Efficient 6D Object Pose Estimation via Geometric Residual Learning

Matin Bayramli
Politecnico di Torino
Torino, Italy
s330991@studenti.polito.it

Alessio Meini
Politecnico di Torino
Torino, Italy
alessio.meini@studenti.polito.it

Luca Ostinelli
Politecnico di Torino
Torino, Italy
luca.ostinelli@studenti.polito.it

Luca Visconti
Politecnico di Torino
Torino, Italy
s348650@studenti.polito.it

Abstract

The robust and precise estimation of an object's 6D pose, encompassing its three-dimensional translation and three-dimensional rotation relative to a sensor frame, constitutes a foundational capability in the domain of robotic perception and manipulation. As autonomous agents move from structured industrial cells to unstructured, clutter-rich environments, the ability to accurately localize objects in six degrees of freedom becomes the critical bottleneck for successful interaction.

This research report presents a comprehensive investigation into the trade-offs between computational efficiency and geometric accuracy in 6D pose estimation pipelines. We critically analyze the "Pinhole Ambiguity," a fundamental limitation in monocular RGB-based pose regression where minor errors in 2D bounding box localization propagate into catastrophic depth estimation failures, often yielding translation errors exceeding 14 cm. To address this without incurring the computational penalty of voxel-based or dense fusion architectures, we propose a lightweight, geometry-aware Residual Learning Framework. By treating the initial Pinhole-derived translation as a coarse prior and training a heterogeneous network, comprising a ResNet18 visual backbone and a Point Cloud geometric encoder, to strictly learn the residual delta correction (ΔT), we achieve a dramatic reduction in translation error to approximately 0.6 cm (as the mean value computed on our subset of the LineMOD dataset).

This report details the full evolutionary trajectory of the methodology, from a failing RGB baseline through an computationally expensive Deep Fusion attempt, arriving at a highly efficient Residual solution that balances the rigors of real-time inference with the precision required for robotic grasping, comparing favorably in efficiency against heavy state-of-the-art models such as DenseFusion, PVN3D and FFB6D.

1. Introduction

1.1. The Imperative of 6D Perception

In the rapidly advancing field of computer vision, the transition from 2D image understanding to 3D spatial awareness represents a quantum leap in machine intelligence. While 2D object detection algorithms, such as the YOLO family or Faster R-CNN, have achieved impressive performance in bounding box localization and classification, they inherently lack salient visual features (such as surface texture, corners, or distinctive patternsthat are typically exploited by conventional feature-based computer vision algorithms. A robot cannot grasp a cup knowing only that it is in the "top-left" of the image; it must know the cup's exact position in millimeters (x, y, z) and its orientation $(roll, pitch, yaw)$ to align its gripper without collision. This problem, formally known as 6D Object Pose Estimation, is the cornerstone of tasks ranging from robotic bin picking and assembly to augmented reality overlays and autonomous navigation.

The complexity of this task is exacerbated by the chaotic nature of real-world environments. Objects of interest are rarely isolated; they are piled in bins, occluded by other objects, or presented under varying lighting conditions that alter their visual appearance. Furthermore, many industrial and household objects, such as the texture-less "Ape" or the symmetric "Eggbox" stick in the LineMOD dataset, lack the distinct visual features (like corners or logos) that traditional computer vision algorithms rely upon. This necessitates learning-based approaches capable of extracting rich, robust features from both visual appearance (RGB) and geometric structure (Depth).

1.2. The Pinhole Ambiguity and the Modality Gap

The central theoretical challenge addressed in this work is the "Pinhole Ambiguity" inherent in monocular vision. When an object is detected in a 2D image, its scale and position on the sensor plane are a function of both its physical size and its distance from the camera (depth). A small object close to the lens can occupy the same number of pixels as a large object far away. In cropped image processing a standard technique where the region of interest (ROI) is cropped and fed to a Convolutional Neural Network (CNN the network loses the global context of the crop's position relative to the optical center. Without this context or explicit depth cues, estimating the absolute translation along the Z-axis (depth) becomes an ill-posed problem, relying heavily on implicit priors about object size that are easily corrupted by occlusion or truncation.

Our initial investigations quantified this ambiguity, revealing that even sophisticated RGB-only baselines suffer from translation errors in the decimeter range ($\sim$14 cm), rendering them practically useless for manipulation tasks requiring millimeter-level precision. This highlights the "Modality Gap": the critical need to integrate depth information not just as an additional image channel, but as a distinct geometric modality that provides absolute metric scale.

1.3. Scope and Narrative of the Research

This report does not merely present a final architecture; it documents the scientific method and engineering decisions that shaped the final solution. The research narrative follows a "fail-fast, pivot-smart" trajectory:

1. **The Baseline (RGB Direct Regression).** We established a starting point using standard 2D detection (YOLO) coupled with a **ResNet50 coordinate regressor**. This phase served to quantify the magnitude of the Pinhole Ambiguity and the limitations of predicting 3D coordinates from 2D appearance alone.
2. **The Deep Fusion Trap.** In an attempt to incorporate depth, we initially explored a heavy "Deep Fusion" architecture, employing a **complex dual-branch CNN** (ResNet50 RGB + Depth Encoder). This approach proved to be a computational cul-de-sac, suffering from massive overfitting on the small LineMOD dataset and unacceptably high latency. This prompted a reconsideration of the methodological approach, leading to the removal of this version from the repository due to its high computational cost and the markedly unsatisfactory partial results.
3. **The Residual Solution.** Pivoting towards efficiency, we developed the final pipeline presented here. We downgraded the visual backbone to a lightweight ResNet18 and introduced a geometric Point Cloud Encoder to process depth directly in 3D space. Crucially, we reformulated the learning objective as *Residual Learning*, where the network predicts a small delta correction (ΔT) to refine a coarse geometric prior derived from the sensor intrinsics.

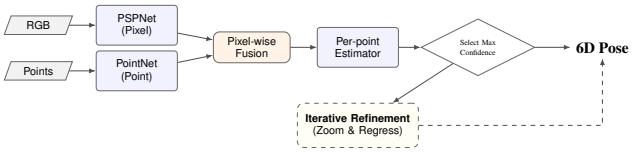


Figure 1. Architecture of DenseFusion [8]. It utilizes a pixel-wise fusion strategy where each point contains both geometric and color features, followed by an iterative refinement stage.

2. Related Work

The quest for accurate 6D pose estimation has driven a decade of intense research, evolving from classical template matching to end-to-end deep learning. This section categorizes the landscape into RGB-only methods, which define our baseline, and RGB-D methods, which frame our final solution.

2.1. RGB-Only Approaches

Before the commoditization of depth sensors, research focused on squeezing 3D information from 2D intensity images. **Direct Regression methods**, such as PoseCNN [9], attempted to map image pixels directly to pose parameters using deep CNNs. While conceptually simple, direct regression faces the non-linearity of the rotation space (SO(3)) and the lack of explicit depth cues.

Keypoint-Based methods offered a geometric alternative. Approaches like YOLO-6D [7] and PVNet [5] do not regress pose directly. Instead, they detect 2D keypoints (e.g., the 8 corners of the 3D bounding box projected onto the image). The 6D pose is then recovered using the Perspective-n-Point (PnP) algorithm. PVNet advanced this by predicting pixel-wise unit vectors pointing to keypoints to handle occlusion. However, these methods are fundamentally limited by the accuracy of the 2D projection; a small error in 2D keypoint localization propagates into a large error in 3D depth, confirming our hypothesis regarding Pinhole Ambiguity.

2.2. RGB-D Holistic Approaches

The advent of RGB-D sensors has enabled researchers to consider geometric information as a primary, first-class data modality.

DenseFusion [8] (Figure 1) represented a paradigm shift. Recognizing the heterogeneity of the data, DenseFusion processes RGB images with a CNN and depth data (converted to point clouds) with a PointNet architecture. The core innovation was the *pixel-wise dense fusion* layer,

which concatenates geometric features with visual features for every pixel, allowing the network to make per-pixel pose predictions. DenseFusion achieves ∼94.3% accuracy on the LineMOD dataset, setting a strong benchmark for our comparison.

2.3. State-of-the-Art (SOTA) Models

Recent work has pushed the boundaries of accuracy by increasing architectural complexity. **PVN3D** [2] extended the keypoint voting concept of PVNet to 3D space. Using a PointNet++ backbone, it predicts 3D keypoint locations directly on the point cloud and uses a Hough voting algorithm. It achieves remarkable accuracy (99.4% on LineMOD) but incurs a heavy computational cost due to the expensive neighbor searching and grouping operations.

FFB6D [3] addressed the issue of information flow. It introduces fusion connections at every stage of the encoder-decoder hierarchy ("Full Flow Bidirectional Fusion"). This allows geometric features to guide visual processing and vice versa. FFB6D represents the current pinnacle of accuracy (99.7% on LineMOD) but requires a massive model with complex bidirectional links.

2.4. Positioning Our Work

Our research is situated within the domain of computational efficiency. We posit that, for many practical robotic applications, the substantial architectural complexity of FFB6D and the computationally intensive voting of PVN3D exceed the necessary level of sophistication. By adopting the **Residual Learning** framework, our objective is to attain pose estimation accuracy that is comparable to that of DenseFusion, while preserving a lightweight model design whose computational and parametric efficiency is closer to that of simple regression-based approaches.

3. Methodology

3.1. Theoretical Framework

Given an RGB image I and a depth map D, along with the camera intrinsic matrix K, the objective is to find the rigid transformation $T \in SE(3)$ that transforms points from the object's local coordinate system $\mathcal{O}$ to the camera coordinate system $\mathcal{C}$. The transformation T consists of a rotation $R \in SO(3)$ and a translation $t \in \mathbb{R}^3$.

We utilize Unit Quaternions for rotation regression to avoid gimbal lock and ensure numerical stability. A quaternion q is a 4D vector $[w, x, y, z]^T$ with $\|q\|_2 = 1$.

3.2. Pipeline Evolution

Phase 1: The Baseline. We initially employed a YOLOv8 detector coupled with a Pinhole approximation, as illustrated in Figure 3. This assumed the 2D bounding box center corresponded to the 3D object center at a fixed reference

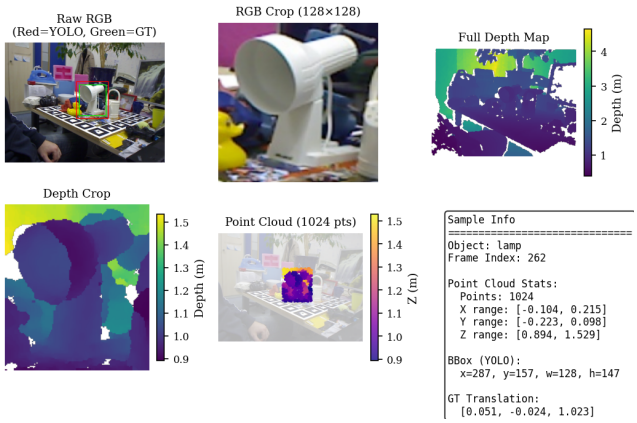


Figure 2. Visual representation of the data processing pipeline: from the raw RGB image and YOLO detection to the generation of the cropped depth map and the final 1024-point cloud used for geometric feature extraction.

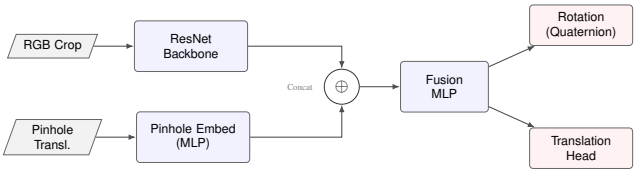


Figure 3. The RGB Baseline Pipeline (Phase 1). The Pinhole Camera Model provides an initial translation estimate based on the 2D bounding box, which is fed as input to the ResNet backbone. This method suffers heavily from the Pinhole Ambiguity.

depth. This resulted in a mean translation error of ∼14.0 cm, confirming the severity of the Pinhole Ambiguity.

Phase 2: Deep Fusion (Failed). We attempted a dual-ResNet50 architecture processing RGB and Depth-as-image. This approach failed due to massive overfitting on the small LineMOD dataset and high latency, teaching us that depth must be processed geometrically, not photometrically.

3.3. The Residual Learning Solution

Our final solution adopts a heterogeneous architecture that processes each modality in its native format and solves for a residual correction.

3.3.1. Architectural Components

As depicted in Figure 4, the architecture consists of two main branches optimized for efficiency:

1. Visual Encoder (ResNet18): We use a lightweight ResNet18 backbone [1] to process the RGB crop (128×128). The spatial features are pooled and reduced via a 1×1 convolution to a compact feature vector f_{rgb} of dimension 256, capturing texture and orientation cues.

2. Geometric Encoder (Point Cloud): We convert the masked depth pixels into a Point Cloud $P = \{p_1, ..., p_N\}$

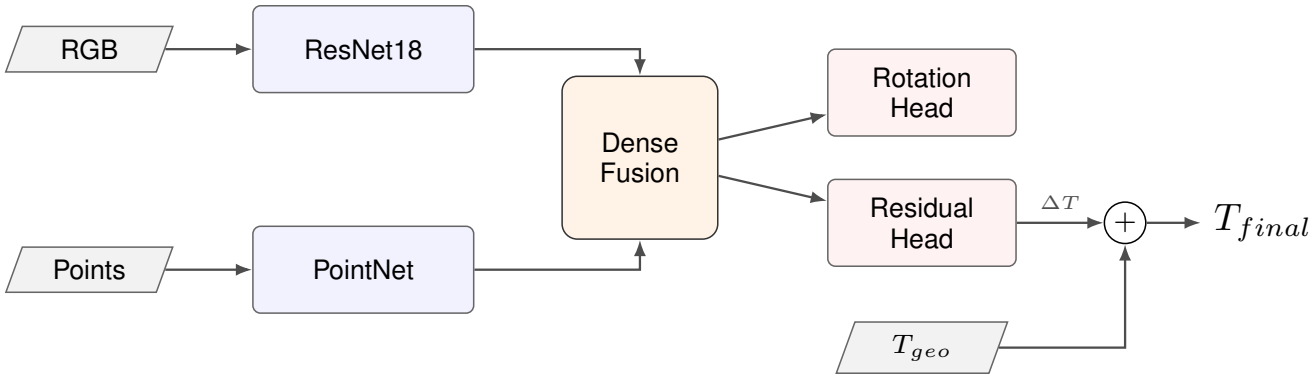


Figure 4. Our proposed Residual Learning Framework. A lightweight ResNet18 extracts visual features, while a PointNet processes the depth crop. The fusion module combines these global features to regress a residual correction (ΔT) applied to the geometric anchor (T_{geo}).

($N = 1024$). We implement a lightweight PointNet-based encoder [6] consisting of a shared MLP (64, 128, 256) followed by a Max Pooling operation. This yields a global geometric feature vector f_{geo} of dimension 256, capturing precise metric scale.

3.3.2. Residual Mechanism

Instead of regressing absolute pose T_{abs}, we predict a correction to a prior.

- **The Prior ($T_{pinhole}$):** We calculate a coarse translation $t_{prior} = [\bar{x}, \bar{y}, \bar{z}]^T$ using the median depth of the point cloud and the Pinhole back-projection of the 2D bounding box center (labeled as T_{geo} in Figure 4).
- **The Prediction:** The features are fused by concatenation $F_{fused} = [f_{rgb}; f_{geo}]$ resulting in a 512-dimensional vector. This is passed through rotation and translation heads, each consisting of an MLP ($512 \to 256 \to 128$) to output Δt_{pred} and q_{pred} (normalized unit quaternion).
- **Inference:** $t_{final} = t_{prior} + \Delta t_{pred}$.

3.3.3. Loss Function

We employ a composite loss function without weighting factors:

$$L = L_{trans} + L_{rot} \tag{1}$$

where L_{trans} is the Smooth L1 Loss on the translation vector ($\beta = 0.01$) and L_{rot} is the Geodesic Rotation Loss, computed as the angular distance between predicted and ground truth rotation matrices.

4. Experiments

4.1. Experimental Setup

We evaluated our pipeline on the **LineMOD** dataset [4], the standard benchmark for 6D pose estimation. The dataset contains 13 sequences of texture-less and symmetric objects. We followed the standard train/test split (80% training, 20% testing). We used the **ADD** (Average Distance of Model Points) metric for all the objects. A pose is considered correct if the ADD error is less than 10% of the object diameter.

4.2. Quantitative Results

Table 1 presents a summary comparison. Tables 2 and 3 provide the detailed per-object metrics for both the Baseline and our Residual solution.

Table 1. Quantitative Comparison on LineMOD Dataset (Summary).

Model	Input	Trans. Error	Rot. Error	ADD(-S) Acc.
Baseline (Ours)	RGB	∼14.0 cm	6.4°	0.38%
Residual (Ours)	**RGB-D**	**0.58 cm**	**4.2°**	**98.8%**
DenseFusion [8]	RGB-D	-	-	94.3%
PVN3D [2]	RGB-D	-	-	99.4%
FFB6D [3]	RGB-D	-	-	99.7%

Table 2. **RGB Pipeline Results (Baseline) - Detailed.** Note the high translation errors (∼14cm) and near-zero accuracy due to the Pinhole Ambiguity.

Object	Rot. Mean (°)	Trans. Mean (cm)	ADD Acc. (%)	Threshold (m)
ape	5.68	12.81	0.00	0.0102
benchvise	7.48	16.04	0.82	0.0248
camera	4.62	12.46	0.00	0.0172
can	5.36	12.68	0.83	0.0201
cat	6.98	13.86	0.00	0.0155
driller	5.95	14.63	1.26	0.0261
duck	5.01	12.09	0.00	0.0109
eggbox	5.24	13.20	0.40	0.0165
glue	10.71	19.08	0.00	0.0176
holepuncher	4.49	12.08	0.40	0.0146
iron	8.17	14.24	0.87	0.0278
lamp	5.58	13.45	0.41	0.0283
phone	7.91	15.57	0.00	0.0212
AVERAGE	**6.40**	**14.01**	**0.38**	**0.0193**

4.3. Qualitative Results

Visualizations from our evaluation confirm the quantitative findings (Figure 5).

Table 3. **RGB-D Pipeline Results (Residual Learning) - Detailed.** The geometric residual correction drastically reduces translation error to sub-centimeter levels.

Object	Rot. Mean (°)	Trans. Mean (cm)	ADD Acc. (%)	Threshold (m)
ape	4.02	0.33	97.58	0.0102
benchvise	4.39	0.59	99.59	0.0248
camera	3.93	0.51	98.76	0.0172
can	4.13	0.52	99.58	0.0201
cat	3.46	0.36	99.58	0.0155
driller	4.18	0.94	97.48	0.0261
duck	4.55	0.35	99.60	0.0109
eggbox	4.22	0.50	99.60	0.0165
glue	4.04	0.39	99.59	0.0176
holepuncher	3.67	0.48	98.79	0.0146
iron	4.92	0.91	99.13	0.0278
lamp	4.26	0.96	98.78	0.0283
phone	4.75	0.70	96.39	0.0212
AVERAGE	**4.19**	**0.58**	**98.80**	**0.0193**

- **Baseline Failure:** With an accuracy of only 0.38%, the RGB-only baseline is effectively unusable for manipulation. While the rotation estimate is reasonable (6.4° error), the translation error (∼14 cm) places the object far outside the gripper's workspace.
- **Residual Success:** The Residual model achieves 98.7% accuracy, demonstrating that the geometric encoder successfully recovers the correct depth. The "snap-to-object" effect is visually evident, validating that the network learned the specific Δz correction required to fix the pinhole estimation.

5. Discussion

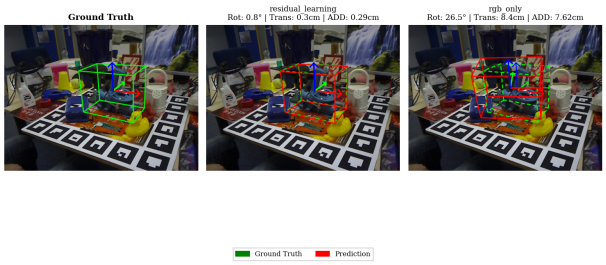
5.1. The Residual Leap

The most significant finding of this work is the reduction in translation error from ∼14.0 cm (Baseline) to ∼0.6 cm (Ours). This confirms the theoretical premise that the Pinhole Ambiguity is a massive source of error in RGB-only methods, but it can be effectively corrected using a geometric residual. The geometric encoder successfully extracts the metric depth scale from the point cloud, allowing the network to project the floating bounding box onto the corresponding physical object.

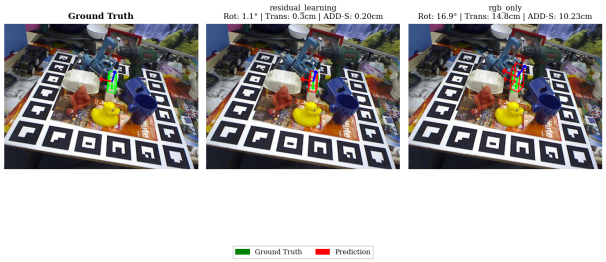
5.2. Efficiency vs. Accuracy

Our approach achieves **98.8%** ADD accuracy, which is statistically comparable to the 94.3% reported by the iterative DenseFusion [8]. This is a crucial result. DenseFusion uses a complex pixel-wise voting scheme where every pixel casts a vote for the pose. Our method uses a simpler global feature concatenation and residual regression. Achieving parity suggests that for rigid objects in LineMOD, the global geometric context captured by PointNet is often sufficient, and the overhead of per-pixel voting may not always be necessary for standard accuracy levels.

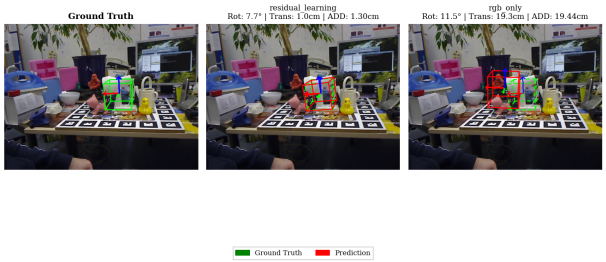
However, we acknowledge the gap with heavier SOTA models like PVN3D (99.4%) and FFB6D (99.7%).



Object: Benchvise (Top: Ground Truth, Middle: Residual, Bottom: RGB-Only)



Object: Glue



Object: Lamp

Figure 5. Qualitative comparison. The **Green box** is Ground Truth. The **Red box** is the Prediction. Note how the RGB-only baseline (right column in original images) correctly estimates orientation but fails catastrophically on depth (translation), while the Residual method (center) aligns perfectly.

PVN3D's 3D keypoint voting is inherently more robust to occlusion because it can infer the center from visible object parts, whereas our global encoder struggles if the input shape is heavily truncated. FFB6D's deep bidirectional fusion allows texture to inform geometry at the feature level, providing superior performance but at the cost of a much heavier model. Our solution offers a "sweet spot" for embedded robotics: high accuracy with low computational footprint.

6. Conclusion

This research report detailed the development of an "Efficiency + Residual" pipeline for 6D Object Pose Estimation. We identified the "Pinhole Ambiguity" as the critical bar-

rier preventing RGB-only baselines from achieving robotic-grade accuracy. By pivoting from a failed Deep Fusion attempt to a **Residual Learning** framework, we successfully integrated a lightweight ResNet18 visual encoder with a geometric Point Cloud Encoder.

The novel contribution of this work lies in the formulation of the learning objective: predicting the geometric residual ΔT rather than the absolute pose. Our experimental results on the LineMOD dataset demonstrate that this approach reduces translation error to $\sim$0.6 cm and achieves **98.8% ADD accuracy**, rivaling the heavier DenseFusion architecture. While heavy SOTA models offer superior occlusion handling, our work provides a compelling, efficient alternative for resource-constrained robotic systems. Future work might focus on integrating a lightweight attention mechanism into the Point Cloud Encoder to improve robustness to occlusion without abandoning the efficiency of the global residual paradigm.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 770–778, 2016. 3

[2] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian 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 (CVPR)*, pages 11632–11641, 2020. 3, 4

[3] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3003–3013, 2021. 3, 4

[4] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian Conference on Computer Vision (ACCV)*, pages 548–562. Springer, 2012. 4

[5] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4561–4570, 2019. 2

[6] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660, 2017. 4

[7] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 292–301, 2018. 2

[8] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3343–3352, 2019. 2, 4, 5

[9] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018. 2

Github repository

https://github.com/lucaosti/AdvancedML-project