

# VI-Net: Boosting Category-level 6D Object Pose Estimation via Learning Decoupled Rotations on the Spherical Representations

Jiehong Lin<sup>1,2</sup>Zewei Wei<sup>1</sup>Yabin Zhang<sup>3</sup>Kui Jia<sup>1\*</sup><sup>1</sup>South China University of Technology<sup>2</sup>DexForce Technology Co. Ltd.<sup>3</sup>Hong Kong Polytechnic University<https://github.com/JiehongLin/VI-Net>

## Abstract

Rotation estimation of high precision from an RGB-D object observation is a huge challenge in 6D object pose estimation, due to the difficulty of learning in the non-linear space of  $SO(3)$ . In this paper, we propose a novel rotation estimation network, termed as **VI-Net**, to make the task easier by decoupling the rotation as the combination of a viewpoint rotation and an in-plane rotation. More specifically, VI-Net bases the feature learning on the sphere with two individual branches for the estimates of two factorized rotations, where a V-Branch is employed to learn the viewpoint rotation via binary classification on the spherical signals, while another I-Branch is used to estimate the in-plane rotation by transforming the signals to view from the zenith direction. To process the spherical signals, a Spherical Feature Pyramid Network is constructed based on a novel design of SPAtial Spherical Convolution (SPA-SConv), which settles the boundary problem of spherical signals via feature padding and realizes viewpoint-equivariant feature extraction by symmetric convolutional operations. We apply the proposed VI-Net to the challenging task of category-level 6D object pose estimation for predicting the poses of unknown objects without available CAD models; experiments on the benchmarking datasets confirm the efficacy of our method, which outperforms the existing ones with a large margin in the regime of high precision.

## 1. Introduction

The task of 6D object pose estimation [35, 30, 13, 16, 31] from an RGB-D object observation is to learn the transformation from the canonical object system to the camera system, represented by a 3D rotation  $\mathbf{R} \in SO(3)$  and a 3D translation  $\mathbf{t} \in \mathbb{R}^3$ . It is demanded in many real-world applications, such as robotic grasping [34, 24], augmented reality [1], and autonomous driving [15, 33, 6, 7].

\*Correspondence to Kui Jia &lt;kuijia@gmail.com&gt;

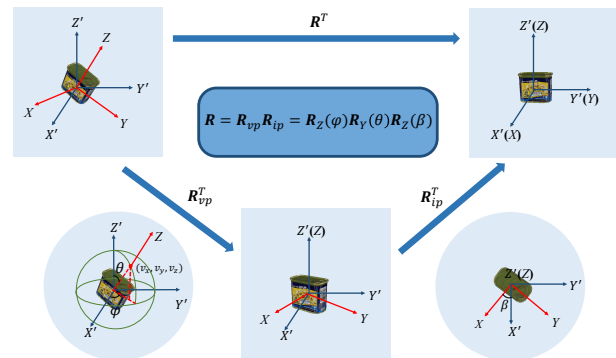


Figure 1. An illustration of the factorization of rotation  $\mathbf{R}$  into a viewpoint (out-of-plane) rotation  $\mathbf{R}_{vp}$  and an in-plane rotation  $\mathbf{R}_{ip}$  (around  $Z$ -axis). Notations are explained in Sec. 3.

For the 6D object pose, translation is easier to be estimated, *e.g.*, initialized as the centroid of the object points, while learning rotation in the whole space of  $SO(3)$  is more challenging due to the non-linearity of  $SO(3)$ . Things become more complicated when estimating rotations of unknown objects without available CAD models. Taking the task of category-level 6D object pose estimation as an example, one typical way [31, 28, 32, 4] is to learn the correspondence between the camera system and the canonical one to solve the poses via Umeyama algorithm[29], with the surrogate objectives of canonical coordinates rather than the true ones of object poses. Another strategy is to learn in the  $SO(3)$  space with rotation-aware features extracted from specially designed encoders, *e.g.*, 3D Graph Convolutional (3DGC) autoencoder [5], Spherical CNN (in the spectral domain) [9, 20], and 3D Steerable CNN [18], whose results, however, are less satisfactory than those based on correspondence learning.

Recently, OVE6D [3] narrows the learning space of rotation  $\mathbf{R}$  by factorizing it, as shown in Fig. 1, into a viewpoint rotation  $\mathbf{R}_{vp}$  and an in-plane rotation  $\mathbf{R}_{ip}$  around the canonical zenith direction (the positive direction of  $Z$ -axis in Fig. 1), and renders images of object CAD models with various of discrete rotations to construct the codebook for viewpoint

rotation retrieval. However, object CAD models are often not available in real-world applications for rendering, *e.g.*, on the focused task of category-level 6D object pose estimation. In this paper, we correlate the factorized rotations to the sphere, and propose a new method of **VI-Net**, which employs two carefully designed heads of **V-Branch** and **I-Branch** to estimate  $\mathbf{R}_{vp}$  and  $\mathbf{R}_{ip}$ , respectively, based on spherical representations with no use of object CAD models; Fig. 2 gives an illustration of VI-Net.

Specifically, given an object observation (centered at the origin with known translation), VI-Net processes its point-wise attributes as the signals assigned on the sphere, represented as a 2D spherical map, and constructs a Spherical Feature Pyramid Network (FPN) to extract high-level spherical features along hierarchy; on top of the Spherical FPN, two individual heads of V-Branch and I-Branch are employed to learn the viewpoint rotation  $\mathbf{R}_{vp}$  and the in-plane rotation  $\mathbf{R}_{ip}$ , respectively. For V-Branch,  $\mathbf{R}_{vp}$  is generated by learning the intersection point of the canonical zenith direction with the unit sphere, which is recognized via binary classification on the high-level spherical feature map. For I-Branch, VI-Net rotates the object through the transformation of its spherical feature map with  $\mathbf{R}_{vp}$ , and could thus estimate  $\mathbf{R}_{ip}$  from the perspective of the canonical zenith direction. Finally, the rotation  $\mathbf{R}$  is given as the multiplication of  $\mathbf{R}_{vp}$  and  $\mathbf{R}_{ip}$ .

Another problem is how to build the Spherical FPN. For convenience, the spherical signals are processed in the form of representations with regular 2D spatial sizes, which yet results in the boundary problem. We thus propose in this paper a novel design of *SPAtial Spherical Convolution*, termed as **SPA-SConv**, which settles the above problem via simple feature padding, and furthermore, extracts viewpoint-equivariant features through symmetric convolutional operations to support the feature transformation in I-Branch. Our SPA-SConv could be flexibly adapted to the existing convolutional architectures by replacing the convolutions, and we thus construct the spherical version of FPN [21].

To confirm the efficacy of our VI-Net on rotation estimation, we apply it to the challenging task of category-level 6D object pose estimation. Experiments are conducted on the benchmarking REAL275 dataset [31], which show that our method outperforms the existing ones with a large margin in the regime of high precision, *e.g.*, an improvement of 4.0% on the metric of  $5^\circ 2\text{cm}$  over the state-of-the-art method of DPDN [19]. Ablation studies also confirm the efficacy of our novel designs.

## 2. Related Work

### 2.1. Category-level 6D Object Pose Estimation

The task of category-level 6D object pose estimation is introduced in [31] to estimate the 6D pose and size of un-

seen objects within certain categories, and could be categorized into two groups, including those based on correspondence learning and those based on direct regression.

**Methods Based on Correspondence Learning** This group of methods first learn the points in the canonical space to align with the observations, and then obtain the object poses from the correspondence via solving of Umeyama algorithm [29]. NOCS [31] regresses the canonical point coordinates directly from an RGB image via MaskRCNN [11], while SPD [28] learns to deform the categorical shape priors to improve the qualities of canonical points. The latter realizes better correspondence by taking the advantage of shape priors, and attracts the followers to embark on this path. For example, CR-Net [32] models the relationship between the observations and the shape priors, and iteratively refines the shape deformations, while SGPA [4] proposes a structured-guided prior adaptation scheme via transformers.

**Methods Based on Direct Regression** Most methods of this group directly regress the object poses from the rotation-aware features via specially designed network architectures. DualPoseNet [20] employs the  $SO(3)$ -equivariant spherical convolutions for the extraction of the global pose-sensitive features, on top of which two pose decoders with different working mechanisms are designed for complementary supervision. 3D Steerable Convolutions are proposed in [18] for  $SE(3)$ -equivariant feature learning, and successfully applied to this task. Both FS-Net [5] and GPV-Pose [8] construct their encoders based on 3D Graph Convolutions (3DGC), from which two perpendicular vectors of rotation are directly regressed; we note that they also decouple the rotation to reduce the learning space, but do not well exploit the properties of the perpendicular vectors for easier learning. Recently, DPDN [19] borrows the idea of correspondence learning to deform the features of shape priors, and builds correspondence in the feature space for direct estimates of object poses.

### 2.2. Convolutions on Spherical Representations

The early works [27, 2, 38] of convolutions, defined on spherical representations, consider less about the properties of equivariance, which, yet, are important for rotation estimation. Then Esteves et al. propose a design of spherical convolution [9] implemented in the spectral domain to learn features with strict  $SO(3)$ -equivariant constraints, which is used in [20] for pose estimation and most related to our SPA-SConv; to distinguish two kinds of convolutions, we term their design as **SPE-SConv**. In our VI-Net,  $SO(3)$ -equivariance of SPE-SConv makes the individual feature points not discriminative to support the binary classification for generating the viewpoint rotation, even resulting in gradient exploding; our SPA-SConv relaxes the constraints while preserving viewpoint-equivariance, which could jointly support the binary classification in V-Branch and the feature transformation in I-Branch.

### 3. Correlation on the Sphere and Rotation Decomposition

A rotation  $\mathbf{R} \in SO(3)$  could be defined as the transformation between two Euclidean coordinate systems with the same origins, *e.g.*, from the canonical  $XYZ$  system to the observed  $X'Y'Z'$  one in Fig. 1. One can determine it with the positive direction of  $Z$ -axis in  $X'Y'Z'$  system (the canonical zenith direction of the  $XYZ$  system) and the in-plane rotation angle  $\beta \in [0, 2\pi]$  around the  $Z$ -axis, which is exactly the decomposition of  $\mathbf{R}$  as multiplication of an out-of-plane (viewpoint) rotation  $\mathbf{R}_{vp}$  and a 2D in-plane rotation  $\mathbf{R}_{ip}$  [3]:

$$\mathbf{R} = \mathbf{R}_{vp}\mathbf{R}_{ip}, \quad (1)$$

with

$$\mathbf{R}_{ip} = \mathbf{R}_Z(\beta) = \begin{bmatrix} \cos \beta & -\sin \beta & 0 \\ \sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2)$$

In the  $X'Y'Z'$  system, let  $\mathbf{v}$  be the final column of the  $3 \times 3$  rotation matrix  $\mathbf{R}$ , which is a unit vector ending at the point  $(v_x, v_y, v_z)$  on a unit sphere  $S^2$ ; the direction of  $\mathbf{v}$  is aligned with the zenith direction of  $XYZ$  system. Transforming the  $X'Y'Z'$  system to spherical coordinate system, we have the spherical coordinate  $(r, \varphi, \theta)$  of the point  $(v_x, v_y, v_z)$  as follows:

$$\begin{cases} r = 1 \\ \varphi = \arctan(v_x/v_z), \\ \theta = \arccos(v_y/r) \end{cases} \quad (3)$$

where  $\varphi \in [0, 2\pi]$  and  $\theta \in [0, \pi]$  are the azimuthal and inclination angles, respectively. Then we can compute  $\mathbf{R}_{vp}$  as follows:

$$\begin{aligned} \mathbf{R}_{vp} &= \mathbf{R}_Z(\varphi)\mathbf{R}_Y(\theta) \\ &= \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}. \end{aligned} \quad (4)$$

Combining (4) and (2), we can find that the rotation decomposition  $\mathbf{R} = \mathbf{R}_{vp}\mathbf{R}_{ip}$  is consistent with the parameterization of  $\mathbf{R}$  with  $ZYZ$ -Euler angles  $\varphi, \theta$  and  $\beta$ .

Motivated by the correlation on the sphere and rotation decomposition, we thus propose in this paper to base the feature learning on the sphere and decouple the estimation of  $\mathbf{R}$ , with no use of object CAD models, into two parts:

- Searching on the sphere for the point  $(v_x, v_y, v_z)$  to obtain the angles  $\varphi$  and  $\theta$ , which jointly give the viewpoint rotation  $\mathbf{R}_{vp}$ ;
- Aligning  $Z'$ -axis with  $Z$ -axis by  $\mathbf{R}_{vp}$  to view from the canonical zenith direction with fewer learning patterns, and then regressing the in-plane rotation  $\mathbf{R}_{ip}$ .

### 4. VI-Net for Rotation Estimation

According to the discussion in Sec. 3, we propose VI-Net for the estimation of rotation  $\mathbf{R}$ , as shown in Fig. 2. Specifically, given a point set  $\mathcal{P}$  of a target object, spherical input signals (Sec. 4.1) are generated for the extraction of high-level viewpoint-equivariant spherical features  $\mathcal{S}$  via a Spherical Feature Pyramid Network (Sec. 4.2); on top of  $\mathcal{S}$ , VI-Net decouples the rotation estimation into two branches: the V-Branch learns the viewpoint rotation  $\mathbf{R}_{vp}$  via binary classification on each sampled grid of the sphere (Sec. 4.3), while the I-Branch estimates the in-plane rotation  $\mathbf{R}_{ip}$  by transforming  $\mathcal{S}$  to view the object from the canonical zenith direction (Sec. 4.4); finally, the rotation  $\mathbf{R}$  is given out as the multiplication of  $\mathbf{R}_{vp}$  and  $\mathbf{R}_{ip}$ . We also provide the training objective in Sec. 4.5.

#### 4.1. Conversion as Spherical Representations

Given a point set  $\mathcal{P} \in \mathbb{R}^{N \times 3}$  with its point-wise attributes  $\mathcal{F} \in \mathbb{R}^{N \times C_0}$  (*e.g.*, radial distance, RGB values, surface normal, *etc.*), we firstly follow [9, 20] to generate the feature map  $\mathcal{S}_0 \in \mathbb{R}^{C_0 \times H_0 \times W_0}$  defined on the sphere, where  $N, H_0 \times W_0$  and  $C_0$  are the point number, the spherical sampling resolution, and attribute dimension (*e.g.*,  $C_0 = 1$  for radial distance and  $C_0 = 3$  for RGB values or surface normal), respectively.

More specifically, we firstly divide, in the spherical coordinate system,  $W_0$  and  $H_0$  bins uniformly along the azimuthal and inclination axes, respectively, resulting in  $H_0 \times W_0$  regions in the space (*c.f.* Fig. 3). Within the region indexed by  $(h, w)$ , we search for the point with the largest radial distance, denoted as  $\mathbf{p}_{h,w}^{max}$ , and set  $\mathcal{S}_0(h, w) = \mathbf{f}_{h,w}^{max} \in \mathcal{F}$ , which corresponds to  $\mathbf{p}_{h,w}^{max}$ ; if there is no point in the region, we set  $\mathcal{S}_0(h, w) = \mathbf{0}$ .

#### 4.2. Spherical Feature Pyramid Network

To process the spherical input  $\mathcal{S}_0$  and model the relationship on the sphere, we construct a Feature Pyramid Network (FPN) [21] with ResNet18[12] by replacing the conventional 2D convolutions with our newly designed *Spatial Spherical Convolutions* (SPA-SConvs), giving out the high-level semantic spherical feature map  $\mathcal{S} \in \mathbb{R}^{C \times H \times W}$ . SPA-SConv is proposed to effectively extract the viewpoint-equivariant spherical features with feature padding and symmetric convolutional operations; we will detail the design of SPA-SConv in Sec. 5.

For the setting of multiple input signals, *e.g.*, two of  $\mathcal{S}_0^{(1)}$  and  $\mathcal{S}_0^{(2)}$ , we employ multiple spherical ResNets to process them individually, and concatenate their features at each stage of ResNets for feature fusion before feeding them into the top-down pathway in FPN; Fig. 2 gives the illustration.

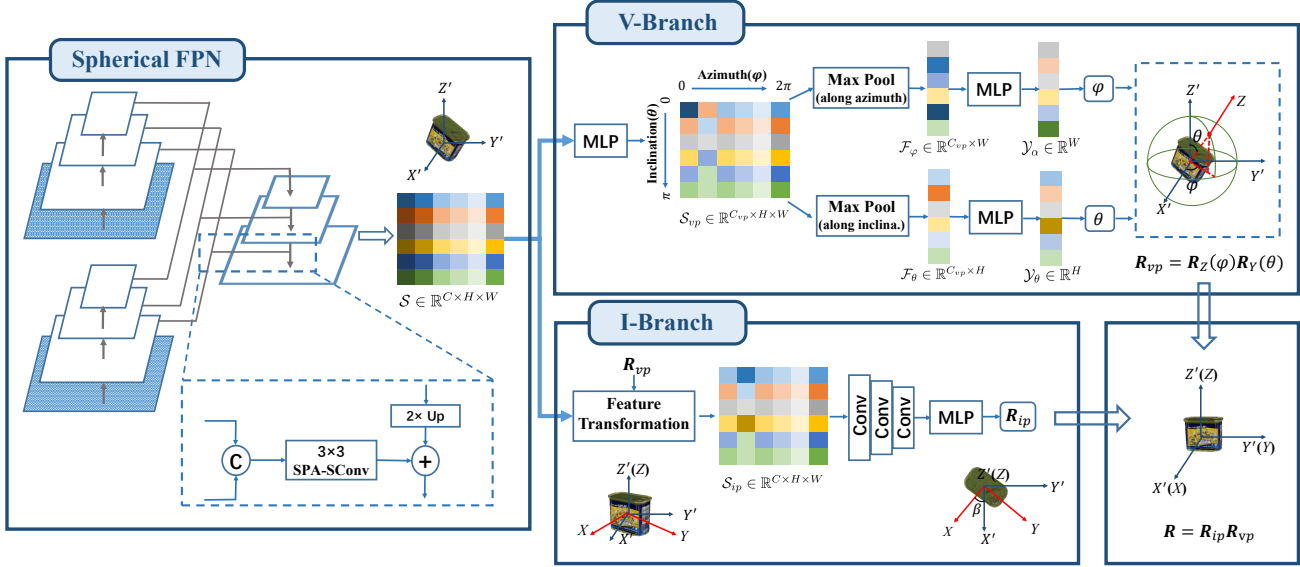


Figure 2. An illustration of VI-Net for rotation estimation. We firstly construct a Spherical Feature Pyramid Network based on spatial spherical convolutions (SPA-SConv) to exact the high-level spherical feature map  $\mathcal{S}$ . On top of  $\mathcal{S}$ , a **V-Branch** is employed to search the canonical zenith direction on the sphere via binary classification for the generation of the viewpoint rotation  $\mathbf{R}_{vp}$ , while another **I-Branch** is used to estimate the in-plane rotation  $\mathbf{R}_{ip}$  by transforming  $\mathcal{S}$  to view the object from the canonical zenith direction. Finally we have  $\mathbf{R} = \mathbf{R}_{vp}\mathbf{R}_{ip}$ . Best view in the electronic version.

### 4.3. V-Branch

Given the spherical feature map  $\mathcal{S}$ , a simple solution to obtain the viewpoint rotation  $\mathbf{R}_{vp}$  is first applying a convolution with a big kernel size (e.g.,  $H \times W$ ) for a global feature and then directly regressing, as done in [20], which is yet required to handle the global relationship on the whole sphere with huge parameters. To ease this problem, we design the V-Branch to search on the sphere for the canonical zenith direction via binary classification.

Consider the spherical feature map  $\mathcal{S} \in \mathbb{R}^{C \times H \times W}$  as  $HW$  spherical anchors. Instead of classifying all of them, we propose to make the task easier by further decoupling it into two sub-tasks, e.g., separate classifications on the azimuthal angle  $\varphi$  and the inclination angle  $\theta$ , which also effectively alleviate the severe imbalance on the ratio of positive and negative spherical anchors ( $1 : HW - 1$ ). Fig. 2 gives the illustration of the V-Branch.

More specifically, we lift the feature channels of  $\mathcal{S}$  with a two-layer MLP to yield  $\mathcal{S}_{vp} \in \mathbb{R}^{C_{vp} \times H \times W}$ . For the learning of azimuthal angle  $\varphi$ , we max-pool  $\mathcal{S}_{vp}$  along the dimension of inclination, giving  $\mathcal{F}_{\varphi} \in \mathbb{R}^{C_{vp} \times W}$ , which is fed into another MLP to generate the probability map  $\mathcal{Y}_{\alpha} \in \mathbb{R}^W$  of  $W$  azimuthal anchors. Each element of  $\mathcal{Y}_{\alpha}$  indicates the possibility of the anchor being the target. Denoting the index of the element with maximum probability as  $w_{max}$ , then we have

$$\varphi = (w_{max} + 0.5)/W \cdot 2\pi. \quad (5)$$

Similarly for the inclination angle  $\theta$ , we max-pool  $\mathcal{S}_{vp}$  along the dimension of azimuth to have  $\mathcal{F}_{\theta} \in \mathbb{R}^{C_{vp} \times H}$ , and generate the probability map  $\mathcal{Y}_{\theta} \in \mathbb{R}^H$  via an MLP. Then  $\theta$  could be computed as follows:

$$\theta = (h_{max} + 0.5)/H \cdot \pi, \quad (6)$$

where  $h_{max}$  is the index of the element with maximum value in  $\mathcal{Y}_{\theta}$ . Finally, combining (5), (6) and (4), we obtain the viewpoint rotation  $\mathbf{R}_{vp} = \mathbf{R}_Z(\varphi)\mathbf{R}_Y(\theta)$ .

### 4.4. I-Branch

With the viewpoint rotation  $\mathbf{R}_{vp}$ , we propose the I-Branch to learn the in-plane rotation  $\mathbf{R}_{ip}$  by viewing the object from the canonical zenith direction to reduce the difficulty of learning, since  $\mathbf{R}_{ip}$  is rather sensitive to  $\mathbf{R}_{vp}$ . The first step is to construct a new spherical feature map  $\mathcal{S}_{ip} \in \mathbb{R}^{C \times H \times W}$  after rotating the observed system by  $\mathbf{R}_{vp}$  to align the zenith directions of both systems, e.g., aligning the  $Z$ -axis in  $XYZ$  system with the  $Z'$ -axis in  $X'Y'Z'$  system (c.f. Fig. 1). The viewpoint-equivariance of  $\mathcal{S}$  makes it possible to realize the transformation in the feature space to obtain  $\mathcal{S}_{ip}$ .

For a regular spherical map with resolution of  $H \times W$ , we denote the center points of all  $HW$  discrete anchors as a point set  $\mathcal{G} = \{\mathbf{g}\}$ . When we rotate the point set  $\mathcal{P} = \{\mathbf{p}\}$  with  $\mathbf{R}_{vp}$  to be  $\mathcal{P}' = \{\mathbf{p}'\} = \{\mathbf{R}_{vp}^T \mathbf{p}\}$ , the anchor points of  $\mathcal{S}$  are also rotated to be  $\mathcal{G}' = \{\mathbf{g}'\} = \{\mathbf{R}_{vp}^T \mathbf{g}\}$ ; we denote

the feature of  $\mathbf{g}'$  as  $\mathcal{S}^{\mathbf{g}'}$ . To view an object from the canonical zenith direction, a new regular spherical feature map  $\mathcal{S}_{ip}$  should be constructed for the transformed  $\mathcal{P}'$ , based on the anchor points  $\mathcal{G}$  with canonical orders. For  $\mathcal{S}_{ip}$ , we generate its feature on each anchor point  $\mathbf{g}$ , denoted as  $\mathcal{S}_{ip}^{\mathbf{g}}$ , from  $\mathcal{S}$  by using the weighted interpolation of point features [26] as follows:

$$\mathcal{S}_{ip}^{\mathbf{g}} = \frac{\sum_{i=1}^k a_i \mathcal{S}^{\mathbf{g}'_i}}{\sum_{i=1}^k a_i}, \quad (7)$$

where  $a_i = \frac{1}{\|\mathbf{g}-\mathbf{g}'_i\|^2}$  is the weight for interpolation measured by point distance, and  $\{\mathbf{g}'_i\}_{i=1}^k \subset \mathcal{G}'$  is the  $k$  nearest points of  $\mathbf{g}$ .

After the feature transformation from  $\mathcal{S}$  to  $\mathcal{S}_{ip}$  via (7), we use several stacked strided convolutions to reduce the resolution of  $\mathcal{S}_{ip}$  and extract a global feature for the regression of  $\mathbf{R}_{ip}$ ; Fig. 2 also gives the illustration. The continuous 6D representation of rotation [39] is used as the output of regression, and then transformed to the rotation matrix  $\mathbf{R}_{ip}$ . We note that  $\mathbf{R}_{ip}$  here is not restricted to have only one degree-of-freedom (*i.e.*, the angle  $\beta$  in (2)), and is learned in  $SO(3)$  instead as the combination of the residual viewpoint rotation and the exact in-plane rotation, since  $\mathbf{R}_{vp}$  generated by V-Branch is a coarse prediction of the exact viewpoint rotation with discretized azimuthal and inclination angles.

#### 4.5. Training of VI-Net

For V-Branch, we use focal loss [22] on top of two binary classifiers, given the ground truth labels  $\hat{\mathcal{Y}}_{\varphi} \in \mathbb{R}^W$  and  $\hat{\mathcal{Y}}_{\theta} \in \mathbb{R}^H$ , as follows:

$$\mathcal{L}_{vp} = \mathcal{D}_{FL}(\mathcal{Y}_{\varphi}, \hat{\mathcal{Y}}_{\varphi}) + \mathcal{D}_{FL}(\mathcal{Y}_{\theta}, \hat{\mathcal{Y}}_{\theta}), \quad (8)$$

where

$$\mathcal{D}_{FL}(\mathcal{Y}, \hat{\mathcal{Y}}) = \frac{1}{M} \sum_{i=1}^M -\alpha(1 - y_{i,t})^{\gamma} \log(y_{i,t}), \quad (9)$$

and

$$y_{i,t} = \begin{cases} y_i & \text{if } \hat{y}_i = 1 \\ 1 - y_i & \text{if } \hat{y}_i = 0 \end{cases}, \quad (10)$$

with  $\mathcal{Y} = \{y_i\}_{i=1}^M$  and  $\hat{\mathcal{Y}} = \{\hat{y}_i \in \{0, 1\}\}_{i=1}^M$ .  $\alpha$  is the weighting factor, and  $\gamma$  denotes the exponent of the modulating factor.

Given the ground truth rotation  $\hat{\mathbf{R}}$ , we supervise the final prediction  $\mathbf{R} = \mathbf{R}_{vp} \mathbf{R}_{ip}$  on top of the I-Branch as follows:

$$\mathcal{L}_{ip} = \|\mathbf{R} - \hat{\mathbf{R}}\| = \|\mathbf{R}_{vp} \mathbf{R}_{ip} - \hat{\mathbf{R}}\|. \quad (11)$$

Combining (8) and (11) gives the following optimization problem to train our VI-Net:

$$\min \mathcal{L} = \mathcal{L}_{ip} + \lambda \mathcal{L}_{vp}, \quad (12)$$

where  $\lambda$  is the balanced parameter.

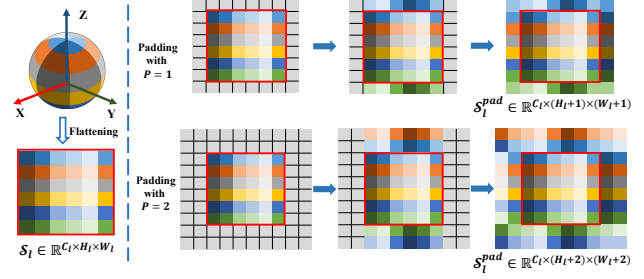


Figure 3. An illustration of the feature padding in SPA-SConv. Best view in the electronic version.

## 5. Spatial Spherical Convolution

We base the feature learning on the sphere for rotation estimation by building the signals as spherical feature maps with regular 2D spatial sizes. Such representations make it possible to process spherical signals via 2D convolutions. However, directly applying 2D convolutions could not realize continuous feature learning on the sphere, resulting in the boundary problem; for example,  $S_0(h, 1)$  has a long distance to  $S_0(h, W)$  on the feature map  $\mathcal{S}_0$  without connection, but their regions border on the sphere actually. Besides, to support the feature transformation in I-Branch for viewing from the zenith direction, convolutions forming the backbone are also required to be viewpoint-equivariant. To settle the above issues, we propose a novel *SPAtial Spherical Convolution*, shortened as **SPA-SConv**, to extract viewpoint-equivariant features continuously on the sphere, which could be flexibly adapted into existing convolutional network architectures, *e.g.*, FPN in Sec. 4.2.

Given the input spherical feature map  $\mathcal{S}_l \in \mathbb{R}^{C_l \times H_l \times W_l}$  and the parameters w.r.t. convolution (*e.g.*, the kernel size  $K$ , stride  $s$ , the number of output channels  $C_{l+1}$ ), we implement our SPA-SConv in two steps, including 1) padding  $\mathcal{S}_l$  to  $\mathcal{S}_l^{pad} \in \mathbb{R}^{C_l \times (H_l+2P) \times (W_l+2P)}$  with  $P = (K-1)/2$ , and 2) applying symmetric convolutional operations to  $\mathcal{S}_l^{pad}$ , based on conventional 2D convolution with no padding, to generate the output spherical feature map  $\mathcal{S}_{l+1} \in \mathbb{R}^{C_{l+1} \times H_{l+1} \times W_{l+1}}$ , where  $H_{l+1} = \lfloor H_l/s \rfloor$  and  $W_{l+1} = \lfloor W_l/s \rfloor$ .<sup>1</sup>

Fig. 3 illustrates the padding of  $\mathcal{S}_l$  to  $\mathcal{S}_l^{pad}$ . Firstly, we have  $\mathcal{S}_l$  at the center of  $\mathcal{S}_l^{pad}$ :

$$\mathcal{S}_l^{pad}(h+P, w+P) = \mathcal{S}_l(h, w), \quad (13)$$

for  $\forall h = 1, 2, \dots, H_l$  and  $\forall w = 1, 2, \dots, W_l$ . Next, we pad  $\mathcal{S}_l$  along inclination ( $\theta$ -axis in the sphere coordinate

<sup>1</sup>For simplicity, We assume the length and width are both  $K$  ( $K$  is an odd number) for the 2D convolutional kernel.

system) as follows:

$$\begin{aligned} \mathcal{S}_l^{pad}(p, w + P) &= \mathcal{S}_l^{pad}(2P - p + 1, w'), \\ \text{and } \mathcal{S}_l^{pad}(H_l + P + p, w + P) &= \mathcal{S}_l^{pad}(H_l + P - p + 1, w'), \end{aligned} \quad (14)$$

where

$$w' = \begin{cases} w + W_l/2 + P & \text{if } w \leq W_l/2 \\ w - W_l/2 + P & \text{otherwise} \end{cases}, \quad (15)$$

for  $\forall p = 1, 2, \dots, P$  and  $\forall w = 1, 2, \dots, W_l$ . Finally we pad the feature map along azimuth ( $\varphi$ -axis in the sphere coordinate system):

$$\begin{aligned} \mathcal{S}_l^{pad}(h, p) &= \mathcal{S}_l^{pad}(h, W_l + p), \\ \text{and } \mathcal{S}_l^{pad}(h, W_l + P + p) &= \mathcal{S}_l^{pad}(h, P + p), \end{aligned} \quad (16)$$

for  $\forall p = 1, 2, \dots, P$  and  $\forall h = 1, 2, \dots, H_l + 2P$ .

With the padded  $\mathcal{S}_l^{pad}$ , we could thus exploit the 2D convolution to realize the symmetric convolutional operations for the extraction of viewpoint-equivariant features as follows:

$$\mathcal{S}_{l+1} = \text{Max}(\text{Conv}(\mathcal{S}_l^{pad}; \kappa_l), \text{Conv}(\mathcal{S}_l^{pad}; \text{Flip}(\kappa_l))), \quad (17)$$

where  $\text{Conv}$ ,  $\text{Flip}$ , and  $\text{Max}$  denotes the 2D convolutional operation, horizontal flip, and element-wise max-pooling, respectively, with  $\kappa_l$  denoting the weight of the convolution.  $\text{Max}$  serves as the symmetric function to aggregate the features [25] and keeps the property of viewpoint-equivariance; we provide the proof of this property in the supplemental material, where the pseudo code is also included to help understand the implementation of SPA-SConv.

## 6. Category-level 6D Object Pose Estimation

The task of category-level 6D object pose estimation is to estimate the poses of unknown objects w.r.t. categorical canonical spaces, including the learning of rotation  $\mathbf{R} \in SO(3)$ , translation  $\mathbf{t} \in \mathbb{R}^3$  and size  $s \in \mathbb{R}^3$  for each object. CAD models are not available during testing time. To confirm the efficacy of our VI-Net on rotation estimation, we apply VI-Net to this challenging task and realize the target with three stages. Specifically, given an RGB-D image of a cluttered scene, we firstly employ MaskRCNN [11] to segment the objects of interest out; for each cropped object with a point set  $\mathcal{P} = \{\mathbf{p}\}$ , we then estimate its translation  $\mathbf{t}$  and size  $s$  via a simple and lightweight network of PointNet++ (PN2) [26] with multi-scale grouping; in the final stage, we normalize the point set as  $\mathcal{P}' = \{(\mathbf{p} - \mathbf{t}) / \|s\|\}$  and feed it into our VI-Net for the estimation of rotation  $\mathbf{R}$ .

### 6.1. Experimental Setups

**Datasets** Experiments are conducted on the benchmarking CAMERA25 and REAL275 datasets [31], both of which

enjoy the same object categories. CAMERA25 is a synthetic dataset generated by a context-aware mixed reality approach; it contains 300,000 RGB-D images of 1,085 objects in total, with 25,000 images of 184 objects set aside for validation. REAL275 is a challenging real-world dataset with various lighting conditions and occlusions; for this dataset, 4,300 images of 7 scenes are split as a training set, while the other 2,750 images of 6 scenes form the test set. Following [31], we mix both training sets to optimize the parameters of VI-Net.

**Implementation Details** We use the instance masks generated by MaskRCNN [11] and provided by [28] to segment objects of interest out. For data processing, we use point-wise radial distances and RGB values as two input signals. The input resolution of  $\mathcal{S}_0$  is set as  $H_0 \times W_0 = 64 \times 64$ , while that of  $\mathcal{S}$  is  $H \times W = 32 \times 32$ . The point number of the sampled point set is  $N = 2,048$ , and  $k$  in (7) is set as 3. For the training of VI-Net, we set  $\alpha = 0.5$  and  $\gamma = 2.0$  in (9); the balanced parameter  $\lambda$  in (12) is set as 100. We use ADAM to train VI-Net for a total of 200,000 iterations. The learning rate is initialized as 0.001, with a cosine annealing schedule used afterward, while the data size of a min-batch is set as 128.

**Evaluation Metrics** We report the mean Average Precision (mAP) of  $n^\circ m$  cm for 6D pose estimation, which denotes the pose precision with rotation error less than  $n^\circ$  and translation error less than  $m$  cm. Following [31, 23], we also report mAP of Intersection over Union ( $\text{IoU}_x$ ) with a threshold of  $x\%$  for object detection.

### 6.2. Comparisons with Existing Methods

To verify the efficacy of VI-Net, we conduct experiments on REAL275 and CAMERA25 datasets [31] to compare with the existing methods, including those without the use of shape priors [31, 5, 20, 8, 18], and those using shape priors [28, 32, 10, 17, 37, 4, 36, 23]. We firstly employ the lightweight network of PointNet++ (PN2) [26] for translation and size estimation, before using VI-Net for rotation estimation, and shape priors (as well as CAD object models) are not used in our method. Quantitative results of different methods are given out in Table 1.

In terms of pose estimation, our method outperforms those without the use of shape priors on all the evaluation metrics of  $n^\circ m$  cm with large margins on REAL275 dataset, as shown in the table. For example, on the metric of  $5^\circ 2$  cm, our method refreshes the result to reach the mAP of 50.0%, which exceeds the  $SE(3)$ -equivariant SS-ConvNet [18] and the DualPoseNet [20] based on  $SO(3)$ -equivariant SPE-SConv [9] by 13.4% and 20.7%, respectively. We note that DualPoseNet also learns object poses on the sphere by applying convolutions with large kernels to model the global relationship, while our VI-Net exploits Spherical FPN with SPA-SConv to model local relationships along

Method	Use of Shape Priors	REAL275					CAMERA25				
		IoU <sub>75*</sub>	5°2cm	5°5cm	10°2cm	10°5cm	IoU <sub>75*</sub>	5°2cm	5°5cm	10°2cm	10°5cm
NOCS [31]	×	9.4	7.2	10.0	13.8	25.2	37.0	32.3	40.9	48.2	64.6
FS-Net [5]	×	-	-	28.2	-	60.8	-	-	-	-	-
DualPoseNet [20]	×	30.8	29.3	35.9	50.0	66.8	71.7	64.7	70.7	77.2	84.7
GPV-Pose [8]	×	-	32.0	42.9	-	73.3	-	72.1	79.1	-	89.0
SS-ConvNet [18]	×	-	36.6	43.4	52.6	63.5	-	-	-	-	-
<b>PN2 + VI-Net (Ours)</b>	×	<b>48.3</b>	<b>50.0</b>	<b>57.6</b>	<b>70.8</b>	<b>82.1</b>	<b>79.1</b>	<u>74.1</u>	<b>81.4</b>	79.3	87.3
SPD [28]	✓	27.0	19.3	21.4	43.2	54.1	46.9	54.3	59.0	73.3	81.5
CR-Net [32]	✓	33.2	27.8	34.3	47.2	60.8	75.0	72.0	76.4	81.0	87.7
CenterSnap-R [14]	✓	-	-	29.1	-	64.3	-	-	66.2	-	81.3
ACR-Pose [10]	✓	-	31.6	36.9	54.8	65.9	-	70.4	74.1	82.6	87.8
SAR-Net [17]	✓	-	31.6	42.3	50.3	68.3	-	66.7	70.9	75.3	80.3
SSP-Pose [37]	✓	-	34.7	44.6	-	77.8	-	64.7	75.5	-	87.4
SGPA [4]	✓	37.1	35.9	39.6	61.3	70.7	69.1	70.7	74.5	<u>82.7</u>	88.4
RBP-Pose [36]	✓	-	38.2	48.1	63.1	<u>79.2</u>	-	73.5	79.6	82.1	<b>89.5</b>
SPD + CATRE [23]	✓	<u>43.6</u>	45.8	<u>54.4</u>	61.4	73.1	<u>76.1</u>	<b>75.4</b>	<u>80.3</u>	<b>83.3</b>	<u>89.3</u>
DPDN [19]	✓	-	<u>46.0</u>	50.7	<u>70.4</u>	78.4	-	-	-	-	-

Table 1. Quantitative comparisons of different methods for category-level 6D object pose estimation on REAL275 and CAMERA25 datasets [31]. Overall best results are in **bold** and the second best results are underlined. ‘\*’ denotes the IoU metrics used in [23] rather than those in [31].

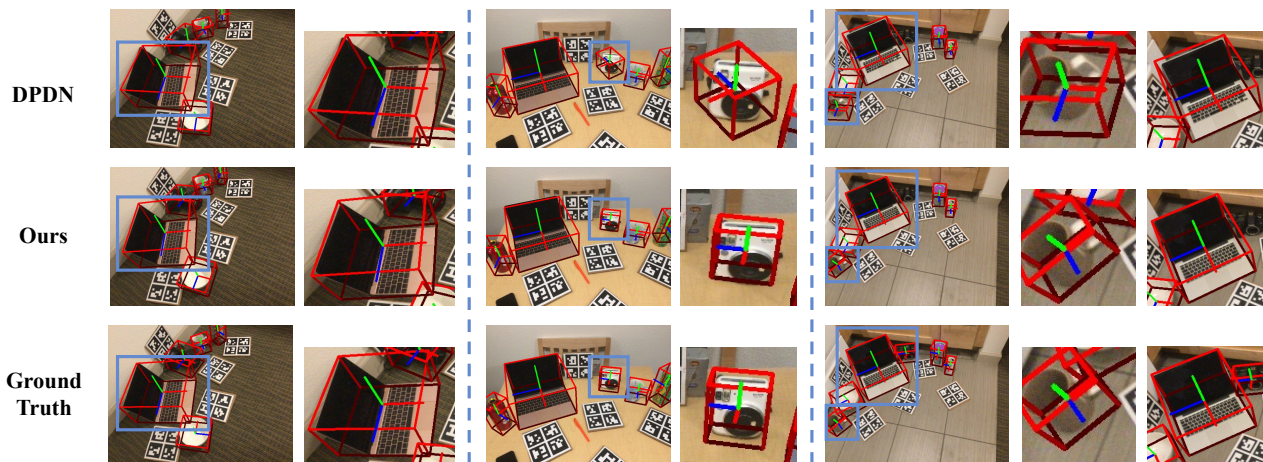


Figure 4. Qualitative comparisons between the state-of-the-art method of DPDN [19] and our proposed one on REAL275 dataset [31].

	R.D.	5°2cm	5°5cm
Baseline1: Avg Pool + MLP	×	45.0	51.8
Baseline2: Flattening + MLP	×	42.7	48.8
VI-Net	✓	<b>50.0</b>	<b>57.6</b>

Table 2. Ablation studies of the variants of our VI-Net with or without rotation decomposition (R.D.) on REAL275 dataset [31].

hierarchy more efficiently, and achieves much better performance with the individual learning of the factorized rotations. Compared to the methods using shape priors, VI-Net also possesses powerful competition on REAL275 dataset; taking DPDN [19] as example, our method improves the results by 4.0% mAP on 5°2cm and 6.9% mAP on 5°5cm,

verified by the quantitative comparisons in Fig. 4. We provide per-category comparisons of our method in terms of rotation error and translation error on REAL275 dataset in Fig. 5. On CAMERA25 dataset, as shown in Table 1, our method also achieves comparable results, especially on the more strict metrics, *e.g.*, IoU<sub>75</sub>.

### 6.3. Ablation Studies and Analyses

To confirm the efficacy of our designs, we conduct experiments to compare the variants of VI-Net in different aspects on REAL275 dataset [31].

**Efficacy of Rotation Decomposition in VI-Net** The foundation of VI-Net is the decomposition of rotation into a viewpoint rotation and an in-plane rotation, which guides

Variants of V-Branch	Feature Trans. in I-Branch	5°2cm	5°5cm	10°2cm	10°5cm
Direct Regression	×	26.3	28.9	54.1	61.1
Binary Classification (1-branch)	×	30.8	34.4	65.5	75.3
Binary Classification (2-branch)	×	33.7	37.9	65.4	75.4
Direct Regression	✓	43.1	48.3	68.0	78.1
Binary Classification (1-branch)	✓	49.3	56.9	69.4	79.7
Binary Classification (2-branch)	✓	<b>50.0</b>	<b>57.6</b>	<b>70.8</b>	<b>82.1</b>

Table 3. Ablation studies of the variants of the V-Branch and I-Branch in our VI-Net on REAL275 dataset [31].

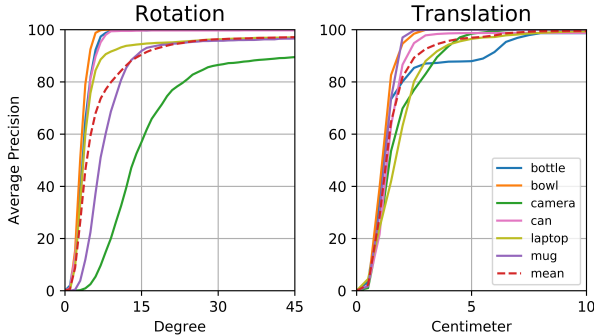


Figure 5. Plottings of per-category average precision versus different rotation/translation error thresholds for our proposed method on REAL275 dataset [31].

Type of Convolution	Padding	Symmetric Operation	5°2cm	5°5cm
SPE-SConv [9]	—	—	35.1	40.8
SPA-SConv	×	×	46.9	53.2
	×	✓	48.5	55.5
	✓	✓	<b>50.0</b>	<b>57.6</b>

Table 4. Quantitative comparisons of SPE-SConv [9] and variants of our proposed SPA-SConv on REAL275 dataset [31].

	5°2cm	5°5cm	10°2cm	10°5cm
VI-Net <sub>+ts</sub>	45.0	56.0	65.9	80.5
PN2 + VI-Net	<b>50.0</b>	<b>57.6</b>	<b>70.8</b>	<b>82.1</b>

Table 5. Quantitative comparisons between VI-Net<sub>+ts</sub> and PN2 for translation and size estimation on REAL275 dataset [31].

Data Type	5°2cm	5°5cm	10°2cm	10°5cm
Depth	43.0	52.1	64.3	77.0
RGB + Depth	<b>50.0</b>	<b>57.6</b>	<b>70.8</b>	<b>82.1</b>

Table 6. Quantitative comparisons of different input data types of VI-Net on REAL275 dataset [31].

to factorize the task into two sub ones to reduce the learning difficulty in the whole  $SO(3)$  space. We thus firstly verify the mechanism of rotation decomposition in VI-Net by building two baselines, which regress the rotation  $\mathbf{R}$  directly from the global features learned by either globally

average-pooling or flattening the spherical feature map  $\mathcal{S}$ . Both baselines employ the same backbone of Spherical FPN as VI-Net, and their results are listed in Table 2, where VI-Net outperforms both of them with large margins on all the evaluation metrics, confirming the effectiveness of rotation decomposition in pose estimation.

### Efficacy of Separate Binary Classifications in V-Branch

We compare three variants of V-Branch, including i) direct regression of  $\mathbf{R}_{vp}$  from the globally average-pooled feature from  $\mathcal{S}$ , binary classification on the whole spherical map, and the two separate binary classifications we introduced in Sec. 4.3. As shown in Table 3, among three variants, the last two ones make more precise predictions by searching on the sphere for the canonical zenith direction via binary classification, which simplifies the regression task as the recognition one(s). The last one further improves the pose precision via the decoupled recognition of the azimuthal angle  $\varphi$  and the inclination angle  $\theta$ , since binary classification on the whole spherical map faces the severe imbalance on the ratio of positive and negative samples (1:1023 in our experiments), even though we use the focal loss (9) to alleviate this problem. we also note that, comparing with the results in Table 2, the key to the success of rotation decomposition for VI-Net lies in exactly the binary classification of V-Branch, *i.e.*, the correlation on the rotation decomposition and sphere. Finally, We visualize some examples in Fig. 6, where object point clouds transformed by the predicted viewpoint rotations from V-Branch are presented.

### Efficacy of Feature Transformation in I-Branch

In Table 3, we also verify, with three different variants of V-Branch, the advantages of feature transformation in I-Branch, which transforms in feature space with viewpoint rotation  $\mathbf{R}_{vp}$  to make the network “see” the object from the canonical zenith direction, and thus reduces the difficulty in estimating the in-plane rotation  $\mathbf{R}_{ip}$ . Applying the feature transformation also makes I-Branch aware of the learned  $\mathbf{R}_{vp}$ , and could include the learning of the residual viewpoint rotation in I-Branch to refine predictions from V-Branch.

### Efficacy of Spatial Spherical Convolutions (SPA-SConv)

We propose SPA-SConv to realize continuous convolution on the sphere, which settles the boundary problems of spherical representations by simply feature padding and re-



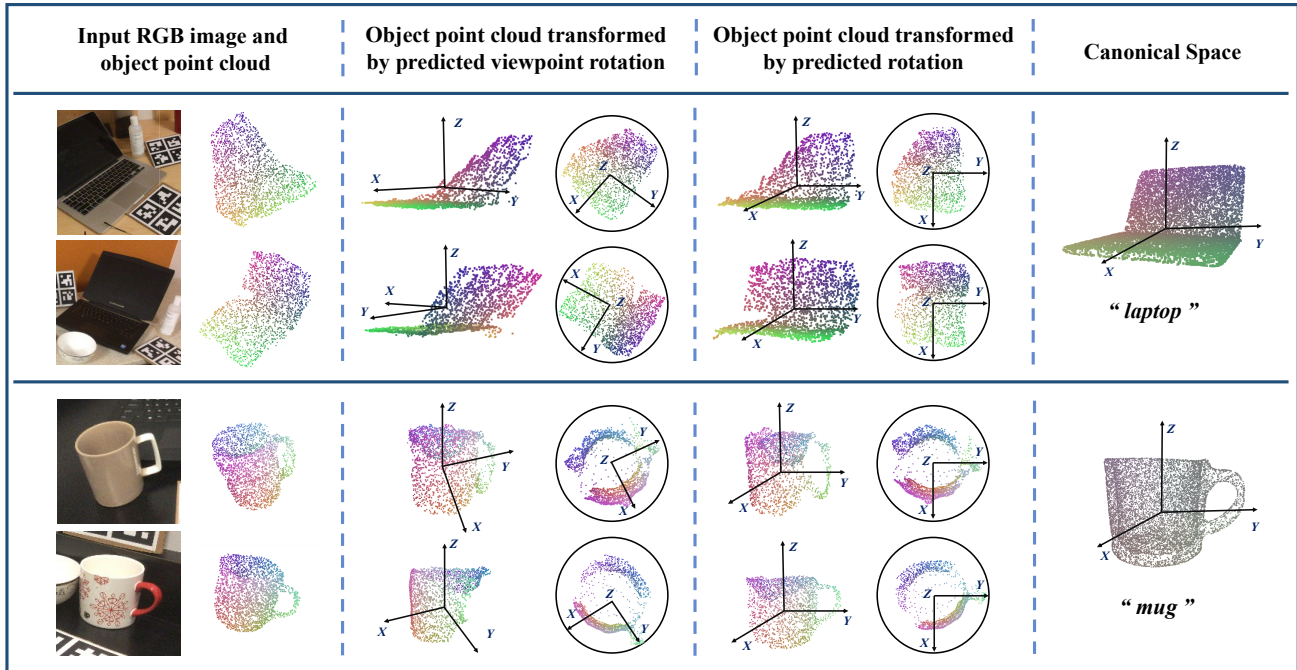


Figure 6. Qualitative results of the predicted viewpoint rotation  $R_{vp}$  from V-Branch and the final output rotation  $R$  from VI-Net.

alizes the properties of viewpoint-equivariance by the symmetric convolutional operations. In Table 4, we demonstrate the advantages of these novel designs individually. As introduced in Sec. 2.2, the  $SO(3)$ -equivariant spherical feature maps learnt by SPE-SConv [9] are less discriminative for the binary classification of V-Branch, even resulting in gradient exploding. Therefore, to avoid gradient exploding, we add 4 additional SPA-Convs in V-Branch for the VI-Net based on SPE-SConvs, and the training thus becomes stable; however, the results based on SPE-SConvs are still less satisfactory compared to those of SPA-SConvs, as shown in Table 4, indicating the importance of SPA-SConv in the design of VI-Net.

**Effects of Including Translation and Size Estimation in VI-Net** We focus on learning decoupled rotations on spherical inputs converted from centralized point sets, and separate translation and size estimation via PointNet++ (PN2). Another choice (dubbed VI-Net<sub>+ts</sub>) is to use the mean coordinate of the point set as initial translation and include the residual translation/size estimation with an MLP from the spherical features  $\mathcal{S}$ . As shown in Table 5, the performance gap between VI-Net<sub>+ts</sub> and PN2 is acceptable, motivating us to explore better combined learning in the future.

**Effects of Different Input Data Types** The primary input data of VI-Net are depth maps (or point clouds), which are used for computing the point locations on the sphere and further transformed as radial distances to be the input signals. Additionally, other point attributes, e.g., RGB values,

could also be the inputs. In Table 6, we demonstrate that VI-Net effectively learns both geometric and appearance information from Depth maps and RGB images, respectively, which indicates VI-Net could be used for handling different input types to jointly make the predictions.

## 7. Conclusion

In this paper, we propose a new method of VI-Net for rotation estimation, which bases the feature learning on the sphere, and factorizes the task into two sub-ones with individual heads of V-Branch and I-Branch. V-Branch learns the viewpoint rotation via two separate binary classifications on top of the spherical feature maps, while I-Branch transforms the feature maps with the viewpoint rotation to view objects from the canonical zenith direction for the estimate of in-plane rotation. To support the learning of these two branches, we also propose a novel design of Spatial Spherical Convolution for the extraction of viewpoint-equivariant spherical features. Finally, we apply our VI-Net to the challenging task of category-level 6D object pose estimation and experimental results confirm its efficacy in the regime of high precision of rotation.

**Acknowledgements** This work is supported in part by Guangdong R&D key project of China (No.: 2019B010155001), and the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (No.: 2017ZT07X183).

## References

- [1] Ronald T Azuma. A survey of augmented reality. *Presence: teleoperators & virtual environments*, 6(4):355–385, 1997.
- [2] Wouter Boomsma and Jes Frellsen. Spherical convolutions and their application in molecular modelling. *Advances in neural information processing systems*, 30, 2017.
- [3] Dingding Cai, Janne Heikkilä, and Esa Rahtu. Ove6d: Object viewpoint encoding for depth-based 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6803–6813, 2022.
- [4] Kai Chen and Qi Dou. Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2773–2782, 2021.
- [5] Wei Chen, Xi Jia, Hyung Jin Chang, Jinming Duan, Linlin Shen, and Ales Leonardis. 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*, pages 1581–1590, 2021.
- [6] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.
- [7] Shengheng Deng, Zhihao Liang, Lin Sun, and Kui Jia. Vista: Boosting 3d object detection via dual cross-view spatial attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8448–8457, 2022.
- [8] Yan Di, Ruida Zhang, Zhiqiang Lou, Fabian Manhardt, Xiangyang Ji, Nassir Navab, and Federico Tombari. 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*, pages 6781–6791, 2022.
- [9] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018.
- [10] Zhaoxin Fan, Zhengbo Song, Jian Xu, Zhicheng Wang, Kejian Wu, Hongyan Liu, and Jun He. Acr-pose: Adversarial canonical representation reconstruction network for category level 6d object pose estimation. *arXiv preprint arXiv:2111.10524*, 2021.
- [11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [13] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3003–3013, 2021.
- [14] Muhammad Zubair Irshad, Thomas Kollar, Michael Laskey, Kevin Stone, and Zolt Kira. Centersnap: Single-shot multi-object 3d shape reconstruction and categorical 6d pose and size estimation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10632–10640. IEEE, 2022.
- [15] Jesse Levinson, Jake Askeland, Jan Becker, Jennifer Dolson, David Held, Soeren Kammel, J Zico Kolter, Dirk Langer, Oliver Pink, Vaughan Pratt, et al. Towards fully autonomous driving: Systems and algorithms. In *2011 IEEE intelligent vehicles symposium (IV)*, pages 163–168. IEEE, 2011.
- [16] Hongyang Li, Jiehong Lin, and Kui Jia. Dcl-net: Deep correspondence learning network for 6d pose estimation. In *European Conference on Computer Vision*, pages 369–385. Springer, 2022.
- [17] Haitao Lin, Zichang Liu, Chilam Cheang, Yanwei Fu, Guodong Guo, and Xiangyang Xue. 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*, pages 6707–6717, 2022.
- [18] Jiehong Lin, Hongyang Li, Ke Chen, Jiangbo Lu, and Kui Jia. Sparse steerable convolutions: An efficient learning of se (3)-equivariant features for estimation and tracking of object poses in 3d space. *Advances in Neural Information Processing Systems*, 34, 2021.
- [19] Jiehong Lin, Zewei Wei, Changxing Ding, and Kui Jia. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In *European Conference on Computer Vision*, pages 19–34. Springer, 2022.
- [20] Jiehong Lin, Zewei Wei, Zhihao Li, Songcen Xu, Kui Jia, and Yuanqing Li. 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*, pages 3560–3569, 2021.
- [21] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.
- [22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.
- [23] Xingyu Liu, Gu Wang, Yi Li, and Xiangyang Ji. Catre: Iterative point clouds alignment for category-level object pose refinement. In *European Conference on Computer Vision*, pages 499–516. Springer, 2022.
- [24] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2901–2910, 2019.
- [25] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification

- and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.
- [26] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.
- [27] Yu-Chuan Su and Kristen Grauman. Learning spherical convolution for fast features from 360 imagery. *Advances in Neural Information Processing Systems*, 30, 2017.
- [28] Meng Tian, Marcelo H Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *European Conference on Computer Vision*, pages 530–546. Springer, 2020.
- [29] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 13(04):376–380, 1991.
- [30] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3343–3352, 2019.
- [31] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. 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*, pages 2642–2651, 2019.
- [32] Jiaze Wang, Kai Chen, and Qi Dou. Category-level 6d object pose estimation via cascaded relation and recurrent reconstruction networks. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4807–4814. IEEE, 2021.
- [33] Zhixin Wang and Kui Jia. Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1742–1749. IEEE, 2019.
- [34] Chaozheng Wu, Jian Chen, Qiaoyu Cao, Jianchi Zhang, Yunxin Tai, Lin Sun, and Kui Jia. Grasp proposal networks: An end-to-end solution for visual learning of robotic grasps. *Advances in Neural Information Processing Systems*, 33:13174–13184, 2020.
- [35] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.
- [36] Ruida Zhang, Yan Di, Zhiqiang Lou, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Rbp-pose: Residual bounding box projection for category-level pose estimation. In *European Conference on Computer Vision*, pages 655–672. Springer, 2022.
- [37] Ruida Zhang, Yan Di, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Ssp-pose: Symmetry-aware shape prior deformation for direct category-level object pose estimation. *arXiv preprint arXiv:2208.06661*, 2022.
- [38] Qiang Zhao, Chen Zhu, Feng Dai, Yike Ma, Guoqing Jin, and Yongdong Zhang. Distortion-aware cnns for spherical images. In *IJCAI*, pages 1198–1204, 2018.
- [39] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. *CoRR*, abs/1812.07035, 2018.