IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

6D Pose Estimation Based on 3D Edge Binocular Reprojection Optimization for Robotic Assembly

Dong Li, Quan Mu, Yilin Yuan, Shiwei Wu, Ye Tian, Hualin Hong, Qian Jiang, and Fei Liu*

***Abstract*—Accurate 6D pose estimation of object is important for robot assembly. This letter presents a novel method for achieving high precision 6D pose estimation by exploiting the reprojection of 3D edges onto binocular RGB image pairs. Our proposed method encompasses three phases: detection, pose initialization, and pose refinement. In the detection phase, an existing detector is employed to identify the objects within the image pairs. Subsequently, the object image patch of interest is extracted and fed into an encoder-decoder network that leverages edge maps and RGB images for the purpose of initial pose estimation. To refine the initial pose and achieve precise 6D pose estimation, we introduce a novel binocular edge-map-based nonlinear optimization technique.**

Our primary contributions entail an improved initial pose estimation network and a novel pose optimization technique. The improved network is dedicated to enhancing the accuracy of initial pose estimation, while the optimization technique focuses on refining the precision of the estimations. Experimental results demonstrate the effectiveness of our method, yielding an average translation precision of 0.48 mm and rotation precision of 0.45 degrees. Consequently, our proposed method can be seamlessly integrated into robotic manipulation platforms to successfully execute diverse assembly tasks.

***Index Terms*—6D pose estimation, pose refinement, robotic assembly**

I. INTRODUCTION

THE pose estimation of rigid objects is a crucial research direction in robotics and computer vision. It aims to determine the 3D translation and rotation of an object relative to a canonical frame. Estimating the 6D pose of objects from images holds significant importance in robot operation, Augmented Reality (AR), and other applications.

Thanks to the development of RGB-D cameras, some recent methods[1-5] use depth information to improve the robustness and accuracy of 6d pose estimation, and have achieved excellent results. However, RGB-D cameras still have some limitations, such as higher costs and challenges in handling strong lighting conditions. In contrast, the usage of RGB cameras seems to be more ubiquitous in practical applications. Therefore, RGB images for 6D pose estimation is promise even it has more challenges such as lighting effects, texture-less objects, and cluttered backgrounds.

Recent methods[6-9] utilize CNNs for 6D object pose estimation in RGB images. These methods effectively address the challenges of illumination, cluttered backgrounds, and occlusion, and demonstrate robust performance. While these networks estimate 6d pose directly from a single RGB image, further pose refinement is still necessary for higher precision. Therefore, some pose refinement methods[10, 11] based on edge cues or neural network iterations are proposed. However, due to the lack of information in the depth direction, current single-RGB based pose refinement methods are still sensitive in depth.

In this letter, we introduce a novel pose refinement method that relies on binocular image pair to address the challenges associated with depth sensitivity in monocular pose estimation methods and achieve high-precision pose estimation. To implement the complete pose estimation process, we also design a pipeline, as shown in the left of **Fig. 2**, which consists of three phases: detection, pose initialization, and optimization. In the detection phase, we employ an existing detector to extract the object regions from the image pair. In the pose initialization phase, we propose an initial pose estimation network Auto-Encoder of Color image and Edge map(AECE) based on AAE[12], as shown in the right of **Fig.**

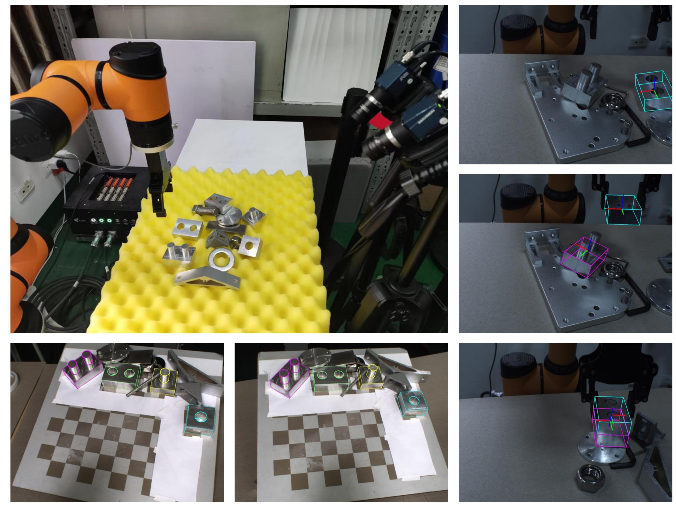

Fig. 1. The proposed method is evaluated by practical robotic assembly task and our metal part dataset. The color points are projected from the edge points on the CAD model using the estimated pose.

Manuscript received: July 10, 2023; Revised: September 26, 2023; Accepted: October 17, 2023. This paper was recommended for publication by Editor Ashis Banerjee upon evaluation of the Associate Editor and Reviewers' comments. This work is supported in part by the National Natural Science Foundation of China (No.T2222018) and Innovation group science fund of Chongqing Natural Science Foundation (cstc2019jcyj-cxttX0003).

Dong Li, Yilin Yuan, Shiwei Wu, Ye Tian, Hualin Hong, Qian Jiang, and Fei Liu are with the State Key Laboratory of Mechanical Transmissions, School of Mechanical and Vehicle Engineering, Chongqing University Chongqing 400000, China. The corresponding author is Fei Liu (e-mail: fei_liu@cqu.edu.cn).

Quan Mu, Foreign Environmental Cooperation Center, Ministry of Ecology and Environment, 100035, P.R.China, Email: mu.quan@fecomee.org.cn

Digital Object Identifier (DOI):

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

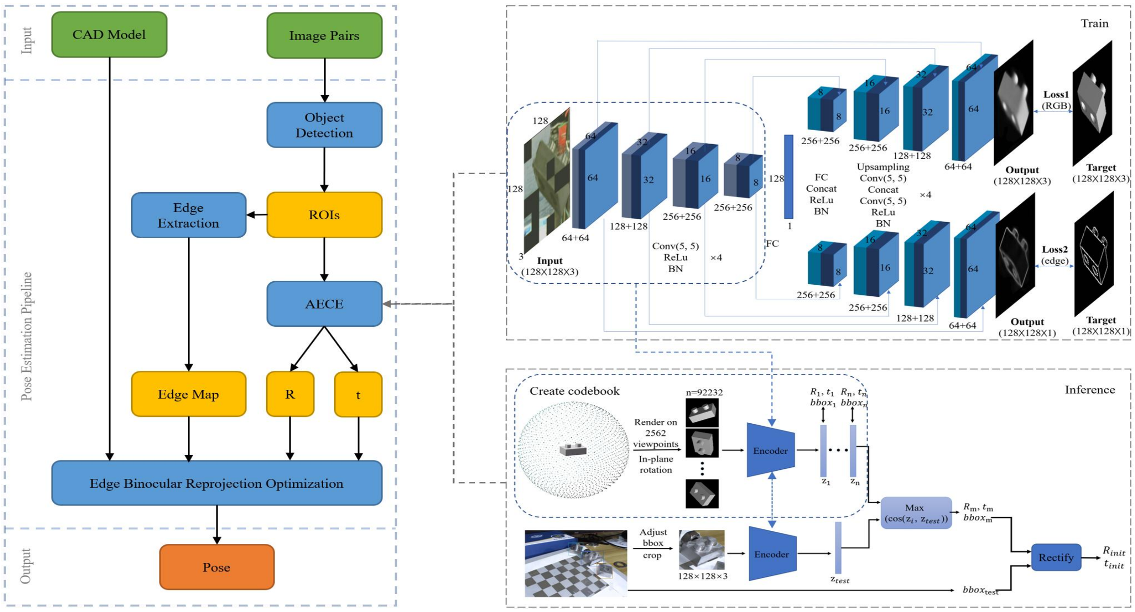

Fig. 2. 6D pose estimation pipeline. By taking binocular image pairs and CAD models of the objects as inputs, our approach involves object detection, initial pose estimation (AECE), and pose refinement (EBRO) processes. Through these steps, we achieve a refined pose estimation with high precision. (Left). The top right dashed box represents the network architecture of AECE, while the bottom right section depicts the inference process, which involves creating the codebook z_{code}.

2. The AECE takes the image patch of object as input to obtain a rough initial pose. In the optimization phase, we extract the edge maps of the image pair and calculate two 3D image tensors based on the edge Directional Chamfer Distance (DCD)[13]. Then, the 3D edge points of the model are extracted and reprojected based on the initial pose for nonlinear optimization, minimizing the cost function based on the binocular 3D tensor to obtain a precise 6d pose. We call it 3D Edge Binocular Reprojection Optimization (EBRO). We demonstrate the improvement of AECE compared to AAE[12] on the T-less dataset[14]. Moreover, we demonstrate the performance of our EBRO on T-less and YCB-Video datasets. We evaluate the precision of our method on our own metal parts dataset. Finally, we deploy our pipeline to the robotic operating system to accomplish stable assembly.

To summarize, our main contributions include: a) We propose AECE based on AAE for initial pose estimation and improve the accuracy by 27.46% on the T-less dataset. b) We propose a novel pose refinement method (EBRO), which can stably achieve 0.48mm average translation precision and 0.45deg average rotation precision. c) We deploy the above pose estimation algorithm to the robotic system and complete a stable assembly task.

II. Related Work

Due to the requirement of robot operation, the object 6d pose estimation has been a long-standing concern. These works can be divided into two categories according to the input conditions: RGB-D methods and RGB methods.

RGB-D methods. Thanks to the development in affordable consumer-level devices, there are more RGB-D methods have been proposed. Some of these methods estimate the initial pose from the RGB images and then employ ICP algorithms to improve accuracy and precision. However, recent methods have been devoted to fusing image and point cloud data. PointFusion[15] utilizes CNN and PointNet to extract features from image and point cloud data, respectively, and employs a fusion module to predict the 3D bounding box. PVN3D[16] introduces a deep Hough voting network for detecting 3D keypoints and estimating pose within a least-squares fitting manner. FFB6D[17] enhances PVN3D by introducing a bidirectional fusion module that integrates texture and geometric information features at each encoding and decoding layer. DenseFusion[18] proposes a novel dense fusion network for pixel-wise feature extraction, enabling pose estimation. Furthermore, they integrate an end-to-end iterative pose refinement procedure that further improves the pose estimation.

RGB methods. Due to some limitations of RGB-D sensors, numerous researchers are actively investigating pose estimation methods tailored specifically for RGB inputs, despite the inherent challenges. Several notable techniques have been developed in this regard. BB8[19] and YOLO6d[6] employ CNNs to directly predict the 2D locations of 3D bounding boxes while PVNet[7] utilizes a pixel-wise voting network to estimate the 2D locations of 3D keypoints. Subsequently, the PnP algorithm[20] is utilized to compute the 6-DoF pose. However, these methods based on 2D-3D correspondence have limitations in handling symmetric objects and rely on richer textures. To address the challenges posed by texture-less and symmetrical objects, AAE[12] employs the Augmented Autoencoder to implicitly encode the 3d rotation into a latent space and predicts the 3d translation using the position and size of the 2d bounding box. Y. Wen et.al[21] further enhance AAE by incorporating an edge decoder and geometric prior. SSD-6D[9] predicts the 2d bounding boxes and object categories, while also classifying object rotations to obtain rough estimates. PoseCNN[22] directly regress the quaternion representation and the 3D translation from the image. Nevertheless, most of these single-network-based methods fall short of directly predicting high-precision 6D pose. As a result, additional pose refinement

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

techniques have gained significant attention in recent years, particularly those leveraging RGB images. Liu et.al[23] introduce the fast directional chamfer matching (FDCM) algorithm, which utilizes edge maps and 3D models to achieve accurate three-dimensional pose estimation. Building upon FDCM, the D2C algorithm[13] employs nonlinear optimization by utilizing a 3D edge distance tensor to optimize the pose. DPOD[24] presents a pose refinement module that extracts features from rendered and real images, encoding the differences between these features to obtain pose differences. Finally, the initial pose is directly combined with these differences to achieve an optimized pose.

These pose refinement methods, which rely on a single RGB image and CAD model, have significantly improved accuracy. However, these monocular methods are sensitive in the depth direction, and a minor pixel error may show significant actual size error in the depth. To address this issue and achieve high-precision estimation, we propose a 3D edge binocular reprojection pose optimization method. This method aims to overcome the depth ambiguity associated with monocular methods by leveraging binocular information, ultimately enhancing the accuracy and precision of the pose estimation.

III. Methods

A. *Overview*

Our proposed pipeline is shown in **Fig. 2**, where any existing detector can be used in the detection phase, so we do not describe it. In this section, our primary emphasis is on detailing our Auto-Encoder of Color image and Edge map (AECE) and 3D Edge Binocular Reprojection Optimization (EBRO).

B. Auto-Encoder of Color image and Edge map

Our AECE is primarily built upon AAE[12]. It encodes the 3D rotation of objects into an implicit vector representation using an encoder-decoder network, comprising an encoder, a color image decoder, and an edge image decoder. Compared to AAE, the main contributions of our AECE include a novel edge decoder and a novel training strategy.

Network. AAE primarily utilizes model-synthesized images to train an encoding network and reconstructs RGB images from the implicit vectors. However, there is always a domain disparity in appearance between synthetic and real images[21], which tends to introduce uncertainty in the encoding of real images, limiting its generalization ability. The edge cues of an object are consistent for both synthetic and real images. Therefore, we add an edge decoder branch to reduce this domain disparity and constrain the training of network. Moreover, using CNN to reconstruct the image poses a challenge in retaining high-frequency information, leading to a potentially blurry output[3, 25]. Inspired by U-Nets[26], to promote the learning of the high-frequency information of the image, we introduce skip connections between equivalent-scale layers allowing high-frequency spatial structure to propagate to the end of the network. Our network architecture is shown in the top right of **Fig. 2**.

Training strategies and details. One of the primary factors limiting the performance of AAE is the disparity between real and synthesized images, resulting in inconsistent encoding of images sharing the same object pose. To better address this issue, we employ a mixture of images for network training, ensuring that synthetic and real images with similar poses yield similar encodings, thereby enhancing the network's ability to generalize to real images. We introduce random black blocks to simulate occlusions, promoting the network's learning of local features and enhancing its robustness to occlusions. Additionally, we employ data augmentation techniques, such as varying image contrast and brightness, to increase data diversity and enhance the network's generalization capabilities. We also insert random background images from the VOC dataset[27]. We utilized the bootstrapped pixel-wise L2 loss with a boot-strap factor of $k = 4$, where only 1/4 of the pixels with the largest reconstruction errors contribute to the loss. In our experiment, the total loss includes color image reconstruction loss and edge map reconstruction loss. The network is trained using TensorFlow with the Adam optimizer. We set the learning rate to 0.0002, the batch size to 64, and perform 50,000 iterations.

Inference. The inference process is shown in the bottom right of **Fig. 2.** In the offline phase, we create a codebook for each object to match the 3D rotation of the test images. Using OpenGL, we render images from 2,652 equidistant spherical viewpoints distributed uniformly around the 3D model of the object. To ensure comprehensive coverage of the entire SO(3) space, each image is rotated 36 times in-plane. Consequently, we obtain a total of 92,232 images and generate their corresponding latent codes $\boldsymbol{z} \in \mathbb{R}^{128}$, forming the codebook $\boldsymbol{z_{code}}$. Additionally, we save the respective 3D rotations $\boldsymbol{R_n}$, 3D translations $\boldsymbol{t_n}$ and 2D bounding boxes for each synthetic object view in the codebook. This takes about 10 minutes.

In the test phase, we encode the test image $\boldsymbol{I_i}$ to obtain the latent code $\boldsymbol{z_i}$ and calculate its cosine similarity with $\boldsymbol{z_{code}}$. We identify the code $\boldsymbol{z_m}$ with the highest similarity and retrieve the corresponding $\boldsymbol{R_m}$, $\boldsymbol{t_m}$, and 2D bounding box $\boldsymbol{bbox_m}$ from the codebook. We consider $\boldsymbol{R_m}$ as the initial estimation $\hat{\boldsymbol{R}}'$. To obtain 3d translation $\hat{\boldsymbol{t}}$, we utilize the pinhole camera model compute the projective distance estimation as described in Eq. (1) and Eq. (2).

$$\hat{t}_z = t_{m,z} \cdot \frac{\|bb_m\|}{\|bb_{real}\|} \cdot \frac{f_{real}}{f_{syn}}. \tag{1}$$

where $\hat{t}_z$, $t_{m,z}$ are the distance of z direction in $\hat{\boldsymbol{t}}$, $\boldsymbol{t_m}$. $\|bb_m\|$, $\|bb_{real}\|$ are the diagonal length of $\boldsymbol{bbox_m}$, $\boldsymbol{bbox_{real}}$, f_{real}, f_{syn} are the focal length of real and synthetic camera.

$$\begin{cases} \boldsymbol{\Delta\hat{t}} = \hat{t}_z \boldsymbol{K}_{real}^{-1} \boldsymbol{bb}_{real,cen} - t_{\mathrm{m},z} \boldsymbol{K}_{syn}^{-1} \boldsymbol{bb}_{m,cen}, \\ \hat{\boldsymbol{t}} = \boldsymbol{t_m} + \boldsymbol{\Delta\hat{t}}. \end{cases} \tag{2}$$

where $\boldsymbol{\Delta\hat{t}}$ is the vector from the synthetic to the real object center, $\boldsymbol{K_{real}}$ and $\boldsymbol{K_{syn}}$ are the intrinsic matrices of real and synthetic camera, $\boldsymbol{bb}_{real,cen}$ and $\boldsymbol{bb}_{m,cen}$ are the centers of $\boldsymbol{bbox_m}$ and $\boldsymbol{bbox_{real}}$.

However, during the generation of the codebook, the object is positioned at the center of the image while the object may not be centered in the image during testing. Therefore, it is necessary to correct the rotation $\hat{\boldsymbol{R}}'$, which is shown in Eq. (3).

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

$\boldsymbol{R}_y(\beta_y)$, $\boldsymbol{R}_x(\alpha_x)$ are the rotation matrices of camera rotating β_y and α_x degrees around its own y and x axes.

$$\begin{cases} \begin{pmatrix} \alpha_x \\ \beta_y \end{pmatrix} = \begin{pmatrix} -\arctan(\hat{t}_y / \hat{t}_z) \\ -\arctan(\hat{t}_x / \sqrt{\hat{t}_z^2 + \hat{t}_y^2}) \end{pmatrix}, \\ \hat{\boldsymbol{R}} = \boldsymbol{R}_y(\beta_y)\boldsymbol{R}_x(\alpha_x)\hat{\boldsymbol{R}}'. \end{cases} \tag{3}$$

C. 3D Edge Binocular Reprojection Optimization

Building upon the initial pose, we formulate pose refinement as a nonlinear optimization problem, aiming to minimize the cost function associated with the directional chamfer distance tensor computed from the binocular edge maps.

3D edge points reprojection. Our method realizes pose refinement by aligning the edge of the object on the binocular images and the 2d reprojected points of the 3d edge sampled from the CAD model. We utilize OpenGL to sample 3d points from the high-curvature regions and outer contour edges of the model with intervals ranging from 1 to 2 mm and obtain the three-dimensional point set $\boldsymbol{\Omega}_M = \{(\boldsymbol{X}_1, \boldsymbol{E}_1), (\boldsymbol{X}_2, \boldsymbol{E}_2), \ldots, (\boldsymbol{X}_m, \boldsymbol{E}_m)\} \in \mathbb{R}^3$ under the object coordinate system, where

$$\boldsymbol{E}_i = \boldsymbol{X}_i + \tau(\boldsymbol{X}_i) \cdot dr. \tag{4}$$

$\tau(\cdot)$ provides the unit tangent vector of the edge to which $\boldsymbol{X}_i$ belongs, and $dr \ll 1$.

According to the initial pose $\boldsymbol{R}$, $\boldsymbol{t}$ obtained previously, we transform the point set into the coordinate systems of the two cameras. We utilize OpenGL z-buffer to process the occluded points and then project visible points onto the image plane according to the intrinsic matrix $\boldsymbol{K}_l$, $\boldsymbol{K}_r$ of the left and right camera to obtain two 2d point sets $\boldsymbol{p} = \{(\boldsymbol{x}_1, \boldsymbol{e}_1), (\boldsymbol{x}_2, \boldsymbol{e}_2), \ldots, (\boldsymbol{x}_{m'}, \boldsymbol{e}_{m'})\}$ and $\boldsymbol{p}' = \{(\boldsymbol{x}_1', \boldsymbol{e}_1'), (\boldsymbol{x}_2', \boldsymbol{e}_2'), \ldots, (\boldsymbol{x}_{m''}', \boldsymbol{e}_{m''}')\}$. The $\boldsymbol{p}_i$ and $\boldsymbol{p}_i'$ projection relation is expressed as:

$$\begin{cases} \boldsymbol{p} = g(\boldsymbol{\Omega}_M, \boldsymbol{K}_l, (\boldsymbol{R}, \boldsymbol{t})) \in \mathbb{R}^2, \\ \boldsymbol{p}' = g(\boldsymbol{\Omega}_M, \boldsymbol{K}_r, (\boldsymbol{R}', \boldsymbol{t}')) \in \mathbb{R}^2. \end{cases} \tag{5}$$

where $\boldsymbol{R}'$, $\boldsymbol{t}'$ are the initial pose of another camera. Since the binocular image pair is used in the optimization, in order to ensure the relative pose relationship among binocular cameras and object, we only select the better initial pose in the image pair (determined by the previous cosine similarity matching score), and the initial pose of another image is transformed according to the binocular camera pose relationship, i.e.

$$\boldsymbol{T}' = \boldsymbol{T}_{l2r} \cdot \boldsymbol{T}. \tag{6}$$

where $\boldsymbol{T}$, $\boldsymbol{T}' \in \mathbb{R}^6$ are the homogeneous transformation matrix of $\boldsymbol{R}$, $\boldsymbol{t}$ and $\boldsymbol{R}'$, $\boldsymbol{t}'$, $\boldsymbol{T}_{l2r}$ is the matrix from left camera to right camera.

Edge distance tensor. We employ Canny operator to extract the edge maps. To capture comprehensive edge information, we extract edges from all three channels of the RGB image and combine them to generate the edge map E_m. Subsequently, we calculate the direction chamfer distance (DCD) tensor $DT3_v$[13] for the edge maps. DCD is obtained by uniformly dispersing the direction of $[0, \pi)$ into n intervals, $\bar{\Psi} = \{\bar{\phi}_1, \bar{\phi}_2, \ldots, \bar{\phi}_n\}$, dividing the edge points into different intervals according to the tangential direction to obtain n edge maps Em_n, and then calculating the distance transformation map for each map. D2C[13] extend $DT3_v$ to encode the minimum distance to an edge point in the joint direction and position. The minimum distance can be recovered as:

$$DT3_v(\boldsymbol{x}_i, \phi(\boldsymbol{x}_i)) = \min_{\varepsilon_j \in Em_n} (\|\boldsymbol{x}_i - \boldsymbol{\varepsilon}_j\| + \lambda \|\bar{\phi}(\boldsymbol{x}_i) - \bar{\phi}(\boldsymbol{\varepsilon}_j)\|). \tag{7}$$

where $\boldsymbol{\varepsilon}_j$ is an edge point, $\phi(\boldsymbol{x}_i)$ is the orientation of the line segment associated with point $\boldsymbol{x}_i$, and $\bar{\phi}(\boldsymbol{x}_i)$ is the nearest quantization level in the direction space $\bar{\Psi}$ to the edge direction $\phi(\boldsymbol{x}_i)$. The calculation of $DT3_v$ is detailed in [13]. We employ this tensor directly. In our experiment, we set $n = 60$ and $\lambda = 100$. We utilize a Gaussian filter to smooth along the direction dimension of the tensor to ensure that the function $DT3_v$ is piecewise smooth.

Pose refinement. We represent initial 6d pose of left camera as the quaternion $\boldsymbol{R} = [r_x, r_y, r_z, r_w]$ and translation vector $\boldsymbol{t} = [t_x, t_y, t_z]$. To refine the pose, we use the 2D points $\boldsymbol{x}_i$, $\boldsymbol{x}_i'$, $\boldsymbol{e}_i$, $\boldsymbol{e}_i'$ in Eq. (5) projected by the 3D edge points of the CAD model according to $\boldsymbol{R}$, $\boldsymbol{t}$, and DCD tensor $DT3_v$ to construct a cost function:

$$E(\boldsymbol{R}, \boldsymbol{t}) = \frac{1}{2}\sum_{i=1}^{m'} DT3_v^{\,2}(\boldsymbol{x}_i, \phi(\boldsymbol{x}_i)) + \frac{1}{2}\sum_{j=1}^{m''} DT3_V^{\,2}(\boldsymbol{x}_j', \phi(\boldsymbol{x}_j')). \tag{8}$$

We assume that the 3D edge points do not change for small viewpoint transformations but their image projections do. Therefore, the direction of the projected points $\phi(\boldsymbol{x}_i)$ need to be update. Define $\boldsymbol{\tau}_i = \boldsymbol{e}_i - \boldsymbol{x}_i \in \mathbb{R}^2$ for both left and right images, where $\boldsymbol{e}_i$ and $\boldsymbol{x}_i$ are calculated previously. Then, $\phi(\boldsymbol{x}_i)$ can be updated as:

$$\phi(g(\boldsymbol{P}_i, \boldsymbol{K}, (\boldsymbol{R}, \boldsymbol{t}))) = \phi(\boldsymbol{x}_i) = \arctan(\frac{\tau_i(1)}{\tau_i(2)}). \tag{9}$$

We aim to find a $\bar{\boldsymbol{R}}$ and $\bar{\boldsymbol{t}}$ to minimize the cost function. So, we apply a non-linear optimization on $E(\boldsymbol{R}, \boldsymbol{t})$, whose derivative can be calculated as:

$$\begin{aligned} \nabla E = & \sum_{i=1}^{m'} DT3_v \nabla DT3_v \nabla(\boldsymbol{x}_i, \phi(\boldsymbol{x}_i)) \\ & + \sum_{j=1}^{m''} DT3_v \nabla DT3_v \nabla(\boldsymbol{x}_j', \phi(\boldsymbol{x}_j')). \end{aligned} \tag{10}$$

where the $DT3_v$ is a discrete tensor, $\nabla DT3_v$ need to be computed in an approximate and numerical way. For x and y derivatives of $\boldsymbol{x}_i$, they are calculated as image derivatives of current distance map. For the derivative of $\phi(\boldsymbol{x}_i)$, it is calculated in a similar way as:

$$\frac{\partial(DT3_v(\boldsymbol{x}, \phi(\boldsymbol{x}_i)))}{\partial(\phi(\boldsymbol{x}_i))} = \frac{DT3_v(\boldsymbol{x}, \bar{\phi}(\boldsymbol{x}_{i+1})) - DT3_v(\boldsymbol{x}, \bar{\phi}(\boldsymbol{x}_{i-1}))}{2} \tag{11}$$

We look up the $DT3_v$ by employing bilinear interpolation. We use Ceres Solver, an open-source library for modeling and solving large, complicated optimization problems, to realize our non-linear optimization. During the process of optimization, we use the Levenberg-Marquardt algorithm and a Huber loss function to reduce the influence of outliers.

IV. EXPERIMENTS

We introduce the datasets and metrics firstly and then some experiments we conduct to verify and evaluate our method.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

TABLE 1
THE PERFORMANCE OF OUR AECE AND AAE WITH SYNTHETIC AND MIXED DATASETS IN THE VSD METRIC ON THE T-LESS

Methods	AAE Syn.	AECE Syn.	AAE mixed	AECE mixed
Mean	38.34	50.50	61.77	**65.80**

TABLE 2
ABLATION STUDY OF COLOR IMAGE DECODER AND EDGE MAP DECODER ON FIVE OBJECTS IN THE T-LESS DATASET

Methods	Color Decoder	Edge Decoder	Both
06	62.7	**64.5**	64.2
10	75.6	70.6	**77.2**
20	46.1	42.5	**48.2**
27	67.5	67.4	**68.8**
30	90.9	89.1	**92.1**
Mean	68.56	66.82	**70.01**

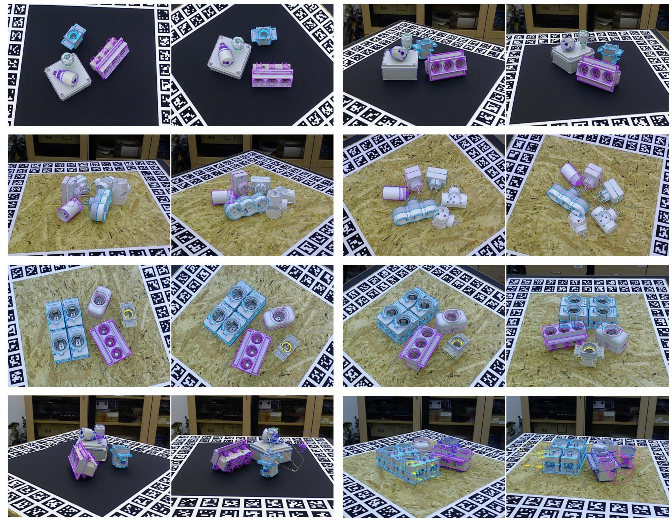

Fig. 3. The qualitative results on T-less. We provide visualizations of example results in three scenes. The ground-truth poses are indicated by navy blue points, while our predictions are represented by other colors. The first three rows are correct examples of three scenarios respectively. Within each row, two images form an image pair. The images of the last row are the failure examples, which are mainly caused by serious occlusion and error detection.

A. Datasets and Metrics

T-less consists of 30 textureless industrial objects, most of which are symmetrical and has no distinguishing color. It provides 39K training images and 10K test images.

YCB-Video consists of 21 objects of varying shape and texture. This dataset is challenging due to the varying lighting conditions, significant image noise and occlusions. we use the frames of 80 video frames for training, and test on 2,949 key frames extracted from the rest 12 test videos.

Our Metal Parts Dataset. To evaluate the precision of our methods, we curate a small test dataset comprising several metal parts. For each object, we prepared approximately 70 high-precision test images to assess the accuracy of our algorithm.

Metrics. For evaluation on the T-less dataset, we employ the VSD[28] metric, which is a widely adopted evaluation metric used by other studies on this dataset. Additionally, on the YCB-Video datasets, we utilize the ADD(-S) AUC[22] metric. In our metal parts dataset, we use the absolute angle error e_R and absolute translation error e_t to quantitatively evaluate the precision:

$$e_R = \arccos(tr(\overline{\boldsymbol{R}}^T \boldsymbol{R} - \boldsymbol{I} / 2)). \tag{12}$$

$$e_t = \sqrt[2]{(\overline{\boldsymbol{t}} - \boldsymbol{t})^2}. \tag{13}$$

where tr(·) denotes the trace of the matrix.

B. Compare with AAE

To validate the enhanced performance of our AECE, we conduct an experimental comparison with AAE on the T-less. In order to mitigate the impact of bounding box predictions, we use the ground truth bounding boxes to crop out all objects with visibility superior to 0.3 for initial pose estimation test.

Table 1 presents a comparison between the results of AECE and AAE. To assess the effectiveness of our edge decoder in mitigating the domain disparity between synthetic and real images, we conduct comparisons using both synthetic and mixed images. As depicted in the **Table 1**, AECE and AAE are trained concurrently with synthetic data, resulting in a 12.16% improvement in AECE results. Furthermore, when trained with mixed images, AECE demonstrate a 4.03% improvement. To confirm the utility of our novel training strategy in reducing domain disparity, we trained AAE and AECE on both synthetic and mixed images. The results indicate a substantial improvement of 23.43% and 15.3% for AAE and AECE, respectively, when trained with mixed images.

To further eliminate the influence of minor network adjustments and validate the role of the edge decoder, we conduct ablation experiments on AECE, separately removing the color encoder and the edge decoder. The results, as shown in **Table 2**, further underscore that the inclusion of our edge decoder, which facilitates the learning of high-frequency edge information, contributes significantly to reducing the encoding differences between synthetic and real images.

C. Evaluation on the T-less and YCB-Video

We conduct evaluations of our pose estimation pipeline on both T-less and YCB-Video datasets. Notably, our EBRO relies on binocular image pairs, we arbitrarily select two images to form an image pair, and calculate the relative pose between the two camera views for estimation.

T-less. We compare our method with several similar RGB-based methods, namely AAE[12], Pix2Pose[29], PVNet[7], SymGAN[30]. To ensure a fair comparison, our experimental setup closely followed the protocols of these methods. Since most of these methods are based on the detections of RetinaNet[31], we utilized the results provided in [29]. For test images with multiple identical objects, we select the one with the highest detection score for pose estimation, i.e., SiSo task[32]. The comparative results are presented in **Table 3**, with the SymGAN results based on ground truth bounding boxes. Our method achieves a significant performance of 54.41%, which surpasses the AAE baseline by 27.62%. Some qualitative results are shown in **Fig. 3.** Through our

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

TABLE 3
THE PERFORMANCE OF OUR METHOD AND THE STATE-OF-THE-ART METHODS IN THE VSD METRIC ON THE T-LESS

Methods	AAE [12]	Pix2Pose [29]	SymGAN + GT [30]	PVNet [7]	Our AECE	Ours	Our AECE +GT	Ours + GT
Mean	26.79	29.5	50.74	40.35	33.29	**54.41**	65.80	**75.86**

TABLE 4
THE ACCURACIES ON THE YCB-VIDEO DATASET IN TERMS OF THE ADD(-S) AUC METRICS

Methods	PoseCNN[22]		Ours	
	ADD	ADD-S	ADD	ADD-S
Mean	53.72	75.9	**58.8**	**82.0**

Fig. 4. The qualitative results on YCB-Video. The ground-truth poses are indicated by navy blue points, while our predictions are represented by other colors. The first two rows are correct examples. The images of the last row are the failure examples, which are mainly caused by serious occlusion.

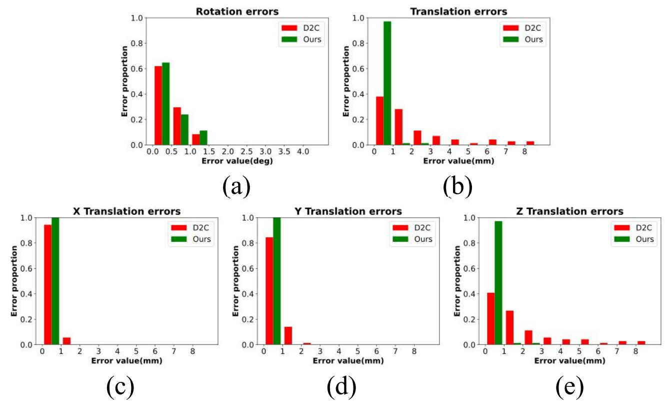

Fig. 5. The error results on Part-3 of our metal parts dataset. The x-axis represents the error value and the y-axis represents the error proportion. The red represents the results of D2C and the green represents ours. (a), (b) illustrate the translation and rotation error proportions, which demonstrates our EBRO achieves higher precision and stability in 3D translation estimation. (c), (d), (e) present the error proportions of the 3D translation in the x, y, and z directions, which shows the sensitivity of monocular method in depth and our EBRO successfully resolves the ambiguity.

experimental analysis, we observe that due to the high similarity among certain objects in the dataset, the detection results from RetinaNet are poor. This leads to substantial errors and mismatches in bounding box regression. As our initial estimation method for 3D translation relies on accurate bounding boxes, this limitation affects our pose estimation performance. Consequently, we also evaluate our method using ground truth bounding boxes for all objects with visibility greater than 0.3. The results demonstrate that our method achieve significant performance improvements when high-quality detections are available.

YCB-Video. We further validate the performance of our pipeline on the textured and severely occluded YCB-Video dataset[22]. We employ the ADD(-S) AUC metric proposed by PoseCNN and compare our results with it. To eliminate the influence of the detector, we utilize the real bounding boxes during evaluation. The experimental results are presented in **Table 4**, demonstrating a 6.1% improvement of our proposed method over PoseCNN. These results indicate that our method is applicable to textured objects as well. Some qualitative results are shown in **Fig. 4.** However, we also observe during our experiments that our method struggles to accurately estimate the precise pose for severely occluded objects. This limitation arises from the reliance of our initial pose estimation network on object shape and edge contour. Moreover, our pose refinement method EBRO directly utilizes edge cues to calculate the directional chamfer distance tensor to complete non-linear optimization. Severe occlusion directly affects the appearance of objects and alters the edge cues present in the image, which poses a challenge for our algorithm in effectively handling such occlusion scenarios. To better discuss the effects of occlusion, we conduct ablation study of occlusion in Part E.

D. Precision Evaluation

Our EBRO aims to address the depth ambiguity in monocular pose estimation and achieve highly precise pose estimation for robot assembly. To validate the precision of our method, we conducted evaluations on our metal parts dataset. For the detection phase, we use the existing YOLOX detector[33].

We perform quantitative evaluation of our method by calculating the absolute translation and rotation errors for the predicted poses. Specifically, we compute the mean errors and the percentages of errors less than certain thresholds for each of the four objects. To provide a comparative analysis, we compare our optimized results with the monocular method D2C[13] using the same initial pose, as presented in **Table 5**. **Table 5(A)** displays the translation errors, while **Table 5(B)** shows the rotation errors. The results reveal that our method achieves an average translation precision of 0.48mm, rotation precision of 0.45 degrees, and a percentage of errors less than 1mm reaching 94.05%, surpassing the performance of [13], which only achieves 37.47%. **Fig. 5** presents a visual comparison between our method and D2C. Additionally, to further analyze the translation errors along the x, y, and z axes of the camera, we separately analyzed the translation errors of Part-3 in each direction, as depicted in **Fig. 5(c)**, **(d)**, and **(e).** It is evident that the errors in the monocular method primarily originate from the translation error along the z-axis, indicating the sensitivity of monocular methods to depth estimation. In

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

TABLE 5
THE ABSOLUTE TRANSLATION AND ANGLE ERRORS OF AECE, EBRO AND D2C ON OUR METAL PARTS DATASET

(A) ABSOLUTE TRANSLATION ERROR

Parts name	AECE	EBRO				D2C[13]			
	Mean error(mm)	Mean error(mm)	< 0.5 mm (%)	< 1 mm (%)	< 2 mm (%)	Mean error(mm)	< 0.5 mm (%)	< 1 mm (%)	< 2 mm (%)
Part-1	8.44	0.36	83.78	97.30	100.00	2.61	14.86	24.32	50.68
Part-2	7.19	0.59	65.67	91.04	95.45	1.53	10.45	32.84	75.76
Part-3	8.14	0.41	77.46	97.18	98.59	2.10	21.13	38.03	66.22
Part-4	8.79	0.57	46.67	90.67	98.67	1.40	28.00	54.67	77.33
Mean	8.14	**0.48**	68.65	94.05	97.93	1.91	18.61	37.47	67.50

(B) ABSOLUTE ANGLE ERROR

Parts name	AECE	EBRO				D2C[13]			
	Mean error(deg)	Mean error(deg)	< 0.5 deg (%)	< 1 deg (%)	< 2 deg (%)	Mean error(deg)	< 0.5 deg (%)	< 1 deg (%)	< 2 deg (%)
Part-1	2.68	0.41	71.23	91.78	100.00	0.55	56.16	86.30	98.63
Part-2	3.18	0.48	66.67	93.94	100.00	0.53	51.52	93.94	98.48
Part-3	2.99	0.37	73.24	92.96	98.59	0.47	61.97	91.55	100.00
Part-4	3.40	0.52	60.00	86.67	98.67	0.71	42.67	76.00	97.33
Mean	3.06	**0.45**	67.79	91.34	99.32	0.57	53.08	86.95	98.61

contrast, our method effectively corrects inaccuracies in depth and achieves high-precision pose optimization.

E. Ablation studies

Ablation study of the EBRO. We compare the accuracy before and after using EBRO on the T-less dataset and the results are presented in the last four columns of **Table 2**. The results show that our EBRO improved by 22.12% accuracy when using predicted bounding boxes and by 10.06% accuracy when using real bounding boxes. Moreover, we compare the precision of estimation before and after using EBRO on our metal parts dataset. The results, as shown in the first two columns of **Table 5**, demonstrate that our EBRO significantly improves the precision of pose estimation. Some qualitative results are shown in **Fig. 6**.

Ablation study of the occlusion. To investigate the impact of occlusion ratios on our AECE and EBRO, we conduct an ablation study on our metal parts dataset. We introduce random noise to simulate occlusions and tested occlusion ratios of 0%, 10%, 20%, 30%, 40%, 50%, and 60%. The results for AECE indicate translation errors of 8.14 mm, 8.62 mm, 9.81 mm, 9.85 mm, 11.34 mm, 14.87 mm, and 40.31 mm, along with angle errors of 3.06°, 3.26°, 3.75°, 4.26°, 4.49°, 5.38°, and 9.15°. As for EBRO, the results show translation errors of 0.48 mm, 0.68 mm, 0.74 mm, 1.18 mm, 1.69 mm, 2.51 mm, and 11.94 mm, coupled with angle errors of 0.45°, 0.62°, 0.85°, 1.07°, 1.65°, 2.08°, and 5.48°. It can be observed that the errors of AECE and EBRO increase as the occlusion percentage rises. However, when occlusion is below 50%, the increase in error is not significant, demonstrating the robustness of our AECE and EBRO to occlusion. Conversely, in cases of severe occlusion, exceeding 50%, the error increase becomes more pronounced.

More detailed data and analysis can be found in the supplementary material.

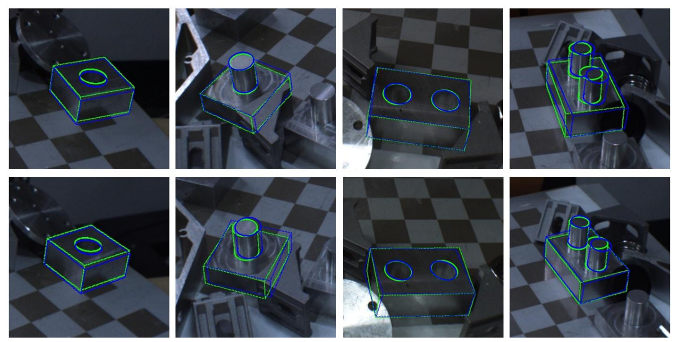

Fig. 6. Some qualitative results on our metal parts dataset. The blue point set represents the projection of initial pose while the green represents the refined pose. In each column, two images form an image pair. The results show the enhances of precision with our EBRO.

F. Running Time

We measure the average runtime for each phase using 20 images of size 1,292×964 on a computer equipped with an Intel i7 CPU and an RTX TITAN GPU. The detection phase exhibits a runtime of 23.44 ms, the pose initialization takes 9.6 ms, the pose optimization requires 745.41 ms. However, this runtime is sufficient for the robotic assembly process.

G. Robotic Assembly

To further demonstrate the practicality of our method, we deployed it in our robotic system to successfully assemble part Part-3 and Part-4, where there is a 2 mm difference in diameter between the shaft and the hole. We randomly place Part-3 and Part-4 in the cluttered scene, as shown in **Fig. 1**.

We primarily focus on the problem of 6D pose estimation. For the purpose of achieving simple grasping and assembly tasks, we utilize predefined grasping coordinates that have fixed transformations with respect to the object coordinates. The robot is then moved to the assembly coordinates, which are determined based on the estimated pose of Part-3 and the predefined assembly pose relationship between Part-3 and Part-4. We conduct 20 experiments and successfully complete 17 of them. A related demo video can be seen in the supplement.

V. CONCLUSION

In this letter, we propose an improved initial pose estimation network AECE and a high precision pose refinement method EBRO. The AECE improves the accuracy of the initial pose estimation by reducing the difference between real and

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

synthetic images with a new edge decoder and a mixed dataset training strategy. The EBRO significantly improves the precision of the pose by minimizing the distance between the binocular reprojected 3D edge points and the edge map. Furthermore, we deploy our pipeline to the robotic assembly system and achieve robust assembly.

Limitations and future work. The efficacy of our initial pose estimation network and pose optimization method is contingent upon the availability of complete object shapes and edge cues, limiting their effectiveness in scenarios characterized by significant occlusions. In the future, we are committed to developing a more powerful methodology to overcome the challenges imposed by severe occlusions.

REFERENCES

[1] T. Cao, F. Luo, Y. Fu, W. Zhang, S. Zheng, and C. Xiao, "DGECN: A Depth-Guided Edge Convolutional Network for End-to-End 6D Pose Estimation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 3773-3782.

[2] Y. Shi, J. Huang, X. Xu, Y. Zhang, and K. Xu, "StablePose: Learning 6D Object Poses from Geometrically Stable Patches," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 15217-15226.

[3] K. Park, A. Mousavian, Y. Xiang, and D. Fox, "LatentFusion: End-to-End Differentiable Reconstruction and Rendering for Unseen Object Pose Estimation," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10707-10716.

[4] L. Liu, H. Xue, W. Xu, H. Fu, and C. Lu, "Toward Real-World Category-Level Articulation Pose Estimation," *IEEE Transactions on Image Processing,* vol. 31, pp. 1072-1083, 2022.

[5] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2637-2646.

[6] B. Tekin, S. N. Sinha, and P. Fua, "Real-Time Seamless Single Shot 6D Object Pose Prediction," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 292-301.

[7] S. Peng, X. Zhou, Y. Liu, H. Lin, Q. Huang, and H. Bao, "PVNet: Pixel-Wise Voting Network for 6DoF Object Pose Estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 44, no. 6, pp. 3212-3223, 2022.

[8] C. Song, J. Song, and Q. Huang, "HybridPose: 6D Object Pose Estimation Under Hybrid Representations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 428-437.

[9] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1530-1538.

[10] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "DeepIM: Deep Iterative Matching for 6D Pose Estimation," *International Journal of Computer Vision,* Article vol. 128, no. 3, pp. 657-678, Mar 2020.

[11] S. Iwase, X. Liu, R. Khirodkar, R. Yokota, K. M. Kitani, and Ieee, "RePOSE: Fast 6D Object Pose Refinement via Deep Texture Rendering," in *18th IEEE/CVF International Conference on Computer Vision (ICCV)*, Electr Network, 2021, pp. 3283-3292, 2021.

[12] M. Sundermeyer, Z. C. Marton, M. Durner, and R. Triebel, "Augmented Autoencoders: Implicit 3D Orientation Learning for 6D Object Detection," (in English), *INTERNATIONAL JOURNAL OF COMPUTER VISION,* vol. 128, no. 3, pp. 714-729, MAR 2020.

[13] M. Imperoli and A. Pretto, "(DCO)-C-2: Fast and Robust Registration of 3D Textureless Objects Using the Directional Chamfer Distance," presented at the COMPUTER VISION SYSTEMS (ICVS 2015), 2015, 2015. Proceedings Paper.

[14] T. Hodan, P. Haluza, O. Š, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D Dataset for 6D Pose Estimation of Texture-Less Objects," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 880-888.

[15] D. Xu, D. Anguelov, and A. Jain, "PointFusion: Deep Sensor Fusion for 3D Bounding Box Estimation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 244-253.

[16] Y. He, W. Sun, H. Huang, J. Liu, H. Fan, and J. Sun, "PVN3D: A Deep Point-Wise 3D Keypoints Voting Network for 6DoF Pose Estimation," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11629-11638.

[17] Y. He, H. Huang, H. Fan, Q. Chen, and J. Sun, "FFB6D: A Full Flow Bidirectional Fusion Network for 6D Pose Estimation," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 3002-3012.

[18] C. Wang *et al.*, "DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3338-3347.

[19] M. Rad and V. Lepetit, "BB8: A Scalable, Accurate, Robust to Partial Occlusion Method for Predicting the 3D Poses of Challenging Objects without Using Depth," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 3848-3856.

[20] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An Accurate O(n) Solution to the PnP Problem," *INTERNATIONAL JOURNAL OF COMPUTER VISION,* vol. 81, no. 2, pp. 155-166, FEB 2009.

[21] Y. Wen, H. Pan, L. Yang, and W. Wang, "Edge Enhanced Implicit Orientation Learning With Geometric Prior for 6D Pose Estimation," *IEEE Robotics and Automation Letters,* vol. 5, no. 3, pp. 4931-4938, 2020.

[22] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes," presented at the ROBOTICS: SCIENCE AND SYSTEMS XIV, 2018.

[23] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T. K. Marks, and R. Chellappa, "Fast object localization and pose estimation in heavy clutter for robotic bin picking," *INTERNATIONAL JOURNAL OF ROBOTICS RESEARCH,* Article vol. 31, no. 8, pp. 951-973, 2012 JUL 2012.

[24] S. Zakharov, I. Shugurov, and S. Ilic, "DPOD: 6D Pose Object Detector and Refiner," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1941-1950.

[25] L. Yen-Chen, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T. Y. Lin, "iNeRF: Inverting Neural Radiance Fields for Pose Estimation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 1323-1330.

[26] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Munich, GERMANY, 2015, vol. 9351, pp. 234-241, CHAM: Springer International Publishing Ag, 2015.

[27] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *INTERNATIONAL JOURNAL OF COMPUTER VISION,* vol. 88, no. 2, pp. 303-338, JUN 10 2010.

[28] T. Hodan, J. Matas, and S. Obdrzalek, "On Evaluation of 6D Object Pose Estimation," presented at the COMPUTER VISION - ECCV 2016 WORKSHOPS, PT III, 2016.

[29] K. Park, T. Patten, and M. Vincze, "Pix2Pose: Pixel-Wise Coordinate Regression of Objects for 6D Pose Estimation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 7667-7676.

[30] P. Ammirato, J. Tremblay, M. Y. Liu, A. C. Berg, and D. Fox, "SymGAN: Orientation Estimation without Annotation for Symmetric Objects," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 1657-1666.

[31] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 42, no. 2, pp. 318-327, 2020.

[32] T. Hodan *et al.*, "BOP: Benchmark for 6D Object Pose Estimation," presented at the COMPUTER VISION - ECCV 2018, PT X, 2018.

[33] Z. Ge, S. T. Liu, F. Wang, Z. M. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," *arXiv preprint arXiv:2107.08430,* 2021.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.