doi: 10.82124/CEAS-GNC-2026-075 CEAS-GNC-2026-075

Performance Assessment of AI-driven relative pose estimation algorithms: YOLO vs CenterPose

Luis Rueda PhD Candidate, Universidad Rey Juan Carlos, Madrid, Spain. l.ruedac.2018@alumnos.urjc.es

Hodei Urrutxua Associate Professor, Universidad Rey Juan Carlos, Madrid, Spain. hodei.urrutxua@urjc.es

Xin Chen Assistant Professor, Universidad Rey Juan Carlos, Madrid, Spain. xin.chen@urjc.es

Miguel Leiva PhD Candidate, Universidad Politécnica de Madrid, Madrid, Spain. miguel.leiva.velez@alumnos.upm.es

Manuel Sanjurjo Full Professor, Universidad Carlos III de Madrid, Madrid, Spain. manuel.sanjurjo@uc3m.es

ABSTRACT

The growing demand for autonomous rendezvous, inspection, and Active Debris Removal (ADR) missions calls for reliable vision-based relative navigation under challenging orbital conditions, where classical feature-based pipelines often fail. Deep learning provides a powerful alternative, as modern CNN/Transformer backbones can extract robust, illumination-tolerant features even under texture poverty, occlusions, or degenerate views. This work benchmarks three representative AI-based 6 Degrees of Freedom (6-DoF) pose estimation approaches for non-cooperative spacecraft: YOLOv8+SQPnP, the new YOLOv11+SQPnP, and NVIDIA CenterPose. A custom Blender-generated dataset of the Deimos-1 satellite comprising 16,200 photorealistic grayscale images with systematically varied viewing and illumination geometries was used for training and evaluation. Results in controlled fly-around scenarios show that YOLOv11+SQPnP achieves the best overall balance between geometric accuracy and temporal stability, with translation errors around 45–60 mm and orientation errors near 2–3° across most viewpoints. CenterPose DLA-34 remains the most robust under adverse illumination or self-shadowing, consistently maintaining low variance and smooth trajectories (translation errors ~52–55 mm, rotation errors ~3.5°). YOLOv8+SQPnP provides the sharpest geometric fits (3DIoU up to 0.92) but suffers from high sensitivity to viewpoint and lighting. Within this controlled synthetic benchmark, these findings indicate that compact detector-plus-PnP pipelines can offer a lower-complexity alternative to heavier integrated backbones and are promising candidates for future onboard assessment, although embedded suitability was not benchmarked here.

Keywords: Relative pose estimation, Deep learning, Optical navigation, YOLO, CenterPose, Active Debris Removal, Proximity operations

Nomenclature

ψ, θ, ϕ = Euler angles (heading, pitch, roll)
ADR = Active Debris Removal
6-DoF = 6 Degrees of Freedom
CNN = Convolutional Neural Network
SQPnP = Sequential Quadratic Perspective-n-Point
3DIoU = 3D Intersection over Union
2DMPE = 2D Mean Pose Error
E_t = Absolute translation error
E_r = Absolute angular error
θ_c = Camera azimuth angle
ϕ_c = Target azimuth angle
d_c = Camera distance
θ_s = Camera zenith angle
ϕ_s = Target zenith angle
d_s = Sun distance
I_s = Sun intensity

1 Introduction

The growing congestion of Earth orbit and the increasing interest in Active Debris Removal (ADR) missions are setting new demands for relative navigation. Reliable, real-time estimation of position and orientation (6-DoF pose) is critical to guarantee safe autonomous proximity operations, rendezvous, and capture. In these missions, the target is typically non-cooperative: it lacks fiducial markers, exhibits irregular geometry, may tumble freely, and is subject to harsh or rapidly changing illumination. Monocular vision-based relative navigation has therefore become an attractive solution due to its reduced mass, power consumption, and system complexity, but it also poses major robustness challenges when operating on uncooperative targets [1]. These conditions make classical vision-based pipelines, built on edge alignment, landmark correlation, or iterative matching, fragile and initialization-dependent. A visual navigation system that is robust to texture poverty and photometric extremes is therefore critical for spacecraft close-range inspection, rendezvous, capture, and de-orbiting, directly impacting mission safety and sustainability.

Deep learning has become a practical enabler here. Recent surveys show that learning-based monocular pose-estimation methods are increasingly replacing hand-engineered pipelines, mainly because they can extract robust and task-relevant visual features under challenging illumination, partial occlusions, and poor-texture conditions. Modern Convolutional Neural Networks (CNN) and Transformer backbones can provide stable keypoints or projected geometric structures from which the pose can be recovered through geometric solvers or unified regression frameworks. At the same time, these methods still face important limitations, especially regarding computational cost, onboard deployability, and the sim-to-real or synthetic-to-real domain gap [2]. Nevertheless, recent advances in radiation-tolerant GPUs and SoCs suggest that such algorithms could soon be executed directly onboard spacecraft, enabling high-fidelity perception without fiducial markers and allowing adaptation to new targets by retraining rather than fully re-engineering the navigation pipeline.

Six Degrees of Freedom (6-DoF) pose estimation is the foundation of the perception pipeline in proximity operations: given a single image (or a short sequence), the task is to infer the target's translation and rotation in the camera frame so that guidance can plan safe approach trajectories and control can regulate relative motion. In practice, this problem is commonly solved by detecting 2D features in the image and associating them with 3D references, from which the pose is recovered via a Perspective-n-Point (PnP) solver.

Within this landscape, one family of methods preserves explicit geometry by detecting 2D features and recovering the final pose with a Perspective-n-Point (PnP) solver, while another integrates correspondence prediction and pose inference more tightly inside the network, as in CenterPose [3]. Despite rapid progress, three issues remain especially relevant for deployment: robustness under harsh lighting and degenerate viewpoints, computational efficiency on resource-constrained onboard hardware, and validation protocols that move beyond purely synthetic indicators toward more representative operating conditions [2, 4].

This work compares two representative and deployable AI-based approaches for spacecraft pose estimation with a known 3D model. Both rely on building 2D–3D correspondences and solving PnP problem, but differ in how these correspondences are obtained and how end-to-end the overall system is:

- YOLO + SQPnP: A keypoint detector based on You Only Look Once (YOLO) models predicts 2D landmarks that are then matched with their known 3D counterparts from a CAD model, recovering pose via a robust Sequential Quadratic Programming PnP (SQPnP) solver.
- NVIDIA's CenterPose: A unified network regresses the 2D projection of the object's 3D cuboid and its relative dimensions, internally establishing 2D–3D correspondences and solving PnP within the network. It supports both convolutional backbones (e.g., DLA-34) and Transformer-based ones (FAN-S/B/L), introducing global self-attention that improves spatial reasoning and robustness to lighting changes at the cost of higher computational demand.

Both models are "pretrained," i.e., their backbones start from weights learned on large, generic image corpora and are then fine-tuned on our spacecraft domain; this transfer learning reduces the data and time needed to reach high accuracy and mitigates the sim-to-real gap.

The models are trained and evaluated on a custom dataset of 16,200 photorealistic grayscale images of the Deimos-1 satellite, synthetically generated in Blender under varied viewing and illumination geometries. Performance is assessed in controlled a fly-around scenario, analyzing accuracy, robustness to occlusions and lighting extremes. A lightweight moving-average filter is also implemented to smooth translation and rotation estimates, reducing jitter.

2 Methodology

The proposed methodology follows a two-stage evaluation pipeline. Both architectures, YOLOv8+SQPnP and CenterPose, are trained and validated on the synthetic dataset introduced in Section 3. Although both rely on establishing 2D–3D correspondences and solving a Perspective-n-Point (PnP) problem, they differ in how these correspondences are obtained and how tightly coupled the overall estimation process is. YOLO+SQPnP follows a modular design, where 2D keypoints are predicted and subsequently processed by an external SQPnP solver. In contrast, CenterPose embeds the PnP computation within the network itself, jointly regressing the 2D cuboid projection and its 3D alignment, resulting in a more integrated, though still not fully end-to-end, pose estimation framework. This setup enables a fair comparison between explicitly modular and internally coupled PnP-based pipelines in terms of keypoint accuracy, reprojection error, and final 6-DoF pose performance.

2.1 YOLOv8-based architecture

The first pipeline integrates the single-stage YOLOv8 detector with a Sequential Quadratic Programming PnP solver (SQPnP). Its role is to predict a fixed set of 2D image keypoints associated with known CAD vertices; SQPnP then recovers the rigid pose that best explains those correspondences. Architecturally, YOLOv8 comprises three main components: a CSP backbone for feature extraction, an FPN+PAN neck for multiscale feature fusion, and a detection head for bounding boxes and keypoints, as shown in Figure 1.

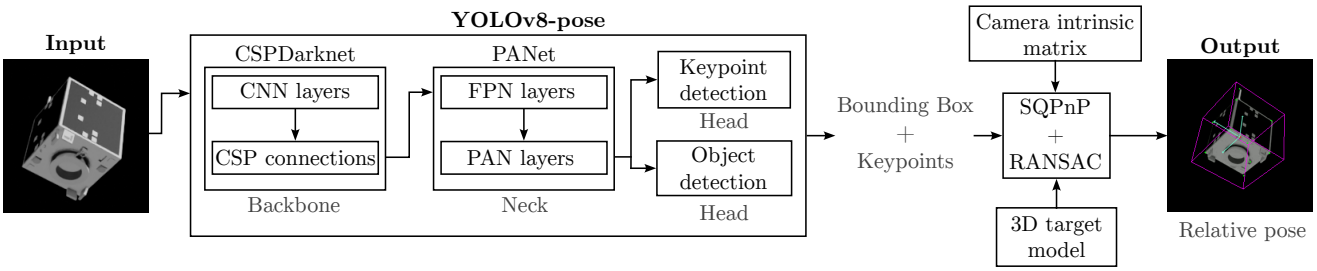


Fig. 1 YOLOv8 + SQPnP architecture for 6-DoF pose estimation.

The CSP backbone (*Cross-Stage Partial Network*) divides feature maps into two paths to enhance gradient flow and reduce parameter redundancy [5]. The neck combines a Feature Pyramid Network (FPN) [6] and a Path Aggregation Network (PAN) [7], merging semantic and spatial cues across scales for accurate localization of small or partially visible targets. The detection head predicts bounding boxes, object confidences, and 2D keypoints in a single forward pass.

Each keypoint detected by YOLOv8 corresponds to a known 3D vertex on the spacecraft model. These 2D-3D correspondences are used to solve the PnP problem via the SQPnP algorithm [8], which formulates it as a quadratic optimization task with guaranteed global convergence and low computational overhead. The inference chain is therefore explicit and interpretable: detect keypoints, map them to CAD vertices, and solve SQPnP for the final 6-DoF pose.

2.2 YOLOv11-based architecture

The second approach adopts *YOLOv11*, a recent single-stage detector in the Ultralytics YOLO family. As in YOLOv8+SQPnP, the network predicts 2D keypoints that are mapped to known CAD vertices and passed to SQPnP; the difference lies in the quality and stability of those image measurements. YOLOv11 preserves the compact single-stage design philosophy of YOLOv8, but introduces architectural refinements that improve multi-scale feature extraction and spatial reasoning under harsh illumination or strong foreshortening.

Architecturally, YOLOv11 preserves the standard three-part layout of modern YOLO detectors (backbone, neck, and detection head), but replaces several core blocks to increase representational power without a proportional increase in parameter count or inference cost. Figure 2 illustrates the overall structure.

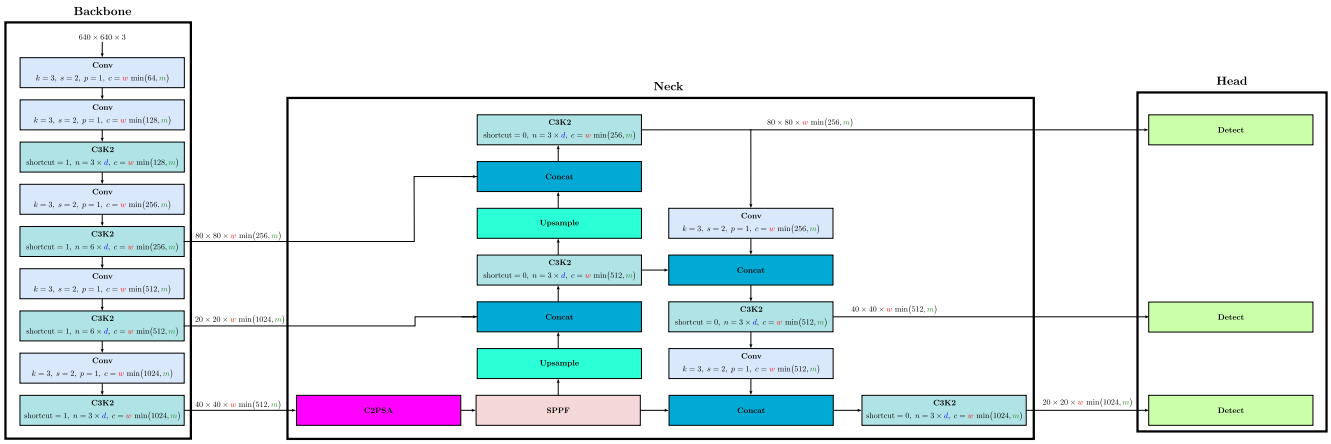


Fig. 2 YOLOv11 architecture [9].

The YOLOv11 backbone extracts semantically rich and spatially precise features through a CSP-based design that enhances gradient flow while reducing redundancy. It replaces the YOLOv8 C2f

bottleneck with the more efficient C3k2 block, which uses smaller-kernel convolutions to improve feature density per FLOP, and incorporates the C2PSA (*Cross-Stage Partial with Parallel Spatial Attention*) module after the SPPF block to focus on salient image regions. These upgrades aim to produce more stable geometry-aware features at similar architectural scale, although onboard suitability is not assessed in this paper.

The neck fuses multi-scale features in a top-down/bottom-up manner using C3k2 modules, enhancing cross-scale consistency and maintaining attention on informative structures even when the spacecraft is small or partially occluded in the image.

The detection head remains lightweight and modular, combining C3k2 and Conv–BatchNorm–SiLU layers for efficient inference. In the pose variant, YOLOv11 directly regresses 2D keypoints that are then fed into a PnP solver (SQPnP in this work) to estimate the 6-DoF pose. In practical terms, the downstream geometry is unchanged with respect to YOLOv8: detect keypoints, match them to the CAD model, and solve PnP.

Compared to YOLOv8+SQPnP, YOLOv11 introduces attention-driven spatial reasoning and more efficient multi-scale fusion, improving robustness to viewpoint and illumination variations without increasing model size. Unlike CenterPose, it retains a modular detect keypoints → solve PnP structure, preserving interpretability while narrowing the robustness gap under occlusion and extreme lighting.

2.3 CenterPose-based architecture

As a third approach, we consider *CenterPose* [3], a single-stage architecture for category-level 6-DoF pose estimation from a single monocular image. In contrast to the modular YOLO+PnP pipeline, which detects keypoints and then solves PnP externally, CenterPose predicts all geometric cues (object center, projected 3D cuboid corners, and relative cuboid dimensions) within one network forward pass, and then recovers the full pose via a standard PnP solver.

As illustrated in Fig. 3, the network first produces a heatmap whose peaks correspond to the 2D centers of the detected objects. From each detected center, the model regresses sub-pixel center offsets, the 2D projections of the eight vertices of the object's 3D bounding cuboid, and the relative cuboid dimensions (width/height/depth ratios). Corner locations are obtained through a dual representation: (i) displacement vectors from the predicted object center to each cuboid vertex, and (ii) per-vertex heatmaps. Combining both sources has been shown to improve robustness under strong intra-class shape variation and partial occlusion [3].

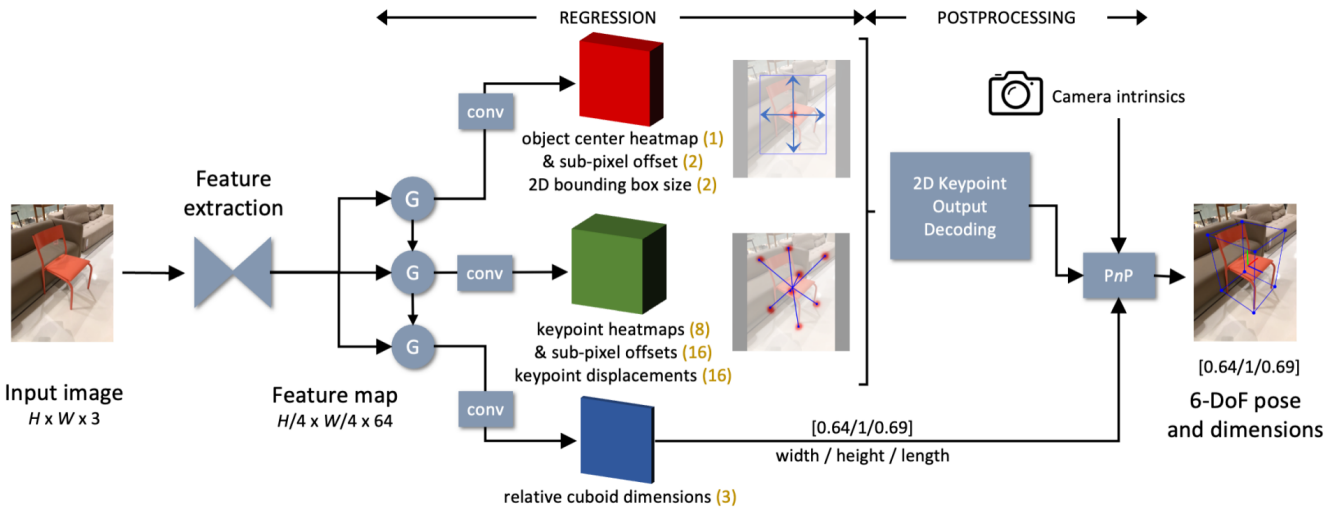


Fig. 3 CenterPose architecture for 6-DoF pose estimation [3].

Unlike purely two-stage designs, CenterPose groups its output heads by increasing difficulty (*center & 2D box → keypoints → 3D cuboid dimensions*) and links them through a convolutional gated recurrent unit (convGRU). The convGRU passes a hidden state forward across these groups, so that later heads (e.g. the cuboid dimension regressor) can reuse structure-aware features inferred by earlier heads (center, keypoints). This sequential feature association improves the stability of scale/shape estimates in challenging viewpoints.

Given the predicted 2D cuboid corners, the estimated relative cuboid dimensions, and the known camera intrinsics, CenterPose then solves for the 6-DoF pose (up to a global scale factor if absolute size is unknown) using a standard PnP solver (Levenberg–Marquardt PnP). In other words, rather than backpropagating through a differentiable pose layer, the network is trained to minimize heatmap/offset and dimension losses so that an off-the-shelf PnP algorithm can recover a stable pose at inference time.

In our implementation, we leverage NVIDIA TAO's CenterPose variant, which supports multiple backbones, from a lightweight DLA-34 CNN to hybrid Fully Attentional Networks (FAN-S/B/L) based on vision transformers [10].

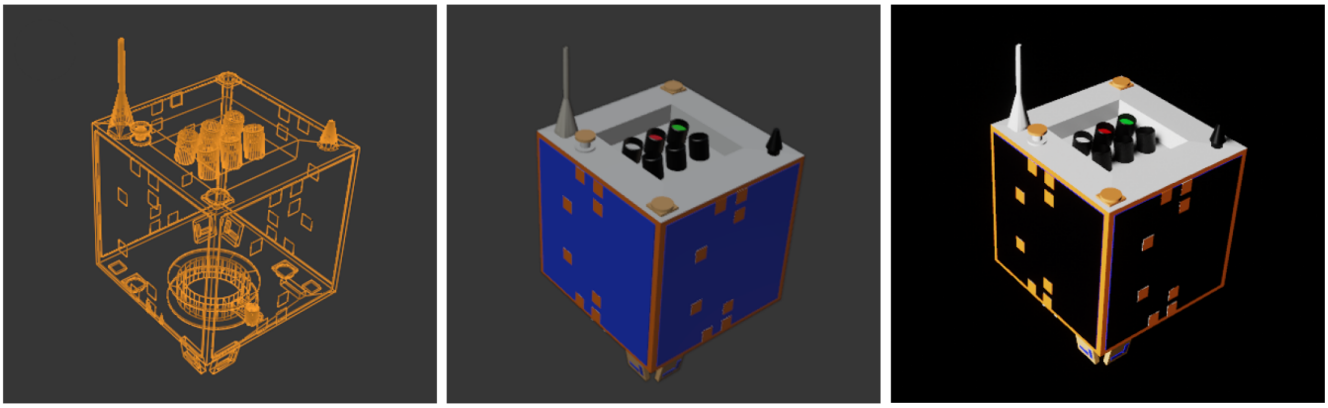
3 Training dataset

To train and evaluate the proposed 6-DoF pose estimators, we generated a synthetic, photorealistic dataset of 16200 grayscale images of the Deimos-1 satellite rendered in Blender. Camera–target geometry is systematically swept in azimuth/elevation while the camera range is sampled in $[3, 6]$ m; solar direction and intensity are varied to emulate realistic illumination. Each image is accompanied by automatic annotations: 2D/3D keypoints, a projected 3D cuboid, and full 6-DoF ground truth. Camera intrinsics replicate a 640×640 sensor with a 30 mm focal length ($f_x = f_y = 600$, $c_x = c_y = 320$); the dataset is split into 80% train and 20% validation.

This Blender corpus is intended as a controlled within-domain benchmark for training and comparing the two pipelines, not as a claim of flight-representative imagery. It captures the factors most relevant to the present comparison, target geometry, broad viewpoint coverage, Sun-angle variation, grayscale imaging, and perfectly consistent labels, but it is not validated against orbital or hardware-in-the-loop images. In particular, sensor-chain effects, background clutter, glints and specular extremes, exposure artifacts, calibration errors, and the broader sim-to-real gap are not modeled here; prior spacecraft-vision studies have shown that these aspects require dedicated domain-gap and validation campaigns beyond purely synthetic evaluation [4, 11].

3.1 Target 3D model

Following prior work, we employ a simplified yet metrically faithful CAD of Deimos-1 tailored for efficient real-time rendering in *Blender*. The model preserves global proportions and salient external details (panels, booms, reflective surfaces) while pruning sub-millimetric parts that do not meaningfully influence vision. This choice yields a representative, moderately symmetric geometry that is well suited to stress pose estimators under ambiguous viewpoints and partial self-occlusions. Figure 4 illustrates the CAD, external geometry, and a representative render of the target.



(a) CAD model. (b) External geometry. (c) Rendered view.

Fig. 4 3D modeling of the Deimos 1 satellite.

3.2 Camera model

The synthetic camera mirrors a square sensor frequently found in spaceborne payloads:

- Image resolution: 640×640 (B&W).
- Sensor size: $32\,\text{mm} \times 32\,\text{mm}$.
- Focal length: $30\,\text{mm} \Rightarrow f_x = f_y = 600$ px.
- Principal point: $c_x = c_y = 320$ px.
- Aperture: f/2.8. No lens distortion is applied.

This configuration provides a field of view adequate for near-range proximity operations while keeping sufficient detail for both keypoint-based and direct 6D heads.

3.3 Viewpoint and illumination sampling

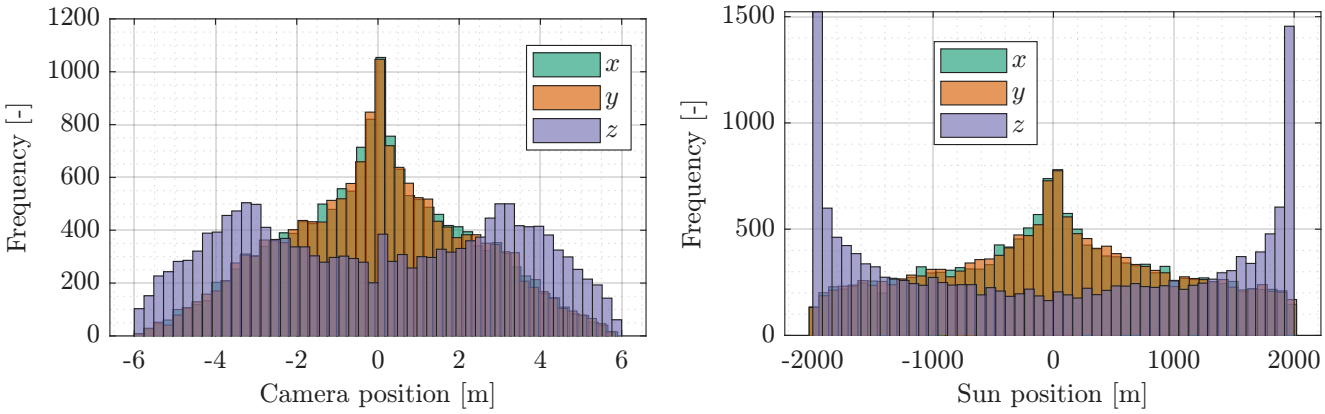
Camera poses are generated on a spherical shell around the satellite with stratified sweeps in azimuth $\phi_c \in [0, 2\pi)$ and elevation $\theta_c \in [-\pi/2, \pi/2]$; per-pose range d_c is drawn uniformly from $[3, 6]$ m. The Sun is modeled as a distant point light with random orientation (ϕ_s, θ_s) and a scene intensity parameter I_s expressed as a fraction of the solar constant at 1 AU ($I_0 \approx 1361\,\text{W}\,\text{m}^{-2}$); unless otherwise stated we set $I_s = 1$. Table 2 summarizes the sampling parameters used to generate the dataset.

Parameter	Value
Azimuth step $\Delta\phi_c$ (deg)	2
Elevation step $\Delta\theta_c$ (deg)	2
Camera range d_c (m)	$[3, 6]$
Sun elevation/azimuth	uniform on sphere
Sun intensity I_s	I_0

Table 2 Generation grid and physical ranges used to synthesize the dataset.

To illustrate the spatial diversity of viewpoints and illumination, Figure 5 shows the frequency distribution of both camera and Sun positions over the sampling sphere. The dense and homogeneous coverage confirms that the rendering process effectively spans the full range of orientations and lighting conditions required for robust model fine-tuning. It is worth noting that, in Blender, the Sun is modeled as a distant point light placed at a fixed distance of 2000 m from the target. This scaling does not affect

illumination geometry but facilitates consistent control of azimuth and elevation angles during dataset generation.



(a) Camera positions. **(b) Sun positions.**

Fig. 5 Frequency distribution of camera (left) and Sun (right) positions in the 3D sampling space.

3.4 Annotations and file structure

Each rendered frame is exported with *two parallel* annotations to enable training both pipelines without conversion: (i) Objectron-style JSON for CenterPose, and (ii) COCO-style JSON for YOLOv8 keypoints detection. All follow the OpenCV convention ($+x$ right, $+y$ down, $+z$ forward), use pixel coordinates in (x, y), and metric units for 3D quantities.

The train/val division is 80/20 with no frame overlap. Intrinsics are fixed and repeated in each Objectron entry to ensure reproducible PnP back-projections and fair cross-method evaluation.

3.5 Data augmentation

To enhance generalization while keeping the synthetic imaging geometry physically consistent, the YOLO-based models were trained with a controlled online augmentation policy. Photometric jitter (hue, saturation, and brightness) was used to widen the illumination distribution; spatial translations up to 10% and isotropic scaling up to 50% were used to reduce sensitivity to framing and apparent target size; diversified lateral appearance; *Mosaic* increased scale and layout diversity by combining four samples; *RandAugment* injected additional mild photometric/geometric variability; and random erasing removed up to 40% of local regions to emulate self-occlusions and partial visibility losses.

All augmentations were applied stochastically on-the-fly at training time, so the effective input distribution varied across epochs. No rotation, flipping, shear, or perspective warping was used, as such transforms would alter the physical consistency of the projected spacecraft geometry and its 2D keypoint labels. These augmentations are therefore intended to improve robustness within the synthetic domain explored here; they do not, by themselves, close the gap to real orbital imagery or replace dedicated sim-to-real validation.

3.6 Training

YOLO models were trained using the official Ultralytics framework. The total loss combines classification, bounding box, keypoint, and confidence terms:

$$\mathcal{L}_{total} = \sum_{s,i,j,k} (\lambda_{cls}\mathcal{L}_{cls} + \lambda_{box}\mathcal{L}_{box} + \lambda_{kpts}\mathcal{L}_{kpts} + \lambda_{kpts_{conf}}\mathcal{L}_{kpts_{conf}}),$$

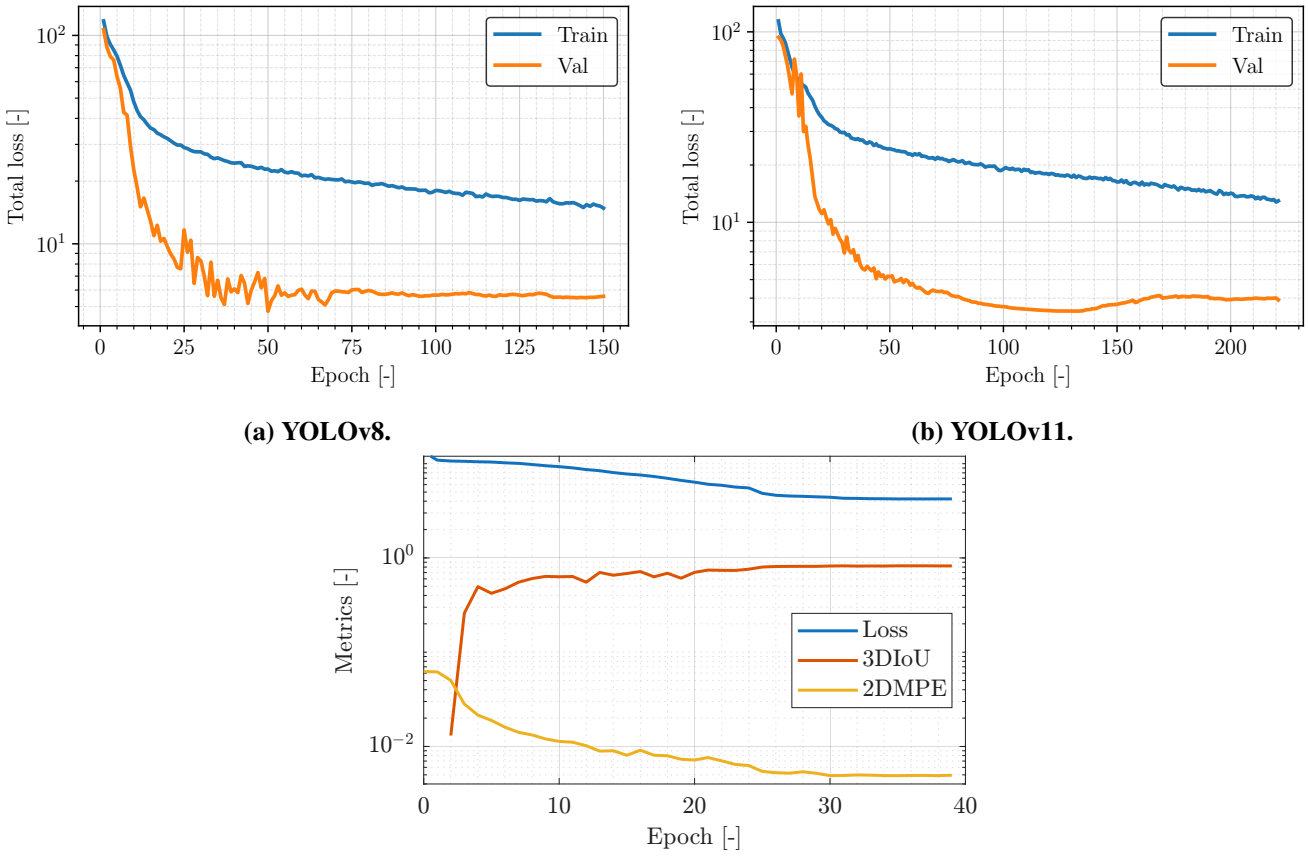
where $\mathcal{L}_{box} = 1 - CIoU(\cdot)$ and $\mathcal{L}_{kpts}$ penalizes Euclidean distances between predicted and true keypoints.

YOLOv8 training ran for up to 150 epochs with an initial learning rate of 10^{-2} decayed to 10^{-5}. As shown in Figure 6(a), the loss steadily decreases until epoch 50, after which the validation curve diverges, indicating overfitting. The model from epoch 50 was thus selected as the inference checkpoint. YOLO11 ran for 250 epochs, with the best validation loss observed at epoch 132.

CenterPose was trained using NVIDIA's TAO Toolkit with a DLA-34 backbone and FP16 precision. The loss follows [3], combining focal losses for heatmaps and L_1 terms for offsets, box size, and cuboid dimensions:

$$\mathcal{L}_{all} = \sum_i \lambda_i \mathcal{L}_i,$$

with $\lambda_{bbox} = 0.1$ and $\lambda_i = 1$ otherwise. Hyperparameters include a batch size of 16, 40 epochs, initial learning rate 10^{-4}, and decay steps at epochs 25, 32, 36. The variables shown in Figure 6c, 3DIoU and 2DMPE, are proxy metrics for the model accuracy, which will be introduced in subsection 4.1. As shown in Figure 6(b), the loss decreases consistently until epoch 30, then stabilizes; the model at that epoch offers the best trade-off between accuracy and generalization. Training lasted ~20 hours on an RTX 3080. Table 3 summarizes the training setups and selected checkpoints for both models.



(a) YOLOv8.

(b) YOLOv11.

(c) CenterPose.

Fig. 6 Training and validation loss evolution for both models. Each curve is smoothed with a short moving average to highlight trends.

Parameter	YOLOv8 + SQPnP	YOLO11 + SQPnP	CenterPose
Framework	Ultralytics	Ultralytics	NVIDIA TAO Toolkit
Backbone	YOLOv8n-pose	YOLO11n-pose	DLA-34
Batch size	16	16	16
Epochs	150	250	40
Learning rate	$10^{-2} \rightarrow 10^{-5}$	$10^{-2} \rightarrow 10^{-5}$	10^{-4} (steps 25/32/36)
Precision	FP16 (AMP)	FP16 (AMP)	FP16
Optimizer	Auto (SGD + cosine)	Auto (SGD + cosine)	MultiStep LR (decay 0.1)
Best epoch	50	132	30

Table 3 Training setup and selected checkpoints for all evaluated models.

4 Performance analysis

4.1 Evaluation metrics

To assess the accuracy of the estimated position and orientation, we employ standard metrics widely adopted in 3D vision and 6-DoF pose estimation, following the Objectron protocol [12] and recent works such as [3]. The following four metrics are considered:

- **3D Intersection over Union (3DIoU)**: Measures the overlap between the predicted 3D cuboid and the ground truth one; values range from 0 (no overlap) to 1 (perfect overlap).
- **2D Mean Projection Error (2DMPE)**: Quantifies the mean Euclidean distance between the projected vertices of the predicted and ground-truth cuboids on the image plane. The error is normalized by the image diagonal to enable comparison across different resolutions.
- **Absolute Angular Error (E_q)**: Measures the orientation discrepancy between the predicted unit quaternion $\tilde{q}$ and the ground-truth quaternion q, computed as

$$E_q = 2 \arccos\left(|\langle \tilde{q}, q \rangle|\right),$$

  where $\langle \tilde{q}, q \rangle$ denotes the inner product between quaternions. The result is expressed in degrees.
- **Absolute Translational Error (E_t)**: Represents the Euclidean distance between predicted and ground-truth translation vectors:

$$E_t = \|\tilde{t} - t\|_2,$$

  where $\| \cdot \|_2$ is the L2 norm, and the result is expressed in meters.

4.2 Simulation scenario

To analyze the performance and robustness of the proposed models, a synthetic test was designed to emulate typical orbital operation scenarios. The Deimos-1 satellite remains fixed at the origin of the coordinate system, while the camera follows controlled and realistic trajectory. The objective is to evaluate pose accuracy under different geometric configurations and illumination conditions. The test is conducted for all combinations of three camera zenith angles ($\theta_c = 0°, 30°, 60°$) and three solar zenith angles ($\theta_s = 0°, 45°, 90°$), thus covering a representative range of observation and lighting situations.

T1: **Fly-around 360° (circular trajectory)**
The camera describes a complete revolution around the satellite, varying the azimuth angle φ_c from 0° to 360° at a fixed distance $d_c = 4\,\text{m}$, while keeping the zenith angle θ_c constant for each

repetition. This test evaluates angular stability and positional/orientational accuracy when the object is observed from all viewpoints on a plane. It highlights potential geometric ambiguities, symmetries, or unfavorable viewpoints for pose estimation. The experiment is repeated for each combination of θ_c and θ_s.

Sequences of 250 frames per configuration are generated at a resolution of 640×640 pixels. Translational and rotational errors, along with the 2DMPE and 3DIoU metrics are recorded.

4.3 Results T1

Before discussing aggregated performance, it is instructive to examine the empirical error distributions. Fig 7–10 shows the histograms of 3DIoU, 2DMPE, E_q and E_t for a representative configuration, $\theta_c = 45°$ and $\theta_s = 30°$. All four metrics exhibit a pronounced central mode with approximately symmetric spread around the mean, close to unimodal Gaussian-like behaviour. This is relevant for two reasons. First, it confirms that the mean μ and standard deviation σ are meaningful summary statistics for these trajectories, i.e. performance is not dominated by a handful of catastrophic outliers except in very specific viewing conditions. Second, it enables a fair comparison between pipelines not only in terms of their peak accuracy (high IoU / low error) but also in terms of stability (low σ).

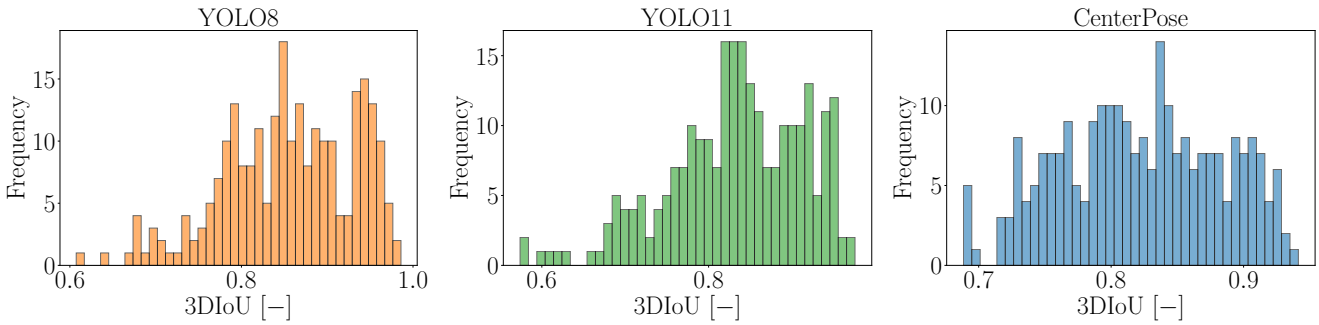


Fig. 7 Histograms of 3DIoU for $\theta_c = 45°$ and $\theta_s = 30°$ in Test T1.

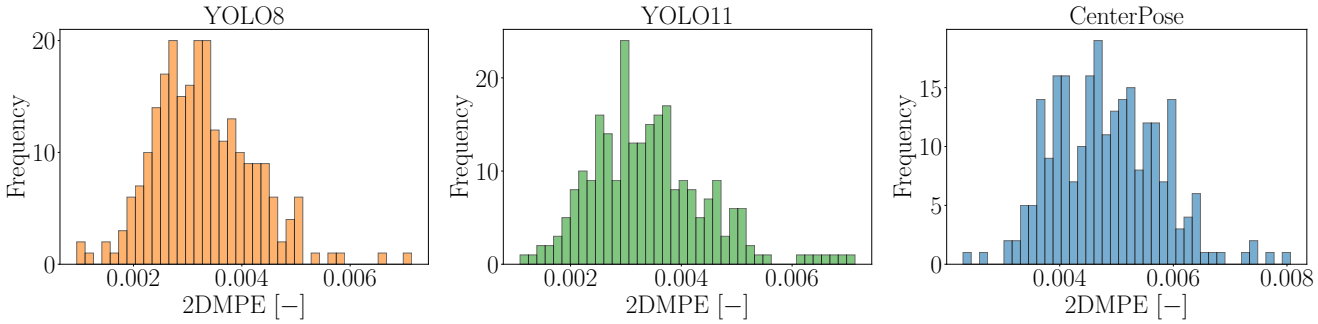


Fig. 8 Histograms of 2DMPE for $\theta_c = 45°$ and $\theta_s = 30°$ in Test T1.

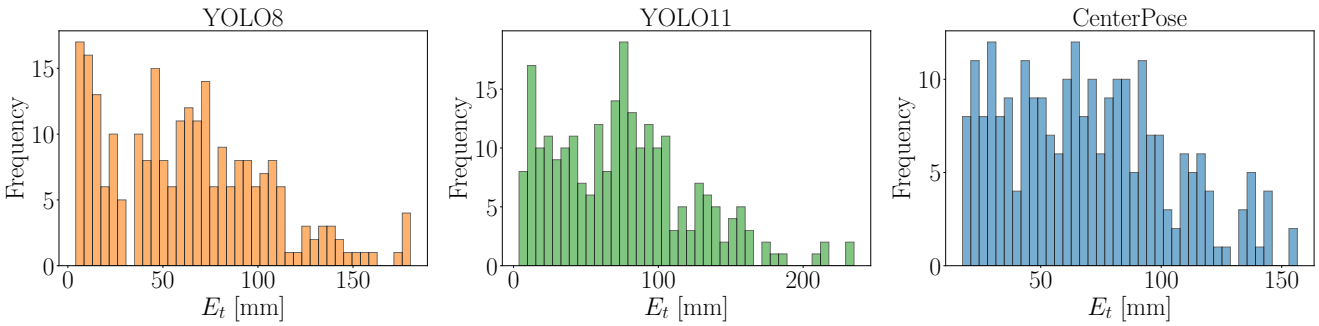


Fig. 9 Histograms of translational error for $\theta_c = 45°$ and $\theta_s = 30°$ in Test T1.

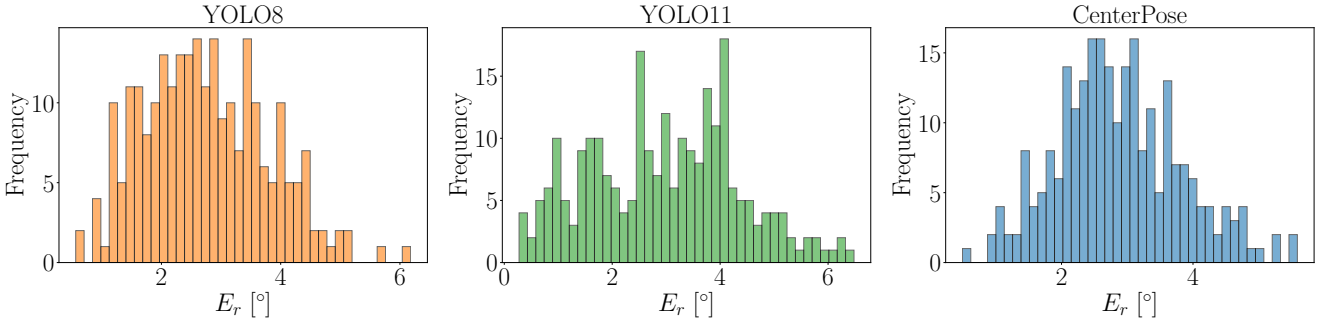


Fig. 10 Histograms of angular error for $\theta_c = 45°$ and $\theta_s = 30°$ in Test T1.

Tables 4–7 report the mean μ and standard deviation σ of the four metrics, grouped by camera zenith θ_c and solar zenith θ_s.

Table 4 Mean (μ) and standard deviation (σ) of the translation error E_t [mm] across all lighting and viewpoint configurations. Lower values indicate higher accuracy in position estimation.

E_t [mm]	Model	$\theta_s = 0°$	$\theta_s = 45°$	$\theta_s = 90°$
$\theta_c = 0°$	YOLO11	78.711 ± 58.861	85.235 ± 64.131	72.941 ± 53.860
	YOLO8	89.252 ± 195.022	109.758 ± 382.684	141.220 ± 744.897
	CenterPose	52.247 ± 31.970	54.965 ± 31.664	53.233 ± 33.489
$\theta_c = 30°$	YOLO11	72.888 ± 48.125	75.937 ± 47.683	69.419 ± 40.801
	YOLO8	59.972 ± 41.458	63.093 ± 41.048	69.586 ± 40.433
	CenterPose	69.805 ± 34.501	69.599 ± 33.449	65.150 ± 32.151
$\theta_c = 60°$	YOLO11	51.857 ± 36.102	41.768 ± 33.004	44.448 ± 34.155
	YOLO8	38.588 ± 24.669	44.230 ± 24.992	27.911 ± 22.056
	CenterPose	48.446 ± 22.571	53.227 ± 25.208	47.761 ± 23.906

Table 5 Mean (μ) and standard deviation (σ) of the rotation error E_r [°], comparing predicted and ground-truth orientations for each camera and solar zenith combination. Lower values indicate better attitude estimation.

E_r [°]	Model	$\theta_s = 0°$	$\theta_s = 45°$	$\theta_s = 90°$
$\theta_c = 0°$	YOLO11	4.659 ± 5.820	4.174 ± 4.689	3.822 ± 3.732
	YOLO8	8.752 ± 22.377	8.840 ± 22.358	9.526 ± 24.746
	CenterPose	3.637 ± 2.260	3.593 ± 2.030	3.389 ± 1.415
$\theta_c = 30°$	YOLO11	3.388 ± 1.181	2.960 ± 1.378	2.554 ± 1.133
	YOLO8	2.878 ± 1.081	2.742 ± 1.039	2.170 ± 1.021
	CenterPose	2.587 ± 1.055	2.861 ± 0.942	2.482 ± 0.985
$\theta_c = 60°$	YOLO11	1.843 ± 0.850	2.182 ± 0.976	1.813 ± 0.784
	YOLO8	2.356 ± 1.058	2.447 ± 1.265	2.044 ± 0.890
	CenterPose	2.164 ± 0.848	2.157 ± 0.864	2.478 ± 0.964

Table 6 Mean (μ) and standard deviation (σ) of the 3D Intersection over Union (3DIoU) between predicted and ground-truth cuboids. Higher values denote greater volumetric consistency of the estimated 6-DoF pose.

3DIoU [−]	Model	$\theta_s = 0°$	$\theta_s = 45°$	$\theta_s = 90°$
$\theta_c = 0°$	YOLO11	0.766 ± 0.124	0.759 ± 0.133	0.780 ± 0.104
	YOLO8	0.766 ± 0.155	0.759 ± 0.161	0.758 ± 0.167
	CenterPose	0.815 ± 0.073	0.809 ± 0.069	0.812 ± 0.071
$\theta_c = 30°$	YOLO11	0.835 ± 0.081	0.833 ± 0.081	0.845 ± 0.074
	YOLO8	0.859 ± 0.080	0.855 ± 0.076	0.847 ± 0.078
	CenterPose	0.822 ± 0.059	0.822 ± 0.060	0.830 ± 0.058
$\theta_c = 60°$	YOLO11	0.878 ± 0.065	0.896 ± 0.064	0.893 ± 0.064
	YOLO8	0.900 ± 0.052	0.886 ± 0.052	0.921 ± 0.044
	CenterPose	0.854 ± 0.052	0.845 ± 0.056	0.855 ± 0.053

Table 7 Mean (μ) and standard deviation (σ) of the normalized 2D Mean Projection Error (2DMPE) between projected cuboid vertices in image space. Lower values correspond to better image-plane alignment.

2DMPE [−]	Model	$\theta_s = 0°$	$\theta_s = 45°$	$\theta_s = 90°$
$\theta_c = 0°$	YOLO11	0.005 ± 0.004	0.004 ± 0.004	0.004 ± 0.003
	YOLO8	0.007 ± 0.012	0.007 ± 0.012	0.008 ± 0.013
	CenterPose	0.005 ± 0.002	0.005 ± 0.002	0.005 ± 0.001

2DMPE [−]	Model	$\theta_s = 0°$	$\theta_s = 45°$	$\theta_s = 90°$
$\theta_c = 30°$	YOLO11	0.004 ± 0.001	0.003 ± 0.001	0.003 ± 0.001
	YOLO8	0.003 ± 0.001	0.003 ± 0.001	0.003 ± 0.001
	CenterPose	0.005 ± 0.001	0.005 ± 0.001	0.005 ± 0.001
$\theta_c = 60°$	YOLO11	0.002 ± 0.001	0.002 ± 0.001	0.002 ± 0.001
	YOLO8	0.003 ± 0.001	0.003 ± 0.001	0.002 ± 0.001
	CenterPose	0.005 ± 0.001	0.005 ± 0.001	0.005 ± 0.001

Results in Tables 4–7 highlight clear performance differences among the three models. At near-nadir views ($\theta_c = 0°$), CenterPose DLA-34 consistently provides the most accurate and stable estimates, achieving the lowest translational errors ($\sim$ 52–55 mm) and angular errors around 3.5°. Its 3DIoU remains high (≈ 0.81) and nearly unaffected by illumination changes, confirming its robustness to self-shadowing and limited texture. In contrast, YOLOv8+SQPnP shows large variability, with translation errors occasionally exceeding 0.1 m and very high standard deviations, revealing instability in the PnP stage. YOLO11 already improves over YOLOv8 in this configuration, producing smaller variances and slightly lower 2D reprojection errors (2DMPE ≈ 0.004).

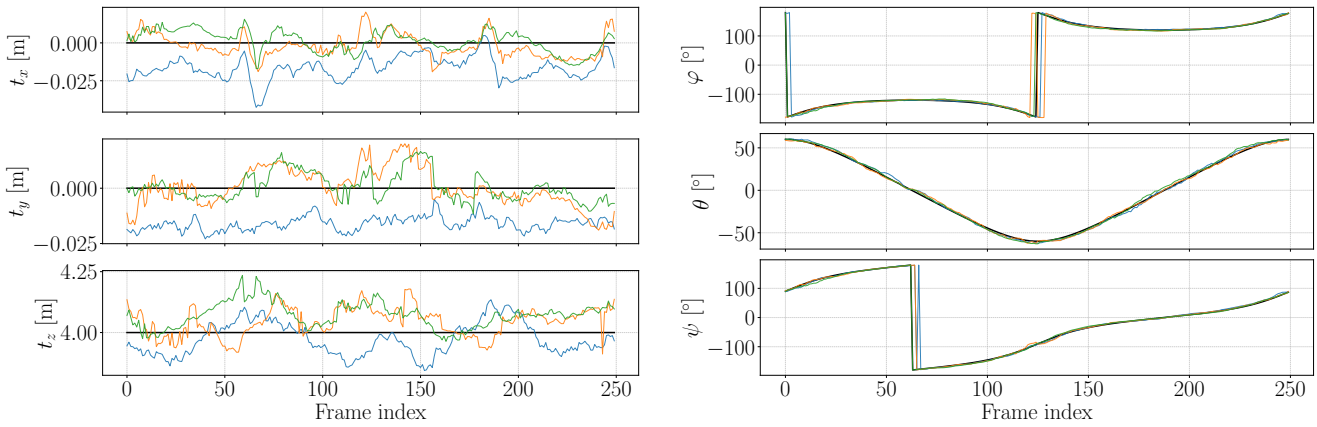
For intermediate viewpoints ($\theta_c = 30°$), the YOLO-based models narrow the gap. YOLOv8 attains the highest geometric consistency (3DIoU = 0.86) and the lowest mean translation errors ($\approx$ 60 mm), while YOLO11 delivers the best orientation estimates ($E_r \approx 2.6°$) with minimal 2D error (2DMPE = 0.003). CenterPose remains competitive (3DIoU ≈ 0.82) but slightly underperforms in accuracy despite its smaller variance, reflecting a more conservative but steadier behaviour.

At steeper geometries ($\theta_c = 60°$), YOLO11 becomes the overall top performer. It achieves the lowest rotation errors (1.8–2.2°) and 2DMPE values (0.002), maintaining consistent translation accuracy ($\sim$ 45 mm). YOLOv8 attains the highest volumetric overlap (3DIoU = 0.92) and the smallest mean translation error ($\sim$ 28 mm) under lateral illumination, indicating stronger geometric precision when sufficient texture is visible. CenterPose, while still stable, shows slightly degraded 3DIoU (0.85) and moderate orientation accuracy ($\sim 2.3°$), suggesting that its center-based formulation may saturate under extreme foreshortening.

Overall, CenterPose DLA-34 excels in robustness and low variance, making it well-suited for unconstrained scenarios or degraded visual conditions. YOLOv8+SQPnP provides the best geometric fit (highest IoU and lowest translation errors) but exhibits high sensitivity to viewpoint and illumination. YOLO11 achieves a strong balance between both extremes, retaining YOLOv8's geometric precision while improving rotational stability and smoothness across views.

In practical terms, CenterPose is preferable for general-purpose navigation with unpredictable lighting or attitude, whereas YOLOv8 and YOLO11 are advantageous when observation geometry can be controlled, such as in structured inspection or approach trajectories. Among the two, YOLO11 stands out as the most consistent and accurate evolution of the YOLO-based pipeline.

The aggregated metrics capture global accuracy, but they hide how the estimators behave along the trajectory. Figures 11a and 11b plot, for the configuration $\theta_c = 30°$ and $\theta_s = 45°$, the frame-by-frame evolution of the estimated translation (x, y, z) and orientation (Euler angles), together with the ground truth. These time histories allow us to assess *temporal smoothness* and to localize bursts of error.



(a) Translation components (x, y, z) [m]. **(b) Euler angles [°].**

Fig. 11 Temporal evolution for Test T1 with $\theta_c = 30°$ and $\theta_s = 45°$. GT in ■, YOLOv8 in ■, YOLO11 in ■, CenterPose in ■.

The three models reproduce the translational trajectory with comparable accuracy. The deviations from the ground truth are very small in magnitude, and all methods maintain consistent trends along the sequence. This indicates that the geometric scale and depth are well preserved, with no noticeable drift in any of the axes.

More pronounced differences appear in the attitude dynamics. During the abrupt transitions in φ and ψ, YOLO11 responds almost instantaneously, closely following the ground truth discontinuity with minimal overshoot. This behaviour reflects a high temporal responsiveness and accurate attitude tracking. By contrast, YOLOv8 exhibits a clear temporal lag: it begins to oscillate before and after the discontinuity, overshooting both sides of the jump and revealing a lack of temporal consistency in its PnP-based orientation estimates. CenterPose behaves in between both extremes—its transitions are smooth and continuous, avoiding oscillations, but they occur with a slight delay relative to the ground truth and YOLO11, suggesting a more conservative temporal response.

5 Conclusions

This work compared three AI-based pipelines for monocular 6-DoF spacecraft pose estimation: YOLOv8+SQPnP, YOLOv11+SQPnP, and CenterPose. Trained and evaluated on a synthetic Deimos-1 dataset, the results highlight complementary strengths within the controlled scenarios studied here. CenterPose achieved the most robust and stable performance under challenging illumination and near-nadir views ($E_t \approx 5$–$6\,\text{cm}$, $E_r \approx 3.5°$), confirming its resilience to texture loss and shadows. YOLOv8 provided the highest geometric accuracy in favorable, well-lit conditions but showed sensitivity to viewpoint and lighting. YOLOv11 successfully bridged this gap, offering smoother attitude estimation, improved temporal stability, and top performance at steep geometries while maintaining the explicit detect-keypoints → solve-PnP structure of the modular family.

These conclusions should be interpreted as the outcome of a controlled synthetic benchmark, not as a demonstration of space-ready perception. The present evidence supports a relative comparison between architectures under a common rendering and labeling pipeline, but operational use would additionally require hardware-in-the-loop or orbital-image validation, explicit treatment of the synthetic-to-real domain gap, and embedded inference benchmarking on representative flight processors [4, 11, 13].

CenterPose is preferable for general-purpose or degraded conditions, whereas YOLOv11+SQPnP appears more attractive for structured inspection or approach trajectories where geometry is predictable and a lower-complexity pipeline is desirable. Embedded suitability, however, was not benchmarked here.

Acknowledgments

Authors wish to acknowledge the Spanish State Research Agency for their support through the Research Grant PID2024-161963OB-C21 funded by MICIU/ AEI / 10.13039/501100011033 / FEDER, UE.

Declaration of Use of Artificial Intelligence

Artificial Intelligence tools, specifically OpenAI's ChatGPT, were employed exclusively to assist in language refinement and text proofreading. All technical content, data analysis, experimental design, and interpretations presented in this work are entirely the author's own.

References

[1] Luca Pasqualetto Cassinis, Robert Fonod, and Eberhard Gill. Review of robustness and applicability of monocular pose estimation for spacecraft relative navigation. *Progress in Aerospace Sciences*, 110:100548, 2019. doi: 10.1016/j.paerosci.2019.100548.

[2] Tobias Pauly and Simone D'Amico. Survey on deep learning-based techniques for spacecraft relative navigation. *Acta Astronautica*, 206:279–297, 2023. doi: 10.1016/j.actaastro.2023.03.028.

[3] Justin Lin, Siyuan Peng, Hao Liu, and Bolei Zhou. Centerpose: One-stage 6-dof object pose estimation with convgru refinement. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3717–3723, 2022.

[4] Vishnu Muralidharan, Sumant Sharma, and Simone D'Amico. On-ground validation of orbital vision-based navigation systems. *Journal of Guidance, Control, and Dynamics*, 47(4):728–741, 2024. doi: 10.2514/1.G007003.

[5] Chien-Yao Wang, Hong-Yuan Mark Liao, I-Hau Yeh, Yueh-Hua Wu, Ping-Yang Chen, and Jun-Wei Hsieh. CSPNet: A new backbone that can enhance learning capability of CNN. *CoRR*, abs/1911.11929, 2019.

[6] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *CoRR*, abs/1612.03144, 2016.

[7] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8759–8768, 2018. doi: 10.1109/CVPR.2018.00913, `http://openaccess.thecvf.com/content_cvpr_2018/html/Liu_Path_Aggregation_Network_CVPR_2018_paper.html`.

[8] George Terzakis and Manolis Lourakis. A consistently fast and globally optimal solution to the perspective-n-point problem. In *European Conference on Computer Vision (ECCV)*, pages 478–494. Springer, 2020. doi: 10.1007/978-3-030-58598-3_28.

[9] S. Nikhileswara Rao. Yolov11 architecture explained: Next-level object detection with enhanced speed and accuracy, oct 2024. Medium article, 10 min read. `https://medium.com/@nikhil-rao-20/yolov11-explained-next-level-object-detection-with-enhanced-speed-and-accuracy-2dbe2d376f71`.

[10] Daquan Zhou, Yujun Wu, Xiaojie Luo, Zichao Chen, Zhijian Yu, Jose M Alvarez, Qibin Wang, Xiangyu Zhang, Xiaolong Yue, et al. Fan: A family of attention networks for image classification and beyond. In *NeurIPS*, 2022.

[11] Tae Ha Park, Marcus Märtens, Gurvan Lecuyer, Dario Izzo, and Simone D'Amico. SPEED+: Next-generation dataset for spacecraft pose estimation across domain gap. In *2022 IEEE Aerospace Conference (AERO)*, pages 1–15, 2022. doi: 10.1109/AERO53065.2022.9843439.

[12] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[13] Sumant Sharma and Simone D'Amico. Pose estimation for non-cooperative rendezvous using neural networks. *CoRR*, abs/1906.09868, 2019. doi: 10.48550/arXiv.1906.09868.