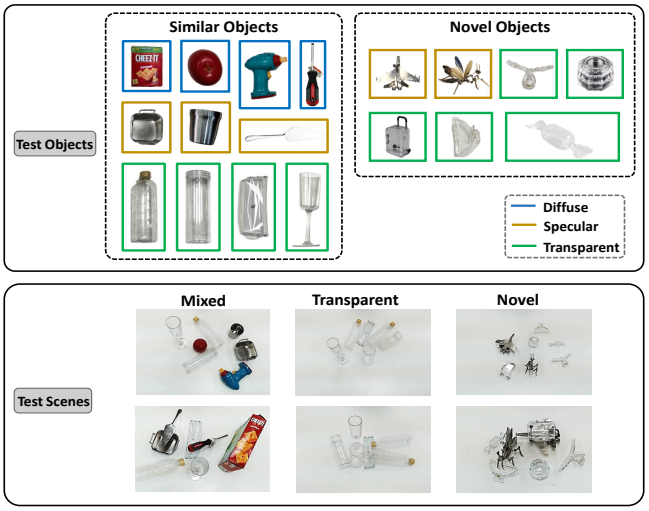

Fig. 3: Real world evaluation dataset

C. Scene Reconstruction Experiments

Evaluation Metrics. We evaluate the performance of transparent objects depth completion by three metrics, following in [19]. 1) **RMSE**: the root mean squared error, 2) **REL**: the mean absolute relative difference, 3) **MAE**: the mean absolute error. For the first-layer depth, we resize the prediction and GT to the resolution of 126×224 for fair comparison, while for the second-layer depth, we resize them to 224×224 resolution consistent with the SwinDR. We evaluate all objects area and challenging area (specular and transparent objects), respectively.

Result comparisons. We first compare our first-layer depth with SOTA methods on the test split of DREDS-CatKnown [7], including LIDF [19], NLSPN [27] and SwinDR [7]. All baselines are trained on the training split

of DREDS-CatKnown. Subsequently, we compare our results with SwinDR on STD-GraspNet test split after fine-tuning on STD-GraspNet training split. As shown in Table I and Table II, we achieve the best performance compared to other methods on DREDS and STD-GraspNet for the first-layer depth. Due to the absence of a suitable baseline, we can only report our performance for the second-layer depth. We also provide a qualitative comparison of the predictions for both layers in Fig. 4. The simulated raw point cloud contains missing and wrong points. We observe that SwinDR restores missing parts well, but performs poorly in areas with incorrect data. Our method performs much better in both the first and second layers, yielding a high-quality complete point cloud reconstruction.

TABLE I: Depth completion quantitative comparison on DREDS-CatKnown dataset.(all objects/specular&transparent)

Methods	RMSE↓	REL↓	MAE↓
NLSPN	0.010/0.011	0.009/0.011	0.006/0.007
LIDF	0.016/0.015	0.018/0.017	0.011/0.011
SwinDR	0.010/0.010	0.008/0.009	0.005/0.006
Ours(D1)	**0.007/0.007**	**0.006/0.006**	**0.004/0.004**

D. Evaluation Metrics and Baselines for Grasping

Evaluation Metrics. We evaluate the performance by two metrics: **Success Rate(SR)**, the ratio of the number of successful grasp and total attempts; and **Declutter Rate(DR)**, the mean percentage of removed objects across all rounds.

Baselines. 1) **GSNet**, which takes input single-view raw depth point cloud; 2) **SwinDR-GSNet** that exploits the previous SOTA depth restoration method for specular and transparent scene, SwinDRNet[7], to first process the raw depth. To be fair, it is fine-tuned on our STD-GraspNet training split to fit GSNet.

E. Simulation Grasping Experiments

Experimental Protocol. We conduct grasping experiments on STD-GraspNet test-split to evaluate all the methods, which contains 90 scenes divided into 3 categories as seen, similar and novel. Each scene includes 5 to 10 objects with a combination of diffuse, transparent, and specular materials and we uniformly sampled 30 veiwpoints from the total 256 viewpoints for testing. And for each trial in one scene, the robot arm performs grasping and removal of objects until the workspace is cleared, no any grasp detection or two consecutive failures are reached.

Results and Analysis. Table III shows our simulation grasping result for different point cloud inputs with GSNet trained with corresponding data. GSNet(RealRaw) with simulated raw depth map which contain incorrect depth or missing on transparent objects, leading the low performance. Although SwinDR improves the depth to some extent, it still suffers from over-smoothing, resulting in a degradation of grasping performance. For GSNet(SynVisible), input with our predicted visible point cloud achieves similar performance with Oracle(SynVisible), thanks to the original RGB and IR images, learning based stereo matching method and diverse dataset. But we also observe that certain metrics even outperformed the Oracle(SynVisible), which could be caused by the bleeding artifacts near object boundaries, resulting the grasping network to favor grasping pose from the front of objects. For GSNet(SynComplete), utilizing our two-layer complete point cloud has the best performance overall. We observe that it outperformed the Oracle(SynVisible). This further shows the potential for improved grasping accuracy with a complete scene representation. However, there is still a noticeable gap compared to the Oracle(SynComplete).

F. Real Robot Experiments

To evaluate the performance of our method in real world, we conduct grasping experiments as well.

Experimental Protocol. In our real-world grasping experiments, there are 18 test objects, including 11 similar objects and 7 novel objects. Novel objects refer to objects not in the training data for both depth restoration and grasp estimation. For the evaluation scenes, we create 6 test scenes, with 2 scenes each of mixed, transparent, and novel objects. The mixed scene contains 2 diffuse objects, 2 specular objects, and 3 transparent objects selected from the group of similar objects. The transparent scene is consisted of 6 similar transparent objects, while the novel scene includes 2 novel specular objects and 5 novel transparent objects. The detailed of test objects and scenes are depicted in Fig. 3. We closely replicate the above 6 scenes for each test baselines. The task is to pick the objects to placement location until the workspace is cleared, no grasp pose generation or 15 attempts are reached.

Results and Analysis. Table V shows our real-world grasping evaluation results on the robot, where our method with complete point cloud achieves the highest success rate and declutter rate all test scenes. Compared with original GSNet, our ASGrasp significantly increases the success rate and declutter rate by 55.8% and 67.5% in average. In addition, both visible point cloud and complete point cloud as input can achieve 100% declutter rate. It is worth note that, without depth restoration, the performance of grasping transparent objects is exceedingly low, with a success rate of only 7.7% and declutter rate of 8.3% in transparent scene. In contrast, our approach achieves a better performance, achieving over 90% success rate and declutter rate in transparent scenes. Moreover, it's worth mentioning that SwinDR-GSNet baseline achieves 90% success rate in novel scene, which is significantly higher than simluation experiments. The reason for this is after several successful grasping, with no further grasp candidate generation, the task ends. In consequence, the novel scene declutter rate of SwinDR-GSNet is lower than its average declutter

TABLE II: Evaluation of reconstruction on STD-GraspNet test split (all objects/specular&transparent).

Methods	Evaluation on STD-GraspNet(first-layer)			Evaluation on STD-GraspNet(second-layer)		
	RMSE↓	REL↓	MAE↓	RMSE↓	REL↓	MAE↓
DREDS	0.0115/0.0124	0.0158/0.0185	0.0061/0.0073	-	-	-
Ours	**0.0076/0.0080**	**0.0090/0.0104**	**0.0036/0.0042**	**0.0181/0.0175**	**0.0200/0.0196**	**0.0090/0.0088**

TABLE III: Grasp success rate(%) of cluster removal in simulation (overall/diffuse/specular/transparent).

Depth Source for Testing	Depth Source for Training	Seen	Similar	Novel	# of grasps
Depth(SimRaw)	GSNet(RealRaw)	49.7/51.5/52.9/36.7	51.7/54.6/51.5/42.6	52.3/56.5/52.8/42.1	13.8k
SwinDR	GSNet(SimVisible)	63.9/65.3/68.8/51.9	61.8/65.3/64.8/48.1	58.5/61.8/62.3/44.7	17.7k
SwinDR	GSNet(RealRaw)	71.1/72.3/73.9/63.9	72.5/74.5/75.3/63.6	67.2/68.8/68.6/61.6	21.0k
Ours($P_{visible}$)	GSNet(SynVisible)	91.4/**91.7**/91.9/90.2	90.6/90.2/91.2/90.4	87.3/89.8/87.8/82.6	22.6k
Ours(P_{comp})	GSNet(SynComplete)	**91.9/91.7/93.0/90.7**	**91.7/90.9/92.3/92.0**	**89.6/91.3/89.8/86.9**	22.4k
Oracle(SynVisible)	GSNet(SynVisible)	91.3/92.1/91.6/89.7	89.1/90.3/87.7/89.7	88.9/89.9/89.3/86.5	22.6k
Oracle(SynComplete)	GSNet(SynComplete)	94.1/95.4/93.2/94.1	93.1/93.8/92.7/93.5	91.7/91.9/93.0/88.7	22.5k

TABLE IV: Ablation study on depth completion (all objects/specular&transparent).

Input		Training Dataset		GraspNet Dataset Test Split(first-layer)			GraspNet Dataset Test Split(second-layer)		
L&R IR	RGB	DREDS	GraspNet	RMSE↓	REL↓	MAE↓	RMSE↓	REL↓	MAE↓
✓	×	✓	✓	0.0084/0.0088	0.0097/0.0113	0.0038/0.0045	0.0187/0.0188	0.0206/0.0205	0.0092/0.0092
✓	✓	×	✓	0.0081/0.0086	0.0101/0.0117	0.0040/0.0047	0.0190/0.0185	0.0222/0.0219	0.0099/0.0098
✓	✓	✓	✓	**0.0076/0.0080**	**0.0090/0.0104**	**0.0036/0.0042**	**0.0181/0.0175**	**0.0200/0.0196**	**0.0090/0.0088**

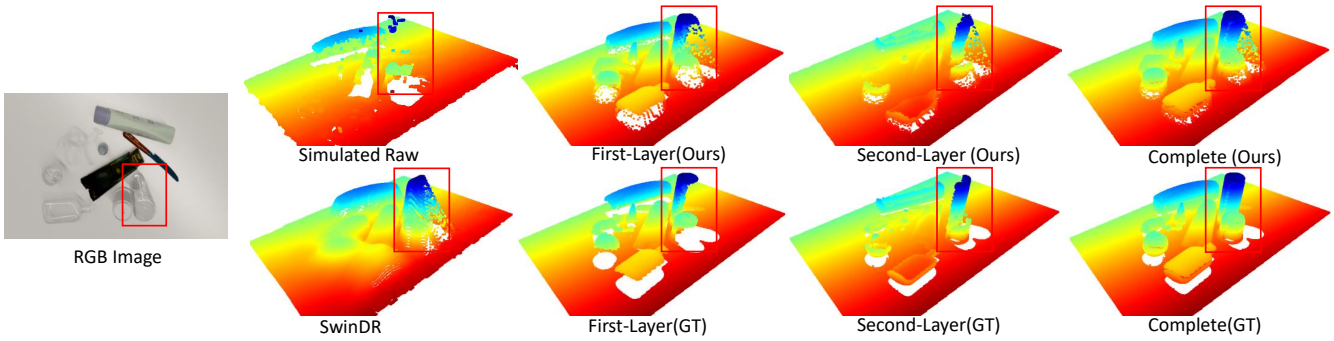

Fig. 4: Qualitative comparison of point cloud reconstruction for an exemplar test data from STD-GraspNet.

TABLE V: Grasp success rate(%)/declutter rate(%) of clutter removal in the real world

	Mixed SR(%)/DR(%)	Transparent SR(%)/DR(%)	Novel SR(%)/DR(%)	Overall SR(%)/DR(%)
GSNet	50.0/50.0	7.7/8.3	83.3/35.7	39.4/32.5
SwinDR-GSNet	80.0/92.8	41.7/83.3	90.0/64.3	57.1/80.0
Ours(P_1)	87.5/**100**	85.71/**100**	82.4/**100**	85.1/**100**
Ours(P_{comp})	**93.3/100**	**100/100**	**93.3/100**	**95.2/100**

rate. In addition, our overall success rate in the real world is even higher than in simulation. This is first due to our seamless sim2real domain transfer and partly because of the differences in setups between the real robot and the simulator. The simulator is sensitive that once a collision is detected, the gripper will stop moving.

G. Ablation Studies

To analyze the design of our method, we conduct an ablation study on scene reconstruction, considering different inputs and training datasets. 1) To varify the effectiveness of RGB images, we remove the RGB context branch and solely utilize the left and right IR images as input. 2) train the model only on the GraspNet Dataset.

The results are shown in Table IV. In comparison to our proposed method, the inclusion of RGB input led to a noticeable enhancement in depth recovery performance at both the first and second layers. Additionally, we observe improvements when incorporating the DREDS dataset, indicating that leveraging diverse data can benefit the learning of scene reconstruction network.

We attempt end-to-end training for the two modules but observe a slight performance drop on the test split of STD-GraspNet. We suspect that this is due to overfitting to the limited size of the finetuning grasping dataset. In this work, we don't further enlarge the grasping training data for a fair comparison with GraspNet-1Billion based method, *e.g.*, GSNet.

V. CONCLUSIONS

In this work, we propose an active stereo camera based 6-DoF grasping method, ASGrasp, for transparent and specular objects. We present a two-layer learning based stereo network which reconstructs visible and invisible parts of 3D objects. The following grasping network can leverage rich geometry information to avoid confused grasping. We also propose a large-scale synthetic data to bridge sim-to-real gap. Our method outperforms competing methods on depth metric and clutter removal experiments in both simulator and real world.

VI. ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (No. 62306016).

References

[1] 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*, pp. 11444–11453, 2020.

[2] M. Breyer, J. J. Chung, L. Ott, R. Siegwart, and J. Nieto, "Volumetric grasping network: Real-time 6 dof grasp detection in clutter," in *Conference on Robot Learning*, pp. 1602–1611, PMLR, 2021.

[3] Z. Jiang, Y. Zhu, M. Svetlik, K. Fang, and Y. Zhu, "Synergies between affordance and geometry: 6-dof grasp detection via implicit representations," *arXiv preprint arXiv:2104.01542*, 2021.

[4] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13438–13444, IEEE, 2021.

[5] M. Gou, H.-S. Fang, Z. Zhu, S. Xu, C. Wang, and C. Lu, "Rgb matters: Learning 7-dof grasp poses on monocular rgbd images," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13459–13466, IEEE, 2021.

[6] C. Wang, H.-S. Fang, M. Gou, H. Fang, J. Gao, and C. Lu, "Graspness discovery in clutters for fast and accurate grasp detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15964–15973, 2021.

[7] Q. Dai, J. Zhang, Q. Li, T. Wu, H. Dong, Z. Liu, P. Tan, and H. Wang, "Domain randomization-enhanced depth simulation and restoration for perceiving and grasping specular and transparent objects," in *European Conference on Computer Vision*, pp. 374–391, Springer, 2022.

[8] H. Fang, H.-S. Fang, S. Xu, and C. Lu, "Transcg: A large-scale real-world dataset for transparent object depth completion and a grasping baseline," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7383–7390, 2022.

[9] S. Sajjan, M. Moore, M. Pan, G. Nagaraja, J. Lee, A. Zeng, and S. Song, "Clear grasp: 3d shape estimation of transparent objects for manipulation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3634–3642, IEEE, 2020.

[10] Q. Dai, Y. Zhu, Y. Geng, C. Ruan, J. Zhang, and H. Wang, "Graspnerf: multiview-based 6-dof grasp detection for transparent and specular objects using generalizable nerf," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1757–1763, IEEE, 2023.

[11] J. Kerr, L. Fu, H. Huang, Y. Avigal, M. Tancik, J. Ichnowski, A. Kanazawa, and K. Goldberg, "Evo-nerf: Evolving nerf for sequential robot grasping of transparent objects," in *6th Annual Conference on Robot Learning*, 2022.

[12] Y. Zhang, S. Khamis, C. Rhemann, J. Valentin, A. Kowdle, V. Tankovich, M. Schoenberg, S. Izadi, T. Funkhouser, and S. Fanello, "Activestereonet: End-to-end self-supervised learning for active stereo systems," in *Proceedings of the european conference on computer vision (ECCV)*, pp. 784–801, 2018.

[13] L. Lipson, Z. Teed, and J. Deng, "Raft-stereo: Multilevel recurrent field transforms for stereo matching," in *2021 International Conference on 3D Vision (3DV)*, pp. 218–227, IEEE, 2021.

[14] D. Shin, Z. Ren, E. B. Sudderth, and C. C. Fowlkes, "3d scene reconstruction with multi-layer depth and epipolar transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2172–2182, 2019.

[15] K. Konolige, "Projected texture stereo," in *2010 IEEE International Conference on Robotics and Automation*, pp. 148–155, IEEE, 2010.

[16] I. Liu, E. Yang, J. Tao, R. Chen, X. Zhang, Q. Ran, Z. Liu, and H. Su, "Activezero: Mixed domain learning for active stereovision with zero annotation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13033–13042, 2022.

[17] L. Keselman, J. Iselin Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel realsense stereoscopic depth cameras," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 1–10, 2017.

[18] Y. Tang, J. Chen, Z. Yang, Z. Lin, Q. Li, and W. Liu, "Depthgrasp: depth completion of transparent objects using self-attentive adversarial network with spectral residual for grasping," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5710–5716, IEEE, 2021.

[19] L. Zhu, A. Mousavian, Y. Xiang, H. Mazhar, J. van Eenbergen, S. Debnath, and D. Fox, "Rgb-d local implicit function for depth completion of transparent objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4649–4658, 2021.

[20] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on robotics*, vol. 30, no. 2, pp. 289–309, 2013.

[21] H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, and C. Lu, "Anygrasp: Robust and efficient grasp perception in spatial and temporal domains," *IEEE Transactions on Robotics*, 2023.

[22] Z. Ma, Z. Teed, and J. Deng, "Multiview stereo with cascaded epipolar raft," in *European Conference on Computer Vision*, pp. 734–750, Springer, 2022.

[23] F. Wang, S. Galliani, C. Vogel, and M. Pollefeys, "Itermvs: Iterative probability estimation for efficient multi-view stereo," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8606–8615, 2022.

[24] X. Liu, R. Jonschkowski, A. Angelova, and K. Konolige, "Keypose: Multi-view 3d labeling and keypoint estimation for transparent objects," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11602–11610, 2020.

[25] O. Blender, "Blender—a 3d modelling and rendering package," *Retrieved. represents the sequence of Constructs1 to*, vol. 4, 2018.

[26] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," 2016.

[27] J. Park, K. Joo, Z. Hu, C.-K. Liu, and I. So Kweon, "Non-local spatial propagation network for depth completion," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pp. 120–136, Springer, 2020.