DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Semester's Thesis in Informatics

# Semester Thesis

Tao Zhou

DEPARTMENT OF INFORMATICS

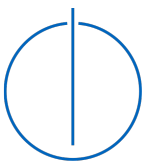
TECHNISCHE UNIVERSITÄT MÜNCHEN

Semester's Thesis in Informatics

# Semester Thesis

# A review on instance-level 6D object pose estimation

| | |
|---|---|
| Author: | Tao Zhou |
| Supervisor: | Prof. Dr.-Ing. habil. Alois Christian Knoll |
| Advisor: | Dr. Yinyu Nie |
| Submission Date: | 16.03.2023 |



I confirm that this semester's thesis in informatics is my own work and I have documented all sources and material used.

Munich, 16.03.2023 Tao Zhou

# Acknowledgments

I would like to express my deep gratitude to all those who have helped and supported me throughout the project of this semester thesis.

First, I would like to thank my advisor, Dr. Yinyu Nie, for his invaluable guidance, support, and encouragement throughout my research journey. His expertise and insight are invaluable in shaping my understanding of the field and helping me to refine my arguments. I would also like to thank my family and friends for their constant encouragement, support, and motivation during the challenging times of the writing process. Finally, I would like to acknowledge Prof. Dr.-Ing. habil. Alois Christian Knoll for the support, which made it possible for me to complete this semester thesis.

I am truly grateful for all the help and support I have received, and I could not have completed this thesis without the contributions of each and every one of these individuals and organizations. Thank you all from the bottom of my heart.

# Abstract

6D object pose estimation of rigid objects has received increasing attention in computer vision communities, which brings broad applications in robotics, augmented reality, autonomous driving, etc. Existing 6D object pose estimation methods can be divided into two categories by their semantic level: instance-level or category-level. In particular, instance-level 6D pose estimation goes further beyond object category semantics and focuses per-object perception. It opens up a wide applications in robotic manipulation and grasping. This paper mainly review, analyze and summarize the prior-art methods on instance-level 6D object pose estimation based on deep neural networks. First, we introduce a taxonomy of 6D object pose estimation methods based on input types and network characteristics. Secondly, we implement these methods by training on the unified benchmarks, e.g., LIMEMOD, and analyze the quantitative and qualitative results, with a summary of the major advantages and disadvantages of exiting backbones, modules and designs. Finally, we will examine potential avenues for future research, which we hope can be used as a guidance for practitioners working on 6D object pose estimation.

# Contents

# 1 Introduction

6D object pose estimation has been playing a significant role in computer vision and robotics in recent years, which focuses on determining the estimation of 3D translation **T** ($T_x$, $T_y$, $T_z$ ) and 3D rotation **R** (pitch, yaw, and roll) of rigid objects from the object's coordinate system to the camera's coordinate system, as illustrated in Figure 1.1.

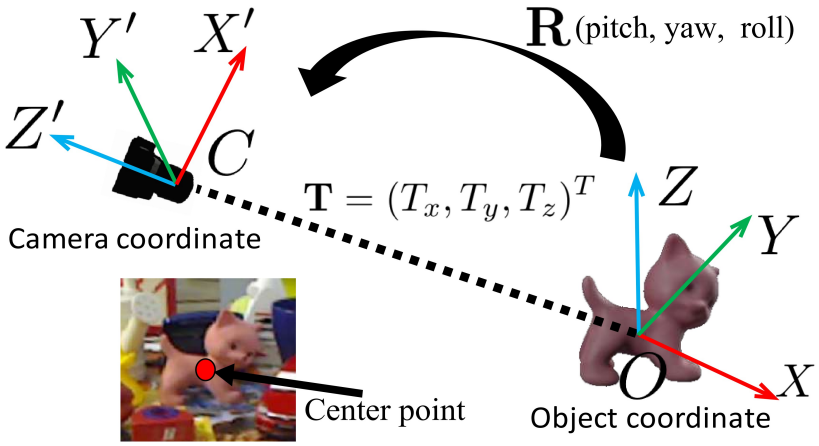
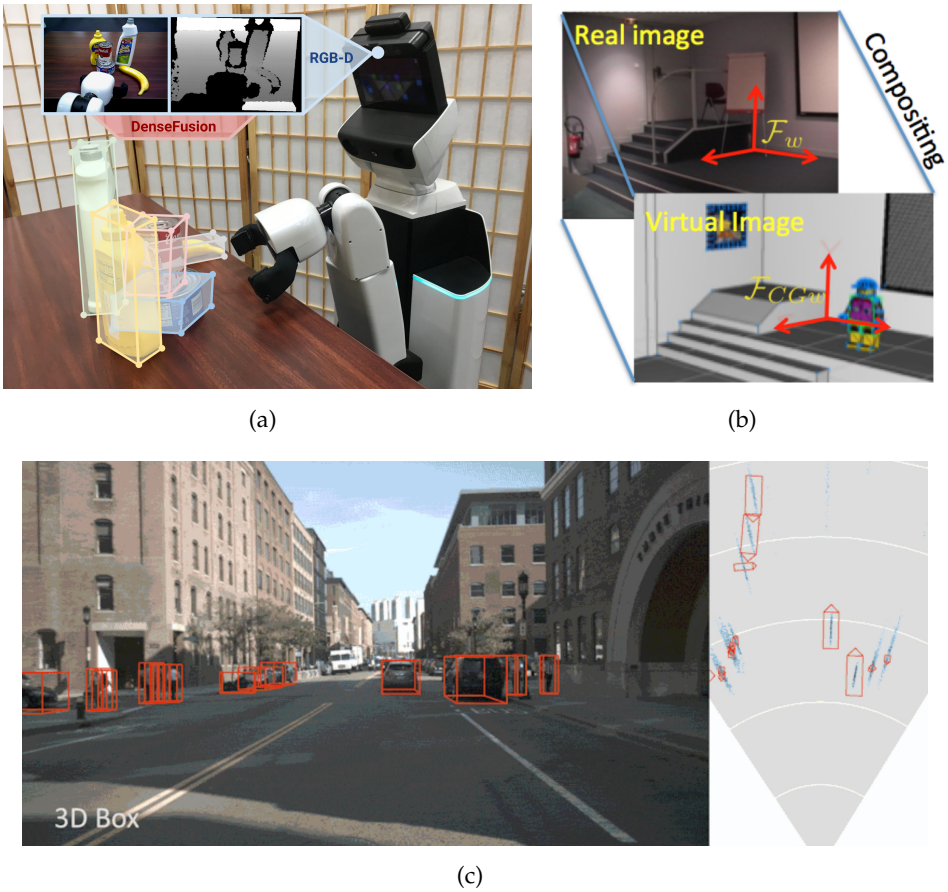



Figure 1.1: Illustration of 6D object pose estimation [1], [2]

6D object pose estimation is also treated as a prediction problem, in which the task is to predict the 6D pose of a specific instance of an object from an input image (e.g., RGB or RGBD) [3]. It's typically divided into two categories based on their semantic level: instance-level and category-level. "Instance-level" is related to a specific instance of an object (e.g., a particular chair), while "category-level" is related to a specific category (e.g., all chairs). In this paper, we will focus specifically on instance-level 6D object pose estimation, which goes beyond object category semantics and focuses per-object perception.

With the ongoing advancements in computer vision techniques, it has been widely applied in the variety of realm, including robotics, autonomous driving, augmented reality (AR), etc., as shown in Figure 1.2. In Robotics, it can be used to generate grasp pose for known objects and it's a way to increase the accuracy and simplicity of the grasp pose in robotics manipulation [4]. In autonomous driving, it can improve

an autonomous vehicle to efficiently understand its surrounding and make informed decisions about how to avoid collisions, follow traffic rules by detecting object and estimating object pose [5]. In AR, it's used to tackle a motion tracking task and enable the AR to provide a more immersive and realistic experience for users by accurately tracking the pose and motion of objects in the real world [6].




(a) (b)

(c)

Figure 1.2: (a) Robotic grasping and manipulation [7] (b) Augmented reality [6] (c) Autonomous driving [8]

Despite the significant advances in 6D object pose estimation, there are still several key challenges that need to be addressed to make this technology more widely applicable in various applications, including Clutter, Occlusion, Viewpoint and Illumination variation, Real-time constraints, etc. In the subsequent section, we will provide more detail on these challenges.

Over the last decade, there have been significant improvement, with the development of new state-of-the-art methods that address many of the challenges associated with this task. These methods can be broadly divided into two main categories: traditional methods and deep learning-based methods.

Traditional methods include template-based method, Point-Based method, etc. For template-based method, it uses a pre-defined template of the object to detect and align features in a reference image to estimate the object's pose. Nevertheless, this method is sensitive to the illumination variation and cluttered scenes [2]. For point-based method, it estimates the 6D object pose by identifying feature points on a point cloud, matching them to a pre-defined model, and aligning the resulting point pairs to estimate the object pose. This method overcomes the shortage of the templated-based method that it can work well in cluttered scenes. Notwithstanding that, it can be affected to the initial alignment of the point pairs and may perform poorly on texture-less objects [9].

Overall, while traditional methods have been addressed some challenges and widely used, these methods are not without their limitations compared with deep learning-based method. For instance, traditional methods may struggle to accurately estimate object's pose with complex shape or symmetries, not generalize well to unknown objects, sometimes require hand-crafted feature, which may be time-consuming.

In view of the rapid developments of the deep learning, especially Convolutional Neural Network (CNN), the above-mentioned limitations have been achieved remarkable improvements. Deep learning-based methods can automatically learn to extract features from big Datasets by using CNN instead of hand-crafted [10]. And deep learning models can often handle noisy and varying data, which is robust for the generalization. The use of various deep learning architectures has greatly improved the accuracy and speed of 6D object pose estimation compared with traditional methods. CNN, RNN, and other architectures have been extensively used in this field and have shown promising results, which are able to handle noisy data more effectively. For example, the CNN architecture has been used in methods such as PoseCNN [1], DenseFusion [7], which have achieved high accuracy in 6D pose estimation. RNNs, on the other hand, have been used in methods such as RNNPose [11], which can incorporate temporal information to improve the accuracy of pose estimation. Simultaneously, deep learning-based methods can be used for 6D pose estimation using either RGB or RGB-D input. In both cases, the methods can also afford accurate results. The choice of input will depend on some specific applications and the availability of depth information.

In this paper, we will review the current state-of-the-art methods in instance-level 6D object pose estimation based on deep neural networks discussing the strengths and limitations of different approaches and highlighting key challenges that need to be addressed in future research. We will evaluate these methods on unified benchmarks, e.g., LIMEMOD Datasets [2]. Our core contributions are as follows:

- We introduce a taxonomy method based on input types and network characteristics.
- Summary of the major pros and cons of exiting backbones, modules and designs.
- Quantitatively and qualitatively analyzing the existing methods.
- Explore possible avenues for future research.

We hope this survey can be used as a guidance for practitioners working on 6D object pose estimation.

# 2 Challenges

In the previous section, we briefly mentioned several key challenges that impact 6D object pose estimation, including Clutter, Occlusion, Viewpoint and Illumination variation, Real-time constraints, etc. Figure 2.1 provides examples of some challenges. Next, we will delve deeper into these challenges and discuss how they impact the task of 6D object pose estimation. This section will help us gain a deeper understanding of the strengths and limitations of the various methods that will be discussed.

- **Clutter:** In cluttered scenes, there may be multiple objects or distractors present, making it difficult to identify and localize the object of interest.
- **Occlusion:** Parts of the object may be occluded or hidden by other objects in the scene, that is not visible to the camera.
- **Viewpoint variation:** The object pose can vary depending on the viewpoint from which it is observed. This can be especially challenging for symmetrical objects.
- **Illumination variation:** Variations in lighting conditions can affect the texture and appearance of the object.
- **Real-time constraints:** In some applications, the object pose must be estimated in real-time, with high accuracy and fast computation, in order to enable specific tasks to respond quickly to changes in its environment.

(a) (b)

Figure 2.1: (a) Occlusion scenarios [12], (b) Symmetric object (eggbox) [2], [13]

# 3 Traditional Methods

Traditional methods have played a significant role in the development of 6D object pose estimation, including template-based and point-based methods. In this section, we will introduce some classical methods, that we can gain insight into the evolution of 6D object pose estimation.

**The template-based method** uses a pre-defined template of the object to estimate its pose in an image by aligning the template with the detected features in the image, as illustrated in Figure 3.1. Hinterstoisser et al. [2] proposed a template-based method for automatic modeling, detection, and tracking of 3D objects. This method is robust for texture-less objects and cluttered scenes, as it can build templates automatically, extract features from a test image and use these features to match templates to estimate the object's pose. However, the method is sensitive to illumination variations, which can make it difficult to accurately align the template with the image.

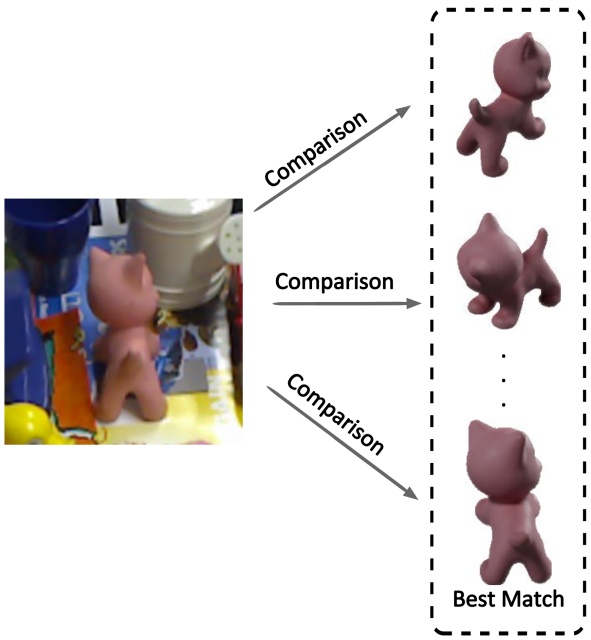



Figure 3.1: Illustration of template-matching method [13], [14]

**The point-based method** estimates the 6D object pose by identifying feature points on a point cloud, matching them to corresponding points on a pre-defined global model, and aligning the resulting point pairs, as illustrated in Figure 3.2. Drost et al. [9] proposed a point-based method for object pose estimation. This method creates a global model description based on oriented point pair features and uses a fast voting scheme to match this model locally and estimate the object pose. This method overcomes the shortage of the templated-based method that it doesn't rely on a pre-defined template. However, one of limitation is that it can be sensitive to the initial alignment of the point pairs, which can impact the accuracy of the pose estimation.

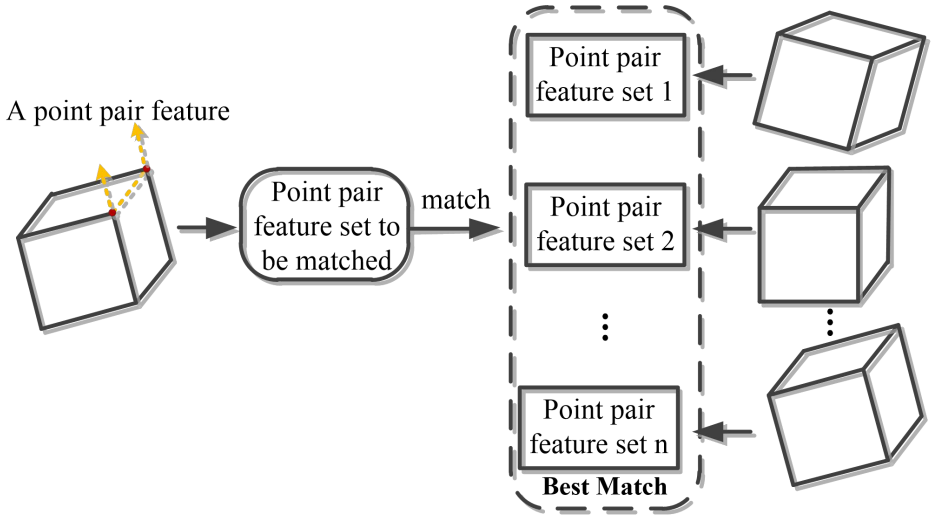



Figure 3.2: Illustration of point-based method [14]

Overall, as the demands for high accuracy and speed in 6D object pose estimation continue to increase in various applications, traditional methods may no longer be sufficient. With the advancements in deep learning, these methods are being increasingly replaced by newer approaches.

# 4 Deep learning-based Methods

Over the past decades, several key novel deep learning architectures have been presented, including CNN, RNN, etc. The different model architectures and input types are crucial factors in affecting the performance of 6D object pose estimation. In this section, we review the recent state-of-the-art methods based on deep learning, grouping them based on their input type (image-based) and network characteristics. These methods have been influential in advancing the field of 6D object pose estimation due to their ability to handle complex real-world scenarios and achieve high accuracy.

## 4.1 The RGB Image-based Deep Learning Method

RGB images, made up of three channels (red, green, and blue) per pixel, are widely used in deep learning-based 6D object pose estimation. As depicted in Figure 4.1, we provide an overview of the process for some RGB image-based deep learning methods. One reason for this is that RGB images are typically smaller in size compared to RGB-D images, making them easier to store, process, and transmit due to the reduced computational resources required for deep learning. However, they are susceptible to interference from factors such as illumination and viewpoint.

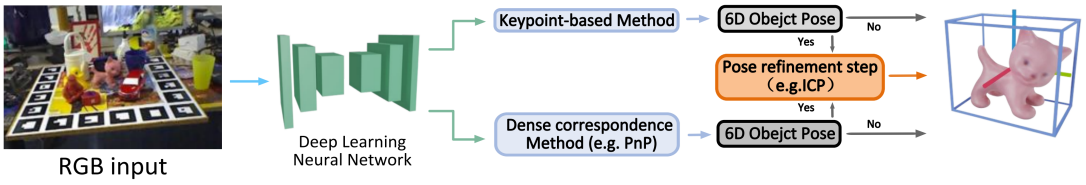



Figure 4.1: Overview of RGB image-based deep learning methods [15]

CNN is widely used deep learning networks that can take RGB images as inputs and automatically detects the features rather than relying on handcrafted features [16], [17]. A variety of classic architectures of CNN are commonly used as the basic network in 6D object pose estimation, including VGGNet [18], ResNet [19], etc. However, some methods utilize custom network architectures specifically designed for pose estimation tasks, in order to better meet the unique requirements of these tasks.

The Pixel-wise Voting Network (PVNet) [20] has been proposed as the backbone for predicting vector fields and performing semantic segmentation, as shown in Figure 4.2. These vectors are then used to localize 2D key points through RANSAC-based voting. Finally, the PnP-algorithm is applied to obtain the 6D pose estimation. However, accurate estimation of the position of 2D keypoints is essential, which can be influenced by various factors such as occlusion and the design of network architectures. On the other hand, the RANSAC algorithm [21] is time-consuming due to its iterative process.

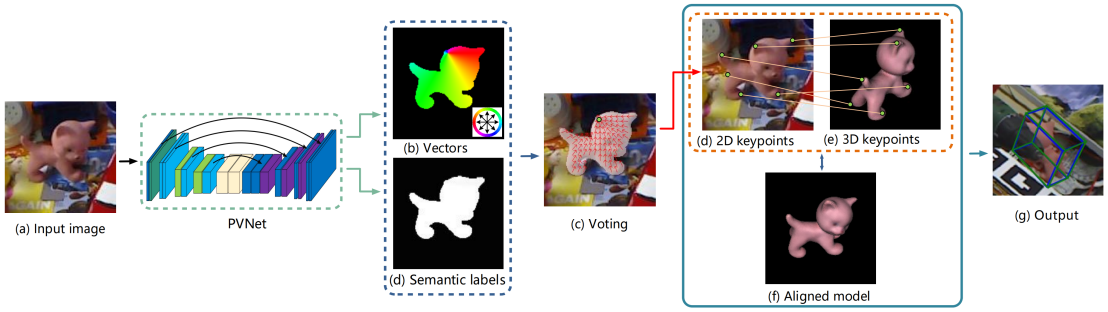



Figure 4.2: Overview of PVNet based method [20]

Wang et al. [22] introduced the Geometry-Guided Direct Regression Network (**GDR-Net**), an end-to-end deep learning method that utilizes CNN and learnable Patch-PnP to return the 6D object pose from a RGB image by utilizing intermediate geometric representations based on dense correspondence, as shown in Figure 4.3. GDR-Net employs a novel continuous 6D representation for **R** under 3D object translation, which is robust against viewpoint variation. Its performance is superior to PVNet due to the novel design of the network and Parameterization of **R** and **T**. Nonetheless, the lack of pose refinement may be a limitation of the GDR-Net approach and could be addressed in future work by integrating a refinement network, thereby making it more accurate.

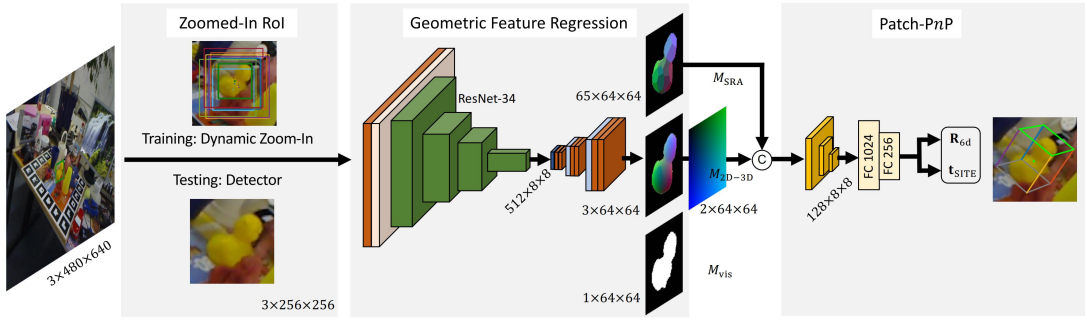



Figure 4.3: Overview of GDR-Net based Method [22]

As previously mentioned, one limitation in object pose estimation has been the lack of pose refinement. However, recent advancements have utilized the differentiability of the optimization algorithm to refine the object pose, resulting in state-of-the-art performance.

A novel end-to-end CNN-based method called End-to-End Probabilistic Perspective-n-Points (**EPro-PnP**) [8] has been proposed, as illustrated in Figure 4.4. This method translates the object pose from a non-differentiable deterministic pose into a differentiable probabilistic density of pose. To make the process differentiable and allow for end-to-end training, EPro-PnP uses the SoftMax to smooth the probability density of the object pose, rather than using the "argmin" function which is non-differentiable in the optimal layer. EPro-PnP is inserted into the CDPN [23] framework, which is a method for learning 2D-3D correspondences of objects by backpropagating the probability density of poses for the purpose of object pose estimation. By using the EPro-PnP with CDPN, this method can allow for stable end-to-end training and rapid improvement in accuracy and robustness through end-to-end training.

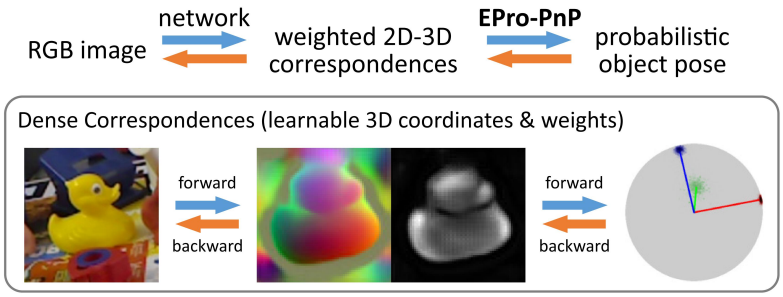



Figure 4.4: Overview of EPro-PnP based Method [8]

In addition to using CNN, RNN also plays an important role in object pose refinement. Xu et al. [11] proposed an RNN-based end-to-end training framework for object pose refinement called **RNNPose**, as depicted in Figure 4.5. This framework treats the object pose refinement as a nonlinear least squares problem based on the estimated correspondence field between a rendered reference image and the observed target image, which is solved using a differentiable Levenberg-Marquardt algorithm. RNNPose was integrated into PVNet [20] framework. The reference image is rendered using the object's CAD model and an initial pose estimate, which are as inputs. And these inputs are used to generate an optimized object pose which is then used in the next-iteration estimations. After several recurrent iterations, this framework returns a refine object pose, achieving state-of-the-art performance. It is also robust against erroneous pose initialization and occlusions. However, one limitation of this method is that it requires a CAD model of the object, which limits its ability to generalize to unknown objects.

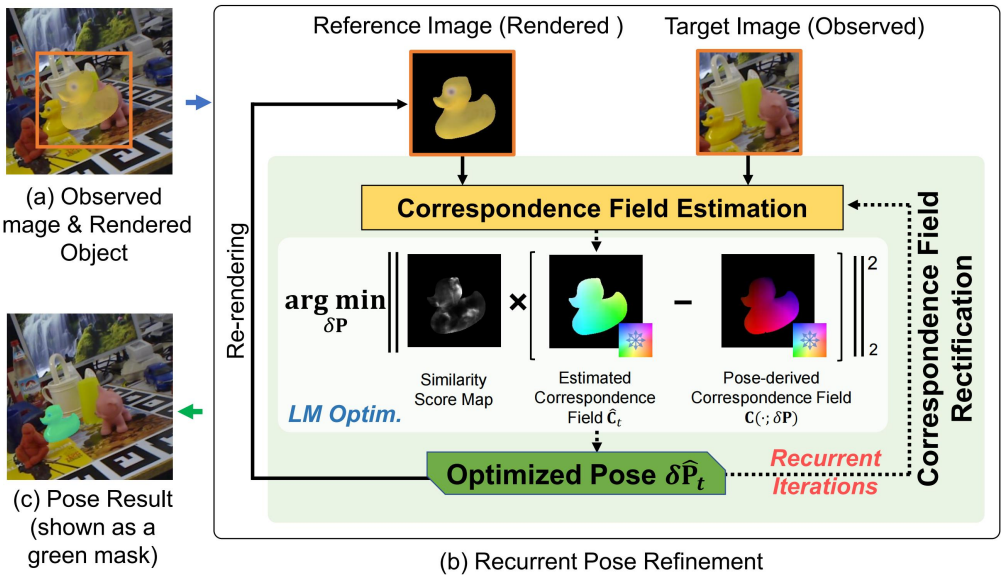



Figure 4.5: Overview of RNNPose [11]

## 4.2 The RGB-D Image-based Deep Learning Method

RGB-D images, which consist of both RGB and depth information, have become increasingly popular in deep learning methods due to the availability of cheap and high-quality RGB-D sensors, which allow for the addition of depth to regular RGB images [24]. At the same time, with the development of more powerful and efficient hardware and software, it has become possible to perform 6D pose estimation using deep learning methods with relatively low computational resources.

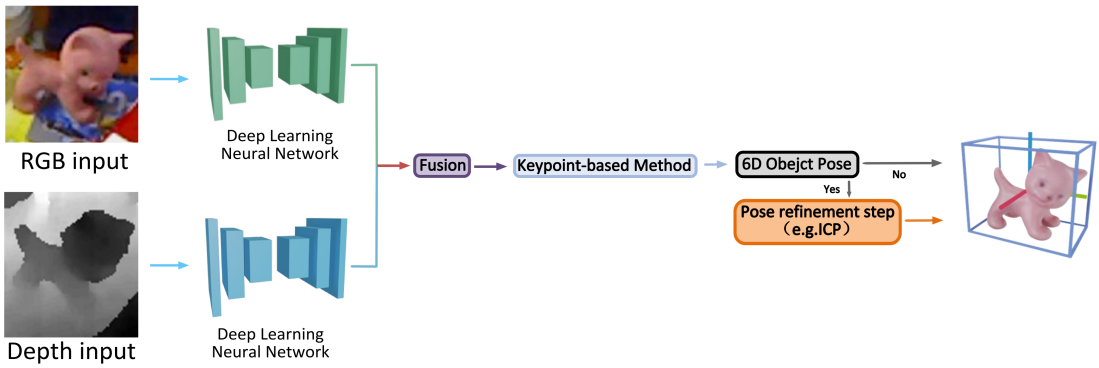



Figure 4.6: Overview of RGB-D image-based deep learning methods [15], [25]

**PVN3D** [26] proposes a deep 3D keypoint Hough voting network that includes instance semantic segmentation, as shown in Figure 4.7. This network fuses features extracted from PSPNet [27] based on ResNet34 [19] and PointNet++ [28], which include appearance information and geometry information. It can detect and select the 3D keypoints of the target object using these features and Hough voting. The method then uses least squares fitting to estimate the 6D pose parameters, taking advantage of the geometric constraint information of rigid objects. This method is designed to improve upon 2D keypoint-based methods, such as PVNet [20], which can suffer from large errors when applied to the real 3D world due to the lack of depth information. It achieves outperforming performance, but its performance depends on the accuracy of the detected 3D keypoints.

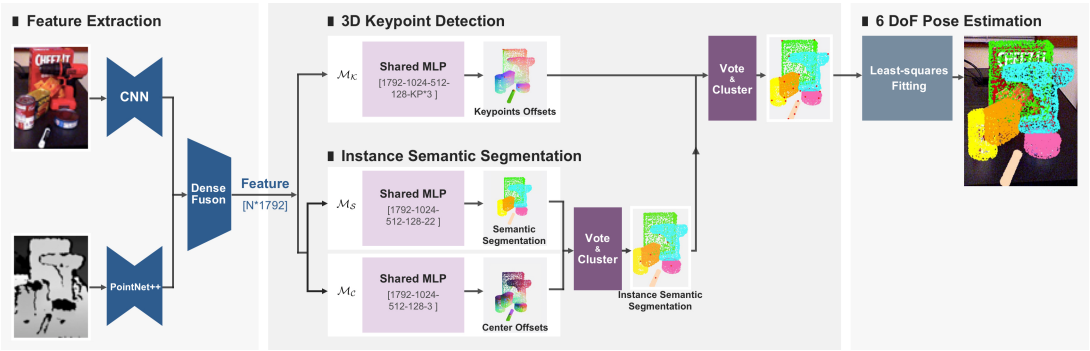



Figure 4.7: Overviews of PVN3D Method [26]

In PVN3D, the networks extract features from different modalities of data separately without any communication until the final layers. **FFB6D** [29] proposes a full flow bidirectional fusion network, which obtains appearance information from the CNN branch and geometry information from a point cloud network (PCN), as shown in Figure 4.8. The fusion is performed on each encoding and decoding layer in order to bridge the information gap and make the information complementary. This method recovers the object pose parameters, similar to PVN3D [26]. Additionally, FFB6D also proposes a SIFT-FPS algorithm which is used to automatically select 3D keypoints, especially in non-salient regions such as smooth surfaces without distinctive texture. This method can achieve superior performance with fewer parameters and faster real-time processing without the need for a time-consuming post-refinement procedure. However, it is not robust in the presence of occlusion, which may be due to the lack of a post-refinement procedure.

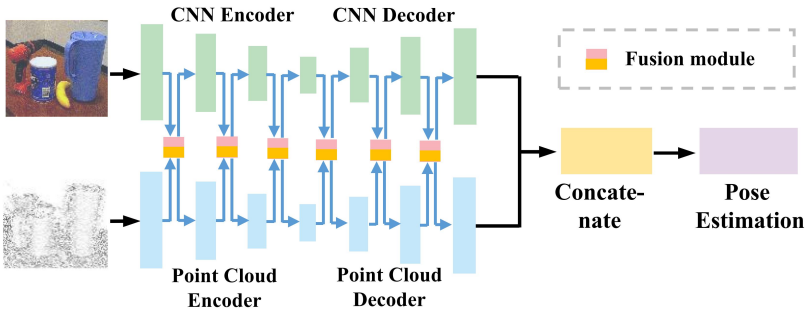



Figure 4.8: Overview of FFB6D Method [29]

The **RCVPose** method proposed by Xu et al. [30] is a novel approach for 6D object pose estimation that uses intersecting spheres and a radial keypoint voting scheme, as illustrated in Figure 4.9. In the training phase, RCVPose uses a CNN to predict the distance, a 1D quantity, between the 3D point corresponding to the depth of each RGB pixel and a set of dispersed keypoints in the object frame. During inference, the method generates a sphere centered at each 3D point and estimates the distance, which is equal to the radius of the sphere. The surfaces of these spheres are used to increment a 3D accumulator space, and the peaks in this space indicate keypoint locations. After applying an iterative closest point (ICP) algorithm for pose refinement, RCVPose has been shown to outperform other methods such as PVNet (2D keypoints based method) and PVN3D (3D keypoints based method) in occlusion and clutter scenarios, even with fewer and more dispersed keypoints. Nevertheless, a limitation of this method is that it is not an end-to-end deep learning approach, as the training, inference, and pose refinement steps are all done separately, which can be time-consuming.

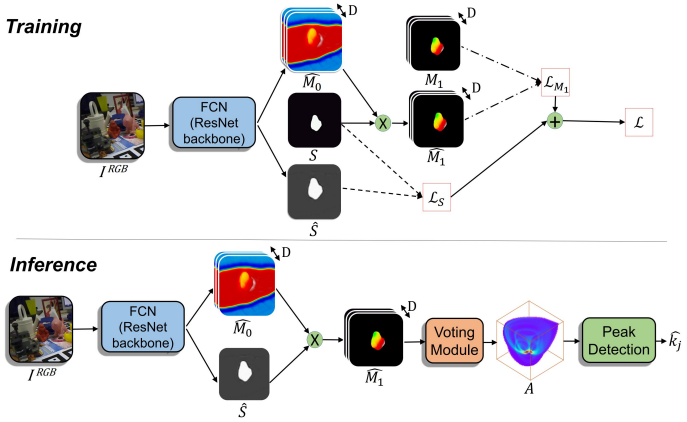



Figure 4.9: Overviews of RCVPose Method [30]

# 5 Experiments

In this section, we will provide an overview of the benchmark datasets and evaluation metrics utilized in the previously mentioned methods. We will explain the process for implementing and evaluating the methods proposed in the methods on the LineMOD datasets and the ADD(-S) metric. Furthermore, we will exhibit the Quantitative and qualitative evaluation results.

## 5.1 Benchmark Datasets

**The LineMOD Datasets (LM)** [2] is widely used as a standard benchmark for research on 6D object pose estimation. This dataset consists of 15 sequences, for each of which contains around 1.2k images and is annotated with the 6D pose and instance mask of the object. The dataset includes both textured and texture-less objects and are captured under a variety of illumination conditions and in cluttered scenes, making the dataset challenging for algorithms to work with. More detail about this dataset can be found in Table 5.1 and Figure 5.1.

| Benchmarks | Categories | Format | Resolution |
|---|---|---|---|
| LineMOD | 15 | RGB-D | 640 $\times$ 480 |

Table 5.1: Statistics of LineMOD Datasets [2], [14], [31]

Figure 5.1: Overviews of the LineMOD Datasets. [2], [32]

## 5.2 Evaluation Metrics

To evaluate the previously mentioned methods, we utilize the commonly employed metrics of ADD [2] and ADD-S [1], [33], collectively referred to as ADD(-S).

**ADD (Average Distance Deviation)** measures the average distance between the 3D model points of a target object in its ground truth pose and its estimated pose, by calculating the mean of the pairwise distances between the transformed model points. It is commonly used for asymmetric objects and is computed over all model points $x$ of the set of 3D object model points $\mathcal{M}$ of the target object. The equation for ADD is given by Equation 5.1.

$$\mathrm{ADD} = \frac{1}{n} \sum_{\mathrm{x} \in \mathcal{M}} \| \left(\mathrm{Rx} + \mathrm{T}\right) - \left(\tilde{\mathrm{R}}\mathrm{x} + \tilde{\mathrm{T}}\right) \| \tag{5.1}$$

where $n$ denotes the number of points in the set of 3D object model points $\mathcal{M}$, and $R$ and $T$ represent the ground truth rotation and translation, respectively. $\tilde{\mathrm{R}}$ and $\tilde{\mathrm{T}}$ represent the estimated rotation and translation, respectively.

**ADD-S** is a variant of ADD and it calculates the average distance from each 3D model point transformed by the estimated pose $\tilde{\mathrm{R}}$ and $\tilde{\mathrm{T}}$ to its closest point on the target model transformed by the ground truth pose $R$ and $T$. The ADD-S is used for symmetric objects as it takes into account the closest point on the surface of the model. The equation for ADD-S is given by Equation 5.2.

$$\mathrm{ADD-S} = \frac{1}{n} \sum_{x_1 \in \mathcal{M}} \min_{x_2 \in \mathcal{M}} \| \left(\mathrm{Rx}_1 + \mathrm{T}\right) - \left(\tilde{\mathrm{R}}\mathrm{x}_2 + \tilde{\mathrm{T}}\right) \| \tag{5.2}$$

The 6D object pose is considered accurate if the average distance, as calculated by ADD or ADD-S, is less than a predefined threshold, as indicated by the inequality Equation 5.3.

$$ADD\,(-S) \leq z_{ADD(-S)} \Phi \tag{5.3}$$

where $\Phi$ is the diameter of 3D object model and $z_{ADD(-S)}$ is a constant that determines the threshold for determining if the 6D object pose is considered accurate [34], [35].

## 5.3 Evaluation on Benchmark Datasets

In this review, the deep learning-based methods in our experiments were implemented using the PyTorch framework and evaluated on a GPU 3070Ti running on Ubuntu 20.04 and we quantified the performance of the previously discussed methods by using the ADD(-S) metric on the LineMOD dataset. The threshold for this metric was set to 10%

of the 3D object model's diameter, represented as the coefficient $z_{ADD(-S)}$ of 0.1, and referred to as ADD(-S)-0.1.

The accuracy results of the quantitative evaluation for the deep learning methods for 6D object pose estimation are presented in Table 5.2, showcasing the performance of each method on each object of the LineMOD dataset. These results reveal how the shape of different objects impacts the performance of each method.

| Methods | PVNet | GDR-Net | EPro-PnP | PVNet+ RNNPose | PVN3D | FFB6D | RCVPose | RCVPose +ICP |
|---|---|---|---|---|---|---|---|---|
| ape | 50.48 | 76.29 | 85.71 | 85.62 | 100.00 | 100.00 | 99.20 | 99.60 |
| benchvise | 99.81 | 97.96 | 99.71 | 100.00 | 99.61 | 100.00 | 99.60 | 99.70 |
| camera | 86.47 | 95.29 | 97.45 | 98.43 | 99.61 | 100.00 | 99.70 | 99.70 |
| can | 95.37 | 97.93 | 99.41 | 99.51 | 99.51 | 99.90 | 99.00 | 99.30 |
| cat | 78.94 | 93.11 | 95.01 | 96.41 | 99.90 | 99.80 | 99.40 | 99.70 |
| driller | 96.33 | 97.72 | 98.12 | 99.50 | 99.31 | 100.00 | 99.70 | 100.00 |
| duck | 57.56 | 80.28 | 84.98 | 89.67 | 97.93 | 98.45 | 99.40 | 99.70 |
| eggbox | 99.25 | 99.53 | 99.91 | 100.00 | 98.97 | 100.00 | 98.70 | 99.30 |
| glue | 95.85 | 98.94 | 99.52 | 97.30 | 99.71 | 99.90 | 99.70 | 100.00 |
| holepuncher | 82.20 | 91.15 | 94.01 | 97.15 | 100.00 | 100.00 | 99.80 | 100.00 |
| iron | 98.77 | 98.16 | 99.08 | 100.00 | 99.69 | 100.00 | 99.90 | 99.90 |
| lamp | 99.42 | 99.14 | 99.42 | 100.00 | 99.81 | 100.00 | 99.20 | 99.50 |
| phone | 92.60 | 92.26 | 97.73 | 98.68 | 99.52 | 99.90 | 99.10 | 99.70 |
| AVERAGE | 87.16 | 93.67 | 96.16 | 97.10 | 99.51 | 99.84 | 99.40 | 99.70 |

Table 5.2: LineMod accuracy results for each object (ADD(-S)-0.1)

In Table 5.3, we present a comparison of various 6D object pose estimation methods based on their input, deep learning architecture, and whether it has a refinement step to optimize its 6D pose. This comparison will provide valuable insights for future research in the field, which will be further discussed later.

| Methods | Year | Input | Trained classifier | Refinement step | LM ADD(-S) |
|---|---|---|---|---|---|
| PVNet [20] | 2019 | RGB | CNN | ✗ | 87.16 |
| GDR-Net [22] | 2021 | RGB | CNN | ✗ | 93.67 |
| EPro-PnP [8] | 2022 | RGB | CNN | ✗ | 96.16 |
| PVNet+RNNPose [11] | 2022 | RGB | CNN+RNN | RNNPose | 97.10 |
| PVN3D [26] | 2020 | RGBD | CNN | ✗ | 99.51 |
| FFB6D [29] | 2021 | RGBD | CNN | ✗ | 99.84 |
| RCVPose [30] | 2022 | RGBD | CNN | ✗ | 99.40 |
| RCVPose+ICP [30] | 2022 | RGBD | CNN | ICP | 99.70 |

Table 5.3: Comparisons of the RGB/RGB-D image-based deep learning methods

In addition, we have also included method performance curves to provide a visual comparison of the effectiveness of RGB and RGB-D image-based deep learning methods. These can be found in Figure 5.2.

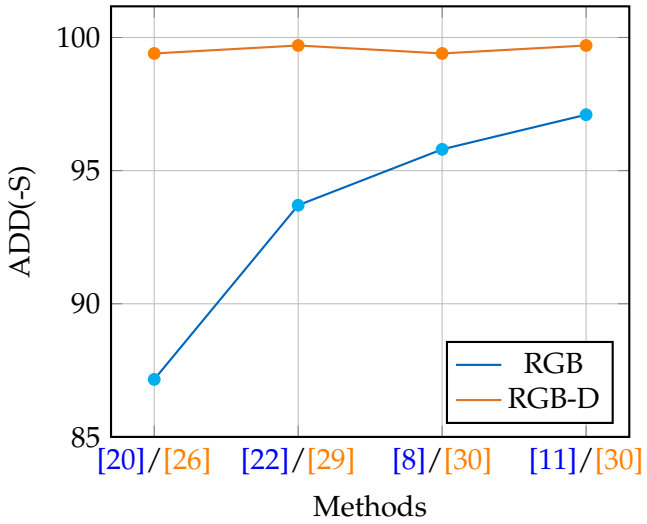



Figure 5.2: Comparisons between the RGB and RGB-D image-based methods

As shown in Figure 5.2, in the RGB image-based curve, the performance of RGB image-based deep learning methods continues to improve with the refinement of previous networks and the introduction of novel networks. We can see that the novel CNN-based end-to-end networks proposed in GDR-Net, and EPro-PnP significantly outperform PVNet. Furthermore, we note that RNNPose, which uses a refinement step and is based on a RNN network, improves accuracy, but at the cost of increased computation time compared to the end-to-end methods. In the RGB-D image-based curve, it is evident that the inclusion of depth information improves performance by allowing for better utilization of both appearance and geometry information. A comparison of PVN3D and FFB6D highlights that the method in which RGB, and depth information is processed also plays a role in determining accuracy, with FFB6D achieving relatively good results. The results of RCVPose and RCVPose with the added refinement step, ICP, also demonstrate that incorporating a refinement step improves performance. Overall, RGB-D image-based methods generally perform better than RGB image-based methods, and methods incorporating a refinement step achieve state-of-the-art results. In the next section, we delve deeper into potential directions for future research based on our experimental data.

# 6 Future research directions

In the previous section, we analyzed and obtained both quantitative and qualitative results for each method and discussed the current state of the research in the field of 6D pose estimation. Now, in this section, we will explore possible avenues for future research.

As highlighted in the evaluation of the experiments, it has been demonstrated that RGB-D image-based methods are significantly superior to RGB image-based methods due to the additional geometric information they provide. This allows the methods to have a better understanding of the environment and improve the accuracy of 6D pose estimation. In future research, there may be opportunities to delve deeper into the fusion of depth information with RGB information to enhance the performance of 6D pose estimation methods even further. One potential avenue of research could be to explore various depth fusion techniques, such as early, intermediate, or late fusion, to determine the optimal approach for combining the appearance information from RGB images with the geometric information from depth information.

The design of a novel network architecture is essential for achieving state-of-the-art performance in future 6D object pose estimation. While CNN-based networks have been commonly used, RNN-based networks like RNNPose have shown superior performance by incorporating time sequence and temporal information to track the 6D pose of a target object in a video sequence and improve accuracy for occlusion scenarios. In contrast, CNN can only process individual frames of a video [14], [31]. Additionally, using a transformer network [36] may also enhance the accuracy of 6D object pose estimation.

RNNPose [11] and RCVPose [30] with refinement step have achieved state-of-the-art performance, however, it requires multiple stages and is time-consuming. To optimize the network architecture, we can make it an end-to-end method by incorporating a multi-task learning approach. This approach not only improves the accuracy of the 6D object pose estimation but also addresses the challenge of real-time constraints. This is particularly useful in real-time scenarios such as autonomous driving and robot manipulation fields where fast and accurate 6D object pose estimation is critical.

In conclusion, for future research directions, the design of novel end-to-end network architectures should be tailored specifically to the task of 6D object pose estimation, taking into account the challenges and limitations of this task. By incorporating a

combination of different architectures and approaches, researchers can optimize the performance of the network and achieve more accurate and efficient 6D object pose estimation. An illustration of an end-to-end deep learning-based method, which includes a pose refinement step, as depicted in Figure 6.1, can serve as a useful guide for future research in this field.

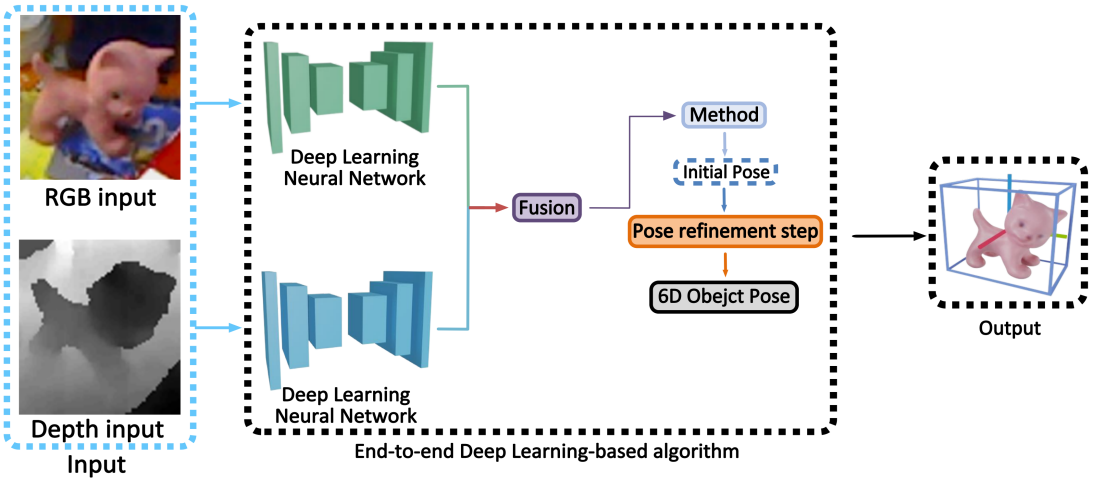



Figure 6.1: End-to-end Deep Learning-based Method with Refinement

# 7 Conclusion

In this review paper, we provide a review of the current state-of-the-art deep learning methods for instance-level 6D object pose estimation. We discussed the key challenges faced in this field, including clutter, occlusion, viewpoint variation, illumination variation, and real-time constraints. We also analyzed the neural network architectures of RGB image-based and RGB-D image-based methods and presented a quantitative and qualitative evaluation of the methods on the LineMOD dataset using the ADD(-S) metrics.

Overall, we found that the RGB-D image-based methods performed better than the RGB image-based methods, and methods incorporating a refinement step achieved state-of-the-art results. In the future, we suggest that further research should focus on developing end-to-end deep learning methods based on RGB-D images, with refinement steps, and incorporating a combination of different neural network architectures. Additionally, it would be valuable to conduct a comprehensive review of the category-level 6D object pose estimation and provide future research directions.

# List of Figures

# List of Tables

# Bibliography


[1] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," *arXiv preprint arXiv:1711.00199*, 2017.

[2] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian conference on computer vision*, Springer, 2012, pp. 548–562.

[3] C. Sahin, G. Garcia-Hernando, J. Sock, and T.-K. Kim, "Instance-and category-level 6d object pose estimation," in *RGB-D Image Analysis and Processing*, Springer, 2019, pp. 243–265.

[4] G. Du, K. Wang, S. Lian, and K. Zhao, "Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: A review," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1677–1734, 2021.

[5] A. Dhall, D. Dai, and L. Van Gool, "Real-time 3d traffic cone detection for autonomous driving," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2019, pp. 494–501.

[6] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: A hands-on survey," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 12, pp. 2633–2651, 2015.

[7] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3343–3352.

[8] H. Chen, P. Wang, F. Wang, W. Tian, L. Xiong, and H. Li, "Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2781–2790.

[9] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3d object recognition," in *2010 IEEE computer society conference on computer vision and pattern recognition*, Ieee, 2010, pp. 998–1005.

[10] H. Liang, X. Sun, Y. Sun, and Y. Gao, "Text feature extraction based on deep learning: A review," *EURASIP journal on wireless communications and networking*, vol. 2017, no. 1, pp. 1–12, 2017.

[11] Y. Xu, K.-Y. Lin, G. Zhang, X. Wang, and H. Li, "Rnnpose: Recurrent 6-dof object pose refinement with robust correspondence field estimation and pose optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 880–14 890.

[12] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6d object pose estimation using 3d object coordinates," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*, Springer, 2014, pp. 536–551.

[13] V. N. Nguyen, Y. Hu, Y. Xiao, M. Salzmann, and V. Lepetit, "Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6771–6780.

[14] Y. Zhu, M. Li, W. Yao, and C. Chen, "A review of 6d object pose estimation," in *2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, IEEE, vol. 10, 2022, pp. 1647–1655.

[15] J. Sun, Z. Wang, S. Zhang, X. He, H. Zhao, G. Zhang, and X. Zhou, "Onepose: One-shot object pose estimation without cad models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6825–6834.

[16] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Journal of big Data*, vol. 8, no. 1, pp. 1–74, 2021.

[17] J. Chen, L. Zhang, Y. Liu, and C. Xu, "Survey on 6d pose estimation of rigid object," in *2020 39th Chinese Control Conference (CCC)*, IEEE, 2020, pp. 7440–7445.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[20] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "Pvnet: Pixel-wise voting network for 6dof pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4561–4570.

[21] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[22] G. Wang, F. Manhardt, F. Tombari, and X. Ji, "Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 611–16 621.

[23] Z. Li, G. Wang, and X. Ji, "Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7678–7687.

[24] P. L. Rosin, Y.-K. Lai, L. Shao, and Y. Liu, *RGB-D image analysis and processing*. Springer, 2019.

[25] W. Hua, Z. Zhou, J. Wu, H. Huang, Y. Wang, and R. Xiong, "Rede: End-to-end object 6d pose robust estimation using differentiable outliers elimination," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2886–2893, 2021.

[26] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 632–11 641.

[27] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[28] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

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

[30] Y. Wu, M. Zand, A. Etemad, and M. Greenspan, "Vote from the center: 6 dof pose estimation in rgb-d images by radial keypoint voting," in *European Conference on Computer Vision*, Springer, 2022, pp. 335–352.

[31] S. Hoque, M. Y. Arafat, S. Xu, A. Maiti, and Y. Wei, "A comprehensive review on 3d object detection and 6d pose estimation with deep learning," *IEEE Access*, 2021.

[32] T. Hodaň, M. Sundermeyer, B. Drost, Y. Labbé, E. Brachmann, F. Michel, C. Rother, and J. Matas, "Bop challenge 2020 on 6d object localization," in *European Conference on Computer Vision*, Springer, 2020, pp. 577–594.

[33] T. Hodaň, J. Matas, and Š. Obdržálek, "On evaluation of 6d object pose estimation," in *European Conference on Computer Vision*, Springer, 2016, pp. 606–619.

[34] C. Sahin, G. Garcia-Hernando, J. Sock, and T.-K. Kim, "A review on object pose recovery: From 3d bounding box detectors to full 6d pose estimators," *Image and Vision Computing*, vol. 96, p. 103 898, 2020.

[35] F. Gorschlüter, P. Rojtberg, and T. Pöllabauer, "A survey of 6d object detection based on 3d models for industrial applications," *Journal of Imaging*, vol. 8, no. 3, p. 53, 2022.

[36] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," *Advances in neural information processing systems*, vol. 28, 2015.