"Enhancing 6D Object Pose Estimation"

TA: Stephany Ortuno Chanelo (stephany.ortuno@polito.it)
Paolo Rabino (paolo.rabino@polito.it)

Reference Paper: DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion

TASK OVERVIEW

This project aims to explore and implement advanced techniques for 6D pose estimation using RGB-D images. The goal is to build an end-to-end pipeline for estimating the 6D pose of objects by initially replicating a model that uses just RGB images. The pipeline will then be enhanced by incorporating depth information to improve accuracy in the pose predictions. You will adapt and implement the methodology, starting from pose prediction and then extend the model with your own innovative improvements.

Literature:
Before starting the project, it is essential to review the existing literature on 6D pose estimation and object detection to gain a solid understanding of the concepts, challenges, and state-of-the-art techniques in the field.

6D Pose Estimation

6D pose estimation involves determining the position and orientation of an object in 3D space, typically represented as a combination of:

- **3D Translation Vector (x, y, z):** Specifies the object's position in a 3D coordinate system.
- **3×3 Rotation Matrix (R):** Defines the orientation of the object.

This task is crucial in various applications such as robotics, augmented reality (AR), and autonomous systems, where accurately determining the pose of the objects allows precise interaction and manipulation.

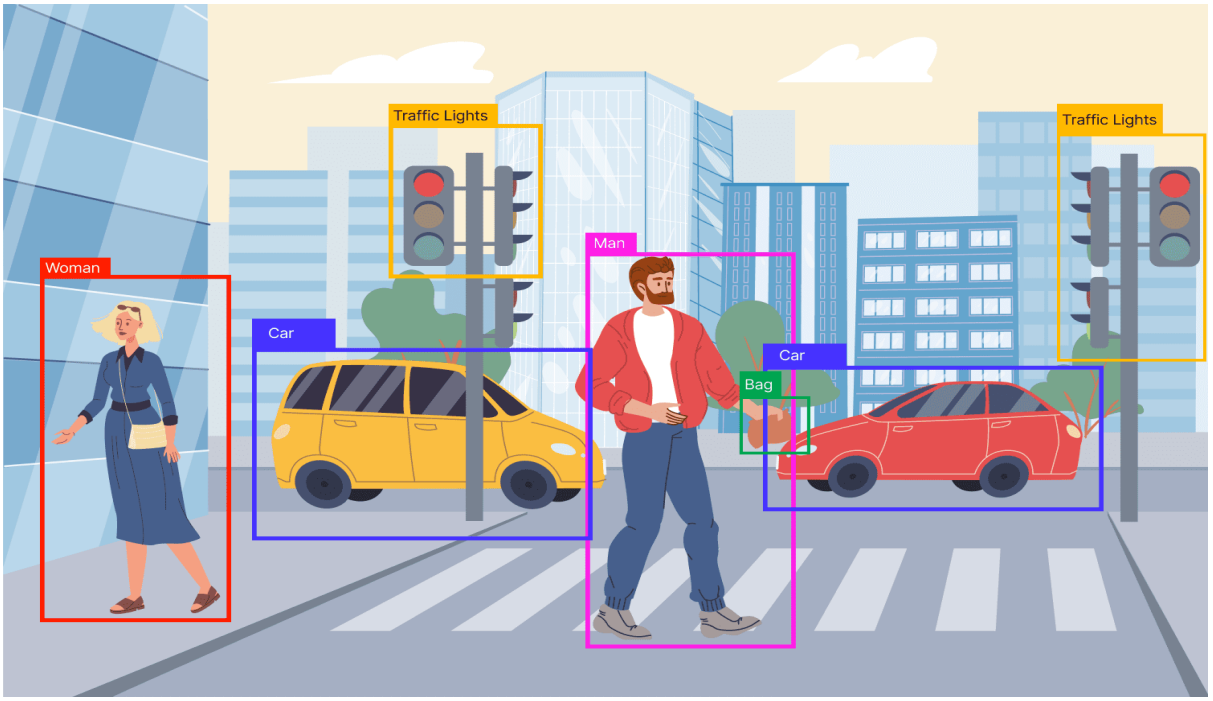
Object Detection

Object detection refers to the task of identifying and locating objects within an image or video by drawing bounding boxes around them. Unlike image classification, which only labels objects, detection involves both classification and localization. It identifies **where** an object is within the image and to **which** category it belongs.

Some of the widely used object detection models include:

- Faster R-CNN
- YOLO (You Only Look Once): https://github.com/ultralytics/ultralytics

These models are widely used in applications like autonomous driving, facial recognition, and robotic perception.



PROJECT GOALS

The project is structured into multiple phases, each designed to incrementally build the 6D pose estimation pipeline: Incremental development allows the team to systematically address challenges and refine solutions at each phase before proceeding to the next one.

Phase 1. Understanding concepts

In the first phase, the project will focus on understanding key concepts related to **6D pose estimation and object detection**. This will involve studying state-of-the-art techniques, and identifying relevant tools for the implementation. This step is crucial as it provides insights into the current advancements and challenges in the field, helping to identify gaps that the project can address. Additionally, this review establishes solid basis, guiding the selection of appropriate methodologies and performance metrics that you will need to implement it during the project. You can start by looking at the reference section for some literature.

Phase 2. Data Exploration and Object Detection

The LineMod dataset provides a valuable benchmark for advancing the field of 6D object pose estimation, especially in scenarios where texture information is limited. For this project we are using a subset of this dataset that you could find in this link: https://drive.google.com/drive/folders/19ivHpaKm9dOrr12fzC8IDFczWRPFxho7

At this point, first we suggest getting acquaintance with the LineMod Dataset and its components.

- RGB and depth images
- Bounding Box
- 6D Pose

- Object Mask
- 3D Models

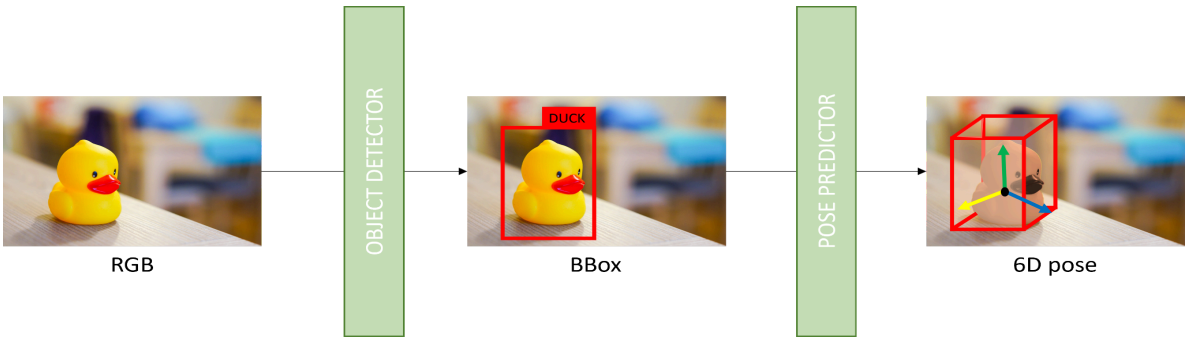
Explore the dataset and recognize the representation of each information needed to the project, for example identify the format in which the bounding box or the 6D pose is represented and determine if they will need some transformation or a pre-process step. This will help you to have a deep understanding of the data and to define in a clear way your future model. Take into account that an accurate data representation ensures that the information fed into the system is both meaningful and relevant, leading to improved model performance and more accurate predictions.

Here you have a **notebook** that will introduce you in the **first steps** of this phase. Please make a copy to have it as a starting point:
https://colab.research.google.com/drive/1OLZMfAb1AAMrovHWrHHj1WE7deOSAFiq?usp=sharing

Then, the second phase will involve implementing a pretrained object detection model, such as **Faster R-CNN** or **YOLO**. The model will be used to detect and localize objects within images, with results being visualized using detected bounding boxes. Additionally, an evaluation module will be implemented to assess the performance of the detection model using mAP metric. This metric provides a measure of how well the predicted bounding box aligns with the actual object.

Phase 3. 6D Pose Estimation Module

This phase will focus on developing a pose estimation module using the detected bounding boxes from the previous step. The **Pose Predictor Model**, responsible for pose estimation, is a neural network composed of **fully connected layers**. It takes extracted object features as input and outputs both **translation** (position in 3D space) and **rotation** (orientation as a matrix). The system trains using a **loss function that combines translation loss** and **rotation loss** (MSE loss) to optimize the accuracy of pose predictions. This approach allows the model to first identify objects, extract relevant features, and then infer their spatial positioning and orientation efficiently.



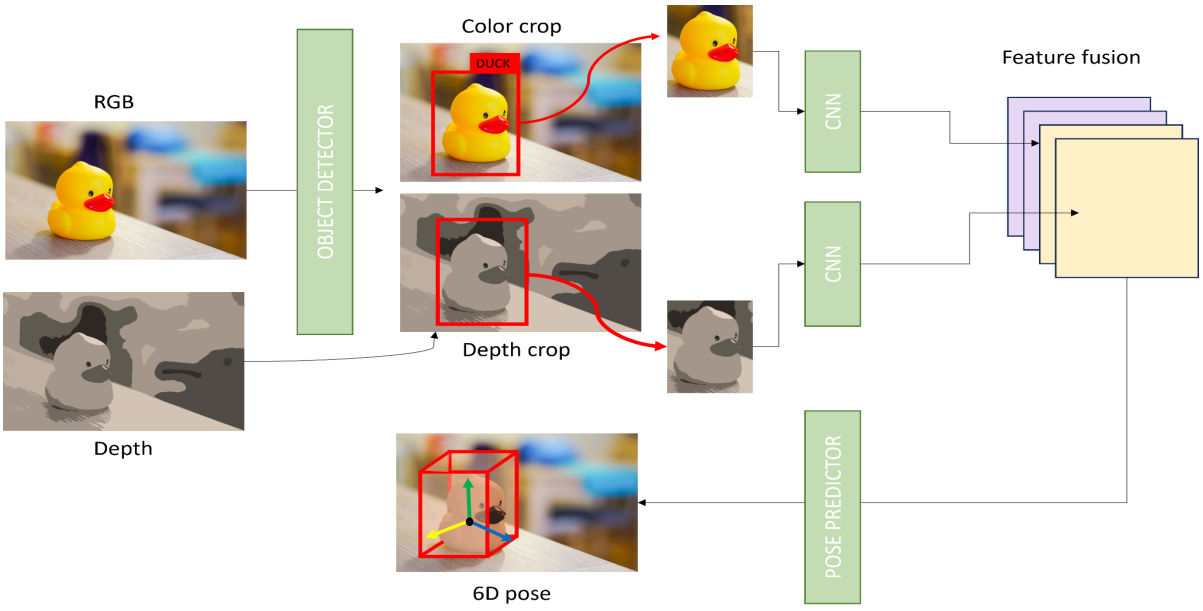
To evaluate the performance of your model, you need to use the ADD metric which calculates the average distance between corresponding points on a 3D model of an object. Until this point, you constructed the baseline of the project that will serve as a benchmark to compare your final model. In the following step we suggest some extensions that you could implement to enhance the 6D pose estimation process.

Phase 4. Extensions

As an extension you need to enhance the pipeline by incorporating depth information. You can select the approach that best suits you, in the following figure we suggest a RGB-D fusion technique inspired by **"DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion"**. This approach looks to leverage the two complementary data sources by processing the two data sources individually and uses a feature fusion step, from which the pose is estimated.

You need to replicate a simplified version of the model presented on the aforementioned paper, by using just a 2D approach instead of 3D, in order to exploit the advantages of the method but by avoiding the computational expense that 3D could represent.The model is a multi-modal pose estimation system that combines RGB and depth information to predict the 3D translation and rotation of objects. First, it detects objects in RGB images, extracting cropped regions for feature processing. Depth information is processed separately using a convolutional neural network. These RGB and depth features are then fused by a simple concatenation step. Then we have the pose estimator which predicts the object's translation (3D coordinates) and rotation (3×3 matrix). This fusion enhances pose estimation accuracy by leveraging both visual texture (RGB) and geometric structure (depth), making the model robust to variations in lighting and object appearance.

We recommend to identify first the main difference between the model proposed in the following diagram and the model proposed in the paper. For example, a key difference is that you need to realize the crop directly on the depth image, instead of using a point cloud. What other differences could you identify? Once you determine the modules that you need to design, you could build them by using the knowledge and the modules designed in the previous phase. At the end, you need to compare and report the performance of your RGB-only model with your novel approach.



Last but not least, remember that you have access to some other information on the dataset, such as the 3D model and mask of the objects. Please feel free to use this information to extend the project or to suggest modules that will improve its performance or that will allow you to explore the topic more deeply.

At The End

- Deliver PyTorch scripts for all the required steps.
- Write a complete PDF report (in paper style). The report should contain a brief introduction, a related work section, a methodological section for describing the algorithm that you're going to use, an experimental section with all the results and discussions, and a final brief conclusion.

Supporting material

1. As the first phase mentions, you need to **dive into the literature** to find the main approaches and metrics used to solve and evaluate this task. In the following presentation you could find some **general recommendation** to exploit the information that you will find in papers.
   https://docs.google.com/presentation/d/1i1qSlxpNuzQgDZxPO1kslc6lUnLdtAEC83wRwh5Q4vo/edit?usp=sharing

2. Please, consider sharing here the interesting **references** that you find during the project. This is a shared folder between all the students that will develop this project.
   https://docs.google.com/document/d/1Fut5ztx1IdUyNPL9yGm5TYTxTPADE6N4vBVGO6Sr5I0/edit?usp=sharing

References

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

[2] Zhao, Z., Peng, G., Wang, H., Fang, H. S., Li, C., & Lu, C. Estimating 6D pose from localizing designated surface keypoints. arXiv 2018. *arXiv preprint https://arxiv.org/pdf/1812.01387*

[3] Xiang, Y., Schmidt, T., Narayanan, V., & Fox, D. (2017). PoseCNN: A convolutional neural network for 6d object pose estimation in cluttered scenes. *https://arxiv.org/pdf/1711.00199*

[4] Yuanwei, C., Zaman, M. H. M., & Ibrahim, M. F. (2024). A Review on Six Degrees of Freedom (6D) Pose Estimation for Robotic Applications. IEEE Access.

[5] Marullo, G., Tanzi, L., Piazzolla, P., & Vezzetti, E. (2023). 6D object position estimation from 2D images: a literature review. *Multimedia Tools and Applications*, *82*(16), 24605-24643.

[6] Li, Y., Wang, G., Ji, X., Xiang, Y., & Fox, D. (2018). Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 683-698).

[7] Park, K., Patten, T., & Vincze, M. (2019). Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 7668-7677).

[8] Kehl, W., Manhardt, F., Tombari, F., Ilic, S., & Navab, N. (2017). Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE international conference on computer vision* (pp. 1521-1529).
[9] Chen, W., Zhang, H., Jiang, Y., Chen, Y., Chen, B., & Huang, C. (2023, November). An iterative attention fusion network for 6d object pose estimation. In *2023 China Automation Congress (CAC)* (pp. 9300-9305). IEEE.
[10] Aing, L., & Lie, W. N. (2021). Detecting object surface keypoints from a single RGB image via deep learning network for 6-DoF pose estimation. *IEEE Access*, *9*, 77729-77741.
[11] Kalra, A., Stoppi, G., Marin, D., Taamazyan, V., Shandilya, A., Agarwal, R., ... & Stark, M. (2024). Towards Co-Evaluation of Cameras HDR and Algorithms for Industrial-Grade 6DoF Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 22691-22701).
[12] Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., & Navab, N. (2012, November). Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision* (pp. 548-562). Berlin, Heidelberg: Springer Berlin Heidelberg.