SUPERVISING 3D OCCUPANCY WHERE THE SENSORS HAVE NOT LOOKED YET

Anonymous authors
Paper under double-blind review

ABSTRACT

Motion changes not only a driving scene, but also which parts of it can be physically verified. We turn this change in observability into supervision for 3D occupancy and a geometric verifier for driving world models. Later LiDAR sweeps often reveal static space that was hidden at an earlier query frame. Rather than collapsing these measurements into denser temporal evidence, we retain when each voxel becomes observable. This yields a query-time partition: space resolved immediately, space resolved only by later measurements (R_{future}), and space never resolved in the log. R_{future} is delayed physical evidence of the hidden present—not an additional model input or a future state to predict.

Extending only the loss region of a matched occupancy learner from R_{obs} to $R_{\text{obs}} \cup R_{\text{future}}$ raises R_{future} F1 by 0.1156 on Occ3D-nuScenes; an independent five-seed retraining gives $+0.112$ with a scene-clustered 95% confidence interval of $[0.103, 0.122]$. Standard occupancy metrics do not decline, and the gain transfers to Waymo, multiple encoders and domains, and a second lifting architecture. The same delayed evidence also gives generated futures an external geometric reference. We decode each video produced by a driving world model into 3D occupancy and check its implied static geometry at the commanded pose against a sensor-derived map. An occupancy reader trained on R_{future} roughly doubles the measured separation between rollouts conditioned on logged and counterfactual actions. When choosing among four stochastic rollouts of the same counterfactual action, the resulting verifier yields about 44 more correctly located occupied voxels per frame than random selection, evaluated in a disjoint region by an independently trained reader. This sensor-grounded score is a candidate geometric reward for future world-model post-training; we do not fine-tune the generator or claim a closed-loop benefit.

1 INTRODUCTION

Motion changes both the world and the observer's access to it. Autonomous systems must reason about each: an occupancy model infers space behind occluders, while a driving world model renders what might follow an action that was never executed. In both cases, the difficult prediction lies beyond what could be checked from the query observation alone.

Temporal aggregation asks what geometry can be reconstructed from the sequence as a whole. We ask a different question: what was verifiable at the query time, and what became verifiable only because the observer moved? Future sweeps have long been used to densify occupancy supervision or train completion (Agro et al., 2024; Liu et al., 2024b); our contribution is not another source of sweeps. It is to retain their measurement provenance instead of erasing it through aggregation. Time of first observation then defines a controlled boundary between immediate and delayed evidence. A voxel first resolved later becomes a physical test of what a model claimed about the hidden scene earlier. Ego motion is thus not merely a way to collect more data; it becomes an instrument for supervising and evaluating predictions that had no reference at the query time.

Concretely, we transform subsequent ego LiDAR sweeps back to a query pose and partition its voxel grid into R_{obs}, resolved at the query frame; R_{future}, unresolved then but resolved by a later sweep; and R_{never}, never resolved in the sequence. This is not a forecast of the future state: rays observed later label the free and occupied geometry that was already present at the query time. We supervise

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.

and evaluate R_{future}, while leaving R_{never} outside the claim because no physical reference exists there.

The partition turns "unobserved" from one ambiguous category into two: hidden geometry that later measurements can falsify, and geometry for which no claim can be checked. On the falsifiable part, six released configurations from five model families occupy a broad coverage–precision frontier that standard scores do not reveal. In a controlled intervention, changing only the loss mask of a matched learner from R_{obs} to $R_{\text{obs}} \cup R_{\text{future}}$ raises R_{future} F1 by 0.1156 (21.7% relative); an independent five-seed retraining estimates $+0.112$ with CI $[0.103, 0.122]$. Standard metrics do not decline, and the result transfers to Waymo, multiple encoders and domains, and a second lifting architecture.

Treating the future as evidence rather than only as a prediction target also changes how a world model can be evaluated. Next-frame training compares a generated future with the one trajectory that occurred. A rollout conditioned on a counterfactual action has no such pair: no camera occupied the pose being rendered. Static geometry, however, can be accumulated from real sensor returns elsewhere in the drive and queried at that pose. We therefore decode every generated video back into the paper's central representation, 3D occupancy, and compare its implied geometry with this sensor-derived map. The future here is an independent physical reference for the generated world, not a target supplied to the generator.

On rollouts from the Cosmos 3 Edge driving world model (NVIDIA et al., 2026), the R_{future}-supervised occupancy reader separates logged from counterfactual actions about twice as strongly as its observed-only counterpart. At poses with no reference image, it also selects among stochastic renderings of the same action; the selected rollout contains about 44 additional correctly located occupied voxels per frame over random selection, measured on a disjoint region by an independently trained reader. This is an offline geometric verification result, not a planning result. It nevertheless identifies a concrete next step: the score could provide a sensor-grounded reward for rejection sampling, preference learning, or reinforcement learning of a world model. We do not take that step here; the generator remains frozen and our planner probe does not establish closed-loop utility.

This paper makes three contributions:

1. We define a query-time observability partition that reserves later sensor returns as physical labels for geometry hidden at the query frame, rather than as model inputs or future states to predict.
2. A matched intervention shows that supervising R_{future} improves hidden-space completion across datasets, encoders, domains and two lifting mechanisms without reducing standard occupancy metrics.
3. The resulting occupancy reader verifies counterfactual world-model rollouts at poses with no reference image, providing a candidate geometric reward for future post-training.

2 Related Work

Where a reference comes from. An evaluation reference should not depend on the system being evaluated. CASCADE (Schmalfuss et al., 2026), for example, uses human-verified annotations to correct model-generated supervision that can contradict the scene. We instead use the ego vehicle's later sensor returns, so each supervised voxel is tied to a physical measurement. This requirement sets the boundary of our study: R_{never} has no independent measurement and receives no label, while a generative model can be audited but cannot provide its own reference (Appendix O).

World models that evaluate trajectories. WoTE (Li et al., 2025) rolls a latent BEV world model forward under candidate trajectories and selects online using scores learned from a driving simulator. Our reader is trained only on occupancy agreement and never sees a planning objective. We therefore treat it as a geometric verifier, not a planner; our probe does not establish transfer to planning (§6.6).

Occupancy supervision and evaluation. Future observation as free supervision is established: UnO (Agro et al., 2024) traces future LiDAR rays into a continuous occupancy field, and Occ4cast (Liu et al., 2024b) joins completion and forecasting while reporting precision and recall. Our novelty is not future sweeps or two-axis reporting but the query-time observability partition: these works

still exclude the region we measure at evaluation. SparseOcc (Liu et al., 2024a) removes ambiguous unscanned voxels from RayIoU; we recover labels for a subset and score it. GS-Occ3D (Ye et al., 2025) instead changes how labels are obtained, discarding its occluded set. These axes are complementary, and our claim is specifically about the dominant driving-occupancy protocol rather than SemanticKITTI-style full-volume completion.

Occupancy for reasoning and world models. Recent systems make occupancy an intermediate representation for language, forecasting and planning (Wei et al., 2024; Liu et al., 2025; Dang et al., 2026; Wang et al., 2026). They ask how occupancy supports a downstream system; we ask which voxels supervise and evaluate the representation in the first place. Conversely, Cosmos 3 Edge is an observation-generating world model rather than an occupancy model; we do not train it, but map its generated frames back to occupancy for a geometric audit.

3 Future Observations as Geometric Supervision

3.1 From hidden now to measured later

On straight-driving validation frames, **57% of voxels occluded at t are directly observed two seconds later**, and 99% of those are still inside the ego grid at that point, against 61% of the grid as a whole. Occluded space lies mostly ahead, so later ego sweeps can label it.

For each frame we accumulate the ego's own subsequent LiDAR sweeps, transform them into the frame's ego coordinates, and ray-cast to separate free from occupied. The partition at query time t is R_{obs} (resolved at t), R_{future} (not resolved at t, resolved by a later sweep) and R_{never} (never resolved). We supervise and score the first two and report nothing on the third. Doubling the future horizon moves free-space error by 0.0006, so R_{never} is structural rather than a window-length artefact. Label-validity checks appear in Section 6.4.

3.2 Why standard supervision omits this region

Occ3D ships `mask_camera` and `mask_lidar` per frame, and the standard protocol evaluates under the former. Despite its name, `mask_camera` records where an occupancy label is available, not what a camera can see. On Occ3D-nuScenes it is a strict subset of the LiDAR mask. Occ3D-Waymo provides a direct check: over 13 segments and 2,578 frames, **40.2% of `mask_camera` lies outside every camera's field of view**; a quarter of that lies behind a vehicle with no rear camera. Standard training therefore leaves R_{future} outside the loss even though later sensor measurements can label it.

3.3 What it looks like

Figure 1 shows both models from the front camera's pose and from directly above, next to the photograph, the LiDAR sweep at t and the ground truth, for a frame chosen by geometry alone before either model was loaded. Each model shows as many voxels as the LiDAR found occupied, so both are compared at one budget, and black marks the wrong ones. The observed-only model's characteristic error is visible in both views: occupancy in open air above the road, and a dense sheet of it around the ego in the unobserved near field, most of which `obs_future` does not assert. Over every scored frame this is systematic: 74% of `obs`'s false positives lie within 10 m of the ego, against 15% of `obs_future`'s, and precision at matched commitment is 0.297 against 0.524 (Figure 3). Scored on equal terms — each model naming as many voxels in the hidden region as the later sweeps found occupied (Appendix A) — `obs_future` is more precise in all eight top-ranked frames (mean $0.309 \rightarrow 0.478$). The supplement shows further frames, and Appendix A gives image projections.

3.4 The boundary: what the arms do in R_{never}

R_{never} is 66.9% of the voxels, larger than R_{obs} and R_{future} together, and nothing scores it. Supervising R_{future} teaches a model to commit in space it cannot see, so it is fair to ask whether that spills where nothing can contradict it. No label exists there by construction, so what can be measured is

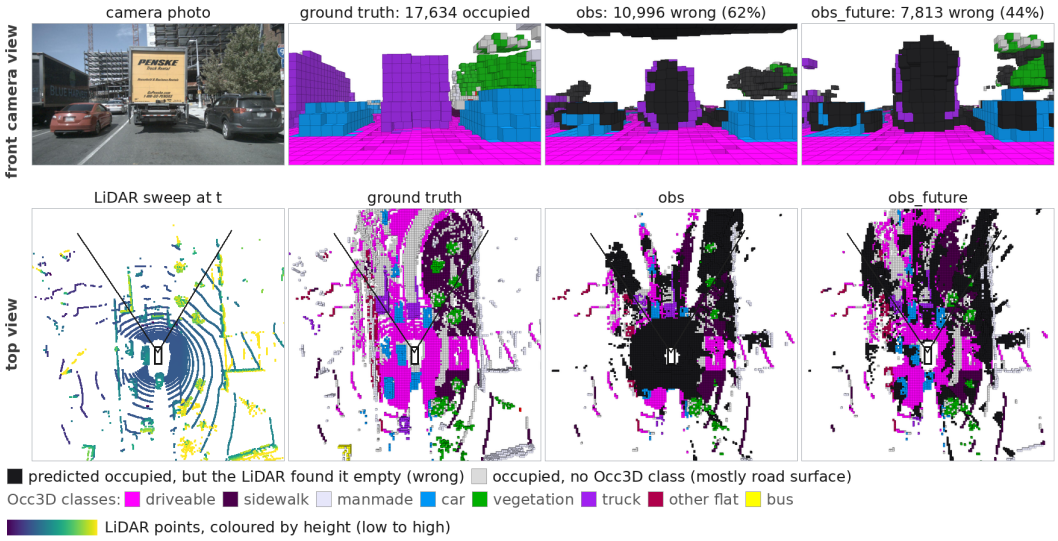


Figure 1: **Both models at the same budget, beside the camera and the LiDAR.** Top: voxels rendered as 0.4 m cubes through the front camera's own intrinsics and extrinsics, beside its photograph. Bottom: the same from directly above, beside the LiDAR sweep at t; the wedge marks that camera's field of view. The ground truth is what the ego's LiDAR found occupied, now or in later sweeps (17,634 voxels), coloured by Occ3D class and light grey where Occ3D has none (mostly road). Each model shows its 17,634 most confident voxels where the free/occupied state is known; black marks those the LiDAR found empty: 10,996 (62%) for `obs`, 7,813 (44%) for `obs_future`. `obs`'s errors form a sheet around the ego and a layer in the air above the road; Appendix A scores both models on the hidden region only. The frame is the top-ranked by visible above-ground R_{future}, chosen before either model was loaded.

assertion rate: thresholded to claim the same fraction of R_{obs}, the `obs_future` arm asserts *less* of R_{never}, by 0.0052 (CI $[-0.0084, -0.0020]$). The spill-over worry is not supported. **But the intervention barely reaches there.** Both arms assert roughly seven times more of R_{never} than of R_{obs} ($7.6\times$, $6.9\times$), which is what a model with no information does. R_{future} supervision improves the region real future evidence supports; it does not solve open-world completion.

3.5 The region is biased, and we report it

R_{future} is what the ego's *later* sweeps resolve, and the ego drives forward, so the region is not a uniform sample of space: it is over-sampled $2.15\times$ in the front $\pm 45^\circ$ sector and concentrated in the near field. Every comparison below is therefore made within distance and azimuth bins with base rates reported per bin, so that the region's shape cannot drive a result.

4 What Published Models Do There

Figure 2 and Table 5 score six released configurations spanning five model families in R_{future} (Appendix B sweeps the whole curve for each, where they cross): a sparse-query model (Wang et al., 2024), a VLM-headed model (Qwen-Drive-1.0) (Zhou et al., 2026), a dense camera model (Yu et al., 2023), the same with a visibility mask, a LiDAR+camera fusion model, and a fully self-supervised one (Gan et al., 2024).

The spread in *willingness to predict* is a factor of two; the spread in *accuracy when they do* is far smaller. No Occ3D number reflects any of this, because every voxel involved is outside what the benchmark scores.

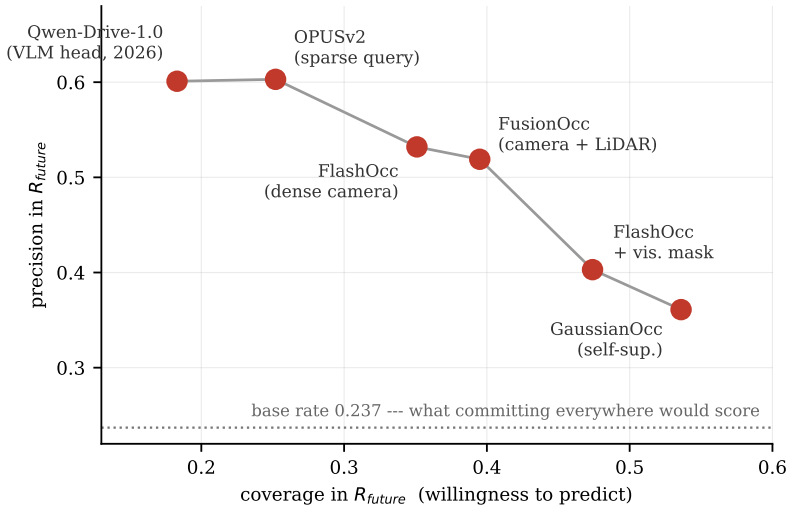


Figure 2: The six released configurations in R_{future}. They lie on one curve: adding an entire sensor modality moves a model *along* it rather than off it. All six are well above the base rate, so none of them is merely guessing — they differ in willingness, not in whether they know anything. The whole picture is invisible to every Occ3D number, because every voxel involved is outside what the benchmark scores.

Two different failures under one number. The dense camera models err by putting the road at the wrong height; GaussianOcc's failure is structural, and the class ranking inverts between them (Appendix C).

Scope. Every checkpoint on this curve is nuScenes-trained, so running one elsewhere would measure a broken input rather than a domain shift — which is why the cross-domain experiment below moves our own students instead.

5 Method: Change the Region, Change Nothing Else

Three supervision regimes, differing **only** in which voxels the loss covers, all read from one label file: `obs` supervises R_{obs} (what the field does); `full` supervises everything, asserting free on never-resolved voxels (the cheapest objection to ours, and knowingly wrong where the claim lives); and `obs_future` supervises $R_{\text{obs}} \cup R_{\text{future}}$ (ours). Architecture, schedule, optimiser, seed and labels are identical across arms.

Matched operating points. Each regime is trained at two or more loss weights. We compare in-region performance at matched coverage so that calibration is not mistaken for model quality. Table 1 reports standard metrics both at each model's argmax and at matched coverage.

6 Experiments

6.1 nuScenes: better in the region, not worse on the benchmark

Seven arms use the full validation split, with five seeds on the headline pair. We report each arm at its own argmax and at matched predicted-occupied rate, which separates model quality from operating point.

Scene- and seed-level replication. An independent five-seed retraining on 5,868 frames from 150 scenes gives an R_{future} F1 gap of $+\mathbf{0.1122}$, scene-clustered CI $[+\mathbf{0.1028}, +\mathbf{0.1219}]$. All five

at each arm's own argmax	obs_future	obs	gap
geometric IoU @ cam. mask	0.4368	0.4082	**+0.0287**
RayIoU	0.3301	0.3234	+0.0068
R_{future} F1	0.6492	0.5336	**+0.1156**
at matched willingness	obs_future	obs	gap
geometric IoU @ cam. mask	0.4368	0.4085	**+0.0283**
RayIoU	0.3286	0.3235	**+0.0051**
second lifting mechanism (proj, no depth predicted), 3 seeds			
geometric IoU @ cam. mask	0.4539	0.4254	**+0.0286**
RayIoU	0.2569	0.2452	**+0.0116**
R_{future} F1	0.6529	0.5758	**+0.0771**

Table 1: Matched intervention on nuScenes. Top two blocks: lift–splat, five seeds; effect/spread is 24.2 for the R_{future} F1 gap and 7.0, 3.8 for the matched standard metrics. Bottom block: a second family whose lift runs in the opposite direction and predicts no depth, three seeds, recipe inherited unchanged. obs_future wins every seed on every row of all three blocks. The two families disagree on absolute RayIoU — a depth-free lift yields weaker ray-surface structure — and agree on the sign and rough size of every gap.

seeds are positive and obs_future leads in **147 of 150 scenes**; the $+0.1156$ in Table 1 lies inside the interval. Component rates, estimator details and per-seed values are in Appendix G.

The $+0.0051$ matched RayIoU margin is small; the benchmark claim is only that standard performance does not decline. F1 is the headline because precision, coverage, hallucination and miss all change, rather than precision alone. Full operating-point sweeps are in Appendix B.

The baseline continues to reduce training loss in the second half of training while its R_{future} F1 is flat on two seeds; learning curves are in Appendix M.

6.2 Waymo: it ports

Nine arms trained from scratch on 2,578 Waymo validation frames reproduce three of four axes: precision $+0.0353$ (effect/spread 3.6), F1 $+0.0510$ (12.7), and miss -0.0217 (4.7). Hallucination is unresolved (-0.0030, 0.2), so we do not count it as a replication. We suspect Waymo's denser returns resolve more of the grid at query time, shrinking the free-space population hallucination is measured over, but we did not test that and leave it open. This changes the LiDAR density, camera coverage and class balance together; the result is not nuScenes-specific.

6.3 Controls and mechanism

The gain is not explained by receiving more targets. Supervising every voxel and asserting free in R_{never} gives R_{future} F1 0.397, below obs's 0.430 and obs_future's 0.468. More decisively, rotating the label field 90° preserves class balance and spatial structure, supplies 58% more voxels than obs_future, but destroys image correspondence; it lands on the obs frontier.

Why it works. The mechanism is confidence separation in unobserved space. Across three seeds, the best precision reachable at any threshold rises from 0.537 ± 0.016 to 0.990 ± 0.001 ($+0.453$, 29 times the seed spread); obs_future reaches the baseline's ceiling while covering roughly four times as much of the region. False-positive confidence relative to true-positive confidence falls from 0.928 to 0.835, and Waymo independently gives a $+0.444$ ceiling gain. This is not a universal calibration improvement: in Waymo's observed region obs_future has worse ECE (0.189 vs. 0.060).

6.4 Where the benefit lands

The advantage decays with distance from the realised path ($+0.257$ precision at 0–2 m to $+0.042$ at 10–20 m). The labels survive two independent checks: a cooperating agent's direct observation

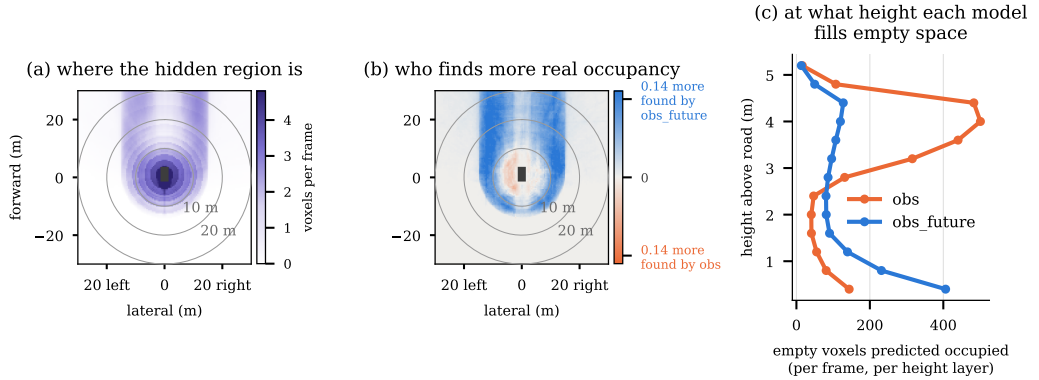


Figure 3: **The gain over all** 5,867 **scored validation frames**, from above, ego at the centre facing up; both models commit the same number of voxels per frame. (a) Where the hidden region R_{future} lies: hidden voxels per frame in each column; it reaches forward, where the ego's later sweeps go. (b) Occupied voxels found by `obs_future` minus `obs`, per column: blue where `obs_future` finds more (positive in 90.2% of touched columns, +774 voxels per frame overall), orange where `obs` does. (c) At what height each model fills empty space: `obs` places 85% of these errors above 2 m, `obs_future` 42%. Unlike Figure 1, no frame is selected. Range and count breakdowns are in Appendix H.

($\kappa = 0.778$, rotated control 0.477) and the physical fact that the future ego body occupies no labelled obstacle voxel in the median scene (Appendix D).

At matched predicted-occupied rate, recall rises $0.716 \rightarrow 0.856$ on objects hidden at t and revealed after they move, and $0.782 \rightarrow 0.914$ on hidden static structure, 3/3 seeds on both. Waymo reproduces comparable gains (+0.075 and +0.070). The drift classifier is ours, so we claim parity across the two subsets, not vehicle-specific improvement.

Class-stratified R_{future}: the gain is not blob suppression. Binary occupancy invites the objection that the arm has only learned to stop asserting near the ego, lifting F1 while improving nothing about what is there. Setting *one* global operating point inside R_{future}, matched between arms, and slicing that same selection by the teacher's class naming — no per-class threshold — the gap is positive in every category, including the two mere suppression would not touch: **vehicles** +0.102**, vulnerable road users** +0.108. These are class-stratified *binary* results, not semantic-classification accuracy, and the vulnerable row rests on 5036 voxels (Appendix N).

The gap also grows sixfold with lead time, from +0.040 at 0–1 s to +0.250 beyond 7 s (Figure 9), 3/3 seeds per bin. Joint range–lead-time bins retain the trend in three of four distance rings. We do not claim a mechanism for this unpredicted result.

Operational readout. Threshold sweeps expose a difference hidden by argmax metrics: the baseline's phantom-clearance rate moves only $0.824 \rightarrow 0.775$, whereas `obs_future` spans $0.806 \rightarrow 0.004$. This is tunability, not demonstrated safety: at matched miss rate the phantom rates are indistinguishable. Appendix E contains the full clearance definition, sweeps and negative controls.

6.5 Architecture, encoder and domain robustness

We repeat the matched pair with frozen DINOv2-small and DINOv2-base (Oquab et al., 2024). The ordering holds on every axis for all three encoders (5/5, 3/3, 3/3 seeds); the in-region gap is non-monotone with encoder size, so we make no scaling claim. We then evaluate ResNet-18 and frozen-DINOv2 pairs, trained on nuScenes, on Occ3D-Waymo (Sun et al., 2020), DAIR-V2X (Yu et al., 2022) and OPV2V (Xu et al., 2022) at matched coverage. Of 32 domain×encoder×coverage cells, 31 are positive for all three seeds and the last for two of three. DINOv2 retains 50% of in-domain

RayIoU on Waymo against ResNet-18's 31%, similarly for both regimes: encoder robustness and supervision coverage are distinct effects (Appendices J–F).

A second, depth-free lifting mechanism. Three encoders inside one architecture answer a weaker question than they appear to: all three feed the same lift–splat head, which predicts a per-pixel depth distribution and scatters features *forward* into the grid. We pre-specified a second family that lifts the opposite way — every voxel projects into each camera and samples the feature it lands on, with no depth prediction anywhere in the model — holding the encoder, decoder, grid, loss and training recipe fixed, so the two families differ in how features reach 3D and in nothing else (Appendix I). The effect transfers (Table 1, bottom block): $+\mathbf{0.0771}$, scene-clustered 95% CI $[+\mathbf{0.0711}, +\mathbf{0.0831}]$, 3/3 seeds, `obs_future` ahead in **146 of 150 scenes**. The absolute gap is smaller than lift–splat's, but *relative to scene-to-scene variation the two are the same size*: 1.98 per-scene standard deviations here against 1.93 there. Coverage is unchanged (-0.0088, interval spanning zero) while precision rises $+0.0650$, so increased conservatism cannot explain the gain in this family. Both variants still decode a dense voxel volume with the same 3D CNN, so transfer to sparse-query heads remains untested.

6.6 R_{future}-SUPERVISED OCCUPANCY AS A VERIFIER OF GENERATED WORLDS

Everything so far asks what supervising R_{future} does to a model's own predictions. It also changes what the model can *read*. This section does not measure planning gain; it asks whether the supervision produced a *transferable geometric readout*, and should be judged on those terms. Appendix O says why a generative model cannot be our label source even though it can be the object of measurement.

Setup. Cosmos 3 Edge (NVIDIA et al., 2026) synthesises $2.0\,\text{s}$ of future camera observations from one nuScenes frame under (i) the ego's true trajectory and (ii) another pick's. Every generated frame is mapped back to 3D occupancy by the *same* architecture trained under either `obs` or `obs_future`, and scored against real geometry inside the front frustum at a matched predicted-occupied rate. **No number here is a video metric**; the arms differ only in which region supervised the occupancy model.

The supervision transfers. On identical rollouts, an `obs_future`-supervised reader recovers about **twice** the action-conditioned signal of an `obs`-supervised one: a true-action rollout agrees with the realised future better than a counterfactual one by $+0.045$ versus $+0.019$ and $+0.041$ versus $+0.020$. The paired differences are $+0.026$ ($p = 0.0003$) and $+0.021$ ($p = 0.026$), or $1.9\times$ and $1.7\times$ after normalising by each reader's ceiling. Every interval excludes zero across two generation seeds, two readers, four horizons and two checkpoints. The effect also appears in R_{obs}, with a region interaction spanning zero: the supervision changes the model's geometry beyond the region where it was applied (Appendix K).

The reader can choose, where no reference image exists. At a counterfactual pose no camera ever stood, so the obvious ranker is unavailable: a grayscale correlation against the realised frame is as good as either reader where such an image exists, but it needs an image *from the pose being judged*, and at a counterfactual pose there is none. A prompted Cosmos-Reason1-7B critic and three DINOv2 consistency signals reach at most $+0.0027$, with intervals spanning zero (Table 2); geometry observed elsewhere in the log still supplies a reference. Ranking four rollouts of one counterfactual action on one region and evaluating the choice on a *disjoint* region with an *independently trained* reader recovers **24–31% of the attainable best-of-four gain, about 44 additional correctly located occupied voxels per frame**. The gain grows with candidate count at constant efficiency and with verifier margin. Still, 28.2% of top-quartile picks lose to the candidate mean, and the cross-fitted contrast against the best fixed-seed policy spans zero.

What this is, and is not. For the same commanded action, Occ-`obs_future` selects generated observations implying more accurate *static* 3D geometry — not a safer situation or a better video. Dynamic voxels are excluded, the reference is an offline whole-scene map, and the generator is not fine-tuned. Our planner probe picks the same anchor on 96% of decisions, leaving downstream utility unresolved (Appendix L). Earlier augmentation with generated frames was null: generated

Table 2: Selecting among four rollouts of one counterfactual action: 18 scenes, balanced over 8 crossed cells (lateral versus range split $\times$ both region directions $\times$ both reader directions), p two-sided exact scene-level sign-flip. † fits a nuisance quantity on other scenes, so it carries a bootstrap interval instead. Appendix K has the remaining baselines and controls.

ranker	gain	95% CI	exact p
Occ-`obs_future` verifier	$\mathbf{+0.0171}$	$[+0.0096, +0.0256]$	$\mathbf{0.0004}$
seed-centred†	$+0.0145$	$[+0.0081, +0.0225]$	—
Occ-`obs` verifier	$+0.0097$	$[+0.0035, +0.0160]$	0.0071
fixed generation seed, cross-fitted†	$+0.0021$	$[-0.0028, +0.0071]$	—
Cosmos-Reason1-7B prompted critic	$+0.0027$	$[-0.0062, +0.0118]$	0.58
DINOv2 temporal / medoid / spatial	$\leq +0.0021$	spans zero	≥ 0.48
Occ-`obs_future` − Occ-`obs`	$+0.0075$	$[+0.0024, +0.0134]$	0.0103
Occ-`obs_future` − strongest baseline	$+0.0144$	$[+0.0046, +0.0246]$	0.0136

frames can be audited, but cannot label themselves because no independent evidence can contradict them.

7 Limitations

R_{future} inherits the logged ego trajectory and is concentrated in near, forward space; stopped scenes resolve less, and no labels exist off-policy or in R_{never}. It therefore measures completion around the realised path, not arbitrary open-world geometry. The labels and the world-model verifier also exclude dynamic actors. Future LiDAR is available when constructing a dataset, not to an online vehicle.

The released-model frontier is nuScenes-only, cross-dataset magnitudes are not comparable, and both lifting families retain a dense voxel decoder. We do not fine-tune Cosmos 3 Edge, and downstream planning utility remains unresolved.

8 Conclusion

Future observations are not only prediction targets; they are delayed measurements of geometry hidden at the query frame. R_{future} makes this operational: later ego sweeps label part of the unobserved volume, while never-measured space remains outside the claim.

Using those labels improves held-out occupancy completion by 0.1156 without reducing standard metrics, across the tested datasets, encoders and lifting mechanisms. The resulting occupancy reader is more sensitive to the action in a generated video and can rank rollouts without a reference image. Future sensor data thus grounds both occupancy learning and world-model evaluation and offers a candidate reward for world-model post-training. Applying that reward and establishing a closed-loop benefit remain future work.

Reproducibility statement

The supplementary material contains the code behind every experiment and figure, with a manifest mapping each reported number to the command that produces it. Its first tier recomputes the headline interval, the second-architecture interval and the selection result from shipped per-frame metrics, without a GPU or the dataset; the other tiers regenerate those metrics and the figures from checkpoints and data, and retrain each arm. The appendix documents data processing, operating-point matching, scene-clustered intervals, controls and pre-specified decisions. We will release the R_{future} labels and checkpoints.

AI use statement

Generative AI assisted hypothesis refinement, experimental and statistical design, code implementation and refactoring, result interpretation, literature search, and drafting and editing. The authors

reviewed all AI-assisted work; reported results were recomputed from saved data or checkpoints, and citations were manually checked. We take responsibility for the final content.

REFERENCES

Ben Agro, Quinlan Sykora, Sergio Casas, Thomas Gilles, and Raquel Urtasun. UnO: Unsupervised occupancy fields for perception and forecasting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Chenxu Dang, Jie Wang, Guang Li, Zhiwen Hou, Zihan You, Hangjun Ye, Jie Ma, Long Chen, and Yan Wang. SparseOccVLA: Bridging occupancy and vision-language models via sparse queries for unified 4D scene understanding and planning. *arXiv preprint arXiv:2601.06474*, 2026.

Wanshui Gan, Fang Liu, Hongbin Xu, Ningkai Mo, and Naoto Yokoya. GaussianOcc: Fully self-supervised and efficient 3D occupancy estimation with gaussian splatting. *arXiv preprint arXiv:2408.11447*, 2024.

Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. arXiv:2504.01941.

Haisong Liu, Yang Wang, Yuan Chen, Jiawei Zhu, and Limin Wang. Fully sparse 3d occupancy prediction. In *European Conference on Computer Vision (ECCV)*, 2024a.

Ruixun Liu, Lingyu Kong, Derun Li, and Hang Zhao. OccVLA: Vision-language-action model with implicit 3D occupancy supervision. *arXiv preprint arXiv:2509.05578*, 2025.

Xinhao Liu, Moonjun Gong, Qi Fang, Haoyu Xie, Yiming Li, Hang Zhao, and Chen Feng. LiDAR-based 4D occupancy completion and forecasting. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024b. arXiv:2310.11239.

NVIDIA et al. Cosmos 3: Omnimodal world models for physical AI. *arXiv preprint arXiv:2606.02800*, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

Jenny Schmalfuss, Despoina Paschalidou, Simon Gerstenecker, German Ros, and Jose M. Alvarez. CASCADE: A spatio-temporal-causal reasoning representation and dataset for driving. *arXiv preprint arXiv:2609.07094*, 2026.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Sheng Zhao, Shuyang Cheng, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Jiabao Wang, Qiang Meng, Liujiang Yan, Ke Wang, Qibin Hou, and Ming-Ming Cheng. OPUS: Occupancy prediction using a sparse set. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Ruoyu Wang, Jingke Wang, Yukai Ma, Yuehao Huang, Shuangming Lei, Guanglin Xu, Aixue Ye, and Yong Liu. SparseWorld: Enhancing end-to-end autonomous driving via world models with sparse scene representation. *arXiv preprint arXiv:2605.24354*, 2026.

Julong Wei, Shanshuai Yuan, Pengfei Li, Qingda Hu, Zhongxue Gan, and Wenchao Ding. OccLLaMA: An occupancy-language-action generative world model for autonomous driving. *arXiv preprint arXiv:2409.03272*, 2024.

Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. OPV2V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

Baijun Ye, Minghui Qin, Saining Zhang, Moonjun Gong, Shaoting Zhu, Zebang Shen, Luan Zhang, Lu Zhang, Hao Zhao, and Hang Zhao. GS-Occ3D: Scaling vision-only occupancy reconstruction with gaussian splatting. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. arXiv:2507.19451.

Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo, Hanyu Li, Xing Hu, Jirui Yuan, and Zaiqing Nie. DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Zichen Yu, Changyong Shu, Jiajun Deng, Kangjie Lu, Zongdai Liu, Jiangyong Yu, Dawei Yang, Hui Li, and Yan Chen. FlashOcc: Fast and memory-efficient occupancy prediction via channel-to-height transformation. *arXiv preprint arXiv:2311.12058*, 2023.

Xin Zhou, Zongchuang Zhao, Zhibo Yang, Mingsheng Li, Humen Zhong, Shuai Bai, Du Chu, Ruizhe Chen, Zhaohai Li, Jun Tang, Qiuyue Wang, Mingkun Yang, Jiazhao Zhang, Dayiheng Liu, Dingkang Liang, and Xiang Bai. Qwen-Drive-1.0: An initial step towards a vision-language foundation model for autonomous driving. *arXiv preprint arXiv:2609.00111*, 2026.

A QUALITATIVE FRAMES

Figure 5 accumulates the comparison of Figure 4 along a drive. Figure 6 projects the hidden volume into the camera image as an overlay; the supplement contains further label-selected scenes at larger scale and full-scene videos.

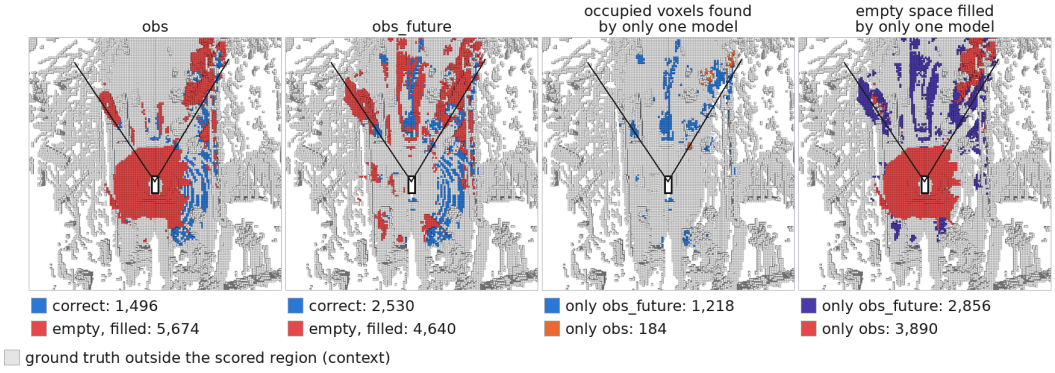


Figure 4: **The frame of Figure 1, scored on the hidden region R_{future} only.** Each model names its 7,170 most confident voxels there, the number the later sweeps found occupied, so precision equals recall; the key under each panel gives its colours and counts. First two panels: correct (blue) and empty space filled (red). Third: occupied voxels found by only one model. Fourth: empty space filled by only one model; the red disc around the ego is `obs`'s. Because both name the same number, the last two panels differ by the same amount, $+1{,}034$ for `obs_future`. Light grey is ground truth outside the scored region. Figure 3 gives the comparison over all frames.

B THE FRONTIER, SWEPT; AND THE LEAD-TIME AXIS, RANGE-CONTROLLED

Two measurements that a single operating point and a pooled average respectively hide. Both are over the same 293 validation frames, through one code path, at `base_occ` 0.237.

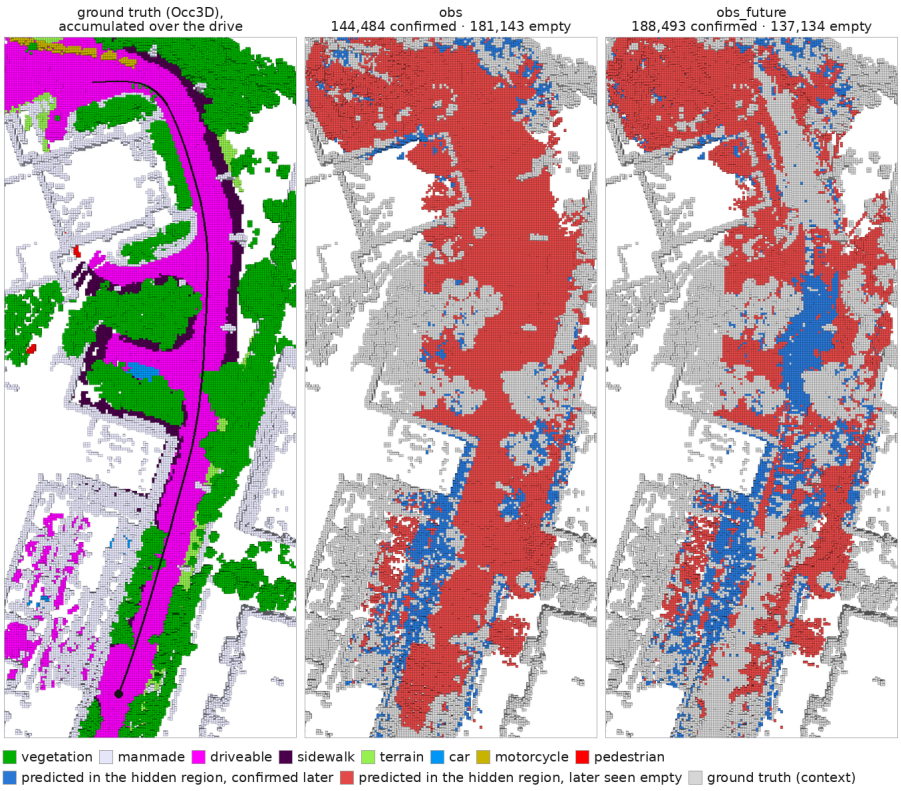


Figure 5: **A 126×53 m map built along one drive.** Scene-0919, the highest-ranked gallery scene in which the ego travels at least 100 m; all 41 keyframes, placed in world coordinates, the ego path drawn on the left. Left: Occ3D ground truth, each voxel's majority class over the frames that see it. Middle and right: every voxel each model commits in a frame's hidden region (same scoring as Figure 4), blue where the later sweeps found it occupied, red where they found it empty; grey is ground-truth context. Summed over the frames, `obs` commits 144,484 confirmed voxels and 181,143 empty ones (precision 0.444), `obs_future` 188,493 and 137,134 (0.579). `obs` lays a continuous phantom corridor along the route, the near-field sheet of Figure 4 repeated at every frame. The top view shows each column's uppermost voxel, so an elevated false positive hides whatever lies beneath it; this is why `obs_future`'s map also looks mostly red though it is the more precise.

The ranking depends on where you stand. Sweeping each model's own score rather than reading its argmax, the six curves *cross*. OPUSv2 owns the sparse end and is last at coverage 0.60; FusionOcc and FlashOcc are the two weakest at 0.05 and the two strongest at 0.60. So **which occupancy model is better" in this region is not a property of the models** — it is a property of the operating point the question is asked at, and an argmax comparison picks one arbitrarily.

precision at coverage	0.05	0.10	0.20	0.30	0.45	0.60
Qwen-Drive-1.0	0.746	0.678	0.580	0.497	0.402	0.336
OPUSv2	0.782	0.751	0.640	0.503	0.376	0.312
FlashOcc	0.510	0.540	0.547	0.527	0.456	0.376
FusionOcc†	0.548	0.568	0.577	0.549	0.467	0.380
FlashOcc + mask	0.646	0.633	0.585	0.516	0.418	0.359
GaussianOcc	0.621	0.535	0.445	0.398	0.369	0.347

Table 3: Precision inside R_{future} at six target coverages. The curves cross between 0.20 and 0.30. †FusionOcc did not reproduce its published mIoU and is a sensor-family data point only.

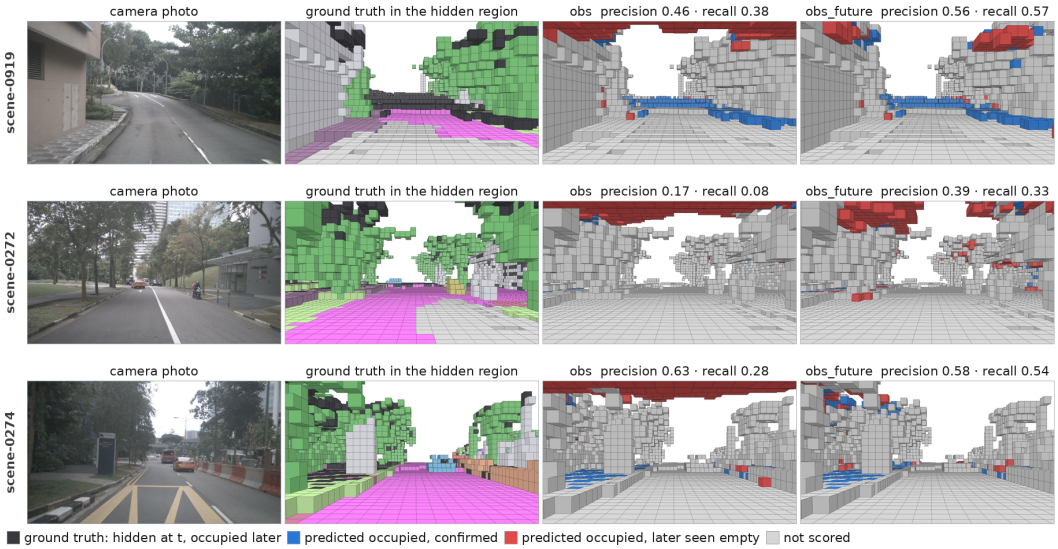


Figure 6: Three further frames in the front camera's own view: the next three ranks, one per scene, by visible above-ground R_{future}, chosen before either model was loaded. Voxels are rendered as 0.4 m cubes through the camera's intrinsics and extrinsics with occlusion. Second column: what the ego's own later sweeps found occupied in the hidden region (dark), over pale Occ3D context. Last two: each model's committed voxels in the hidden region, confirmed (blue) or later seen empty (red), on grey unscored context; each model names as many voxels as the later sweeps found, over the whole region, so the numbers reported here are for the subset inside this camera's view. `obs_future` recovers far more of the hidden structure (recall $0.38 \to 0.57$, $0.08 \to 0.33$, $0.28 \to 0.54$); precision in this view rises in two frames ($0.46 \to 0.56$, $0.17 \to 0.39$) and falls in the third ($0.63 \to 0.58$). The two models' errors differ in *where* they fall rather than how high: `obs` lays a flat sheet of false positives across the top of every frame, whereas `obs_future`'s cluster against real canopy, and in these three frames the median false-positive heights are similar (1.6 vs 1.2 m, 4.4 vs 4.4 m, 4.0 vs 4.0 m). Over every scored frame they are not: `obs`'s false positives sit markedly higher (Figure 3c).

A pooled lead-time curve is mostly a range curve. Recall falls monotonically with lead time for every one of the six when ranges are pooled — and the mean range of the bins grows from 12.3 m to 29.2 m across them, because a far surface is resolved later. Holding range fixed, most of that decay dissolves: OPUSv2's -0.266 becomes -0.013 in the far ring, and FlashOcc and FusionOcc turn *positive*. Only Qwen-Drive keeps a substantial decay in all three rings. **The pooled reading would have credited five of six models with a temporal effect that is a distance effect.**

recall change, 0–1 s to >7 s	pooled	0–15 m	15–25 m	25–40 m
Qwen-Drive-1.0	-0.372	-0.339	-0.214	-0.223
OPUSv2	-0.266	-0.165	-0.078	-0.013
FlashOcc	-0.137	+0.052	+0.028	+0.165
FusionOcc†	-0.043	-0.022	+0.029	+0.283
FlashOcc + mask	-0.258	-0.112	-0.023	-0.012
GaussianOcc	-0.351	-0.144	-0.165	+0.013

Table 4: The same quantity pooled over range and inside three fixed rings, at a matched predicted-occupied rate of 0.20 inside R_{future}.

C DECOMPOSING THE WILLINGNESS GAP

Inside the driving corridor — R_{future}, at least 10 m from the ego, within 5 m of its own future path — the sparse model commits a fifth as much as the dense ones (0.75 M against 2.60–3.65 M over-commitments) and is right about twice as often (corridor precision 0.645 against 0.327–0.353).

Two *different* failures sit underneath one number. The dense camera models' corridor errors are ∼80% road surface, and a height histogram puts 99.9% of them below +2.2 m with 79% in the bottom two voxel layers: they place the road at the wrong height rather than filling the driving volume. We initially read the 80% road share as a thickness bias extending upward, and the height check we added to confirm it refuted it instead. GaussianOcc's failure is genuinely structural: 61% of its corridor errors are `manmade` at 1.96× over-representation.

The ranking also inverts by class. Comparing precision at matched coverage on each model's own confidence curve, GaussianOcc — weakest in aggregate — is the strongest on the road class at low coverage (0.954 against OPUSv2's 0.909 at coverage 0.020), while on `manmade` it is 2–4× worse than both others. And FlashOcc's confidence is nearly *uninformative* on the road class: sweeping its threshold across 0.00–0.99 moves precision only 0.335 → 0.383, where GaussianOcc moves 0.485 → 0.975.

We report the decomposition as a diagnostic and build no method on it. The natural next step — route regions to whichever model is specialised there — has no non-circular implementation we could find: routing by class presupposes the class being predicted, and routing by height fails because the two failure modes occupy the same heights (45% of GaussianOcc's `manmade` errors are in the bottom voxel layer, alongside 31% of FlashOcc's road). We tested it and report it as a negative rather than leaving it as future work.

D WHERE THE BENEFIT LANDS, AND LABEL TRUST

Binning R_{future} by distance to the ego's own realised future path, the advantage decays monotonically: +0.257 precision at 0–2 m, +0.229 at 2–5 m, +0.132 at 5–10 m, +0.042 at 10–20 m. Distance to the path correlates with distance to the ego, so we bin *jointly* and read in lift; the effect survives in the two inner rings and does not beyond 20 m. Weighting the loss by path proximity does approximately nothing (largest band movement +0.008), and we report that as a negative result rather than omitting it.

Label trust. R_{future} has no ground truth by construction. A cooperating agent supplies one that shares none of the temporal method's failure modes: on OPV2V our labels agree with a second agent's direct observation at $\kappa = 0.778$, against 0.477 for a 90°-rotated control and 0.329 for an unrelated scene.

A second reference comes from a fact rather than a sensor: **the ego drove down its own future trajectory, so the space its body passed through was free**. Building that swept volume and restricting to voxels a drift classifier calls static, **in the median scene our labels place exactly zero occupancy there**; pooled the rate is 0.0045, the IQR is 0–0.003, and 2.3% of scenes exceed 0.05. We report the distribution because the pooled number alone is not stable: a 279-frame subsample reads 0.0136 and a 38-frame one 0.1585, both single scenes from the tail.

E THE CLEARANCE SWEEP, IN FULL

The closed-loop consequence is not hypothetical, though it is not our result: a sibling study wired occupancy into exactly such a brake and **lost 47% of route progress to phantom braking where ground-truth geometry lost 3%**. No planner runs here and no vehicle moves. Its model scores normally on the benchmark (0.302 mIoU) yet reaches exactly one operating point, because the integration is `argmax` plus a class filter, and there it declares **a third of forward directions clear when an obstacle is there**. The integration and the model are separate problems, and only the first is a configuration change.

model	coverage	precision	lift
Qwen-Drive-1.0 (VLM head, 2026)	0.183	0.601	2.53
OPUSv2 (sparse query)	0.252	0.603	2.54
FlashOcc (dense camera)	0.351	0.532	2.24
FusionOcc† (camera + LiDAR)	0.395	0.519	2.19
FlashOcc + visibility mask	0.474	0.403	1.70
GaussianOcc (self-sup.)	0.536	0.361	1.52

Table 5: Six models in R_{future}, ordered by willingness, all scored through one code path on the same 293 frames. `base_occ` is 0.237; lift is precision over that. Adding a sensor modality (FusionOcc) or a 4B VLM backbone (Qwen-Drive) moves a model *along* this curve rather than off it. The one departure from monotonicity is Qwen-Drive itself, which commits to 27% fewer voxels than OPUSv2 and is 0.002 less precise — inside the curve rather than beyond it, by a margin well within noise at this sample size. †FusionOcc did not reproduce its published mIoU (full-volume 0.127 against 0.566) and is included only as a sensor-family data point, never as an accuracy claim.

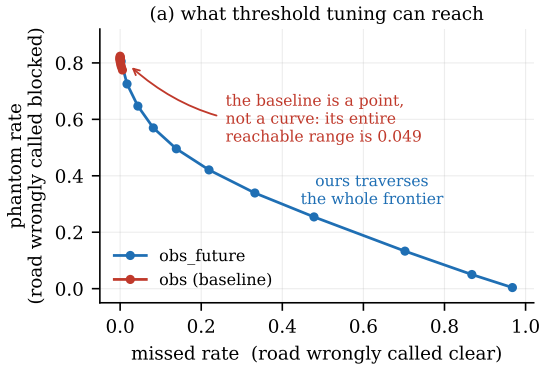


Figure 7: Sweeping each arm's *own* threshold over its whole range, three seeds. The baseline is a point, not a curve: no threshold moves its forward brake off "four fifths of the corridor is blocked". Per-threshold numbers are in Table 6.

Five of the six published models. Running the same sweep on the five models whose released artefacts expose a per-voxel confidence — Qwen-Drive's does not, so it is absent here — four traverse a usable range (OPUSv2 0.161, FlashOcc 0.176, FlashOcc+mask 0.185, FusionOcc† 0.152) against 0.049 for our `obs`. Only GaussianOcc is immovable (0.046), and it is self-supervised. What the immovable models share is *dense commitment with uninformative confidence* — GaussianOcc predicts 79% of the grid occupied with a near-constant confidence, where OPUSv2 commits to 6.2% with confidence spread over 0.29–0.83 — so this is a calibration failure our baseline exhibits and most deployed models do not. Gating the deployed model's decision on its own confidence recovers a range (missed 0.000–0.129) but not a better operating point.

F Cross-domain lifts, per cell

Four nuScenes-trained arms — {ResNet-18, frozen DINOv2} × {`obs`, `obs_future`} — on three out-of-domain sets whose shifts differ in kind: Occ3D-Waymo (Sun et al., 2020), DAIR-V2X (Yu et al., 2022) (one forward camera, different country) and OPV2V (Xu et al., 2022) (CARLA; real to simulated). Three seeds each, **compared at matched coverage**, because out of domain the arms' operating points diverge far more than in domain — their coverage ratio is 1.1× on nuScenes, 0.43× on DAIR and 1.6× on OPV2V, in *opposite* directions, which is the signature of an operating-point artefact rather than a model property.

On DAIR-V2X the `obs_future` lift over `obs` at matched coverage 0.02–0.08 is +0.863 to +0.416 for ResNet-18 and +0.351 to +0.257 for DINOv2; on OPV2V it is +0.095 to +0.144 and +0.096 to +0.149. Across the two sets that is 32 cells of (domain × encoder × matched coverage), and **31 are positive on all three seeds**; the remaining one — ResNet-18 on OPV2V at the lowest commitment

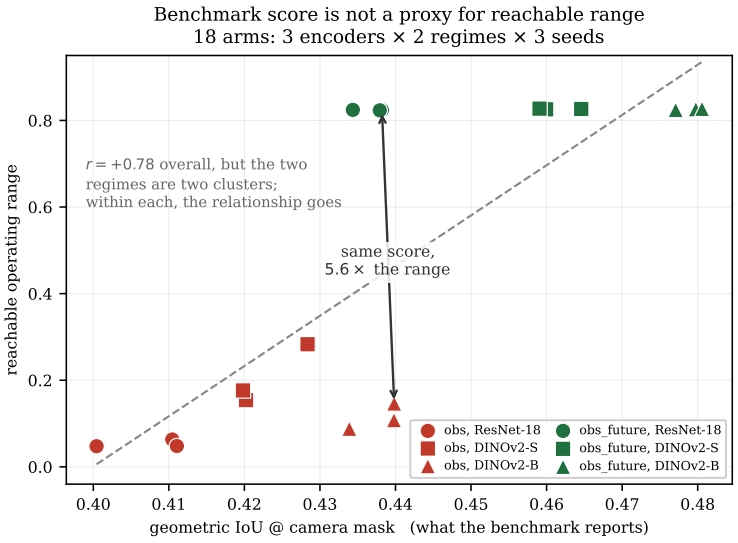


Figure 8: Benchmark score against reachable operating range, for every arm we trained (three encoders, two regimes, three seeds). The two regimes form two clusters, which is why the aggregate correlation is $r = +0.78$; within each cluster the relationship goes. The arrow marks the pair quoted in §6.4 — DINOv2-base with `obs` at 0.4398 geometric IoU and range 0.146, against ResNet-18 with `obs_future` at 0.4382 and range **0.824**: the same benchmark score to within 0.4%, 5.6× the range. We draw the regression line rather than omit it, because it is what makes the claim the weaker one: the benchmark is unusable as a proxy for range, not blind to it.

threshold	0.10	0.50	0.90	0.98
`obs` missed	0.0000	0.0000	0.0015	0.0057
`obs` phantom	0.8238	0.8139	0.7943	**0.7749**
`obs_future` missed	0.0025	0.1386	0.7025	0.9675
`obs_future` phantom	0.8057	0.4957	0.1330	**0.0039**

Table 6: Sweeping each arm's own threshold over its whole range, three seeds. `obs` declares four fifths of the forward corridor blocked at *every* operating point.

measured anywhere — is positive on two of three. **"Supervising the region does not transfer" is false.**

The encoder is what buys domain robustness. Retained fraction of in-domain RayIoU on Occ3D-Waymo: ResNet-18 **31%**, DINOv2 **50%**, nearly identically for both supervision regimes. Encoder quality and supervision region do different jobs — the first survives the domain change, the second is what the benchmark cannot see — and they are partly redundant rather than competing. **DINOv2 with `obs_future` is the best arm in every setting we measured**, in domain and out.

G HEADLINE INTERVAL: ALL COMPONENT RATES

Every row below comes from the single estimator of §6.1 applied to the same five retrained pairs and the same bootstrap draws, so the intervals are mutually consistent rather than five independent resamplings. Scenes are drawn with replacement with repeats kept; a scene counts as won when the five-seed mean gap on its own frames has the predicted sign.

	obs	obs_future	gap	95% CI	scenes
coverage	0.4872	0.4211	-0.0662	$[-0.0754, -0.0564]$	127/150
precision	0.4225	0.5368	$+0.1143$	$[+0.1039, +0.1249]$	144/150
hallucination	0.2777	0.1964	-0.0813	$[-0.0897, -0.0726]$	131/150
miss	0.0532	0.0381	-0.0151	$[-0.0171, -0.0133]$	144/150
R_{future} F1	0.5489	0.6611	$\mathbf{+0.1122}$	$[\mathbf{+0.1028}, \mathbf{+0.1219}]$	**147/150**

Per-seed F1 gaps $+0.1192$, $+0.1103$, $+0.1084$, $+0.1111$, $+0.1119$; seed sd 0.0042. The 151 frames excluded from the 6019 lack a scorable R_{future} region at the minimum-voxel threshold. Two orderings are wrong here and both were live in our own code before being caught: averaging per-frame F1 rather than forming F1 from aggregated rates turns $+0.116$ into $+0.011$, and dropping bootstrap multiplicity narrows an interval until it excludes zero. Both are covered by tests in the released code.

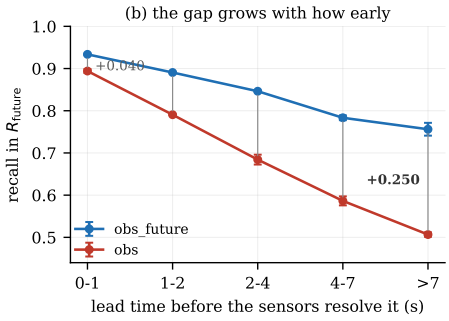


Figure 9: Recall in R_{future} against how long the sensors take to resolve the voxel, at a matched predicted-occupied rate of 0.45; error bars are the spread over three seeds. The gap is smallest where the sensors were about to answer anyway.

H Where the gain lives: full map

In Figure 3, the gain in (b) divides 4.2% inside 10 m, 42.0% from 10 to 20 m, 28.7% from 20 to 30 m and 25.1% beyond 30 m: it is not a near-field artefact. Of the $24{,}141$ columns either model ever commits in, 90.2% are positive, 3.5% negative and 6.3% exactly zero, with positive mass $64\times$ the negative; net, `obs_future` finds $+774$ more occupied voxels per frame. Pooled over all frames, `obs` fills $2{,}402$ empty voxels and finds $1{,}017$ occupied ones per frame (precision 0.297), `obs_future` $1{,}629$ and $1{,}790$ (0.524); 74% of `obs`'s false positives lie within 10 m of the ego, against 15% of `obs_future`'s. These figures score structure centred at least 0.4 m above the road, the same region as Figures 1–6; the voxel layer centred at 0.0 m spans -0.2 to 0.2 m and so contains the road surface itself.

I Second architecture family: design, pre-specification, full table

What differs. `lss` predicts a per-pixel depth distribution, takes its outer product with the context feature and scatters the resulting frustum into the voxel grid — information is *pushed* outward. `proj` holds a fixed reference point at every voxel centre, projects it into each camera, bilinearly samples the feature it lands on and averages over the cameras whose image it fell inside — information is *pulled* inward, and no depth is predicted at any point. This is BEVFormer/SurroundOcc spatial cross-attention in its simplest non-deformable form. Because sampling alone cannot separate two voxels on one ray, which read the same pixel, a *fixed* (not learned) Fourier encoding of the voxel centre and the per-voxel camera-visibility count are concatenated before the decoder. Held identical to `lss`: frozen DINOv2-B features, the same `VoxelDecoder`, grid, loss and class weighting, optimiser, LR schedule, epoch budget, stride and seeds. Trainable parameters differ by construction (4.27 M vs 3.64 M, $+17\%$) because the decoder takes the encoding and count as extra input channels; we report this rather than equalising it, since equalising would mean tuning one family's width against the other's.

Pre-specification. We write "pre-specified", not "pre-registered": the design document was written to the repository before the first training run began, but it carries no immutable external timestamp, so a reader cannot verify the ordering independently. Fixed before any arm was trained: the primary quantity is the three-seed mean R_{future} F1 gap within the `proj` family; it *supports generality* if the mean is positive, all three seeds agree in sign and the mean is at least $+0.03$; it *fails* if the mean is ≤ 0 or the seeds disagree, in which case the paper's claim is narrowed to lift–splat architectures and the negative result becomes the headline limitation. The $+0.03$ bar is far below the lift–splat effect on purpose: the claim under test is that the effect is not an artefact of the lifting mechanism, not that it has the same magnitude everywhere. No `proj` hyperparameter was tuned against any R_{future} number.

Result. Supports generality on all three conditions.

	`obs`	`obs_future`	gap	95% CI	scenes
coverage	0.4399	0.4311	-0.0088	$[-0.0173, +0.0008]$	101/150
precision	0.4604	0.5254	$+0.0650$	$[+0.0580, +0.0718]$	133/150
hallucination	0.2344	0.2055	-0.0288	$[-0.0359, -0.0212]$	121/150
miss	0.0572	0.0371	-0.0201	$[-0.0225, -0.0179]$	144/150
R_{future} F1	0.5758	0.6529	$\mathbf{+0.0771}$	$[\mathbf{+0.0711}, \mathbf{+0.0831}]$	**146/150**

Per-seed gaps $+0.0937$, $+0.0777$, $+0.0599$; seed sd 0.0169, so the effect is $4.6\times$ it. Across the 150 scenes the per-scene gap is $+0.0749 \pm 0.0378$, i.e. 1.98 standard deviations, against $+0.1098 \pm 0.0569$ (1.93) for lift–splat: the two families differ in the absolute size of the effect but not in its size relative to the variation it has to overcome. 117 of 150 scenes exceed $+0.05$. In voxel terms, at the average scored region of $\sim$28,000 voxels per frame, the supervised arm locates $\sim$675 more occupied voxels and commits $\sim$920 fewer voxels that are in fact free, raising recall of occupied space from 0.771 to 0.862.

A second observation, not pre-specified and reported as such: the two supervised arms land within 0.008 of each other (0.6529 here, 0.6611 for lift–splat) from baselines 0.027 apart and through opposite lifting mechanisms, while the observed-only arms stay architecture- and seed-dependent (R_{future} F1 seed sd 0.0163 for `obs` against 0.0006 for `obs_future` in this family). Two architectures cannot establish that as a law. Same estimator as §6.1: scenes drawn with replacement with repeats kept, one draw applied to all paired seeds, rates aggregated within a seed before F1 is formed, gaps averaged across seeds. Coverage is the one row whose interval spans zero and is reported as such.

Secondary check (pre-specified, non-gating). Standard metrics must not fall inside the family. They rise, on 3/3 seeds for both: geometric IoU at the camera mask $0.4254 \rightarrow 0.4539$ ($+0.0286$; per-seed $+0.0309$, $+0.0275$, $+0.0272$) and RayIoU $0.2452 \rightarrow 0.2569$ ($+0.0116$; $+0.0155$, $+0.0063$, $+0.0131$). The geometric IoU gap is $+0.0286$ here against $+0.0287$ for lift–splat; we note the coincidence and draw nothing from it, since with three and five seeds the agreement sits well inside the seed spread. Note that `proj`'s *absolute* RayIoU (0.2452) is far below `lss`'s (0.3234): a depth-free lift yields weaker ray-surface structure. That is expected, is why the gating comparison is within-family, and is why `proj` should not be read as a competitive occupancy model. It is a controlled second lifting mechanism, and the pre-specification committed to not tuning it into anything else.

Correctness of the new lift. The projection geometry is the only place a silent error could hide, so it is pinned against the family that already works rather than a re-derivation, which would agree with itself while both were wrong. `lss`'s frustum map from (u, v, d) to ego coordinates, composed with `proj`'s ego-to-(u, v) map, is the identity to 10^{-2} px for cameras with a real rotation and nonzero translation. Voxels behind the lens are dropped rather than sampled at their mirrored projection; voxels no camera saw receive exactly zero rather than `grid_sample`'s clamped border feature; and a backward pass confirms no gradient reaches any depth parameter.

in-domain gap	ResNet-18 (5 seeds)	DINOv2-S (3)	DINOv2-B (3)
geometric IoU	+0.0287	+0.0384	**+0.0413**
RayIoU	+0.0068	+0.0097	**+0.0113**
R_{future} F1	+0.1156	+0.0953	+0.1043
R_{future} prec.	+0.1100	+0.0853	+0.1010
abs. geom. IoU (`obs_future`)	0.4368	0.4612	**0.4791**
abs. RayIoU (`obs_future`)	0.3301	0.3614	**0.3821**

Table 7: Three encoders. On both benchmark axes the gap *grows* monotonically with encoder strength while absolute performance also rises.

J ENCODER COMPARISON, FULL TABLE

K READING A WORLD MODEL: FULL RESULTS

§6.6 states the two findings this section supports. Everything else we measured about Cosmos 3 rollouts is here.

Robustness of the action-fidelity contrast. The turning contrast is positive with the interval excluding zero in all four cells of generation seed × reader seed ($+0.045$, $+0.041$, $+0.031$, $+0.031$); at every horizon from 1 to 4 s ($+0.023$, $+0.045$, $+0.046$, $+0.049$), where it saturates after 2 s although the price of generation keeps climbing ($D_{R_{\text{future}}}$ rises from 0.054 to 0.167); under three motion-gate settings; with the region's boundary shell eroded ($+0.055$); and on a second, larger checkpoint, Cosmos 3 Nano ($+0.054$, $+0.050$). Edge and Nano are two checkpoints of one family, not two architectures, and their paired difference is $+0.009$, CI $[-0.017, +0.044]$ — no evidence that capacity helps. The scene-level primary estimate is $+0.052$ (14 of 18 scenes, exact $p = 0.002$); on straight driving it is $+0.015$ ($p = 0.022$), so the signal is present in both regimes and about $3.5\times$ stronger on turns.

Not only camera placement. Scoring each arm against a static map at *its own* commanded pose, with base rates equalised by construction, the gap survives against a borrowed trajectory ($+0.048$, 12 of 18 scenes, exact $p = 0.014$) and does not reach significance against a reflected one ($+0.027$, $p = 0.16$). The measure earns that reading only because it detects displacement at all: front-frustum restricted it reaches $3.2\times$ its base rate and falls monotonically as the grid is displaced, 0.444 to 0.391 over 10 m.

An explanation we withdrew. Under the borrowed construction the gap tracks how far the two actions diverge in *angle* ($r = +0.202$, CI $[+0.022, +0.381]$) and not how far apart their endpoints land ($r = +0.031$) nor how much distance differs ($r = +0.013$); the two regimes are already matched on endpoint separation (6.17 against 6.05 m) while differing in angle ($1.65°$ against $10.82°$). The reflected control breaks it: it diverges *more* in angle ($15.97°$ against $12.94°$) and produces half the gap, and within it angle does not predict the gap at all ($r = -0.058$). A second account — that reflections score well in laterally symmetric scenes — also fails ($r = +0.012$ against a predicted negative). We report the angle relationship as a post-hoc pattern under one construction of "a different action" and leave the mechanism open.

What one run would have told us. An earlier generation of the straight-driving arm gave $+0.017$, CI $[+0.004, +0.029]$ — four times what we now measure. Regenerating moved it to $+0.006$; two readers then agreed with each other and not with the original, and changing only the generation seed moves a turning estimate by 0.014 absolute, which is the whole of the straight-driving effect. Two generation seeds do not define a true value and we do not claim one.

Selection: baselines and scaling. Reference-free rankers are computed from the rollout alone. `nvidia/Cosmos-Reason1-7B-Reward`, which the Cosmos cookbook and the cosmos-rl reward service both reference, is not publicly downloadable, so we prompt the public

`Cosmos-Reason1-7B` as a video critic instead; counting sampled verdicts saturates (96 of 96 "plausible", 5 of 6 groups scoring identically), so we read the first answer token's YES/NO logits, as cosmos-rl itself does. Selection gain grows with the candidate pool at roughly constant efficiency — best-of-2/3/4 giving $+0.0081$, $+0.0122$, $+0.0152$ against oracles of 0.028, 0.043, 0.053 — and concentrates by the verifier's own margin, rising from $+0.0152$ at full coverage to $+0.0275$ on the most confident quarter. Confidence is not a guarantee: 28.2% of top-quartile picks still lose to the candidate mean.

The world model under-turns. Measured from the generated pixels (SIFT matches and an essential matrix; the same estimator recovers real keyframe headings to a median $0.28°$), Cosmos realises about 29% of a borrowed action's heading change over 2 s (median $5.3°$ of $11.7°$), 37% of the true action's and 9% of a mirrored one's: it follows the road in the image more than the action. Each rollout is therefore scored against the map at a pose its camera did not quite reach. All seeds of a pick share that pose, so the offset lowers their scores alike rather than favouring one, and selection does not pick the seeds that turned furthest (within-pick Spearman between ranking score and realised turn $+0.02$, $p = 0.38$; the commanded-pose gain on the same picks is $+0.0198$, 14 of 16 scenes). Scoring each seed at its own pixel-estimated pose is not a usable alternative: the gain rises to $+0.077$, but assigning each seed *another* seed's pose raises it further, to $+0.095$, so what drives it is per-seed pose variation shared by the two regions, not the frame's geometry.

L Does better selection change a decision? Undetermined

Selection is measured on geometry. Whether that reaches a decision is a separate question, and we built three probes to ask it. None could answer, and the reasons are worth recording because each failure is diagnosable.

The final, frozen probe. Following WoTE (Li et al., 2025), candidates are 64 trajectory anchors k-means clustered from 4232 real nuScenes ego futures, with the 18 evaluation scenes excluded from the fit; 25 of them traverse the scored region and the rest are dropped, since an anchor that never enters it cannot be judged there. A frozen planner reads the occupancy predicted from *one* rollout and takes the anchor it can get furthest along; the choice is then judged against the accumulated static map. The verifier ranks on the near field and the planner is scored only beyond 20 m, so selection cannot see the region its trajectory is judged in, and the map enters only at the end.

planner given	fraction of scored portion completed	collision rate
a random seed	0.2222	0.7778
the `obs`-selected rollout	0.2381	0.7619
the `obs_future`-selected rollout	0.2381	0.7619
the best available rollout	0.2460	0.7540

The probe cannot discriminate. The arms chose *different anchors on* 4% *of picks* (5 of 126); on the rest the comparison is vacuous by construction. `obs_future` and `obs` selection differ on *none* of the 18 scenes — their paired difference is exactly 0.0000 — and even an oracle changes the decision in two. This is not a null result but an instrument without the resolution to produce one. The reason is legible: the planner's output is a discrete argmax over a coarse candidate set, while the occupancy difference between rollouts is real but diffuse ($+0.017$ in held-out precision), and it almost never crosses a decision boundary. **We therefore claim neither that the geometric gain transfers to planning nor that it fails to.**

Three designs, and why we stopped at three. The first used nine hand-drawn 18 m arcs against a region scored beyond 20 m, so no corridor reached it and all four arms returned byte-identical numbers. The second used the clustered anchors but scored absolute progress, which anchor length dominates: one anchor won 122 of 126 picks and only 4 of 64 were ever chosen. The third is above. Each failure was identified from its own signature — identical outputs, a single winning anchor, near-identical choices — before any arm comparison was read, and the third was frozen before its result was seen. We stopped there: a fourth design chosen after three outcomes could not be distinguished from searching for a positive.

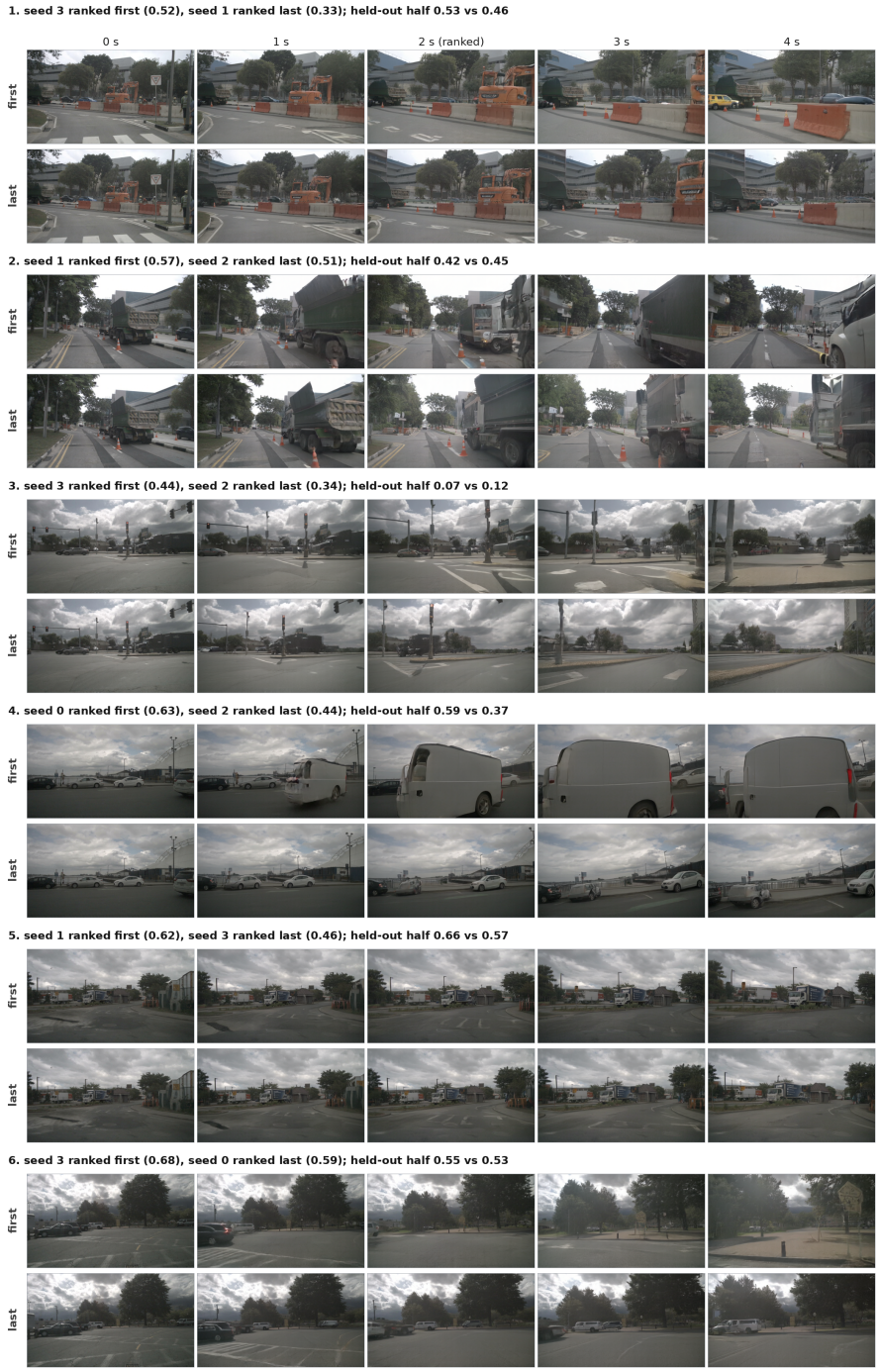


Figure 10: **Rollouts the verifier ranks first and last, drawn at random.** Six picks sampled at random from all 159 (the seed was fixed before any was viewed). Each pick is four rollouts of one counterfactual action that differ only in generation seed; shown are the one Occ-`obs_future` ranks first and the one it ranks last, front camera, 0–4 s. The ranking reads the occupancy decoded from the 2 s frame on one lateral half of the grid; each header gives that score and the score on the disjoint held-out half, which selection never sees. Over all 159 picks the first-ranked rollout beats the last-ranked one on the held-out half in 97 (7 ties), by $+0.024$ on average (95% scene-bootstrap interval $[+0.011, +0.040]$, 18 scenes); in this sample it does in four of six. Cosmos turns only about a third as far as commanded (Appendix K), which all four seeds share.

M LEARNING CURVES: WHERE THE BASELINE STOPS IMPROVING



Figure 11: **The observed-only baseline keeps learning, but not in R_{future}.** Training loss (left) falls throughout for both regimes. R_{future} precision (centre) and F1 (right) are flat for the two baseline seeds over the second half of training while the `obs_future` arm sits roughly 0.09 higher and is still rising slightly. Curves are the periodic validation probe run during training on a fixed 38-frame subset; F1 uses recall = coverage × precision/base occupancy, with base occupancy constant across steps and arms, so it scales the curve without affecting any trend. Slopes are descriptive: consecutive checkpoints are strongly correlated, so we report no significance for them, and we do not compare loss slopes between regimes, whose objectives differ.

N CLASS-STRATIFIED R_{future}, FULL TABLE

One global operating point is set inside R_{future}, matched between the two arms, and the same selection is then sliced by the FlashOcc teacher's class naming. No threshold is chosen per class. 390 validation frames carrying a teacher shard. Groups follow the Occ3D vocabulary: *vehicle* is bus, car, construction vehicle, trailer and truck; *vulnerable road user* is bicycle, motorcycle and pedestrian; *ground* is driveable surface, other flat, sidewalk and terrain; *structure* is manmade.

class group	labelled occupied voxels	`obs` F1	`obs_future` F1	gap
vehicle	46,366	0.5593	0.6615	+0.1022
vulnerable road user	5,036	0.5307	0.6391	+0.1084
barrier / cone	8,805	0.5085	0.7186	+0.2101
ground	1,790,186	0.6700	0.8114	+0.1414
structure	196,583	0.3948	0.5519	+0.1571
vegetation	441,174	0.4911	0.6249	+0.1338

All six are positive, spanning +0.102 to +0.210. An arm that had merely learned to stop hallucinating near the ego would lift ground and little else; vehicles and vulnerable road users gaining about +0.10 is what rules that reading out. Two limits travel with the table. These are the *binary* arms stratified by class, so nothing here says either arm names classes better; and the vulnerable row rests on an order of magnitude fewer voxels than any other group.

Training genuinely semantic arms was attempted and is not possible on this machine: the run stops with 28130 train frames dropped for want of a teacher shard, because the FlashOcc shards available here cover the validation split only.

O WHY NOT A GENERATIVE WORLD MODEL

Generative world models are the obvious candidate for the volume R_{never} occupies, and we did not use one. Every claim here rests on the labels being real sensor returns: it is what makes them annotation-free, what lets an independent agent validate them, and what makes "the model asserted something the world later contradicted" a meaningful sentence. Supervising R_{never} with a generative completion replaces observation with prediction in exactly the region that has no reference, so the resulting model could not be evaluated there by us or by anyone else. Using a model's output to teach a model not to hallucinate is not something we know how to make falsifiable. What would make

it tractable is to treat the generative prior as the *object of measurement* rather than the label source, and score its completion against a real observation from a viewpoint the ego never reaches — which is exactly what the cooperative instrument above provides, and what §6.6 does: it reads Cosmos 3 Edge rollouts with our own `obs_future` student and finds that they encode action identity more strongly on turns against an unrelated counterfactual action; a reflected counterfactual preserves the estimate but does not reach significance, and the geometric mechanism remains unresolved. That result is a reason to be careful with generated frames as action-conditioned supervision; it is not a reason to trust them as labels in R_{never}, where no observation exists to score them against at all.