electronics

MDPI

Article

Category Level Object Pose Estimation via Global High-Order Pooling

Changhong Jiang [1], Xiaoqiao Mu [2], Bingbing Zhang [3], Mujun Xie [1,*] and Chao Liang [4,*]

[1] School of Electrical and Electronic Engineering, Changchun University of Technology, Changchun 130012, China; jch@ccut.edu.cn
[2] School of Mechanical and Electrical Engineering, Changchun University of Technology, Changchun 130012, China; 1202001003@stu.ccut.edu.cn
[3] School of Computer Science and Engineering, Dalian Minzu University, Dalian 116602, China; icyzhang@dlnu.edu.cn
[4] Collage of Computer Science and Engineering, Changchun University of Technology, Changchun 130012, China
* Correspondence: xiemujun@ccut.edu.cn (M.X.); liangchao@ccut.edu.cn (C.L.)

Abstract: Category level 6D object pose estimation aims to predict the rotation, translation and size of object instances in any scene. In current research methods, global average pooling (first-order) is usually used to explore geometric features, which can only capture the first-order statistical information of the features and do not fully utilize the potential of the network. In this work, we propose a new high-order pose estimation network (HoPENet), which enhances feature representation by collecting high-order statistics to model high-order geometric features at each stage of the network. HoPENet introduces a global high-order enhancement module and utilizes global high-order pooling operations to capture the correlation between features and fuse global information. In addition, this module can capture long-term statistical correlations and make full use of contextual information. The entire network finally obtains a more discriminative feature representation. Experiments on two benchmarks, the virtual dataset CAMERA25 and the real dataset REAL275, demonstrate the effectiveness of HoPENet, achieving state-of-the-art (SOTA) pose estimation performance.

Keywords: pose estimation; pooling; high-order

Citation: Jiang, C.; Mu, X.; Zhang, B.; Xie, M.; Liang, C. Category Level Object Pose Estimation via Global High-Order Pooling. *Electronics* **2024**, *13*, 1720. https://doi.org/10.3390/electronics13091720

Academic Editor: Zhenhua Guo

Received: 20 March 2024
Revised: 19 April 2024
Accepted: 26 April 2024
Published: 29 April 2024

1. Introduction

Object pose estimation aims to identify an object's six-dimensional pose in the camera coordinate system. It is a very challenging problem for computer vision and robotics technology with critical applications in robotic grasping [1,2], scene understanding, augmented reality [3], and autonomous driving [4,5]. Currently, there are two types of 6D object pose estimation methods: instance-level pose estimation [2,6] and category-level pose estimation [7–11]. Instance-level pose estimation requires pre-obtaining the 3D CAD model of each object and its dimensions. However, since most objects have never been seen and no known CAD model exists, this approach is limited. In contrast, category-level pose estimation does not require precise CAD models, can be generalized to unseen objects, and is more generally applicable.

Recently, many methods for category-level pose estimation have emerged, which can be divided into two types: prior-free [9,12–14] and prior-based [8,15–19] methods. Prior-free methods primarily emphasize the design of the model structure to extract features fitting the data, yet their performance tends to be poor. Conversely, prior-based methods leverage prior knowledge to guide models, leading to significant progress and widespread attention. Existing prior-based class-level pose estimation methods, such as NOCS [9], define a shared space with consistent object scaling and orientation known as normalized object coordinate space (NOCS) and aim to recover the perspective in NOCS for pose estimation. To address

intra-class shape variations, prior studies [8] pre-learn dense and static point clouds from given object instances as classification shape priors to explicitly model deformations for reconstructing 3D object models. However, this approach results in redundant shape priors and increases computational costs. Therefore, Query6DoF [20] introduces sparse and learnable category queries instead of dense point clouds as shape priors, utilizing implicit shared semantics to encode object shapes and reduce computational overhead.

Additionally, since the input is point cloud data and point clouds are unordered, most methods [12] employ techniques similar to PointNet [21]. However, PointNet's main structure, a pooling operation, leads to the loss of detailed local geometric structure information and cannot adaptively learn regions of interest, making it unsuitable for extracting feature information. Building upon Query6DoF [20], we further explore effective modeling of complex geometric feature distributions. Traditional methods typically employ global average pooling (first-order) operations to explore geometric features, obtaining only first-order statistical information and failing to fully utilize the network's potential. Hence, it is crucial to consider feature distribution comprehensively. Researchers have investigated higher-order pooling methods and achieved unexpected results in fields such as image classification and video recognition [22–27]. Most methods integrate high-order pooling at the end of the network to substitute global average pooling (GAP). However, the emergence of GSoP [28] effectively introduces high-order representations in early layers, thereby enhancing the nonlinear capabilities of convolutional networks. Motivated by this, we investigate the impact of high-order modeling methods on category-level 6D object pose estimation.

Specifically, we propose a high-order pose estimation network (HoPENet), which enhances feature representation by collecting high-order statistics to model high-order geometric features at each network stage. The network structure is depicted in Figure 1. HoPENet follows the model architecture of Query6DoF. Firstly, it employs category queries as implicit shape priors to capture the most representative object features and disregard insignificant shape details, thereby reducing computational overhead. Subsequently, global high-order enhancement modules are incorporated into each network stage, facilitating the gradual acquisition of high-order geometric information to capture feature correlations. Differing from ordinary pooling operations, the global high-order enhancement module employs global high-order pooling operations to gather high-order statistics from instance features, making it more adept at extracting geometric feature information. Consequently, the network ultimately achieves a more discriminative feature representation. public data benchmarks. The proposed model is evaluated on two public data benchmarks. The contributions of this paper can be summarized as follows:

- We propose the HoPENet model for category-level 6D object pose estimation. HoPENet incorporates global high-order enhancement modules into each stage of the model and utilizes global high-order information throughout the network to model complex feature distributions, thereby enabling the model to learn a more discriminative feature representation.
- The global high-order enhancement module incorporates high-order information into the attention mechanism, employs global high-order pooling operations to capture feature correlations, integrates global information, and enhances features. By modeling the high-order statistics of the entire tensor, the proposed module can capture long-term statistical correlations and fully leverage contextual information.
- We conduct comprehensive ablation studies to validate the effectiveness of the proposed HoPENet network in 6D object pose estimation. Experimental results on the REAL275 and CAMERA25 datasets demonstrate that the proposed method surpasses the baseline model.

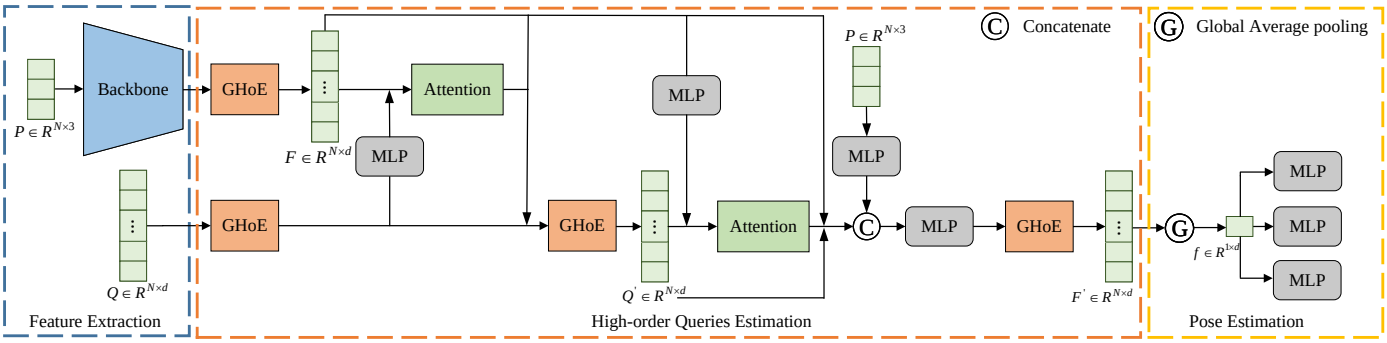

Figure 1. Diagram of the high-order pose estimation network structure. The network comprises feature extraction, high-order query, and pose estimation components.

2. Related Works

2.1. Category-Level 6D Object Pose Estimation

NOCS [9] defines a normalized object coordinate space to handle distinct and unseen object instances within a given category. However, NOCS lacks an explicit representation of shape changes. SPD [8] proposes reconstructing object models in NOCS to capture intra-class shape variations explicitly, modeled by pre-learning shape priors. DualPoseNet stacks two parallel pose decoders (explicit and implicit encoders) on top of a shared pose encoder for complementary supervision. SGPA [15] is influenced by the class prior advantage and introduces a structure-guided prior adaptive scheme to estimate the 6D pose of the object. DPDN [19] utilizes self-supervised methods to reduce domain gaps and implements a deep version of shape prior deformation in feature space for direct regression. However, the class above priors mostly use dense point clouds, resulting in redundancy and high computational cost. Query6DoF [20] employs sparse queries as shape priors and an attention mechanism to select object features. Similarly, we build upon this network architecture but focuses more on effectively extracting geometric features.

2.2. Higher Order Pooling

Researchers have explored the effects of higher-order pooling methods on tasks such as images and videos in the context of deep convolutional neural networks [22–27]. In comparison to first-order methods like global average pooling, high-order pooling can capture richer statistics. DeepO2P [22] initially incorporates covariance pooling at the end of the final convolutional layer to enhance the network's feature expression capabilities significantly. G2DeNet [29] enhances DeepO2P [22] by effectively utilizing a global gaussian distribution to model feature statistics. Meanwhile, MPN-COV [23], proposed by Li et al., demonstrates exceptional performance by judiciously leveraging the geometric structure of covariance. However, this method involves computing the intrinsic decomposition (EIG) of the square root of the matrix, which may pose limitations on computational power. Subsequently, iSQRT-COV [26] introduced a method to rapidly compute the square root of a matrix, suitable for existing GPU operations. The aforementioned works employ high-order modeling at the end of the network. However, GSoP [28] explores introducing high-order representation in early layers, which can leverage overall information across the entire network and yield favorable outcomes in image recognition. Our study will investigate the impact of high-order modeling at different stages of the network on category-level 6D object pose estimation.

3. Method

3.1. High-Order Pose Estimation Network

Our network aims to utilize only point clouds as input to estimate the pose and size of category-level objects, enhance the network's capability to extract geometric features, and facilitate the model in learning a more discriminative feature representation. The method proposed in this paper is grounded on the Mask R-CNN framework. It in-

volves transferring the target segmentation results to the three-dimensional point cloud and uniformly sampling the point cloud data. Utilize this point cloud data as input for our method.

The input point cloud data can be expressed as $P, P \in R^{N\times 3}$, where N represents the number of center points, and each point has 3-dimensional coordinates. The point cloud data is sent to the feature extraction module to extract the geometric features of the target $F, F \in R^{N\times d}$, where d represents the dimension of the space. Referring to [20], generate a set of queries for each object category, utilizing $Q, Q \in R^{N\times d}$ as an implicit shape prior. Traditional approaches with shape priors pre-learn dense and static point clouds from given object instances to model deformations for explicit 3D object model reconstruction. However, such explicit shape priors are redundant and lead to increased computational costs. In Literature [20], query Q is employed as an implicit shape prior. The number of queries Q is significantly smaller than that of points P, thereby reducing the computational burden. Additionally, Q is learnable and does not require additional training.

Specifically, the structure of the proposed high-order pose estimation network is depicted in Figure 1. For feature extraction, PointNet++ is employed as the backbone to extract geometric features from the point cloud data. The extracted features are then utilized for high-order query estimation to establish similarity relationships. The specific operation is as follows: Initially, the feature F and the query Q are forwarded to a global high-order enhancement module to capture feature correlations, fuse global information, and enhance geometric features. The detailed operation of this module is described in the subsequent section.

Next, compute the cross-attention between feature F and query Q to determine the relevant features. The calculation process involves:

$$A^i = Attn(F, Q) \tag{1}$$

$A^i, A^i \in R^{N_Q\times N_P}$ is the i-th attention map. N_Q is the number of queries. N_P denotes the number of points. Parts with similar semantics to the query can be obtained by computing attention. Therefore, the result of the cross-attention is:

$$A = (A^i F)W \tag{2}$$

where $A \in R^{N_Q\times d}$. Add the attention-extracted feature A to Q to obtain a new query:

$$Q^{'} = Q + A \tag{3}$$

where $Q^{'} \in R^{N_Q\times d}$. Additionally, prior to this step, a global high-order enhancement module was incorporated to enhance features. Consequently, the new query $Q^{'}$ acquires useful features from F. Then, referring to the feature space established by reference [20], calculate the similarity between F and $Q^{'}$ to obtain the corresponding matrix:

$$M = Norm(MLP)(FQ^{'T}) \tag{4}$$

where $M \in R^{N_Q\times N_P}$. Similar to the attention calculation method, the sampled features of M and $Q^{'}$ are obtained, with residual connections added:

$$M^{'} = MQ^{'} + F \tag{5}$$

Next, match the obtained sampled features with the original object features and concatenate them using a simple MLP. To further enhance the geometric features, a global high-order enhancement module is inserted at this stage. Subsequently, the feature $F^{'}$ is obtained, where $F^{'} \in R^{N_P\times d}$. Finally, in the pose estimation stage, three MLPs are employed to predict the object's rotation, translation, and size, respectively, to determine the final pose and size of the object.

3.2. Global High-Order Enhancement Block

Given the unordered nature of point cloud data, most networks employ the PointNet model for feature extraction. However, this model primarily consists of pooling operations, which may lead to the loss of geometric structure in features and hinder overall modeling. Therefore, we propose a global high-order enhancement module that utilizes global high-order pooling operations to capture feature correlations, integrate global information, and enhance features.

The structure of the global high-order enhancement module is shown in Figure 2. Given an input tensor $t, t \in R^{N \times D}$, where N represents the number of points in the point cloud, and each point has D-dimensional coordinates. Firstly, convolutional layers downsample N to $N^{'}$ to alleviate the computational burden. Subsequently, a second-order pooling operation is employed to compute the feature correlation, resulting in a covariance matrix of $N \times N$. It can be expressed as:

$$c = conv(2^{nd} pooling(t)) \tag{6}$$

Here, $2^{nd} pooling$ represents the global second-order pooling operation. Attention modules typically utilize global first-order pooling to compute the average value of a specific dimension, which limits their modeling capabilities for geometric features. We employ a global second-order pooling method, which calculates the statistical correlation between a specific value and all values in that dimension. This approach effectively captures the geometric relationships of features and enhances the modeling ability of the network. Next, two consecutive convolutional layers are utilized to obtain the weight vector $t^{'}$, as follows:

$$t^{'} = conv1(conv2(c)) \tag{7}$$

where $t^{'} \in R^{N^{'} \times 1}$. Then, multiply the weight vector with the input tensor to obtain the new feature f.

$$f = t \cdot t^{'} \tag{8}$$

Here, $f \in R^{N \times D}$. The aforementioned process represents an operation in the quantity dimension (Number-wise GHoE Block) and can also be extended to the position dimension (Position-wise GHoE Block). At this point, the correlation between spatial positions is computed, resulting in the covariance matrix of $D \times D$. We also compared two types of calculation methods in the experiment. The specific results are presented in Section 4.2.

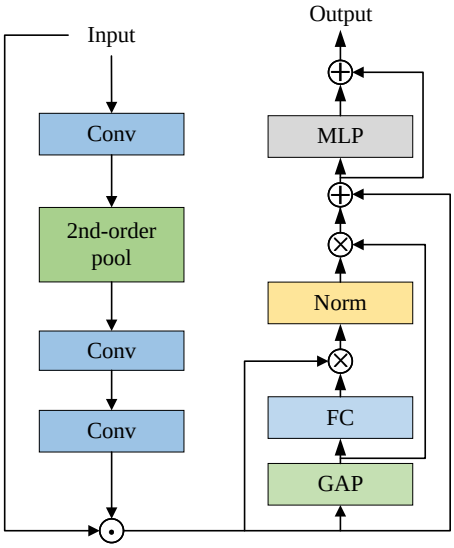

Figure 2. Overview of global high-order enhancement module.

After obtaining the feature f, the global features are fused. This can be formulated as follows:

$$m = FC(GAP(f)) \tag{9}$$

$$f^{'} = f + Norm(fm^{T})g \tag{10}$$

Firstly, global average pooling is employed for the above process to obtain global features. Subsequently, the attention operation is utilized, with f as the query and m as the key and value, to derive $f^{'}$. Finally, $f^{'}$ is passed through an MLP with a residual connection added.

4. Experiments

4.1. Implement Details

4.1.1. Datasets

We evaluate the proposed HoPENet using two benchmarks: the virtual dataset CAMERA25 and the real dataset REAL275 [9]. The CAMERA25 dataset is generated by compositing virtual objects into natural scenes context-awarely, with 1085 object instances rendered. The CAMERA25 dataset contains 300 K synthetic RGB-D images, of which 25 K images are used for testing. The REAL275 dataset is complementary to the CAMERA25 dataset and contains 8K RGB-D images, of which 4.3 k real-world images are used for training, 0.9 k for validation, and 2.7 k for testing. Both datasets contain six categories: bottles, bowls, cameras, cans, laptops, and mugs.

4.1.2. Implement Details

To build HoPENet, we employ the same data processing method as [8] and select PointNet++ to extract features. For the model training, we utilize the AdamW optimizer, set the initial learning rate to 1e-4, and the weight decay to 1e-4. The model undergoes training for a total of 100 epochs, with a batch size of 15×4. The model was trained and tested using the open-source PyTorch deep learning framework, and all experiments were conducted on a PC equipped with 4 NVIDIA GeForce RTX 3090 GPUs. For evaluation indicators, we adhere to the NOCS [9] evaluation scheme and utilize the mean Average Precision (mAP) and n°m cm under various intersection-over-union (IoU) thresholds as evaluation criteria. 3D IoU calculates the overlap between two 3D bounding boxes under predicted and ground truth poses. A prediction is deemed correct if the overlap ratio exceeds the fixed threshold. We use thresholds of 25, 50, and 75 for measurement, respectively. n°m represents the calculated rotation and translation error between predicted and ground truth poses. A prediction is deemed correct if both the rotation and translation errors are below the angle and distance thresholds. Four standards of 5°2 cm, 5°5 cm, 10°2 cm, and 10°5 cm are used, respectively.

4.2. Ablation Studies

4.2.1. The Impact of Covariance Size

The global high-order enhancement module (GHoE module) plays a central role in the network proposed in this paper, where calculating the covariance matrix is a critical operation. Therefore, the impact of the size of the covariance matrix on model performance was evaluated on the REAL275 dataset, and the results are shown in Table 1. Firstly, the table presents the performance of the original network without employing the global high-order enhancement module. It is evident that regardless of the size of the covariance matrix used, the proposed network outperforms the baseline model across most evaluation indicators. This demonstrates the effectiveness of the GHoE module in improving the feature expression capability of the model. Subsequently, the table illustrates the impact of the number-wise GHoE Block on model performance for N values of 64, 128, and 256, respectively. When $N = 128$, HoPENet achieves the best results, with mAP values of

84.3%, 82.7%, and 76.2% for IoU_{25}, IoU_{50}, and IoU_{75}, respectively, slightly surpassing those of the baseline model. For the 5°2 cm metrics, HoPENet achieves a 50.5% mAP, a 3.7% improvement over the baseline model. It also outperforms the baseline model significantly in other metrics. Performance begins to degrade as the dimensions of N increase or decrease. Lastly, the table presents the results obtained by the position-wise GHoE Block when $d = 128$. Despite the results showing a slight improvement of 1% over the baseline model for the 5°2 cm metrics, there remains a gap compared to the number-wise GHoE Block. This indicates that performing covariance pooling using multiple points is more effective than using coordinate locations for covariance pooling.

Table 1. Effect on covariance matrix size on REAL275 dataset.

		mAP						
Configuration	**Size**	IoU_{25}	IoU_{50}	IoU_{75}	**5°2 cm**	**5°5 cm**	**10°2 cm**	**10°5 cm**
baseline	-	-	**82.9**	76.0	46.8	54.7	67.9	81.6
number-wise cov size N	64 × 64	84.3	81.9	75.2	48.0	55.1	69.1	81.6
number-wise cov size N	128 × 128	**84.3**	82.7	**76.2**	**50.5**	**57.9**	**70.6**	**82.5**
number-wise cov size N	256 × 256	84.3	82.4	75.4	47.8	56.5	67.9	81.1
position-wise cov size d	128 × 128	84.2	81.6	75.4	47.8	55.0	69.7	81.6

4.2.2. Fusion of Number- and Position-Wise GHoE Block

Number- and Position-wise GHoE Block can be combined to capture high-order statistical information across various dimensions. Specifically, the average, maximum, and cascade operations are employed to fuse tensors of two dimensions, respectively. The experimental results are presented in Table 2. The fusion effect is evaluated on the CAMERA25 and REAL275 datasets, respectively. The left side of Table 2 displays the experimental results of the CAMERA25 dataset. At this point, the average fusion method demonstrates optimal effectiveness, surpassing the other two fusion approaches and outperforming the number- and position-wise GHoE Block alone. On the right side of the table, results from the REAL275 dataset reveal that leveraging covariance pooling based on the number of points remains the most effective. However, fusion results are less favorable, particularly with cascade operations performing the worst. Notably, the average result aligns closely with using the Number-wise GHoE Block alone. These disparities stem from differences in the datasets. Consequently, we designate the average fused network as HoPENet, while the model solely employing covariance pooling based on the number of points is defined as HoPENet*. Both model architectures serve as methods in this article.

Table 2. Comparison of different fusion schemes on the REAL275 and CAMERA25 dataset.

	CAMERA25							REAL275						
Configuration	IoU_{25}	IoU_{50}	IoU_{75}	**5°2 cm**	**5°5 cm**	**10°2 cm**	**10°5 cm**	IoU_{25}	IoU_{50}	IoU_{75}	**5°2 cm**	**5°5 cm**	**10°2 cm**	**10°5 cm**
position-wise	94.5	92.2	88.4	78.4	84.0	83.9	90.4	84.2	81.6	75.4	47.8	55.0	69.7	81.6
number-wise	94.5	92.2	88.4	78.4	84.0	83.9	90.4	**84.3**	**82.7**	**76.2**	**50.5**	57.9	70.6	**82.5**
average	**94.5**	**92.6**	**89.3**	**79.8**	**85.2**	**84.8**	**91.3**	84.3	82.3	76.2	49.0	56.7	**71.1**	81.7
maximum	94.5	92.5	88.7	79.2	84.4	84.6	90.8	84.3	81.6	75.0	48.7	**58.9**	68.5	81.9
concatenation	94.4	92.3	89.1	78.7	84.1	84.4	91.1	84.1	82.1	71.2	34.6	50.4	57.4	81.5

4.2.3. The Impact of the Position of Global Higher-Order Enhancement Modules

As described in the methods section, global higher-order enhancement modules are inserted at four different locations. In this experiment, we investigate the impact of placing global high-order enhancement modules at various locations on model performance on

the REAL275 dataset. The results are presented in Table 3. Each position is tested with only one global higher-order module inserted at a time. It is observed that placing the module closer to the end of the network yields better results. Specifically, when the global high-order enhancement module is inserted at Position3, our method achieves the best performance according to the 5°2 cm metric. Furthermore, transitioning to Position4 and inserting the global high-order enhancement module results in optimal measurements at 5°5 cm, 10°2 cm, and 10°5 cm. However, this operation has little impact on indicator IoU.

Table 3. Comparison of global high-order enhancement module numbers on the REAL275 dataset.

	mAP						
	IoU_{25}	IoU_{50}	IoU_{75}	5°2 cm	5°5 cm	10°2 cm	10°5 cm
Positon1	84.3	**82.7**	76.0	48.0	56.5	68.7	81.7
Positon2	84.3	82.4	74.4	48.0	55.7	68.0	80.9
Positon3	84.3	82.0	75.1	**48.7**	56.9	70.1	80.2
Positon4	**84.3**	82.1	**75.2**	48.5	**57.0**	**70.4**	**82.6**

4.2.4. The Impact of the Number of Global Higher-Order Enhancement Modules

We assess the impact of the number of global high-order enhancement modules on model performance, as presented in Table 4. Firstly, the upper section of the table displays the results obtained by inserting two global high-order enhancement modules. Notably, when positioned at 3 and 4, the model achieves optimal measurements at 10°2 cm and 10°5 cm, respectively, with other indicators also exhibiting relatively high performance. This indicates that the insertion of two global high-order enhancement modules is effective, with closer proximity to the end of the network resulting in improved model efficacy. Subsequently, the middle section of Table 4 compares the results obtained by inserting three global high-order enhancement modules. Interestingly, the results do not significantly differ from those obtained with only two modules. This similarity may be attributed to the comparable enhancement strength between the two and three modules. Finally, the last row of data presents the results obtained by inserting four global high-order enhancement modules, with the model achieving the highest mAP of 50.5% under the 5°2 cm metric. This highlights the importance of considering both the number and positioning of inserted global high-order enhancement modules, aligning with the network architecture design.

Table 4. Comparison of global high-order enhancement module positions on the REAL275 dataset.

	mAP						
	IoU_{25}	IoU_{50}	IoU_{75}	5°2 cm	5°5 cm	10°2 cm	10°5 cm
Position1+2	84.2	82.8	75.6	45.2	53.7	69.4	82.6
Position1+3	84.3	82.8	76.4	49.4	58.9	68.9	81.5
Position1+4	84.3	83.0	74.9	46.9	55.9	68.7	82.6
Position2+3	84.3	82.6	75.3	47.1	56.0	69.8	82.5
Position2+4	84.3	82.2	76.0	47.7	55.5	71.0	82.0
Position3+4	84.3	82.7	76.6	48.8	57.3	70.6	83.7
Position1+2+3	84.3	81.9	75.2	47.6	55.5	69.5	81.9
Position1+2+4	84.3	81.4	74.2	45.7	53.9	67.9	80.7
Position2+3+4	84.3	82.4	76.1	47.4	54.8	70.0	82.2
Position1+2+3+4	84.3	82.7	76.2	50.5	57.9	70.6	82.5

4.3. Comparisons with Existing Methods

We compared the proposed HoPENet with existing methods on the CAMERA25 and REAL275 datasets, and the results are presented in Table 5. It is observed that for the CAMERA25 dataset, besides measuring IoU_{50}, the HoPENet* method introduced in this

paper outperforms all existing methods in other indicators. Notably, IoU is not particularly sensitive to object pose estimation. Moreover, HoPENet* is only 0.8% lower than GPV-Pose and achieves the second-highest mAP in the table, demonstrating the effectiveness of this method. For the REAL275 dataset, HoPENet achieves a 50.5% mAP under the 5°2 cm metric, showcasing the best performance compared to existing methods. Specifically, it surpasses NOCS by 42.3%, Query6DoF by 1.5%, and exhibits significantly superior performance compared to other representative methods such as DualPoseNet, GPV-Pose, SPD, SAR-Net, SGPA, and RBP-Pose. Additionally, HoPENet also attains the highest mAP in the table at the 10°2 cm metric. In addition, we visualize several metrics of HoPENet alongside other methods on the CAMERA256 dataset in Figure 3 for intuitive observation. The above indicates that the proposed method of using higher-order pooling is effective for pose estimation tasks.

Table 5. Comparison with state-of-the-art methods on CAMERA25 and REAL275 datasets is presented. HoPENet represents the result of covariance pooling after averaging by number and location, while HoPENet* denotes a model solely performing covariance pooling based on the number of points. The best results are highlighted in bold.

	CAMERA25						REAL275					
Method	IoU_{50}	IoU_{75}	5°2 cm	5°5 cm	10°2 cm	10°5 cm	IoU_{50}	IoU_{75}	5°2 cm	5°5 cm	10°2 cm	10°5 cm
NOCS [9]	83.9	69.5	32.3	40.9	48.2	64.6	78.0	30.1	7.2	10.0	13.8	25.2
DualPoseNet [13]	92.4	86.4	64.7	70.7	77.2	84.7	79.8	62.2	29.3	35.9	50.0	66.8
GPV-Pose [30]	**93.4**	88.3	72.1	79.1	-	89.0	**83.0**	64.4	32.0	42.9	-	73.3
SPD [8]	93.2	83.1	54.3	59.0	73.3	81.5	77.3	53.2	19.3	21.4	43.2	54.1
SAR-Net [31]	86.8	79.0	66.7	70.9	75.6	80.3	79.3	62.4	31.6	42.3	50.3	68.3
SGPA [15]	93.2	88.1	70.7	74.5	82.7	88.4	80.1	61.9	35.9	39.6	61.3	70.7
RBP-Pose [32]	93.1	89.0	73.5	79.6	82.1	89.5	-	67.8	38.2	48.1	63.1	79.2
Query6DoF [20]	92.3	88.6	78.4	83.9	84.0	90.5	82.9	76.0	46.8	54.7	67.9	81.6
HoPENet	92.2	88.4	78.4	84.0	83.9	90.4	82.7	**76.2**	**50.5**	**57.9**	70.6	**82.5**
HoPENet*	92.6	**89.3**	**79.8**	**85.2**	**84.8**	**91.3**	82.3	**76.2**	49.0	56.7	**71.1**	81.7

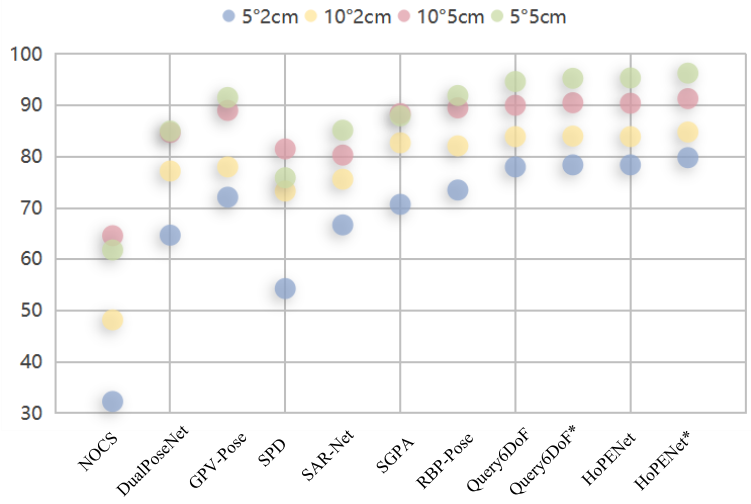

Figure 3. Visualized comparison between HoPENet and other recent works.

5. Conclusions

We propose HoPENet, a category-level 6D object pose estimation method designed for unseen objects. This method introduces global high-order enhancement modules at each network stage to gradually learn high-order geometric information for capturing

feature correlations. The global high-order enhancement module utilizes global high-order pooling operations to gather high-order statistics from instance features, making it more suitable for extracting geometric feature information. Consequently, the network achieves a more discriminative feature representation, which competes effectively with other methods. The advanced statistical metrics used in this paper employ global second-order pooling operations. In the future, we aim to investigate the impact of higher-order operations on pose estimation tasks.

Author Contributions: Conceptualization: C.J., B.Z. and X.M.; methodology, B.Z., M.X. and C.L.; software, B.Z. and M.X.; validation, C.J., M.X., B.Z., X.M. and C.L.; formal analysis, B.Z. and M.X.; investigation, M.X. and C.L.; resources, M.X. and C.L.; data curation, M.X.; writing—original draft preparation, M.X.; writing—review and editing, C.J., B.Z., X.M. and C.L.; visualization, X.M. and C.L.; supervision, X.M. and C.L.; project administration, X.M. and C.L.; funding acquisition, X.M. and C.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Science and Technology Development Program Project of Jilin Province (20230201111GX and 20220201071GX).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Collet, A.; Martinez, M.; Srinivasa, S.S. The moped framework: Object recognition and pose estimation for manipulation. *Int. J. Robot. Res.* **2011**, *30*, 1284–1306. [CrossRef]
2. Tremblay, J.; To, T.; Sundaralingam, B.; Xiang, Y.; Fox, D.; Birchfield, S. Deep object pose estimation for semantic robotic grasping of household objects. *arXiv* **2018**, arXiv:1809.10790.
3. Marchand, E.; Uchiyama, H.; Spindler, F. Pose estimation for augmented reality: A hands-on survey. *IEEE Trans. Vis. Comput. Graph.* **2015**, *22*, 2633–2651. [CrossRef] [PubMed]
4. Xu, D.; Anguelov, D.; Jain, A. Pointfusion: Deep sensor fusion for 3d bounding box estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 244–253.
5. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3d object detection network for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1907–1915.
6. Peng, S.; Liu, Y.; Huang, Q.; Zhou, X.; Bao, H. Pvnet: Pixel-wise voting network for 6dof pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 4561–4570.
7. Lin, Y.; Tremblay, J.; Tyree, S.; Vela, P.A.; Birchfield, S. Single-stage keypoint-based category-level object pose estimation from an RGB image. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 1547–1553.
8. Tian, M.; Ang, M.H.; Lee, G.H. Shape prior deformation for categorical 6d object pose and size estimation. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XXI 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 530–546.
9. Wang, H.; Sridhar, S.; Huang, J.; Valentin, J.; Song, S.; Guibas, L.J. Normalized object coordinate space for category-level 6d object pose and size estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 2642–2651.
10. Zhang, J.; Wu, M.; Dong, H. Generative Category-level Object Pose Estimation via Diffusion Models. *Adv. Neural Inf. Process. Syst.* **2024**, *36* .
11. Tatemichi, H.; Kawanishi, Y.; Deguchi, D.; Ide, I.; Murase, H. Category-level Object Pose Estimation in Heavily Cluttered Scenes by Generalized Two-stage Shape Reconstructor. *IEEE Access* **2024** , *12*, 33440–33448. [CrossRef]
12. Chen, W.; Jia, X.; Chang, H.J.; Duan, J.; Shen, L.; Leonardis, A. Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 1581–1590.
13. Lin, J.; Wei, Z.; Li, Z.; Xu, S.; Jia, K.; Li, Y. Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3560–3569.
14. Wang, C.; Martín-Martín, R.; Xu, D.; Lv, J.; Lu, C.; Fei-Fei, L.; Savarese, S.; Zhu, Y. 6-pack: Category-level 6d pose tracker with anchor-based keypoints. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 10059–10066.
15. Chen, K.; Dou, Q. Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 2773–2782.

16. Chen, X.; Dong, Z.; Song, J.; Geiger, A.; Hilliges, O. Category level object pose estimation via neural analysis-by-synthesis. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XXVI 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 139–156.
17. Ze, Y.; Wang, X. Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and a new dataset. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 27469–27483.
18. Lee, T.; Lee, B.U.; Shin, I.; Choe, J.; Shin, U.; Kweon, I.S.; Yoon, K.J. UDA-COPE: Unsupervised domain adaptation for category-level object pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14891–14900.
19. Lin, J.; Wei, Z.; Ding, C.; Jia, K. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 19–34.
20. Wang, R.; Wang, X.; Li, T.; Yang, R.; Wan, M.; Liu, W. Query6dof: Learning sparse queries as implicit shape prior for category-level 6dof pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 14055–14064.
21. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
22. Ionescu, C.; Vantzos, O.; Sminchisescu, C. Matrix backpropagation for deep networks with structured layers. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2965–2973.
23. Li, P.; Xie, J.; Wang, Q.; Zuo, W. Is second-order information helpful for large-scale visual recognition? In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2070–2078.
24. Lin, T.Y.; RoyChowdhury, A.; Maji, S. Bilinear CNN models for fine-grained visual recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1449–1457.
25. Yu, T.; Cai, Y.; Li, P. Toward faster and simpler matrix normalization via rank-1 update. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XIX 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 203–219.
26. Li, P.; Xie, J.; Wang, Q.; Gao, Z. Towards faster training of global covariance pooling networks by iterative matrix square root normalization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 947–955.
27. Wang, Q.; Xie, J.; Zuo, W.; Zhang, L.; Li, P. Deep CNNs meet global covariance pooling: Better representation and generalization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 2582–2597. [CrossRef] [PubMed]
28. Gao, Z.; Xie, J.; Wang, Q.; Li, P. Global second-order pooling convolutional networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 3024–3033.
29. Wang, Q.; Li, P.; Zhang, L. G2DeNet: Global Gaussian distribution embedding network and its application to visual recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2730–2739.
30. Di, Y.; Zhang, R.; Lou, Z.; Manhardt, F.; Ji, X.; Navab, N.; Tombari, F. Gpv-pose: Category-level object pose estimation via geometry-guided point-wise voting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 6781–6791.
31. Lin, H.; Liu, Z.; Cheang, C.; Fu, Y.; Guo, G.; Xue, X. Sar-net: Shape alignment and recovery network for category-level 6d object pose and size estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 6707–6717.
32. Zhang, R.; Di, Y.; Lou, Z.; Manhardt, F.; Tombari, F.; Ji, X. Rbp-pose: Residual bounding box projection for category-level pose estimation. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 655–672.