GUIDE: SELF-CONSISTENT GEOMETRIC DECEPTION IN SPATIAL FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatial foundation models jointly predict camera poses, depth, and 3D points from shared visual features, making geometric consistency a useful signal for validating their predictions. However, consistency among these outputs does not necessarily imply correctness, as coordinated errors may remain mutually compatible and pass internal geometric checks. We propose GUIDE (Geometric Untruths In Dependent Estimates), the first attack that systematically constructs self-consistent geometric deception against spatial foundation models. Specifically, GUIDE first specifies a target camera trajectory and constructs shared world anchors under the target geometry. It then projects these anchors across views to derive compatible depth and point targets. Finally, it optimizes bounded image perturbations to jointly steer the model toward these prescribed geometric predictions. We evaluate GUIDE on three models and five datasets. GUIDE achieves 29/36 joint successes, compared with 12/36 for a camera-only attack. We further evaluate its downstream impact on depth-label generation and robot control, where GUIDE leads to 103/144 accepted erroneous depth predictions and reduces task success by 39.5 and 37.4 percentage points for FALCON and R3DP, respectively.

1 INTRODUCTION

Spatial foundation models are becoming a central component of 3D scene understanding. Given one or more images, these models jointly predict camera poses, depth, 3D points, and spatial features from shared visual representations (Wang et al., 2025a; 2024; Sun et al., 2021; Wang et al., 2025b; Fang et al., 2025). Their geometric outputs support a growing range of applications, including 3D reconstruction, automatic geometric annotation, and embodied control (Asim et al., 2025). Since these predictions describe the same underlying scene, geometric consistency across views and output modalities provides a natural signal for validating model predictions (Paul et al., 2026; Yao et al., 2018). This form of validation is particularly important in automated pipelines, where ground-truth geometry is often unavailable and predicted geometry may be directly reused for supervision or downstream decision making.

Geometric consistency alone, however, is insufficient to establish correctness (Paul et al., 2026; Asim et al., 2025). Camera poses, depth, and 3D points are produced from shared visual features, creating dependencies among their prediction errors (Fang et al., 2025; Wang et al., 2025a). A perturbation can therefore induce coordinated changes across multiple outputs while preserving the geometric relations among them. Such predictions may remain self-consistent even when they describe an incorrect scene. This creates a failure mode for validation mechanisms that rely on relations among the model's own geometric outputs (Yin et al., 2022). Existing geometric attacks have primarily studied degradation of depth, visual odometry, or reconstruction quality (Wong et al., 2020; Nemcovsky et al., 2022; Xie et al., 2023; Jiang et al., 2024; Horváth & Józsa, 2023), leaving open whether an attacker can deliberately construct a prescribed geometric error that remains internally consistent and passes geometric validation (Song et al., 2026; Wang et al., 2026b). This raises a central question: *can an attacker induce an incorrect geometric interpretation while preserving the geometric relations used to validate it?*

Our key insight is that coordinated geometric errors can be organized around a shared target geometry. A prescribed camera deformation defines a common geometric interpretation of the scene across views. This shared structure induces compatible constraints for depth and 3D points, allowing mul-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tiple geometric outputs to evolve coherently toward the same erroneous scene interpretation (Yao et al., 2018; Wu et al., 2024; Wang et al., 2024). As a result, the predicted camera poses, depth, and 3D points can continue to support one another under cross-view geometric relations, even as the overall geometry deviates from the true scene.

Based on this insight, we propose GUIDE (**G**eometric **U**ntruths **I**n **D**ependent **E**stimates), the first attack that systematically constructs self-consistent geometric deception against spatial foundation models. GUIDE first specifies a target camera trajectory and constructs shared world anchors under the target geometry. It then projects these anchors across views to derive compatible depth and point targets. Finally, GUIDE optimizes bounded image perturbations to jointly steer the model toward these prescribed geometric predictions. We evaluate attack success using a joint criterion that requires the same returned prediction to attain the prescribed camera target, exhibit a material error relative to an independent scene reference, and remain accepted by a calibrated geometric checker. This criterion directly captures whether an attack produces a prescribed, externally incorrect, and internally self-consistent geometric prediction. (Yin et al., 2022; Horváth & Józsa, 2023; Jiang et al., 2024; Song et al., 2026; Wang et al., 2026b)

Our contributions are threefold:

- We formulate *self-consistent geometric deception* in spatial foundation models and identify a failure mode in geometric validation caused by dependencies among camera, depth, and 3D point predictions.
- We propose GUIDE, which constructs a shared target geometry across views and uses it to jointly steer camera, depth, and 3D point predictions under bounded image perturbations.
- We evaluate GUIDE on three spatial foundation models and five datasets, demonstrating higher joint attack success and downstream impact on depth-label generation and robot control.

2 Related Work

Geometric attacks. Attacks on monocular depth estimation and visual odometry manipulate geometric predictions through bounded input perturbations (Wong et al., 2020; Nemcovsky et al., 2022; Xie et al., 2023). MVAP-G trains a feed-forward generator that produces multi-view perturbations to degrade VGGT geometry, while PoInit-of-View perturbs initialization views and studies transfer across reconstruction systems through rendering quality (Song et al., 2026; Wang et al., 2026b). Targeted multiview scene attacks and perturbation inheritance have also been studied for generalizable NeRFs and multiview NeRF reconstruction (Horváth & Józsa, 2023; Jiang et al., 2024). These methods optimize geometric degradation without a prescribed target or a calibrated acceptance decision. GUIDE instead studies targeted geometric deception, where camera, depth, and 3D point predictions are coordinated around a shared target geometry and evaluated under the joint endpoint $A \wedge B \wedge Q$.

Geometric consistency checks. ADC combines prediction manipulation with evasion of context consistency checks (Yin et al., 2022). Learned reconstruction can also produce internally consistent geometry for unrelated images or inputs unsuitable for reconstruction (Paul et al., 2026). RobustVGGT selects views using VGGT attention or feature similarity and reconstructs the scene from the retained views (Han et al., 2026). These works show that consistency is useful for validation while also exposing settings in which internally coherent predictions may remain unreliable. GUIDE directly studies this failure mode in spatial foundation models by constructing coordinated geometric errors and testing whether they remain accepted under fixed geometric relations, view filtering, and protected observations.

Adaptive attack evaluation. Reliable robustness evaluation requires matching attacker access, optimization budgets, and validation procedures (Athalye et al., 2018; Tramér et al., 2020). Following this principle, we compare attacks under matched targets, perturbation budgets, checker access, and optimization resources. Final success is measured on the same returned output through target attainment, material external error, and checker acceptance.

3 THREAT MODEL

Attack capability. We consider a white-box attacker against a frozen spatial foundation model $f(I) = (T, D, P, H)$. The attacker can perturb the specified multi-view input images and compute gradients through the model, with $\|\delta\|_\infty \leq 8/255$. Model parameters and geometric outputs are not directly modified, and all predictions are recomputed from the perturbed inputs.

Attacker knowledge. The attacker knows the model and the prescribed camera target. Base GUIDE has no access to checker scores, while GUIDE+checker receives the same checker scores and thresholds as the corresponding direct baseline. Independent scene references and protected RGB-D observations are unavailable during optimization.

Attack goal and success. The attacker aims to steer the predicted camera trajectory toward a prescribed target while inducing compatible changes in depth and 3D points. A returned prediction is considered successful only when it attains the target (A), exhibits a material error relative to an independent reference (B), and is accepted by the checker (Q): $J = A \wedge B \wedge Q$. The independent reference is used only for final evaluation.

4 GEOMETRIC UNTRUTHS IN DEPENDENT ESTIMATES (GUIDE)

Overview. GUIDE couples a prescribed camera deformation with shared point-map and depth targets. It first constructs a target camera trajectory, then builds world anchors and projects them across target views to obtain compatible geometric supervision. Finally, it optimizes bounded image perturbations under camera and support objectives. The returned prediction is evaluated for target attainment, external error, and checker acceptance.

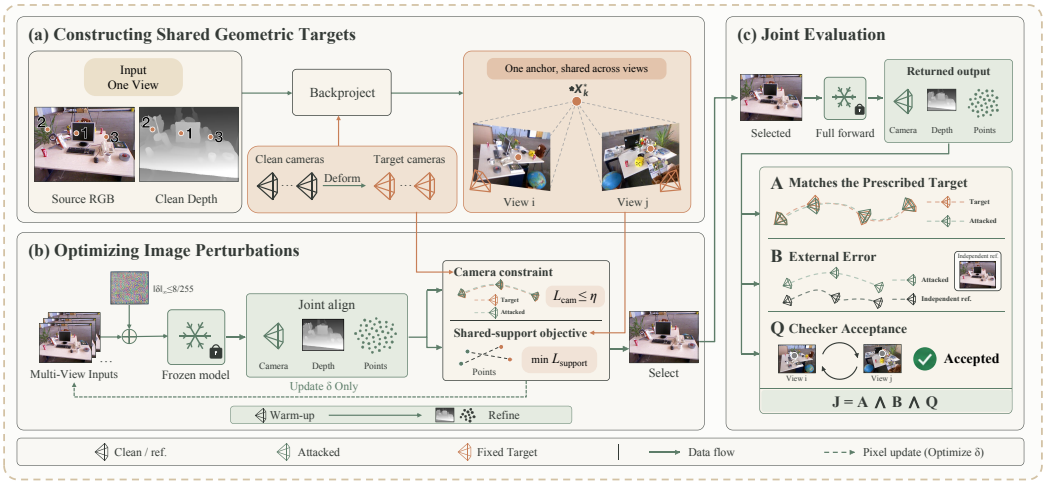


Figure 1: GUIDE constructs shared world anchors and cross-view support targets, optimizes bounded image perturbations, and evaluates the same returned prediction for target attainment, external error, and checker acceptance.

4.1 CONSTRUCTING SHARED GEOMETRIC TARGETS

A frozen spatial model maps an image sequence to $y = f(I) = (T, D, P, H)$. The pose $T_t = [R_t, c_t]$ maps camera coordinates to world coordinates, and depth follows the adapter's camera-axis convention. P denotes the available world-point output, and H denotes spatial features (Wang et al., 2025a; 2024; Fang et al., 2025).

Given the clean camera prediction T_t^0, we define the target camera as

$$T_t^\star = \mathrm{Exp}_{\mathrm{SE}(3)}(\lambda_t \xi)\, T_t^0, \qquad \lambda_t \in [0, 1], \tag{1}$$

where $\xi \in \mathfrak{se}(3)$ is the prescribed endpoint twist and λ_t is the predefined normalized command coefficient for frame t. Here, t indexes frames, while λ_t controls the deformation: $\lambda_t = 0$ recovers the

clean pose and $\lambda_t = 1$ applies the full endpoint deformation. The schedule $\{\lambda_t\}$ is fixed before optimization and shared by all methods. After the sequence gauge is fixed by a single global similarity alignment, the frame-varying command remains in the target trajectory. Simultaneous observations from multiple cameras share a time index.

The camera target specifies the desired trajectory but does not by itself determine the accompanying point-map and depth targets. We construct these targets through shared world anchors, so supervision across views refers to the same three-dimensional entities. A source pixel u_k, its clean depth d_k with gradients stopped, and its source view $s(k)$ define

$$X_k^\star = R_{s(k)}^\star d_k K_{s(k)}^{-1} \tilde{u}_k + c_{s(k)}^\star, \tag{2}$$

where $\tilde{u}_k$ denotes homogeneous pixel coordinates.

Projecting each anchor into the target cameras determines its fixed image location and camera-coordinate depth in each view. Sparse and dense supervision use this same shared geometry, linking the camera targets to the corresponding point-map and depth predictions. Before optimization, valid relations are fixed based on positive depth, image boundaries, occlusion, and correspondence identity.

4.2 Optimizing Image Perturbations

Camera, point-map, and depth scales are aligned to the target coordinate system through the same similarity transform. The camera loss L_{cam} combines normalized camera-center error and mean rotation error. The support loss L_{support} brings predicted points and depths toward the fixed anchors and their projected depth targets. To fit this shared support while maintaining camera feasibility, we optimize

$$\begin{aligned} \min_{\delta} \quad & L_{\mathrm{support}}(\delta) \\ \text{subject to} \quad & \|\delta\|_\infty \le \epsilon, \qquad L_{\mathrm{cam}}(\delta) \le \eta. \end{aligned} \tag{3}$$

The white-box attack uses $\epsilon = 8/255$ and $\eta = 0.08$. We parameterize $\delta = \epsilon \tanh w$ and use Adam with learning rate 0.02 for 200 updates. The first 20% of updates, or 40 updates, are devoted to camera warm-up. Sparse point-map and depth weights are 1 and 0.5, respectively; dense support has relative weight 0.25. Algorithm 1 specifies the feasibility penalty, candidate selection, and final recomputation. Strict target attainment is evaluated separately by condition A below.

Base GUIDE optimizes and selects candidates using target geometry (Song et al., 2026; Wang et al., 2026b). GUIDE+checker adds the acceptance-margin penalty described in Section 5.2 and receives the same checker scores and thresholds as the corresponding direct baseline (Athalye et al., 2018; Tramér et al., 2020). Independent scene references are used only for final evaluation.

Shared support constrains predictions through fixed anchor identities across views (Yao et al., 2018; Wu et al., 2024; Wang et al., 2024). The checker uses a separate pixel grid to compute cycle errors and the fraction of invalid candidates from the returned predictions. Section 5.2 compares these optimization choices with controls that use checker scores directly. Appendices B and D describe residuals under different output dependencies and held-out relations, respectively.

Section 5.2 tests the role of shared anchor identity: independent per-view targets remove shared identity across views, while shuffled associations preserve the number of loss terms but change the correspondence between anchors and target views.

4.3 Joint Evaluation

Joint success requires three conditions on the same returned output: A, strict camera-target attainment; B, material external error relative to an independent scene reference; and Q, checker acceptance under fixed relations and calibrated thresholds (Paul et al., 2026; Han et al., 2026; Asim et al., 2025). The joint event and its success rate are

$$J = A \wedge B \wedge Q, \qquad JSR = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[J_i]. \tag{4}$$

Algorithm 1 Base GUIDE candidate construction for one optimization start

Require: Images I, frozen model f, camera command, update budget K, initial w
Run clean inference; construct target cameras and fixed shared support (Eq. 2).
Initialize best-feasible and best-camera candidates to empty.
for $k = 1, \ldots, K$ **do**
    $I' \leftarrow \mathrm{clip}(I + \epsilon \tanh w, 0, 1)$, where $\epsilon = 8/255$.
    Compute $f(I')$ and jointly align cameras, depth, and points to the target.
    Evaluate L_{cam} and L_{support}.
    Retain I' as best-camera when its camera loss is the smallest so far.
    If $L_{\mathrm{cam}} \leq 0.08$, retain I' when its support loss is the best feasible.
    if $k \leq \lfloor 0.2K \rfloor$ **then**
        $L \leftarrow L_{\mathrm{cam}}$.
    else
        $L \leftarrow L_{\mathrm{support}} + 100 \max(L_{\mathrm{cam}} - 0.08, 0)^2$.
    end if
    Update w by one Adam step on L with learning rate 0.02.
end for
Select best-feasible, or best-camera with an infeasible status if none exists.
return selected images and a recomputed complete model output.

Condition A requires a mean direction cosine of at least 0.95, a mean displacement-magnitude ratio in $[0.9, 1.1]$, and a normalized target root-mean-square error of at most 0.1. The direction cosine measures agreement between the observed displacement and the prescribed change. Condition B requires both $e_a - e_0 \geq 0.02$ and $e_a \geq 2e_0$, where e_a and e_0 are attacked and clean trajectory errors normalized by the reference scale. Appendix A describes alignment and the treatment of degenerate cases.

If view filtering or geometric repair produces a new output, target attainment, external error, and checker acceptance are all recomputed on that output.

5 Experiments

5.1 Experimental Setup

Datasets and models. We evaluate VGGT, VGGT-Ω, and Déjà View (Wang et al., 2025a; 2026a; Burzio et al., 2026) on 16 scenes: ten TUM sequences and two scenes each from CO3D, Sintel, and Tanks and Temples, with 30–60 frames per sequence (Sturm et al., 2012; Reizenstein et al., 2021; Butler et al., 2012; Knapitsch et al., 2017). This yields 48 model–scene combinations, including 36 with independent trajectory references. Confirmation uses 48 SceneNet RGB-D instances with eight input views each, yielding 144 units with ground-truth camera poses and depth (McCormac et al., 2017). Development, calibration, clean test, and confirmation sets are disjoint at the scene level (Appendix C).

Baselines and attack settings. The core comparison includes targeted PGD, camera-only, camera+checker, GUIDE, and GUIDE+checker, each with 200 updates (Madry et al., 2018). Confirmation uses 200, 600, or 2,000 updates per start and three starts per configuration. Methods share inputs, camera targets, and pixel budgets; score-aware controls also share checker scores, thresholds, preprocessing, and candidate-selection rules. Settings are selected on 80 development sequences per model and fixed before confirmation. Total costs include all starts and final validation.

Evaluation and statistics. We report joint success $J = A \wedge B \wedge Q$. The checker combines camera–depth cycle errors and invalid-projection rates, with thresholds calibrated on 199 sequences per model at a nominal 5% rejection level and tested on 400 independent clean sequences (Angelopoulos & Bates, 2021; Uesato et al., 2018; Croce & Hein, 2020). Paired 95% intervals use 5,000 scene-cluster bootstrap samples , resampling whole scenes while retaining all model and method results and fixing calibration. The core trajectory analysis has 12 independent scenes; confirmation has 48 independent instances. All planned units remain in the denominator for their respective endpoints.

Table 1: Core attack results. Score denotes checker access; brackets report 95% scene-cluster bootstrap CIs. Dashes indicate unreported model-level results.

		Overall			Joint success by model (/12)		
Method	Score	Target /48	Joint /36	JSR [95% CI]	VGGT	VGGT-Ω	Déjà View
Targeted PGD	No	40	11	30.6% [19.4, 41.7]	–	–	–
Camera-only	No	41	12	33.3% [19.4, 47.2]	4	4	4
Camera+checker	Yes	41	24	66.7% [47.2, 83.3]	7	8	9
GUIDE	No	41	29	80.6% [61.1, 97.2]	9	10	10
GUIDE+checker	Yes	42	31	86.1% [63.9, 100.0]	10	10	11

Table 2: Attack controls. (a) Joint success on 144 units; budgets denote updates per start, with three starts and final validation included in cost. (b) Target attainment and held-out $A \wedge Q$ on 48 units, excluding condition B.

(a) Matched optimization budgets

Method	200	600	2,000
Direct attack + checker	85	99	111
GUIDE, no score access	113	118	122
GUIDE+checker	120	125	128

(b) Shared-anchor associations

Association setting	Target	$A \wedge Q$
Correctly shared anchors	41	31
Independent per-view targets	40	23
Shuffled associations	40	20

Appendix E defines the paired statistics. Depth-label and robot metrics are specified with their results.

5.2 Attack Evaluation

Main geometric results. Projection analysis uses 42 model–scene combinations from 14 scenes; independent trajectory-error analysis uses 36 combinations from 12 scenes. CO3D reconstruction references and independent trajectory references are identified separately.

GUIDE yields more accepted errors at the same number of attained camera targets. In the core matrix, GUIDE and the camera-only attack each attain 41/48 strict targets. Among the 36 combinations with independent trajectory references, both attain 31 targets and produce 32 material external errors. The checker accepts 33 GUIDE outputs and 14 camera-only outputs; their joint success counts are 29/36 and 12/36, respectively (Table 1). Shared geometric support increases the acceptance of targeted errors while the numbers of attained targets and material errors remain the same.

GUIDE increases the joint success rate by 47.2 percentage points over the camera-only attack, with a paired interval of [30.6, 63.9]. GUIDE without score access exceeds the score-aware direct baseline by 13.9 percentage points, with an interval of [2.8, 27.8]. When both methods have checker access, GUIDE+checker exceeds camera+checker by 19.4 percentage points, with an interval of [8.3, 33.3].

Shared support increases joint success counts for all three models. The camera-only attack achieves 4/12 on each model; GUIDE achieves 9/12 on VGGT and 10/12 on both VGGT-Ω and Déjà View (Table 1). Statistical comparisons use paired results from the same 12 scenes. Figure 2 compares clean and perturbed depth predictions from the three models for the same scene and frame.

Matched budgets and checker access. The confirmation set fixes the data source and input size. Smooth scores, exact forward passes with approximate backward passes, and component-wise constraints receive the same development-search resources. Camera+checker adds a normalized acceptance-margin penalty to the camera objective; GUIDE+checker adds the same penalty to the shared support objective. Both obtain gradients through differentiable sampling and approximations to validity, while candidate selection and final scoring use the exact checker with hard decisions.

GUIDE and its score-aware variant achieve more joint successes than the direct attack at all three matched budgets (Table 2(a)). The confirmation comparison comprises 48 independent SceneNet instances and three models, for a total of 144 evaluation units. All counts in panel (a) use these 144 units as their denominator.

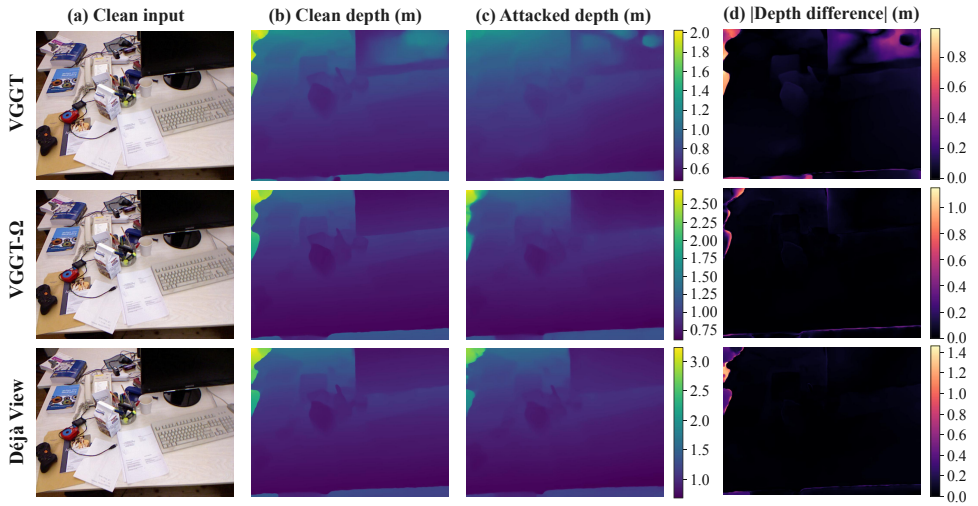


Figure 2: Depth predictions for frame 30 of TUM sequence, internal identifier S09. Rows correspond to VGGT, VGGT-Ω, and Déjà View. Columns (a–d) show the clean input for reference, the clean prediction, the perturbed prediction, and their absolute difference. Columns (b) and (c) share a color scale within each row; column (d) uses a separate scale. Depth is shown in meters, with each model retaining its native scale. Gray denotes invalid depth.

With 2,000 updates per start, three starts, and matched checker access, GUIDE+checker achieves 128 joint successes, compared with 111 for the direct attack with checker access. The difference is $17/144$, or 11.8 percentage points, with a scene-clustered paired interval of 4.2–20.1 percentage points. Both methods produce 137 outputs that meet the material external-error condition B. Of these, the checker rejects 5 GUIDE+checker outputs and 22 direct-attack outputs. Each method also produces 4 accepted erroneous outputs that miss the prescribed target and 7 outputs that fall below the material-error threshold. The gain from shared support is concentrated in the acceptance of material errors. Appendix E provides the complete paired counts.

GUIDE without score access achieves 113/144 joint successes with 200 updates per start, compared with 111/144 for the direct baseline at the long budget. The difference is 1.39 percentage points, with a two-sided interval of -2.08 to 5.56.

Shared-anchor ablation. Correctly shared anchors yield more held-out relation acceptances at similar camera-target success counts (Table 2(b)). The three association settings attain 41, 40, and 40 strict camera targets, and achieve 31, 23, and 20 held-out relation acceptances, respectively. Shuffling associations changes correspondences while preserving the number of loss terms. The resulting decrease in held-out acceptance indicates that cross-view associations affect the geometric support for camera targets and their acceptance.

Acceptance of erroneous depth labels. The geometric checker also determines which depth predictions are accepted for label export (Yang et al., 2024; Li & Snavely, 2018). Under a fixed scale convention, condition B_D requires the sequence-level absolute relative depth error to increase by at least 0.05 and reach at least twice the clean error. The erroneous-label acceptance rate is $\mathrm{LAR} = N^{-1} \sum_{i=1}^{N} \mathbf{1}[Q_i \wedge B_{D,i}]$, using all N planned units. Missing or invalid candidates contribute zero.

Among the 144 evaluation units, 103 erroneous GUIDE depth outputs are accepted, compared with 74 for the direct attack. Shared support allows more erroneous predictions to enter the label-export pipeline.

Closed-loop robot outcomes. Spatial predictions also enter robot policies through cached visual representations. We evaluate online geometric attacks on FALCON in CALVIN and on R3DP in RoboTwin (Zhang et al., 2026b; Mees et al., 2022; Zhang et al., 2026a; Mu et al., 2025). Each clean/attack pair starts from the same initial task state. FALCON covers 20 tasks with ten initial

Table 3: Closed-loop task outcomes under the $8/255$ cumulative-drift attack. SR and ASR are percentages; Δ is the clean success rate minus the attacked success rate, in percentage points.

Policy	N	SR (clean/attack)	F / R	ASR [95% CI]	Δ [95% CI]
FALCON	200	68.0 / 28.5	81 / 2	59.6 [51.2, 67.4]	39.5 [32.4, 46.6]
R3DP	198	71.7 / 34.3	78 / 4	54.9 [46.7, 62.9]	37.4 [30.1, 44.7]

states per task. The upstream R3DP release provides only three task-specific policy checkpoints, covering block hammering, block stacking, and shoe placement; we evaluate all three released checkpoints. R3DP yields 198 valid pairs from 200 initial-state attempts. The checkpoint source and task names are listed in Appendix G.1.

We instantiate GUIDE with the shared-anchor construction and support objective of Secs. 4.1–4.2 under the online optimization and delivery settings below. These experiments measure downstream task impact; Tables 1 and 2(b) evaluate the role of shared support in geometric acceptance. An asynchronous worker optimizes $\delta = \epsilon \tanh(w)$ with $\epsilon = 8/255$. The cumulative-drift target has a window of two, with a translation increment of 0.04 and a rotation increment of 2° per window. FALCON requests an update every four outer policy steps, and R3DP every eight observations. Control continues while the worker constructs the update.

For N valid pairs, let C and U denote the numbers of clean and attacked successes. Let F count transitions from clean success to attack failure, and R the reverse transitions. The conditional attack success rate is $ASR = F/C$, and the net decrease in task success rate is $\hat{\Delta} = (F - R)/N$. Table 3 reports both transition directions and retains all valid pairs, including episodes in which no update was delivered. Brackets report Wilson 95% intervals for conditional ASR and descriptive paired 95% intervals for the net decrease.

FALCON successes decrease from 136/200 to 57/200, a net reduction of 39.5 percentage points, with 81 forward and 2 reverse transitions. R3DP successes decrease from 142/198 to 68/198, a net reduction of 37.4 percentage points, with 78 forward and 4 reverse transitions. The aggregate conditional ASRs are 59.6% and 54.9%, respectively. Figure 3 shows task progression in two paired recordings. The clean trajectories complete drawer opening or hammering, whereas the attacked trajectories fail to complete the corresponding interactions.

5.3 Defense Evaluation

Geometric consistency checks. The primary checker uses round-trip cycles defined by predicted cameras and depth as a common geometric relation. It determines acceptance for evaluation units with the required outputs across all three models.

A source pixel is backprojected using predicted depth and projected into a target view. After sampling the target depth, it is projected back into the source view to obtain the cycle error relative to the original pixel. The checker combines the median error, the 90th-percentile error, and the fraction of invalid candidates. Source pixels and view pairs are fixed in advance, errors are normalized by the image diagonal, and invalid projections contribute to the score. Robust centers, scales, and anomaly directions for each component are determined from the development set. The maximum standardized component defines the final calibration score.

Calibration scores are sorted in ascending order. With $n = 199$ independent calibration sequences and nominal rejection level $\alpha = 0.05$, the threshold is $q = s_{(\lceil (n+1)(1-\alpha) \rceil)} = s_{(190)}$, and acceptance is $Q = \mathbf{1}[s(I, y) \leq q]$. The three models reject 18/400, 21/400, and 19/400 sequences on their respective independent clean test sets, establishing the checker's observed operating points. Under exchangeability of scores, this threshold controls the marginal rejection rate of clean samples (Angelopoulos & Bates, 2021). Clean outputs from the same scene are used to diagnose residual changes; the deployment score is computed from the current input and its returned output.

We evaluate geometric deception under view filtering and protected RGB-D validation. View filtering selects reconstruction inputs using signals internal to the model; protected RGB-D uses additional scene observations to validate predictions.

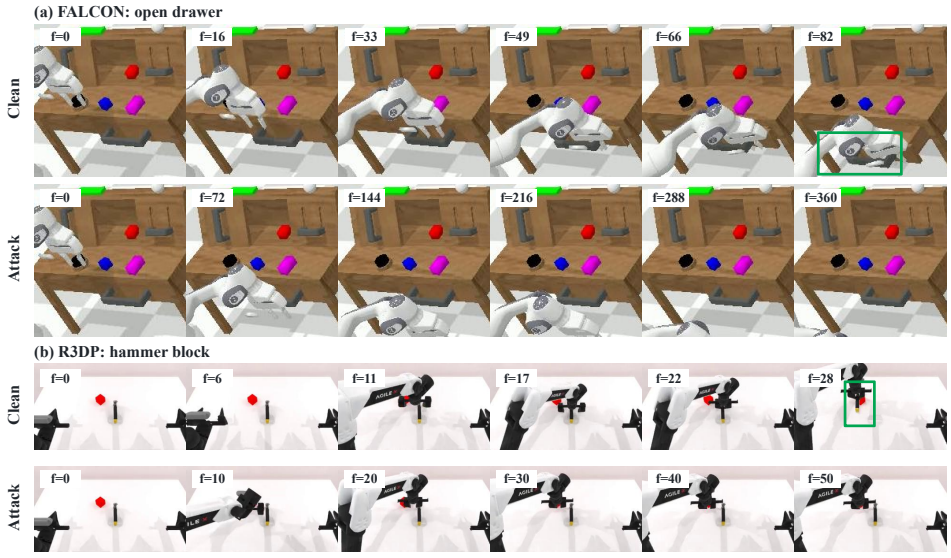


Figure 3: Two examples of paired robot trajectories under GUIDE with cumulative-drift targets and an ℓ_∞ perturbation budget of $8/255$. The first two rows show FALCON on the drawer-opening task; the last two show R3DP on the block-hammering task. Within each pair, the clean trajectory is shown above the attacked trajectory. Green boxes mark the region where the clean trajectory completes the task. f denotes the zero-based video frame index. Frames are sampled according to each trajectory's own progression, so columns are not temporally synchronized.

View filtering. After RobustVGGT view filtering, the VGGT subset retains 9/12 joint successes and at least 90% of the original view coverage. This procedure selects images using internal attention or feature similarity, then reruns VGGT reconstruction on the retained views (Han et al., 2026). Target attainment and external error are computed on the geometry reconstructed after filtering.

Protected RGB-D validation. Each confirmation instance also provides protected RGB-D observations hidden from the attacker. These observations supply additional scene evidence for validation. Under this condition, geometric joint success counts on the confirmation set are 16/144 for the direct attack and 24/144 for GUIDE. Appendix D describes observation and calibration conditions. The same validation also reduces the acceptance of erroneous depth labels: 12 direct-attack outputs and 19 GUIDE outputs are accepted out of 144 planned units, compared with 74 and 103 under the geometric checker.

Discussion. Geometric validation depends on the evidence it observes (Paul et al., 2026; Han et al., 2026; Asim et al., 2025). Self-consistency checks can accept coordinated errors. Protected RGB-D validation reduces joint attack success but does not eliminate it, while RobustVGGT filtering retains 9/12 joint successes on VGGT (Section 5.3). These findings support complementing internal consistency with independent scene observations.

6 Conclusion

We presented GUIDE, which coordinates camera, depth, and point predictions through shared target geometry to induce targeted geometric errors with compatible cross-view support. Core comparisons and independent confirmation experiments show that shared support increases joint attack success, while the association ablation identifies the role of cross-view target correspondences in acceptance on held-out relations. The downstream experiments demonstrate increased acceptance of erroneous depth labels and reduced task completion rates for FALCON and R3DP under the evaluated asynchronous update schedules. These findings reveal a limitation of validating geometric predictions through their internal agreement and motivate validation with independent scene evidence.

AI USE STATEMENT

Generative AI tools were used to aid or polish writing, for retrieval and discovery of related work, and for research ideation or execution. The authors reviewed the AI-assisted material and take responsibility for the final content of this submission.

ETHICS STATEMENT

This work studies adversarial manipulation of spatial foundation models and its effects on geometric validation, depth-label export, and downstream robot-policy evaluation. The setting is dual-use: the techniques can support controlled security evaluation, but could also be misused to induce erroneous geometric predictions. We therefore limit our claims to the threat model, datasets, perturbation budgets, validation procedures, and downstream settings described in this paper; we do not claim deployment-level security or a formal robustness certificate. The study does not involve human participants or the collection of private personal data. Any release of code, configurations, or model-related artifacts should preserve the anonymity, licensing, and access restrictions of the underlying datasets, checkpoints, and evaluation environments.

REPRODUCIBILITY STATEMENT

The manuscript specifies the GUIDE threat model, target construction, optimization procedure, checker conditions, evaluation sets, statistical definitions, native reconstruction protocol, and robot update protocol. The main text and Appendices A–G provide the corresponding implementation and evaluation details, including perturbation budgets, update schedules, candidate selection, calibration rules, denominators, paired statistics, and checkpoint/task descriptions. Public datasets and external model resources are identified by their original sources. The source package provides the manuscript source and figures; third-party datasets, model weights, and checkpoints are not redistributed. Our code is available in an anonymous repository: `https://anonymous.4open.science/r/GUIDE_code-B25B`.

REFERENCES

Anastasios N. Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021. URL https://arxiv.org/abs/2107.07511.

Mohammad Asim, Christopher Wewer, Thomas Wimmer, Bernt Schiele, and Jan Eric Lenssen. Met3r: Measuring multi-view consistency in generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Asim_MET3R_Measuring_Multi-View_Consistency_in_Generated_Images_CVPR_2025_paper.pdf.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, 2018. URL https://proceedings.mlr.press/v80/athalye18a.html.

Alessandro Burzio, Tobias Fischer, Sven Elflein, Qunjie Zhou, Riccardo de Lutio, Jiawei Ren, Jiahui Huang, Shengyu Huang, Marc Pollefeys, Laura Leal-Taixé, Zan Gojcic, and Haithem Turki. Déjà view: Looping transformers for multi-view 3d reconstruction. *arXiv preprint arXiv:2605.30215*, 2026. URL https://arxiv.org/abs/2605.30215.

Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision*, 2012. URL https://rd.springer.com/chapter/10.1007/978-3-642-33783-3_44.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on*

Machine Learning, volume 119, pp. 2206–2216, 2020. URL https://proceedings.mlr.press/v119/croce20b.html.

Xianze Fang, Jingnan Gao, Zhe Wang, Zhuo Chen, Xingyu Ren, Jiangjing Lyu, Qiaomu Ren, Zhonglei Yang, Xiaokang Yang, Yichao Yan, and Chengfei Lyu. Dens3r: A foundation model for 3d geometry prediction. *arXiv preprint arXiv:2507.16290*, 2025. URL https://arxiv.org/abs/2507.16290.

Jisang Han, Sunghwan Hong, Jaewoo Jung, Wooseok Jang, Honggyu An, Qianqian Wang, Seungryong Kim, and Chen Feng. Emergent outlier view rejection in visual geometry grounded transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 427–437, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/papers/Han_Emergent_Outlier_View_Rejection_in_Visual_Geometry_Grounded_Transformers_CVPR_2026_paper.pdf.

András Horváth and Csaba M. Józsa. Targeted adversarial attacks on generalizable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 3718–3727, 2023. URL https://openaccess.thecvf.com/content/ICCV2023W/AROW/html/Horvath_Targeted_Adversarial_Attacks_on_Generalizable_Neural_Radiance_Fields_ICCVW_2023_paper.html.

Wenxiang Jiang, Hanwei Zhang, Xi Wang, Zhongwen Guo, and Hao Wang. Nerfail: Neural radiance fields-based multiview adversarial attack. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(19):21197–21205, 2024. doi: 10.1609/aaai.v38i19.30113. URL https://doi.org/10.1609/aaai.v38i19.30113.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 2017. URL https://snu.elsevierpure.com/en/publications/tanks-and-temples-benchmarking-large-scale-scene-reconstruction/.

Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Li_MegaDepth_Learning_Single-View_CVPR_2018_paper.html.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. URL https://arxiv.org/pdf/1706.06083.

John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J. Davison. Scenenet rgb-d: 5m photorealistic images of synthetic indoor trajectories with ground truth. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017. URL https://openaccess.thecvf.com/content_iccv_2017/html/McCormac_SceneNet_RGB-D_Can_ICCV_2017_paper.html.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022. URL https://arxiv.org/abs/2112.03227.

Yao Mu, Tianxing Chen, Zanxin Chen, Shijia Peng, Zhiqian Lan, Zeyu Gao, Zhixuan Liang, Qiaojun Yu, Yude Zou, Mingkun Xu, Lunkai Lin, Zhiqiang Xie, Mingyu Ding, and Ping Luo. Robotwin: Dual-arm robot benchmark with generative digital twins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27649–27660, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Mu_RoboTwin_Dual-Arm_Robot_Benchmark_with_Generative_Digital_Twins_CVPR_2025_paper.pdf.

Yaniv Nemcovsky, Matan Jacoby, Alex M. Bronstein, and Chaim Baskin. Physical passive patch adversarial attacks on visual odometry systems. In *Proceedings of the Asian Conference on Computer Vision*, pp. 1795–1811, 2022. URL

https://openaccess.thecvf.com/content/ACCV2022/papers/Nemcovsky_Physical_Passive_Patch_Adversarial_Attacks_on_Visual_Odometry_Systems_ACCV_2022_paper.pdf.

Soumava Paul, Prakhar Kaushik, and Alan Yuille. Can these views be one scene? evaluating multiview 3d consistency when 3d foundation models hallucinate. *arXiv preprint arXiv:2605.18754*, 2026. URL https://arxiv.org/abs/2605.18754.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10901–10911, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Reizenstein_Common_Objects_in_3D_Large-Scale_Learning_and_Evaluation_of_Real-Life_ICCV_2021_paper.html.

Qi Song, Ziyuan Luo, Haoliang Han, and Renjie Wan. Generating multi-view adversarial examples for visual geometry grounded transformer. *arXiv preprint arXiv:2608.20748*, 2026. URL https://arxiv.org/abs/2608.20748.

Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *IEEE International Conference on Intelligent Robots and Systems*, 2012. doi: 10.1109/IROS.2012.6385773. URL https://doi.org/10.1109/IROS.2012.6385773.

Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/papers/Sun_NeuralRecon_Real-Time_Coherent_3D_Reconstruction_From_Monocular_Video_CVPR_2021_paper.pdf.

Florian Tramér, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020. URL https://neurips.cc/virtual/2020/public/poster_11f38f8ecd71867b42433548d1078e38.html.

Jonathan Uesato, Brendan O'Donoghue, Aäron van den Oord, and Pushmeet Kohli. Adversarial risk and the dangers of evaluating against weak attacks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 5025–5034, 2018. URL https://proceedings.mlr.press/v80/uesato18a.html.

VGGT Authors. Vggt official colmap export and bundle-adjustment implementation. GitHub repository, 2026. URL https://github.com/facebookresearch/vggt/blob/main/demo_colmap.py.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5294–5306, 2025a. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Wang_VGGT_Visual_Geometry_Grounded_Transformer_CVPR_2025_paper.pdf.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026a. URL https://arxiv.org/pdf/2605.15195.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10510–10522, 2025b. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Wang_Continuous_3D_Perception_Model_with_Persistent_State_CVPR_2025_paper.pdf.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20697–20709, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/papers/Wang_DUSt3R_Geometric_3D_Vision_Made_Easy_CVPR_2024_paper.pdf.

Weijie Wang, Songlong Xing, Zhengyu Zhao, Nicu Sebe, and Bruno Lepri. Poinit-of-view: Poisoning initialization of views transfers across multiple 3d reconstruction systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20670–20679, 2026b. URL https://arxiv.org/pdf/2604.16540.

Alex Wong, Safa Cicek, and Stefano Soatto. Targeted adversarial perturbations for monocular depth prediction. In *Advances in Neural Information Processing Systems*, volume 33, pp. 8486–8497, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/609e9d4bcc8157c00808993f612f1acd-Abstract.html.

Jiang Wu, Rui Li, Haofei Xu, Wenxun Zhao, Yu Zhu, Jinqiu Sun, and Yanning Zhang. Gomvs: Geometrically consistent cost aggregation for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/papers/Wu_GoMVS_Geometrically_Consistent_Cost_Aggregation_for_Multi-View_Stereo_CVPR_2024_paper.pdf.

Xi-Jin Xie, Long-Long Liao, Yuanlong Yu, Di Guo, and Hua-Ping Liu. Universal adversarial attacks for visual odometry systems. In *IEEE International Conference on Development and Learning*, pp. 288–293, 2023. doi: 10.1109/ICDL55364.2023.10364418. URL https://doi.org/10.1109/ICDL55364.2023.10364418.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10371–10381, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/papers/Yang_Depth_Anything_Unleashing_the_Power_of_Large-Scale_Unlabeled_Data_CVPR_2024_paper.pdf.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *European Conference on Computer Vision*, 2018. URL https://openaccess.thecvf.com/content_ECCV_2018/html/Yao_Yao_MVSNet_Depth_Inference_ECCV_2018_paper.html.

Mingjun Yin, Shasha Li, Chengyu Song, M. Salman Asif, Amit K. Roy-Chowdhury, and Srikanth V. Krishnamurthy. Adc: Adversarial attacks against object detection that evade context consistency checks. *arXiv preprint arXiv:2110.12321*, 2022. URL https://arxiv.org/abs/2110.12321.

Yuhao Zhang, Wanxi Dong, Yue Shi, et al. R3dp: Real-time 3d-aware policy for embodied manipulation. *arXiv preprint arXiv:2603.14498*, 2026a. URL https://arxiv.org/abs/2603.14498.

Zhengshen Zhang, Hao Li, Yalun Dai, Zhengbang Zhu, Lei Zhou, Chenchen Liu, Dong Wang, Francis E. H. Tay, Sijin Chen, Ziwei Liu, Yuxiao Liu, Xinghang Li, and Pan Zhou. From spatial to actions: Grounding vision-language-action model in spatial foundation priors. In *International Conference on Learning Representations*, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/3cbe9fcdccb2399bcd6e6d01cbcae1fd-Abstract-Conference.html.

A Geometry Targets and Evaluation Endpoints

Poses follow the camera-to-world convention, $T_t = [R_t, c_t]$, where $R_t \in \mathrm{SO}(3)$ is the camera orientation and $c_t \in \mathbb{R}^3$ is the camera center. When images are resized, pixel centers are transformed as

$$u' = (u + 0.5)s - 0.5,$$

where s is the resize factor. Cropping adds origin offsets in the order in which the operations are applied, and the intrinsics are updated accordingly. A valid projection requires finite coordinates, positive camera depth, and an image location within bounds. Missing depth, nonfinite coordinates, nonpositive depth, and out-of-bounds projections are counted separately.

Prescribed camera target. For frame t, the prescribed target camera is

$$T_t^\star = \mathrm{Exp}_{\mathrm{SE}(3)}(\lambda_t \xi)\, T_t^0, \qquad \lambda_t \in [0, 1], \tag{5}$$

where T_t^0 is the clean camera pose, $\xi \in \mathfrak{se}(3)$ is the prescribed endpoint twist, and λ_t is the predefined normalized command coefficient for frame t. Thus, t is used only as the frame index: $\lambda_t = 0$ gives the clean pose and $\lambda_t = 1$ applies the full prescribed endpoint deformation. The schedule $\{\lambda_t\}$ is fixed before optimization and is shared by all methods.

For $\mathrm{Exp}_{\mathrm{SE}(3)}([\omega; v])$, v is the translational twist coefficient rather than the endpoint translation itself. The prescribed endpoint translation p is converted to v using the inverse left Jacobian of $\mathrm{SO}(3)$. All methods use the same predefined command specification.

Gauge fixing and target alignment. A reconstruction is defined only up to a common orientation-preserving similarity transform. Let $G = (s, Q, q) \in \mathrm{Sim}^+(3)$, with $s > 0$, $Q \in \mathrm{SO}(3)$, and $q \in \mathbb{R}^3$. Its action on a camera-to-world pose is

$$c_t \mapsto sQc_t + q, \qquad R_t \mapsto QR_t.$$

The same transform is applied consistently to cameras, depths, and 3D points, preserving the corresponding projections. We therefore remove this global gauge once for each evaluated trajectory before computing trajectory-based endpoints.

For condition A, the returned attacked trajectory is aligned to the prescribed target trajectory using one orientation-preserving similarity transform fitted over the complete set of evaluated frames. The resulting transform is global: the same scale, rotation, and translation are applied to every frame. It is not fitted separately at individual time steps.

Such a global alignment can absorb the prescribed command only in the degenerate case where a single $G \in \mathrm{Sim}^+(3)$ maps the complete clean trajectory to the complete target trajectory,

$$G \cdot T_t^0 = T_t^\star \qquad \text{for every evaluated frame } t. \tag{6}$$

Equivalently, one fixed (s, Q, q) would have to satisfy

$$sQc_t^0 + q = c_t^\star, \qquad QR_t^0 = R_t^\star$$

for every evaluated frame. The prescribed frame-varying target schedules used in the evaluation are outside this degenerate case. Consequently, removing one sequence-level similarity gauge does not remove the prescribed frame-dependent deformation.

Strict target attainment. Let $\overline{c}_t^0$, $\overline{c}_t^a$, and $\overline{c}_t^\star$ denote the clean, attacked, and target camera centers expressed in the target-aligned coordinate system used for condition A. Define the observed and prescribed displacements by

$$a_t = \overline{c}_t^a - \overline{c}_t^0, \qquad b_t = \overline{c}_t^\star - \overline{c}_t^0.$$

For frames with a nonzero prescribed displacement, their direction cosine and displacement-magnitude ratio are

$$\gamma_t = \frac{a_t^\top b_t}{\|a_t\|_2 \, \|b_t\|_2}, \qquad m_t = \frac{\|a_t\|_2}{\|b_t\|_2}. \tag{A1}$$

Let

$$\mathcal{T}_+ = \{t : \|b_t\|_2 > 0\}$$

denote the frames for which the prescribed displacement is nonzero. The mean direction cosine and mean magnitude ratio are computed only over this set:

$$\overline{\gamma} = \frac{1}{|\mathcal{T}_+|} \sum_{t \in \mathcal{T}_+} \gamma_t, \qquad \overline{m} = \frac{1}{|\mathcal{T}_+|} \sum_{t \in \mathcal{T}_+} m_t.$$

Thus, a frame for which both the prescribed and observed displacements are zero does not enter the denominator of either mean. If $\|b_t\|_2 = 0$ but $\|a_t\|_2 > 0$, the returned trajectory does not attain the prescribed target. Target schedules with $\mathcal{T}_+ = \varnothing$ are not used.

Let $\mathrm{ATE}(T^a, T^\star)$ denote the root-mean-square camera-center error of the globally aligned attacked trajectory relative to the prescribed target trajectory, and let $r_\star$ be the root-mean-square spatial scale of the target trajectory. The normalized target error is

$$e_\star = \frac{\mathrm{ATE}(T^a, T^\star)}{r_\star}.$$

Condition A holds when

$$\overline{\gamma} \geq 0.95, \qquad 0.9 \leq \overline{m} \leq 1.1, \qquad e_\star \leq 0.1. \tag{7}$$

These quantities are computed after the single sequence-level alignment described above; no per-frame alignment is performed.

Material external error. Condition B is evaluated against an independent scene trajectory reference and is separate from the target alignment used for condition A. For this endpoint, the clean and attacked trajectories are evaluated relative to the same independent reference using the orientation-preserving trajectory-alignment convention of Sturm et al. (2012); Butler et al. (2012).. The alignment used to evaluate condition B is not reused to determine target attainment in condition A.

Let ATE_0 and ATE_a denote the clean and attacked trajectory errors relative to the independent scene reference after the corresponding global trajectory alignment. Let r_{GT} denote the root-mean-square spatial scale of that reference trajectory. We report the normalized errors

$$e_0 = \frac{\mathrm{ATE}_0}{r_{\mathrm{GT}}}, \qquad e_a = \frac{\mathrm{ATE}_a}{r_{\mathrm{GT}}}. \tag{A2}$$

A material external error is present when

$$e_a - e_0 \geq 0.02, \qquad e_a \geq 2e_0. \tag{8}$$

Condition B is evaluated only on the evaluation units for which an independent scene reference is designated in advance. Within such an endpoint, a unit that is assigned to the evaluation but whose returned output lacks a quantity required to compute the endpoint is counted as unsuccessful.

Errors for datasets expressed in different physical units are computed in their native units before the scale normalization above. The normalized endpoint definitions are otherwise unchanged across datasets.

B Loss Implementation and Optimization Settings

We fix the set of valid anchor–view correspondences $\mathcal{V}$. For each $(t,k) \in \mathcal{V}$, the target camera coordinates, projected location, and axial depth are $y^\star_{tk} = (R^\star_t)^\top (X^\star_k - c^\star_t)$, $u^\star_{tk} = \pi(K_t y^\star_{tk})$, and $z^\star_{tk} = [y^\star_{tk}]_z$, respectively. Jointly aligned predictions are denoted by $\overline{R}, \overline{c}, \overline{P}, \overline{D}$. The point-map and depth support losses are

$$L_P = \frac{1}{|\mathcal{V}|\,\overline{z}^\star} \sum_{(t,k)\in\mathcal{V}} \|\overline{R}_t^\top \left[\overline{P}_t\left(u^\star_{tk}\right) - \overline{c}_t\right] - y^\star_{tk}\|_2, \tag{B1}$$

$$L_D = \frac{1}{|\mathcal{V}|} \sum_{(t,k)\in\mathcal{V}} \frac{\left|\overline{D}_t\left(u^\star_{tk}\right) - z^\star_{tk}\right|}{\max\left(z^\star_{tk}, 10^{-3}\right)}. \tag{B2}$$

Dense targets come from the same valid target surface. Sparse anchors are selected from a 4×4 source grid and two strata of the clean-depth distribution. Each anchor must be visible in at least two target views and satisfy a fixed angular-baseline requirement. All returned outputs are evaluated against the predefined targets.

Derived point maps share computational pathways with cameras and depth. When $P = Rq + c$, $R^\top(P - c) = q$, so same-frame point-map residuals reflect this algebraic dependence. Adapters define residuals according to the actual structure of derived point maps, independent point heads, or learned rays.

At an exact pixel location, a single unnormalized residual for a derived point map equals $|\overline{D} - z^\star| \, \|K^{-1}\tilde{u}^\star\|_2$. Substitution into Equation B1 gives the sum and normalization over valid correspondences. Bilinear sampling of a preconstructed point map may introduce a depth–ray covariance term. Independent point heads use their own point-map residuals.

The first 20% of offline optimization updates are used for camera warm-up, after which shared support losses are introduced. A complete forward pass is rerun on the selected images for every returned candidate. Optimization costs on the confirmation set include all three starts and final validation.

Table B1: Offline optimization settings.

Setting	Offline controls
Modifiable inputs	Specified multi-view inputs
Pixel budget	8/255
Starts	Three, all included in the cost
Updates per start	200 / 600 / 2,000
Returned output	Fully recomputed
External references	Evaluation only
Timing unit	Complete candidate construction

C Evaluation Sets and Checking Rules

Table C1: Evaluation sets and their purposes.

Set	Independent groups	Model–scene units	Purpose
Core matrix	16 scenes	48	Targets and controls
Calibratable geometry	14 scenes	42	Projection diagnostics
Trajectory-reference subset	12 scenes	36	Independent external error
Confirmation set	48 SceneNet instances	144	Matched joint endpoints

The projection and independent-trajectory subsets are determined by reference availability.

The round-trip cycle consists of source-pixel backprojection, projection into the target view, target-depth sampling, and projection back into the source view. Every step requires finite positive depth; out-of-bounds projections are invalid. The median error, tail error, and invalid fraction are computed over a fixed candidate population. An output is rejected when all candidates are invalid.

D Held Out Relations and Image Supported Validation

Support and held-out relations are identified by source frame, source coordinates, target frame, and sampling region. The held-out set excludes view edges used during optimization and bilinear sampling regions that overlap the support set. It is used to check relations after optimization.

The association ablation in Section 5.2 compares correctly shared anchors, independent per-view targets, and shuffled associations on held-out relations, testing how target organization affects relations that were not supervised during optimization.

Image-correspondence checks use matched observations from the input images. The candidate relative pose defines the fundamental matrix used to compute Sampson error; coordinate conventions, denominator rules, and match coverage are fixed. Feature projections are also computed from the candidate being checked. These observations come from the model inputs, whereas protected RGB-D provides additional observations.

Protected RGB-D observations are hidden from the optimizer and have independent timing and calibration constraints. Under this condition, the geometric joint success counts are 16/144 for the direct attack and 24/144 for GUIDE. Each validation rule is calibrated independently, and successes are counted using the actual acceptance decisions.

E Statistical Definitions and Result Consistency

The paired counts at the long budget are $(108, 20, 3, 13)$, corresponding to success by both methods, success by GUIDE only, success by the direct attack only, and failure by both. The marginal success counts are 128 and 111, with a difference of $100(20-3)/144 \approx 11.8056$ percentage points. The scene-cluster bootstrap uses the scene identities of these paired records and always keeps all three models and both methods together.

The scene difference $d_s = \sum_m (J^G_{sm} - J^D_{sm})$ aggregates the difference in joint success counts across the three models in scene s. Intervals are computed from complete paired records for the corresponding endpoint.

F Native Reconstruction as a Separate Returned Output

VGGT's native reconstruction pipeline performs confidence filtering and export through demo_colmap.py. The –use_ba branch performs tracking, reprojection filtering, and bundle adjustment (VGGT Authors, 2026);; the branch without bundle adjustment uses a confidence threshold of 5.0. The remaining settings are an 8-pixel reprojection threshold, a visibility threshold of 0.2, eight query frames, and at most 4,096 query points.

Let $\mathcal{R}(I')$ denote the output returned by the native pipeline. Its endpoint is $A(\mathcal{R}(I')) \wedge B(\mathcal{R}(I')) \wedge C_{\mathcal{R}}$, with completion and coverage thresholds fixed in advance. If point filtering leaves the cameras unchanged, the original camera-target labels are retained. After bundle adjustment, cameras are realigned and target and external errors are recomputed.

The evaluation protocol distinguishes transfer of fixed attacks from optimization against the native pipeline, with matched information access. Pipeline-specific optimization uses surrogate gradients, while final scoring uses exact forward computation. Completion and coverage are measured relative to the predefined set of views.

G Robot Protocol

G.1 Online Update Records

Both systems use cumulative-drift targets, an $8/255$ image budget, two Adam updates, and a learning rate of 1.2. The target window is two, with a translation increment of 0.04 and a rotation increment of 2°. FALCON requests an update every four outer policy steps; R3DP requests one every eight observations, with the feature-refresh setting $\tau = 8$.

For Table 3, each online optimization request uses the shared-anchor target construction of Sec. 4.1 and the camera and support objectives of Sec. 4.2 and Appendix B. Targets and anchor–view correspondences remain fixed during the two Adam updates, and the returned candidate is selected using the feasibility and selection rules of Algorithm 1. Table 3 reports episode-level task outcomes under the stated asynchronous schedule; its conditional ASR, F/C, is distinct from the geometric joint success criterion $J = A \wedge B \wedge Q$.

R3DP task coverage follows the three released task-specific policy checkpoints in wxdong/R3DP: block hammering, block stacking, and shoe placement. The official implementation links this check-

point release at https://github.com/dazazh/R3DP#3-prepare-model-weights; the checkpoints are hosted at https://huggingface.co/wxdong/R3DP.

FALCON's episode-level trajectories record 1,226 delivered updates out of 1,532 requests (80.0%), whereas the service counter records 1,224. Delivery counts are taken from the episode-level trajectories. Image verification covers 9,192 original/perturbed frame pairs, each with a maximum absolute difference of 8 on the 0–255 pixel scale. Nine episodes receive no update and succeed in both conditions. The saved frame pairs show observations before and after perturbation within attacked episodes; paired clean outcomes are recorded separately.

R3DP has 7,002 successful online optimization records. Both camera channels have a maximum pixel difference of 8, and the median optimization time is 4.351 seconds.

G.2 Statistical Definitions

The paired net decrease is $\hat{\Delta} = (F - R)/N$. The descriptive 95% interval is

$$\hat{\Delta} \pm 1.96\sqrt{\frac{(F+R)/N - \hat{\Delta}^2}{N}}.$$

Conditional ASR uses C as its denominator and a Wilson interval. The two-sided exact McNemar test uses the $F + R$ discordant pairs, giving $p = 7.21 \times 10^{-22}$ for FALCON and $p = 7.61 \times 10^{-19}$ for R3DP. These nominal intervals and tests treat complete pairs as the units of analysis and do not adjust for shared tasks or asynchronous cross-episode dependence.