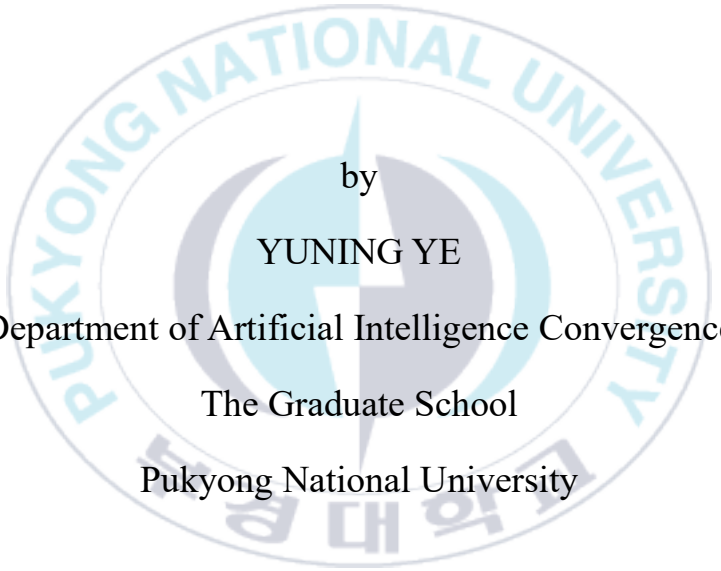
Thesis for Degree of Master of Engineering

Robust 6D Pose Estimation of Occluded Objects Using RGB Images



by

YUNING YE

Department of Artificial Intelligence Convergence

The Graduate School

Pukyong National University

February 2024

Robust 6D Pose Estimation of Occluded Objects Using RGB Images

RGB 영상을 사용한 가려진 객체의 강인한 6D 포즈 추정

Advisor: Prof. Hanhoon Park

by

YUNING YE

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Engineering in Department of Artificial Intelligence Convergence, The Graduate School, Pukyong National University

February 2024

Robust 6D Pose Estimation of Occluded Objects Using RGB Images

A thesis

by

YUNING YE

Approved by:

Professor WanYoung Chung
(Chairman)

Professor SeungWook Kim
(Member)

Professor Hanhoon Park
(Member)

February 16, 2024

ABSTRACT

The 6D pose estimation problem involves both object detection and the determination of their translations and rotations. In three-dimensional space, these properties each have three degrees of freedom, collectively referred to as the 6D pose. Specifically, the term encompasses the translational and rotational motions of a rigid body along the x, y, and z axes in a three-dimensional Cartesian coordinate system. The challenge in object pose estimation lies in determining the translation and rotation of an object. Translation refers to the displacement along the three orthogonal coordinate axes-x, y, and z. Rotation involves the angle of rotation around these three right-angle coordinate axes, encompassing pitch, yaw, and roll operations.

Estimating the pose of objects is crucial for enabling machines to interact or manipulate them effectively. This capability finds applications in various domains, including augmented reality, virtual reality, autonomous driving, robotics, and more. However, addressing the associated challenges is non-trivial, involving issues such as cluttered backgrounds, occlusions, untextured objects, and scenarios where images are not readily available. In such cases, minor variations in rotation, translation, or scaling can pose challenges in accurate pose estimation. In industrial settings, robots leverage 6D pose technology to estimate and manipulate objects accurately. In augmented reality applications, 6D pose estimation plays a crucial role in measuring the poses of real-world objects, enabling the seamless integration of virtual objects into the environment with correct spatial positioning. This capability enhances the overall functionality and immersive experience of augmented reality applications.

6D pose estimation involves leveraging discernible details from a reference 2D image to deduce the 3D rotation and 3D translation of an object in relation to the known shape of the camera. Typical cameras used for capturing scenes include RGB cameras and RGBD (color and depth) cameras. However, the use of depth cameras is not always feasible, especially outdoors where they can be prone to failure due to lighting conditions. Therefore, there is a preference to rely solely on color images for 6D pose estimation, even though this poses additional challenges. Despite the challenges, it is crucial to address occlusion—the partial visibility of the object due to obstructions-which significantly impacts pose detection. Occlusion complicates the inference of an object's position as only a portion of the object is visible. As a result, estimating a 6D pose is not always straightforward, and overcoming occlusion remains an important consideration in the development of accurate pose estimation systems.

This thesis introduces a methodology for estimating the 6D pose of an occluded object using only RGB images. The approach employs a neural network to identify keypoints by predicting vectors that point to these keypoints for each pixel in the RGB image. These vectors are generated through the prediction of pixels for semantic segmentation. The accuracy of localizing keypoints crucially depends on the results obtained for the target pixels. The neural network automatically adjusts the weights of the pixels based on its prediction results, enhancing the network's learning capability, especially in occluded regions. Consequently, our approach excels at extracting features from occluded regions, ensuring robust performance even in cases of object occlusion. Comparative analysis against existing approaches demonstrates that our method achieves higher accuracy in 6D pose estimation.

ACKNOWLEDGMENTS

I wish to take this moment to convey my sincere appreciation to all those who played a pivotal role in my successful completion of the master's program.

Foremost, my heartfelt gratitude extends to Prof. Hanhoon Park, whose guidance and unwavering support defined my academic journey. His encouragement and mentorship were instrumental in shaping my goals and achievements. I also want to express profound thanks to Prof. Seonhan Choi for recommending me to Prof. Park, facilitating the culmination of my master's studies. My gratitude extends to the members of the IVC Lab, whose invaluable assistance, camaraderie, and shared experiences enriched both my academic and personal life.

Additionally, I am deeply thankful to my family for their steadfast support and understanding throughout my overseas study, particularly during challenging moments. Their encouragement and endorsement of my decision to pursue studies abroad have been invaluable.

Lastly, I extend my thanks to Prof. YoonTae Kim and Prof. Jaehyo Jung for their special care during my graduate studies. Their guidance has left an indelible mark on my academic journey.

TABLE OF CONTENTS

LIST OF TABLES

TABLE PAGE

LIST OF FIGURES

FIGURE PAGE

CHAPTER I

INTRODUCTION

The 6D pose of an object is characterized by its translation and rotation vectors. Specifically, an object's 6D pose is described by a 3D rotation matrix R belonging to the special orthogonal group SO *(3)* and a 3D translation vector $T = [t_x, t_y, t_z]^T$. This translation vector represents the displacement between corresponding points $X_o = [x_o, y_o, z_o]^T$ of the object in its own coordinate system $\{O\}$ and the camera's coordinate system $\{C\}$ with corresponding points $T = [t_x, t_y, t_z]^T$. The 6D pose captures both the rotational and translational aspects of the object's spatial orientation. All points in a rigid object have the same rotation and translation matrices, which we call transformation matrices. In general, the object pose in the camera is parameterized to other forms. For example, the rotation matrix is usually represented as a quaternion. As shown in Figure 1, 3D translation and rotation are expressed as $T = [x_t, y_t, z_t]^T$ and $R = R_z(\alpha)R_y(\beta)R_x(\gamma)$, respectively. The former is denoted by the origin of the object coordinate system within the camera coordinate system, while the latter represents the angles defining the orientation of the object coordinate system with respect to the camera coordinate system axes. The relationship between a point X_o in the object coordinate system $\{O\}$ and the corresponding point X_c in the camera coordinate system $\{C\}$ can be expressed as follows:

$$X_c = [R \mid T]X_o , \tag{1-1}$$

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathrm{k}\,[\,R\;T\,] \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix}, \tag{1-2}$$

Where k is camera intrinsic and u, v, z_c refer to object coordinates in camera system, R is:

$$R = R_Z(\alpha)R_y(\beta)R_x(\gamma)\ , \tag{1-3}$$

$$= \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}, \tag{1-4}$$

$$= \begin{bmatrix} \csc\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \csc\alpha\ \cos\beta & \cos\alpha\sin\beta\sin\gamma + \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma - \sin\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix}, \tag{1-5}$$

where *α, β, γ* are the angles between the axis of the object coordinate system and the axes of the corresponding camera coordinate system when the points of the object coordinate system are converted to the points of the camera coordinate system. R_Z, R_y, R_x refer to the correspondence rotation matrix. In recent years, there has been a growing use of pose estimation, particularly in applications like Augmented Reality (AR). The information provided, including the 6D pose and scale of objects, proves valuable for integrating objects into virtual environments. Additionally, pose estimation models play a crucial role in robot grasping, and prominent challenges like the Amazon Challenge [1] heavily rely on 6D pose estimation. Moreover, the precise pose estimation constraints imposed on the camera object have a profound impact on elevating the performance of object-oriented Simultaneous Localization and Mapping (SLAM). This is particularly significant as 3D detection plays a pivotal role in advancing the development of autonomous driving techniques for motor vehicles. The synergy between accurate pose estimation and SLAM contributes to a more robust unde

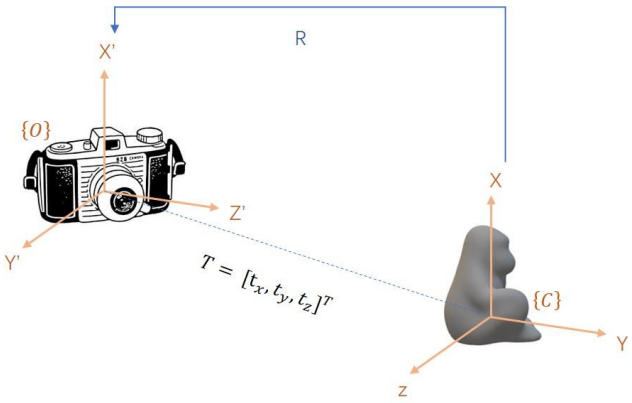


Figure 1. *Illustration of pose definition. {C} is camera frame, {O} is object frame, R is the rotation matrix, T is the translation.*

-rstanding of the environment, while advancements in 3D detection are instrumental in enhancing safety and efficiency in autonomous driving technologies.

Although the research on 6D pose estimation techniques has made great progress in recent years, it is still very challenging when encountering problems such as background clutter, large changes in viewpoints and lighting, or small scene textures in real scenes. While robustness and performance need to be improved, the available solutions are also very limited.

To enhance the performance of pose estimation, an increasing number of studies explore the utilization of texture information, geometric data, color information, and more to estimate 6D pose. For example, these studies often involve the indirect prediction of one or more intermediate representations from such information to estimate the pose, resulting in enhanced robustness. RGB-D cameras, while effective, pose challenges due to their difficulty and expense of use, intricate calibration processes, and the potential drawbacks of heavy and sensitivity to external factors like lighting

conditions, making them prone to failure. As a result, a pose estimation method that relies solely on RGB images becomes a more practical and preferable choice.

In recent years, the widespread attention to 6D pose estimation algorithms based on deep learning can be attributed to the rapid advancements in deep learning and neural network technology. Certain methods in this domain establish 2D-3D correspondence keys between images by generating keypoints for localization on the object's surface. Subsequently, these correspondences are utilized to compute the 6D pose [2, 3]. Usually, these methods follow two main steps: 1. Estimating the 2D keypoint coordinates in the input image; and 2. Utilizing the PnP algorithm (Perspective-n-Points) to calculate the final 6D pose results.

In this thesis, we introduce a novel method for estimating the 6D pose of occluded objects. Notably, our approach involves dynamically adjusting the weights within the neural network, compelling it to learn additional features of the target. This innovative technique enables the inference of prediction the object's pose even when it is heavily occluded, relying on information from the visible portions.

CHAPTER II

OVERVIEW OF ESTIMATING 6D POSE OF OCCLUDED OBJECTS USING KEYPOINT FROM RGB IMAGE

II.1. Background

Keypoints refer to points in an image that encapsulate what is considered interesting or salient in the visual content. They possess invariance to image rotation, contraction, translation, distortion, and other transformations. 6D pose estimation methods based on keypoints leverage local features extracted from all pixels within a specified region of interest or image. These features are compared to those on the 3D model, establishing a 2D-3D match. The overall process involves two key phases: in the first phase, local features are extracted and compared with the 3D keypoints; in the second phase, 2D-

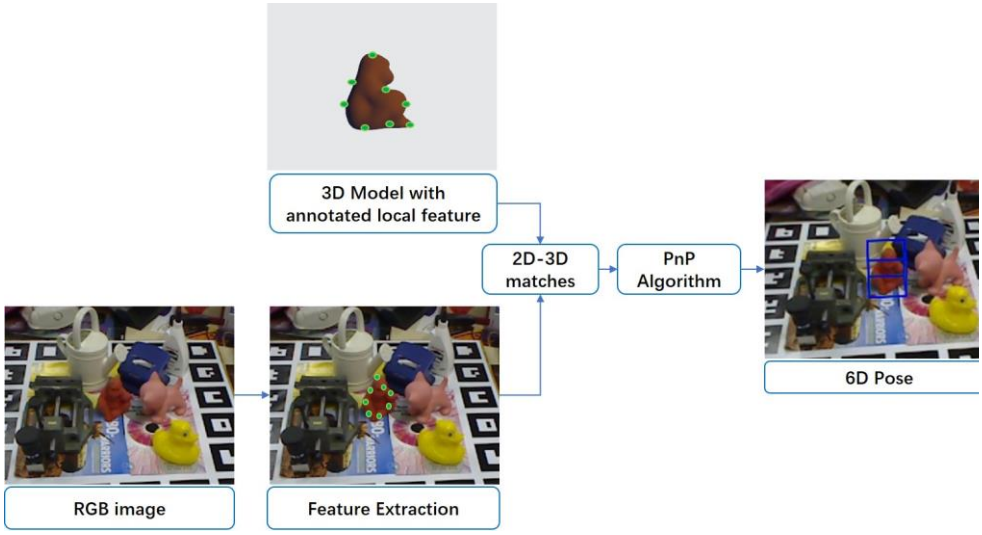


Figure 2. *Schematic diagram of the keypoint-based pose estimation method. At first, extracting features from images is crucial, Features are compared with the annotated 3D model to find matches, and the PnP algorithm estimates the object's 6D pose, enhancing robustness in handling images.*

3D mapping is employed to solve the geometric problem, and the 6D pose is determined using the PnP algorithm. Figure 2 provides a schematic representation of this process.

The methodology underlying keypoints-based 6D pose estimation involves predefining a set of keypoints on the object's surface, serving as specific positional indicators. The RGB image captures the authentic scene of the object through camera detection, with the neural network predicting the 2D keypoint positions in the RGB image. This prediction step is typically facilitated by the neural network. Finally, the 6D pose is computed using the Perspective-n-Points (PnP) algorithm, considering the correspondence between 2D-3D point pairs. This comprehensive approach enables accurate and robust pose estimation in various applications.

II.2. Challenges of 6D pose estimation

II.2.1. Texture-less objects

Estimating the pose of texture-less objects presents a distinct challenge within the realm of pose estimation. Due to the absence of reliable texture information, conventional pose estimation methods reliant on surface feature extraction encounter difficulties in extracting meaningful information features. Given the prevalence of texture-less objects in real industrial scenes, enhancing the performance of pose estimation for such objects is imperative.

The researchers introduced a novel 6D pose estimation framework called Pix2Pose [4].

This framework utilized an untextured 3D model during training to robustly regress pixel-level 3D coordinates of the target from RGB images. The motivation behind this pose estimation method stems from the inherent difficulty in constructing 3D models with precise textures, particularly without expert knowledge or specialized scanning equipment in real-world scenarios. In the context of robot manipulation, capturing an unknown object demands a three-dimensional scene. Employing a convolutional neural network facilitates the identification of grasping points within depth images. Schaub et al. [5] expanded the initial algorithm through the integration of real and virtual viewpoints, projecting anticipated grasping quality information onto object surfaces. This innovative method merges semantic insights from human-labeled datasets with geometric object analysis, enhancing the capability to achieve dependable grasps for various unknown objects.

Figure 3. *Cases of inaccurate and failed detection due to occlusion and cluttered background. The green and blue bounding boxes represent the gound truth and the prediction results respectively.*

II.2.2. Occluded objects

In complex scenes, object occlusion occurs frequently. The extraction of target features is interfered because the target is occluded by other objects or self-objects. In addition, it is hardly to estimate the accurate pose result due to the missing information of some objects.

Paul et al. [6] proposed an approach where the discriminative features of the object images associated with the corresponding 3D poses are first learnt and then pose estimation can be achieved by simply retrieving the similarity of the input images to the templates. Nevertheless, while effective for lightly occluded objects, this method is susceptible to detection failure in the case of heavy occlusion. Additionally, the template matching method proves sensitive to cluttered backgrounds. Recent research has showcased the dominance of using keypoints as intermediate representations for pose prediction. These keypoints encapsulate specific positional coordinates of the object, providing a more robust approach. The system in [7] proposes a process that includes object detection, keypoint localization, and pose refinement. Peng et al. [8] proposed Pixel-by-Pixel Voting Network (PVNet), a method to predict 2D-3D correspondence by generating vectors pointing to the key pixel by pixel. The generated keypoint hypotheses are filtered by the spatial probability distribution of the keypoints, a process they call voting, and then the estimated pose results are obtained using the PnP algorithm. The method of estimating keypoints by direction vectors is effective for pose estimation of most occluded objects.

II.2.3. Background clutter

Indeed, background clutter poses a significant challenge in the context of 6D pose estimation. The presence of complex or distracting backgrounds can interfere with the accurate localization and identification of objects in an image, making it more difficult for the pose estimation algorithms to precisely determine the position and orientation of the objects in three-dimensional space. Strategies and algorithms need to be robust enough to handle various background scenarios and still provide reliable pose estimates. Directly estimating the 6D pose poses a challenge due to the abundance of irrelevant information surrounding the target. However, in real-world scenarios, there are frequent instances where it is imperative to measure the 6D pose of objects within cluttered environments.

He et al. [9] proposed Mask R-CNN for accurate object detection. This method generates segmentation masks while detecting objects in the image, which is a very efficient method for object detection. In [27], They expanded upon the investigation of Mask R-CNN by incorporating an object mask prediction branch alongside the existing bounding box recognition branch. Some previous studies have proposed methods for estimating 6D pose using clutter, exploiting the relational nature of scene-level physical interactions to that well predict the accuracy of the pose. The similarity between the given depth and the predicted generated scene renderings is then used as a criterion to search through Monte Carlo Search (MCTS) for the best candidate pose combinations.

II.3. Algorithms for estimating 6D pose from RGB image

II.3.1. Overview

An RGB image, also known as a color image, is stored as an array of $M \times N \times 3$ data, delineating the red, green, and blue components for each pixel. This color representation is fundamental in various applications, serving as the base color space for devices like cameras, monitors, and scanners. Its versatility enables seamless display without the necessity for conversion, making RGB images ubiquitous in modern technology.

Estimating pose using only RGB images is one of the most challenging tasks in 6D pose estimation. The shape and geometry of the object are indispensable information for pose estimation, and the pose can be inferred from the shape of the object regardless of its appearance. Estimating the pose using only RGB images implies that all the information that can be captured comes from the RGB images, so a large number of RGB images for training is essential. Using an annotation approach to infer poses is robust but the annotation process is more time-consuming and costlier. For example, in the Pix3D dataset [10], keypoints must first be labelled between the collected 3D CAD models and the RGB images. The final 6D pose is obtained by utilizing the 2D-3D correspondence through the Efficient Perspective-n-Point (EPnP) algorithm [11]. This step is a common procedure in most keypoint-based two-stage 6D pose solving algorithms.

Furthermore, the continuous exploration of this field is motivated by the widespread prevalence of RGB datasets, the straightforward accessibility of real-world RGB

images, and the ongoing efforts to overcome the associated challenges.

II.3.2. Template-based methods

Template matching involves generating a template dataset by rendering a 3D CAD model and then comparing original images with these templates to identify the most similar one. The pose of the most similar template is then utilized as the final result, typically determined by a pixel-by-pixel comparison between the original image and the template. Fundamentally, in the 6D pose estimation method using template matching, the goal is not to directly calculate the pose but rather to search for the template with the highest similarity to the original image. The pose associated with this most similar template is then returned as the final estimation result.

It is hardly possible to process 3D data directly because it is too large and computationally expensive. Therefore, some researches aiming at reducing the complexity of such tasks have emerged. Nguyen et al. [12] present GigaPose, utilizing discriminative templates for out-of-plane rotation recovery and patch correspondences for the remaining parameters. Unlike conventional methods, GigaPose samples templates in a two-degree-of-freedom space, achieving a 38x speedup with fast nearest neighbor search in feature space. Additionally, GigaPose exhibits increased robustness to segmentation errors. Since YOLO [13] is time consuming, a 2D bounding box is used to segment the field point cloud. Furthermore, a proposed method involves extracting keypoints from the template point cloud by retaining points with more informative features, such as edges, while eliminating other surface points to compress

the overall point cloud. In [14], researchers enhance LINEMOD's template-based detection and pose estimation for texture-less objects to improve robustness in the presence of partial foreground occlusions. They divide the template into four parts, independently matching each. Using an image pyramid searching method accelerates template matching, boosting the accuracy of fine pose estimation. Experimental results demonstrate increased robustness, particularly in scenarios with partial foreground occlusions.

Due to these limitations mentioned above, the latest template matching based pose estimation methods are dedicated to address the effects of factors such as occlusion. Yann et al. [15] introduce MegaPose, a method for estimating the 6D pose of novel objects unseen during training. At inference, it requires only a region of interest and a estimation methods are dedicated to address the effects of factors such as occlusion. Yann et al. [15] introduce MegaPose, a method for estimating the 6D pose of novel

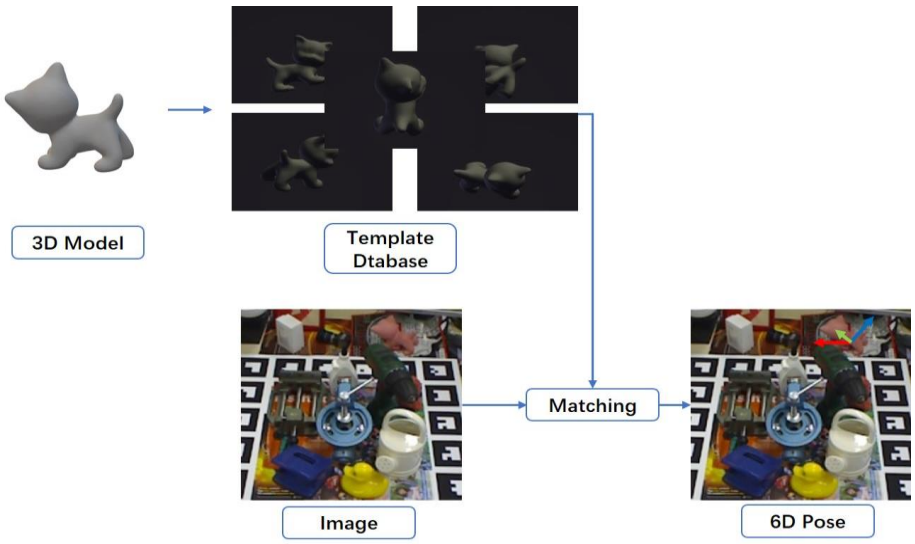


Figure 4. *Flowchart of template-based method for 6D pose estimation. The final result is obtained by returning the pose of the most similar template.*

objects unseen during training. At inference, it requires only a region of interest and a CAD model of the observed object. They present a 6D pose refiner using a render-and-compare strategy, taking inputs of the novel object's shape and coordinate system through synthetic views of its CAD model.

Disadvantages:

- Such methods are very sensitive to changes in external light and occlusion or self-occlusion between objects. Therefore, when these noises are present in the objects, the calculated similarity is affected and it is difficult to get accurate comparison results.
- The speed of the run is influenced by the number of templates; a higher number improves pose accuracy but slows down the process. A rich template database is crucial to enhance the probability of finding the correct pose.

II.3.3. Regression-based methods

These methods regress the 6D pose parameters directly from the input image, and an object detector is usually used to obtain a preliminary object position prior to pose regression. This class of methods is categorized as single-stage methods, where the pose problem is solved by designing a neural network to receive the input images for training, and then learning the 3D translations and 3D rotations of the represented objects.

PoseCNN [9] is a notable approach for pose estimation in RGB images, utilizing a

comprehensive Convolutional Neural Network (CNN) with two stages. This CNN performs object segmentation, estimates rotation of multimedia tools, and determines distance from the camera. The network extracts and integrates feature maps from input images, providing outputs of semantic labels, 3D translations, and 3D rotations. Despite its effectiveness, PoseCNN faces challenges with input images containing multiple instances of the same object and may require refinement for improved accuracy. Another approach [18] extends previous work on semantically segmenting cluttered bin-picking scenes to isolate individual objects. An additional network, trained on synthetic scenes, estimates object poses from a cropped, object-centered encoding based on segmentation results. The proposed method is evaluated on synthetic validation data and real-world cluttered scenes.

Some recent work has used set prediction to improve the speed of multi-object pose estimation, which is basically based on Transformer unfolding. Some stu dies [16, 17] have obtained impressive results thanks to the fact that the transf

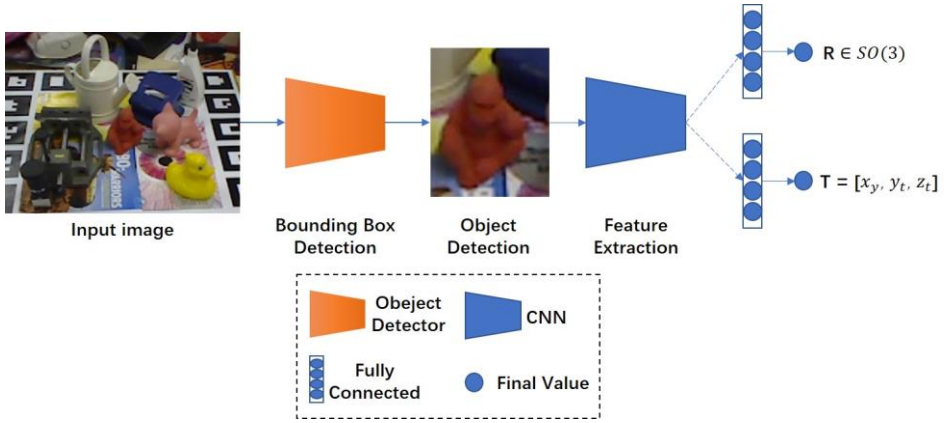


Figure 5. *Illustration of 6D pose estimation through regression.*

-ormer synchronizes the features and captures feature dependencies better.

Disadvantages:

- This method is not accurate for the detection of occluded objects because not all of the target is visible in a given image, so it is difficult to extrac t useful feature information when the occlusion area is too large.

II.3.4. Classification-based methods

This approach aims to solve the 6D pose of an object as a single-shot classification problem. They use CNNs to obtain the probability distribution of the 3D CAD model in the pose space to infer 3D translations and 3D rotations.

Tekin et al. [39] introduce a novel single-shot approach designed to simultaneously detect objects in RGB images and predict their 6D pose, streamlining the process by eliminating the need for multiple stages or examining multiple hypotheses. Central to their method is a new CNN architecture, drawing inspiration from the YOLO [13] network design, which directly forecasts the 2D image locations of the projected vertices belonging to the object's 3D bounding box. The 6D pose of the object is subsequently estimated through a PnP algorithm. Notably, when applied to single-object and multiple-object pose estimation tasks on datasets such as LINEMOD and LINEMOD-Occlusion, their approach consistently outperforms other recent CNN-based methods, even in scenarios without post-processing. Moreover, they suggest that incorporating a pose refinement step during post-processing can further elevate the

overall accuracy of the system. The authors introduce VI-Net [19], a novel rotation estimation network that simplifies the task by decomposing rotations into point-of-view and in-plane rotations. VI-Net, based on the sphere, estimates these rotations through separate V and I branch. The V-branch learns point-of-view rotation through binary classification of the spherical signal, while the I-branch estimates in-plane rotation by converting the signal to be viewed from the zenith direction. They address spherical signals using a spherical feature pyramid network with SPAtial Spherical Convolution (SPA-SConv) to handle boundary issues and achieve variable feature extraction. VI-Net is applied to category-level 6D object pose estimation for predicting the pose of an unknown object without an available CAD model. In the study [20], several prior works are referenced to enhance the algorithm's efficiency for complex scenarios. They adopt a two-stage pipeline for obtaining high-precision poses for multiple objects, with the first stage focusing on key point detection and the second stage employing PnP to determine the 6D pose. Introducing a simpler and more efficient classification-based key point detection algorithm for object surface key points is their proposed solution to address these challenges. Addressing the issue of substantial annotation requirements in existing category-level 6D pose estimation methods, Wanli et al. [21] proposes a self-supervised framework. Leveraging Deep SDF as the 3D object representation, novel loss functions are designed to enable the model to predict unseen object poses in real-world scenarios without the need for explicit labels or 3D models.

Disadvantages:

- Rotation and translation exhibit significantly different properties and are influenced by distinct factors. For instance, the size and position of an object in an

image have a minor impact on rotation but a substantial effect on translation. Conversely, the appearance of an object in an image strongly affects rotation and minimally affects translation. Therefore, attempting to predict the overall pose with a uniform binary classification approach would result in reduced accuracy in the predicted pose.

II.3.5. Keypoint-based methods

Keypoints, also referred to as points of interest, are specific points identified within textures. These points are typically characterized by abrupt changes in the direction of an object boundary or by being intersections between multiple edge segments. Keypoints have a well-defined and well-localized location within the image space. Importantly, even in the presence of local or global perturbations such as variations in illumination and brightness, keypoints remain stable. This stability allows keypoints to be computed repeatedly and reliably, making them valuable in various computer vision applications.

Methods in this category often follow a two-stage pipeline: first predicting the 2D keypoints of an object and then computing its pose. The authors [22] propose a Deep Fusion Transformer (DFTr) block for enhanced pose estimation by aggregating cross-modality features. DFTr leverages semantic similarity to model cross-modality correlation, enabling better integration of globally enhanced features. They introduce a weighted vector-wise voting algorithm with a non-iterative global optimization strategy for precise 3D keypoint localization, achieving near real-time inference with improved

robustness and efficiency.

In [23], REDE is proposed as an end-to-end object pose estimator using RGB-D data. It employs a network for keypoint regression and a differentiable geometric pose estimator for error back-propagation. To address outlier keypoint predictions, a differentiable outliers elimination method is introduced, simultaneously regressing candidate results and confidence. Confidence-weighted aggregation of multiple candidates reduces the impact of outliers in the final estimation. Finally, a learnable refinement process is applied for further improvement. Heng et al. [24] introduce conformal keypoint detection and geometric uncertainty propagation to the two-stage paradigm, creating the first pose estimator with provable and computable worst-case error bounds. Conformal keypoint detection converts heuristic keypoints into circular or elliptical prediction sets, covering ground truth keypoints with a user-specified probability. Geometric uncertainty propagation extends geometric constraints on keypoints to the 6D object pose, forming a Pose Uncertainty Set (PURSE) that guarantees coverage of the ground truth pose with the same probability. Hu [25] points out drawbacks in this approach, emphasizing its absence of an end-to-end system and the insufficiency of the neural network's loss function in accurately representing 6D position estimation, the authors propose a single-stage approach. This method directly regresses the 6D pose using the collection of 3D-to-2D correspondences associated with individual 3D object keypoints.

The keypoint-based pose estimation method is the most popular method today. Some recent studies [28,8] have predicted keypoints by means of indirect prediction representations, resulting in an impressive improvement in the model's prediction

performance for occluded objects. CDPN [29] analyses the different properties of translation and rotation and combines direct and indirect methods to estimate translation and rotation, respectively. GDRNet [30] and DPOD [31] extend the research of CDPN, and the performance of pose estimation is further improved. An experiment [32] showed that since the PnP algorithm is an offline algorithm and does not participate in the learning of the model, this is detrimental to the prediction of the model. Therefore, they proposed EPro-PnP algorithm for end- to-end training to learn 6D pose.

Disadvantages:

- The keypoint alone is the sparse representation of object pose, whose potential to improve estimation accuracy is limited.
- Keypoints need to be pre-labelled and most keypoint-based methods are two-stage pose estimation methods. Therefore, the process from processing the data to computing the final pose is more time consuming.

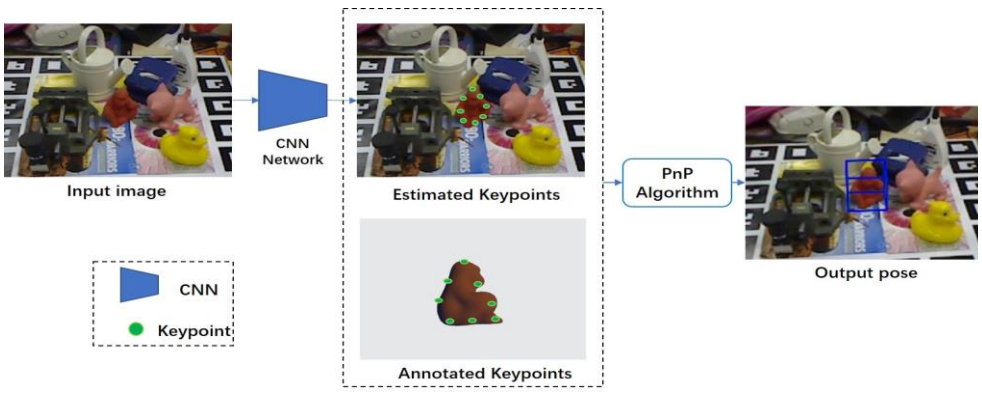


Figure 6. *Schematic of keypoint-based 6D pose estimation.*

II.4. PVNet 6D pose estimation approach

II.4.1. Overview

Addressing the challenge posed by objects in environments with substantial noise and potential occlusion causing unpredictable poses, Peng et al. [8] introduced a method grounded in keypoints. This approach, named PVNet, robustly calculates 2D keypoints through a pixel-wise voting mechanism. Rather than directly regressing 2D projections of 3D keypoints, they trained the network to regress direction vectors for each pixel pointing to the keypoints. Consequently, even when occluded, invisible keypoints could be accurately located by utilizing direction vectors derived from visible pixels of the obstructed object. In their strategy, the network was also trained to predict a semantic segmentation map to identify pixels belonging to the target object. Consequently, the success of the voting-based keypoint localization relies heavily on the precision of segmentation. However, challenges arise, especially in cases of extensive occlusion, where inaccurate segmentation can lead to failures. If pixels corresponding to the object are erroneously segmented, the associated direction vectors are compromised, resulting in the inaccuracy or unavailability of the voting-based keypoint localization. This issue becomes more pronounced in scenarios of severe occlusion where the count of visible object pixels is minimal.

In a more detailed breakdown, PVNet undertakes two distinct tasks: semantic segmentation and vector-field prediction. For each pixel, denoted as p, PVNet generates two outputs. First, it provides the semantic label that categorizes the pixel, associating it with a particular object. Additionally, it produces a unit vector, denoted as $v_k(p)$,

which signifies the direction from the pixel p to a 2D keypoint x_k of the object. This vector, $v_k(p)$, is precisely defined as

$$V_k(p) = \frac{X_k - P}{||X_k - P||_2} , \quad (2\text{-}1)$$

After PVNet predicts the predicted semantic labels and direction vectors, keypoints are generated by voting. Specifically, the semantic labels are first used to find the target pixel, and then the intersection of the predicted direction vectors from two random pixels of the target is used to locate the keypoint. Then from the generated keypoints the group with the highest accuracy is selected as the final prediction using a RANSAC-based [26] filtering method, and they refer to this process of filtering keypoints as voting.

The utilization of RANSAC-based voting [30] effectively deals with the discrete prediction of points by establishing a spatial probability distribution for key points. This voting methodology introduces a degree of uncertainty in the generation of key points, contributing to an enhanced capability of the PnP algorithm to accurately predict the final pose. By incorporating this approach, the system gains resilience in handling uncertainties and variations in key point predictions, ultimately refining the accuracy of the pose estimation process.

II.4.2. Weakness

In Figure 7, PVNet employs a method where it estimates the direction vectors for each pixel, indicating the direction toward the keypoints, instead of directly estimating 2D projections of 3D keypoints. This approach enables the identification of invisible keypoints by utilizing the direction vectors derived from visible pixels of occluded objects. The network is trained to predict a semantic segmentation map to identify pixel

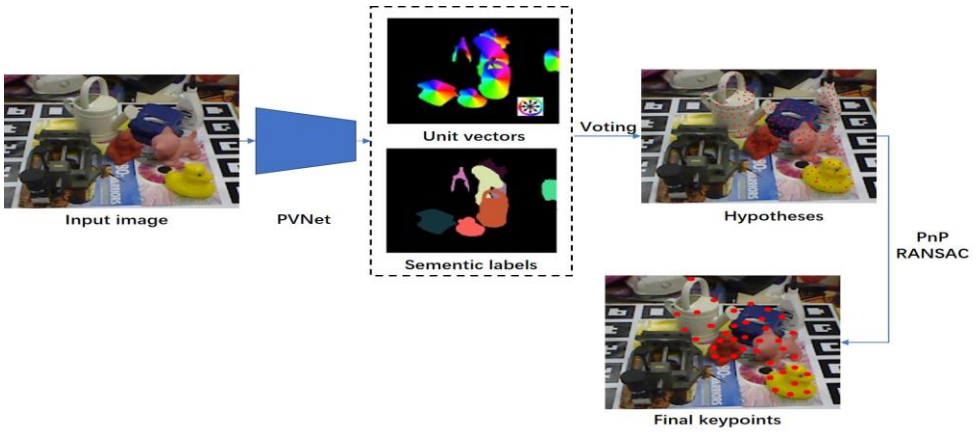


Figure 7. *PVNet pose estimation schematic.*

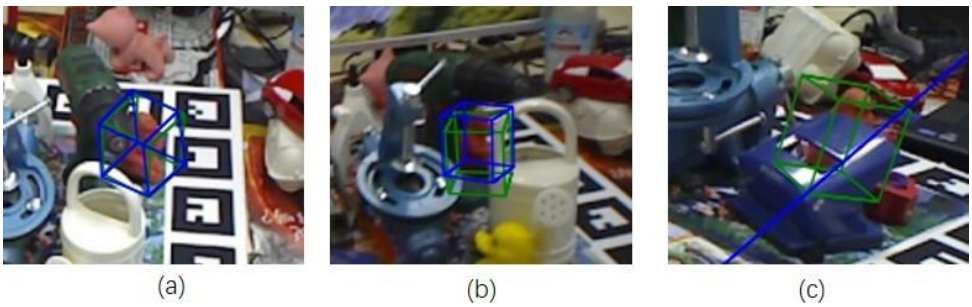


Figure 8. *Pose estimation results of PVNet: (a) slight occlusion, (b) moderate occlusion, and (c) severe occlusion. The green bounding boxes indicate the ground-truth poses, and the blue ones indicate the estimated poses. The two-colored boxes overlap in most cases, showing that the accuracy.*

-s belonging to the target object. Consequently, the accuracy of segmentation significantly influences the effectiveness of the voting-based keypoint localization. This method tends to fail when dealing with large occluded areas, particularly when incorrectly segmented pixels result in the loss of direction vectors, leading to inaccurate or unavailable voting-based keypoint localization. The challenge intensifies in cases of severe occlusion, where the number of visible object pixels is limited. Figure 8 illustrates the outcomes of our experiments with PVNet. It is evident that the detection performance of PVNet remains consistent under conditions of light occlusion; however, its efficacy diminishes as the level of occlusion deepens, eventually resulting in failure under severe occlusion. Moreover, PVNet is susceptible to misdirection by objects sharing similar colors or shapes, resulting in detection failures, as exemplified in Figure 9. Our experimental findings indicate that this susceptibility is attributed to the visual components being excessively small.

Figure 9. *Failure cases1. The network is easily misled by other objects with similar shapes or colors, leading to detection failures.*

II.5. Perspective-n-point (PnP) algorithm

The PnP algorithm serves as a solution for determining the camera position based on the coordinates of a 3D point, the corresponding 2D point coordinates, and the internal reference matrix. Widely applied in Computer Vision, Robotics, and Augmented Reality, it has garnered significant interest in both the Photogrammetry and Computer Vision communities. Notably, it finds practical use in applications like feature point-based camera tracking, where real-time processing involves handling numerous noisy feature points. Consequently, there is a need for computationally efficient methods to address the challenges posed by these applications.

The camera pose involves 6 degrees of freedom, encompassing 3D rotation (roll, pitch, and yaw) and 3D translation relative to the world. As a result, obtaining information from at least three pairs of corresponding points becomes essential to solve a PnP problem. While many existing solutions are designed for the general case where the number of points (n) is greater than 3, there are also solutions specifically tailored for scenarios when n equals 3. This versatility allows for the effective application of PnP algorithms across a range of cases, accommodating both common and specific situations.

In most solutions, a prevalent assumption is that the camera is pre-calibrated, with intrinsic properties such as focal length, principal image point, skew parameter, and other relevant parameters already known. The PnP problem can yield multiple solutions, and selecting a specific solution from the set often involves post-processing. To mitigate the impact of noisy data, employing a greater number of point correspondences

in solving PnP is recommended. Commonly, Random Sample Consensus (RANSAC) is used for selecting point correspondences to enhance the robustness of the solution by effectively dealing with outliers [26]. Two widely used methods for solving the PnP problem are P3P and EPNP, with Figure 10 providing a schematic representation of P3P. P3P employs the geometric relationship of similar triangles and the cosine theorem to determine the camera pose. The input data for P3P consists of three pairs of matching 3D-2D points.

In the context of P3P, let's denote the 3D points as A, B, and C, and the corresponding 2D points as a, b, and c. Here, the lowercase letters represent the projections of the corresponding uppercase letters on the camera imaging plane. Additionally, P3P necessitates the inclusion of a pair of validation points to discern the correct solution from the potential set of solutions. The optical center of the camera is denoted as O in this process.

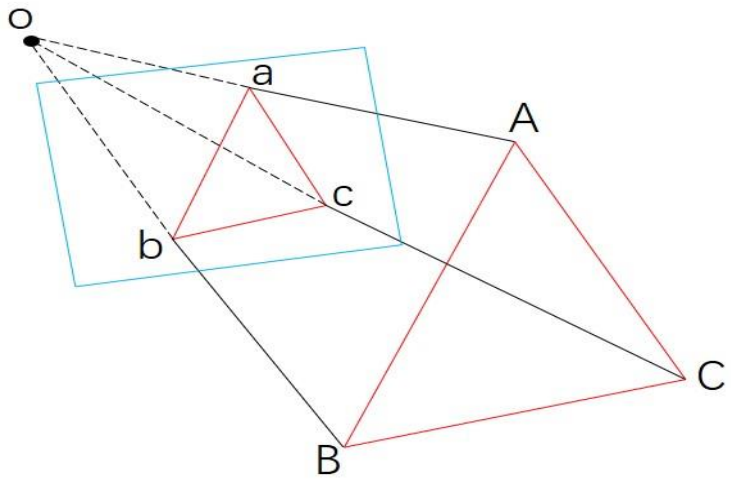


Figure 10. *Schematic diagram of the P3P problem.*

II.6. Focal loss

In object detection, detectors are commonly classified into two categories: two-stage detectors and one-stage detectors. The former includes detection algorithms such as Faster R-CNN [27], which involve region proposal and can achieve high accuracy but at a slower speed. While speed can be improved by reducing the number of proposals or decreasing the resolution of the input image, there is no significant qualitative enhancement in speed. On the other hand, the latter category comprises detection algorithms like YOLO [13], which do not require region proposals and instead regress directly. These methods are faster but generally exhibit lower accuracy compared to two-stage detectors.

Lin et al. [33] propose that the challenge in object detection is attributed to the category imbalance within the samples. In many cases, an image may generate thousands of candidate locations, but only a small percentage of them actually contain objects, leading to a significant category imbalance. This imbalance hinders efficient training, as a substantial portion of locations are easy negatives that contribute minimal useful learning signals.

To address the issue of category imbalance, the authors introduce focal loss, a modified version of the standard cross-entropy loss. Focal loss aims to make the model focus more on challenging-to-categorize samples during training by reducing the weight assigned to easy-to-categorize samples. The formula for focal loss is expressed as follows:

$$\text{Focal loss} = -a_t(1-p_t)^y \log(p_t) \ , \tag{2-2}$$

In the given formula, y represents the focusing parameter, and its role is to reduce the weight assigned to easy-to-categorize samples. This adjustment aims to make the model more focused on hard-to-categorize samples during the training process. Additionally, a serve as a hyperparameter that allows for the control of the shared weight of positive and negative samples in the total loss. The value of a determines the balance between the contributions of positive and negative samples to the overall loss. Meanwhile, p_t denotes the probability of the current category.

CHAPTER III

THE PROPOSED 6D POSE ESTIMATION APPROACH OF OCCLUDED OBJECTS USING KEYPOINT FROM RGB IMAGE

III.1. Introduction

As mentioned before, pose estimation can easily be affected by occlusion leading to inaccuracy or even failure. Given the swift advancements in deep learning neural network research, the research on 6D pose estimation based on deep learning has also been very much advanced, but its performance for complex detection scenarios still needs to be improved. To achieve robust pose estimation, depth information has been utilized in many researches. However, depth cameras are very sensitive to noise such as light in the field, and the problem of high-power consumption is difficult to solve if used as sensing on mobile devices. Therefore, most of the researches have started to move towards estimating 6D pose directly from RGB images only. Keypoint detection as a popular method for estimating pose using RGB images only has two general steps. (1) predict the keypoints of the surface from the RGB image; (2) compute the 6D pose using the perspective-n-point (PnP) algorithm by exploiting the correspondence of 2D-3D point pairs. These methods have good performance for objects without any occlusion, but the performance for objects with occlusion leaves much to be desired, especially when the occlusion is severe.

To address the challenge of detecting occluded objects, Peng et al. [8] do not use the method of neural network to predict keypoints directly, but generate keypoints by an intermediate representation for voting. Briefly, they created a neural network structure called PVNet to predict the direction vectors from each pixel to the keypoints, and then used the direction vectors generated from these visible pixels to locate the invisible keypoints. So, they first had to obtain the target pixels needed to generate the direction vectors through semantic segmentation, and the localization of the keypoints depended heavily on the segmented target pixels. For slightly occluded objects PVNet can estimate the 6D pose well, when the object is in severe occlusion the performance of this method still needs to be improved. If the pixels of an object are incorrectly segmented or the segmented pixels are too small, the direction vectors will be lost and the keypoint localization will become inaccurate or impossible.

III.2. Overview of proposed approach

Our goal is to improve the performance of PVNet [8] for detecting the pose of depth-obscured objects. Having analyzed the pose estimation process of PVNet, we find that the performance of pose estimation depends largely on the performance of semantic segmentation, since the direction vectors used to locate the keypoints are generated from the target pixels of semantic segmentation.

In the field of object detection, Lin et al. proposed focus loss [33] to address the imbalance in the proportion of positive and negative samples by reducing the weight of

negative samples in training. Similarly, in semantic segmentation we consider the image as a sample space, while the target pixels are positive samples and the rest of the pixels are negative samples. In case of masking, the pixels of positive samples are obviously much smaller than negative samples and this problem of imbalance of samples causes the network to have difficulty in extracting the target pixels. To solve this problem, we introduce focus loss into PVNet to equip the network with an attentional mechanism to extract as many useful targets features as possible, even in the case of severe occlusion.

In our experiments, we kept the original network of PVNet and only modified the loss function. The process of pose estimation is shown in Figure 11. To estimate the 6D pose of the target object P = [R, t] (where R is the rotation matrix and t is the translation vector) from the input RGB image, our network maintains the same pipeline as PVNet and outputs the segmentation map and pixel direction vector. But our segmentation map is a more accurate obtained through the attention mechanism, which we call focus segmentation map. Then keypoints are generated from the predicted pixel and direction vectors and finally the PnP [20] algorithm is used to compute the 6D pose.

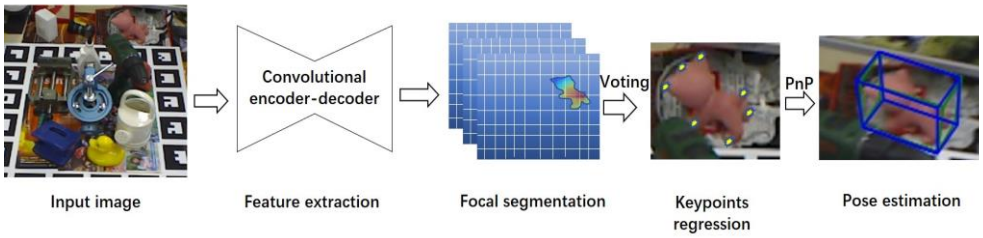


Figure 11. *Process of our method.*

III.3. Approach

In PVNet [8], the loss function consists of a pixel-wise direction vector loss and a segmentation loss as follows:

$$L_{total} = L_{vec} + kL_{seg} \ , \tag{3-1}$$

where κ represents a segmentation loss weight and is set to 1. The segmentation loss is computed by the cross-entropy function and the direction vector loss is computed as follows:

$$L_{vec} = \sum_{k=1}^{K} \sum_{p \in o} l_1(\triangle v_k(p)|_x) + l_1(\triangle v_k(p)|_y) \ , \tag{3-2}$$

where K is the number of keypoints and l_1 represents the smooth L1 loss [34]. $\Delta v_k(p)$ is the direction error in pixel p and defined as follows:

$$\triangle v_k(P) = \tilde{v}_k(p) - \frac{X_k - P}{||X_k - P||_2} , \tag{3-3}$$

Where the $\tilde{v}_k$ is the predicted unit vector and x_k is the ground-truth coordinates of the k-th keypoint.

In our method, we modify the cross-entropy loss for semantic segmentation based on an attention mechanism as follows:

$$L_{seg} = \sum_{i \in M} h * f_i \ , \tag{3-4}$$

where · represents the inner product, h is the one-hot encoded ground-truth, and f_i is defined as

$$f_i = -aW_i * log(S_i), \tag{3-5}$$

where α is a hyperparameter and set to 0.25 in our experiments, and S_i is the soft-max probability of pixel i. W_i is a dynamic weight and computed as

$$W_i = (1 - S_i)^{\beta}, \tag{3-6}$$

where β is a hyperparameter and set between 1.5 and 2 in our experiments.

In the training stage, employment of W_i causes the network to add more weight to the segmentation error of the target object pixels that fall within the segmentation mask M, that is, the image regions that truly belong to the target objects, thereby alleviating the imbalance between the number of target pixels and the number of the others.

The attention mechanism illustrated in Eq. 3-5 enables the network to concentrate more on segmenting the target pixels than the other pixels. This mechanism is inspired by focal loss and improves semantic segmentation; thus, it is named focal segmentation. Specifically, the focal segmentation mechanism enables the target object pixels to be fully and more accurately segmented. Even for severely occluded objects, sufficient number of target object pixels can be obtained, resulting in successful keypoint localization.

III.4. Result and discussion

III.4.1. Setup

For a fair comparison of experiments and results, we used the source code of PVNet [8] and the experimental setup was kept the same as PVNet. Unfortunately, due to the limitations of the experimental equipment, we were only able to reduce the mini-batches size to 6, which resulted in a slight decrease in accuracy for both PVNet and our method.

III.4.2. Datasets

The LINEMOD dataset, which is the standard dataset used for evaluating 6D pose estimation (Figure 12), contains a total of 18,273 RGB-D images of 15 cluttered and slightly masked home objects. Given the bounding box, rotation and translation matrices the target object is located at the center of the image, and the provided masks represent the effective region of the target pixels. Additionally, the given 3D CAD model can be used to generate a composite image.

LINEMOD-Occlusion [36] consists of RGB-D images of 20 home objects, real poses, masks and 3D CAD models. This dataset is an extended version of the LINEMOD dataset, and most of the objects in the dataset are in the occlusion state, so detecting the poses of these objects can be more challenging. Additional bounding boxes and poses are included for all seven other objects that appear in the Benchvise sequence.

For our experiments, we used the “ape”, “can”, “cat”, “duck”, “driller”, “eggbox”,

Figure 12. *Part of images from the LINEMOD dataset used in our experiments.*

"glue," and "holepunches" models commonly included in both datasets.

III.4.3. Evaluation metrics

For evaluation, we measure the percentage of images where the object pose was estimated correctly. The pose correctness was determined using the following two different metrics.

2D reprojection error [37] is the mean distance between the 2D projection of the object's 3D mesh vertices obtained by applying the predicted and the ground-truth pose, and the predicted pose is correct if the error is less than 5 pixels.

Average distance (ADD) metric [38] is the mean 3D distance between model vertices transformed by the ground-truth pose and the predicted pose. The estimated pose is correct when the distance is less than 10% of the model's diameter.

III.4.4. Results and discussion

Results of the pose estimation experiments on the LINEMOD dataset are shown in Tables 1 and 2. For each object, the number of images used in the evaluation was different, ranging from 1,002 to 1,050, and the metric results were averaged. The comparison with other state-of-the-art pose estimation methods is provided as well. The best results are marked in bold. Although a slight difference between both evaluation metrics was noted, our method outperformed the others, except the RGB-D-based method. Without depth information, our method showed similar performance to the RGB-D-based methods.

In the occlusion-free cases, although the performance difference was not significant, our method exhibited better performance than PVNet (consistently better in terms of the 2D projection error), indicating that it is crucial to fully segment object pixels regardless of the existence of occlusion for accurate pose estimation. In other words, PVNet fails to segment certain object pixels (even if the whole object is visible), and the pose accuracy is lower than our method due to the difficulty of obtaining accurate direction vectors from the missing object pixels.

Results of the pose estimation experiments on the LINEMOD-Occlusion dataset are shown in Tables 3 and 4, where 1,170 - 1,214 images have been used for each object. Since PoseCNN proposed by Xiang et al. [43] is a direct method and extremely sensitive to occlusion, its accuracy is low. YOLO6D proposed by Tekin et al. [39] is an accuracy is low in the presence of occlusion. GDRNet is a direct method proposed by Wang et al. [44], nonetheless, it extracts intermediate representations, including 2D-3D

Table 1. *ADD metric of different pose estimation methods on the LINEMOD dataset.*

Method	Image	Ape	Can	Cat	Driller	Duck	Eggbox	Glue	Puncher	Avg
[10]	RGB	27.9	48.1	45.2	58.6	32.8	40	27	42.40	40.29
[31]	RGB	53.28	94.1	60.38	97.72	66	99.72	93.83	65.83	78.86
[39]	RGB	21.62	68.8	41.82	63.51	27.23	69.58	80.02	42.63	49.38
[40]	RGB	0.00	1.35	0.51	2.58	0	8.9	0	0.3	0.68
[8]	RGB	62.28	97.53	77.45	95.04	57.93	99.81	77.89	85.06	81.62
Ours	RGB	46.95	97.93	82.44	95.34	48.23	100	90.44	83.06	80.42
[41]	RGBD	80.00	87	89	78	76	100	99	79	86
[42]	RGBD	58.10	84.4	65	76.3	43.8	96.8	79.4	74.8	72.33

Table 2. *2*D reprojection error of different pose estimation methods on the LINEMOD dataset.

Method	Image	Ape	Can	Cat	Driller	Duck	Eggbox	Glue	Puncher	Avg
[10]	RGB	95.3	84.1	97.0	74.1	81.2	87.9	89	90.5	87.31
[39]	RGB	92.1	97.44	97.41	79.41	94.65	90.33	96.53	92.86	92.91
[8]	RGB	98.95	99.7	99.89	96.44	98.77	99.24	97	99.9	98.74
Ours	RGB	99.05	99.8	99.91	97.02	98.96	99.43	98.74	100	99.11

dense correspondences, which render it robust to occlusion. PVNet exhibits suitable performance in the presence of occlusion owing to the voting-based keypoint localization scheme. However, our method outperformed PVNet an d the others. In terms of ADD metric, only GDR-Net was comparable to ours. The difference between PVNet and our method shows that improvement in the segmentation process has a positive effect on pose estimation, particularly in the presence of occlusion. From the results of 2D reprojection error evaluation, it is clear that our method has improved the performance of pose estimation for models such as Ape, Duck, and Glue. The average

Table 3. *2D reprojection error of different pose estimation methods on the LINEMOD-Occlusion dataset*

Method	Image	Ape	Can	Cat	Driller	Duck	Eggbox	Glue	Puncher	Avg
[10]	RGB	34.6	15.1	10.4	31.8	7.4	1.9	13.8	23.1	17.26
[39]	RGB	7.01	11.2	3.62	5.07	1.4	-	4.7	8.26	5.89
[8]	RGB	55.21	83.51	59.39	64.51	41.8	1.36	51.27	60.92	52.25
Ours	RGB	59.4	85.83	58.21	68.12	46.88	2.13	54.7	57.07	54.04

Table 4. *ADD metric of different pose estimation methods on the LINEMOD-Occlusion dataset*

Method	Image	Ape	Can	Cat	Driller	Duck	Eggbox	Glue	Puncher	Avg
[43]	RGB	9.6	45.2	0.93	19.6	41.4	22	38.5	22.1	25.33
[39]	RGB	2.48	17.48	0.67	1.14	7.66	-	10.08	5.45	6.42
[44]	RGB	41.3	71.1	18.2	54.6	41.7	40.2	59.5	52.6	48.43
[8]	RGB	18.29	63.21	17.44	60.37	11.74	26.8	36.76	36.15	33.85
Ours	RGB	77.78	69.34	20.31	66.06	26.3	51.09	51.09	45.31	49.08

accuracy improvement for all models is about 2%. Actually, there are not many images in the datasets that contain severe occlusion on which our method has worked correctly but PVNet has not. This may be the reason why the evaluation metrics of our method are not significantly higher than those of PVNet in Tables 1, 2, 3, and 4. Therefore, we will visually show the accuracy of the pose estimated from images with different degrees of occlusion later. Before that, to show the performance difference between PVNet and our method more clearly, the mean of 2D reprojection pixel errors was

compared on three objects, as shown in Table 5. Here, since an error value greater than 5 implies pose estimation failure, the error values greater than 5 were adjusted to 5. Our method has 0.052 and 0.125 lower pixel errors than PVNet in the LINEMOD and LINEMOD-Occlusion datasets, respectively. Furthermore, our method has lower standard deviations of pixel errors, indicating that our method is more stable.

Table 5. *ADD metric of different pose estimation methods on the LINEMOD-Occlusion dataset.*

Method	LINEMOD			LINEMOD-Occlusion		
	Ape	Cat	Driller	Ape	Cat	Driller
[8]	1.31±0.33	1.33±0.37	2.07±0.5	3.88±1.15	3.63±1.14	3.6±1.14
Ours	1.23±0.31	1.32±0.34	2.02±0.48	3.75±1.15	3.64±1.14	3.34±1.11

III.4.5. Comparison of segmentation performance with PVNet

We believe that our method can acquire keypoints more accurately by improving the semantic segmentation process of PVNet [8]. To ensure that our method actually produces improved segmentation results, we compared semantic segmentation results of our method and PVNet on five objects. The accuracy of segmentation results was measured by two metrics: mean intersection of union (mIOU) and pixel accuracy (PA), as shown in Tables 6 and 7.

Contrary to our expectation, PVNet's segmentation results are slightly more accurate than our method. This is because PVNet emphasizes more on the accuracy of segmentation so some object pixels are lost, whereas our method is more likely to over-

Table 6. *Accuracy comparison of semantic segmentation results of our method and PVNet on the LINEMOD dataset.*

Method	mIOU					PA				
	Ape	Cat	Driller	Eggbox	Puncher	Ape	Cat	Driller	Eggbox	Puncher
[8]	96.4	99.89	96.9	97.53	97.21	99.95	99.91	99.82	99.91	99.9
Ours	95.46	99.9	96.5	96.79	99.88	99.94	99.91	99.76	99.88	96.46

extract object pixels and the over-extracted pixels are larger than the lost pixels. Therefore, focal segmentation may not result in segmentation maps that are more similar to the ground truth. However, by over-extracting with focus segmentation, the object pixels can be completely segmented out and enough object pixels can be obtained to robustly localize the keypoints. It simply means that focal segmentation acquires more pixels while maintaining a certain level of accuracy. This is crucial for the localization of occluded objects, especially for objects where only a small portion is visible.

Table 7. *Accuracy comparison of semantic segmentation results of our method and PVNet on the LINEMOD-Occlusion dataset.*

Method	mIOU					PA				
	Ape	Cat	Driller	Eggbox	Puncher	Ape	Cat	Driller	Eggbox	Puncher
[8]	84.81	64.43	91.91	73.37	85.16	99.81	99.86	99.68	99.49	99.46
Ours	83.48	62.13	91.52	71.84	83.27	99.98	99.85	99.66	99.46	96.39

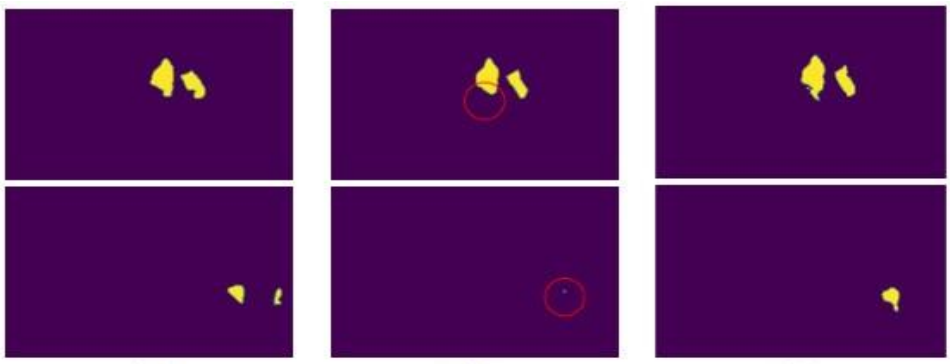


Figure 13. *Semantic segmentation results for the same target object with different methods ((a) ground-truth, (b) PVNet, (c) our method) in the LINEMOD-Occlusion dataset. PVNet lost certain parts of the target object. Under severe occlusion (the images below), PVNet could not find any target pixels.*

In the presence of occlusion, we need to analyze the segmentation process of both methods in more detail. In Figure 13 that shows semantic segmentation results for occluded objects, we observe that PVNet provides rather accurate segmentation results (close to the ground-truth). Nonetheless, as marked by the red circle, certain useful target pixels were lost, causing sufficient number of keypoints not to be correctly obtained, and the resulting pose was less accurate. On the contrary, our method shows the obtained segmentation results that have most target pixels but contain incorrect peripheral pixels. How- ever, the incorrect pixels can be excluded at the vector voting stage; this does not affect the pose estimation. This strategy is more effective for severely occluded objects with only few target pixels. As shown in the below images of Figure 13, under severe occlusion, PVNet does not find any target pixels and fails to generate keypoints, whereas our method can find a part of relevant pixels, enabling the generation of keypoints. This explains why our method had slightly lower segmentation accuracy than PVNet in Tables 6 and 7. However, pose estimation results shown in

Tables 1, 2, 3, and 4 are more accurate.

III.4.6. Visual comparison of pose estimation accuracy with PVNet

We first compared our findings with the PVNet pose results on the LINEMOD dataset as shown in Figure 14. Our pose results are more accurate, as per observation of the overlap between the predicted bounding box and the ground-truth.

Subsequently, we compared our results with the pose results of PVNet on the LINEMOD-Occlusion dataset, as shown in Figure 15. Although the targets in the images are occluded to different degrees, our method can estimate accurate poses with a 2D projection error below 5 pixels even under severe occlusion. However, as the occluded area increases, PVNet can detect and locate objects less accurately. When only a tiny area of objects is visible, PVNet fails to detect and locate the target object because it cannot segment any target pixel. Therefore, it cannot generate direction vectors for localization. On the contrary, under these challenging conditions, our method was able to locate objects robustly by successfully segmenting tiny target objects, although the estimated pose was not accurate owing to the limited number of segmented pixels. In Tables 1, 2, 3, and 4, the pose estimation accuracies are slightly different, depending on the target object. “Ape” was more challenging; thus, the pose estimation results were less accurate, as shown in Figure 15. However, our method is still more accurate than PVNet.

To clearly demonstrate the effectiveness of our method under severe occlusion, the diffe

Figure 14. *Visualization of the pose estimation results on the LINEMOD dataset. Top: PVNet results and bottom: our results. The images were cropped and enlarged to improve visibility. The green and blue rendered bounding boxes denote the ground-truth and predicted pose, respectively.*

Figure 15. *Visualization of the pose estimation results on the LINEMOD-Occlusion dataset. First and third rows: PVNet results, second and fourth rows: The images were cropped and enlarged to improve visibility. The green and blue rendered bounding boxes denote the ground-truth and predicted pose, respectively.*

-rence in segmentation results and pose estimation results between our method and PVNet was further analyzed in Figure 16. Under severe occlusion, PVNet often segmented no or few object pixels. thus, its pose results were not accurate. Even when a few object pixels were successfully segmented (third and fourth rows in Figure 17), keypoint localization using the direction vector of the segmented object pixels did not yield satisfactory pose results. Even with voting-based keypoint localization, more object pixels need to be segmented to produce accurate pose results. This indicates a close dependence between the segmentation results and the pose estimation results.

In the LINEMOD-Occlusion dataset, truncated objects are also present (an example is shown in the first row of Figure 16). To further evaluate our method on truncated objects, pose estimation results were visualized in Figure 16 and compared with those of PVNet. PVNet failed pose estimation under severe truncation, and its pose results were consistently less accurate than our method, indicating that our method has higher robustness to truncation than PVNet as well.

Finally, as shown in Table 8, our method took 34.2 ms to complete a pose estimation from an image of size 480 × 640 pixels on a desktop with an Intel i7-11700 2.5GHz CPU and a Nvidia RTX 2060 GPU. The computation time is not different from that

Table 8. *Computation time (ms) of our method and PVNet for 480 × 640 input images.*

Method	LINEMOD			LINEMOD-Occlusion			Average
	Ape	Cat	Driller	Ape	Cat	Driller	
PVNet	34.0	33.9	34.1	34.0	33.9	34.3	34.0
Ours	34.2	34.8	34.0	34.0	34.2	34.3	34.2

(34.0 ms) of PVNet and is sufficiently short, enabling real-time pose estimation.

III.4.7. Limitations

After analyzing our method from many aspects, we found certain failure cases, as shown in Figure 18. In the cases where the target object is barely visible, extracting sufficient pixels to generate keypoints is difficult; however, our network attempts to obtain the target pixels to the extent possible. To obtain more pixels, our network comes to regard other objects with similar colors or shapes as the target, resulting in the generation of wrong keypoints. However, in these cases, previous methods, including PVNet, have failed even to detect target objects.

Figure 17. *Robustness to truncation. First and third rows: PVNet results, second and fourth rows: our results. The images were cropped and enlarged to improve visibility. The green and blue rendered bounding boxes denote the ground-truth and predicted pose, respectively.*

Figure 16. *Close dependence of segmentation results and pose estimation results. (a) Input image and ground-truth mask image, (b) PVNet results, and (c) results obtained by applying our method. The more the object pixels are successfully segmented, the more accurate the pose can be estimated. The images were cropped and enlarged to improve visibility. The green and blue bounding boxes denote the ground-truth and predicted pose, respectively.*

***Figure 18**. **Failure cases2.** Under too severe occlusion, objects with very similar colors and shapes can mislead the box regression (left and middle images) and the box regression may fail (right image).*

III.5. Summary

In this study, we aim to develop a 6D object pose estimation method that is robust to occlusion and propose a novel method that is based on PVNet with improved semantic segmentation process using the strategy of focal loss. Our method did not produce a segmentation map more similar to the ground-truth but could fully segment object pixels, which enabled more accurate pose estimation than PVNet, both in occlusion-free cases and in the presence of occlusion. In the experiments on LINEMOD and LINEMOD-Occlusion datasets, our method outperformed other 6D object pose estimation methods, including RGB-D-based methods. Under severe occlusion, wherein no other method can even detect a target object, our method is able to reliably detect the target object and estimate its pose. Furthermore, our method demonstrates enhanced performance in the presence of truncation. The computation time of our method is similar to that of PVNet, which is sufficiently short, allowing real-time pose estimation. However, as mentioned before, our method is still unable to completely segment object pixels under severe occlusion. To improve the capability of segmenting and detecting severely occluded objects, we are considering modifying or replace the

PVNet's backbone network that is based on ResNet [45]. Unfavorably, the performance degraded when we replaced it with DetNet, a backbone network particularly designed for small object detection [46]. However, we will continue to seek and utilize alternate suitable candidates. In addition, the hyperparameters in Eqs. 3-5 and 3-6 were heuristically set in our experiments. Therefore, analyzing their influence on the accuracy of our method in detail is necessary, which remains to be explored in a future study.

CHAPTER IV

CONCLUSION

IV.1. Conclusion

Predicting the 6D rotation and translation of a partially hidden object using only a single RGB image presents a significant challenge in computer vision, particularly with the proliferation of deep learning solutions leading to the development of novel applications. This thesis initially addresses the complexities associated with estimating the 6D pose of occluded objects, shedding light on the underlying reasons for these challenges. Subsequently, it introduces diverse estimation methods, delving into their limitations within the scope of pose estimation.

Template-based methods, the initial approaches employed for pose estimation, exhibit sensitivity to occluding objects, and the template generation process is time-consuming. Regression or classification-based methods face challenges in accommodating the distinct properties of rotation and translation, leading to limited performance in occluded scenarios. In contrast, the keypoint-based approach proves more adept at handling pose estimation issues for occluded objects. Leveraging the intermediate representation of pose (e.g., symmetry and directionality of keypoints), this indirect pose estimation approach better utilizes information. In conclusion, we propose the focal segmentation method to enhance the stability of estimating the 6D pose for occluded objects. The suggested method outperforms PVNet in heavily occluded situations, providing a more reliable 6D pose estimation.

IV.2. Challenges and limitations

Object estimation and the determination of their 6D poses play a crucial role in spatial 3D perception, in various scenarios like semantic simultaneous localization and mapping, target-oriented navigation, autonomous driving, object manipulation, and augmented reality, recent advancements in deep learning techniques have shown notable progress in training models for estimating object poses. However, these models frequently face challenges in achieving effective generalization, especially when dealing with the same object instance in diverse environments, particularly in real-world data settings.

Estimating the 6D pose of objects in situations where they are partially obscured by occlusions presents a multifaceted challenge. One of the primary hurdles stems from the inherent incompleteness of object visibility due to obstructing elements, making it arduous for the model to encapsulate all the requisite visual cues essential for accurate pose estimation. This difficulty is exacerbated by the fact that occluded regions often result in a dearth of discriminative features, impeding the model's capacity to precisely ascertain the object's pose. Furthermore, the introduction of occlusion introduces an additional layer of complexity by instigating ambiguity. The presence of occlusions means that multiple plausible pose configurations can arise based on the observable parts of the object. This inherent ambiguity amplifies the intricacy of the prediction task, rendering it more challenging to discern the correct pose accurately. The variability in occlusion patterns across diverse scenes and environments further complicates matters, posing a significant hurdle to the generalization capabilities of pose estimation models. In real-world scenarios, the dynamics and unpredictability of

occlusions add yet another dimension of intricacy. The model must dynamically adapt to varying degrees and types of occlusions, necessitating a high degree of robustness and flexibility in its predictive capabilities. Summarily, the innate complexity of occlusion, encompassing factors such as incomplete information, ambiguity stemming from occluded regions, and dynamic variations in real-world settings, collectively contribute to the formidable challenge of achieving precise 6D pose prediction for partially obscured objects.

In addition to occlusion impacting 6D pose detection, various factors contribute to the complexity of the task. Changes in illumination conditions, the presence of noise and distortion in images, the diverse shapes and textures of objects, accuracy of camera parameters, and the challenges posed by both rigid and non-rigid object motions all play significant roles. Furthermore, the reliability of data annotation adds another layer of influence, as inaccuracies in labeled training data can affect the model's learning patterns. Navigating these multifaceted factors is essential when developing pose estimation models, ensuring robust performance and effective generalization in real-world scenarios.

IV.3. Future works

We provide a concise overview of methods for object pose estimation, categorized based on fundamental properties impacting performance and target objects. Nevertheless, these methods may encounter difficulties when faced with new objects or challenging scenarios. One approach is to develop models with a more profound

comprehension of object structure, while another option is to leverage intermediate representations of features.

The proposed method in this paper aims to refine future work by identifying a technique that can effectively extract features from the target region without being unduly affected by the background, thereby minimizing false predictions. One potential avenue involves developing a mechanism capable of filtering out extraneous parts of the features while concentrating on the extraction of the desired target features.

REFERENCES

1. Eppner, C., H¨ofer, S., Jonschkowski, R., Mart´ın-Mart´ın, R., Sieverling, A., Wall, V and Brock, O.: Lessons from the amazon picking challenge: Four aspects of b-uilding robotic systems. In IJCAI, 2017.

2. Arash, A., Arul, Selvam, P., Sven, B.: YOLOPose: Transformer-based multi-object 6D pose estimation using keypoint regression. In CVPR, 2022.

3. Yinlin, H., Pascal, F., Wei, W., Mathieu S.: Single-stage 6D object pose estimatio-n. In CVPR, 2020.

4. Park, K., Patten, T., Vincze, M.: “Pix2pose: Pixel-wise coordinate regression of ob-jects for 6d pose estimation”, IEEE/CVF International Conference on Computer V-ision, pp. 7668-7677, 2019.

5. Schaub, H., Schöttl, A.: “6-DOF Grasp detection for unknown objects”, Internatio-nal Conference on Advanced Computer Information Technologies, pp. 400-403, 20 20.

6. Paul, W and Vincent, L.: Learning descriptors for object recognition and 3D pose Estimation. In Conference on Computer Vision and Pattern Recognition (CVPR), 2 015.

7. Pavlakos, G., Zhou, X., Chan, A., Derpanis KG.: Daniilidis, K. 6-DoF object pos-e from semantic keypoints. (2017) Multimedia Tools and Applications.

8. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: PVNet: pixel-wise voting networ-k for 6-DoF pose estimation. 2019 IEEECVF Conf. Comput. Vis. Pattern Recogn-it. CVPR, pp 4-556–4565.

9. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A Convolutional neural network for 6D object pose estimation in cluttered scenes. arXiv 2017, arXiv:1711.00199.

10. Rad, M., Lepetit, V.: BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth. In Proceedings of t-he 2017 IEEE International Conference on Computer Vision (ICC-V), Venice, Italy, 22–29October 2017; pp. 3848–3856.

11. Lepetit, V., Moreno-Noguer, F and Fua, P.: Epnp: An accurate o(n) solution to the pnp problem. International Journal of Computer Vision, 81:155–166, 2008.

12. Nguyen, Nguyen, V., Groueix, T., Salzmann, M., Lepetit, V.: GigaPose: fast and r-obust novel object pose estimation via one correspondence. In CVPR, 2023.

13. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, r-eal-time object detection. In CVPR, 2015.

14. Dewei, Z., Qi, C., Zilong, Z., Haozhe, H., Ruize, G., Wei, Q.: An Improved method for model-based training, detection and pose estimation of texture-less 3D obj-ects in Occlusion scenes. In Procedia CIRP, Volume 83, 2019.

15. Yann, L., Lucas, M., Arsalan, M., Stephen, T., Stan, B., Jonathan, T., Justin, C., Mathieu, A., Dieter, F., Josef, S.: MegaPose: 6D pose estimation of novel objects via render & compare. In CVPR, 2022.

16. Arash, A., Arul, Selvam P., Sven B.: T6D-Direct: Transformers for multi-Object 6D pose direct regression. In CVPR, 2021.

17. Lin, S., Wang, Z., Ling, Y., Tao, Y and Yang, C.: E2EK: End-to-end regression network

based on keypoint for 6D pose estimation, in IEEE Robotics and Automation Letters, 2022.

18. Periyasamy, A.S., Schwarz, M., Behnke, S.: Robust 6D object pose estimation in cluttered scenes using semantic segmentation and pose regression networks. In: IR OS (2018).

19. Jiehong, L., Zewei, W., Yabin, Z., Kui, J.; VI-Net: Boosting category-level 6D ob -ject pose estimation via learning decoupled rotations on the spherical representatio -ns. Proceedings of the IEEE/CVF International Conference on Computer Vision (I -CCV), 2023, pp. 14001-14011.

20. Wanhu, S and Maolin, Z and Sheng, Zhang.: EP-Net: More efficient pose estimat- ion network with the classification-based keypoints detection. In Association for C -omputing Machinery, 2021.

21. Wanli, P., Jianhang, Y., Hongtao, W and Yi S.: Self-supervised category-level 6D object pose estimation with deep implicit shape representation. Proceedings of the AAAI Conference on Artificial Intelligence, 2022.

22. Jun, Z., Kai, C., Linlin, X., Qi, D., Jing, Q.; Deep fusion transformer network wi -th weighted vector-wise keypoints voting for robust 6D object pose estimation. Pr -oceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023, pp. 13967-13977.

23. Weitong, H., Zhongxiang, Z., Jun, W., Huang, H., Yue, W., Rong, X.: REDE: En- d-to-end object 6D pose robust estimation using differentiable outliers' elimination. In https://arxiv.org/pdf/2010.12807, 2020.

24. Heng, Y., Marco, P.: Object pose estimation with statistical guarantees: conformal keypoint detection and geometric uncertainty propagation. In CVPR, 2023.

25. Hu, Y., Fua, P., Wang, W., Salzmann, M.: Single stage 6D object pose estimatio-n. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and P-attern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 2927–2936.

26. Strutz T.: Data fitting and uncertainty (2nd edition) [M]// data fitting and uncertai-nty. Vieweg+Teubner, 2016.

27. Ross, G., Jeff, D., Trevor, D., Jitendra, M.: Rich feature hierarchies for accurate o-bject detection and semantic segmentation. In CVPR, 2013.

28. Xu, Y., Kunbo, L., Jinge, W., Xiumin, F.: ER-Pose: Learning edge representation for 6D pose estimation of texture-less objects. In Neurocomputing, Volume 515, 2-023.

29. Zhigang, L., Gu, W., Xiangyang J.: CDPN: Coordinates-based disentangled pose n-etwork for real-time RGB-based 6-DoF object pose estimation. In ICCV, 201-9.

30. Gu, W., Fabian, M., Federico, T., Xiangyang, J.: GDR-Net: Geometry-guided dire-ct regression network for monocular 6D object pose estimation. In CVPR, 2021.

31. Sergey, Z., Ivan, S., Slobodan, I.: DPOD: 6D pose object detector and refiner. In CVPR, 2019.

32. Hansheng, C., Wei, T., Pichao, W., Fan, W., Lu, X., Hao, L.: EPro-PnP: Generali-zed end-to-end probabilistic perspective-n-points for monocular object pose estimat e-on. In CVPR, 2023.

33. Lin, T., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object de-tection. IEEE Transactions on Pattern Analysis and Machine Intelligence 42(02), 318–327 (2020).

34. Girshick, R.: Fast R-CNN. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1440–1448 (2015).

35. Hinterstoisser, S., Cagniart, C., Ilic, S., Sturm, P., Navab, N., Fua, P., Lepetit, V.: Gradient response maps for real-time detection of texture-less objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 34(5), 876–888 (2012).

36. Brachmann, E., Krull, A., Michel, F., Gumhold, S., Shotton, J., Rother, C.: Learni -ng 6D object pose estimation using 3D object coordinates. In Computer Vision E CCV 2014, pp. 536–551 (2014).

37. Brachmann, E., Michel, F., Krull, A., Yang, M.Y., Gumhold, S., Rother, C.: Uncer tainty-driven 6D pose estimation of objects and scenes from a single RGB image. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3364–3372 (201-6).

38. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., Nava- b, N.: Model based training, detection and pose estimation of texture-less 3D obje -cts in heavily cluttered scenes. In Computer Vision – ACCV 2012, pp. 548–562 (2013).

39. Tekin, B., Sinha, S., Fua, P.: Realtime seamless single shot 6D object pose predic -tion. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognitio- n (CVPR), pp. 2-92–301 (2018).

40. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: SSD-6D: Making rgb-bas -ed 3D detection and 6D pose estimation great again. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 1530–1538 (2017).

41. Wang, C., Xu, D., Zhu, Y., Martin-Martin, R., Lu, C., Fei-Fei, L., Savarese, S.:

Dense Fusion: 6D object pose estimation by iterative dense fusion. In: 2019 IEEE /CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3338 –3347 (2019).

42. Park, K., Patten, T., Vincze, M.: Pix2Pose: Pixel-wise coordinate regression of obj -ects for 6D pose estimation. In: 2019 IEEE/CVF International Conference on Co mputer Vision (ICCV), pp. 7667–7676 (2019).

43. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. ArXiv abs/1711.00199 (2018).

44. Wang, G., Manhardt, F., Tombari, F., Ji, X.: GDRNet: Geometry guided direct reg -ression network for monocular 6D object pose estimation. In: 2021 IEEE/CVF C o-nference on Computer Vision and Pattern Recognition (CVPR), pp. 16606–16616 (2021).

45. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2-016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016).

46. Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., Sun, J.: DetNet: Design backbone for object detection. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision–ECCV 2018, pp. 339–354 (2018).