2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 1-5, 2023. Detroit, USA

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 an improved keypoint approach for 6-DoF grasp pose synthesis from RGB-D input. Keypoint-based grasp detection from image input demonstrated promising results in a previous study, where the visual information provided by color imagery compensates for noisy or imprecise depth measurements. However, it relies heavily on accurate keypoint prediction in image space. We devise a new grasp generation network that reduces the dependency on precise keypoint estimation. Given an RGB-D input, the network estimates both the grasp pose and the camera-grasp length scale. Re-design of the keypoint output space mitigates the impact of keypoint prediction noise on Perspective-n-Point (P*n*P) algorithm solutions. Experiments show that the proposed method outperforms the baseline by a large margin, validating its design. Though trained only on simple synthetic objects, our method demonstrates sim-to-real capacity through competitive results in real-world robot experiments.**

I. Introduction

Robotic grasping is a fundamental and challenging problem, requiring both object perception as well as geometric reasoning from visual input. Past reasearch simplified the problem by constraining the grasp poses to SE(2) space, assuming that the camera has a (nearly) top-down view of the scene, and the gripper approaches perpendicular to the support plane [1]–[3]. The restriction permits 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 impedes $SE(2)$ grasp recognition utility in constrained environments [6], [7].

The limitation of planar grasps has motivated exploration of 6-DoF grasp synthesis, which outputs grasp poses in SE(3). Point cloud methods, utilizing point set feature extractors like PointNets [8], [9], have achieved 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 at the cost of increased computation due to the need for an additional instance segmentation network. The sensitivity to input uncertainty or imprecision remains a concern.

Consequently, the use of 2D/2.5D input for 6-DoF grasp detection has gained attention due to the additional visual information offered by color images. Visual clues provided from a color image, as extracted by modern convolutional neural networks (CNNs), not only facilitate the discernment of small objects imperceptible to depth sensors, but also improve robustness to depth sensor measurement noise [11]. Despite demonstrating promising results, existing methods [11], [12] still utilize direct regression of 3D grasp pose representations and force the network to estimate 3D structural information from 2D projective input, which is coupled to camera geometry. Such regression requires more extensive annotation information, such as surface normals, for training [12]; or dense discretization of SO(3) space [11].

To avoid directly estimating 3D pose parameters, Keypoint-GraspNet (KGN) [13] isolates the 2D-to-3D recovery stage from 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 a 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 predictions cause unstable estimation of the *scale factor* (here, the magnitude of the translation of the grasp pose relative to the camera), especially in novel test domains, such as when training on synthetic data and testing on real-world data. KGN heuristically addresses the issue by adopting the perceived depth as the scale, whose accuracy is affected by depth measurement error and occlusion.

This paper introduce KGNv2, an improved keypoint-based grasp detection network with more accurate/less sensitive grasp pose estimation. The network eliminates the need for accurate keypoint proximity estimation by predicting pose and scale separately, which improves the accuracy of the generated poses. The keypoint output space is re-designed to be (length scale) normalized, which reduces sensitivity to keypoint error and enhances the precisioin of the estimated pose. The simple modifications improve grasp prediction performance across all tests applied using the primitive shape dataset from [13]. The network generalizes to actual objects with shape variation in real-world experiments, indicating the potential of training grasp detectors on virtual data with primitive geometries, for which obtaining ground-truth labels is easier and faster.

[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

*This work was supported in part by NSF Award #2026611.

2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-9190-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/IROS55552.2023.10342514

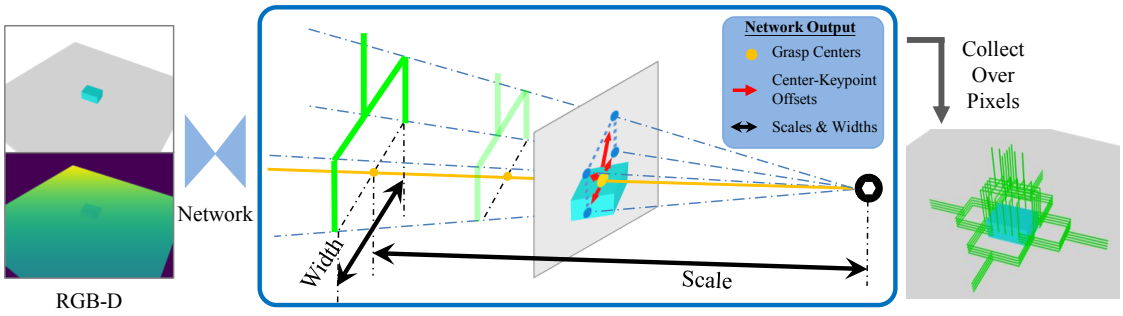


Fig. 1: **Overview of KGNv2.** Given an RGB-D input, our model predicts pixel-wise candidate grasp poses. It estimates the *pose up to a scale* by applying P*n*P algorithm on generated image-space keypoint coordinates with camera intrinsic matrix. The keypoints are obtained by predicted grasp centers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

II. Related Work

The literature review scoping is narrowed to learning-based 6-DoF grasp detection with antipodal end-effector. Other related areas include dexterous 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

The emergence of point set encoders like PointNets [8], [9] and DeCo [17] led to a shift of emphasis in 6-DoF grasp pose detection to point cloud inputs [18]. Early efforts employed 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, whose trained network maps points in candidate grasp regions to grasp pose scores. The variable quality and insufficient density of the sampled grasp poses limits recognition performance. 6DoF-GraspNet [22] replaces the sampling-based candidate proposal approach with deep generator trained with Generative Adversial Network (GAN) or Variational Autoencoder (VAE) objectives. Other approaches [23], [24] investigate refining the initial pose proposal by increasing the score estimated from a learnt evaluator.

The above point cloud pipelines are time-consuming due the use of multiple forward passes. 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 detections per point, GDN [28] extends the idea with a coarse-to-fine representation idea, first performing classificaiton on a set of discrete angular grids, then regressing translation and rotation refinement values for high-confidence candidates. Another thread of inquiry argues in favor of explicit contact physics reasoning. The adopted representation is two contact points plus pitching angle [29], [30], with the assumption that one of the contact points references a visible point of the object's partial point cloud.

Point cloud methods share common drawbacks, as studied in recent literature. Due to the high processing time to extract geometric information from the point coordinates enumeration, truncating the point cloud volume is necessary; usually by downsampling [27] or target segmentation [22]. L2G [30] alternatively designs an learnable sampler, which can be jointly tuned in end-to-end training. It assumes that a properly designed sampling procedure will retain critical information for grasp synthesis, which is not always the case. Especially for high-resolution input. Another limitation of point cloud methods is their bias towards larger objects due to having higher point counts. Bias reduction was achieved through balanced sampling based on instance segmentation masks [10] at the additional computational cost of an external segmentation module.

B. Image-based Methods

Relative to point clouds, images are faster to process with modern networks and preserve pixel proximity relationships, which can address the above issues. Hence, recent investigations into 6-DoF grasp detection from image input [31], where the color modality is shown to improve robustness to depth uncertainty [11]. However, the intial exploration still relies on 3D representaion of grasp poses, such as a contact-point-based description [12], which fails to leverage existing knowledge about 3D-to-2D camera projection.

In contract, KGN [13] describes a 3D grasp pose by 4 image space keypoints, and exploits the efficient keypoints synthesis of keypoint detectors like CenterNet [5], [32]. Designing the 4 keypoints to model the projection of virtual 3D points in the gripper frame with predefined coordinates, the application of a P*n*P algorithm recovers their original 3D structure by leveraging camera projective geometry. Given fixed 3D coordinates, the *relative location* and *absolute distance* between 2D keypoints determines the *pose up to scale* and *scale factor*, respectively (e.g. closer keypoints means the gripper frame is further to the camera plane). However, the prediction of keypoint to camera length scale was found to be unstable in novel test environments

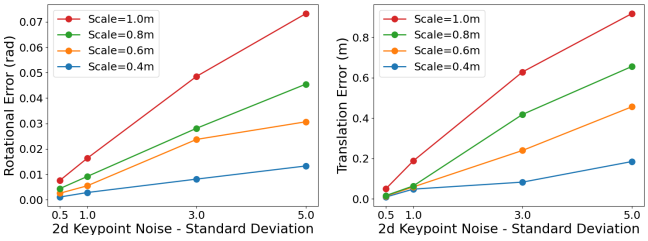


Fig. 2: Synthetic study on the relationship between pose scale and pose recovery error from P*n*P algorithm [14] due to noise. With larger grasp pose scale (grasp is further from the camera), both rotational and translational error increases under all noise levels. The observation motivates us to predict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

[13]. This work relaxes the requirement of precise distance prediction by separately predicting the scale. The predicted scale permits further redesign of the keypoint output space based on the influence keypoint prediction noise has on the scale factor. We show that our modified network, named KGNv2, achieves superior performance relative to KGN in the single- and multi-object settings.

III. Methodology

A. Problem Definition

Given a monocular RGB-D input, the objective is to synthesize a set of 6-DoF grasps with grasp pose $g \in \mathrm{SE}(3)$ and associated open width w to pick up objects perceived by the image *without* converting the input into 3D point representations. The problem is challenging since the input is in 2D image space while the output is in 3D space. The proposed solution, as illustrated in Fig. 1, predicts grasp poses from separately estimated keypoints, pose scale, and grasp open width outputs.

B. Pose Estimation with Scale-Normalized Keypoint

Inspired by KGN [13], we adopt a keypoint-based strategy that leverages camera 3D-to-2D projective structure to estimate grasp poses up to 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 orientation classes enable simultaneous detection of multiple grasps sharing the same center. It is useful for generating diverse candidate sets of rotationally symmetric objects (e.g. grasps for a ball or cylinder). It also functions as a non-maximum suppression mechanism by considering grasps with overlapping centers and similar orientations to be highly similar, resulting in only one grasp being retained. Grasp centers are detected from a heatmap $Y \in [0,1]^{W' \times H' \times M}$, where W' and H' represent the resolution of the downscaled feature map.

From grasp centers, keypoints' locations $\{(p_{i1}^m, p_{i2}^m, p_{i3}^m, p_{i4}^m)\}$ are predicted based on offset estimates. Our network learns to generate center-keypoint offset vector maps O, which encode the displacement from the center keypoint to the grasp keypoints for each center and orientation candidate. Keypoints locations, obtained from the centers and offsets, input to the IPPE [14] algorithm specifically designed for the coplaner P*n*P problem produce 6-DoF grasp poses. The final synthesized grasp set is the collection of results from high confidence grasp centers.

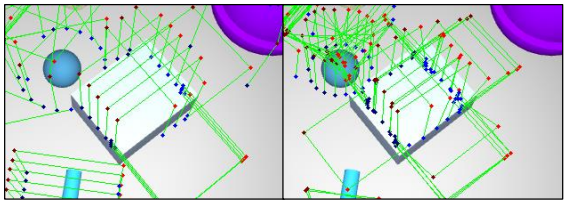
Scale-Normalized Keypoint Prediction. A natural choice of keypoint design is to define the ground-truth keypoints so that they form a square whose sides have a length equal to the grasp open width. In this way, 2D keypoints on the image conform to the diameter across the grasped subregion. However, such a design **naturally favors grasp poses with smaller *scale* (in close proximity to camera) during training**, under distance-based loss functions. Unlike human or object pose estimation [5], grasp poses are larger in quantity per image due to the continuity of feasible grasps for each object and exhibit a distribution across scales. As an object recedes from the camera, its fixed width grasps generate keypoints that project closer to each other due to perspective projection. However, one property of P*n*P algorithm is that *recovered grasp pose for closer keypoints are prone to larger error under the same noise level, based on PnP problem conditioning.* Hence, similar error proportions in keypoint image space translates to larger error in pose space for more distant grasps.

To empirically demonstrate this known property, we conduct a synthetic experiment examining the relationship between scale and pose estimation error. Orientation is randomly sampled for grasps at the origin with the camera optical axis pointed at the origin and sampled at variable distances from it. The grasp pose is then estimated from the keypoint projections injected by Gaussian noise. The average rotational and translational error, defined in §IV-C, is computed from the grasp pose estimate and the known ground truth. When organized by noise and scale, then plotted as in Fig. 2, the error trend shows an increase as a function of scale conditioned on noise level.

Consequently, we propose to predict image-space keypoints normalized by scale. The idea is related to human/object keypoints prediction with area-normalization [33], [34] or object-size-normalization [35]. Here, grasp pose normalization will be based on the relative grasp-camera distance to achieve scale invariance. Specifically, we scale the offset value related to keypoint proximity. For each grasp center c and associated *actual* offset vectors $\tilde{O}_c$ and scale S_c, the network is tasked to predict:

$$O_c = \tilde{O}_c / S_c \tag{1}$$

where the predicted scale (see §III-C) is used in the inference. The scale-normalized keypoint design reduces pose estimation sensitivity to noise for *more distant* grasp poses. If the scaled offset predictions exhibit zero-mean Gaussian noise $\varepsilon \sim \mathcal{N}(0,\sigma^2)$, then the effective perturbation to the offsets reduces to $(1/S_c)\varepsilon \sim \mathcal{N}(0,\sigma^2/S_c^2)$, whose variance decreases by S_c, leading to more accurate pose recovery. Section IV-C shows that this design does improve accuracy.



Ground Truth **Prediction**

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.

C. Scale Prediction

Another reason to normalize by scale has to do with keypoint prediction errors induced by domain shift. For example, when testing on multi-object scenes from the Primitive Shape dataset [13], a keypoint detector trained with single-object data tends to produce keypoints that are more closely grouped than the ground truth; see Fig. 3 . 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 impacting the translation scale or the gripper width scale.

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 will generically not be a visible 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 [36]. As a result, raw depth values are less informative for accurate scale estimation for the grasp pose.

Rather than predicating 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 class. Each grasp pose directly matches to the scale prediction at the corresponding grasp center location and orientation class. Although similar to depth prediction problem, scale estimation given only RGB input is essentially ill-posed across object-grasp space. We assume the noisy depth map input serves as a signal that reduces scale 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 class m, the rotation and translation given by the P*n*P algorithm is: $\tilde{g} = \{\tilde{R}, \tilde{T}\}$, then the final pose combined with the scale prediction is:

$$\hat{g} = \{\hat{R}, \hat{T}\}) = \{\tilde{R}, \frac{S_{c,m}}{\|\tilde{T}\|}\tilde{T}\}) \tag{2}$$

D. Final Loss

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

$$L = \gamma_Y L_Y + \gamma_O L_O + \gamma_W L_W + \gamma_S L_S \tag{3}$$

We chose: $\gamma_Y = 1$, $\gamma_O = 1$, $\gamma_W = 10$, $\gamma_S = 10$ to balance the relative dynamic ranges of the loss components.

IV. Experiments

A. Synthetic Dataset

Following [13], we use the Primitive Shape (PS) dataset for network training. It 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 simple shape categories are: *Cylinder*, *Ring*, *Stick*, *Sphere*, *Semi-sphere*, and *Cuboid*. Ground truth grasps are annotated by sampling evenly distributed instances from *grasp families* [37] — the closed-form grasp pose distributions parameterized by the shape type and sizes. The parameter ranges are 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 and driving force in grasp synthesis research for years [37]–[40]. The grasp label generation approach is significanly more cost-effective compared to sample-then-verify strategies based on simulation [25], [41], [42]. Furthermore, it mitigates potential bias or inaccuracies in the labeling process that can arise from sampling artifacts [43], thereby avoiding the subsequent negative impact on training and/or evaluation.

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. The grasp poses causing collisions are removed. We increase the annotation sample density for test splits to verify the extrapolation ability of the trained grasp detector from sparse examples.

B. Implementation Details

The vision encoder used in our method is DLA-34 [44] modified with deformable convolution layer [45], and with a modified first layer consisting of 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 [46], whose blue channel parameters are duplicated for the depth

TABLE I: Vision Dataset Evaluation

Methods	**Single-Object Evaluation (GPR% / GCR% / OSR%)**			**Multi-Object Evaluation (GPR% / 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 GPR% / 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.

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. Training is 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

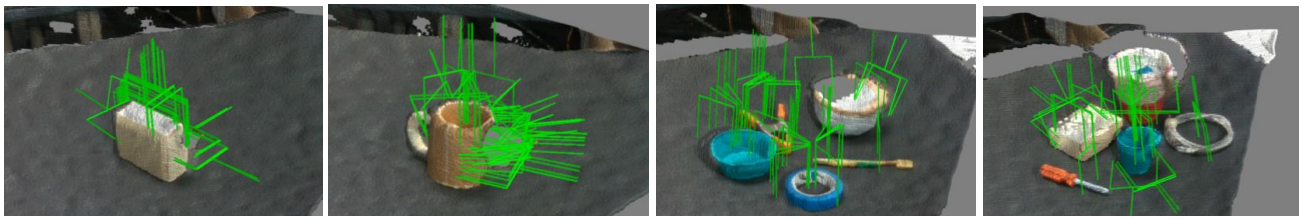
We first test our method on the Primitive Shape dataset test split to examine its ability to learn the annotated grasp distribution. The performance is compared against KGN [13] to demonstrate the effectiveness of the proposed modifications. An ablation study break down the contribution of each of the proposed components to the overall performance.

Metrics: Evaluation compares the predicted grasp pose set to the ground truth (GT) set. Following [13], the three evaluation metrics are: (1) *Grasp Precision Rate (GPR)*: Percentage of grasp predictions with a nearby GT grasp; (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, defined as L_2 norm between translations and the minimum angle required to align rotations; see [47] and [48], respectively. Evaluation occurs for three different tolerances 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 KGN 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 strictest threshold values for GPR, GCR, and OSR, respectively. When trained on single-object scenes and tested on more complex multi-object scenarios, KGNv2 shows 10.6%, 9.8%, and 14.2% performance gains, comparable to Contact-Graspnet, which is considered to be an upper bound in this setting [13].

We also observe that our method trained on multi-object data performs better in the single-object benchmark versus when trained on single-object scenarios. This suggests that the KGNv2 network learns to reason about critical scene structure regarding grasp-object geometry, which benefits all grasping tasks including single-object picking. A similar trend is not observed for [13] - trained on multi-object data, its GPR for single-object evaluation is 47.8% lower than that of KGNv2 under the most strict error tolerance thresholds in single-object testing, while being only 14% lower than itself in multi-object object testing. The fact that KGN cannot generalize to a simpler task suggests a deficit in its design that prevents reasoning about geometric information, which is mitigated by the modifications described here.

Ablation study: To better understand the benefits of the modifications, an ablation study removes the scale-normalized keypoint and scale prediction branch design one-by-one. To test the capability of grasp detection under domain shift, the networks are trained on single-object data then tested on multi-object data. Tab. II reports the average of the three metrics across all error tolerance levels. The results demonstrate the impact of both modifications - the simple scale branch improves the performance of grasp prediction, and the scale-normalized keypoint further enhances performance.



(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

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 [49]. 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 intervals 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. The annotated grasp poses are generated in as gripper-agnostic a manner as possible. Gripper-specific context is applied in the grasp selection stage. To select the pose for grasp execution, we first rank the candidate poses based on model confidence (as output by KGNv2). 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 a non-empty volume of the grasp region point cloud based on gripper attributes [50]. For comparative purposes, we collect reported results from related papers as references.

Single-Object Grasping Results. We conduct pick-and-place experiments for individual objects that require the robotic arm to retrieve a randomly placed target and move it to a predetermined location. In this experiment, we utilize the same set of eight objects with diverse shapes as 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], [37], we also collect the results from published grasping research efforts to place the performance of KGNv2 within a greater context. KGNv2 demonstrates a top-performing success rate in spite of being trained on basic, synthetic primitive shapes, indicating that the critical 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. Observed failure modes involve the prediction of unstable grasps such as choosing off-center grasp poses for the ball causing it to roll away, or targeting the metallic, slippery section of the clamp.

Multi-Object Grasping Results Experiments conducted with multiple objects involve randomly selecting five objects from a set of objects to place on the table for each scene. We iteratively select grasp poses generated by the model for pick-and-place 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 penalizing the system when stuck in a persistent failure situation.

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 to state-of-the-art methods, which further validates our method in real-world tasks. For failure cases, the single-object picking failure causes still exist. Additionally, we observed some failures where grasps aim for occluded regions, probably due to unreasonable extrapolation by the network.

V. Conclusion

This work describes a 6-DoF grasp pose detection method from RGB-D image input. The method first generates pose up to translational scale based on image-space keypoint detection and the P*n*P algorithm, It also regresses pose scale as well as open width. Based on numerical analysis on PnP algorithm, a scale-normalized keypoint design improves pose estimation accuracy and reduces sensitivity to keypoint pixel error. On the Primitive Shape dataset, we verify that our method learns to generate grasp distribution from the designed labels better than a previous approach, and demonstrate the impact of the modifications via ablation study. Physical experiments are conducted to further validate our approach's generalization ability in relation to possible sim-to-real gaps.

While the physical experiments show that KGNv2 successfully learns geometric reasoning skills that generalize 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 [51], [52].

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] 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.

[34] 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.

[35] 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.

[36] 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.

[37] 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.

[38] 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.

[39] 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.

[40] 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.

[41] 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.

[42] 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.

[43] 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.

[44] 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.

[45] 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.

[46] 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.

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

[48] 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.

[49] 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.

[50] 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.

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

[52] 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.