KGNv2: Separating Scale and Pose Prediction for Keypoint-based 6-DoF Grasp Synthesis on RGB-D input

Yiye Chen[1], Ruinian Xu[1], Yunzhi Lin[1], Hongyi Chen[1], and Patricio A. Vela[1]

***Abstract*— We propose a new 6-DoF grasp pose synthesis approach from 2D/2.5D input based on keypoints. Keypoint-based grasp detector from image input has demonstrated promising results in the previous study, where the additional visual information provided by color images compensates for the noisy depth perception. However, it relies heavily on accurately predicting the location of keypoints in the image space. In this paper, we devise a new grasp generation network that reduces the dependency on precise keypoint estimation. Given an RGB-D input, our network estimates both the grasp pose from keypoint detection as well as scale towards the camera. We further re-design the keypoint output space in order to mitigate the negative impact of keypoint prediction noise to Perspective-n-Point (P*n*P) algorithm. Experiments show that the proposed method outperforms the baseline by a large margin, validating the efficacy of our approach. Finally, despite trained on simple synthetic objects, our method demonstrate sim-to-real capacity by showing competitive results in real-world robot experiments. Code is available at: `https://github.com/ivalab/KGN`.**

I. Introduction

Robotic grasping is a fundamental yet demanding problem, requiring both object perception as well as geometric reasoning based solely on sensor input. Past reasearchers simplified the problem by constraining the grasp poses into SE(2) space, assuming that the camera looks at the scene vertically from the top, and the gripper reaches perpendicularly to the support plane [1], [2], [3]. The restriction allows the planar grasp methods to represent grasps as simple oriented rectangles or keypoints in the image space, which permits directly adopting existing data-driven tools from computer vision tasks, such as object [4] or keypoint [5] detectors. However, it also neglects possible grasp poses reaching from other directions, which potentially impedes its utility in constrained environments [6], [7].

The limitation of planar grasp assumption has motivated recent exploration of 6-DoF grasp synthesis, which allows grasp poses in full SE(3) space. Point-cloud-based methods, utilizing point set feature extractors like PointNets [8], [9], have achieved great success in generating or evaluating 6-DoF grasp poses directly from depth sensor data. However, these methods face empirical limitations such as poor grasp poses for small-scale objects due to limited point perception [10], and compromised performance in the presence of sensor noise. A point sampling strategy [10] has been proposed to balance object scales, but it increases computational cost due to the need of an additional instance segmentation network. And the vulnerability to input disturbance remains a concern.

Consequently, the utilization of 2D/2.5D input for 6-DoF grasp detection has gained attention due to the additional visual information offered by color images. The visual clue provided by RGB modality, which can be effectively extracted by modern convolutional neural networks (CNN), can not only facilitate discerning small objects that are negligible from depth point cloud input, but also improves robustness against noise in depth sensor [11]. Despite demonstrating promising results, existing methods [11], [12] still utilize a 3D representation of grasp poses, necessitating the network to estimate 3D information from 2D input. As a result, expensive annotation, such as surface normal, is needed for training [12], or heavy distretization of SO(3) space is required [11].

To avoid the need for directly estimating 3D parameters, Keypoint-GraspNet (KGN) [13] seperates the 2D-to-3D recovery stage out of the network. Instead of using a 3D representation, KGN represents a grasp pose as a set of gripper keypoints in the image space and recovers the SE(3) pose from the 2D keypoints with the P*n*P algorithm [14]. KGN avoids discretization error, as keypoint coordinates are continuous in the image space, and removes the requirement for estimating surface normal directions. However, imprecise keypoint proximity prediction causes unstable estimation of the *scale factor*, which is the magnitude of translation of a grasp pose from the camera, especially in a novel test domain, such as training on synthetic data and testing on real-world data. KGN heuristically addresses the issue by adopting the perceived depth as scale, which reduces the pose accuracy due to occlusion and depth sensor noise.

In this paper, we introduces KGNv2, an improved keypoint-based grasp detection network that enhances the accuracy of grasp pose detection. The network gets around the above issue by predicting pose and scale separately, which eliminates the need for accurate keypoint proximity estimation and improves the accuracy of generated pose. The keypoint output space is re-designed by normalizing with estimated scale, which further enhances the precisioin of estimated pose. The proposed modifications are simple yet effective, greatly improving performance on the primitive shape dataset from [13], and the network can generalize to realistic objects with significant shape variations in real-world experiments, indicating the potential of training grasp detectors on virtual data with primitive geometries, where obtaining ground-truth labels is easier.

[1] Y. Chen, R. Xu, Y. Lin, H. Chen, and P.A. Vela are with the School of Electrical and Computer Engineering, and the Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA. {yychen2019, rnx94, ylin466, hchen657, pvela}@gatech.edu

II. Related Work

In this paper, we narrow our review on literature of learning-based 6-DoF grasp detection with anti-padel end-effector. Other related areas include dexeterious grasp detection, planar grasp synthesis, and model-based grasp detection. They have been thoroughly reviewed by other survey papers [15], [16] and are outside the scope of this paper.

A. Point-Cloud Methods

With the emergent deep point set encoders such as PointNets [8], [9] and DeCo [17], the majority of literature explores detecting 6-DoF grasp poses from point cloud input [18]. Early effort employs a generate-then-evaluate process, where a discriminative model to predict grasp outcome is necessary [19]. PointNetGPD [20] uses a geometry-based heuristic approach [21] to sample from SE(3) spaces, and trains a network to process enclosing point sets to score the grasp pose. But the insufficient quality of sampled grasp poses limits the overall performance. 6DoF-GraspNet [22] replaces the sampling-based candidate proposal approach with deep generator trained with Generative Adversial Network (GAN) or Vairational Autoencoder (VAE) objective. Other approaches [23], [24] investigate refining the initial pose proposal by increasing the score estimated by a learnt evaluator.

The above pipeline is time-consuming due to multiple forward passes of point-cloud networks. Driven by large scale grasping datasets [25], [26], recent approaches turn their attention to end-to-end grasp detection - with both grasp pose parameter and confidence estimated by a single model. The key difference lies in the grasp representation choice. S4G [27] chooses the SE(3) representation, and directly regresses the rotational and translational parameters anchored on point with high confidence. To enable multiple detection per point, GDN [28] extends the idea with a coarse-to-fine representation idea, where they first perform classificaiton on a set of discrete angular grids, and then regress translation and rotation refinement values for high-confidence candidates. Another line of approach argues in favor of explicit contact physics reasoning. They adopt the representation of two contact points plus pitching angle [29] [30], where the assumption is at least one contact point should be visible from partial point cloud.

Point-cloud methods share some common drawbacks, which are studied by recent literature. Due to the high processing time to extract geometric information from the coordinates enumeration, truncating input point volume, by either downsampling [27] or target segmentation [22], is necessary. L2G [30] alternatively designs an learnable sampler, which can be jointly tuned in the end-to-end training process. It assumes properly designed sampling procedure can retain critical information for grasp synthesis, which is not always the case especially for high-resolution input. Another limitation for point-cloud methods is their bias towards larger objects due to dominating point number. Ma et al. [10] allieviate the issue by balanced sampling based on instance segmentation mask, which relies on external segmentation module that increases computational cost.

B. Image-based Methods

Contrary to point cloud, images are faster to process with modern networks and are explicit with visual relationship, which can address the above issues. Hence, several recent research start to study 6-DoF grasp detection from image input [31]. It is shown that the color modality can also improve the robustness against depth noise [11]. However, the intial exploration still relies on 3D representaion of grasp poses, such as contact-point-based description [12], which fails to leverage existing knowledge about 3D-to-2D camera projection.

KGN [13], on the other hand, adopts 4 keypoints in the image space to describe a 3D grasp pose, in which the keypoints can be efficiently synthesized by well-studied keypoint detector such as CenterNets [5], [32]. Designing them to be the projection of virtual 3D points in the gripper frame with predefined coordinates, a PnP algorithm is applied to recover 3D pose from 2D keypoints leveraging well-established camera projection principle. Given fixed 3D coordinates, the *relative location* and *absolute distance* between 2D keypoints determine the *pose up to a scale* and *scale factor*, respectively (e.g. closer keypoints means the gripper frame is further to the camera plane). However, the prediction of keypoint proximity is found to be unstable under novel test environment [13]. In this work, we relax the requirement of precise distance prediction by separately predicting the scale. Based on the estimated scale, we further redesign the keypoint output space factoring in the influence of prediction noise on the relative location. We show that our modified network, named KGNv2, achieves great performance gain.

III. Methodology

A. Problem Definition

Given a monocular RGB-D input, our goal is to synthesize a set of 6-DoF grasps with grasp pose $g \in SE(3)$ and associated open width w that can pick up objects perceived by the image *without* converting the input into 3D representations such as point cloud. The problem is challenging since the input is in 2D image space while output is in 3D space.

Our method, as illustrated in Fig. 1, separately predicts poses from keypoints, the scale of pose, and the open width.

B. Pose Estimation with Scale-Normalized Keypoint

Inspired by KGN [13], we adopt a keypoint-based strategy that leverages well-established camera 3D-to-2D projection principle to estimate grasp poses up to a scale. Specifically, given RGB-D input, KGNv2 predicts a set of *grasp centers* $\{c_i^m\}_i$, defined as the center point between gripper tips, for each orientation interval of line segment between gripper tips in the image space: $m \in \{1, 2, \cdots, M\}$. The design of orientation interval is to enable simultaneous detection of multiple grasps that share the same center, which is useful for generating diverse candidate set for rotationally symmetric

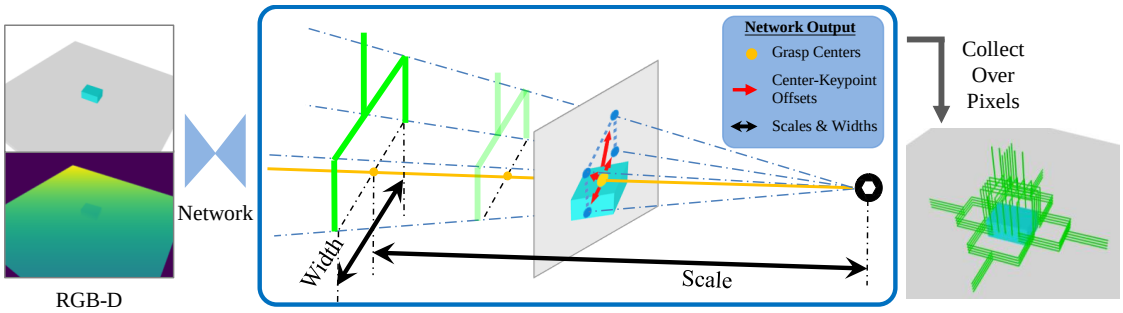


Fig. 1: **Overview of KGNv2.** Given an RGB-D input, our model predicts pixel-wise candidate grasp poses. It estimates the *pose up to a scale* by applying P*n*P algorithm on generated image-space keypoint coordinates with camera intrinsic matrix. The keypoints are obtained by predicted grasp centers and offsets. Then grasp *scale* (the magnitude of translation) as well as the open width are inferred, which complete the grasp pose parameters. The final synthesized grasp set is the collection of results over high-confident pixels.

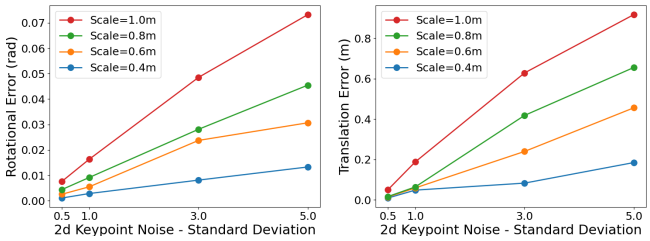


Fig. 2: Synthetic study on the relationship between pose scale and pose recovery error from P*n*P algorithm [14] due to noise. With larger grasp pose scale (grasp is further from the camera), both rotational and translational error increases under all noise levels. The observation motivates us to predict scaled keypoint location as in §III-B.

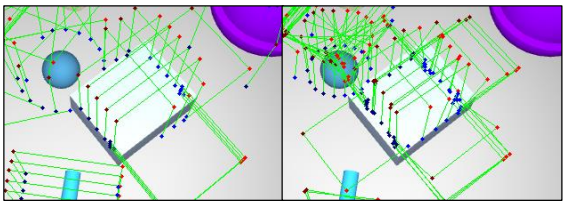


Fig. 3: Example of inaccurate keypoints proximity prediction on primitive shape data. The predicted keypoints exhibit greater proximity to each other than the ground truth keypoints, possibly due to the influence of visual disturbance from surrounding objects, resulting in imprecise scale estimation.

objects(e.g. grasps for a ball). It also functions as a non-maximum suppression mechanism by considering grasps with overlapping centers and similar orientations to be highly simillar, resulting in only one grasp being retained. The grasp centers are detected from heatmap $Y \in [0,1]^{W' \times H' \times M}$, where W' and H' represents resolution of downscaled feature map. As we will discuss soon, center-based strategy provides a simple way to group keypoints. It is also straightforward to fuse with features extracted from other modalities, such as language [33].

With grasp centers, the keypoints' locations $\{(p^m_{i1}, p^m_{i2}, p^m_{i3}, p^m_{i4})\}$ are predicted based on offset estimation. Our network learns to generate center-keypoint offset vector map O, which encodes the displacement from center to keypoints for each center and orientation. The keypoints locations can be obtained from the centers and offsets, and are then fed into IPPE [14] algorithm specifically designed for coplaner P*n*P problem to produce 6-DoF grasp pose. The final synthesized grasp set is the collection of results from grasp centers with high confidence.

Scale-Normalized Keypoint Prediction A natural choice of keypoint design is to define pair-wise distance equal to open width, which places two keypoints on gripper tips and other two above to form a square. In this way, 2D keypoints on the image are adaptive to the size of graspable part.

However, such a design **naturally bias towards grasp pose with small *scale* (in close proximity to camera) in training.**. Unlike human pose or object pose estimation problem [5], grasp poses are larger in quantity per image due to multiple grasps feasible for each object, exhibiting a distribution of scales. For fixed size grasps, the keypoints appear closer on the image for further poses as a result of perspective projection. However, one property for P*n*P algorithm is that *recovered grasp pose for close-by points are prone to larger error under the same level of noise, as keypoint structure is more strongly disrupted.* Hence, same error in Euclidean keypoint space transfers to larger error in pose space for large-scale grasp annotations.

To empirically confirm this property, we conduct a synthetic experiment to examine the relationship between scale and pose recovery error. We randomly sample grasp orientation at the origin, and place cameras with various distance along optical axis. Then we recover gripper pose with canonical keypoint projections injected by gaussian noise, and calculate the average rotational and translational error, defined in the same way as in §IV-C. The result is shown in Fig. 2, which indicates that both errors increase as a function of scale conditioned on any noise level.

We propose to predict image-space keypoints normalized

by the scale. The idea is related to human/object keypoints prediction with area-normalization [34], [35] or object-size-normalization [36], but for grasp pose estimation problem we normalize with pose proximity to camera center to introduce scale invariance. Specifically, we scale the offset which determines the proximity of keypoints. For each grasp center c and associated *actual* offset vectors $\tilde{O}_c$ and scale S_c, our network is tasked to predict:

$$O_c = \tilde{O}_c / S_c$$

where the predicted scale (see Sec. III-C) is used in the inference. The scale-normalized keypoint design reduces the susceptibility of pose recovery to noise for further grasp poses. Assuming the prediction of scaled offsets suffers from zero-mean Gaussian noise $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Then the perturbance of original offsets is reduce to $(1/S_c)\varepsilon \sim \mathcal{N}(0, \sigma^2/S_c^2)$, whose standard deviation is decreased by the scales, leading to more accurate pose recovery. We show in Sec. IV-C that this design improves the pose estimation accuracy.

C. Scale Prediction

Although keypoints with open width prediction are sufficient to recover full SE(3) pose from P*n*P algorithm, empirically the generated scales are inaccurate as the proximity between keypoints cannot be stably predicted, especially under domain shifts. For example, when testing on multi-object scenes from the Primitive Shape dataset [13], keypoint detector trained on single-object data tends to produce keypoints that are more closely grouped than the ground truth. Since the distance to the camera optical center is inversely proportional to the size of objects in the image, imprecise image-space proximity can lead to erroneous scale estimation.

KGN [13] mitigates the problem by heuristically replacing the predicted scale with the perceived depth at the grasp center from the sensor. That leads to two potential problems. First, center depth is not identical to grasp translation scale, as the center point is not observable surface point. When grasping a box, the gripper enclosing center would fall inside of the box, which makes it unperceivable. Hence, the heuristics will introduce additional error. Furthermore, depth sensors may not always provide reliable measurements. Perceived depth maps can be affected by various sources of noise and may contain missing values [37]. As a result, the raw depth values may be less informative for accurately estimating the scale of a grasp pose.

Rather than conditioning the scale estimation on keypoints or raw depth value, KGNv2 directly predicts a scale map $S \in \mathbb{R}^{H' \times W' \times M}$ for each pixel and orientation. Each grasp pose can be easily assocatiated with the scale prediction at the corresponding grasp center location and orientation interval. Although similar to depth prediction problem, scale estimation given solely RGB input is essentially ill-posed, we assume the noisy depth map input serves as additional signal that can reduce the ambiguity. With accurate scale predictions, the translation magnitude from the P*n*P can be easily refined: Suppose for a predicted grasp center c and an orientation interval m, the rotation and translation given by P*n*P algorithm is: $\tilde{g} = \{\tilde{R}, \tilde{T}\}$, then the final pose combined with the scale prediction:

$$\hat{g} = \{\hat{R}, \hat{T}\}) = \{\tilde{R}, \frac{S_{c,m}}{\|\tilde{T}\|}\tilde{T}\})$$

D. Final Loss

The training of the network requires labels for all branch outputs, which can be generated easily given only annotation of grasp poses and camera intrinsic and extrinsic matrices. The objective for training given ground truths involves the penalty-reduced focal loss for heatmap L_Y, plus the L_1 regression loss for center-keypoint offsets L_O, open width L_W, and translation scale L_S on labeled grasp centers. The final loss is the weight sum of the four losses:

$$L = \gamma_Y L_Y + \gamma_O L_O + \gamma_W L_W + \gamma_S L_S$$

In this paper we choose: $\gamma_Y = 1$, $\gamma_O = 1$, $\gamma_W = 10$, $\gamma_S = 10$.

IV. Experiments

A. Synthetic Dataset

Following [13], we use the Primitive Shape (PS) dataset for training our network. The dataset is a synthetic dataset generated by spawning objects of simple shapes with random pose on the tabletop, which is the most common evaluation scenerio for grasp detectors. The shapes we use involves: *Cylinder*, *Ring*, *Stick*, *Sphere*, *Semi-sphere*, and *Cuboid*. Ground truth grasps are annotated by sampling evenly distributed instances from *grasp families* [38] - the closed-form grasp pose distributions parameterized by the shape type and sizes, which is assumed to sufficiently cover the feasible grasp modes for a given primitive shape based on human expertise due to the simplicity of the object geometry.

We choose the dataset since shape decomposition proves to be a very effective strategy that drives force in grasp synthesis research for years [39], [40], [38], [41]. The grasp label generation approach is significanly more cost-effective compared to sample-then-verify strategy based on simulators [25], [42], [43]. Furthermore, it mitigates potential bias or inaccuracies in the labeling process that can arise from sampling artifacts [44], thereby avoiding negative impact on training or evaluation process.

The PS dataset contains 1000 *single-object* scenes, divided into 800 training scenes and 200 test scenes. For each scene, RGB-D data is rendered from 5 random camera poses, leading to 4000 training data and 1000 test data. *In addition*, we generate a multi-object PS dataset of the same quantity, where all 6 shapes with random size and color are spawned in each scene and the grasp poses causing collisions are removed. We increase the annotation sample density for test splits to verify the extrapolation ability of trained grasp detector from sparse examples.

TABLE I: Vision Dataset Evaluation

Methods	**Single-Object Evaluation (GSR% / GCR% / OSR%)**			**Multi-Object Evaluation (GSR% / GCR% / OSR%)**		
	1cm + 20°	2cm + 30°	3cm + 45°	1cm + 20°	2cm + 30°	3cm + 45°
Contact-Graspnet†	29.9 / 24.9 / 77.0	60.1 / 32.0 / 81.7	81.6 / 36.5 / 84.2	22.1 / 15.5 / 44.1	54.2 / 28.5 / 51.4	78.4 / 34.5 / 54.4
KGN [13] (single)[1]	55.5 / 42.9 / 97.0	78.5 / 63.3 / 99.6	86.9 / 73.2 / 99.9	10.8 / 5.48 / 28.7	30.6 / 18.7 / 51.8	49.6 / 33.8 / 62.4
KGN [13] (multi)[1]	38.6 / 18.5 / 63.7	63.8 / 33.1 / 85.0	78.4 / 46.2 / 91.0	52.6 / 40.7 / 86.5	78.1 / 66.7 / 93.1	88.2 / 78.2 / 94.8
KGNv2 (single)[1]	81.4 / 59.1 / 98.8	92.7 / 70.9 / 99.7	96.0 / 77.4 / 99.8	21.4 / 15.3 / 42.9	41.1 / 32.2 / 58.4	56.7 / 45.9 / 68.7
KGNv2 (multi)[1]	**86.4 / 61.8 / 99.7**	**93.4 / 72.5 / 1.00**	**95.7 / 80.4 / 1.00**	**80.4 / 58.5 / 93.1**	**91.0 / 73.5 / 94.6**	**95.1 / 80.5 / 94.9**

[1] Single and multi in the paratheneses means trained on single-object or multi-object data.
† The evaluated model is trained on Acronym [25] dataset.

TABLE II: Ablation Study Results

Methods	Components		Mult-Object Evaluation
	sBranch[1]	sKpt[2]	*Avg GSR% / GCR% / OSR%
KGN			30.4 / 19.3 / 47.6
KGNv2	✓		38.8 / 30.7 / 53.2
KGNv2	✓	✓	**39.7 / 31.1 / 56.7**

* Numbers averaged over three error tolerance levels.
[1] sBranch - Scaled branch (§.III-C).
[2] sKpt - Scale-normalized keypoints (§III-B).

Fig. 4: Objects used for physical experiments. Yellow bounding box selects the object set for single-object grasping.

B. Implementation Details

The vision encoder used in our method is DLA-34 [45] modified with deformable convolution layer [46], except that the first layer is modified with 4-channel kernels for RGB-D input. A shallow two-layer convolution network is used for each task head. We finetune the network on PS training splits starting from pretrained weights on CoCo dataset [47], whose blue channel parameters are duplicated for the depth channel in the input layer. The network is trained for 400 epochs using the ADAM optimizer, with initial learning rate as 1.25×10^{-4} and is decayed by 10x at epoch 350 and 370. We adopts image augmentation, including random cropping, flipping, and color jittering, for better generalization ability. All the training on done on a single NVIDIA RTX 3090 GPU, and testing on a single NVIDIA RTX 1080Ti. The training takes 16 hours, and the inference speed is 9 FPS.

C. Synthetic Dataset Experiments

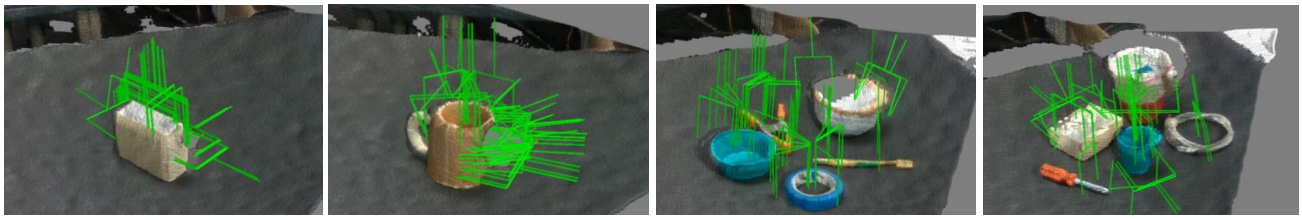
We first test our method on the Primitive Shape dataset test split to exmaine its ability to learn the annotated grasp distribution. The performance is compared against KGN [13] to demonstrate efficacy of proposed modifications. We further conduct ablation study to break down the contribution by each proposed components under domain shift.

Metrics: We evaluate predicted grasp pose set by comparing against the ground truth (GT) set. Following [13], we use three metrics for the evaluation: (1) *Grasp Precision Rate (GPR)*: Percentage of grasp predictions with closeby GT; (2) *Grasp Coverage Rate (GCR)*: Percentage of GT pose with closeby predictions; (3) *Object Success Rate (OSR)*: Percentage of objects targeted by near-GT predictions. The similarity between two poses are determined by thresholding both the translational and rotational errors, which are defined as L_2 norm between translations and the minimum angle required to align rotations. [48], [49], respectively. We collect the results under three different error levels from strict to loose: $(1\text{cm}, 20°)$, $(2\text{cm}, 30°)$, and $(3\text{cm}, 45°)$.

Dataset Evaluation Results: We first evaluate KGNv2 and baseline KGN on both single- and multi-object test sets, while training both methods on either single- or multi-object training sets. The results are tabulated in Tab. I, which includes the performance of Contact-GraspNet trained on clutter scenes from Acronym [25] for reference. We first notice that *KGNv2 outperforms the baseline significantly under all settings.* For example, when trained and tested both on multi-object data, KGNv2 achieves 27.8%, 17.8%, and 6.6% performance improvement under the most sctrict threshold values for GSR, GCR, and OSR, respectively. When trained on snigle-object scenes and generalizing to more complex multi-object scenarios, KGNv2 gets 10.6%, 9.8%, and 14.2% performance gain for the above metrics, which is comparable to Contact-Graspnet considered as an upper bound under this setting [13].

We also observe that our method trained on multi-object data performs even better in single-object benchmark compared to it trained also on single-object scenarios. This suggests that our network learns to reason about physical relationship between the objects, which is beneficial for all grasping tasks including single-object picking. A similar trend is not observed for [13] - trained on multi-object data, its precision in single-object evaluation is 47.8% lower than that of KGNv2 under the most strict error tolerance thresholds in single-object testing, and is also 14% lower than itself in multi-object object testing. The fact that it cannot generalize to even simpler task suggests deficit in its design that prevents it from reasoning with geometric information, which is mitigated by our modifications.

Ablation study: To better understand the benefits of



(a) **(b)** **(c)** **(d)**

Fig. 5: Demonstration of generated grasp candidates in physical experiments. (a)(b) Single-object experiment results. (c)(d) Multi-object experiment results. Only at most 40 grasps are randomly selected for visualization.

TABLE III: Single-Object Grasping Comparison from Published Works

Approach⋆	**Settings**			**Success Rate (%)**
	Modality	Obj	Trial/Obj	
PointNetGPD[20]	PC	10	10	82.0
6DoF-GraspNet[22]	PC	17	3	88.0
L2G[30]	PC	48	5	50.5
RGBMatters[11]	RGB-D	9	20	91.67
MonoGraspNet[12]	RGB	12	15	75.95
KGN	RGB-D	8	5	87.5±9.6
KGNv2	RGB-D	8	5	92.5±6.7

⋆ All results for baselines adopted from original paper for reference, following [3].

TABLE IV: Multi-Object Grasping Comparison from Published Works

Approach⋆	**Settings**			**Success Rate (%)**	**Clear Rate (%)**
	Modality	Sn†	Obj/Sn†		
PnGPD[20]	PC	10	8	77.77	97.5
CGN[29]	PC	9	4-9	90.20	N/A‡
Pn++[9]	PC	20	6	77.19	94.5
RGBMatters[11]	RGB-D	6	5-8	91.1	100
MonoGN[12]	RGB	8	4-5	N/A‡	80.6
KGNv2	RGB-D	10	5	80±10.1	96±7.4

⋆ All results for baselines adopted from original paper for reference, following [3].
† Calculated as the average success number over average attempts number.
‡ Not released by original paper

our introduced upgrades, we conduct ablation study by removing the scale-normalized keypoint and scale prediction branch design one-by-one. To examine the capability of grasp detector under domain shift, the networks are trained on single-object data while tested in multi-object environments. We report the average number of three metrics under all error tolerance levels in Tab. II. The result demonstrates the contribution by both modifications - simple scale branch dramatically improves the performance, while scale-normalized keypoint further enhances the accuracy.

D. Physical Experiments

To validate the sim-to-real generalization ability, we apply the proposed grasp detector in real-world physical experiments. The robotic system is composed of an Intel RealSense D435 camera mounted at a fixed position for perception, and a custom-made 7-DoF robotic arm for execution. The trajectory is planned with MoveIt [50]. The object set used for experiments is depicted in Fig. 4. Both *single-object* and *multi-object* grasping experiments are performed. For all physical experiments, we use the KGNv2 weight trained on Primitive Shape multi-object training set, as it demonstrates superior performance even in single-object vision dataset evaluation. 95% confidence interval are reported.

Grasp selection strategy. A grasp selection strategy is necessary to choose a pose for execution from the rich candidate set generated by KGNv2. Furthermore, the grasp poses in dataset are annotated in a gripper-agnostic way without considering specific gripper geometry. Hence, gripper-specific prior is injected in the selection stage. To select the pose for execution, we first rank the candidate poses based on method fidelity. We calculate a score for each grasp pose $s(g)$ by combining the center confidence generated during the keypoint detection stage with the reprojection error (RE) introduced by the pose recovery stage: $s(g) = Y_{c,m} + RE$. Then, we choose the feasible pose with top confidence that causes no collision and encloses non-empty volume of point cloud based on gripper attributes [51]. For experiment outcome, we collect numbers from related papers as reference.

Single-Object Grasping Results. We conduct experiments on pick-and-place task of individual objects, requiring the robotic arm to successfully retrieve a target that has been randomly placed, and transport it to a predetermined location. In this experiment, we utilize the same set of eight objects with diverse shapes as those employed in [13], shown in Fig. 4. For each object, we conduct 5 trials and calculate the success rate.

The results are collected in Table. III. Following [3], [38], we also collect the results from published grasping research efforts to place the performance of KGNv2 within greater context. KGNv2 demonstrates a satisfacory success rate in spite of trained on basic, synthetic primitive shapes, indicating that geometric information is learnt. Furthermore, it achieves a 5% performance gain compared to KGN, suggesting the proposed modifications lead to more accurate grasp pose prediction. Typical cause of failure is the prediction of unstable grasps for target objects, involving off-center grasp pose for the ball causing it to roll off, or targeting metallic, slippery section on the clamp.

Multi-Object Grasping Results In addition, we conduct experiments in scenarios with multiple objects, where five objects are randomly selected and placed on the table for

each scene. We iteratively select grasp poses generated by the model for execution. The termination criteria for each trial consists of two conditions, namely: (1) succesful removal of all objects; (2) three consecutive failed attempts, with the purpose of preventing the system from becoming stuck in a consistent failure mode.

We evaluate the succcess rate for grasp attempts and clearance rate for the objects. The experiment results are tabulated in Tab. IV, which collects results from related papers as before for reference. Our approach obtains a comparable success rate and clearance rate as state-of-the-art methods, which further validates our method in real-world tasks. For failure cases, the cause for single-object picking failure still exist. In addition, we observe some failure where the grasps aim for occluded area, probably due to unreasonable extrapolation by the network.

V. Conclusion

In this work, we present a 6-DoF grasp pose detection method from RGB-D image input. Our method first generates pose up to a scale based on image-space keypoint detection and P*n*P algorithm. Then it regresses pose scale as well as open width. Based on numerical analysis on PnP algorithm, we further propose the scale-normalized keypoints design to improve the pose estimation accuracy. On the Primitive Shape dataset, we verify that our method learns to generate grasp distribution from labels better, and demonstrate the efficacy of designed components via ablation study. Physical experiments are conducted to further validate our approach's generalization ability over sim-to-real gap.

While the physical experiments show that KGNv2 successfully learns geometric reasoning skills that is generalizable to a set of common household objects from simple primitive geometric data, the uniform color of the primitive shapes may limit the model's capacity to recognize diverse visual appearances in the open world. Future efforts could explore augmenting the dataset with authentic textures leveraging generative methods such as diffusion model [52], [53].

References

[1] 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," *arXiv preprint arXiv:1703.09312*, 2017.

[2] F.-J. Chu, R. Xu, and P. A. Vela, "Real-world multiobject, multigrasp detection," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3355–3362, 2018.

[3] R. Xu, F.-J. Chu, and P. A. Vela, "GKNet: Grasp keypoint network for grasp candidates detection," *The International Journal of Robotics Research*, vol. 41, no. 4, pp. 361–389, 2022.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[5] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," in *arXiv preprint arXiv:1904.07850*, 2019.

[6] X. Lou, Y. Yang, and C. Choi, "Collision-aware target-driven object grasping in constrained environments," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 6364–6370.

[7] C. C. B. Viturino, D. M. de Oliveira, A. G. S. Conceição, and U. Junior, "6D robotic grasping system using convolutional neural networks and adaptive artificial potential fields with orientation control," in *2021 Latin American Robotics Symposium (LARS), 2021 Brazilian Symposium on Robotics (SBR), and 2021 Workshop on Robotics in Education (WRE)*. IEEE, 2021, pp. 144–149.

[8] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 652–660.

[9] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[10] H. Ma and D. Huang, "Towards scale balanced 6-dof grasp detection in cluttered scenes," *arXiv preprint arXiv:2212.05275*, 2022.

[11] 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 *IEEE International Conference on Robotics and Automation*, 2021, pp. 13 459–13 466.

[12] G. Zhai, D. Huang, S.-C. Wu, H. Jung, Y. Di, F. Manhardt, F. Tombari, N. Navab, and B. Busam, "Monograspnet: 6-dof grasping with a single rgb image," *arXiv preprint arXiv:2209.13036*, 2022.

[13] Y. Chen, Y. Lin, and P. Vela, "Keypoint-graspnet: Keypoint-based 6-dof grasp generation from the monocular rgb-d input," *IEEE International Conference on Robotics and Automation*, 2023.

[14] T. Collins and A. Bartoli, "Infinitesimal plane-based pose estimation," *International Journal of Computer Vision*, vol. 109, no. 3, pp. 252–286, 2014.

[15] K. Kleeberger, R. Bormann, W. Kraus, and M. F. Huber, "A survey on learning-based robotic grasping," *Current Robotics Reports*, vol. 1, pp. 239–249, 2020.

[16] R. Newbury, M. Gu, L. Chumbley, A. Mousavian, C. Eppner, J. Leitner, J. Bohg, A. Morales, T. Asfour, D. Kragic, *et al.*, "Deep learning approaches to grasp synthesis: A review," *arXiv preprint arXiv:2207.02556*, 2022.

[17] A. Alliegro, D. Valsesia, G. Fracastoro, E. Magli, and T. Tommasi, "Denoise and contrast for category agnostic shape completion," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4629–4638.

[18] P. Ni, W. Zhang, X. Zhu, and Q. Cao, "PointNet++ grasping: learning an end-to-end spatial grasp generation algorithm from sparse point clouds," in *IEEE International Conference on Robotics and Automation*, 2020, pp. 3619–3625.

[19] X. Yan, J. Hsu, M. Khansari, Y. Bai, A. Pathak, A. Gupta, J. Davidson, and H. Lee, "Learning 6-Dof grasping interaction via deep geometry-aware 3D representations," in *IEEE International Conference on Robotics and Automation*, 2018, pp. 3766–3773.

[20] H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang, "PointNetGPD: Detecting grasp configurations from point sets," in *IEEE International Conference on Robotics and Automation*, 2019, pp. 3629–3635.

[21] A. Ten Pas and R. Platt, "Using geometry to detect grasp poses in 3D point clouds," in *Robotics Research*. Springer, 2018, pp. 307–324.

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

[23] Y. Zhou and K. Hauser, "6Dof grasp planning by optimizing a deep learning scoring function," in *Robotics: Science and Systems Workshop on Revisiting Contact-turning a Problem into a Solution*, vol. 2, 2017, p. 6.

[24] Q. Lu, K. Chenna, B. Sundaralingam, and T. Hermans, "Planning multi-fingered grasps as probabilistic inference in a learned deep network," in *Robotics Research*. Springer, 2020, pp. 455–472.

[25] C. Eppner, A. Mousavian, and D. Fox, "Acronym: A large-scale grasp dataset based on simulation," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 6222–6227.

[26] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: A large-scale benchmark for general object grasping," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 444–11 453.

[27] 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," in *Conference on robot learning*. PMLR, 2020, pp. 53–65.

[28] K.-Y. Jeng, Y.-C. Liu, Z. Y. Liu, J.-W. Wang, Y.-L. Chang, H.-T. Su, and W. Hsu, "GDN: A coarse-to-fine (c2f) representation for end-to-

end 6-Dof grasp detection," in *Conference on Robot Learning*. PMLR, 2020, pp. 220–231.

[29] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-GraspNet: Efficient 6-Dof grasp generation in cluttered scenes," *IEEE International Conference on Robotics and Automation*, 2021.

[30] A. Alliegro, M. Rudorfer, F. Frattin, A. Leonardis, and T. Tommasi, "End-to-end learning to grasp from object point clouds," *arXiv preprint arXiv:2203.05585*, 2022.

[31] X. Zhu, L. Sun, Y. Fan, and M. Tomizuka, "6-dof contrastive grasp proposal network," 2021, pp. 6371–6377.

[32] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6569–6578.

[33] Y. Chen, R. Xu, Y. Lin, and P. A. Vela, "A joint network for grasp detection conditioned on natural language commands," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4576–4582.

[34] Z. Geng, K. Sun, B. Xiao, Z. Zhang, and J. Wang, "Bottom-up human pose estimation via disentangled keypoint regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 676–14 686.

[35] D. Maji, S. Nagori, M. Mathew, and D. Poddar, "Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2637–2646.

[36] Y. Lin, J. Tremblay, S. Tyree, P. A. Vela, and S. Birchfield, "Single-stage keypoint-based category-level object pose estimation from an RGB image," in *IEEE International Conference on Robotics and Automation*, 2022.

[37] C. Sweeney, G. Izatt, and R. Tedrake, "A supervised approach to predicting noise in depth images," in *IEEE International Conference on Robotics and Automation*, 2019, pp. 796–802.

[38] Y. Lin, C. Tang, F.-J. Chu, R. Xu, and P. A. Vela, "Primitive shape recognition for object grasping," *arXiv preprint arXiv:2201.00956*, 2022.

[39] J. Aleotti and S. Caselli, "A 3d shape segmentation approach for robot grasping by parts," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 358–366, 2012.

[40] C. Goldfeder, P. K. Allen, C. Lackner, and R. Pelossof, "Grasp planning via decomposition trees," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 4679–4684.

[41] K. Huebner and D. Kragic, "Selection of robot pre-grasps using box-based shape approximation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 1765–1770.

[42] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 3511–3516.

[43] C. Wu, J. Chen, Q. Cao, J. Zhang, Y. Tai, L. Sun, and K. Jia, "Grasp proposal networks: An end-to-end solution for visual learning of robotic grasps," *Advances in Neural Information Processing Systems*, vol. 33, pp. 13 174–13 184, 2020.

[44] 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," in *The International Symposium of Robotics Research*. Springer, 2019, pp. 890–905.

[45] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2403–2412.

[46] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 764–773.

[47] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[48] R. Hartley, J. Trumpf, Y. Dai, and H. Li, "Rotation averaging," *International Journal of Computer Vision*, vol. 103, no. 3, pp. 267–305, 2013.

[49] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, *et al.*, "Benchmarking 6Dof outdoor visual localization in changing conditions," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8601–8610.

[50] M. Görner, R. Haschke, H. Ritter, and J. Zhang, "Moveit! task constructor for task-level motion planning," in *IEEE International Conference on Robotics and Automation*, 2019, pp. 190–196.

[51] 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.

[52] L. Zhang and M. Agrawala, "Adding conditional control to text-to-image diffusion models," 2023.

[53] E. Richardson, G. Metzer, Y. Alaluf, R. Giryes, and D. Cohen-Or, "Texture: Text-guided texturing of 3d shapes," *arXiv preprint arXiv:2302.01721*, 2023.