Efficient Heatmap-Guided 6-DoF Grasp Detection In Cluttered Scenes

Authors: Siang Chen, Wei Tang, Pengwei Xie, Wenming Yang, Guijin Wang

Jakub Nowacki & Kamil Pilkiewicz

January 23, 2025

Importance of Robotic Grasping

- Robotic grasping is crucial in manufacturing, service, and medical applications.
- Challenges include:
  - Handling cluttered environments.
  - Achieving fast and accurate grasp detection.
  - Adapting to unseen and diverse objects.

Current Limitations in Grasp Detection

- Traditional methods use entire point clouds and lack efficiency.
- Limited real-time performance and precision for 6-DoF grasping.

Recent Advances in Deep Learning

- Recent advances in deep learning have enabled data-driven methods to generalize to unseen objects.
- Representative methods generate grasp configurations as oriented grasp rectangles by adopting pixel-wise heatmaps to represent planar grasps.
- These methods achieve good performance in simple scenarios with high efficiency.
- **Limitation**: Forces gripper perpendicular to the camera plane, restricting applications.

Advances in 6-DoF Grasping

- 6-DoF grasping allows robots to grasp from arbitrary directions.
- Early methods use sampling-evaluation strategies but are time-consuming.
- Direct regression of grasp attributes improves efficiency but lacks reliability due to missing local geometric context.
- Recent methods leverage locally aggregated features for better grasp poses, but real-time performance remains challenging.

Heatmaps for 6-DoF Grasp Detection

- Heatmaps have shown success in object detection, human pose estimation, and planar grasping.
- This work extends heatmaps to high-quality 6-DoF grasp detection with high efficiency.
- **Key Insight**: Grasp heatmaps guide aggregation of local points into graspable regions, reducing input size and enabling precise grasp pose generation.

Heatmaps for 6-DoF Grasp Detection

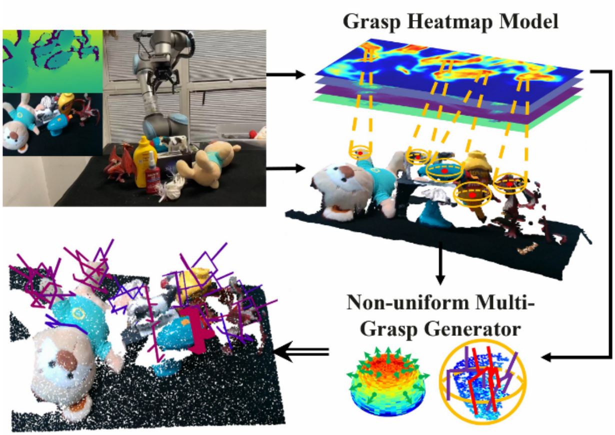

Fig. 1. The key insight of our method is generating the grasp heatmaps as guidance for regional geometric feature mining and further grasp pose generation via a novel local grasp generator.

Key Contributions

- **Framework**: A global-to-local semantic-to-point 6-DoF grasp detection system achieving real-time, state-of-the-art performance with low-cost training.
- **Efficiency**: Gaussian encoding and grid-based strategy improve heatmap prediction efficiency and reduce input size.
- **Innovation**: A local grasp generator with non-uniform anchor sampling ensures precise, dense grasps, while semantic-to-point feature fusion enhances robustness.

Problem Statement

- **Input**: A monocular RGBD image $\chi \in \mathbb{R}^{H \times W \times 4}$ and camera intrinsics c.
- **Objective**: Efficiently learn parallel-jaw grasp configurations G in cluttered scenes.
- **Representation**: Grasp pose defined as $(u, v, \theta, w, d, \gamma, \beta)$, where:
  - (u, v): Grasp center in the image plane.
  - θ, w, d: Grasp orientation, width, and depth.
  - γ, β: Grasp angles for precise 6-DoF positioning.

Problem Statement

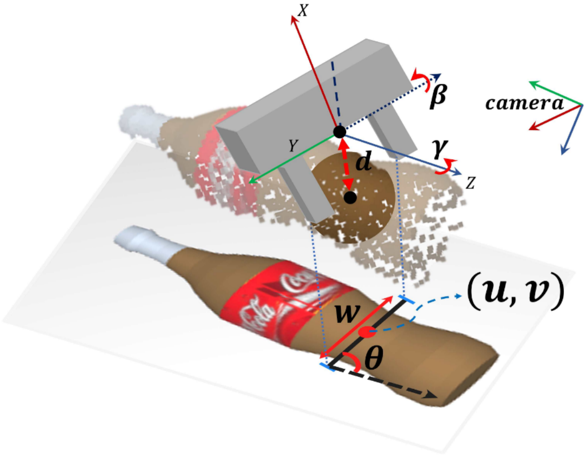

Fig. 2. Proposed grasp representation as $(u, v, \theta, w, d, \gamma, \beta)$.

HGGD Framework Overview

- **Goal**: Efficiently generate high-quality, diverse grasps from monocular RGBD images.
- **Approach**: Using grasp heatmaps as region guidance.
- **Key Modules**:
  - **Grasp Heatmap Model (GHM)**:
    - Preprocesses RGBD images with CNN.
    - Generates robust grasp heatmaps using Gaussian encoding and a grid-based strategy.
  - **Non-uniform Multi-Grasp Generator (NMG)**:
    - Uses heatmaps to focus on graspable regions.
    - Employs non-uniform anchor sampling for higher grasp quality.
    - Incorporates semantic-to-point feature fusion for robust detection.

HGGD Framework Overview

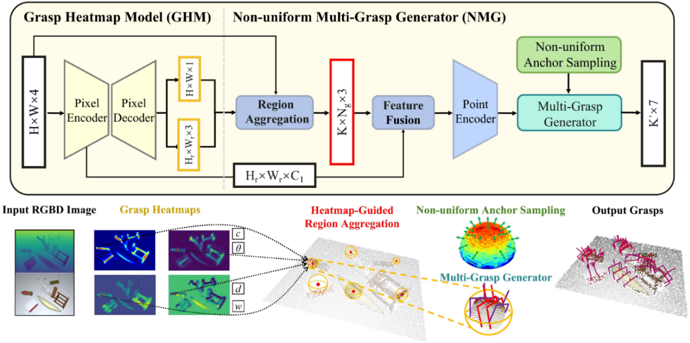

Fig. 3. The architecture of HGGD. Taking a monocular RGBD image as input, GHM generates grasp confidence heatmap Q_c and grided attributes heatmaps (Q_θ, Q_w, Q_d). Then NMG transfers the depth image to the point cloud through camera intrinsics **c** for region aggregation under the guidance of heatmaps. Feature fusion and the point encoder extract regional features fused with semantic information from GHM. Finally, a multi-grasp generator combined with a novel non-uniform anchor sampling mechanism utilizes the fusion features to output the grasps.

Grasp Heatmap Model (GHM)

- **Model Type**: Encoder-decoder architecture with two output branches:
  - **Confidence Branch**: Constructs the grasp confidence heatmap (Q_c).
  - **Attribute Branch**: Generates attribute heatmaps (Q_θ, Q_w, Q_d).
- Uses Gaussian encoding and grid-based strategy to decouple the task for different heatmap characteristics.
- Ground truth 6-DoF grasps are projected onto the image plane and encoded as heatmaps ($\hat{Q}_c$, $\hat{Q}_\theta$, $\hat{Q}_w$, $\hat{Q}_d$).

Grasp Heatmap Model (GHM)

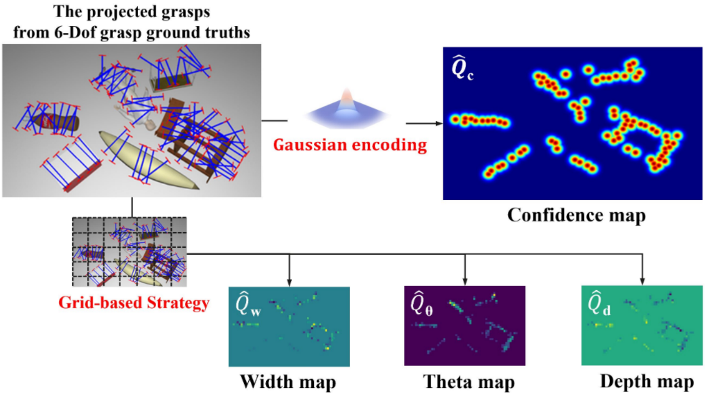

Fig. 4. Visualization of how the ground truth 6-Dof grasps are projected. Grasp confidence heatmap $\hat{Q}_c$ and attribute heatmaps $(\hat{Q}_\theta, \hat{Q}_w, \hat{Q}_d)$ are encoded with Gaussian kernel and grids, respectively.

Gaussian Encoding Strategy

- **Objective**: Encode projected grasp ground truth centers using a 2D Gaussian kernel.
- **Formula**:

$$q = \exp\left(-\frac{(u - u_0)^2 + (v - v_0)^2}{2\sigma_g^2}\right)$$

  - (u_0, v_0): Center point of a grasp ground truth.
  - σ_g: Standard deviation depending on the grasp width.
- **Confidence Prediction**: Supervised by $\hat{Q}_c$, the confidence branch applies pixel-wise classification to predict Q_c.

Intuition behind Gaussian Encoding

- Ground truths are typically sparse.
- The model can struggle to generalize.
- Minor variations in position may significantly impact grasp success
- Effectively highlights grasp centers while considering nearby pixels as additional guidance.

Grid-Based Strategy for Attribute Prediction

- **Objective**: Encode and predict grasp attributes (θ, w, d) within local grids instead of direct pixel-wise regression.
- **Methodology**:
  - The image is divided into $H_r \times W_r$ grid cells with side length r.
  - Multiple oriented anchors (k_a) are introduced per grid cell with uniformly sampled angles.
  - Ground truth θ is assigned to the nearest anchor.
  - Anchor distributions are calculated, and a sigmoid function is applied to predict $\hat{Q}_\theta$.
  - Average normalized w and d values in each grid are used to generate $\hat{Q}_w$ and $\hat{Q}_d$.
- **Attribute Prediction**:
  - θ: Predicted using anchor classification and offset regression.
  - w, d: Predicted via direct regression.

Grid-Based Strategy Advantages

- Exploits the geometric similarity of adjacent grasps for more robust predictions.
- Improves attribute prediction efficiency and accuracy.

Attribute Prediction Process

- **Prediction of θ (Orientation)**:
  - Anchor classification is performed to predict the closest orientation anchor.
  - Offset regression refines the anchor-based orientation to improve precision.
- **Prediction of w (Width) and d (Depth)**:
  - Predicted using direct regression based on the local grid's geometric attributes.
- **Outcome**:
  - Robust and precise grasp attribute estimation through anchor-based classification and regression.

Limitations of Previous Methods

- **Previous Approaches**:
  - Encoded grasps as pixel-wise rectangles.
- **Defects in Previous Methods**:
  - Failed to highlight the grasping probability at the center point.
  - Ground truth attribute heatmaps ($\hat{Q}_\theta$, $\hat{Q}_w$, $\hat{Q}_d$) lacked smoothness compared to confidence heatmap ($\hat{Q}_c$) due to dense grasp annotations in cluttered scenes.
- **In comparison GHM**:
  - Effectively highlights grasp centers.
  - Predicts robust grasp attributes, even in cluttered environments.

HGGD Framework

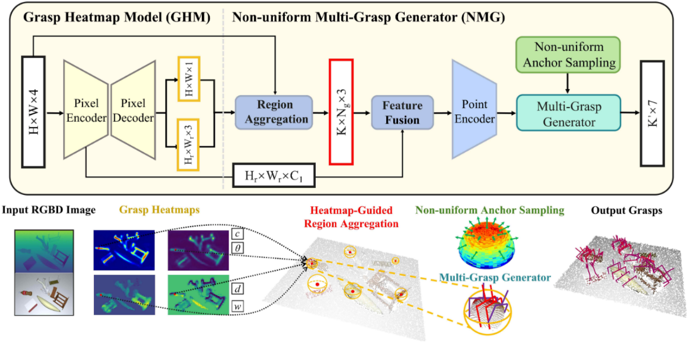

Fig. 3. The architecture of HGGD. Taking a monocular RGBD image as input, GHM generates grasp confidence heatmap Q_c and grided attributes heatmaps (Q_θ, Q_w, Q_d). Then NMG transfers the depth image to the point cloud through camera intrinsics **c** for region aggregation under the guidance of heatmaps. Feature fusion and the point encoder extract regional features fused with semantic information from GHM. Finally, a multi-grasp generator combined with a novel non-uniform anchor sampling mechanism utilizes the fusion features to output the grasps.

Non-uniform Multi-Grasp Generator (NMG)

- **Overview (What's left to do?)**:
  - Combines heatmaps and point cloud data to efficiently aggregate multiple graspable local areas.
  - Utilizes grasp attributes in each grid to predict the remaining rotation attributes and refine previously generated grasps.

Structure of Non-uniform Multi-Grasp Generator (NMG)

- **Two Main Components**:
  - **Heatmap-Guided Region Aggregation**:
    - Uses heatmaps to focus on and aggregate graspable regions from the point cloud (region aggregation).
    - Efficiently reduces the search space for grasp prediction.
    - Enhances robustness by combining semantic and geometrical features (feature fusion).
  - **Non-uniform Multi-Grasp Generator**:
    - Predicts remaining grasp rotation attributes and refines initial grasp candidates.
    - Employs a novel non-uniform anchor sampling mechanism to improve grasp diversity and quality.

Region Aggregation

- **First step**:
  - **Downsampling**:
    - Grasp confidence heatmap is downsampled using bilinear interpolation to $H_r \times W_r$, matching the size of attribute heatmaps.
    - Top k_{center} grids with the highest predicted confidence are selected, containing k_{center} local peaks in total as regional centers
    - Suppresses center density to reduce duplicates in aggregated areas.
- $k_{\textbf{center}}$ **Parameter**:
  - **During Training**: k_{center} is set to a larger value to extract most graspable regions.
  - **During Inference**: k_{center} is adjusted to balance coverage and precision in grasp detection.

Bilinear Interpolation

- **Definition**:
  - Bilinear interpolation is a method for estimating the value of a function at an intermediate point within a grid, based on the values at its surrounding four grid points.
  - It performs linear interpolation first in one direction (e.g., x) and then in the other direction (e.g., y).
- **Application in HGGD**:
  - Used to downsample grasp confidence heatmaps to match the resolution of attribute heatmaps ($H_r \times W_r$).
  - Maintains a smooth transition between grid points, ensuring high-quality interpolation.

3D Point Transformation and Sampling

- **Pixel-to-Point Transformation**:
  - Pixel centers (u, v) are transformed into 3D point centers (x, y, z) using the corresponding depth d and camera intrinsics c.
- **Ball Query for Region Cropping**:
  - A ball query is used to crop points within a spherical region.
  - Radius of the sphere is defined by the predicted grasp width w for each center.
- **Point Sampling**:
  - N_g points are sampled within each ball region using farthest point sampling.
  - This reduces computational complexity while preserving essential local geometric information.

Farthest Point Sampling (FPS)

- **Purpose**:
  - Efficiently sample a subset of points from a larger point cloud.
  - Preserve essential geometric structure while reducing computational complexity.
- **How It Works**:
  - Starts with an initial random point from the point cloud.
  - Iteratively selects the point farthest from the already sampled points.
  - Continues until the desired number of points (N_g) is sampled.
- **Advantages**:
  - Ensures a uniform spread of sampled points across the region (avoiding dense regions).
  - Captures the overall shape and geometry of the local region.

Feature Fusion

- **Purpose**:
  - Enrich local point cloud data with semantic information extracted from GHM.
  - Improve grasp robustness, particularly when point cloud input is unreliable.
- **Methodology**:
  - Use a lightweight PointNet-based network for feature extraction with semantic-to-point fusion.
  - Pixel-wise features from GHM are grouped to local points via a KNN operation.
  - Combine pooled pixel features with point features using point-wise concatenation.
- **Pipeline**:
  - Perform KNN grouping for local region analysis.
  - Apply shared Multi-Layer Perceptrons (MLP) followed by max-pooling to extract features.
  - Leverage both local geometric and semantic representations in the subsequent grasp generator.

HGGD

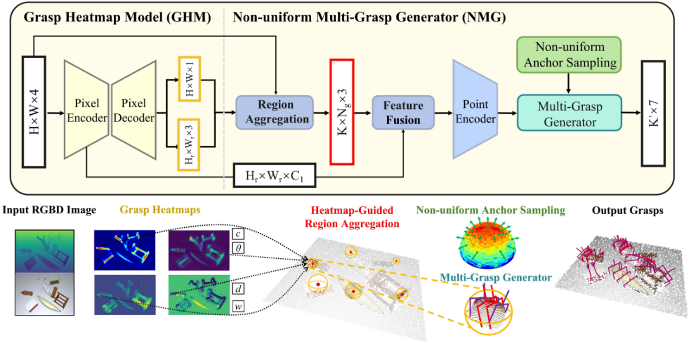

Fig. 3. The architecture of HGGD. Taking a monocular RGBD image as input, GHM generates grasp confidence heatmap Q_c and grided attributes heatmaps (Q_θ, Q_w, Q_d). Then NMG transfers the depth image to the point cloud through camera intrinsics **c** for region aggregation under the guidance of heatmaps. Feature fusion and the point encoder extract regional features fused with semantic information from GHM. Finally, a multi-grasp generator combined with a novel non-uniform anchor sampling mechanism utilizes the fusion features to output the grasps.

Feature Fusion

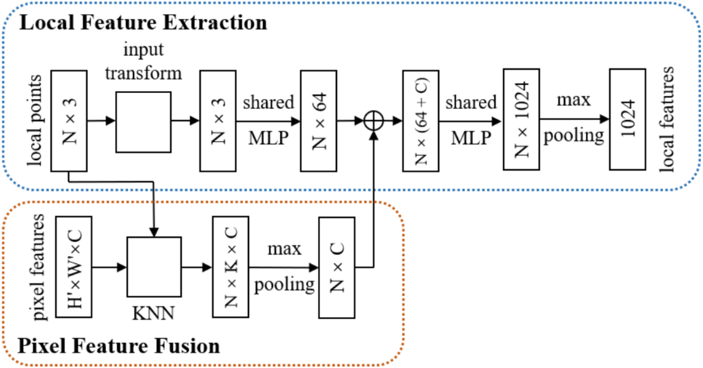

Fig. 5. The pipeline of local region feature extraction with semantic-to-point feature fusion.

Problem Statement

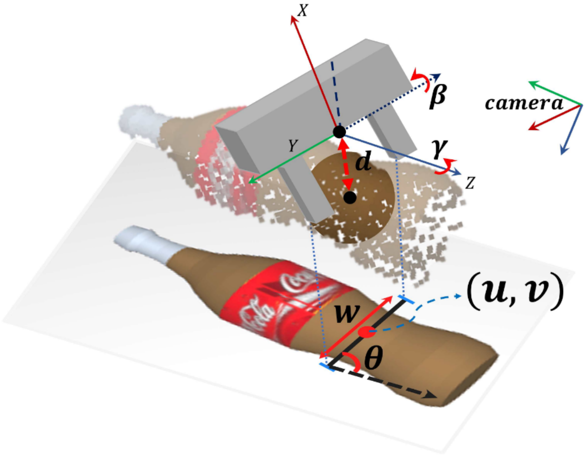

Fig. 2. Proposed grasp representation as $(u, v, \theta, w, d, \gamma, \beta)$.

Non-uniform Grasp Generator

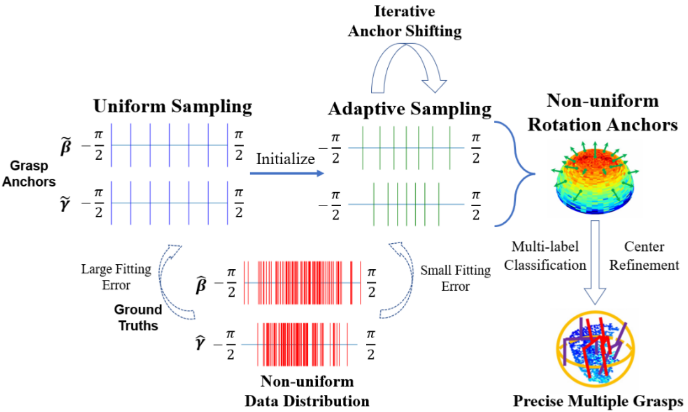

Fig. 6. Visual illustration for the procedure of the anchor shifting algorithm and the multi-grasp generation.

Anchor-Shifting Algorithm

- **Problem**: Predicting 2D rotations (γ, β) for grasp poses:
  - Rotations are continuous in $[-\frac{\pi}{2}, \frac{\pi}{2}]$.
  - Anchor-based methods achieve better localization accuracy than direct regression.
- **Anchor-Shifting Algorithm**:
  - Applied during the training process.
  - Gradually shifts anchors to minimize the fitting error between the anchor distribution and the acquired grasp rotation distribution.
  - Achieves higher performance with fewer anchors, improving both efficiency and accuracy.

Limitations of Existing Methods

- Most approaches use predefined approach vectors uniformly distributed on a sphere surface.
- This uniform distribution fails to account for uneven grasp rotation distributions.
- Trade-off exists between accuracy and speed:
  - Denser anchors improve accuracy but slow down computation.

Anchor-Shifting Algorithm

- Let's treat γ and β equally and focus on γ as an example.
- **Objective**:
  - Minimize the fitting error between grasp anchors $\tilde{\gamma}$ and ground truth rotations $\hat{\gamma}$.
- **Formula**:

$$\tilde{\gamma}^*, B_\gamma^* = \arg\min_{\tilde{\gamma}, B_\gamma} \|B_\gamma^T \tilde{\gamma} - \hat{\gamma}\|_2^2$$

- **Definitions**:
  - $\tilde{\gamma} \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]^{k_r \times 1}$: Grasp anchors.
  - $B_\gamma \in \{0, 1\}^{k_r \times K}$: One-hot encodings of the nearest anchor for each ground truth.
  - $\hat{\gamma} \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]^{K \times 1}$: Ground truth rotation angles.
  - k_r: Number of defined anchors.
  - K: Number of selected grasp ground truths during training.

Anchor-Shifting Algorithm

- **Algorithm equations**:
  3. **Update Anchor Encodings Matrix**:

$$B_{\gamma}^{(i,j)} = \begin{cases} 1, & \text{if } \arg\min_{k \in \{1,\ldots,k_r\}} \|\hat{\gamma}(j) - \tilde{\gamma}(k)\| = i, \\ 0, & \text{otherwise.} \end{cases}$$

  4. **Update Anchors Using Least Squares**:

$$\tilde{\gamma}^{*} = \left(B_{\gamma} B_{\gamma}^{T}\right)^{-1} B_{\gamma} \hat{\gamma}.$$

- **Outcome**:
  - Dynamically shifts anchors during training to reduce fitting error.
  - Achieves better performance with fewer anchors.

Algorithm Pseudocode

Algorithm 1 Non-uniform anchor shifting during training

Parameters: $\tilde{\gamma}, \tilde{\beta} \in [-\frac{\pi}{2}, \frac{\pi}{2}]^{k_r \times 1}$ - current anchors
K - grasp number threshold, T - shifting iterations
Python-Style Pseudocode:

$Grasps = list()$
while $training$ **do**
    $G = GetGraspGroudTruthsInEachRegion()$
    $Grasps.extend(G)$
    if $len(Grasps) > K$ **then**
        $\hat{\gamma}, \hat{\beta} = Grasps.\gamma, Grasps.\beta$
        for $t = 1 \rightarrow T$ **do**
            Get $\mathbf{B}_{\gamma,t}, \mathbf{B}_{\beta,t}$ with $\tilde{\gamma}_{t-1}, \tilde{\beta}_{t-1}$ per Eq.(3)
            Update $\tilde{\gamma}_t, \tilde{\beta}_t$ with $\mathbf{B}_{\gamma,t}, \mathbf{B}_{\beta,t}$ per Eq.(4)
        end for
        $Grasps.clear()$
    end if
end while

Multi-Grasp Generator: Process and Refinement

- **Input**: Region-aggregated features supervised by local grasp ground truths.
- **Multi-Label Classification**:
  - Combines anchors of the two angles (γ, β) into a k_r^2-class multi-label classification problem.
  - Uses an MLP to generate multi-label classification results, forming multiple grasps in each local region.
- **Handling First-Stage Errors**:
  - Errors in center localization during the first stage can affect the grasp generator's performance.
  - To address this, the generator:
    - Predicts grasp rotation attributes (γ, β).
    - Refines grasp centers by regressing 3-dimensional center offsets for each anchor.

Heatmap Losses in HGGD

- **Loss function**: The training objective of HGGD combines heatmap and anchor losses:

$$L = L_{Q_c} + a \times L_{\text{cls}} + b \times L_{\text{reg}} + L_{\text{anchor}} + c \times L_{\text{offset}}$$

- **Heatmap Loss Components**:
  - L_{Q_c}:
    - Pixel-wise cross-entropy loss between predicted grasp confidence (Q_c) and ground truth ($\hat{Q}_c$).
    - Uses a penalty-reduced focal loss to align with Gaussian-based heatmaps.
  - L_{cls}:
    - Focal loss for multi-label classification of θ (rotation angle).
  - L_{reg}:
    - Masked Smooth L1 loss applied to regression problems in the Grasp Heatmap Model (GHM).

Masked Smooth L1 Loss

- **What is Smooth L1 Loss?**
  - A robust loss function that combines L1 and L2 losses:

$$\text{Smooth L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1, \\ |x| - 0.5 & \text{otherwise.} \end{cases}$$

  - Provides smooth gradients for small errors and reduces sensitivity to outliers.
- **What is Masked Smooth L1 Loss?**
  - Applies the loss only to valid regions (masked areas) in the heatmap.
  - Mask ensures the loss is computed only for valid grasp points, avoiding noise from irrelevant areas.
- **Benefits**:
  - Enhances accuracy by focusing on graspable regions.
  - Prevents overfitting to noisy or invalid data points.

Anchor Losses in HGGD

- **Anchor Loss Components**:
  - L_{anchor}:
    - Local grasp rotation anchor classification loss.
    - Calculated using a focal loss to manage imbalanced anchor distributions.
  - L_{offset}:
    - Smooth L1 loss used to predict grasp center offsets for different rotation candidates.
- **Final Loss Function**:

$$L = L_{Q_c} + a \times L_{\text{cls}} + b \times L_{\text{reg}} + L_{\text{anchor}} + c \times L_{\text{offset}}$$

Implementation Details

- **Grasp Heatmap Model (GHM)**:
  - ResNet-34 is used as the pixel encoder.
  - Output channels reduced to 128 for efficient inference.
  - Pixel Decoder:
    - Includes skip connections.
    - Uses deconvolution layers for upsampling feature maps to match heatmap resolution.
  - Input Image Resolution: 640×360.
  - Grid Size: $r = 8$.
- **Non-uniform Multi-Grasp Generator (NMG)**:
  - Training: $k_{\text{center}} = 128$: Covers as many areas as possible.
  - Inference: $k_{\text{center}} = 32$ (D1) and $k_{\text{center}} = 48$ (D2).
  - Anchor Shifting Iterations: $T = 1$ (gentler anchor value updates).
  - Points Aggregated per Region: $N_g = 512$.
  - Generated Grasps per Region: $k_r = 7$.
- **End-to-End Training**

Datasets

- **Real vs Synthetic**:
  - Grasp datasets can be roughly divided into real and synthetic according to the type of observations.
- GraspNet-1Billion builds a large-scale grasp dataset in which the observations are captured in the real world.
- Simulating observations provides a more scalable alternative.
- Testing performance on both real and synthetic datasets.

Evaluation Metrics

- **For TS-ACRONYM**:
  - Collision Free Ratio (CFR) - describes the possibility of not colliding with the scene
  - Antipodal Score (AS) - describes the force closure property.
  - Coverage Rate (CR) - describes the diversity of the grasps and measures how well the generated grasps cover all ground truths.
- **For GraspNet-1Billion**:
  - Average Precision (AP) - friction coefficient of the top 50 grasp poses by force-closure metric after non-maximum suppression.

Results On TS-ACRONYM

TABLE I
RESULTS ON TS-ACRONYM DATASET

Method	CR ↑	CFR ↑	AS ↑	Time[1,2] (ms) ↓
GPD (3 channels) [4]	-	69.3 %	0.408	20342
GPD (12 channels)	-	72.9 %	0.412	19756
PointNetGPD [5]	-	74.4 %	0.434	10212
S4g [7]	0.177	83.2 %	0.618	432
REGNet [8]	0.296	94.3 %	0.662	441
HGGD	**0.503**	**98.2 %**	**0.686**	**28**

[1] Including network inference time and post-processing time.
[2] Evaluated with AMD 5600x CPU and single NVIDIA RTX 3060Ti GPU.

Grasp Quality To Coverage Rate

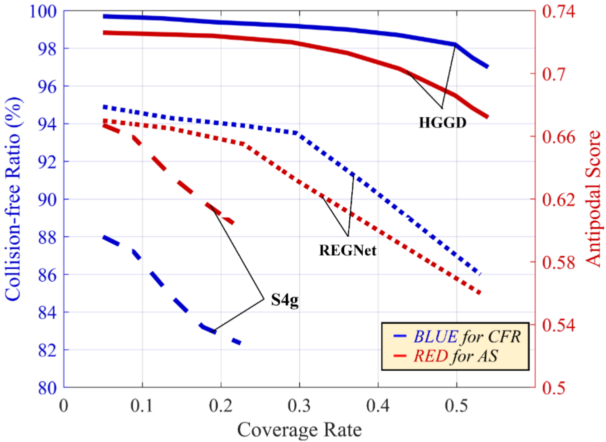

Fig. 7. (CFR, CR) and (AS, CR) curves. The red lines represent the Collision-Free Ratio and the blue lines represent the Antipodal Score. When CR increases, compared with baselines, HGGD still remains a relatively high grasp quality.

Results On GraspNet

TABLE II
DETAILED RESULTS ON GRASPNET DATASET, SHOWING APS ON REALSENSE/KINECT SPLIT AND METHOD TIME USAGE

Method	Seen			Similar			Novel			Time[1] /ms
	AP	$AP_{0.8}$	$AP_{0.4}$	AP	$AP_{0.8}$	$AP_{0.4}$	AP	$AP_{0.8}$	$AP_{0.4}$	
GPD [4]	22.87/24.38	28.53/30.16	12.84/13.46	21.33/23.18	27.83/28.64	9.64/11.32	8.24/9.58	8.89/10.14	2.67/3.16	-
PointnetGPD [5]	25.96/27.59	33.01/34.21	15.37/17.83	22.68/24.38	29.15/30.84	10.76/12.83	9.23/10.66	9.89/11.24	2.74/3.21	-
GraspNet-1B [18]	27.56/29.88	33.43/36.19	16.95/19.31	26.11/27.84	34.18/33.19	14.23/16.62	10.55/11.51	11.25/12.92	3.98/3.56	296
RGB Matters [19]	27.98/32.08	33.47/39.46	17.75/20.85	27.23/30.40	36.34/37.87	15.60/18.72	12.25/13.08	12.45/13.79	5.62/6.01	440
REGNet [8]	37.00/37.76	- / -	- / -	27.73/28.69	- / -	- / -	10.35/10.86	- / -	- / -	452
TransGrasp [35]	39.81/35.97	47.54/41.69	36.42/31.86	29.32/29.71	34.80/35.67	25.19/24.19	13.83/11.41	17.11/14.42	7.67/5.84	-
GSNet [20]	**67.12/63.50**	**78.46/74.54**	60.90/**58.11**	**54.81/49.18**	**66.72/59.27**	**46.17/41.89**	24.31/**19.78**	**30.52/24.60**	14.23/11.17	~100[2]
HGGD	64.45/61.17	72.81/69.82	**61.16**/56.52	53.59/47.02	64.12/56.78	45.91/38.86	**24.59**/19.37	30.46/23.95	**15.58/12.14**	**36**

"-": Result Unavailable.
[1] Evaluated with AMD 5600x CPU and single NVIDIA RTX 3060Ti GPU.
[2] Reported in [20] on NVIDIA RTX 1080Ti GPU since the code is not available.

Qualitative Results

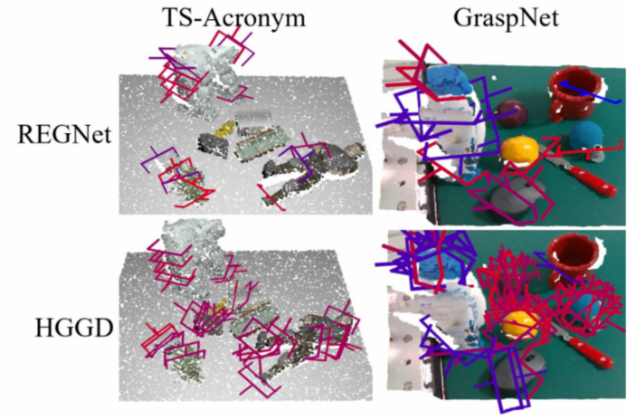

Fig. 8. Qualitative results on TS-Acronym and GraspNet-1Billion datasets. Grasps are color-coded based on their test (antipodal/force-closure) scores in RGB space, with red indicating better quality and blue indicating lower quality.

Ablation Studies

- **Analyzing the role of each module - building baseline model**:
  - Random center selection strategy.
  - Single-label classification.
  - Uniformly sampled anchors.
  - No center refinement for generated grasps.
- Apply the proposed modules to the baseline in order
- Conduct experiments

Ablation Studies

TABLE III
ABLATION ANALYSIS OF EACH MODULE

TS-ACRONYM	CR ↑	CFR ↑	AS ↑
baseline	0.144	59.7 %	0.338
+ heatmap guidance	0.450	96.9 %	0.656
+ center refinement	0.467	97.5 %	0.669
+ non-uniform anchor	0.481	97.8 %	0.679
+ multi-label classification	0.498	**98.2 %**	**0.686**
+ feature fusion	**0.503**	**98.2 %**	**0.686**

Ablation Studies

- When the point cloud is unreliable, it is difficult for point-cloud-only methods to mine adequate information for grasp detection.

TABLE IV
ABLATION ANALYSIS OF METHOD ROBUSTNESS

TS-ACRONYM with extra noise	CR $\uparrow$	CFR $\uparrow$	AS $\uparrow$
REGNet	0.159	92.5 %	0.629
HGGD w/o feature fusion	0.464	97.5 %	0.636
HGGD	**0.469**	**97.9 %**	**0.653**

Real-world Experiment

TABLE V
RESULTS OF ROBOTICS EXPERIMENTS

Scene	Object	Success	Attempt
1	9	9	10
2	8	8	8
3	10	10	11
4	8	8	9
5	9	9	10
6	8	8	8
7	10	10	10
Success Rate[1]	62 / 66 = **94%**		
Completion Rate[2]	7 / 7 = **100%**		

[1] The sum of **Attempt** dividing the sum of **Success**.
[2] The total scene number dividing the successfully cleared scene number.

Thank you for your attention!