RELATIVE POSE ESTIMATION PIPELINE FOR DOCKING OF SMALLSATS

A. Agbaria*, O. J. Jarosik*, B. Jäger*
* Technische Universität Berlin, Institute of Aeronautics and Astronautics, Straße des 17. Juni 135, 10623 Berlin, Germany

Abstract

This work proposes an accurate real-time 6 DoF pose estimation pipeline implemented around a rendezvous and docking ground simulator for SmallSats using a stereo-camera setup. The pipeline processes synchronized stereo frames with an interface segmentation model to isolate the docking target, and masked stereo outputs are generated frame-wise for efficient downstream processing. Additionally it considers recent breakthroughs in deep learning by utilizing VSS models, specifically a VMamba-based keypoint backbone [1], in order to efficiently extract image keypoints in real-time, followed by different post-processing and robustness strategies to ensure accurate results. The prediction model is trained on a synthetic dataset generated from a Gaussian Splatting render of the docking interface, followed by dedicated preprocessing and augmentation steps. Lastly we attempt to robustly explore the correspondence hypothesis space by using homography scoring and temporal consistency, eliminating erroneous and ambiguous solutions enabling us to continuously predict the relative pose between the observed docking interface and the rest pose.
The docking interface under investigation is the intelligent Space System Interface (iSSI) from iBOSS. It is primarily designed for modular satellites and OOS. Accuracy requirements are strict to mate the active iSSI module mounted on a chaser to the passive iSSI module mounted on a target. The goal of the 6DoF pose estimation pipeline is to reach a rotational precision of 0.5° and a translational precision of 1 mm at 1-2 Hz.

Keywords

Satellite docking systems; Deep learning; Computer vision; 3D pose estimation; Stereo vision; Gaussian splatting

NOMENCLATURE

Abbreviations

3DGS	3D Gaussian Splatting
CAD	Computer-Aided Design
DoF	Degrees of Freedom
IPPE	Infinitesimal Plane-based Pose Estimation
iSSI	intelligent Space System Interface
LEO	Low Earth Orbit
NeRF	Neural Radiance Field
PCK	Percentage of Correct Keypoints
PnP	Perspective-n-Point
RANSAC	Random Sample Consensus
RMS	Root Mean Square
VSS	Visual State Space

1. INTRODUCTION

As low cost, fast to produce, and useful for Earth observation, the population of SmallSats in LEO is expected to grow substantially. This exponential increase in space debris necessitates active debris removal or on-orbit servicing missions, which utilize docking interfaces that enable either removal or servicing of spacecraft using the SmallSat platform. The success of these missions requires precision, reliability, and real-time adjustment of docking interfaces in harsh environments with potential occlusions and extreme lighting conditions. Hence, navigation and knowledge of the relative pose are crucial.
Autonomous satellite servicing missions require reliable rendezvous and docking between a servicer and a client satellite. This problem is investigated in the CubeSat Robotic Environment for Educational Research (CAREER), a ground testbed at the Institute of Aeronautics and Astronautics of the TU Berlin.
The final approach and mating phase is especially demanding, because it requires accurate and stable 6DoF relative pose estimation at low latency. In practice, vision-based methods are currently the only realistic option to achieve the required precision in this

©2026

doi: 10.25967/680406

stage. For this reason, the CAREER servicer uses a stereo camera configuration (e.g., ZED Mini) mounted in the lower sensor openings.
In addition to accuracy, onboard feasibility is a core requirement. The target deployment profile is therefore an edge-AI platform (e.g., Jetson-class hardware), with practical power limits on the order of ≤ 100 W and preferably lower. This strongly favors lightweight, throughput-efficient pipelines that can sustain 1-2 Hz and scale toward higher frame rates.
Previous attempts have been completed, under the same project, but did not reach satisfactory accuracy and frame rates. This project aims to improve upon the previous pipeline through improving robustness and decreasing the computation time for onboard feasibility. The implemented runtime pipeline processes synchronized stereo frames, performs interface-focused segmentation, generates masked stereo outputs, regresses keypoint detection heatmaps, and performs various PnP robustness attempts, to predict a 6DoF temporally conditioned relative object rigid transformation, which provides a robust perception backbone for docking-interface tracking in CAREER.

2. RELATED WORKS

Vision-based 6DoF estimation commonly predicts 2D–3D correspondences and recovers pose geometrically. PVNet votes for projected keypoints and is robust to partial occlusion, whereas GDR-Net directly regresses pose from geometry-guided dense correspondences [2, 3]. IRPE instead reconstructs instance-level geometry to improve robustness in clutter and under heavy occlusion [4]. These methods target general objects and typically assume RGB-D input, a CAD model, or substantial object-specific training. CMT-6D reduces the computational burden through iterative cross-modal attention [5]; however, it still relies on RGB-D fusion. Our setting is narrower but more constrained: the target is a known, nearly planar docking interface, metric stereo is available, and inference must remain feasible on spacecraft edge hardware.

When labeled target data are scarce, cross-view consistency can supervise 3D keypoints without dense pose annotation [6]. Neural rendering provides another route: C2Fi-NeRF estimates pose by coarse-to-fine inversion of a learned radiance field [7]. Such analysis-by-synthesis methods optimize against the scene representation at inference time. In contrast, we use 3D Gaussian Splatting only offline to reproduce the as-built interface and render metrically labeled stereo views [8]; online estimation remains a feed-forward keypoint-and-geometry pipeline.

The closest baseline is the preceding CAREER study [9]. It detected the interface, regressed six heatmaps with a U-Net, triangulated stereo keypoints, and aligned them by Kabsch–Umeyama. With ground-truth depth it achieved 1.7044 ± 1.9680 mm translation error, but disparity-derived depth increased this to 80.3338 ± 72.7101 mm.

3. PROBLEM DEFINITION

CAREER consists of two Universal Robots manipulators with CubeSat mockups as end-effectors. Both platforms use iBOSS's intelligent Space System Interface (iSSI®), demonstrated in orbit in 2022. The client and servicer geometry in this work follows the CAD-based setup, with dedicated sensor openings and a docking-interface-centered front layout.

3.1. Inputs, Outputs, and Assumptions

The pipeline takes as input:
- a rectified stereo pair $(I_{\mathrm{L}}, I_{\mathrm{R}})$,
- known pinhole intrinsics $K_{\mathrm{L}}, K_{\mathrm{R}}$ (zero distortion),
- known stereo extrinsics with R_{base} and t_{base},
- the 3D rest-frame keypoint positions $\{\hat{\mathbf{p}}_i\}_{i=0}^{10}$ in $\mathcal{F}_{\mathrm{obj}}$ at the canonical docked pose,
- the sign of the interface face normal (front-facing constraint).

It outputs the instantaneous pose of the interface in the left camera frame $\mathcal{F}_{\mathrm{L}}$:
- a rotation $R \in \mathrm{SO}(3)$ and translation $\mathbf{t} \in \mathbb{R}^3$ mapping $\mathcal{F}_{\mathrm{obj}} \to \mathcal{F}_{\mathrm{L}}$,
- equivalently, a roll angle $\tilde{\theta}_z$ about the interface Z-axis relative to rest, and a translation $\tilde{\mathbf{t}}$ in $\mathcal{F}_{\mathrm{L}}$.

We assume:
- The interface keypoints are coplanar:

$$\hat{\mathbf{p}}_i^{\top} \mathbf{n}_{\mathrm{obj}} = d, \quad \forall i, \tag{1}$$

  with a common plane normal $\mathbf{n}_{\mathrm{obj}}$ (aligned with the interface Z-axis) and offset d.
- The stereo rig is rectified with a known metric baseline and aligned horizontal epipolar lines. The extrinsics between the cameras are

$$R_{\mathrm{base}} = I, \quad t_{\mathrm{base}} = \begin{bmatrix} b & 0 & 0 \end{bmatrix}^{\top}, \tag{2}$$

  where $b > 0$ is the fixed baseline distance along the X-axis of $\mathcal{F}_{\mathrm{L}}$.
- The front face is visible, and its normal sign is known, excluding back-facing solutions (Coplanar mirrored solution ambiguity).

4. DATA GENERATION

To reduce the sim-to-real gap while keeping data generation scalable, we reconstruct the iSSI client geometry with 3D Gaussian Splatting (3DGS) [8] from real video captures of the CAREER setup. The resulting splat serves as a metric 3D template and as an automatic labelling engine for stereo training and test data.

©2026

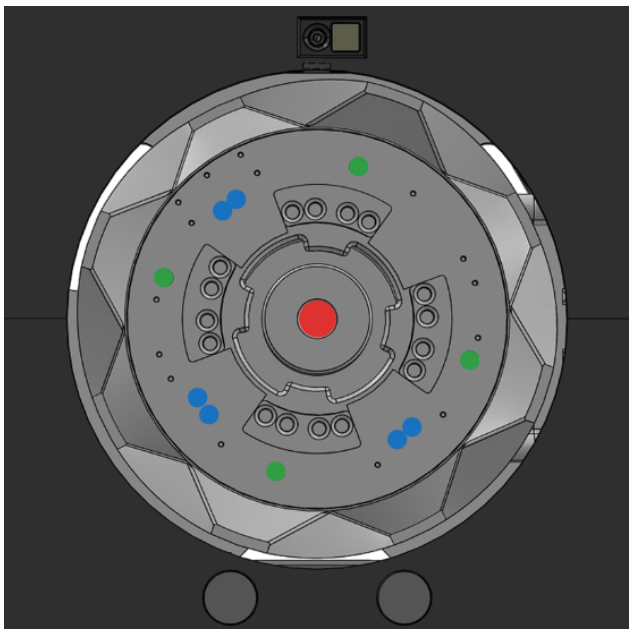


FIG 1. The hole pattern is designed to be rotationally asymmetric via a single angular gap in the ring; the keypoints are split into three groups. Red: Center keypoint. Green: Singular keypoints. Blue: Paired keypoints

4.1. Capture and Reconstruction

Video capture was performed handheld around the CAREER docking mockup using an iPhone 15 main camera (26 mm-equivalent lens, $f/1.6$), recording vertical 4K video (2160×3840) at approximately 60 fps. The operator followed a slow helical trajectory around the interface, gradually shifting vertically to cover the full object.

We reconstruct the scene using NerfStudio's `splatfacto` [7] implementation of 3DGS. Image–pose pairs for reconstruction are obtained via NerfStudio's `ns-process-data` CLI, which runs COLMAP [10] to estimate camera poses and undistorted frames. The reconstructed splat is cropped to remove background geometry and axis-aligned to define a stable object coordinate frame $\mathcal{F}_{\text{obj}}$. Metric scale is fixed by aligning the splat to a known physical dimension of the interface (e.g., the outer ring diameter), yielding a scale factor s_{obj} that maps renderer units to metres.

4.2. View Sampling and Stereo Synthesis

The 3D representation is intentionally factorized to 11 keypoints splat files alongside an object splat file, to enable automatic keypoint labeling. From this representation, we synthesize novel-view stereo samples and export:

1) Stereo RGB image pairs.
2) Semantic masks of the front face and labeled keypoints.
3) Per-pixel depth maps.
4) Camera intrinsics, extrinsics, and the ground-truth rigid transform of the object.

Training and test data are generated with different sampling strategies:

- Training data are sampled independently and randomly.
- Test data are generated as temporally coherent sequences.

The camera bounds is generally limited to a front conical distribution of views. The object is rotated randomly on its Z-axis in both sets, which ultimately is the main goal we are trying to predict. Furthermore, backgrounds are composited after geometric rendering using the per-pixel depth and alpha masks. We used three categories of backgrounds: photographic textures, procedural patterns (gradients, Perlin-like noise), and environment images of realistic lab scenes.

5. MODEL TRAINING

The synthetic data described in Sec. 4 are used to train two networks that support keypoint localization: a lightweight detector to localize and crop the docking interface, and a heatmap regressor to predict keypoint positions within the cropped region.

5.1. Interface Detector (YOLOv8)

Processing full-resolution frames (1280×720) is inefficient because the docking interface typically occupies a small region. We therefore fine-tune a single-class YOLOv8-seg detector [11], initialized from COCO-pretrained weights, to localize the front face of the interface and produce a segmentation mask. The resulting mask is used to crop images to 256×256 before keypoint prediction.

Training data are generated automatically from the synthetic renders. From the rendered masks and 3D keypoints, we derive image–label pairs with bounding-box and segmentation annotations and split them into training and validation sets using an 80/20 split and apply a random translation around the image space coordinates. Manual annotation is not required. We train for 300 epochs with early stopping patience 25. The detector thus provides a fully automated interface crop that removes most background clutter prior to keypoint estimation.

5.2. Keypoint Heatmap Regressor

Our pipeline requires predicting the image-space locations of the interface holes within the cropped region. The holes are visually indistinguishable at the pixel level; their identity is defined solely by their position within the global ring pattern. A network that assigns a distinct output channel to each hole therefore cannot rely on local appearance cues alone. Instead, the receptive field must encompass the entire interface at a resolution where the holes are resolved, so

©2026

that each channel can be disambiguated by its context within the full pattern. This requirement favors backbones with global receptive fields.
VMamba [1] provides such global context at linear complexity in the number of tokens, unlike self-attention-based transformers, which scale quadratically. At 256×256 resolution, this difference is significant and makes a VMamba-Tiny backbone(ImageNet-pretrained, with the FPN-style lateral fusion head) the more efficient choice for realtime requirements.

We opt for using a heatmap head that predicts a Gaussian per hole, over a segmentation head, which predicts a pixel-wise mask per hole. The segmentation formulation suffers from severe foreground–background imbalance and cannot disambiguate visually identical holes from local appearance alone.
We therefore formulate keypoint localization as heatmap regression. Given a cropped interface image $\mathbf{x} \in \mathbb{R}^{3\times256\times256}$, the network predicts

$$\mathbf{H} = f_\theta(\mathbf{x}) \in \mathbb{R}^{C\times256\times256}, \tag{3}$$

where C denotes the number of modelled keypoints.

Each output channel is assigned to one predefined keypoint identity and is therefore semantically distinct from all other channels. The spatial maximum of channel i defines the predicted image location of keypoint i. This design choice embeds the identity of the keypoint within the network's output channel, which allows us to skip cross-view pairing techniques such as descriptor matching and epipolar search.
During inference, the network logits are sigmoid-activated and passed to the subsequent heatmap-processing stage for peak extraction, confidence assessment, and geometric validation.

We train over 2000 samples with an 80/20 validation split, up to 200 epochs with patience 30. Each training sample consists of an image and a stack of C class-specific Gaussian heatmaps. We utilize an AdamW optimizer and an increased head learning rate to allow it to catch up to the pretrained backbone ($LR_{backbone} = 3e^{-5}, LR_{head} = 3e^{-3}$).
The training objective combines four terms:

$$\mathcal{L} = \lambda_{\mathrm{AW}}\mathcal{L}_{\mathrm{AW}} + \lambda_{\mathrm{MSE}}\mathcal{L}_{\mathrm{MSE}} + \lambda_{\mathrm{CC}}\mathcal{L}_{\mathrm{CC}} + \lambda_{\mathrm{S}}\mathcal{L}_{\mathrm{S}}, \tag{4}$$

comprising an Adaptive Wing term, a mean-squared error term, a distance-aware cross-channel term, and a peak-sharpness term. The full expressions and design rationale are given in Appendix A.

6. PIPELINE ARCHITECTURE

The runtime pipeline consumes synchronized stereo images and produces a time series of 6-DoF poses of the docking interface in the left camera frame. The two learned modules described in Sec. 5 are used as fixed, pretrained components within a larger geometric processing chain.

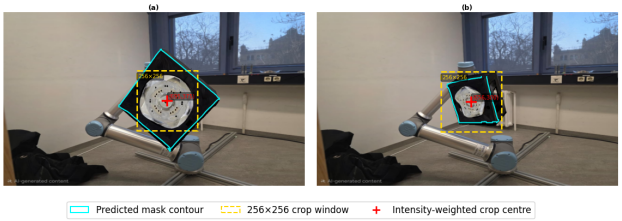


FIG 2. Interface detector cropping to the relevant parts of the image in a) a head-on view and b) oblique angle

6.1. Model predictions

Given a synchronized stereo pair $(I_{\mathrm{L}}, I_{\mathrm{R}})$, and a single instance assumption, the first stage inputs both views into the YOLOv8 detector. For each image, the detector outputs a binary foreground mask of the interface, which is used to suppress background pixels prior to keypoint prediction, and the crop centre is defined as the mask centroid (Fig. 2).
The detection confidence threshold is set intentionally low (0.1). Missed detections can cost us the entire frame, on the other hand spurious detections can be filtered out in the later steps.
Cropping both views to a fixed 256×256 window without resizing induces a pure principal-point shift in the camera intrinsics: the focal lengths f_x, f_y remain unchanged, and only the principal point (c_x, c_y) is offset by the crop translation. Consequently, poses estimated in the cropped coordinate system are directly valid in the original full-frame camera frame after applying this principal-point offset, with no additional scaling required.
The cropped and masked stereo pair is then passed as a batch to the VMamba keypoint regressor. For each view, the network produces a heatmap tensor with $C = 11$. Channel i encodes the model confidence of keypoint i at each pixel location in the cropped coordinate system.

6.2. Heatmap Processing

First, a per-channel background floor is removed, followed by clipping to non-negative values and normalization by the channel maximum.
For each keypoint channel, the processor first identifies local maxima and selects the strongest one as the primary peak. The peak location is refined to subpixel accuracy by computing a weighted centroid within a small window around the argmax. Around each refined peak, a local spread σ and a confidence score are estimated from the heatmap shape. The module also retains a small set of runner-up candidates per channel, obtained from the next-strongest local maxima and refined in the same way. These alternative peaks are not used immediately but are

©2026

exposed to the downstream correspondence-filtering stage, which can promote them if the primary assignment is inconsistent with stereo geometry or temporal continuity. Additional checks reject detections that duplicate another channel's location within a small radius using cross-channel non-maximum suppression.

6.3. Correspondence Filtering

Our model provides us with its best correspondence candidate per channel C, alongside the following candidates for those claims. However, that claim is occasionally unreliable due to oblique views with distorted geometry, which can lead the model to produce non-confident or even confidently wrong answers. The matcher's job is therefore not detection. It is: given candidate detections and a known planar 3D model, decide which detection is which model point, jointly for both views, using geometry rather than the network's claim.

We first fit an ellipse to the detections in each view with a RANSAC-robustified [12] direct conic fit and whiten by it, which restores an angular coordinate that is monotone in the true ring angle under arbitrary perspective.
We then enumerate every cyclic-order-preserving assignment of detections to model slots by dynamic programming scoring each with the mean of the left and right angular residuals so that a single labelling is evaluated in both images. Cyclic order is preserved exactly under any homography, so the constraint is exact at any obliquity.
The leading hypotheses are re-scored with the exact plane-to-image homography reprojection error in both views, augmented by the face-centre keypoint as a free coplanar constraint, and combined with a soft temporal-continuity prior on the object's spin.
Finally, alternative heatmap peaks are substituted where doing so lowers the refitted reprojection residual, recovering channels whose leading peak landed on a neighbouring hole. The output is a single shared model-index set with the corresponding 2D points in both views, so stereo consistency is structural rather than enforced post hoc (Appendix.B).

6.4. Stereo Pose Estimation

Given a labelled set of corresponding 3D rest-frame keypoints and left–right image coordinates, the pose estimator computes the rigid transformation from the object rest frame to the left camera frame. When a temporally propagated pose is available, it is included both as an iterative PnP warm start and as a direct prediction hypothesis, alongside the given IPPE provided hypotheses.
Before refinement, hypotheses are filtered using geometric visibility constraints. The object must lie in front of the camera, and the estimated interface normal must face the camera according to the known front-face normal. This removes back-facing and physically invalid solutions.

Algorithm 1 Correspondence filtering

Require: $\{\mathbf{p}_i^L\}, \{\mathbf{p}_i^R\}, \{\boldsymbol{\phi}_p\}_{p=1}^M, \delta_{t-1}$
$(\mathbf{c}^v, \mathbf{T}^v) \leftarrow \textsc{ConicFit}(\{\mathbf{p}_i^v\}), \quad v \in \{L, R\}$
$\theta_i^v \leftarrow \angle[\mathbf{T}^v(\mathbf{p}_i^v - \mathbf{c}^v)]$
$\mathcal{H} \leftarrow \textsc{CyclicDP}(\theta^L, \theta^R, \boldsymbol{\phi})$
$\mathcal{H}_k \leftarrow \mathrm{Top}_k(\mathcal{H})$
for $h \in \mathcal{H}_k$ **do**
    $s(h) \leftarrow \textsc{Score}(h, \delta_{t-1})$
end for
$h_1 \leftarrow \arg\min_{h \in \mathcal{H}_k} s(h)$
repeat
    $h_1 \leftarrow \textsc{Refine}(h_1)$
until $\Delta s < \varepsilon$
return $\{(h_{1,i}, \mathbf{p}_i^L, \mathbf{p}_i^R)\}$

Notation:
v view; $\mathcal{H}_k$ top-k hypotheses; $s(h)$ hypothesis score; δ_{t-1} prior spin; model ring $\{\boldsymbol{\phi}_p\}_{p=1}^M$; ε refinement tolerance.

Each surviving hypothesis is refined by minimizing the stereo reprojection residuals jointly in both views. Let $\boldsymbol{\xi} = (\boldsymbol{\omega}, \mathbf{t})$ denote the rotation-vector and translation parameters, and let π_{L} and π_{R} denote the left- and right-camera projection functions. The residual vector is

$$\mathbf{r}(\boldsymbol{\xi}) = \begin{bmatrix} w_{\mathrm{L},i}\,(\pi_{\mathrm{L}}(R\mathbf{X}_i + \mathbf{t}) - \mathbf{u}_{\mathrm{L},i}) \\ w_{\mathrm{R},i}\,(\pi_{\mathrm{R}}(R\mathbf{X}_i + \mathbf{t}) - \mathbf{u}_{\mathrm{R},i}) \end{bmatrix}_{i=0}^{N}, \tag{5}$$

where $\mathbf{X}_i$ is a rest-frame keypoint, $\mathbf{u}_{\mathrm{L},i}$ and $\mathbf{u}_{\mathrm{R},i}$ are its measured image coordinates, and

$$w_{\mathrm{v},i} = \frac{c_{\mathrm{v},i}}{\max(\sigma_{\mathrm{v},i}, 0.25)}, \qquad v \in \{L, R\} \tag{6}$$

are confidence based weights obtained from heatmap processing. The parameters are optimized with a Levenberg-Marquardt-type least-squares procedure using a Huber robust loss, reducing the influence of remaining outlier correspondences.
The refined hypotheses are ranked using a combined stereo and temporal score:

$$S = e_{\mathrm{L}}^{\mathrm{RMS}} + e_{\mathrm{R}}^{\mathrm{RMS}} + \lambda_{\mathrm{inl}}(N - N_{\mathrm{inl}}) + \lambda_{\mathrm{temp}}\, C_{\mathrm{temp}} \tag{7}$$

where $e_{\mathrm{L}}^{\mathrm{RMS}}$ and $e_{\mathrm{R}}^{\mathrm{RMS}}$ are the weighted reprojection errors over the inliers in the two views, N_{inl} is the number of inliers, and C_{temp} penalizes deviation from the temporally predicted pose. The inlier penalty prevents a hypothesis from achieving a low score by explaining only a small subset of the available correspondences. The hypothesis with the lowest valid score is selected, provided that the minimum number of inliers and reprojection-error requirements are satisfied. The estimator returns the rotation matrix and translation vector in the left camera frame, together with the inlier set, per-view reprojection errors, and an ambiguity margin relative to the second-best hypothesis.

©2026

7. EXPERIMENTAL SETUP

We evaluate the pipeline on synthetic stereo sequences generated from the Gaussian-splat model (Sec. 4). The default test set contains 500 temporally ordered stereo frames. Each frame provides synchronized left/right images, camera parameters, and ground-truth object pose.

Test data and sequences

Images are rendered at 1280×720 pixels with a focal length of 800 renderer units. Viewing conditions are:
- camera radii in $[5, 12]$ renderer units,
- unrestricted azimuth $\phi \in [0°, 360°)$,
- polar angles $\theta \in [125°, 180°]$,
- object rotations about Z clipped to $[-359°, 359°]$,
- stereo baseline of 1.0 renderer unit.

The camera trajectory is obtained by sampling eight anchor viewpoints and interpolating with a cubic spline. Object rotation follows piecewise-constant angular-velocity segments of 20–70 frames, with speeds uniformly sampled from $2°$–$4°$ per frame and direction changes smoothed over five frames.

Metrics

Keypoint accuracy is reported using mean radial error and PCK. Pose accuracy is quantified using wrapped roll error, geodesic rotation error, and translation norm error in the left camera frame. We additionally report per-view reprojection RMS. All metrics are computed over the full set of attempted frames.

Hardware and timing

Experiments are run on a laptop with an NVIDIA RTX 4070 GPU, Intel i7-13620H CPU, and 16 GB RAM. Timing is measured using wall-clock timestamps around each pipeline stage, excluding GPU kernel initialization. A brief warm-up pass is executed before benchmarking. Reported latencies are averaged over multiple runs to achieve a stable number.

8. RESULTS

8.1. Interface Detection

We evaluated two fine-tuned YOLOv8 detectors for interface localization in FP16 precision. The initial YOLOv8-nano model achieved strong training performance with an average inference time of 35.3 ms per stereo pair, whilst YOLOv8-small incurred only a modest increase in latency 57.4 ms per pair.
The validation split for the results in table 1 is a random 80/20 split of the same 2000 renders from the same splat. Although it shows that the models are somewhat equal in performance and saturated around 0.995 mAP50, when deployed on the target hardware, we observed a substantial drop in detection rate due to the synthetic-to-real domain shift. On a 300-frame real test set, the nano model successfully detected the interface in only 38% of frames, compared to 97.3% for the small variant. Consequently, we adopted YOLOv8-small, which offers greater capacity to adapt to real imagery, while remaining well within our 1-2 Hz computational goal the development GPU; deployment on the target edge platform is not yet demonstrated.
The performance on the test set shows us 100% on both the interface detection rate and the keypoint inclusion (inside of the crop) rate.

TAB 1. YOLOv8 interface detector performance on synthetic validation data. Precision is reported at the default Ultralytics threshold (confidence 0.25, IoU 0.5).

Model	Precision	mAP_{50}	$mAP_{50–95}$
YOLOv8n(bbox)	0.997	0.995	0.956
YOLOv8s(bbox)	1.000	0.995	0.989
YOLOv8n(mask)	0.998	0.995	0.932
YOLOv8s(mask)	1.000	0.995	0.967

8.2. Keypoint detection

8.2.1. Per-channel accuracy

Table 2 reports accuracy for all eleven channels over the $11{,}000$ scored instances (1000 views $\times$ 11 channels.
Aggregated over all channels the stage attains an MRE of 0.98 px with a median of 0.47 px and a 95th percentile of 2.03 px. The gap between mean and median shows that the mean is carried by a small tail rather than by a broad loss of precision: 94.6% of instances fall within 2 px and 98.4% within 5 px. The latency of the heatmap inference and the subsequent postprocessing steps are by far the most dominant compute time requirements (Table 3) taking roughly 250ms combined per pair of images.
The centre channel is the most accurate of the eleven (0.45 px MRE, 100% PCK@2), which is consistent with its support being roughly $3.5\times$ larger than that of a hole (57.4 vs. 13–18 px on average). This matters operationally, since the centre is the PnP anchor and the origin of the reported object axes.
Among the ring channels, hole 6 (88.3% PCK@2), hole 9 (89.2%) and hole 2 (92.1%) are the hardest, but they fail in two distinguishable ways. Hole 6 has one of the *best* medians (0.41 px) alongside the worst tail ($p_{95} = 4.68$ px), i.e. it is normally localised as well as any channel and occasionally lands far away. Hole 2, in contrast, has the worst median (0.72 px) and a moderate tail ($p_{95} = 2.22$ px), producing the lowest PCK@1 in the table (63.9%): it is consistently slightly wrong rather than occasionally very wrong, which is the signature of a systematic sub-pixel bias. Hole 6 carries the largest MRE (1.62 px) on a median of 0.51 px and is again tail-dominated.

©2026

Hardness is not explained by how much of each hole is visible. Across the ten ring channels, Pearson's correlation coefficient between mean ground-truth hole area and MRE is $r = -0.08$, and between area and PCK@2 it is $r = +0.11$; both are negligible. Foreshortening and apparent hole size are therefore not the mechanism behind the per-channel spread.

8.2.2. Dependence on viewing angle

Table 4 resolves accuracy against the polar angle θ of the camera that produced each view, recovered from the per-frame ground-truth extrinsics. Left and right views are binned independently.

Localisation degrades systematically as the face turns away from the camera, but the tail degrades roughly twice as fast as the centre of the distribution: the median error rises from 0.32 px in the $(170°, 175°]$ bin to 0.73 px in the most oblique $(125°, 130°]$ bin, a factor of 2.3, while p_{95} rises from 0.63 px to 3.06 px, a factor of 4.9. Oblique viewing therefore does not blur the estimate uniformly; it raises the rate at which individual channels fail outright, which is the same failure mode identified per-channel in Section 8.2.1. Beyond $\approx 145°$ the median saturates at 0.31–0.35 px, a floor imposed by the centroid definition of the ground truth and the refinement window rather than by the network.

The trajectory samples angles unevenly: 66% of views lie at $\theta \leq 140°$ and 86.5% at $\theta \leq 155°$. The aggregate figures in Table 2 are consequently weighted towards oblique views and are conservative relative to a uniformly sampled trajectory.

8.3. Ablation: Correspondence filtering

We built our proposed correspondence filtering to assist our keypoint labeling scheme in making it more geometrically accurate and consistent across the stereo pair. To justify our decisions, we performed an ablation study on our full test set (n=500) alongside a new hard test set (n = 500) which includes only oblique views $\theta < 145°$: only oblique views near the edge.

The ablation results (Table 5) show that most components of our design contribute only marginally on the full test set, which resembles smooth, continuous orbital motion around the interface. On the hard subset, however, which is constructed specifically to stress the labeling under near-failure viewing conditions, the picture changes: elliptical whitening and temporal continuity emerge as clearly important, while the remaining mechanisms have a comparatively small effect on labeling accuracy. Overall, the full method still yields a net improvement in accuracy, at negligible additional compute cost.

8.4. Pose Accuracy

The full pipeline returns a pose on 497 of 500 test frames (99.4%); all accuracy figures below are over this set. Rotation about the docking (Z) axis has a median error of $0.28°$ (mean $0.36°$, p95 $1.02°$, max $3.45°$), and the full-rotation geodesic error has a median of $1.14°$ (mean $1.26°$). Translation error is subcentimetre at the median (2.53 mm, mean 2.69 mm, p95 5.27 mm), with a small systematic bias along the camera x-axis (-1.34 ± 1.08 mm) and negligible bias on y/z. Reprojection residuals remain under one pixel in both views (left: 0.72 px median; right: 0.77 px median), and the matcher retains a mean of 10.7 of 11 keypoints per frame, indicating that most of the ring geometry survives to the PnP stage rather than being discarded by the matcher's guards.

9. LIMITATIONS

The evaluation is carried out entirely on synthetic renderings of a single interface, generated from the same Gaussian splat used to train both networks. Test views are held out and unseen, but they are not an independent domain.

Sensor noise, motion blur, specular reflection from the metallic interface, and the harsh directional illumination of the orbital environment are all absent from the renders.

The translation accuracy falls short of the 1 mm requirement, and the error is partly systematic: the -1.34 ± 1.08 mm bias along the camera x-axis indicates a scale or baseline effect rather than random localisation noise. Rotation about the docking axis meets the 0.5° requirement at the median but not at the 95th percentile, and the full-attitude geodesic error does not meet it at all, which is expected for a near-planar target whose out-of-plane tilt is weakly observable.

No comparison against an alternative backbone was performed, so the choice of VMamba is argued from receptive-field requirements rather than measured against a convolutional baseline.

Heatmap post-processing, at 145 ms, is the largest single cost and is an unoptimised implementation rather than a fundamental limit.

Finally, the runtime figures were obtained on a laptop RTX 4070 rather than the intended Jetson-class platform, so the 2.2 Hz result does not yet demonstrate that the 2 Hz requirement is met on target hardware.

10. CONCLUSION AND FUTURE WORK

We presented a stereo, markerless, CAD-free pipeline that estimates the 6-DoF pose of the iSSI docking interface from synchronised image pairs. The entire training set is generated from a 3D Gaussian-splat reconstruction of the physical hardware, with keypoint labels derived automatically from a factorised splat rather than from manual annotation or a CAD model. Identity is assigned to visually identical holes not by the network alone but geometrically, through a cyclic-order-preserving assignment evaluated jointly in both views and re-scored by the exact plane-to-image homography.

©2026

TAB 2. Per-channel accuracy on the stereo test sequence (500 pairs, 1000 views, 1000 scored instances per channel). MRE, median and p_{95} are in pixels over accepted detections; PCK@τ is the percentage of instances both accepted and localised within τ px, so a rejected channel counts as a miss. $\text{MRE}_{\text{argmax}}$ is the raw per-channel arg-max without sub-pixel refinement or post-processing. "Area" is the mean number of ground-truth mask pixels per instance. C is the face centre keypoint.

Ch.	Valid (%)	MRE	Median	p_{95}	PCK@1	PCK@2	PCK@5	$\text{MRE}_{\text{argmax}}$	Area
0	100.0	0.45	0.38	1.05	94.4	99.6	100.0	0.62	13.3
1	99.8	0.65	0.39	1.10	93.7	98.7	99.1	0.85	16.1
2	100.0	1.03	0.72	2.22	63.9	92.1	99.4	1.16	13.3
3	100.0	1.22	0.49	2.47	77.3	93.4	98.8	1.37	13.0
4	98.9	0.93	0.46	1.38	90.1	95.7	98.0	1.48	16.0
5	99.7	1.62	0.51	2.64	82.2	93.3	96.9	1.97	15.2
6	100.0	1.36	0.41	4.68	76.4	88.3	95.3	1.49	16.6
7	99.3	0.84	0.64	1.88	78.6	95.2	98.4	1.30	16.2
8	99.5	0.89	0.38	1.74	88.3	95.5	97.9	1.17	17.6
9	100.0	1.36	0.58	2.85	73.2	89.2	98.5	1.47	13.2
C	100.0	0.45	0.41	0.93	96.5	100.0	100.0	0.61	57.4
All	**99.75**	**0.98**	**0.47**	**2.03**	**83.2**	**94.6**	**98.4**	**1.23**	**18.9**

TAB 3. Per-stage latency, mean over 500 frames, averaged over 5 runs.

Stage	ms/frame	Share
Heatmap post-processing	145.2	31.5%
Heatmap inference	109.4	23.7%
PnP estimation	78.3	17.0%
Interface detection	57.5	12.5%
Frame loading	38.7	8.4%
Correspondence filtering	15.6	3.4%
Preprocessing	15.3	3.3%
Evaluation / logging	1.3	0.3%
Total	**461.3**	**100.0%**

TAB 4. Accuracy of ring holes prediction as a function of camera polar angle θ (180° = face-on). "Views" counts monocular views per bin and "Det." the accepted detections scored in that bin. Median and p_{95} are keypoint errors in pixels.

θ bin (deg)	Views	Det.	Median (px)	p_{95} (px)
$(125, 130]$	114	1238	0.73	3.06
$(130, 135]$	352	3860	0.56	1.69
$(135, 140]$	194	2134	0.49	1.27
$(140, 145]$	101	1111	0.39	1.01
$(145, 150]$	63	693	0.35	0.84
$(150, 155]$	41	451	0.35	0.70
$(155, 160]$	40	440	0.31	0.65
$(160, 165]$	43	473	0.34	0.63
$(165, 170]$	31	341	0.34	0.68
$(170, 175]$	21	231	0.32	0.63
Total	1000	9972	0.47	—

On a 500-frame synthetic stereo sequence the pipeline returns a pose with an improved error rate over the previous attempts at CAREER. However, major improvements, domain transitions and explicit calibration of the metric scale of the splat are a necessity. Furthermore, deployment and optimisation on Jetson-class hardware, with ONNX/TensorRT export and a vectorised reimplementation of the heatmap post-processing that currently dominates the frame budget. Finally, harder evaluation: partial occlusion, extreme and directional lighting, longer stand-off ranges, and ultimately closed-loop evaluation in which the estimated pose commands the manipulator and success is measured by mating rather than by pose error.

Contact address:

awsam.agbaria@campus.tu-berlin.de

References

[1] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. In *Advances in Neural Information Processing Systems*, volume 37, pages 103031–103063, 2024. DOI: 10.52202/079017-3273.

[2] 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*, pages 4556–4565, 2019. DOI: 10.1109/CVPR.2019.00469.

©2026

TAB 5. Ablation of the stereo ring matcher on 500 test frames. Each row disables one mechanism of the proposed method while holding all others fixed.

	Latency	Label. acc. (%)	
Removed step	(ms/frame)	Test	Hard
Elliptical fitting	12.94	96.55	89.42
Joint cost	16.27	95.98	93.33
Homography rescoring	7.98	94.67	94.31
Temporal continuity	14.80	94.26	90.87
Runner-up peak polish	11.10	96.46	94.89
Proposed (full)	15.04	**96.65**	**95.22**

[3] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. GDR-Net: Geometry-guided direct regression network for monocular 6D object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16611–16621, 2021. DOI: 10.1109/CVPR46437.2021.01634.

[4] Le Jin, Guoshun Zhou, Zherong Liu, Yuanchao Yu, Teng Zhang, Minghui Yang, and Jun Zhou. IRPE: Instance-level reconstruction-based 6D pose estimator. *Image and Vision Computing*, 154:105340, 2025. DOI: 10.1016/j.imavis.2024.105340.

[5] Suyi Liu, Fang Xu, Chengdong Wu, Jianning Chi, Xiaosheng Yu, Longxing Wei, and Chuanjiang Leng. CMT-6D: A lightweight iterative 6DoF pose estimation network based on cross-modal transformer. *The Visual Computer*, 41(3):2011–2027, 2025. DOI: 10.1007/s00371-024-03520-1.

[6] Shaobo Zhang, Wanqing Zhao, Ziyu Guan, Wei Zhao, Jinye Peng, and Jianping Fan. Learning cross-view consistent 3D keypoints for object 6D pose estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(7):6816–6831, 2025. DOI: 10.1109/TCSVT.2025.3538018.

[7] Jiguang Zhang, Zhaohui Zhang, Xuxiang Feng, Shibiao Xu, Rongtao Xu, Changwei Wang, Kexue Fu, Jiaxi Sun, and Weilong Ding. C2Fi-NeRF: Coarse to fine inversion NeRF for 6D pose estimation. *Expert Systems with Applications*, 264:125837, 2025. DOI: 10.1016/j.eswa.2024.125837.

[8] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, 2023. DOI: 10.1145/3592433.

[9] Christian Helms. CAD-free and lightweight 6DoF pose estimation of the iSSI interface using a stereo camera as part of the CAREER project. Space robotics written report, Technische Universität Berlin, Institute of Aeronautics and Astronautics, 2025.

[10] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. DOI: 10.1109/CVPR.2016.445.

[11] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8, 2023.

[12] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. DOI: 10.1145/358669.358692.

[13] Xinyao Wang, Liefeng Bo, and Fuxin Li. Adaptive wing loss for robust face alignment via heatmap regression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6970–6980, 2019. DOI: 10.1109/ICCV.2019.00707.

[14] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment — a modern synthesis. In *Vision Algorithms: Theory and Practice*, volume 1883 of *Lecture Notes in Computer Science*, pages 298–372. Springer, 2000. DOI: 10.1007/3-540-44480-7$_2$1.

A. LOSS FUNCTION DETAILS

This appendix provides the full definitions of the loss components used for Vmamba in Sec. 5.

A.1. Adaptive Wing loss ($\mathcal{L}_{\mathrm{AW}}$).

This term drives accurate heatmap regression, with stronger emphasis on pixels near the Gaussian peaks where small errors most affect keypoint localization. Let $\mathbf{H}$ and $\mathbf{H}^*$ denote the predicted and target heatmaps, respectively. The Adaptive Wing loss [13] is applied element-wise and then averaged, with an additional weighting factor $(1 + 4\mathbf{H}^*)$ that increases the contribution of high-value target pixels:

$$\mathcal{L}_{\mathrm{AW}} = \mathrm{mean}\left[(1 + 4\mathbf{H}^*) \odot \ell_{\mathrm{AW}}(\mathbf{H}, \mathbf{H}^*)\right]. \tag{8}$$

For a single pixel with prediction p and target value $y \in [0, 1]$, let

$$d = |p - y| \tag{9}$$

The unweighted Adaptive Wing penalty is defined as

$$\ell_{\mathrm{AW}}(p,y) = \begin{cases} \omega \log\left(1 + \left(\frac{d}{\epsilon}\right)^{\alpha-y}\right), & d < \theta, \\ A(y)d - C(y), & d \geq \theta, \end{cases} \tag{10}$$

where the slope and offset for the linear branch are

$$A(y) = \frac{\omega(\alpha-y)(\theta/\epsilon)^{\alpha-y-1}}{\epsilon\left[1+(\theta/\epsilon)^{\alpha-y}\right]}, \tag{11}$$

and

$$C(y) = \theta A(y) - \omega \log\left(1+(\theta/\epsilon)^{\alpha-y}\right). \tag{12}$$

The hyperparameters α, ω, ϵ, and θ control the shape of the loss and the transition between the logarithmic and linear regimes.

A.2. Distance-aware cross-channel loss ($\mathcal{L}_{\mathrm{CC}}$).

Each output channel corresponds to a unique keypoint identity. To discourage channel i from activating near the target peak of a different channel j, we introduce a cross-channel penalty that is active only outside a safe neighbourhood of channel i's own peak. Let $\mathbf{q}_i$ denote the target image location of keypoint i. We define:

$$G_j(x,y) = \exp\left(-\frac{\|\mathbf{u}-\mathbf{q}_j\|^2}{2\sigma^2}\right) \tag{13}$$

as a soft neighbourhood around keypoint j, and

$$M_i(x,y) = 1 - \exp\left(-\frac{\max\left(0, \|\mathbf{u}-\mathbf{q}_i\| - r\right)^2}{2\sigma^2}\right), \quad r = \kappa\sigma \tag{14}$$

as a mask that suppresses the penalty near keypoint i. The cross-channel loss is then:

$$\mathcal{L}_{\mathrm{CC}} = \frac{1}{C(C-1)} \sum_{i \neq j} \mathrm{mean}_{x,y}\left[H_i(x,y)\, M_i(x,y)\, G_j(x,y)\right]. \tag{15}$$

This formulation tolerates natural overlap between neighbouring Gaussian targets while discouraging confusion between distinct keypoint channels, and it utilizes the same Gaussian scale σ as the target heatmaps. The safe radius is set as $r = \kappa\sigma$, with $\kappa > 1$ controlling how strongly activations near a channel's own peak are protected from the cross-channel penalty.

A.3. Peak-sharpness loss ($\mathcal{L}_{\mathrm{S}}$).

The final term encourages each channel to concentrate its activation around its own target peak rather than spreading energy diffusely. For channels whose target peak exceeds a presence threshold (set $\mathcal{P}$ of present keypoints), we penalize the fraction of predicted activation outside a Gaussian neighbourhood of the target:

$$\mathcal{L}_{\mathrm{S}} = \frac{1}{|\mathcal{P}|} \sum_{i \in \mathcal{P}} \frac{\sum_{x,y} H_i(x,y)\left[1 - \exp\left(-\frac{\|\mathbf{u}-\mathbf{q}_i\|^2}{2(4\sigma)^2}\right)\right]}{\max\left(\sum_{x,y} H_i(x,y),\, 0.5\right)}. \tag{16}$$

Together, these terms yield heatmaps that are accurate near the peaks, globally consistent, semantically separated across channels, and spatially concentrated.

The peak-sharpness loss operates only on channels deemed present, i.e., those whose target maximum exceeds a threshold $y_{\min}$. This prevents penalizing channels that are intentionally inactive in a given sample. The neighbourhood width for the "inside" region is set to 4σ, encouraging predictions to be tightly concentrated around the target while allowing smooth Gaussian-like profiles.

B. CORRESPONDENCE FILTERING

This appendix provides the full definitions of the components in the Stereo correspondence filtering algorithm used in Sec. 6.3.

Whitening. To remove the anisotropic part of the projective distortion. We fit an ellipse to the set of detected hole centres with a direct conic fit, following the total-least-squares formulation of Fitzgibbon [14]: each point contributes a row of the design matrix

$$\left[x^2,\ xy,\ y^2,\ x,\ y,\ 1\right], \tag{17}$$

The fitted conic is converted from its algebraic coefficients into a centre $\mathbf{c}$ and a symmetric positive-definite matrix $\mathbf{T}$ satisfying ($\|\mathbf{T}(\mathbf{p}-\mathbf{c})\| = 1$ on the fitted ellipse):

$$\theta_i = \mathrm{atan2}\big([\mathbf{T}(\mathbf{p}_i-\mathbf{c})]_y,\ [\mathbf{T}(\mathbf{p}_i-\mathbf{c})]_x\big). \tag{18}$$

Cyclic-order constraint. A labelling π assigning detection i to model slot $\pi(i)$ is admissible only if, in detection cyclic order,

$$\pi(i_1) < \pi(i_2) < \cdots < \pi(i_n) \pmod M, \tag{19}$$

which a planar homography preserves exactly at any obliquity; violating assignments are excluded from the dynamic program rather than penalised.

Joint stereo DP cost. For detection i and model slot p, with $\Delta\theta_i^v, \Delta\phi_p$ the angular offsets from the seam detection/slot in view v:

$$c(i,p) = \tfrac{1}{2}\Big(\big|\angle(\Delta\theta_i^L - \Delta\phi_p)\big| + \big|\angle(\Delta\theta_i^R - \Delta\phi_p)\big|\Big). \tag{20}$$

Hypothesis score. With homography RMS reprojection error rms^v (face centre included as a free coplanar point), missing-detection count n_m, and spin estimate δ_h predicted from δ_{t-1}:

$$s(h) = \tfrac{1}{2}\left(\mathrm{rms}_h^L + \mathrm{rms}_h^R\right) + \lambda_a\, \bar{c}_h + \lambda_m\, n_m + \lambda_t\, \big|\delta_h - \hat{\delta}_{t-1}\big|. \tag{21}$$

©2026