Single-Lens Multi-object 3D Shape Reconstruction and 6D Attitude and Size Estimation

Abstract

The estimation of 6D attitude of objects by using vision technology is widely used in robot, human-computer interaction, logistics management and other tasks.This paper focuses on 3D reconstruction and 6D attitude and size estimation in the single view RGB-D case.In contrast to instancelevel pose estimation, we focus on a more challenging problem where CAD models are not available at inference time. Existing approaches mainly follow a complex multi-stage pipeline which first localizes and detects each object instance in the image and then regresses to either their 3D meshes or 6D poses. These approaches suffer from high-computational cost and low performance in complex multi-object scenarios, where occlusions can be present. Hence, we present a simple onestage approach to predict both the 3D shape and estimate the 6D pose and size jointly in a bounding-box free manner. In particular, our method treats object instances as spatial centers where each center denotes the complete shape of an object along with its 6D pose and size.What's more,we utilized the paper "Density-aware Chamfer Distance as a Comprehensive Metric for Point Cloud Completion" which published in NeurIPS in 2021 to implement improvement. The results show that the object recognition and modeling have been greatly improved.

Keywords: RGB-D,3D Shape reconstruction,Density-aware Chamfer Distance

1 Introduction

The six-dimensional attitude estimation of objects has always been an important research topic in the field of computer vision. The so-called six-dimensional attitude estimation of an object refers to an observation source, usually an RGB image or an RGB-D image. Using the 3D model of the target object, the translation T and rotation R of all the target objects in the observation region covered by the data are estimated.

Three-dimensional reconstruction technology is a technology that obtains the single perspective of each point cloud by using equipment such as depth camera, calculates the rigid body transformation matrix of each point cloud perspective relative to the previous point cloud perspective, completes point cloud fusion, and finally builds multi-perspective point cloud data to form a complete three-dimensional structure of an object.

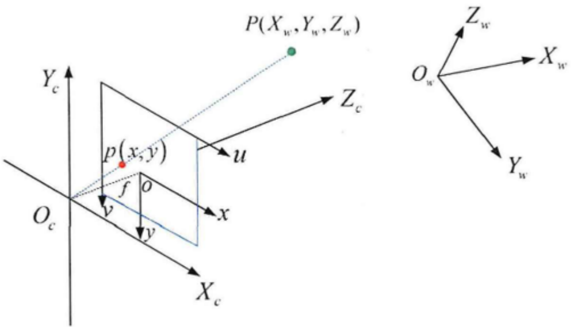


Figure 1. Coordinate mapping.Represents a mapping from the physical photo to the camera

3D reconstruction and 6D pose recognition and size estimation based on vision play an important role in many fields, such as robot operation [1], map navigation and so on.Real-time pose estimation is crucial for robots as it enables instantaneous perception of their own spatial orientation. This capability enhances robotic autonomy by providing immediate awareness of the robot's position and orientation in the environment. Accurate pose estimation facilitates precise navigation, manipulation of objects, and interaction with surroundings. Whether avoiding obstacles or executing complex tasks, real-time pose estimation empowers robots to adapt swiftly to dynamic environments. This capability is pivotal for a wide range of applications, from industrial automation to service robotics, ensuring efficient and responsive robotic performance in diverse scenarios.It helps the robot to grasp [2].Recent 6D attitude and size estimates have been performed using complex multi-stage pipelines.They are divided into two stages, one for 2D detection and the other for object detection or 6D attitude size estimation.This pipeline is computationally expensive, not scalable, and has low performance on real-world novel object instances.In this paper, we introduce Center-based Shape reconstruction and 6D pose and size estimation (CenterSnap), a single-shot approach to output complete 3D information (3D shape, 6D pose and sizes of multiple objects) in a bounding-box proposal-free and per-pixel manner.

2 Related works

2.1 Objects are represented by pixels

Utilizing pixel information for object reconstruction is a critical technique in computer vision, commonly applied in 3D reconstruction and virtual reality domains. This process involves recovering the geometric structure and surface details of a scene from two-dimensional images. Primarily, each pixel in the images captured by a camera contains information about the scene. Depth perception from these pixels can be achieved through techniques like stereo vision, which compares images from different camera angles to calculate the distance and position of objects.

In the object reconstruction process, features of individual pixels are extensively exploited. Detecting key points, edges, and textures in the image allows for more accurate localization of object boundaries and surface characteristics. Matching and tracking these feature points provide robust support for subsequent reconstruction steps. Additionally, leveraging color information from pixel points enhances the recreation of surface details. For instance, texture mapping involves projecting captured image textures onto a 3D model, resulting in a more realistic representation of the object's appearance.

Furthermore, considerations for lighting, shadows, and other factors are essential when using pixel information for object reconstruction to accurately reproduce the object's appearance. Simulating pixel changes under different lighting conditions enhances the accuracy and stability of 3D reconstruction. Overall, pixel-based object reconstruction is a complex and multi-faceted engineering task. By integrating technologies such as deep learning, computer vision, and graphics, efficient and precise 3D reconstruction of real-world objects can be achieved.This [3]represent instances as their centers in a spatial 2D grid in Figure2.Figure3 shows explore visual patterns within a single cropping area with minimal cost

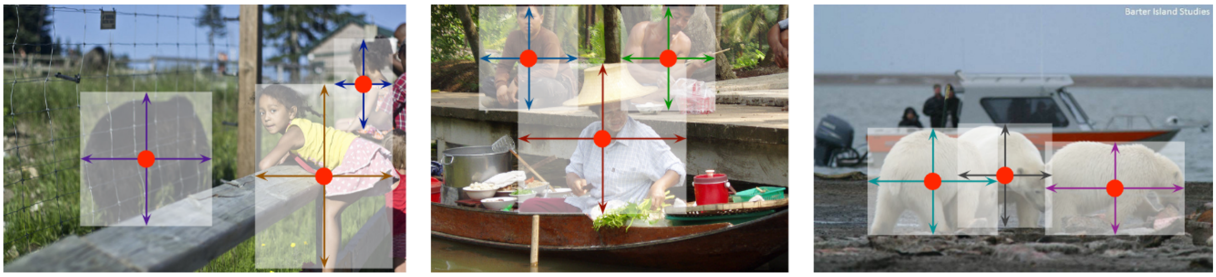


Figure 2. The bounding box size and other object properties are inferred from the keypoint feature at the center

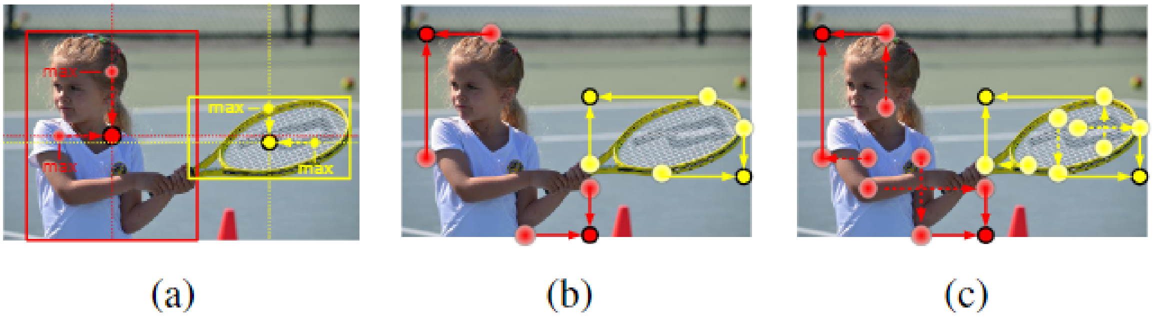


Figure 3. Keypoint Triplets for Object Detection

2.2 Instance level attitude estimation

Instance-level pose estimation is a sophisticated computer vision task aimed at simultaneously detecting multiple object instances in images or videos while estimating their poses, encompassing position, orientation, and shape. This task is pivotal in various applications, including robotics, autonomous driving, human-computer interaction, and augmented reality.

A key challenge in instance-level pose estimation lies in concurrently handling multiple object instances, each with distinct shapes, poses, and sizes. Typically, this task combines both object detection and pose estimation. Initially, through object detection, the system locates instances of objects in the image, followed by pose estimation tailored to each detected instance.

Significant strides in deep learning have greatly advanced instance-level pose estimation. Convolutional Neural Networks (CNNs) and other neural network architectures are extensively employed to learn feature representations and spatial relationships among object instances. Through end-to-end training, these models can learn complex pose variations and handling occlusions from large-scale datasets.

For instance, in human instance-level pose estimation, a system can simultaneously detect multiple individuals and estimate joint positions for each person, forming a three-dimensional pose for each joint. This technology holds significant implications in applications such as human motion capture, sports analysis, and virtual reality. [4]present a 3D object detection method that uses regressed descriptors of locally-sampled RGB-D patches for 6D vote casting.

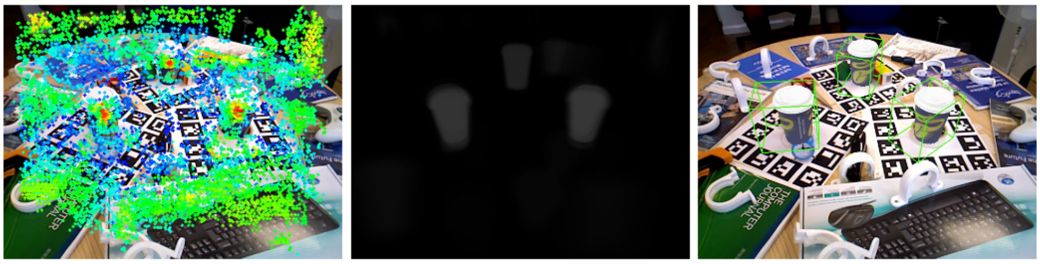


Figure 4. Instance level attitude estimation

2.3 Single view 3D reconstruction using RGB

Single-view 3D reconstruction using RGB imagery is a cutting-edge technique in computer vision that leverages a lone 2D image to infer the underlying 3D structure of a scene. This method capitalizes on the rich color information provided by Red, Green, and Blue channels in an RGB image.

The process begins with feature extraction, identifying key points and their corresponding descriptors in the 2D image. These features serve as anchor points for subsequent depth estimation. Employing deep learning architectures, such as convolutional neural networks (CNNs), the algorithm learns to predict depth maps from single images, capturing intricate spatial relationships.

Through the fusion of geometric and semantic cues, the model refines its understanding of the scene's 3D geometry. Advanced techniques, including multi-scale feature extraction and attention mechanisms, enhance the network's ability to discern depth disparities within the image.

The reconstruction process involves generating a dense point cloud from the inferred depth map. This point cloud encapsulates the 3D coordinates of the scene, offering a comprehensive representation of its structure. Texture mapping is then applied, aligning the RGB information with the corresponding 3D points to create a visually realistic 3D model.

Challenges in single-view 3D reconstruction include handling occlusions, accurately estimating depth in textureless regions, and addressing ambiguities inherent in a single 2D projection. Despite these challenges, RGB-based 3D reconstruction holds immense potential for applications in augmented reality, robotics, and virtual environments, offering a pathway towards immersive and intelligent systems that can perceive and interact with the world using minimal input information.A 3D point cloud of the complete object can be reconstructed from a single image. Each point is visualized as a small sphere [5].

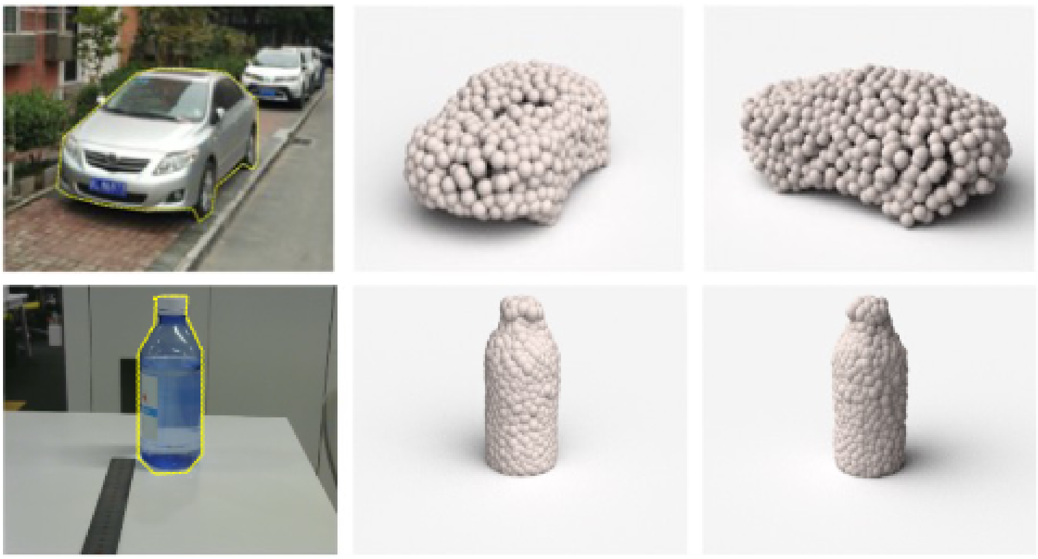


Figure 5. 3D reconstruction from a single image, generating a straight-forward form of output – point cloud coordinates

3 Method

3.1 Overview

Overall process Figure 10:

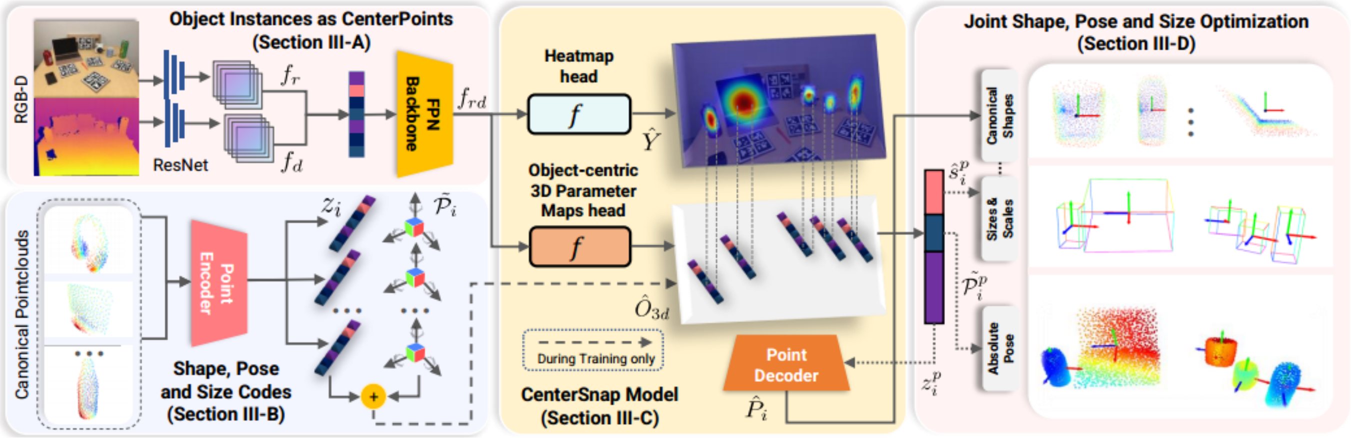
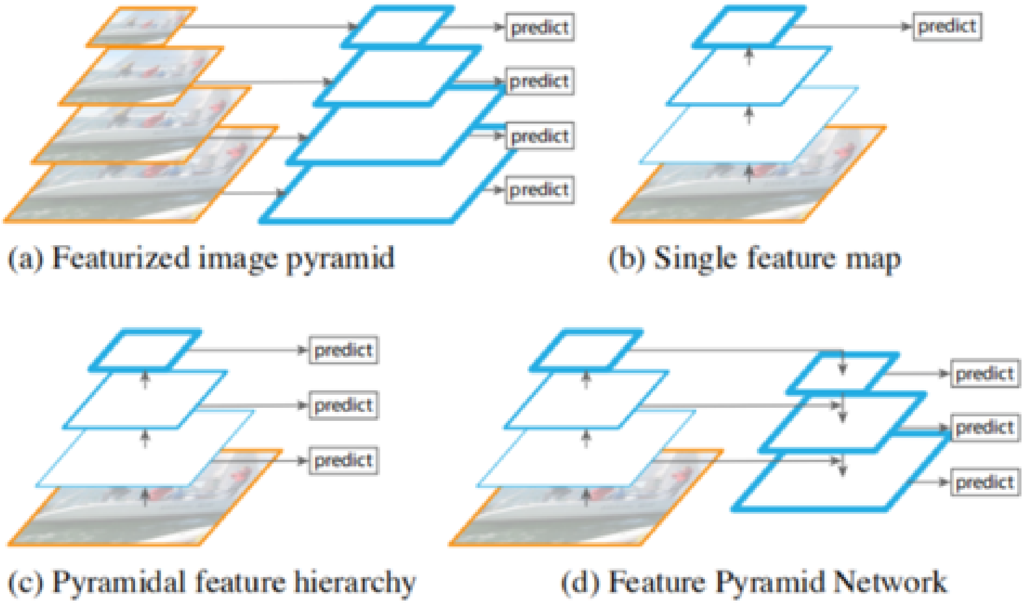


Figure 6. Given a single-view RGB-D observation, our proposed approach jointly optimizes for shape, pose, and sizes of each object in a single-shot manner

The paper method comprises a joint FPN backbone for feature extraction a pointcloud auto-encoder to extract shape codes from a large collection of CAD models , CenterSnap model which constitutes multiple specialized heads for heatmap and object-centric 3D parameter map prediction and joint optimization for shape, pose, and sizes for each object's spatial center.

3.2 Feature Pyramid Network

For general neural networks, the method shown in Figure b is used for prediction. The image is downsampled several times and the prediction is made at the last layer. The disadvantage of this method is that it has poor detection effect on small targets. Figure c. Using the information of the first few layers to make multi-scale prediction, the disadvantage of this method is that the semantic information at the lower level is not enough Figure d. A top-down path is added on the basis of Figure c, the main purpose is to solve the problems existing in the previous three ways. Through the top-down path, the low-level feature map has better semantic information.



(a) Featurized image pyramid

(b) Single feature map

(c) Pyramidal feature hierarchy

(d) Feature Pyramid Network

Figure 7. FPN

3.3 Object instances as center points

Given a depth map $I \in R^{h_o \times w_o \times 3}$,$D \in R^{h_o \times w_o\}}$.We use ResNet to generate a low-resolution representation of the spatial features$f_r \in R^{h_o/4 \times w_o/4 \times C_s}$,while the $C_s = 32$.We connect the calculated features f_r and f_d along the channel dimension before feeding them to the Resnet18-FPN backbone to compute the feature pyramid (f_{rd}) with a resolution range of 1/8 to 1/2.

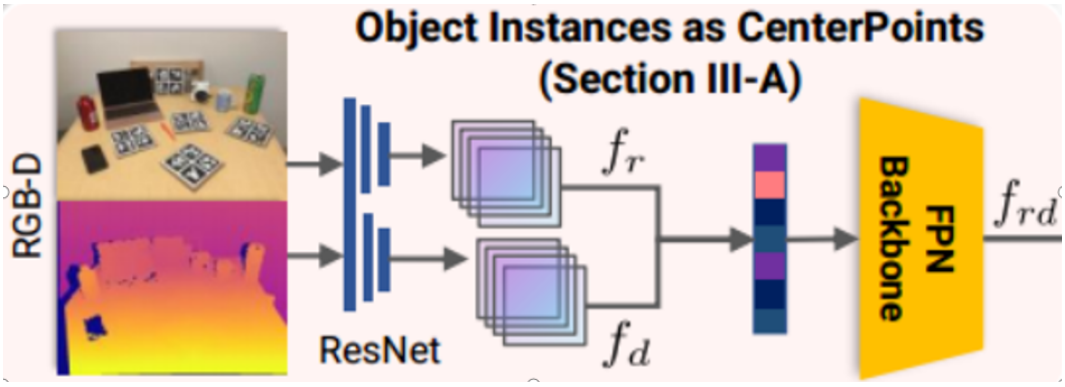


Figure 8. First-step feature

ResNet, or Residual Network, revolutionized deep learning by addressing the challenges associated with training extremely deep neural networks. Introduced by Kaiming He and his team in 2016, ResNet's key innovation lies in its use of residual blocks with skip connections. Unlike traditional architectures, ResNet learns residual functions, allowing the model to focus on the difference between the input and the desired output. The introduction of identity shortcuts facilitates the flow of gradients during backpropagation, mitigating the vanishing gradient problem associated with deep networks. ResNet architectures, ranging from ResNet-18 to ResNet-152, offer various depths, with deeper models generally exhibiting superior performance. The incorporation of global average pooling (GAP) instead of fully connected layers contributes to reduced overfitting and parameter efficiency.

3.4 Shape, Pose, and Size Codes

For joint optimization of object-based heatmaps, 3D shapes and 6D pose and sizesThe complete object-based 3D information is represented as an object-centered 3D parameter graph.The point cloud representation of each object is stored in an object-centered 3D parameter map in the form of an underlying shape encoding.To train the autoencoder, we sample 2048 points from the ShapeNet [6]CAD model library and use them as ground-truth shapes.

To learn the shape code (z_i) of each object, we design an auto-encoder.In addition, we normalized the input point cloud by applying scaling transform to each shape so that the shape is centered on the origin and normalized by units. We use reconstruction errors represented by Chamfer distance D_{cd}(Chamfer distance) to jointly optimize the encoder and decoder networks.The formula is as follows:

$$D_{cd}(\mathbf{P}_i, \hat{\mathbf{P}}_i) = \frac{1}{|\mathbf{P}_i|} \sum_{x \in \mathbf{P}_i} \min_{y \in \hat{\mathbf{P}}_i} \|x - y\|_2^2 + \frac{1}{|\hat{\mathbf{P}}_i|} \sum_{\mathbf{y} \in \hat{\mathbf{P}}_i} \min_{x \in \mathbf{P}_i} \|x - y\|_2^2$$

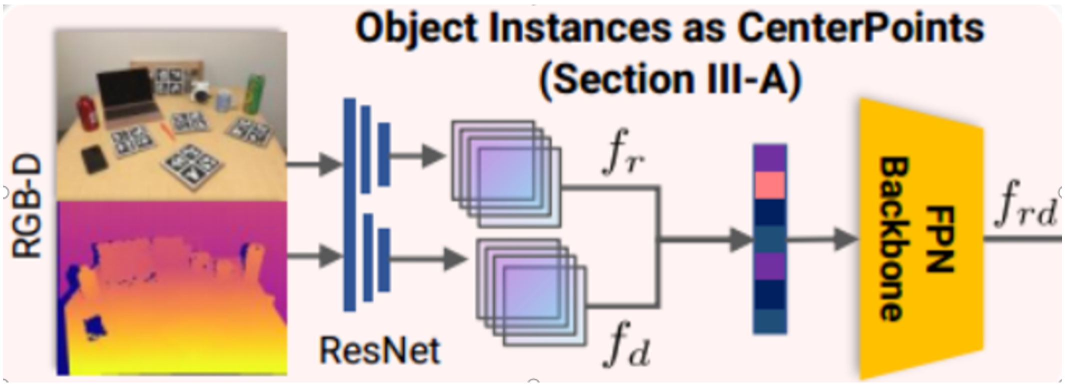


Figure 9. First-step feature

3.5 Improvement

In 2021, there is an article on NeurIP for improvements to D_{cd} , Density-aware Chamfer Distance as a Comprehensive Metric for Point Cloud Completion(d_{DCD}). In this paper, Density-aware Chamfer Distance is used for improvement. Chamfer Distance (CD) and Earth Mover's Distance (EMD) are two widely used metrics to measure the similarity between two point sets. But CD is usually insensitive to the local density of mismatches, and EMD is usually dominated by the global distribution, thus ignoring the authenticity of the detailed structure. In addition, their unbounded range of values causes serious outliers. These deficiencies prevent them from providing a consistent assessment.

$$d_{DCD}(S_1, S_2) = \frac{1}{2}\left(\frac{1}{|S_1|}\sum_{x\in S_1}\left(1 - \frac{1}{n_{\hat{y}}}e^{-\alpha||x-\hat{y}||_2}\right) + \frac{1}{|S_2|}\sum_{y\in S_2}\left(1 - \frac{1}{n_{\hat{x}}}e^{-\alpha||y-\hat{x}||_2}\right)\right),$$

Figure 10. First-step feature

3.6 Joint Shape, Pose, and Size Optimization

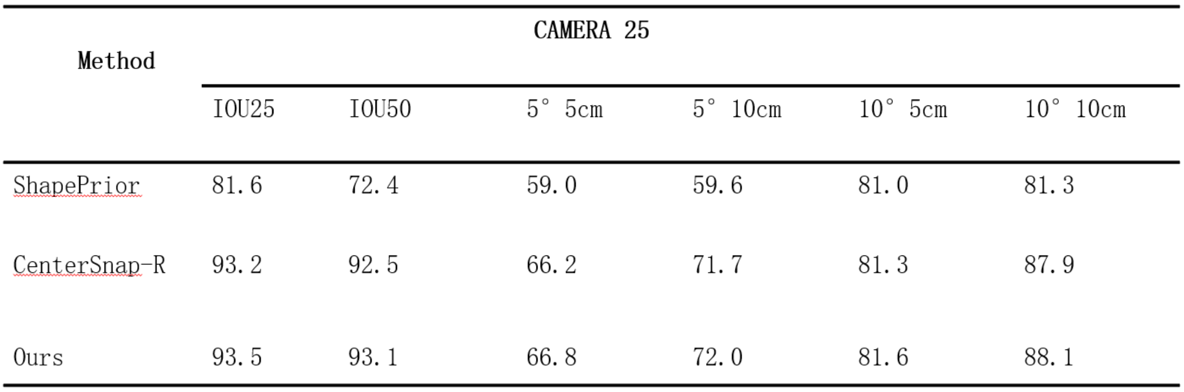
We minimized the combination of heatmap instance instance detection, object-centric 3Dmap prediction, and auxiliary depth losses.

$$L = \lambda_l + L_{inst} + \lambda_{O_{3d}} + \lambda_d L_d$$

During inference, peaks in the heatmap output execution are detected to obtain the detection center points (C_i) for each object, which are the local maximum values in the heatmap output.

4 Results

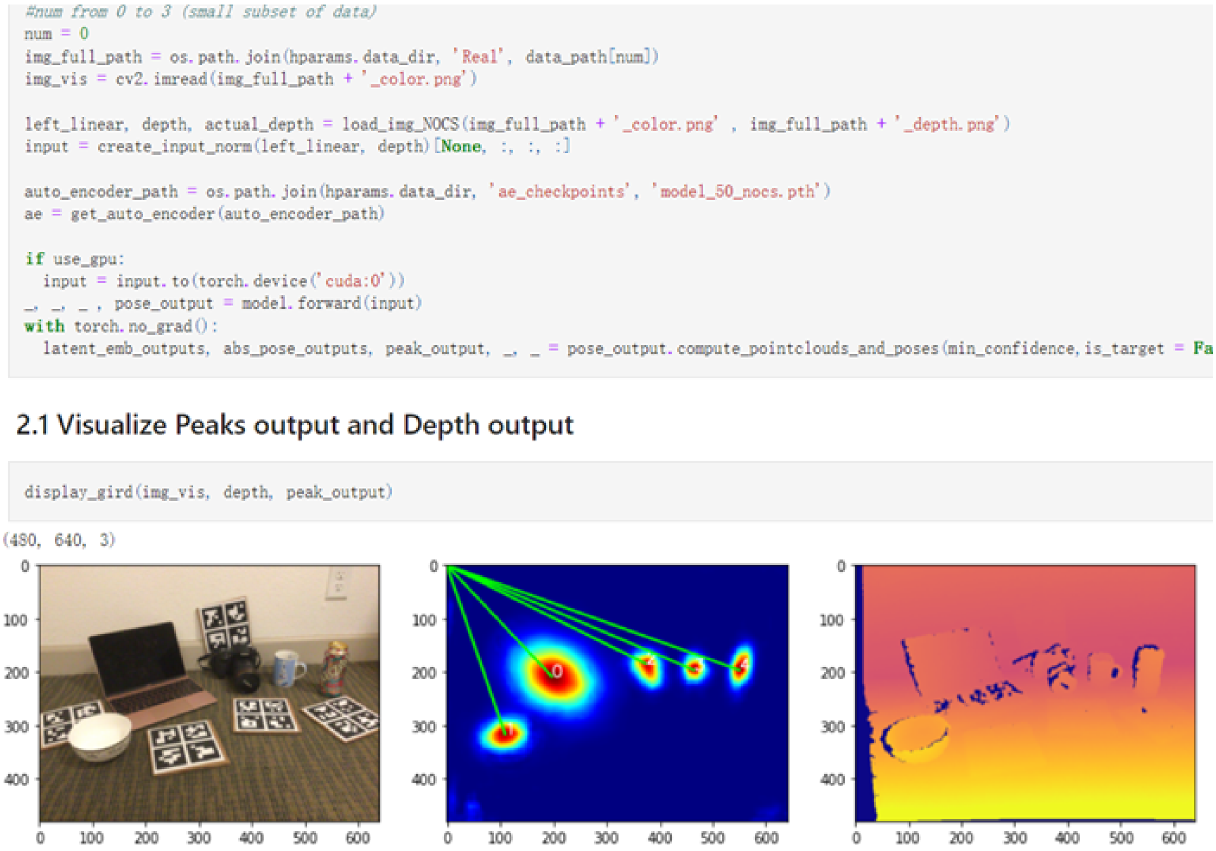
Following , we independently evaluate the performance of 3D object detection and 6D pose estimation using the following key metrics: 1) Averageprecision for various IOU-overlap thresholds (IOU25 and IOU50). 2) Average precision of object instances for which the error is less than n ○ for rotation and m cm for translation (5°5 cm, 5°10 cm and 10°10 cm). For shape reconstruction we use Chamfer distance (CD) following.We can see that the results have improved very well:



Method	CAMERA 25					
	IOU25	IOU50	5° 5cm	5° 10cm	10° 5cm	10° 10cm
ShapePrior	81.6	72.4	59.0	59.6	81.0	81.3
CenterSnap-R	93.2	92.5	66.2	71.7	81.3	87.9
Ours	93.5	93.1	66.8	72.0	81.6	88.1

Figure 11. Comparison of results

Below is a demonstration of my code implementation process



```
#num from 0 to 3 (small subset of data)
num = 0
img_full_path = os.path.join(hparams.data_dir, 'Real', data_path[num])
img_vis = cv2.imread(img_full_path + '_color.png')

left_linear, depth, actual_depth = load_img_NOCS(img_full_path + '_color.png' , img_full_path + '_depth.png')
input = create_input_norm(left_linear, depth)[None, :, :, :]

auto_encoder_path = os.path.join(hparams.data_dir, 'ae_checkpoints', 'model_50_nocs.pth')
ae = get_auto_encoder(auto_encoder_path)

if use_gpu:
  input = input.to(torch.device('cuda:0'))
_, _, _ , pose_output = model.forward(input)
with torch.no_grad():
  latent_emb_outputs, abs_pose_outputs, peak_output, _, _ = pose_output.compute_pointclouds_and_poses(min_confidence,is_target = Fa
```

2.1 Visualize Peaks output and Depth output

```
display_gird(img_vis, depth, peak_output)
```

```
(480, 640, 3)
```

Figure 12. Implementation code

References

[1] M. Svetlik K. Fang Z. Jiang, Y. Zhu and Y. Zhu. Synergies between affordance and geometry: 6-dof grasp detection via implicit representations. *Robotics: science and systems*, 2021.

[2] C. Ferrari and J. F. Canny. Planning optimal grasps. *Planning Optimal Grasps*, 3(4):6, 1992.

[3] D. Wang X. Zhou and P. Krahenbuhl. Objects as points. *Computer Vision and Pattern Recognition*, 2019.

[4] F. Tombari S. Ilic W. Kehl, F. Milletari and N. Navab. Deep learning of local rgb-d patches for 3d object detection and 6d pose estimation. *European conference on computer vision*, pages 205–220, 2016.

[5] H. Su H. Fan and L. J. Guibas. A point set generation network for 3d object reconstruction from a single image. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017.

[6] L. Guibas P. Hanrahan Q. Huang Z. Li S. Savarese M. Savva S. Song H. Su et al A. X. Chang, T. Funkhouser. Shapenet: An information-rich 3d model repository. *Planning Optimal Grasps*, arXiv preprint arXiv:1512.03012, 2015.