International Journal of Computational Science and Engineering

ISSN online: 1742-7193 - ISSN print: 1742-7185
https://www.inderscience.com/ijcse

Rotation-invariant 3D convolutional neural networks for 6D object pose estimation

Zhizhong Chen, Zhihang Wang, Xue Hui Xing, Tao Kuai

DOI: 10.1504/IJCSE.2025.10069950

Article History:

Received:	20 September 2024
Last revised:	18 December 2024
Accepted:	19 February 2025
Published online:	20 March 2025

Rotation-invariant 3D convolutional neural networks for 6D object pose estimation

Zhizhong Chen*, Zhihang Wang, Xue Hui Xing and Tao Kuai

Northwest Institute of Mechanical and Electrical Engineering,
No. 5 Biyuan East Road, Weicheng District,
Xianyang City, Shaanxi Province, 712000, China
Email: chenzz_swx@163.com
Email: 1350152250@qq.com
Email: 981429757@qq.com
Email: ktao1998@163.com
*Corresponding author

Abstract: 6D object pose estimation, crucial for applications such as scene understanding, AR/VR, and robotic grasping, focuses on determining an object's rotation and translation from single-view images. Despite advancements in 3D deep learning, existing methods still struggle with large shape variations, high training demands, and unseen poses. This paper addresses these issues by introducing a rotation-invariant neural network. We propose a rotation-invariant 3D convolutional network that processes a point cloud to predict per-point canonical coordinates, from which the 6D pose, is estimated. The network utilises relative distances and angles within a representative point set. Experiments on a public dataset show that our method outperforms several state-of-the-art baselines, excelling in handling novel poses and severe occlusions. Ablation studies further highlight the importance of the individual components.

Keywords: rotation-invariant; geometric feature extraction; pose estimation.

Reference to this paper should be made as follows: Chen, Z., Wang, Z., Xing, X.H. and Kuai, T. (2025) 'Rotation-invariant 3D convolutional neural networks for 6D object pose estimation', *Int. J. Computational Science and Engineering*, Vol. 28, No. 8, pp.1–9.

Biographical notes: Zhizhong Chen is a Master's candidate at the Northwest Institute of Mechanical and Electrical Engineering. He holds a BE in Software Engineering from Sichuan University. His research centres on 3D vision for robotic grasping systems, with practical experience in visual perception algorithms for dexterous manipulation tasks.

Zhihang Wang is a third-year Master's student at the Northwest Institute of Mechanical and Electrical Engineering. He earned his BE in Electronic Information Engineering from Chang'an University. His work focuses on signal processing and embedded control for electromechanical systems.

Xue Hui Xing is a Master's candidate at the Northwest Institute of Mechanical and Electrical Engineering. He graduated with a BE in Detection, Guidance, and Control Technology from Nanjing University of Aeronautics and Astronautics. His research involves navigation algorithms and sensor fusion for UAV applications.

Tao Kuai is a final-year Master's student at the Northwest Institute of Mechanical and Electrical Engineering. He holds a BE in Automation from Harbin Engineering University. His research explores adaptive control of hydraulic systems for marine robotics.

1 Introduction

Estimating the 6D pose of an object, involving the estimation of both its 3D rotation and translation from a single-view image, constitutes a major challenge within robotics. This task is pivotal in various applications, including scene interpretation (Nie et al., 2020; Zhang et al., 2021; Peng et al., 2023, 2024), AR/VR (Su et al., 2019), and robotic manipulation (Deng et al., 2020). 6D object pose estimation can be classified into two distinct types, depending on whether a CAD model of the target object is available: instance-level pose estimation (Kehl et al., 2017; Zakharov et al., 2019) and category-level pose estimation (Chen et al., 2020; Wang et al., 2019b). The former is commonly applied in industrial contexts, such as bin picking of predefined objects, whereas the latter pertains to broader applications that demand more sophisticated

capabilities to accommodate untrained objects within specified categories.

Learning-based approaches have dominated the recent progress in 6D object pose estimation (Chen et al., 2020; Tian et al., 2020; Zakharov et al., 2019). Various methods have been proposed to tackle this problem. Most of these methods employ learned or manually designed regularisation networks to predict the pose, and utilise either the raw point clouds or pose consistency to better handle variations in shapes within the same category (Chen et al., 2020, 2021; Chen and Dou, 2021; Zakharov et al., 2019). Despite the progress, existing approaches still face various challenges, including large shape variations, high training consumption, and unseen poses.

In this paper, we advocate using rotation-invariant neural networks to tackle the above problems. Standard 3D convolution networks are not rotation-invariant, resulting in networks that cannot generalise to arbitrary rotations. Rotating the input data would lead to a serious performance drop. Incorporating the 3D convolution networks with rotation-invariant modules would potentially augment the training data and make the networks generalise to unseen rotations without training on all possible shape poses in the SO(3) space.

To do so, we propose a rotation-invariant 3D convolutional network that takes a point cloud representing an object as input and predicts the per-point canonical coordinates. The 6D object pose is then estimated from the per-point canonical coordinates. The proposed network leverages several fundamental geometric features, such as the relative distances and angles within a representative local point set, aggregating the rotation-invariant features. To incorporate these fundamental geometric features with a 3D graph convolutional network (3D-GCN) (Lin et al., 2020), we achieve rotation-invariant feature extraction with strong capability on complex shapes. The proposed network is easy to implement and requires less training data and time, facilitating the application of it to downstream applications. We extract rotation-invariant from point cloud rotation and scaling, which include geometric feature extraction method and 3DGC feature extraction. We further incorporate losses for the three parallel branches, which are used to predict the object's pose, the reconstructed point cloud P, and the relevant parameters for each point regarding the bounding box (Figure 1).

Experiments on a public dataset demonstrate the advantages of the proposed method. It achieves better performance compared to several state-of-the-art baselines. In particular, the proposed method is especially suitable for handling challenging scenarios, including objects with novel poses and severe occlusion. Moreover, ablation studies reveal the significance of the individual components.

Our work makes the following contributions:

- A novel design of rotation-invariant 3D convolutional network.
- A specialised geometric relationship is introduced to encode the rotation-invariant geometric structure.
- Three parallel losses branches are incorporated to predict the object's pose, the reconstructed point cloud P, and the relevant parameters for regarding the bounding box.

2 Related works

2.1 Instance-level 6D pose estimation

Instance-level 6D pose estimation represents a critical domain in computer vision, concentrating on precisely determining the three-dimensional pose of an object instance from image or sensor data (Chen et al., 2021; Kehl et al., 2017; Labbé et al., 2020; Li et al., 2018b; Manhardt et al., 2019, 2018; Xiang et al., 2018; Zhang et al., 2024b). This involves estimating both translation and rotation, typically represented as six degrees of freedom (DOF). During both training and testing, CAD models for the objects of interest are accessible (Kehl et al., 2017; Zakharov et al., 2019). Specifically, when CAD models of the target instances and its corresponding monocular RGB/RGBD image is available, the next step involves estimating the pose $P \in SO(3)$ of the target instance within the image, where P can be decomposed into camera rotation $R \in SO(3)$ and translation $T \in R^3$. Instance-level pose detection techniques can be classified into RGB-based and RGBD-based methods due to different forms of data input. The RGB-based methods typically employ deep learning models to estimate pose-related parameters directly (Di et al., 2021; Hu et al., 2020; Wang et al., 2021a). However, the estimation of 6D pose from a single RGB image is a challenging problem due to the absence of depth information, which introduces significant complexity. The presence of CAD models can help by establishing 2D-3D correspondences between the object model and the input image. Nevertheless, the development of monocular RGBD cameras has improved RGBD-based 6D pose estimation methods, which use depth masks or RGBD images as inputs and leverage point cloud representations to predict object poses (He et al., 2021, 2020; Li et al., 2018a; Wang et al., 2019a). However, instance-level pose estimation often deals with a single or a few targets, and finding a CAD model for the specified object can be challenging, which limits its widespread practical application.

2.2 Category-level pose estimation

This research area, significant in computer vision, focuses on predicting object instance poses by extracting latent information from single-view images, without the need for a pre-existing accurate CAD model of the target (Di et al., 2022; Manhardt et al., 2020; Sahin and Kim, 2018; Wang et al., 2019b, 2021b; Zhang et al., 2024a). Studies in this area are generally categorised into two types. Methods based on correspondence identify the alignment between canonical space coordinates (such as NOCS and NUNOCS) and the target points, and then refine pose and scale through post-processing. In contrast, direct regression methods, such

as DualPoseNet (Lin et al., 2021b) and FS-Net (Chen et al., 2021), focus on extracting pose-sensitive features from the input using advanced network architectures and learning schemes to achieve accurate pose estimation (Deng et al., 2022; Lin et al., 2022; Liu et al., 2022; Zhang et al., 2022). Additionally, category-level methods aim to predict poses of previously unseen objects using techniques like the intrinsic structure adapter and the Umeyama algorithm (Umeyama, 1991), as introduced by Wang et al. (2019b) and Sahin and Kim (2018). Despite its development, category-level pose estimation has recently gained renewed attention due to new deep learning approaches. Many of these algorithms utilise either learned or manually constructed canonical object spaces to determine poses, and they incorporate pose consistency constraints or point cloud-based shape priors to address intra-class shape variations. Despite notable advancements in benchmark performance, these methods still face limitations due to insufficient exploitation of geometric relationships between poses and point clouds.

2.3 Point cloud feature extraction based on 3DGCN

3D graph convolutional networks (3DGCNs) (Lin et al., 2020) are deep learning models designed for performing convolution operations on graph data in three-dimensional space, extending traditional GCNs to accommodate 3D data characteristics. The core idea of 3DGCNs is to learn feature representations of nodes in 3D space. Unlike 2DGCNs, 3DGCNs use additional adjacency matrices to capture the relative positions and relationships of nodes in three dimensions. This enables the model to better understand the topological structure between nodes and infer patterns and features within 3D data more accurately. 3DGCNs are extensively utilised in domains such as computer vision, medical image processing, and molecular structure analysis. They were developed to address the limitations of 2DGCNs in processing 3D data. By incorporating an additional spatial dimension, 3DGCNs significantly enhance the model's capacity to capture relationships between nodes. This ability makes them especially powerful when handling data that exhibits complex spatial structures, such as those encountered in 3D object recognition tasks.

3 Method

The rotation-invariant 3D convolutional network takes a point cloud representing an object as input and predicts the per-point canonical coordinates. The network structure is as depicted in Figure 1, where firstly, an existing object detector, such as mask-RCNN (He et al., 2017), is employed to segment the objects of interest from the depth map. Subsequently, back-projection techniques are used to derive the corresponding point cloud of these objects, which is then utilised as the input for the proposed network. Similar to 3DGCN (Lin et al., 2020), we extract rotation-invariant from point cloud rotation and scaling, which include geometric feature extraction method (Section 3.1) and 3DGC feature extraction (Section 3.2). We further incorporate losses for the three parallel branches(Section 3.3), which are used to predict the object's pose {R, g, t}, the reconstructed point cloud P, and the relevant parameters for each point regarding the bounding box.

The rotation-invariant 3D convolutional network takes a point cloud representing an object as input and predicts per-point canonical coordinates. The network structure is depicted as Figure 1. Initially, an existing object detector, such as mask-RCNN (He et al., 2017), is employed to segment the objects of interest from the depth map. Subsequently, back-projection techniques are used to derive the corresponding point cloud of these objects, which is then used as input for the proposed network. Similar to 3DGCN (Lin et al., 2020), we derive rotation-invariant features by applying transformations such as rotations and scaling to the point clouds. This process encompasses both a geometric feature extraction technique (Section 3.1) and the extraction of 3DGC features (Section 3.2). We further incorporate losses for three parallel branches (Section 3.3), which predict the instance's pose {R, g, t}, the reconstructed point cloud P, and relevant parameters for each point concerning the bounding box.

3.1 Geometric feature extraction

To enhance the uniqueness of the extracted features, a specialised geometric relationship is introduced to encode the rotation-invariant geometric structure of the points. The core concept of this geometric relationship involves measuring the relationship using the distance between two points and the angle of a point to a plane. Specifically, for a central point p_i, the k-nearest neighbours P_1 are identified based on the Euclidean distance of the features using the k-nearest neighbour algorithm. After performing k-means clustering on these k_1 neighbours, the k_2 closest points to the central point are selected as the feature domain P_2. In the neighbourhood P_2 of the central point p_i, each neighbouring point p_j forms k_2 point pairs with the central point. The geometric relationship of each point pair is described as follows:

1 Point-to-point distance: the Euclidean distance between two points is given by $d_{i,j} = |p_i - p_j|$.

2 Angle between the centre point normal vector and the point pair vector: two nearest neighbours p_i^1 and p_i^2 are selected to form a local plane around p_i. The normal vector n_i of the local plane is computed using the cross product. The angle between the point and the plane is calculated as $a_{i,j} = \theta(n_i, p_j - p_i)$.

3 Angle between the neighbour point normal vector and the point pair vector: similarly, the angle between the normal vector of the neighbouring point p_j and the point pair vector is computed. This is given by $a_{j,i} = \theta(n_j, p_j - p_i)$, where n_j is the local normal vector of p_j, computed using the cross product.

Finally, the geometric feature of each point pair is derived by aggregating the distances between points and the angles between each point and the plane. This is represented as:

$$f_{i,j} = [d_{i,j}, a_{i,j}, a_{j,i}] \tag{1}$$

and [-,-] signifies the operation of concatenation. The feature corresponding to the central point is derived by computing the average of the geometric features from all point pairs within the set of k-nearest neighbours:

$$f_1(p_i) = \frac{1}{k}\sum\nolimits_{1\le j\le k} f_{i,j} \tag{2}$$

3.2 3DGC feature extraction

A 3D point cloud object with N points is represented as $P = \{p_i | \ i = 1, 2, \ldots, N\}$, where $p_i \in R^3$.

To characterise the features corresponding to each point in 3D-GCN (Lin et al., 2020), we use $f_2(p) \in R^D$ to denote the corresponding D-dimensional feature vector. In the context of 3D-GCN (Lin et al., 2020), the feature vector associated with each point p_i is denoted $f_2(p) \in R^D$, where D represents the dimensionality of the feature space. To capture the local geometric information of each point p_i, a 3D receptive field for p_i is determined by a set of K neighbouring points. Specifically, $R_i^K = \{p_i, p_k \mid \forall p_k \in N(p_i, K)\}$, where $N(p_i, K)$ denotes the K nearest neighbours of p_i based on Euclidean distance. For the variable convolution kernel K^S in 3DGCN, $K^S = \{k^C, k^1, k^2, \ldots, k^S\}$, where S represents the number of supports in the kernel, $k^C = \{0, 0, 0\}$ denotes the centre of the kernel, and k^1 through k^S represent the corresponding supports (Lin et al., 2020). A weight vector $w(k) \in R^D$ is defined for each kernel point. Given the corresponding convolution kernel and the receptive field, the feature convolution operation can be defined as:

$$\begin{aligned} Conv(R_i^K, K^S) = &< f(p_i), w(k_C) > \\ &+ g(A) = f_2 \end{aligned} \tag{3}$$

where $\langle\cdot\rangle$ denotes the inner product operation and $A = \left\{ \dfrac{< f^2(p_i), w(k_s) >< d_{i,k}, k_s >}{\|d_{i,k}\|\|k_s\|} \middle| \forall k \in (1, K), \forall s \in (1, S) \right\}$. For further details, please refer to Lin et al. (2022).

In summary, the feature F of point P can ultimately be represented as: $F = [f_1, f_2]$.

3.3 Loss functions

We incorporate losses for the three parallel branches, comprising the basic pose loss for predicting the object's pose {R, T, S}, which represents {rotation, translation, size} respectively, the point cloud reconstruction loss for predicting the reconstructed point cloud P (Di et al., 2021; Li et al., 2018b; Wang et al., 2021a), and the bounding Box-Pose Loss for predicting the parameters of the bounding boxes.

3.3.1 Basic pose loss

For the rotation matrix R, we decompose the ground truth rotation matrix R_{gt} into the normal vectors of the bounding box, i.e., $R_{gt} = [r_x^{gt}, r_y^{gt}, r_z^{gt}]$. We only need to estimate the first two parameters of the rotation matrix, r_x and r_y, and define the corresponding loss term as:

$$\mathcal{L}_{rot}^{Basic} = \left\|r_x^{gt} - r_x\right\|_1 + \left\|r_y^{gt} - r_y\right\|_1 \tag{4}$$

To estimate the translation t, the mean of the point cloud, denoted as M_P, and the residual translation t_* are first computed. This relationship is expressed as $t = t_* + M_P$. Consequently, the translation loss can be defined as follows:

$$\mathcal{L}_{trans}^{Basic} = \left\|t^{gt} - t\right\|_1 \tag{5}$$

For size s, we predict the residual size s_*, which represents the difference between the actual prediction ss and the pre-computed category average size C_m, i.e., $s = s_* + C_m$. The corresponding loss term is defined as:

$$\mathcal{L}_{size}^{Basic} = \left\|s^{gt} - s\right\|_1 \tag{6}$$

In summary, the overall loss function is:

$$\mathcal{L}^{Basic} = \lambda_{rot}\mathcal{L}_{rot}^{Basic} + \lambda_{trans}\mathcal{L}_{trans}^{Basic} + \lambda_{size}\mathcal{L}_{size}^{Basic} \tag{7}$$

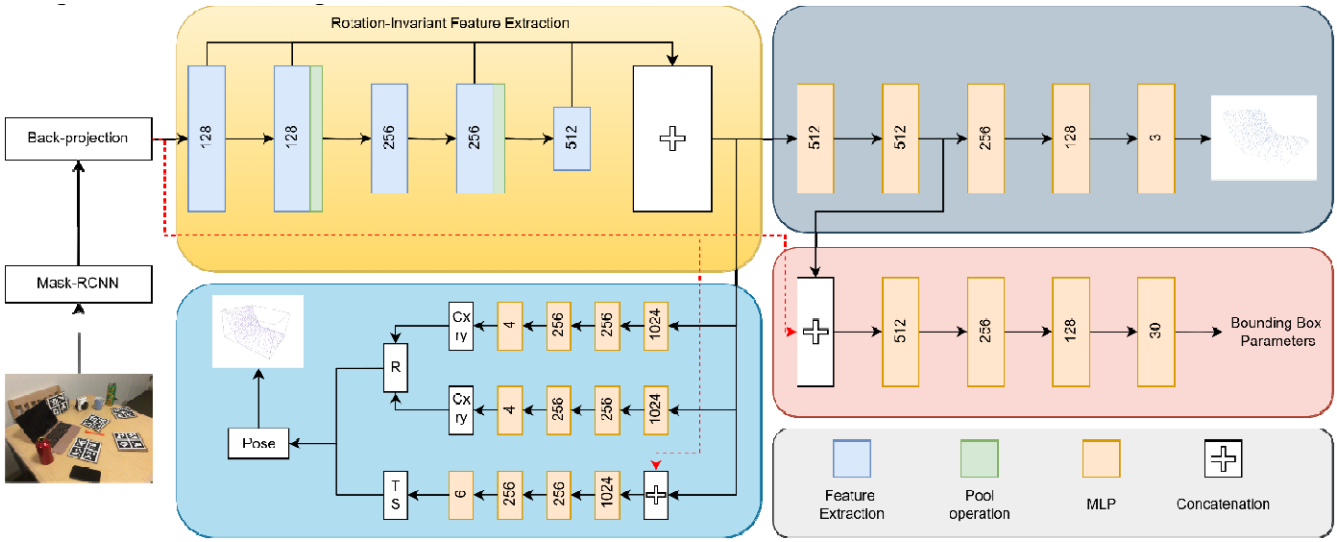
3.3.2 Point cloud reconstruction loss

Although most class-level methods derive pose estimates from uniformly sampled point clouds, they frequently overlook the impact of the relative positions of points within the cloud, thereby missing valuable information. In contrast, these methods depend on carefully crafted loss terms to infer high-dimensional pose information. By transforming the point cloud to a canonical view, it becomes simpler to compute the relative positions of peripheral points. Analogous to the loss functions used in instance-level pose estimation, we define the following loss function:

$$\mathcal{L}^{PC} = \sum\nolimits_{p\in P}\left\|R^T(p - t) - p^c\right\|_1 \tag{8}$$

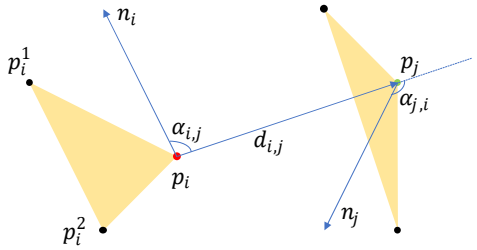
where R, t denote the predicted pose parameters, p represents the predicted point cloud, and p^c refers to the ground truth in the canonical view.

Figure 1 Architecture of our network (see online version for colours)

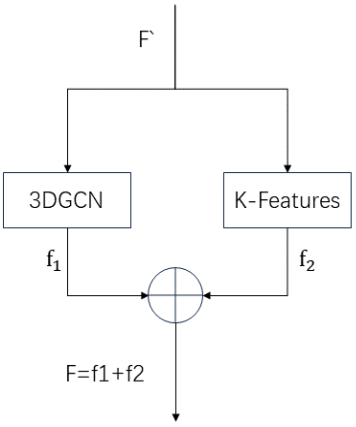


Notes: In the initial stage, mask-RCNN segments the target instance from the depth image, providing input for the subsequent algorithm. A back-projection algorithm then generates 512 points from the target depth maps, which are used as inputs to the algorithm. The target point cloud features are extracted using the proposed rotationally invariant 3D convolutional layer (highlighted in yellow), which serves as input to the next convolutional network for further feature extraction. The first branch (depicted in blue) outputs the rotation parameters $\{r_x, r_y, C_x, C_y\}$, the translation vector T, and the scale S. From these, the predicted pose {R, T, S} is derived. The subsequent processing, indicated by the pink boxes, follows the approach described in Di et al. (2022). Here, parameters for each point in the target point cloud – such as distance, orientation, and confidence level relative to the bounding box – are predicted, serving as potential geometric constraints in loss computation.

Figure 2 (a) Geometric feature representation and (b) final feature representation of point p (see online version for colours)



(a)



(b)

3.3.3 Bounding Box-Pose loss

We follow the loss terms for bounding boxes as proposed in GPV-Pose (Di et al., 2021). For each observation point p_j, estimate the confidence c_i^j, distance d_i^j, and direction n_i^j for each of the six bounding box surfaces, where $i \in \mathcal{B}$ and $\mathcal{B} = \{y\pm, x\pm, z\pm\}$. Taking the front bounding box plane $x+$ as an example, the corresponding point p_j' on the front plane $x+$ is given by:

$$p_j' = p_j + n_{x+}^j d_{x+}^j \tag{9}$$

The plane parameters are characterised by the distance D_{x+} and the normal vector N_{x+} from the origin to the plane. These parameters can be inferred from the weights assigned to each point in the point cloud, as well as the associated confidence levels. Since we can obtain predictions for n_i^j and d_i^j, we directly supervise them using the L1 loss. The following loss term can be defined:

$$\mathcal{L}_{pc}^{BB} = \sum_{p_j \in P} \sum_{i \in \mathcal{B}} \left\| c_i^j - \exp\left(\frac{\left| d_i^j n_i^j - f_j^i(p_j) r_i^{gt} \right|}{-a} \right) \right\|_1 \tag{10}$$

where a is a constant, and $f_j^i(p_j)$ represents the true distance from p_j to the plane i in B. Considering additionally the front plane $x+$, we have:

$$f_j^{x+}(p_j) = \frac{s_{[x+]}^{gt}}{2} - R_{gt}^T (p_j - t^{gt}) \tag{11}$$

where $s_{[x+]}$ is the size along the $x+$ direction, and R_{gt} and t^{gt} denote the translation and ground truth rotation, respectively.

As previously mentioned, the rotation matrix R is decomposed into three columns $R = [r_x, r_y, r_z]$. However, only r_x and r_y need to be predicted along with their associated confidence, as they completely describe the corresponding 3D rotation, i.e., $r_z = r_x \times r_y$. Additionally, r_x and r_y correspond to the normal vectors of the bbox planes. Therefore, conditional on the predicted bounding box parameters $\{N_i, D_i\}$, $i \in B$, the following loss function can be defined:

$$\mathcal{L}_{(R,t,s)}^{BB} = \lambda_R^{BB} \mathcal{L}_R^{BB} + \lambda_t^{BB} \mathcal{L}_t^{BB} + \lambda_s^{BB} \mathcal{L}_s^{BB} \tag{12}$$

where each loss term is defined as follows:

$$\mathcal{L}_R^{BB} = \sum\nolimits_{i \in \mathcal{B}} \left\| r_i' - N_i \right\|_1 \tag{13}$$

$$\mathcal{L}_t^{BB} = \sum_{i \in \{x,y,z\}} \left\| \left| N_{i+}^T t - D_{i+} \right| - \left| N_{i-}^T t - D_{i-} \right| \right\| \tag{14}$$

$$\mathcal{L}_{(s)}^{BB} = \sum\nolimits_{i \in |\mathcal{B}|} \left\| s_i / 2 - \left| N_i^T t - D_i \right| \right\|_1 \tag{15}$$

By leveraging the normals of the bounding box planes to predict rotations, which corresponds to utilising the first two columns of the rotation matrix R, we address the inherent discontinuities within the SO(3) group, thus enhancing the learning process. Specifically, rotations are estimated based on the normals of two planes from the 3D bounding box. To overcome the challenges of accurately recovering distinct normals and to increase the robustness of the final rotation prediction, we assess the confidence associated with each normal. The goal is to ensure that normals with higher confidence values correspond to more precise rotation predictions. Accordingly, we define:

$$\mathcal{L}_{rc}^{BB} = \sum_{i \in x,y} \left\| c_i - \exp\left(-b \left| r_i - r_i^{gt} \right|^2 \right) \right\|_1 \tag{16}$$

where b is a constant, r_i^{gt} represents the ground truth plane normal, and $\|\cdot\|$ represents the L_1 loss function.

In summary, we define the boundary box loss as:

$$\mathcal{L}^{BB} = \mathcal{L}_{pc}^{BB} + \mathcal{L}_{(R,t,s)}^{BB} + \mathcal{L}_{rc}^{BB} \tag{17}$$

3.4 Implementation details

We provide implementation details of the training and inference. All experiments were based on GTX 3060. The object detector Mask-RCNN was employed for object segmentation, and a reprojection algorithm was used to uniformly sample 512 points from depth maps as input to the network. A range of data augmentation techniques was employed, encompassing random rotation, translation, scaling, and the addition of noise. The Ranger optimiser was employed with a total of 150 training epochs, a learning rate of 0.001, and a batch size of 16. A cosine schedule was employed to reduce the learning rate during 50% of the training phase. The total loss function is as follows:

$$\mathcal{L} = \lambda_{Basic} \mathcal{L}^{Basic} + \lambda_{PC} \mathcal{L}^{PC} + \lambda_{BB} \mathcal{L}^{BB} \tag{18}$$

The parameters for all loss components remained constant during experimentation, with $\{k_1, k_2\} = \{20.5\}$, $\{a, b\} = \{1/303.5, 13.7\}$, and $\{\lambda_{rot}, \lambda_{trans}, \lambda_{size}, \lambda_{Basic}, \lambda_{BB}, \lambda_{PC}\} = \{1.0, 1.0, 1.0, 8.0, 1.0, 1.0\}$.

4 Results and evaluation

4.1 Dataset

Our network is trained and evaluated using the NOCS-REAL275 and NOCS-CAMERA25 datasets (Wang et al., 2019b). These pioneering datasets are specifically designed for category-level 6D object pose estimation and cover six categories: bottles, bowls, cameras, cans, laptops, and mugs. The CAMERA25 dataset comprises 300,000 synthesised RGB-D images (Li et al., 2019). Conversely, the NOCS-REAL275 dataset includes 2,750 test images and 4,300 training images, all of which are RGB-D images captured from real-world scenes.

Table 1 Quantitative results on the NOCS-REAL275 dataset

Method	*IoU_{25}*	*IoU_{50}*	*IoU_{75}*	*5°2cm*	*5°5cm*	*10°5cm*	*10°10cm*
NOCS (Wang et al., 2019b)	84.9	80.5	30.1	7.2	10	25.2	26.7
CASS (Chen et al., 2020)	84.2	77.7	-	-	23.5	58	58.3
SPD (Tian et al., 2020)	83.4	77.3	53.2	19.3	21.4	54.1	-
CR-Net (Zhang et al., 2021)	83.4	79.3	55.9	27.8	34.3	60.8	-
DO-Net (Lin et al., 2021a)	-	80.4	63.7	24.1	34.8	67.4	-
FS-Net(Chen et al., 2021)	*95.1*	*92.2*	63.5	-	28.2	60.8	64.6
GPV* (Di et al., 2022)	84.2	81.3	53.8	18.5	27.8	58.2	67.4
Ours	84.2	82.3	*68.1*	*29.2*	*36.6*	*72.5*	*74.5-*

Notes: Italic indicates the best of the results, underlining indicates the second best result. The locally replicated GPV-Pose (denoted as GPV*) is used as a key reference.

Table 2 Whether rotation-invariant features participate in convolution

Method	*IoU_{25}*	*IoU_{50}*	*IoU_{75}*	*5°2cm*	*5°5cm*	*10°5cm*
A1	84.1	80.9	57.5	21.3	31.8	65.1
Ours	*84.2*	*82.3*	*68.1*	*29.2*	*36.6*	*72.5*

Table 3 Impact of different points on the network

Total points	*Neighbour points*	*IoU_{25}*	*IoU_{50}*	*IoU_{75}*	*5°2cm*	*5°5cm*	*10°5cm*	*Chamfer distance*
512	5	83.5	81.7	58.9	18.9	25.4	60.7	0.044569
512	10	83.7	79.6	56.3	19.6	25.7	57.0	0.04376
1,024	20	84.1	81.6	*68.7*	26.6	34.9	52.8	0.043091
Ours	20	*84.2*	*82.3*	68.1	*29.2*	*36.6*	*72.5*	*0.038832*

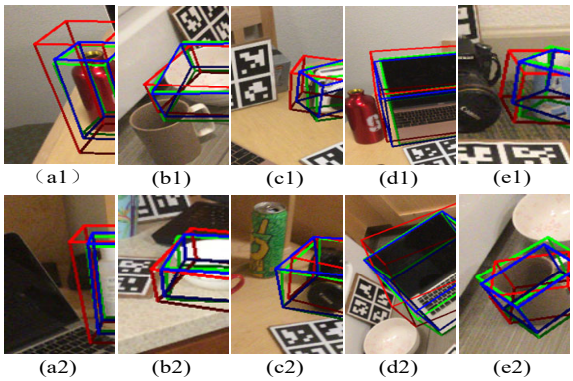
4.2 Evaluation metrics

For evaluating category-level 6D pose estimation, we employed standard performance metrics on the NOCS-REAL275 dataset. IoUX, which refers to the intersection over union (IoU), measures the accuracy of 3D object detection by assessing overlap at various thresholds. A prediction is considered valid if the volume overlap between the predicted and ground-truth 3D bounding boxes exceeds a specified percentage. Following the guidelines outlined in references (Chen and Dou, 2021; Lin et al., 2019b; Wang et al., 2019b), we report evaluation thresholds of 25%, 50%, and 75% to jointly assess rotation, translation, and object size. The notation n°m cm is used to denote errors in rotation and translation pose estimation, where the results are accepted if the rotational error is less than n° and the translational error is under m cm. Specifically, we use the following metrics: 5°2 cm, 5°5 cm, 10°5 cm, and 10°10 cm. Additionally, to evaluate the similarity between the ground-truth and reconstructed point clouds, Chamfer distance is utilised as a metric.

4.3 Performance on NOCS-REAL275

Table 1 presents a comparison of our algorithm's prediction results with those from alternative methods on the NOCS-REAL275 dataset. Noteworthy findings include that our approach exhibits superior performance in position estimation, outperforming previous methods on the 5°2 cm, 5°5 cm, 10°5 cm, and 10°10 cm metrics, thereby confirming its effectiveness in predicting object pose. However, there is a noticeable gap in object detection performance compared to FS-Net, with the IoU25 and IoU50 metrics indicating areas for improvement and suggesting that further enhancements are needed in object detection.

Figure 3 Visual comparison of the estimated pose (see online version for colours)



Note: Green bbox represents the ground truth, the red box depicts the predictions from the replicated GPV-Pose method, and the blue box indicates the results obtained using our algorithm.

Figure 3 presents a qualitative comparison of our method on REAL275. Our method accurately forecasts the displacement and rotation of the target instance, as illustrated in Figures 3 (a1), (c1), (e1), among others, whereas the GPV method frequently fails to achieve comparable results in these scenarios.

4.4 Ablation study

Table 2 shows an ablation study quantifying the contribution of the rotation-invariant feature extraction network to our method. Table 3 examines the impact of different network constructs on overall performance. All experiments were performed using the NOCS-REAL275 dataset for both training and validation.

- *Network structure:* we investigated the effect of incorporating rotation-invariant features on overall network performance. By excluding these features from subsequent convolution operations, we obtained result A1. The findings indicate that including rotation-invariant features significantly enhances network performance, confirming their beneficial role in pose estimation algorithms.
- *Number of points:* as mentioned in Section 3, rotation-invariant features are computed from neighbouring points of target points, which may affect network performance. Multiple experiments with varying numbers of neighbouring points were conducted to retrain our network, as shown in Table 3. We observed that a relatively small number of neighbouring points significantly degrades network performance, likely due to the loss of high-dimensional information from the point cloud, leading to inaccurate predictions. Additionally, while increasing the total point cloud sample size correlated with improved final prediction results, excessively large point clouds may negatively impact algorithmic performance.

5 Conclusions

This paper introduces a technique for extracting rotationally invariant features, which are used in a category-level bit-position estimation algorithm. The integration of these features substantially enhances the precision of pose and size estimation and exhibits consistent performance across different object rotations, yielding promising outcomes. Additionally, the method is capable of reconstructing corresponding point clouds. Future research will aim to advance the network's capabilities to enhance performance with more intricate objects and to integrate pose estimation with point cloud reconstruction.

References

Chen, D., Li, J., Wang, Z. and Xu, K. (2020) 'Learning canonical shape space for category-level 6D object pose and size estimation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.11973–11982.

Chen, K. and Dou, Q. (2021) 'Sgpa: structure-guided prior adaptation for category-level 6d object pose estimation', in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp.2773–2782.

Chen, W., Jia, X., Chang, H.J., Duan, J., Linlin, S. and Leonardis, A. (2021) '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 (CVPR)*, pp.1581–1590.

Deng, X., Geng, J., Bretl, T., Xiang, Y. and Fox, D. (2022) 'Icaps: iterative category-level object pose and shape estimation', in *IEEE Robotics and Automation Letters*, Vol. 7, No. 2, pp.1784–1791.

Deng, X., Xiang, Y., Mousavian, A., Eppner, C., Bretl, T. and Fox, D. (2020) 'Self-supervised 6d object pose estimation for robot manipulation', in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, pp.3665–3671.

Di, Y., Manhardt, F., Wang, G., Ji, X., Navab, N. and Tombari, F. (2021) 'So-pose: exploiting selfocclusion for direct 6D pose estimation', in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp.12396–12405.

Di, Y., Zhang, R., Lou, Z., Manhardt, F., Ji, X., Navab, N. and Tombari, F. (2022) '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 (CVPR)*, pp.6781–6791.

He, K., Gkioxari, G., Dollár, P. and Girshick, R. (2017) 'Mask R-CNN', in *Proceedings of the IEEE International Conference on Computer Vision*, pp.2961–2969.

He, Y., Huang, H., Fan, H., Chen, Q. and Sun, J. (2021) 'Ffb6d: a full flow bidirectional fusion network for 6d pose estimation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3003–3013.

He, Y., Sun, W., Huang, H., Liu, J., Fan, H. and Sun, J. (2020) 'PVN3D: a deep point-wise 3D keypoints voting network for 6D of pose estimation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.11632–11641.

Hu, Y., Fua, P., Wang, W. and Salzmann, M. (2020) 'Single-stage 6d object pose estimation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.2930–2939.

Kehl, W., Manhardt, F., Tombari, F., Ilic, S. and Navab, N. (2017) 'SSD-6D: making RGB-based 3D detection and 6d pose estimation great again', in *The IEEE International Conference on Computer Vision (ICCV)*, pp.1530–1538.

Labbé, Y., Carpentier, J., Aubry, M. and Sivic, J. (2020) 'Cosypose: consistent multi-view multi-object 6D pose estimation', in *European Conference on Computer Vision*, pp.574–591.

Li, C., Bai, J. and Hager, G.D. (2018a) 'A unified framework for multi-view multi-class object pose estimation', in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp.254–269.

Li, Y., Wang, G., Ji, X., Xiang, Y. and Fox, D. (2018b) 'Deepim: deep iterative matching for 6D pose estimation', in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp.638–698.

Li, Z., Wang, G. and Ji, X. (2019) 'CDPN: coordinates-based disentangled pose network for realtime RGB-based 6-DoF object pose estimation', in *The IEEE International Conference on Computer Vision (ICCV)*, pp.7678–7687.

Lin, H., Liu, Z., Cheang, C., Zhang, L., Fu, Y. and Xue, X. (2021a) *Donet: Learning Category Level 6D Object Pose and Size Estimation from Depth Observation*, arXiv preprint arXiv:2106.14193.

Lin, J., Wei, Z., Li, Z., Xu, S., Jia, K. and Li, Y. (2021b) '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 (ICCV)*, pp.3560–3569.

Lin, J., Wei, Z., Ding, C. and Jia, K. (2022) 'Category-level 6D object pose and size estimation using selfsupervised deep prior deformation networks', in *Computer Vision-ECCV 2022: 17th European Conference*, Tel Aviv, Israel, 23–27 October, Proceedings, Part IX, pp.19–34.

Lin, Z-H., Huang, S-Y. and Wang, Y-C.F. (2020) 'Convolution in the cloud: learning deformable kernels in 3d graph convolution networks for point cloud analysis', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1800–1809.

Liu, X., Wang, G., Li, Y. and Ji, X. (2022) 'Catre: iterative point clouds alignment for category-level object pose refinement', in *Computer Vision-ECCV 2022: 17th European Conference*, Tel Aviv, Israel, 23–27 October, Proceedings, Part II, pp.499–516.

Manhardt, F., Arroyo, D.M., Rupprecht, C., Busam, B., Birdal, T., Navab, N. and Tombari, F. (2019) 'Explaining the ambiguity of object detection and 6d pose from visual data', in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp.6841–6850.

Manhardt, F., Kehl, W., Navab, N. and Tombari, F. (2018) 'Deep model-based 6D pose refinement in RGB', in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp.883–849.

Manhardt, F., Wang, G., Busam, B., Nickel, M., Meier, S., Minciullo, L., Ji, X. and Navab, N. (2020) *Cps++: Improving Class-Level 6D Pose and Shape Estimation from Monocular Images with Selfsupervised Learning*, arXiv preprint arXiv:2003.05848v3.

Nie, Y., Han, X., Guo, S., Zheng, Y., Chang, J. and Zhang, J.J. (2020) 'Totaled understanding: joint layout, object pose and mesh reconstruction for indoor scenes from a single image', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.55–64.

Peng, Q., Ce, Z. and Chen, C. (2023) 'Source-free domain adaptive human pose estimation', *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp.4803–4813.

Peng, Q., Ce, Z. and Chen, C. (2024) 'A dual-augmentor framework for domain generalization in 3D human pose estimation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sahin, C. and Kim, T-K. (2018) 'Category-level 6d object pose recovery in depth images', in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp.665–681.

Su, Y., Rambach, J., Minaskan, N., Lesur, P., Pagani, A. and Stricker, D. (2019) 'Deep multi-state object pose estimation for augmented reality assembly', in *2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp.222–227.

Tian, M., Ang, M.H. and Lee, G.H. (2020) 'Shape prior deformation for categorical 6D object pose and size estimation', in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp.530–546.

Umeyama, S. (1991) 'Least-squares estimation of transformation parameters between two point patterns', in *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 13, No. 4, pp.376–380.

Wang, C., Xu, D., Zhu, Y., Mánguez-Martín, R., Lu, C., Fei-Fei, L. and Savarese, S. (2019a) 'DenseFusion: 6D object pose estimation by iterative dense fusion', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3343–3352.

Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S. and Guibas, L.J. (2019b) '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 (CVPR)*, pp.2642–2651.

Wang, G., Manhardt, F., Tombari, F. and Ji, X. (2021a) 'GDR-net: geometry-guided direct regression network for monocular 6D object pose estimation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.16606–16616.

Wang, J., Chen, K. and Dou, Q. (2021b) 'Category-level 6D object pose estimation via cascaded relation and recurrent reconstruction networks', in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.4807–4814.

Xiang, Y., Schmidt, T., Narayanan, V. and Fox, D. (2018) 'PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes', in *Robotics: Science and Systems XIV*.

Zakharov, S., Shugurov, I. and Ilic, S. (2019) 'Dpod: dense 6D pose object detector in RGB images', in *The IEEE International Conference on Computer Vision (ICCV)*.

Zhang, C., Cui, Z., Zhang, Y., Zeng, B., Pollefeys, M. and Liu, S. (2021) 'Holistic 3D scene understanding from a single image with implicit representation', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.8833–8842.

Zhang, R., Di, Y., Lou, Z., Manhardt, F., Tombari, F. and Ji, X. (2022) 'RBP-pose: residual bounding box projection for category-level pose estimation', in *Computer Vision-ECCV 2022: 17th European Conference*, Tel Aviv, Israel, 23–27 October, Proceedings, Part I, pp.655–672.

Zhang, C., Ling, Y. and Lu, M. (2024a) 'Category-level object detection, pose estimation and reconstruction from stereo images', in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp.332–349.

Zhang, R., Huang, Z. and Wang, G. (2024b) 'LaPose: Laplacian mixture shape modeling for RGB-based category-level object pose estimation', in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp.467–484.