DECISION-FLIP: BUDGET-FREE STRUCTURED PRUNING FOR GAUSSIAN OCCUPANCY MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Occupancy predictors rely on large persistent banks of Gaussians or learned queries whose size is fixed by hand. Common pruning proxies, such as opacity, activation frequency, or parameter magnitude, do not measure whether removing a row changes the model's voxel decisions. We introduce **Decision-FLIP**, a budget-free compilation framework whose scoring and structure selection need only unlabeled calibration inputs. Exact Mean-FLIP scores each persistent row by the number of voxel argmax decisions that change when the row is removed from the readout; a guarded Otsu split of the log-score distribution selects the survivor count without a target and abstains when the split is unsupported; and the survivors are physically compiled into a smaller checkpoint, so camera sampling, attention, and aggregation shrink with the bank. For streaming banks, scores are integrated over each row's lifecycle through warp, death, and refill. Across four Gaussian-bank model-dataset settings and one persistent-query transfer, the automatically selected cuts remove **12–79%** of eligible rows and cut bank/query-dependent latency by **7–45%** across settings with component profiles, with whole-model savings of up to **8.0% on the primary endpoints** bounded by fixed-cost image, voxel, and sensor modules and peak-memory savings of up to **16.6%**. Under matched one-epoch recovery controls, semantic mIoU changes by at most 0.18 point. Same-K controls show that the ranking, not only the bank size, matters: random removal collapses GaussianFormer-1 and OccFormer, while label-free Mean-FLIP matches or outperforms supervised Taylor ranking on all five settings where that control was run. The method applies when removable primitives keep a stable checkpoint-row identity; it is budget-free in survivor count, not parameter-free.

1 INTRODUCTION

Real-time 3D occupancy prediction for autonomous driving must balance spatial coverage against latency and memory. Sparse Gaussian occupancy models replace dense 3D feature volumes with a bank of learned scene primitives. The attributes of each Gaussian depend on the input, but the number of generator slots is usually an architecture constant chosen by hand. Larger banks improve coverage at the cost of encoder, camera-sampling, and aggregation time. Activity criteria such as opacity or fire rate do not answer the question that matters for structural pruning: would removing this primitive change the final voxel decision? A highly opaque or frequently active row is not necessarily decisive. Overlapping primitives may cast redundant votes for the same class, or act on voxels whose winning class already leads by a wide margin. Conversely, a rarely active row can decide a small set of close-margin voxels. Most structured pruning methods also require a target row count or pruning ratio, and the right value varies substantially across architectures.

Decision-FLIP answers this question directly. Given a trained model and unlabeled calibration inputs, we remove each generated primitive from the occupancy readout alone and recompute the full class competition. The importance of a persistent bank row is the mean number of voxel decisions that change over the calibration frames. This score is output-aware, label-free, and tied to the prediction we want to preserve.

A ranking alone does not say how many rows to remove. Across architectures, log Mean-FLIP scores often show a concentrated low-impact group separated from a more influential tail, although we do not assume this is universal. We therefore treat budget selection as a two-population split and use

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.

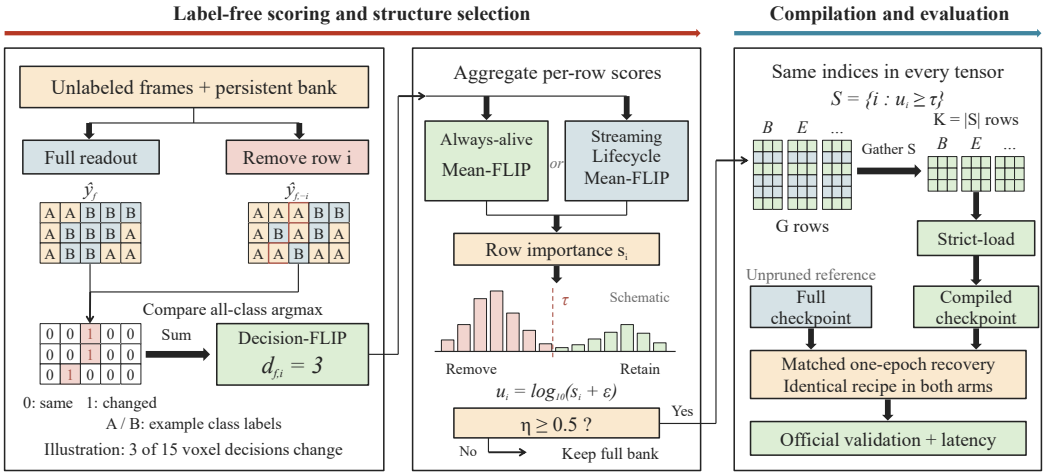


Figure 1: Overview of Decision-FLIP. (a) On unlabeled calibration frames, removing row i from the cached full readout and recomputing the all-class argmax gives per-frame Decision-FLIP; scores aggregate as Mean-FLIP or lifecycle Mean-FLIP. (b) Guarded log-Otsu selects a survivor set only when $\eta \geq 0.5$, otherwise retaining the full bank. (c) The same survivor indices are gathered across aligned tensors, strict-loaded, and compared with the unpruned reference under matched one-epoch recovery and official evaluation.

Otsu's criterion to propose a data-dependent boundary without a target sparsity. Because Otsu always returns a threshold, a fixed separation gate rejects weakly supported splits and keeps the full bank instead. Section 3 gives the criterion and the gate.

Our contribution is not Otsu thresholding on its own, but an end-to-end framework that connects decision-level importance to physical model compilation (Figure 1):

1. **Decision-Level Scoring and Budget-Free Selection.** *Decision-FLIP* measures primitive importance by exact voxel-decision changes, without labels, and pairs it with guarded Otsu selection to set the survivor count without a target sparsity.
2. **Lifecycle Integration and Physical Compilation.** We extend the criterion to streaming banks by following each persistent row through warp, death, and refill, and we compile the selected rows across all aligned tensors into strictly loadable checkpoints.
3. **Validation Across Models and Datasets.** Across four Gaussian-bank model-dataset settings and one persistent-query transfer, Decision-FLIP removes 12–79% of eligible rows and reduces bank-dependent latency by up to 45%, while the semantic difference after matched recovery stays within 0.18 point (Table 1).

2 Related Work

2.1 Gaussian occupancy prediction

GaussianFormer (Huang et al., 2025) represents scenes with sparse semantic Gaussians, while GaussianWorld (Zuo et al., 2025) extends Gaussian-space modeling to streaming occupancy. GaussianFormer3D (Zhao et al., 2025) initializes Gaussians from LiDAR voxel features, and GaussianFlowOcc (Boeder et al., 2025) predicts per-Gaussian temporal offsets. The released implementations of these models expose checkpoint-indexed primitive rows that meet the eligibility conditions in Section 3. OccFormer (Zhang et al., 2023) exposes a checkpoint-indexed learned-query bank in the same way. GaussTR (Jiang et al., 2025), in contrast, reuses 300 learned queries per view but resamples their 2D reference points at random on every forward pass, so a checkpoint row does not keep a stable primitive identity across calls.

2.2 Output-aware pruning for occupancy

Occupancy prediction combines a dominant empty class with dense all-class competition among overlapping primitives. A row can therefore be often active yet redundant for the decision, or rarely active but decisive. Whether local magnitude and activity proxies track the voxel decisions that compilation must preserve is thus an empirical question. Decision-FLIP measures exact voxel-decision changes under a readout intervention, and its offline compilation removes the selected rows from the checkpoint. This reduces row-dependent encoder, sampling, and readout computation, rather than filtering outputs only after inference. Efficient occupancy architectures attack related costs at the representation level. SparseOcc (Tang et al., 2024) replaces dense 3D latent processing with a sparse latent volume and sparse operators, while OPUS (Wang et al., 2024) formulates occupancy as set prediction with learned queries instead of classifying a dense voxel grid. These methods redesign and train the predictor around a sparse representation; Decision-FLIP instead compiles redundant rows from an already trained, checkpoint-indexed bank.

2.3 Output-aware compression and fair evaluation

Structured pruning commonly uses magnitude or sensitivity criteria (Han et al., 2015; Li et al., 2017; Molchanov et al., 2019), while output-aware methods propagate final-response importance or preserve downstream features (Yu et al., 2018; Luo et al., 2017; Zhuang et al., 2018). Exact Mean-FLIP directly measures nonlinear 0–1 voxel-decision changes instead of using a linearized proxy. Pruning structures have been selected using prescribed global thresholds (Liu et al., 2017), learned compression policies (He et al., 2018), or measured-latency constraints (Yang et al., 2018). Controlled studies at matched architectures or sparsities (Liu et al., 2019; Renda et al., 2020) show that conclusions can depend strongly on the retraining protocol; we therefore give full and compiled models the same recovery budget. We apply Otsu's criterion (Otsu, 1979) to log counterfactual scores behind a separation gate, producing a target-free candidate operating point without assuming universal bimodality.

2.4 Dynamic token reduction and offline compilation

DynamicViT (Rao et al., 2021) and Sparse DETR (Roh et al., 2022) select input-dependent token subsets, while ToMe (Bolya et al., 2023) merges input-dependent similar token pairs under a fixed per-layer merge count. These methods determine token reduction during inference and therefore retain runtime selection or similarity computation. Sparse DETR specifically sparsifies encoder-token updates according to expected decoder use; it does not prune a fixed persistent decoder-query bank. Decision-FLIP calibrates once and stores a physically reduced checkpoint, requiring no runtime selection policy. Per-scene Gaussian-compaction methods rank, mask, resample, or budget primitives optimized separately for each scene (LightGaussian (Fan et al., 2024), Compact3DGS (Lee et al., 2024), Mini-Splatting (Fang & Wang, 2024), and Taming 3DGS (Mallick et al., 2024)); our setting instead compiles checkpoint-indexed rows shared across inputs.

3 Method

3.1 Persistent primitive banks

Let $B = \{b_i\}_{i=1}^{G}$ be a learned bank. For frame f, row b_i generates a primitive $p_{f,i}$. The method requires:

- a stable mapping from checkpoint row i to generated primitive $p_{f,i}$;
- an occupancy readout that can be recomputed after removing one primitive; and
- aligned persistent tensors that can be gathered with the same survivor set.

We check these conditions in the implementation and, for streaming banks, in the explicit warp/death/refill lineage. Split-half stability tells us whether the calibration set is large enough; it does not establish row identity, and it cannot make an input-resampled row eligible. Appendix A.2.6 shows what happens on GaussTR: the separation gate passes, yet the ranking is no better than random.

GaussianWorld and GaussianFormer-1 satisfy these requirements for their full banks. GaussianFormer3D, GaussianFlowOcc, and OccFormer also expose eligible persistent rows under the model-specific protocols in Appendix A.

3.2 Exact all-class removal

Let $z_{f,c}(x)$ be the full class logit for voxel x, and let $\hat{y}_f(x) = \arg\max_c z_{f,c}(x)$. For an additive Gaussian readout,

$$z_{f,c}(x) = e_c(x) + \sum_{i=1}^{G} q_{f,i,c}(x),$$

where $q_{f,i,c}(x)$ is primitive i's local contribution and $e_c(x)$ collects readout terms independent of the removable bank rows, including a base or empty-class term when present. Removing primitive i is exact within the readout:

$$z_{f,-i,c}(x) = z_{f,c}(x) - q_{f,i,c}(x).$$

We recompute the argmax over **all classes**, including the empty class:

$$\hat{y}_{f,-i}(x) = \arg\max_c z_{f,-i,c}(x).$$

Additivity is required across cached row contributions, not within the upstream row generator: the query decoder and per-row heads may be nonlinear. For OccFormer, $q_{f,i,c}(x)$ is the post-softmax class probability multiplied by the post-sigmoid mask probability, and the final occupancy score sums these terms over queries. Removing one cached query contribution is therefore exact at the readout. Models whose readouts cannot be exactly decomposed or recomputed under row deletion are ineligible.

3.3 Exact Mean-FLIP

The per-row score is

$$s_i = \frac{1}{|\mathcal{C}|} \sum_{f \in \mathcal{C}} \sum_{x \in \mathcal{V}} \mathbf{1}[\hat{y}_f(x) \neq \hat{y}_{f,-i}(x)],$$

where $\mathcal{C}$ contains fixed training-calibration frames and $\mathcal{V}$ is the voxel grid. Scoring uses no ground-truth labels. Equivalently, $s_i/|\mathcal{V}|$ is the empirical expected Hamming distance between the full model's voxel decisions and the single-row-intervened decisions. Exact Mean-FLIP is therefore an output-aware importance criterion that counts actual decision changes rather than approximating them.

Mean-FLIP measures a single-row readout intervention. It provides a ranking, but does not imply additivity or guarantee that a set of individually low-scoring rows can be removed jointly without decision drift. We therefore evaluate the complete selected mask before recovery and report the resulting zero-epoch operating point. A held-out set-removal ablation additionally measures joint drift against same-K controls. Matched recovery adapts the physically reduced computation graph but is not treated as a guarantee of joint-removal safety.

To disambiguate, **Decision-FLIP** is the per-frame, per-row count of changed voxel decisions; **Mean-FLIP** is its calibration-set average; and **lifecycle Mean-FLIP** is the streaming aggregate that follows one persistent row's lineage through warp, death, and refill.

3.4 Lifecycle Mean-FLIP aggregation

For a non-recurrent bank, checkpoint row i exists on every frame and the score above already averages its whole trajectory. Streaming models need a stricter notion of identity. A Gaussian can be warped

into the next frame, leave the spatial range, die, or be replaced by a refill Gaussian. Treating the storage slot after a refill as the same primitive mixes different lifecycles and credits future evidence to the wrong checkpoint row.

We therefore follow each initial primitive through the model's exact warp validity mask. Let ℓ_i denote the initial Gaussian decoded from checkpoint row i, let $\mathcal{L}_i$ be the frames before that lineage leaves the valid range, and let d_{f,ℓ_i} be its exact all-class Decision-FLIP count on frame f. Its sequence-integrated importance is

$$s_i^{\text{life}} = \frac{1}{|\mathcal{C}|} \sum_{f \in \mathcal{L}_i} d_{f,\ell_i}.$$

The denominator is the whole calibration horizon, not only the frames in which the Gaussian exists, so the score measures contribution over the full horizon rather than importance while alive. A short-lived primitive must therefore be highly decisive to outrank a consistently useful one, which means rare but consequential events can be underweighted. Death ends a lineage. GaussianWorld refill rows come from the decoder's fixed spatial prior rather than from a learned checkpoint row, so they start new, unscored lineages and carry no evidence across that boundary. Compilation sets both the initial learned-bank size and the decoder's live-state budget to the selected K. For always-alive banks (GaussianFormer-1, GaussianFormer3D, GaussianFlowOcc, and OccFormer), $\mathcal{L}_i = \mathcal{C}$ and s_i^{life} equals Mean-FLIP.

3.5 Guarded Otsu budget selection

We transform the scores as

$$u_i = \log_{10}(s_i + \epsilon)$$

and choose the Otsu threshold τ that maximizes the between-group variance. The offset ϵ is fixed before thresholding to 10^{-3} times the first percentile of the positive scores (10^{-12} if every score is zero). Its only role is to place exact zeros on the log axis; it lies below every positive score and does not change their order. The normalized separation quality of the split is

$$\eta = \frac{\sigma^2_{\text{between}}}{\sigma^2_{\text{total}}}.$$

One configuration is fixed for the whole study, with no model- or dataset-specific tuning:

- minimum separation: $\eta_{\min} = 0.5$;
- no target K or fixed removal fraction; and
- abstain rather than invent a cut when the separation gate fails.

If the gate passes, we keep rows with $u_i \geq \tau$ and remove the rest; ties at the threshold are kept. If it fails, the selector abstains and returns the original bank ($K = G$). The survivor count is thus read off the score distribution rather than supplied. The gate decides only whether to accept or abstain; once it accepts, the Otsu threshold alone determines K. **Budget-free** therefore means that no survivor count or pruning ratio is given, not that the protocol has no constants: ϵ and $\eta_{\min}$ are fixed. The gate checks score separation, not statistical confidence; split-half stability and the full selected mask are evaluated separately.

3.6 Physical compilation and matched evaluation

The survivor indices are gathered across every aligned persistent tensor, and the reduced checkpoint must load under strict shape and key checking. Scoring and selection end here and use no labels. We report the compiled checkpoint before recovery, and also after an optional one-epoch recovery that lets row-dependent sampling, attention, and normalization adapt to the smaller tensors. For a fair comparison, the full and compiled models receive the same recovery recipe while the survivor

mask and compiled shape stay fixed. Recovery trains every parameter of the compiled checkpoint, including the surviving bank rows, unless an experiment states that a module is frozen, and it uses task labels unless marked self-distillation-only.

4 EXPERIMENTS

4.1 PROTOCOL

We evaluate five eligible model-dataset settings. Four use Gaussian banks: GaussianWorld, GaussianFormer-1, and GaussianFormer3D on nuScenes-SurroundOcc, and GaussianFlowOcc on Occ3D-nuScenes (Tian et al., 2023). The fifth, OccFormer on SemanticKITTI, tests transfer to a non-Gaussian persistent-query bank. We use each benchmark's official split and evaluator where possible (Wei et al., 2023; Caesar et al., 2020; Tian et al., 2023; Behley et al., 2019). GaussianFormer3D uses keyframe LiDAR only in every arm, and GaussianFlowOcc is our 3D-supervised reproduction; both are compared only within architecture. GaussTR is a negative control without stable row identity (Appendix A.2.6).

Scores are computed on unlabeled training-calibration inputs. Guarded Otsu with the fixed $\eta_{\min} = 0.5$ gate chooses the removal count, survivor indices are gathered across every aligned tensor, and the reduced checkpoints load under strict shape and key checking. Every primary comparison gives the full and compiled models the same one-epoch recovery budget; same-K ranking controls and GaussTR receive no recovery. The extra epoch alone raises GaussianFormer-1 and GaussianFormer3D by about 1.7 and 2.7 mIoU, so matched full controls are essential. Evaluations use each complete split, including GaussianWorld's 5,700-frame streaming protocol. Latency is measured in PyTorch fp32 at batch size 1 on one NVIDIA L4, with both configurations resident and interleaved after warm-up, except where stated; bank/query-dependent and whole-model changes are reported separately. Peak GPU memory is measured for the full and compiled shapes in separate fresh processes. Calibration sizes, evaluator details, matched-recovery results, checkpoint fidelity, and timing breakdowns are in Appendix A.

4.2 RESULTS ACROSS MODELS AND DATASETS

Table 1 combines the automatically selected structure, matched accuracy, paired latency, and peak-memory result for every eligible model-dataset setting, separating the Gaussian-bank study from the SemanticKITTI persistent-query transfer. The semantic difference after matched recovery is at most 0.171 point across the primary adaptive rows, while automatic removal ranges from 12% to 79% of the eligible bank (minimum 12.0% on GaussianFormer3D with nuScenes-SurroundOcc; maximum 79.0% on OccFormer). Peak-memory savings reach 16.6%, although fixed activations make the change negligible on several models and GaussianFormer-1 increases by 0.8%. For the five primary pairs with complete per-class outputs, no class that is nonzero in its matched full model newly collapses to zero after adaptive compilation.

For GaussianWorld, the lifecycle row is the primary result: it follows persistent-row identity through warp, death, and refill over the streaming sequence. The static-frame reference instead scores checkpoint rows independently per frame and ignores lineage events; it selects a more aggressive K and is reported only as a separate deployment trade-off whose values must not be combined with the lifecycle row. The lifecycle compilation costs 0.082 mIoU while cutting whole-model latency by 2.4%. Across settings with component profiles, the slot-dependent kernels speed up considerably, but the image backbone, voxel encoder or decoder, or LiDAR branch runs unchanged, and Amdahl's law caps the end-to-end gain. The 7.0–45.0% bank/query-dependent reductions verify that physical compilation removes the intended row-dependent work; they are mechanism measurements, not substitutes for the paired 2.4–8.0% primary whole-model results. Memory effects are likewise architecture dependent: OccFormer saves 16.6% peak allocated memory and GaussianFlowOcc saves 7.0%, whereas GaussianFormer-1 is unchanged within measurement variation (+0.8%). We therefore report component, whole-model, and memory results together rather than extrapolating unmeasured gains to other deployment hardware.

Figure 2 compares the primary lifecycle-compiled GaussianWorld model with its matched full control on two fixed validation samples. Scene geometry and semantic layout are closely preserved in both. Most differences lie on occupancy boundaries; semantic relabeling among voxels that stay occupied

Table 1: Automatic compilation across model-dataset settings. Changes are compiled minus full in the displayed metric order.

Model / metrics	**Kept / total (removed)**	**Before Δ**	**After recovery Δ**	**Bank/ query latency Δ**	**End-to-end latency Δ**	**Peak memory Δ**
GaussianWorld, primary lifecycle compilation (mIoU / geometry IoU)	22,427 / 25,600 (12.4%)	-0.179 / -0.290	-0.082 / -0.119	-7.0%	-2.4%	-1.2%
GaussianWorld, non-primary static-frame reference (mIoU / geometry IoU)	*16,461 / 25,600 (35.7%)*	*-0.412 / -0.574*	*-0.225 / -0.270*	*-24.3%*	*-9.3%*	*0.0%*
GaussianFormer-1 (mIoU / geometry IoU)	17,365 / 25,600 (32.2%)	-0.157 / -0.435	-0.171 / -0.246	-23.3%	-5.5%	+0.8%
GaussianFormer3D (mIoU / geometry IoU)	22,535 / 25,600 (12.0%)	-0.044 / -0.093	-0.024 / -0.116	-13.2%	-4.7%	-8.3%
GaussianFlowOcc (voxel mIoU / occupied IoU)	8,628 / 10,000 (13.72%)	+0.10 / +0.57	+0.01 / +0.03	-10.7%	-8.0%	-7.0%
OccFormer transfer, SemanticKITTI (SSC mIoU / SC IoU)	21 / 100 (79.0%)	-0.16 / 0.00	+0.03 / +0.08	-45.0%	-5.7%	-16.6%

Deltas are compiled minus full at the same stage, and italics mark the non-primary GaussianWorld static-frame reference. Peak memory is fresh-process PyTorch peak allocated memory after warm-up and reset for every row; reserved and NVML process measurements are diagnostics only (Appendix A.1).

accounts for less than 0.10% of the voxel grid in either sample. Qualitative samples for the remaining architectures, fixed before any output was inspected, are in Appendix A.3.

Figure 3 compares guarded Otsu with Kneedle (Satopaa et al., 2011), a two-component Gaussian mixture model (GMM) (McLachlan et al., 2019), and a largest-gap rule (cut at the widest gap between sorted log scores) under a common zero-recovery protocol; these are baselines, and guarded Otsu remains the fixed selector for all primary comparisons. Guarded Otsu selects non-trivial cuts of 12–79% on all five settings. GMM removes more rows but incurs larger losses on GaussianWorld, GaussianFormer-1, and GaussianFlowOcc; Kneedle collapses GaussianFormer-1 and GaussianFormer3D; and largest-gap often removes almost nothing. Some alternatives match or slightly exceed Otsu on individual architectures. This supports guarded Otsu as a non-degenerate, Pareto-competitive operating point, not as universally accuracy-optimal. On GaussianFlowOcc, guarded Otsu retains 8,628 rows, whereas Kneedle and largest-gap share the distinct 8,725-row mask. Appendix A.6.6 gives the complete operating points and the recovered GaussianFormer-1 analysis.

4.3 SAME-K RANKING-CRITERION ABLATION

To separate ranking quality from supervised re-adaptation, Table 2 changes only the ranking at each automatically selected survivor count, applies no recovery, and evaluates the complete validation split from the same source checkpoint. Mean-FLIP and the heuristic controls use unlabeled calibration frames; Taylor (Molchanov et al., 2019) uses training labels only to form its ranking. To fit every architecture in the main paper, entries are changes in the primary semantic metric relative to the same full source checkpoint (mIoU, or SSC mIoU for OccFormer); full metric pairs and absolute values are in Appendix A. Random removal collapses GaussianFormer-1 and OccFormer, and on OccFormer the fire-rate proxy erases three rare classes (bicycle, motorcycle, traffic-sign) that Mean-FLIP keeps. At the same K, label-free Mean-FLIP matches or outperforms supervised Taylor on all five settings,

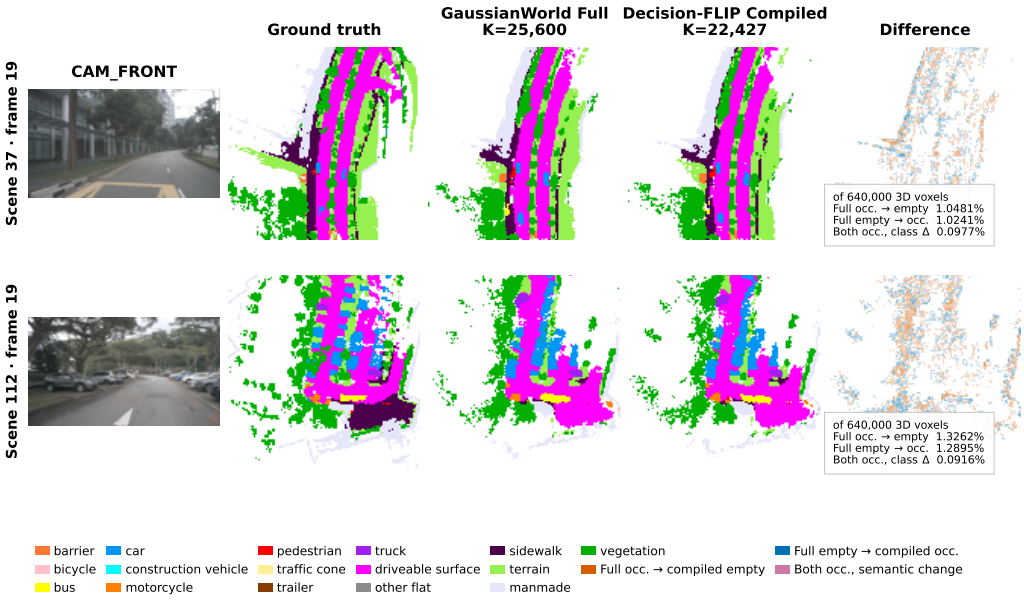


Figure 2: Qualitative preservation under lifecycle-aware GaussianWorld compilation. The two validation samples (scene indices 37 and 112, frame 19) were fixed before inspecting predictions. Each model processes the same 20-frame streaming prefix after matched recovery. Decision-FLIP retains 22,427 of 25,600 rows (12.4% removed). Difference panels separate full occupied to compiled empty, full empty to compiled occupied, and semantic changes where both predictions remain occupied; percentages use all 640,000 voxels in each $200\times200\times16$ grid.

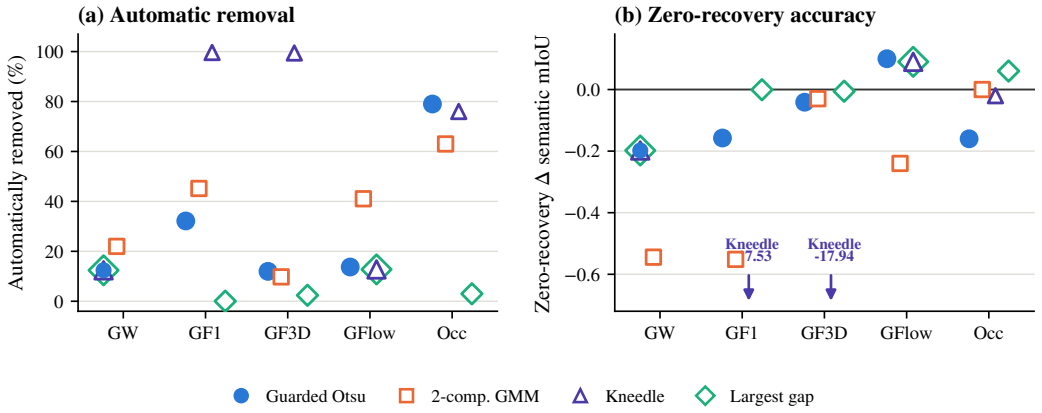


Figure 3: Automatic selector comparison. (a) Automatically removed fraction. (b) Zero-recovery semantic mIoU change from each model's full source. Concentric markers indicate identical masks; arrows annotate clipped Kneedle failures. Selection never uses validation metrics.

by a wide margin on GaussianFormer-1 and OccFormer. This supports the decision-level ranking signal; it is not a guarantee about joint removal. The selector comparison above addresses budget selection separately from these ranking controls.

4.4 Per-class preservation

Aggregate mIoU can hide failures on rare categories, so complete per-class outputs and a worst-change summary are in Appendix A.4. Among the five primary matched pairs with complete per-class

Table 2: Same-K ranking-criterion ablation before recovery (primary semantic metric Δ vs. full). We report one primary automatically selected operating point per architecture; the non-primary GaussianWorld static reference is excluded.

Model	K	Mean-FLIP	Taylor	Random	Activity	Opacity	Magnitude
GaussianWorld	22,427	**-0.179**	-0.194	-0.390	-0.252	-0.244	-0.326
GaussianFormer-1	17,365	**-0.157**	-1.221	-8.310	-8.652	-0.183	-0.474
GaussianFormer3D	22,535	**-0.044**	-0.075	-0.730	-0.100	-0.105	-0.071
GaussianFlowOcc	8,628	**+0.10**	**+0.10**	n/a	n/a	n/a	n/a
OccFormer (SemanticKITTI transfer)	21	**-0.16**	-7.98	-12.68	-0.33	n/a	-11.97

Proxy definitions. Taylor uses training labels and ranks rows by the calibration mean of $\sum_{p\in i}|w_p\,\partial\mathcal{L}/\partial w_p|$ over each aligned row i. Random uses seed 0; activity and opacity are calibration statistics; magnitude/query norm is each implementation's non-counterfactual parameter-magnitude proxy. All arms use zero recovery and the same K within each model. Bold marks the best result in each row; `n/a` marks a proxy that is undefined for the architecture or was not run at this K.

outputs, no class that is nonzero in its matched full model newly collapses to zero, and the largest class-level decrease is 0.59 points (OccFormer other-vehicle).

5 Discussion

Decision-FLIP applies when rows are identifiable, exactly removable at the all-class readout, and physically compilable across aligned tensors. OccFormer shows that these conditions extend beyond Gaussian banks to persistent learned queries, and GaussianWorld shows why streaming state needs lifecycle-aware scoring. The experiments support effective ranking and architecture-specific budget selection, not universal dominance or statistical equivalence; only GaussianFormer-1 was run with three recovery seeds, and the other settings use one. Same-K controls isolate ranking quality, and zero-recovery results show that the signal is present before any fine-tuning. Finally, fixed-cost image, voxel, and sensor modules can limit end-to-end acceleration even when bank-dependent computation falls substantially.

6 Limitations

Decision-FLIP measures changes relative to a trained model's readout, so it cannot tell correct predictions from teacher errors. Scoring reuses cached upstream outputs rather than rerunning row-dependent sampling, attention, and feature interaction for every deletion. Individually low scores therefore do not guarantee drift-free joint removal of correlated rows, and frame-averaged scores may underweight rare but consequential events. The method needs a stable mapping from checkpoint rows to generated primitives, which rules out architectures that resample or reorder query slots on every input. Streaming models must follow spatial warps and end lineages explicitly at death and refill boundaries. Optional recovery adds training cost and normally uses labels. Finally, whole-model speedups remain bounded by fixed-cost image, voxel, or sensor modules, so absolute and relative savings depend on the target hardware and kernels.

7 Conclusion

Decision-FLIP compiles persistent occupancy banks by ranking rows with unlabeled all-class readout interventions and selecting K through guarded Otsu. Across four Gaussian model-dataset settings and one persistent-query transfer, it removes 12–79% of eligible rows, reduces row-dependent latency by up to 45% and whole-model latency by up to 8.0% on the primary endpoints, and keeps the primary matched-recovery semantic changes within 0.18 point. Same-K controls confirm that decision-aware ranking matters. These results establish compilation without a target sparsity for identifiable persistent rows, with end-to-end gains bounded by fixed-cost modules.

ETHICS STATEMENT

This work studies compression of 3D occupancy prediction systems using public autonomous-driving benchmarks. It introduces no human-subject data. As with other perception compression methods, deployment requires validation under the target sensor, hardware, and safety conditions; the reported benchmark results do not by themselves establish safety for real-world driving.

REPRODUCIBILITY STATEMENT

Sections 3 and 4 specify the scoring, lifecycle, thresholding, compilation, matched-recovery, and evaluation protocols. The appendices report recovery controls and temporal diagnostics. Code, fixed checkpoint hashes, calibration splits, survivor counts, complete-validation metrics, and strict-loading checks will be released upon acceptance.

AI USE STATEMENT

Generative AI tools were used to assist with language editing and LaTeX formatting. The authors take full responsibility for the final content of this work, including all AI-assisted text.

REFERENCES

Luca Barsellotti, Lorenzo Bianchi, Nicola Messina, Fabio Carrara, Marcella Cornia, Lorenzo Baraldi, Fabrizio Falchi, and Rita Cucchiara. Talking to DINO: Bridging self-supervised vision backbones with language for open-vocabulary segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22025–22035, 2025. doi: 10.1109/ICCV51701.2025.02045. URL https://openaccess.thecvf.com/content/ICCV2025/html/Barsellotti_Talking_to_DINO_Bridging_Self-Supervised_Vision_Backbones_with_Language_for_ICCV_2025_paper.html.

Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jürgen Gall. SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9297–9307, 2019. doi: 10.1109/ICCV.2019.00939. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Behley_SemanticKITTI_A_Dataset_for_Semantic_Scene_Understanding_of_LiDAR_Sequences_ICCV_2019_paper.html.

Simon Boeder, Fabian Gigengack, and Benjamin Risse. GaussianFlowOcc: Sparse and weakly supervised occupancy estimation using Gaussian splatting and temporal flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24943–24954, 2025. doi: 10.1109/ICCV51701.2025.02313. URL https://openaccess.thecvf.com/content/ICCV2025/html/Boeder_GaussianFlowOcc_Sparse_and_Weakly_Supervised_Occupancy_Estimation_using_Gaussian_Splatting_ICCV_2025_paper.html.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JroZRaRw7Eu.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11618–11628, 2020. doi: 10.1109/CVPR42600.2020.01164.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1280–1289, 2022. doi: 10.1109/CVPR52688.2022.00135.

Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, Dejia Xu, and Zhangyang Wang. Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information*

Processing Systems, volume 37, pp. 140138–140158. Curran Associates, Inc., 2024. doi: 10.52202/079017-4447. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/fd881d3b625437354d4421818f81058f-Paper-Conference.pdf.

Guangchi Fang and Bing Wang. Mini-Splatting: Representing scenes with a constrained number of Gaussians. In *Computer Vision – ECCV 2024*, volume 15135 of *Lecture Notes in Computer Science*, pp. 165–181, 2024. doi: 10.1007/978-3-031-72980-5_10. URL https://doi.org/10.1007/978-3-031-72980-5_10.

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/ae0eb3eed39d2bcef4622b2499a05fe6-Paper.pdf.

Yihui He, Ji Lin, Zhijian Liu, Hanrui Wang, Li-Jia Li, and Song Han. Amc: Automl for model compression and acceleration on mobile devices. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (eds.), *Computer Vision – ECCV 2018*, pp. 815–832, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01234-2.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 376–393, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73383-3.

Haoyi Jiang, Liu Liu, Tianheng Cheng, Xinjie Wang, Tianwei Lin, Zhizhong Su, Wenyu Liu, and Xinggang Wang. GaussTR: Foundation model-aligned Gaussian transformer for self-supervised 3D spatial understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11960–11970, 2025. doi: 10.1109/CVPR52734.2025.01117. URL https://openaccess.thecvf.com/content/CVPR2025/html/Jiang_GaussTR_Foundation_Model-Aligned_Gaussian_Transformer_for_Self-Supervised_3D_Spatial_Understanding_CVPR_2025_paper.html.

Joo Chan Lee, Daniel Rho, Xiangyu Sun, Jong Hwan Ko, and Eunbyung Park. Compact 3d gaussian representation for radiance field. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21719–21728, 2024. doi: 10.1109/CVPR52733.2024.02052.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient ConvNets. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rJqFGTslg.

Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2755–2763, 2017. doi: 10.1109/ICCV.2017.298.

Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJlnB3C5Ym.

Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5068–5076, 2017. doi: 10.1109/ICCV.2017.541.

Saswat Subhajyoti Mallick, Rahul Goel, Bernhard Kerbl, Markus Steinberger, Francisco Vicente Carrasco, and Fernando De La Torre. Taming 3DGS: High-quality radiance fields with limited resources. In *SIGGRAPH Asia 2024 Conference Papers*, 2024. doi: 10.1145/3680528.3687694. URL https://dl.acm.org/doi/10.1145/3680528.3687694.

Geoffrey J. McLachlan, Sharon X. Lee, and Suren I. Rathnayake. Finite mixture models. *Annual Review of Statistics and Its Application*, 6(1):355–378, 2019.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11256–11264, 2019. doi: 10.1109/CVPR.2019.01152.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979. doi: 10.1109/TSMC.1979.4310076. URL https://ieeexplore.ieee.org/document/4310076.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. In *Advances in Neural Information Processing Systems*, volume 34, pp. 13937–13949, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/747d3443e319a22747fbb873e8b2f9f2-Abstract.html.

Alex Renda, Jonathan Frankle, and Michael Carbin. Comparing rewinding and fine-tuning in neural network pruning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1gSj0NKvB.

Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim. Sparse DETR: Efficient end-to-end object detection with learnable sparsity. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RRGVCN8kjim.

Ville Satopaa, Jeannie Albrecht, David Irwin, and Barath Raghavan. Finding a "Kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pp. 166–171, 2011. doi: 10.1109/ICDCSW.2011.20.

Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6105–6114, 2019. URL https://proceedings.mlr.press/v97/tan19a.html.

Pin Tang, Zhongdao Wang, Guoqing Wang, Jilai Zheng, Xiangxuan Ren, Bailan Feng, and Chao Ma. SparseOcc: Rethinking sparse latent representation for vision-based semantic occupancy prediction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15035–15044, 2024. doi: 10.1109/CVPR52733.2024.01424.

Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3D: A large-scale 3D occupancy prediction benchmark for autonomous driving. In *Advances in Neural Information Processing Systems*, volume 36, pp. 64318–64330, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/cabfaeecaae7d6540ee797a66f0130b0-Abstract-Datasets_and_Benchmarks.html. Datasets and Benchmarks Track.

Jiabao Wang, Zhaojiang Liu, Qiang Meng, Liujiang Yan, Ke Wang, Jie Yang, Wei Liu, Qibin Hou, and Ming-Ming Cheng. OPUS: Occupancy prediction using a sparse set. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 119861–119885. Curran Associates, Inc., 2024. doi: 10.52202/079017-3809. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/d8ca28a32c05cd3b9b0940e43720f31b-Paper-Conference.pdf.

Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21672–21683, 2023. doi: 10.1109/ICCV51070.2023.01986.

Tien-Ju Yang, Andrew Howard, Bo Chen, Xiao Zhang, Alec Go, Mark Sandler, Vivienne Sze, and Hartwig Adam. Netadapt: Platform-aware neural network adaptation for mobile applications. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (eds.), *Computer Vision*

– *ECCV 2018*, pp. 289–304, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01249-6.

Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3D: Towards zero-shot metric 3D prediction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9043–9053, 2023. doi: 10.1109/ICCV51070.2023.00830. URL https://openaccess.thecvf.com/content/ICCV2023/html/Yin_Metric3D_Towards_Zero-shot_Metric_3D_Prediction_from_A_Single_Image_ICCV_2023_paper.html.

Ruichi Yu, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I. Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S. Davis. Nisp: Pruning networks using neuron importance score propagation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9194–9203, 2018. doi: 10.1109/CVPR.2018.00958.

Yunpeng Zhang, Zheng Zhu, and Dalong Du. OccFormer: Dual-path transformer for vision-based 3D semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9433–9443, 2023. doi: 10.1109/ICCV51070.2023.00865. URL https://openaccess.thecvf.com/content/ICCV2023/html/Zhang_OccFormer_Dual-path_Transformer_for_Vision-based_3D_Semantic_Occupancy_Prediction_ICCV_2023_paper.html.

Lingjun Zhao, Sizhe Wei, James Hays, and Lu Gan. GaussianFormer3D: Multi-modal Gaussian-based semantic occupancy prediction with 3D deformable attention. *arXiv preprint arXiv:2505.10685*, 2025. URL https://arxiv.org/abs/2505.10685. Accepted to IEEE ICRA 2026.

Zhuangwei Zhuang, Mingkui Tan, Bohan Zhuang, Jing Liu, Yong Guo, Qingyao Wu, Junzhou Huang, and Jinhui Zhu. Discrimination-aware channel pruning for deep neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/55a7cf9c71f1c9c495413f934dd1a158-Paper.pdf.

Sicheng Zuo, Wenzhao Zheng, Yuanhui Huang, Jie Zhou, and Jiwen Lu. Gaussianworld: Gaussian world model for streaming 3d occupancy prediction. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6772–6781, 2025. doi: 10.1109/CVPR52734.2025.00635.

A Experimental protocols and supporting evidence

Throughout the appendices, GW, GF1, and GF3D abbreviate GaussianWorld, GaussianFormer-1, and GaussianFormer3D. Metric values are percentages unless a fraction is explicitly reported on the $[0, 1]$ scale; changes are in percentage points. *Decision drift* is the fraction of voxels occupied by the full model whose occupied/semantic decision changes after compilation. *Mass recall* is the fraction of full-model Mean-FLIP mass assigned to rows retained by a router. CVaR denotes the conditional mean in the upper tail of a row's framewise scores; the exploratory *composite* in Appendix C combines mean score, CVaR, and activation frequency. A *selective SSM* is a causal state-space router whose input-dependent transitions predict the rows retained at readout time.

A.1 Evaluation protocols and checkpoint fidelity

We use each repository's released evaluation code, with the protocol deviations listed below. Inspecting the checkpoints and implementations identifies the following eligible tensors:

- **GaussianWorld:** all 25,600 persistent rows are eligible.
- **GaussianFormer-1:** all 25,600 anchor/feature rows are eligible.
- **OccFormer (Zhang et al., 2023) (non-Gaussian transfer, SemanticKITTI (Behley et al., 2019)):** all 100 persistent Mask2Former (Cheng et al., 2022) queries are eligible; evaluated with its official semantic scene completion mIoU (SSC mIoU, 19 classes) and scene completion IoU (SC IoU) on the SemanticKITTI validation sequence 08 (815 frames, 256×256×32 voxels).

- **GaussianFormer3D (Zhao et al., 2025) (LiDAR+camera Gaussian bank, nuScenes-SurroundOcc):** all 25,600 persistent Gaussian anchors are eligible; evaluated with the SurroundOcc 17-class mIoU and geometric IoU on the 6,019 nuScenes validation keyframes. Protocol deviation: the released recipe aggregates 10 LiDAR sweeps and consumes a ground-truth depth map derived from them; the nuScenes sweep files are not available in our environment, so every arm (released, compiled, controls, recovered) uses keyframe LiDAR only and a depth map projected from that keyframe. Absolute numbers are therefore below the README value and are compared only across our own arms.

- **GaussianFlowOcc (camera-only Gaussian bank with temporal flow, Occ3D-nuScenes (Tian et al., 2023)):** all 10,000 persistent slots (initial means and query features) are eligible; evaluated with the repository's Occ3D mIoU (15 classes, excluding `others` and `other flat`; camera-visible mask) and occupied IoU on the 6,019 validation frames. Protocol deviation: no checkpoint is released, so we trained the repository's own 3D-supervised path and calibrated the full model's occupancy threshold ($\tau = 0.2$); Appendix A.2.5 gives the complete recipe. Absolute numbers are therefore not comparable to the GaussianFlowOcc paper and are compared only across our own arms.

Absolute accuracy and latency are compared only within an architecture. Unless a row explicitly states otherwise, reported latency uses PyTorch fp32, batch size 1, on one NVIDIA L4. TensorRT, mixed precision, and deployment-specific kernels may change both absolute latency and relative savings. Every Table 1 peak-memory value uses the same primary metric: fresh-process `torch.cuda.max_memory_allocated()` after model warm-up and a peak-statistics reset, with full and compiled shapes resident separately. Allocator-reserved and independently sampled NVML process peaks are retained only as diagnostics and are never mixed into the headline column.

Same-K proxy definitions. Each control keeps the adaptive arm's survivor count and changes only the ranking. Random uses seed 0. Fire/activity is the calibration-mean activity under the model's native gate (Gaussian opacity above the scorer's 0.1 fire threshold, or query-mask probability above 0.5). Opacity is median predicted opacity over calibration frames. Magnitude/contribution is the implementation's non-counterfactual readout-mass score; query norm is the joint ℓ_2 norm of the aligned learned query tables. Taylor ranks each row by the calibration mean of $\sum_{p \in i} |w_p \, \partial \mathcal{L} / \partial w_p|$ over its aligned parameters and uses training labels; the other criteria are label-free. All data-dependent controls use training calibration inputs, and every arm is compiled and evaluated with the same strict-load and official-validation protocol as Exact Mean-FLIP.

Every model except GaussianFlowOcc starts from a public checkpoint; GaussianFlowOcc uses the supervised reproduction described in Appendix A.2.5. The checkpoint-provenance table reports, per architecture and under one evaluator, the released checkpoint, the source checkpoint that the pipeline compiles, and the matched full control. The first two anchor our evaluators to published numbers and separate the effect of the recovery epoch from the effect of compilation; they are references, not controls, because they receive no recovery epoch.

Our evaluators track the released baselines, while the large recovery-only shifts on GF1 and GF3D show why every pruning result must be compared with an identically trained full control.

The released GaussianWorld checkpoint scores 21.759 under the streaming protocol (38-frame windows over 150 validation scenes, 5,700 frames) against a README value of 22.13, so the evaluator is faithful. The released GaussianFormer-1 weights score 1.74 points below their README value in the same evaluator and recover to 19.291 after one matched unpruned epoch on our data pipeline; we attribute the gap to pipeline adaptation and report it rather than correct for it. Every compiled result is therefore compared with a control that receives the identical recovery epoch: for GaussianFormer-1 the epoch alone moves semantic mIoU by +1.72 points, whereas compilation costs 0.17 point. Peak memory is measured in a fresh process. Every adaptive result records the Otsu threshold, separation η, survivor indices, strict-load certificate, and official validation metrics. Appendix A.2.1 distinguishes the lifecycle and static GaussianWorld protocols.

Table A1: Checkpoint and evaluator provenance

Model	Released checkpoint (README mIoU)	Released, our evaluator, 0 ep (mIoU / geo IoU)	Source checkpoint compiled by the pipeline, 0 ep	Matched full, 1 recovery epoch
GaussianWorld	released streaming checkpoint (22.13)	21.759 / 32.804	Opacity-calibrated source checkpoint: 21.902 / 33.140	21.874±0.025 / 33.097±0.053
GaussianFormer-1	`NonEmpty` 25,600 solid (19.31)	17.571 / 28.779	Same as released	19.291±0.002 / 30.050±0.002
OccFormer (SemanticKITTI)	`occformer_kitti.pth` (13.50 SSC mIoU)	13.50 / 36.42 (SSC mIoU / SC IoU)	Same as released	12.99 / 36.22 (paired evaluation; one epoch, lr 5e-5, official 8-GPU batch)
GaussianFormer3D (nuScenes-SurroundOcc, keyframe LiDAR only)	`surroundocc_release.pth`	18.40 / 30.06 (keyframe-only inputs; README value uses 10 LiDAR sweeps)	Same as released	21.152 / 35.496 (one epoch, lr 4e-5, image backbone frozen; Appendix A.2.4 explains the +2.75 mIoU adaptation gain)
GaussianFlowOcc (Occ3D-nuScenes (Tian et al., 2023), our 3D-supervised reproduction, $\tau = 0.2$)	none released; ours trained for 18 epochs (weak-recipe paper numbers not comparable)	24.65 / 54.86 (mIoU / occupied IoU; 20.79 / 38.54 at the weak-recipe default $\tau = 0.05$)	Same as trained	24.70 / 54.75

Table A2: GaussianWorld static-frame deployment reference

Stage / metric	**Matched full** $K = 25{,}600$	**Adaptive** $K = 16{,}461$
Otsu separation η	N/A	0.6361
Automatically removed	0	9,139 (35.699%)
Image backbone	514.38 ms	513.65 ms (-0.1%)
Neck	7.81 ms	7.83 ms (+0.2%)
Encoder	170.35 ms	120.74 ms (-29.1%)
– camera sampling (DFA)	111.67 ms	74.91 ms (-32.9%)
Head incl. aggregation	146.94 ms	119.40 ms (-18.7%)
– local aggregation	68.45 ms	42.86 ms (-37.4%)
Gaussian-side subtotal	**317.29 ms**	**240.14 ms (-24.3%)**
Whole model	**839.80 ms**	**761.92 ms (-9.3%)**
Peak GPU memory	5.792 GiB	5.791 GiB
Semantic mIoU / Geometry IoU, compiled, 0 recovery epochs	21.902 / 33.140 (calibrated source)	21.490 / 32.566 (-0.412 / -0.574)
Semantic mIoU	21.874±0.025	**21.649**
Geometry IoU	33.097±0.053	**32.827**

A.2 Architecture-specific protocols, results, and interpretation

A.2.1 GaussianWorld

Protocol. The static reference scores isolated frames and is reported separately from the primary streaming lifecycle protocol.

Quantitative result. Static Mean-FLIP removes 35.7% of the Gaussians and cuts Gaussian-side latency by 24.3% and end-to-end latency by 9.3%, at a cost of 0.225 mIoU and 0.270 geometry IoU against the recovered full control. Before recovery the gap is 0.41 / 0.57, so the matched epoch closes 0.16 / 0.26 of it. The adaptive latency is the mean of two 100-round profiles (756.61 and 767.22 ms, a 1.39% difference). This is a deployment reference, not the primary lifecycle-aware result.

Interpretation. The lifecycle run follows each initial learned row through streaming state updates and refill boundaries, applies the same guarded Otsu rule without a target survivor count, and locks both `lifter.num_anchor` and `future_decoder.num_anchor` to 22,427. Its automatic cut is substantially more conservative than the static-frame cut above and reduces the post-recovery gap to

Table A3: GaussianWorld lifecycle-integrated compilation

Stage / metric	**Matched full** $K = 25{,}600$	**Lifecycle-adaptive** $K = 22{,}427$
Otsu separation η	N/A	**0.8406**
Automatically removed	0	**3,173 (12.395%)**
Image backbone	529.52 ms	529.06 ms (-0.1%)
Neck	7.86 ms	7.87 ms (+0.2%)
Encoder	171.29 ms	154.98 ms (-9.5%)
– camera sampling (DFA)	112.90 ms	100.58 ms (-10.9%)
Head incl. aggregation	99.61 ms	96.96 ms (-2.7%)
– local aggregation	89.64 ms	87.80 ms (-2.1%)
Gaussian-side subtotal	**270.90 ms**	**251.94 ms (-7.0%)**
Whole model	**808.54 ms**	**789.13 ms (-2.4%)**
Peak GPU memory	5.854 GiB	5.782 GiB
Semantic mIoU / Geometry IoU, compiled, 0 recovery epochs	21.902 / 33.140 (calibrated source)	**21.723 / 32.850 (-0.179 / -0.290)**
Semantic mIoU, matched one epoch	21.874±0.025	**21.792 (-0.082)**
Geometry IoU, matched one epoch	33.097±0.053	**32.978 (-0.119)**
Recovery gain of compiled model	—	**+0.069 / +0.128**
Evaluation coverage	5,700 streaming frames	5,700 streaming frames

Table A4: GaussianFormer-1 compilation, efficiency, and matched recovery

Stage / metric	**Matched full** $K = 25{,}600$	**Adaptive lifecycle Mean-FLIP + Otsu**
Otsu separation η	N/A	**0.6593**
Automatically selected K	25,600	**17,365**
Automatically removed	0	**8,235 (32.168%)**
Image backbone	535.66 ms	533.67 ms (-0.4%)
Neck	7.75 ms	7.75 ms (+0.0%)
Encoder	123.53 ms	89.29 ms (-27.7%)
– camera sampling (DFA)	86.52 ms	61.06 ms (-29.4%)
Head incl. aggregation	31.45 ms	29.54 ms (-6.1%)
– local aggregation	21.31 ms	19.91 ms (-6.6%)
Gaussian-side subtotal	**154.98 ms**	**118.83 ms (-23.3%)**
Whole model	**698.61 ms**	**660.44 ms (-5.5%)**
Peak GPU memory	2.031 GiB	2.048 GiB
A100 whole-model diagnostic	230.1 ms	216.6 ms (-5.9%)
A100 peak GPU memory diagnostic	2.03 GiB	2.05 GiB
Semantic mIoU / Geometry IoU, compiled, 0 recovery epochs	17.571 / 28.779 (released weights)	17.414 / 28.344 (-0.157 / -0.435)
Semantic mIoU, three-seed matched recovery	19.291±0.002	**19.120±0.003 (-0.171±0.002)**
Geometry IoU, three-seed matched recovery	30.050±0.002	**29.804±0.001 (-0.246±0.001)**

0.082 mIoU / 0.119 geometry IoU. On the local NVIDIA L4-24Q, two independent 100-round round-robin profiles give whole-model means of 809.10 / 807.98 ms for the full model and 788.83 / 789.42 ms for the lifecycle-compiled model. The cross-run differences are 0.14% and 0.07%, respectively, passing the prespecified 1% guard; each run also passes the 2% mean-versus-median guard. These latencies are paired within the streaming configuration of Table A3; the static reference's absolute latencies belong to its separate configuration and $K = 16{,}461$ budget and are not comparable.

A.2.2 GAUSSIANFORMER-1

Protocol. The scorer uses 500 training-calibration frames; compilation is strict-loaded, and each fixed-checkpoint evaluation covers all 753 distributed validation batches. The frozen $K = 17{,}365$ mask is shared across recovery seeds.

Quantitative result. On GaussianFormer-1, removing 32.2% of the bank cuts Gaussian-side latency by 23.3% and whole-model latency by 5.5%. Across matched recovery seeds 0, 1, and 2, the difference is -0.171 ± 0.002 mIoU and -0.246 ± 0.001 geometry IoU; no checkpoint is selected by validation performance.

Because every GF1 row is alive on every scored frame, the lifecycle integral of Section 3.4 equals the saved Mean-FLIP score up to float32 accumulation error (maximum absolute difference 1.94×10^{-6}); its Otsu mask and measured recovery results are therefore unchanged. Every seed uses the same frozen mask and covers all 753 validation batches. No validation checkpoint was selected.

Interpretation. The same run also contains a fixed-$K = 17{,}296$ exact arm and matched opacity and random controls at that budget; before recovery they score 17.406 / 28.318 (exact, fixed K), 17.384 / 27.545 (opacity ranking) and 9.142 / 11.648 (random ranking), and after the matched epoch 19.108 /

29.698, 19.105 / 29.698 and 18.504 / 28.711. Compilation alone therefore costs the exact ranking 0.16 mIoU / 0.44 IoU at a 32% cut, whereas a random cut of the same size collapses the model before recovery and remains 0.6 mIoU below the exact ranking after it.

The L4 profile uses one NVIDIA L4-24Q at batch size 1 with both configurations resident and interleaved round-robin: 10 warm-up and 100 measured rounds in each of two runs. Whole-model means were 697.31 / 699.90 ms (full) and 658.63 / 662.26 ms (compiled), with inter-run variation of 0.37% and 0.55% and mean-versus-median gaps no larger than 0.70%. The full arm loads the released checkpoint; the compiled arm strictly loads the saved $K = 17{,}365$ structure. Peak memory is measured in a fresh process. The A100 measurements are diagnostic and are not combined with the L4 protocol.

A.2.3 OccFormer

Protocol. The 100 persistent Mask2Former queries are scored on 200 training frames (20 per sequence, seed 0) at the full $256 \times 256 \times 32$ decision volume; labels are not used. Compiled checkpoints are strict-loaded and evaluated on all 815 frames of SemanticKITTI sequence 08.

Quantitative result. OccFormer's persistent queries are rankable: a conservative 40% cut matches or exceeds the full control, while the automatic 79% cut reduces query-dependent work by 45% and whole-model latency by 5.7% with a small, recovery-sensitive mIoU change.

Interpretation. Inspection of the released OccFormer (Zhang et al., 2023) SemanticKITTI implementation shows a single-image EfficientNet-B7 (Tan & Le, 2019) backbone, a lift-splat-shoot voxel lift, a 3D occupancy encoder (3D convolutions followed by a 3D deformable-attention neck), and a Mask2Former head with 100 persistent learnable queries (`query_embed` and `query_feat`, the only per-query tables in the checkpoint). Its inference readout is additive over queries: $\text{score}[c, x] = \sum_q \text{softmax}(\text{cls}_q)[c]\, \sigma(\text{mask}_q[x])$ with the label $\arg\max_c$ over 20 classes (class 0 is empty), so deleting a query is an exact subtraction from the cached scores. Sixty-nine of the 100 queries change at most one voxel decision per frame and carry a no-object probability above 0.99, three change none, and one query alone flips 83% of the voxels because it carries the empty class.

The Otsu valley on log10 scores falls at $K = 21$ (split-half thresholds 0.56 / 0.71 in log10 units), so the adaptive rule removes 79 of the 100 queries without a target count; the fixed $K = 60$ column removes only the 40 lowest-ranked queries as a reference. At $K = 60$ the compiled model is not worse than the released checkpoint before any recovery (13.60 versus 13.50 SSC mIoU, SC IoU unchanged), whereas the same-K random control collapses to 4.39, so the ranking carries signal. At the adaptive budget the compiled model loses 0.16 point with SC IoU unchanged; the fire-rate control at the same $K = 21$ loses 0.33 point and erases three rare classes (bicycle, motorcycle, traffic-sign) that the exact flip score keeps, and the random control collapses to 0.82.

The paired evaluation gives every arm the same one-epoch recovery under the official training batch (8 GPUs $\times$ 1 sample, synchronized BatchNorm), half base learning rate (5e-5), and 100-iteration warm-up. It evaluates all 815 frames: the matched full reaches 12.99 / 36.22 and adaptive $K = 21$ reaches 13.02 / 36.30, a compiled-minus-full change of +0.03 / +0.08. Before recovery the compiled adaptive model is within 0.16 point of the released checkpoint, so for OccFormer compilation itself needs no recovery. Our evaluator reproduces the README value of the released checkpoint exactly (13.50 SSC mIoU; the released training log peaks at 13.50 in epoch 21).

L4 latency. One NVIDIA L4-24Q, PyTorch 1.10.1 + CUDA 11.3, fp32, batch size 1, one 384×1280 left image (validation frame 08/000500), the three query budgets resident and timed round-robin (10 warm-up + 100 measured rounds, mean of two runs; whole-model runs 639.26 / 639.48 ms, 621.36 / 621.58 ms and 603.11 / 603.29 ms, inter-run deviation below 0.05%; the head breakdown rows come from a third identical run whose head totals agree within 0.7 ms). Compiled query banks are the strict-loaded sliced checkpoints; latency does not depend on the weight values. The query-dependent work is the head forward (nine decoder layers and the per-query mask logits) plus the readout that OccFormer runs outside the head module: every per-query mask is upsampled to the 256×256×32 decision grid, weighted by its class probabilities, summed and arg-maxed. The hooks do not cover that readout, so we report it as the residual between the whole-model time and the sum of hooked modules; it shrinks from 26.7 to 14.6 ms as the query count falls, consistent with that attribution. Head and readout together hold 12.6% of the per-frame time and fall by 23% at $K = 60$ and 45% at $K = 21$; Amdahl's law then predicts a 5.7% whole-model saving at $K = 21$, matching the measured

Table A5: OccFormer query compilation, efficiency, and matched recovery

Stage / metric	**Matched full** $K = 100$	**Fixed-K reference,** $K = 60$ **(same flip ranking)**	**Adaptive Exact Mean-FLIP + Otsu**
Eligible population	100 persistent Mask2Former queries	Same population	Same population
Otsu separation η	N/A	N/A (fixed budget)	**0.795** (69 queries change $\leq$1 voxel per frame; 3 exactly zero)
Automatically selected K	100	60 (fixed)	**21**
Automatically removed	0	40 (40.0%, fixed)	**79 (79.0%)**
Image backbone (EfficientNet-B7)	61.43 ms	61.56 ms (+0.2%)	61.44 ms (+0.0%)
Image neck (SECOND FPN)	0.45 ms	0.45 ms	0.45 ms
LSS voxel lift	18.45 ms	18.44 ms (-0.1%)	18.40 ms (-0.3%)
3D occupancy encoder, conv backbone	260.69 ms	261.18 ms (+0.2%)	260.68 ms (-0.0%)
3D occupancy encoder, deformable-attention neck	217.72 ms	217.54 ms (-0.1%)	217.86 ms (+0.1%)
Mask2Former head (queries + mask readout)	53.93 ms	41.58 ms (-22.9%)	29.75 ms (-44.8%)
– nine query decoder layers	22.87 ms	18.13 ms (-20.7%)	13.60 ms (-40.5%)
– per-query mask logits inside the head forward (mask embedding $\times$ mask feature)	30.62 ms	22.96 ms (-25.0%)	15.85 ms (-48.2%)
Readout outside the head module: trilinear upsampling of every per-query mask to 256$\times$256$\times$32, class-weighted sum, argmax (residual = whole minus hooked modules)	26.70 ms	20.72 ms (-22.4%)	14.62 ms (-45.2%)
Query-dependent subtotal (head + readout)	**80.63 ms**	**62.30 ms (-22.7%)**	**44.37 ms (-45.0%)**
Whole model	**639.37 ms**	**621.47 ms (-2.8%)**	**603.20 ms (-5.7%)**
Peak GPU memory	3.704 GiB	3.089 GiB (-16.6%)	3.089 GiB (-16.6%)
SSC mIoU / SC IoU, released checkpoint, 0 recovery epochs	13.50 / 36.42	**13.60 / 36.42 (+0.10)** (compiled, before recovery)	**13.34 / 36.42 (-0.16)** (compiled, before recovery)
Same-K random control, 0 recovery epochs	N/A	4.39 / 14.60	0.82 / 7.89
Same-K fire-rate control, 0 recovery epochs	N/A	13.62 / 36.42	13.17 / 36.41 (bicycle, motorcycle, traffic-sign fall to 0.00)
SSC mIoU / SC IoU, paired matched epoch (lr 5e-5, 100-iteration warm-up, official 8-GPU SyncBN batch)	12.99 / 36.22	N/A	**13.02 / 36.30 (+0.03 / +0.08 vs full)**

5.66%. The 3D occupancy encoder (conv backbone 40.8%, deformable-attention neck 34.1%) and the image backbone (9.6%) do not depend on the query count, so the whole-model saving is 2.8% and 5.7%, respectively. Peak memory drops from 3.704 GiB at $K = 100$ to 3.089 GiB at $K = 60$, then remains at 3.089 GiB at $K = 21$. Reducing query-mask storage causes the initial decrease, after which fixed activations and CUDA allocator granularity determine the measured peak. The transfer claim is an accuracy and selection claim: the exact flip score identifies the 69 near-idle queries (three exactly zero) without a target count, and at a 79% cut it preserves rare classes that the fire-rate proxy discards.

Table A6: GaussianFormer3D compilation under the keyframe-LiDAR protocol

Stage / metric	**Matched full** $K = 25{,}600$	**Adaptive Exact Mean-FLIP + guarded Otsu** $K = 22{,}535$
Otsu separation η	N/A	**0.569** (gate 0.5 passed; log10 threshold -1.060, equivalent to 0.0872 flips/frame)
Automatically removed	0	**3,065 (12.0%)**
Gaussian encoder	382.8 ms	**332.2 ms (-13.2%)**
– DeformableFeatureAggregation3D	328.3 ms	**283.6 ms (-13.6%)**
Head incl. local aggregation readout	151.2 ms	150.0 ms (-0.8%)
– local aggregation	73.4 ms	70.7 ms (-3.7%)
Slot-independent modules	707–710 ms	Unchanged within noise
Whole model, L4, batch 1	**1230.0 ms**	**1172.6 ms (-4.7%)**
Whole-model median	1160.4 ms	1107.3 ms (-4.6%)
Peak GPU memory	3.810 GiB	**3.492 GiB (-8.3%)**
Semantic mIoU / Geometry IoU, 0 recovery epochs	18.40 / 30.06	**18.36 / 29.96 (-0.04 / -0.10)**
Semantic mIoU / Geometry IoU, matched one epoch	21.152 / 35.496	**21.128 / 35.380 (-0.024 / -0.116)**

A.2.4 GAUSSIANFORMER3D

Protocol. Every arm uses keyframe LiDAR only, as specified in Appendix A.1. Scores are accumulated on 500 training keyframes at stride 12, and all reported comparisons use the complete 6,019-frame validation set.

Quantitative result. On GaussianFormer3D, the automatically selected 12% cut has a single-seed difference of -0.024 mIoU / -0.116 geometry IoU while reducing whole-model latency by 4.7% and peak memory by 8.3%.

Interpretation. The same-K controls for GaussianFormer3D (exact, fire-rate, random at $K = 22{,}535$) are reported in Table A16. Every compiled arm there retains the same slot count, so its latency and memory structure is identical and the comparison isolates ranking quality.

GaussianFormer3D keeps the GaussianFormer-1 anchor bank (28 anchor parameters and a 128-d instance feature per slot) and readout (linear local aggregation), and adds a LiDAR branch (voxelization, sparse convolution) and a depth head whose ground-truth depth input is projected from LiDAR. The bank therefore satisfies both structural conditions, and the exact removal identity is the one verified for GaussianFormer-1. The LiDAR and image branches do not depend on the slot count, so the whole-model saving is again bounded by the encoder and readout share of latency. Flip scores are accumulated on 500 training keyframes (stride 12). The analytic reconstruction mirrors the aggregation kernel's per-Gaussian box (radius $\lceil 3\sigma_{\max}/0.5\,\mathrm{m} \rceil$ voxels, up to 11 for this model's 1.8 m scale cap, versus 4 for GaussianFormer-1) and is checked against brute-force delete-and-recompute with the official CUDA aggregator on the highest-contribution slots: maximum absolute logit difference 1.2×10^{-5}, zero voxels outside tolerance, argmax identical. The exact flip score correlates with the ground-truth hit rate (Spearman 0.86) and less with fire rate (0.77) and median opacity (0.72), the same pattern as GaussianFormer-1. The valley removes only 12% of the bank: with a LiDAR branch feeding the anchors, far fewer slots are decision-idle than in the camera-only GaussianFormer banks. At that budget the compiled model loses 0.04 mIoU / 0.10 IoU, the same-K fire-rate control 0.10 / 0.20, and the same-K random control 0.73 / 1.62. All arms share the keyframe-only inputs described in Appendix A.1.

Latency follows the same Amdahl decomposition as the other models: removing 12% of the slots cuts the slot-proportional deformable aggregation by 13.6% and the Gaussian encoder by 13.2%, while the image backbone (54% of the 1.23 s whole-model time on the L4 at 6×928×1600), the LiDAR voxelization and sparse convolution, and the depth head do not move, so the whole model gains 4.7% (mean) / 4.6% (median) and peak memory falls 8.3%. Absolute per-module times on the shared L4 workstation vary by up to 20% between the two runs; the relative savings agree within 0.3 points across runs, which is why they are reported as ratios of interleaved round-robin means.

The strict matched full control trained for one epoch under the recovery recipe reaches 21.152 / 35.496, far above the released 18.40 / 30.06: the released weights were trained with ten aggregated LiDAR sweeps, and a single epoch on keyframe-only inputs adapts the LiDAR branch and depth head to the sparser point cloud. This makes the matched control indispensable for GaussianFormer3D — the compiled bank after its own recovery epoch must be read against 21.152, not against the released number. In the strict full-versus-compiled run, the compiled bank reaches 21.128 / 35.380, a 0.024 mIoU / 0.116 IoU reduction.

Table A7: GaussianFlowOcc compilation of the supervised reproduction

Stage / metric	**Full** 10,000	**Adaptive** K
Eligible population	10,000 persistent slots (initial mean + query feature)	Same population
Otsu separation η	N/A	**0.856**
Automatically selected K	10,000	**8,628**
Automatically removed	0	**1,372 (13.72%)**
Primary $K = 8{,}628$ L4 whole-model / slot-dependent latency	318.9 / 231.1 ms	**293.5 / 206.5 ms (-8.0% / -10.7%)**
Primary $K = 8{,}628$ peak GPU memory (torch allocated)	1.486 GiB	**1.382 GiB (-7.0%)**
Semantic mIoU, 0 recovery epochs (τ = 0.2)	24.65	**24.75 (+0.10)**
Occupied IoU, 0 recovery epochs (τ = 0.2)	54.86	**55.43 (+0.57)**
Canonical same-K heuristic proxy controls and decision drift	—	unavailable (Taylor: Table A16)
Same quantities at the weak-recipe default τ = 0.05 (mis-calibrated; $K = 8{,}861$ there)	20.79 / 38.54	20.91 / 38.86 (+0.12 / +0.32), drift 3.1%; random 20.90 / 39.88 (drift 4.5%); fire-rate 21.48 / 46.50 (drift 13.9%)
Semantic mIoU / occupied IoU, after matched recovery	24.70 / 54.75	**24.71 / 54.78 (+0.01 / +0.03)**

A.2.5 GAUSSIANFLOWOCC

Protocol. GaussianFlowOcc is the sole exception to the public-checkpoint starting condition: no checkpoint is released. We therefore use the repository's 3D-supervised training path and compare only matched arms from that reproduction. The sequential evaluation advances temporal state through 8,971 consecutive forwards while scoring 300 frames. Guarded Otsu selects the primary $K = 8{,}628$ at $\eta = 0.856$; Kneedle and largest-gap instead share the distinct $K = 8{,}725$ mask.

Quantitative result. On our supervised GaussianFlowOcc at its calibrated threshold, sequential guarded Otsu removes 13.72% and changes mIoU / occupied IoU by +0.10 / +0.57 before recovery. Matched recovery reaches 24.71 / 54.78 versus the matched full 24.70 / 54.75 (+0.01 / +0.03). On the L4 protocol, the primary compiled model reduces the slot-dependent subtotal by 10.7%, whole-model latency by 8.0%, and torch peak allocated memory by 7.0%. Primary $K = 8{,}628$ heuristic same-K proxy results remain unavailable; complete matched-recovery per-class results are in Table A11.

Interpretation. GaussianFlowOcc keeps a bank of 10,000 slots, each an initial mean and a query feature that the decoder refines every frame, and reads the refined Gaussians out by splatting opacity-weighted contributions into a 200×200×16 grid. The density is the clamped sum of contributions, the semantic logit of a voxel is the opacity-weighted mean of the slots' logits, and a voxel is occupied when its density exceeds a threshold. The mean's denominator is a positive scalar per voxel, so the class argmax equals the argmax of the numerator, which is linear in each slot's term. Removing a slot is therefore exact: subtract its density and its weighted logits before the clamp, then apply the two post-processing steps (density below 0.001 set to zero, roof rows removed) as in the model's own inference. We verified this on the trained checkpoint against the model's own readout with the slot physically removed: 80 of 80 probed slots on two frames give identical decisions, the scorer's readout matches the model's on every voxel, and the density error is 3×10^{-7}. The temporal module carries the previous frame's Gaussians forward, so flip scores are accumulated with the model run sequentially while scoring one frame in 30 to keep temporal state identical to the evaluation path. The evaluation contains 8,971 sequential forwards and 300 scored frames.

No checkpoint is released for this model and the weakly-supervised recipe starts from about 826 GB of generated depth and open-vocabulary pseudo labels, so we trained the repository's own 3D-supervised path (binary cross-entropy on the voxel density and cross-entropy on the voxel semantics against Occ3D labels, temporal supervision on the flowed frames, 18 epochs, 8 GPUs, batch 16, learning rate 10^{-4}). The repository's evaluation threshold of 0.05 belongs to the weakly-supervised model; for the supervised density it marks 57% of the grid occupied. We therefore calibrated the

Table A8: GaussTR boundary case without stable row identity

Stage / metric	Released full $K = 300$ per view	Exact Mean-FLIP + Otsu (compiled, no recovery)
Eligible population	300 query embeddings reused over six views (1,800 generated Gaussians; reference points resampled every forward)	Same query rows
Otsu separation η	N/A	0.64 (gate passes)
Automatically selected K	300	158
Automatically removed	0	142 (47.3%)
Image backbone (GaussTR (Jiang et al., 2025): frozen DINOv2 (Oquab et al., 2024) ViT-B/14)	530.36 ms	530.00 ms (-0.1%)
Neck	69.72 ms	69.77 ms (+0.1%)
Query decoder	18.54 ms	16.41 ms (-11.5%)
Gaussian head incl. voxelizer	418.35 ms	233.55 ms (-44.2%)
– voxelizer (Gaussian-to-voxel)	408.10 ms	223.61 ms (-45.2%)
Query-dependent subtotal (decoder + head)	**436.89 ms**	**249.96 ms (-42.8%)**
Whole model	**1049.18 ms**	**862.06 ms (-17.8%)**
Peak GPU memory	2.272 GiB	1.867 GiB
Semantic mIoU / IoU, compiled only	12.28 / 44.66 (README 12.27)	11.47 / 40.32 (-0.81 / -4.34 points)
Random ranking, same $K = 158$	—	11.47 / 40.31
Fire-rate ranking, same $K = 158$	—	11.46 / 40.31

threshold of the full model on 25 whole training scenes disjoint from the calibration frames: mIoU 23.2 at 0.05, 26.2 at 0.1, 27.7 at 0.15, 28.2 at 0.2 (maximum), 28.1 at 0.25 and 27.5 at 0.3, with occupied IoU peaking at 0.3; the threshold that maximizes mIoU, $\tau = 0.2$, is used for every arm. The full model then scores 24.65 mIoU / 54.86 IoU on the validation frames instead of 20.79 / 38.54. The mis-calibrated row of Table A7 shows why this matters for a pruning comparison: at 0.05 every deletion removes false-positive occupancy, so the same-K fire-rate control gains 8 IoU while changing 13.9% of the full model's occupied decisions, and a random cut gains 1.3 IoU. Only the exact flip cut leaves the decisions comparatively stable (drift 3.1%, +0.12 / +0.32).

The score itself behaves as on the other Gaussian banks: the readout is both-type (emptied share 0.52), the flip score correlates with the ground-truth hit rate (Spearman 0.73) and with splatted mass (0.80) and barely with fire rate (0.12), and the two halves of the calibration frames rank the slots almost identically (split-half Spearman 0.985). The valley falls at the guarded-Otsu boundary and removes 1,372 slots (13.72%). The primary compiled model scores 24.75 / 55.43 before recovery and 24.71 / 54.78 after matched recovery, compared with 24.65 / 54.86 and 24.70 / 54.75 for the corresponding full checkpoints.

The primary efficiency profile strict-loads the exact $K = 8{,}628$ structure and uses two fresh-process 100-forward latency runs. The two runs give slot-dependent reductions of 10.7% and 10.6% and whole-model reductions of 8.0% and 7.9%. Separate memory runs reproduce the 1.486 versus 1.382 GiB allocated peaks in the table. Detailed checkpoint and run provenance will be released with the code.

A.2.6 GaussTR boundary case without stable row identity

Protocol. Compilation is applied only as a boundary test because reference points are resampled on every forward pass and checkpoint rows therefore lack persistent primitive identity. All arms use the released evaluator without recovery.

Quantitative result. GaussTR is the negative control: without stable row identity, Decision-FLIP ranks no better than random, so its 17.8% speedup is the effect of a smaller bank, not of identified redundancy.

Interpretation. In the released GaussTR (Jiang et al., 2025) Talk2DINO (Barsellotti et al., 2025) variant, 300 query embeddings are reused for each of six views, but their reference points are drawn at random on every forward pass. A checkpoint row therefore does not generate the same primitive across frames. The per-query deletion is still exact (20 verified queries match brute-force deletion) and the score distribution passes the separation gate ($\eta = 0.64$, $K = 158$); but at the same K the exact ranking, a random ranking, and fire rate are indistinguishable (11.47 / 11.47 / 11.46 mIoU), and accuracy degrades monotonically with the number of removed queries regardless of which are removed (fixed $K = 200$: 11.83 / 42.45; fixed $K = 150$: 11.38 / 39.78). This is the failure mode predicted by Section 3.1 when the removable unit has no persistent identity. The separation gate does not detect it, which is why eligibility is an architectural condition rather than a score statistic, and why the method makes no compilation claim on such models. Metrics are mIoU / IoU on the Occ3D-nuScenes (Tian et al., 2023) validation split; no recovery epoch is applied to any arm.

L4 latency. Same protocol as the other models (one L4-24Q, PyTorch 2.1.2, fp32, batch size 1, both budgets resident, round-robin, 10 warm-up + 100 measured rounds, mean of two runs; whole-model runs 1047.97 / 1050.39 ms and 860.59 / 863.53 ms, inter-run deviation 0.2% and 0.3%). The timed call is the model's own predict path on one calibration frame with its pre-computed Metric3D (Yin et al., 2023) depth; the frozen DINOv2 backbone is timed through its `forward_features` call.

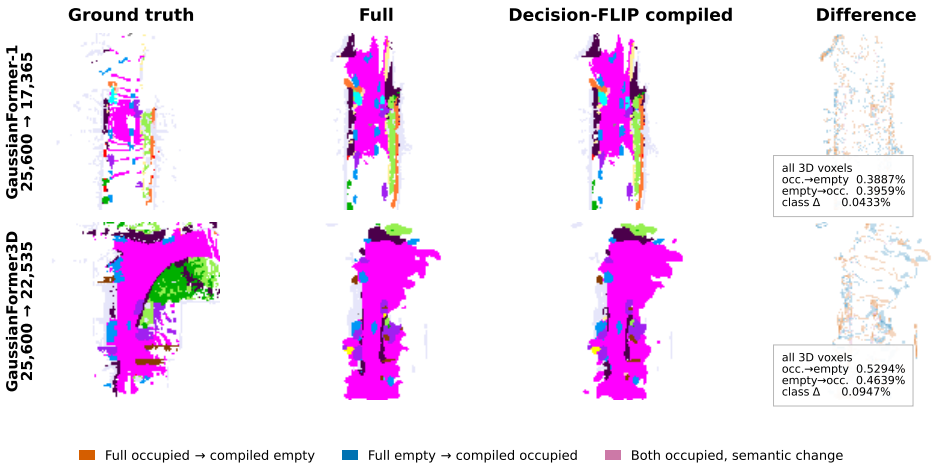


Figure A1: Cross-architecture SurroundOcc qualitative results. Rows show the primary endpoints after matched recovery for GaussianFormer-1 (full $K = 25{,}600$, compiled $K = 17{,}365$) and GaussianFormer3D (full $K = 25{,}600$, compiled $K = 22{,}535$). Columns are ground truth (GT), Full, Decision-FLIP Compiled, and Difference. Difference percentages use all 640,000 voxels in each $200 \times 200 \times 16$ grid as the denominator.

Table A9: Per-class preservation across architectures

Model	Largest class drop	Largest upward change
GaussianWorld	vegetation -0.26	trailer +0.15
GaussianFormer-1	vegetation -0.40	none (best: construction vehicle -0.02)
GaussianFormer3D	manmade -0.18	other flat +0.16
OccFormer (paired evaluation)	other-vehicle -0.59	terrain +0.72
GaussianFlowOcc	trailer -0.33	barrier +0.32

Statistics use every official semantic class after matched recovery; changes are compiled minus matched full in class-IoU points. All five primary matched pairs have complete per-class outputs.

GaussTR is the one model in this study whose Gaussian-side cost dominates: every query becomes a Gaussian per view and the voxelizer splats all of them into the $200 \times 200 \times 16$ grid, so removing 47% of the queries cuts the voxelizer by 45% and the whole model by 17.8%. This saving is what any 47% reduction of the query bank buys; because the ranking has no signal, it is the effect of choosing a smaller bank, not of Decision-FLIP, and it comes with a 0.81-point (6.6% relative) mIoU loss with no recovery. We therefore report it as a boundary case and not as a compression result.

A.3 Cross-architecture qualitative results

The samples in Figures A1 and A2 were selected by a fixed deterministic rule over each complete validation set before inspecting any model output.

A.4 Full per-class results

This section reports every semantic class for the compiled checkpoints, including changes on rare categories. All entries are IoU percentages. The benchmark-specific tables pair the matched full model after one recovery epoch with the adaptive compiled model after the same epoch, grouped by benchmark because the class ontologies differ. Boldface identifies the compiled column for visual grouping; it does not claim that the compiled value is better. All evaluations use the complete evaluation set defined in Appendix A.1. Lifecycle Mean-FLIP reduces exactly to Mean-FLIP for the always-alive GF1, GF3D, OccFormer, and GaussianFlowOcc banks; GaussianWorld uses its streaming lifecycle integral. Endpoint choices that differ between models are given in the table notes.

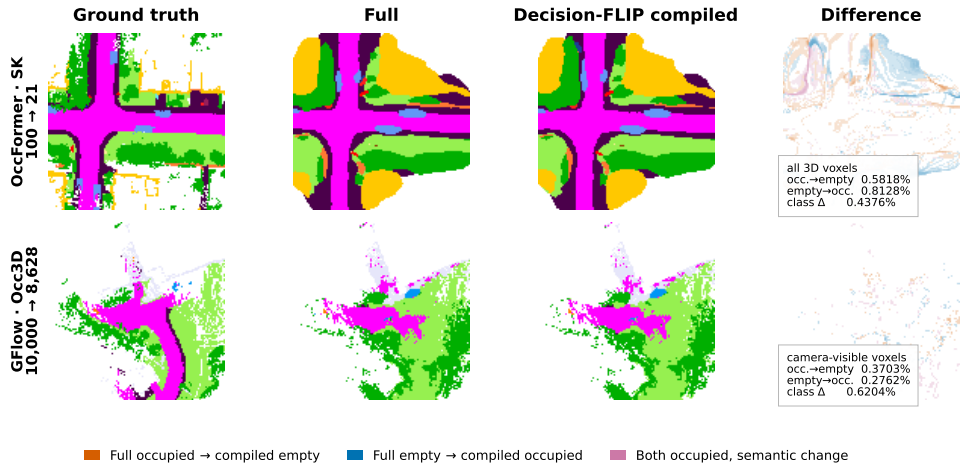


Figure A2: Cross-architecture qualitative results for the exact primary matched-recovery endpoints: OccFormer on SemanticKITTI (full $K = 100$, compiled $K = 21$) and GaussianFlowOcc on Occ3D-nuScenes (Tian et al., 2023) (full $K = 10{,}000$, guarded-Otsu compiled $K = 8{,}628$, $\tau = 0.2$). Both samples were fixed before inspecting model output. Columns are ground truth (GT), Full, Decision-FLIP Compiled, and Difference. OccFormer percentages use all 2,097,152 voxels in its 256×256×32 grid; GaussianFlowOcc percentages use the 130,720 camera-visible voxels in its 200×200×16 grid.

Table A10: SurroundOcc per-class preservation after matched recovery

Metric / class	GW full	GW + ours	GF1 full	GF1 + ours	GF3D full	GF3D + ours
Geometry IoU	33.097	32.978	30.049	29.808	35.496	35.380
barrier	21.77	21.57	19.64	19.44	18.91	18.96
bicycle	13.56	13.45	12.44	12.29	11.99	12.05
bus	26.86	26.72	25.74	25.52	28.03	28.10
car	31.29	31.07	29.41	29.37	31.37	31.40
construction vehicle	13.72	13.47	10.40	10.38	15.01	14.95
motorcycle	16.79	16.91	15.50	15.12	16.93	17.01
pedestrian	13.24	13.23	12.80	12.69	16.26	16.10
traffic cone	11.44	11.51	8.93	8.84	5.81	5.78
trailer	15.03	15.18	12.82	12.43	15.96	15.82
truck	23.65	23.50	20.73	20.55	24.63	24.56
driveable surface	42.23	42.20	39.14	39.10	38.06	38.05
other flat	24.69	24.65	21.82	21.77	18.12	18.28
sidewalk	28.69	28.57	24.49	24.37	23.24	23.25
terrain	26.29	26.32	23.05	22.84	22.57	22.51
manmade	15.64	15.51	11.57	11.41	19.69	19.51
vegetation	25.08	24.82	20.22	19.82	31.83	31.72
Semantic mIoU	**21.874**	**21.792**	**19.293**	**19.121**	**21.152**	**21.128**

Note. GF1 columns are the seed-0 matched-recovery pair, with aggregates 19.293 / 30.049 (full) versus 19.121 / 29.808 (compiled); they are not the three-seed aggregate in Appendix B.1. GF3D uses one fixed matched-recovery run.

Across the five primary settings with complete per-class outputs, no class that is nonzero in its matched full model newly collapses to zero after adaptive compilation and matched recovery. The largest drop is 0.59 points (OccFormer other-vehicle); GaussianFlowOcc's largest drop is 0.33 points (trailer), and its largest gain is 0.32 points (barrier).

Table A12 adds the recovery stage for GaussianWorld and GaussianFormer-1: each class is shown before and after the recovery epoch, so the reader can see how much of the matched gap recovery closes and verify that no class that is nonzero in the matched full model newly collapses to zero.

Table A11: Occ3D-nuScenes (Tian et al., 2023) per-class preservation for GaussianFlowOcc after matched recovery

Metric / class	**Full** $K = 10{,}000$	**Guarded-Otsu** $K = 8{,}628$	**Change**
Occupied IoU	54.75	**54.78**	+0.03
barrier	24.75	**25.07**	+0.32
bicycle	9.16	**9.24**	+0.08
bus	24.85	**24.85**	+0.00
car	31.32	**31.55**	+0.23
construction vehicle	13.06	**13.17**	+0.11
motorcycle	10.33	**10.19**	-0.14
pedestrian	12.91	**12.86**	-0.05
traffic cone	8.64	**8.45**	-0.19
trailer	17.99	**17.66**	-0.33
truck	21.22	**21.21**	-0.01
driveable surface	73.42	**73.47**	+0.05
sidewalk	42.39	**42.48**	+0.09
terrain	45.41	**45.39**	-0.02
manmade	21.49	**21.50**	+0.01
vegetation	13.55	**13.50**	-0.05
Semantic mIoU	**24.70**	**24.71**	**+0.01**

Note. Both columns are the exact matched-recovery endpoints evaluated at $\tau = 0.2$ on all 6,019 validation frames. The repository evaluator averages the 15 classes shown (it excludes `others` and `other flat`). Checkpoint and evaluation-log identifiers will be released with the code.

Table A12: Recovery-stage per-class results for GaussianWorld and GaussianFormer-1

Metric / class	**GW full**	**GW static reference (0 ep)**	**GW static reference (1 ep)**	**GF1 full**	**GF1 adaptive (0 ep)**	**GF1 adaptive (1 ep)**
Geometry IoU	33.097±0.053	32.566	**32.827**	30.049	28.344	**29.808**
barrier	21.77	21.58	21.68	19.64	17.92	**19.44**
bicycle	13.56	12.92	13.57	12.44	10.63	**12.29**
bus	26.86	26.83	26.81	25.74	21.57	**25.52**
car	31.29	30.94	30.80	29.41	26.23	**29.37**
construction vehicle	13.72	13.81	13.60	10.40	9.57	**10.38**
motorcycle	16.79	16.62	16.61	15.50	13.41	**15.12**
pedestrian	13.24	13.11	13.11	12.80	11.35	**12.69**
traffic cone	11.44	11.28	11.15	8.93	7.61	**8.84**
trailer	15.03	14.68	14.72	12.82	11.85	**12.43**
truck	23.65	23.47	23.31	20.73	18.59	**20.55**
driveable surface	42.23	41.93	41.98	39.14	36.39	**39.10**
other flat	24.69	24.49	24.37	21.82	19.40	**21.77**
sidewalk	28.69	28.34	28.50	24.49	22.67	**24.37**
terrain	26.29	25.83	26.13	23.05	21.88	**22.84**
manmade	15.64	15.15	15.41	11.57	11.38	**11.41**
vegetation	25.08	24.65	24.63	20.22	18.17	**19.82**
Semantic mIoU	21.874±0.025	21.490	**21.649**	19.293	17.414	**19.121**

Note. The comparison is matched full one-epoch recovery versus adaptive compiled one-epoch recovery; the "(0 ep)" columns give the same compiled checkpoints before recovery. GW columns retain the static $K = 16{,}461$ reference; the lifecycle result is reported in Appendix A.2.1 and Table A15. GF1 uses the seed-0 matched-recovery pair, and all GaussianFormer values cover 753 validation batches.

Table A13 is the primary recovered per-class comparison for OccFormer; both arms use the same one-epoch recipe and cover all 815 validation frames.

Table A14 lists evaluator anchors. Its columns anchor the provenance chain (released $\rightarrow$ source $\rightarrow$ compiled) and establish evaluator comparability only; they are not matched controls, and no pruning conclusion rests on them.

A.5 Full compilation and matched-recovery table

Table A15 gives one primary row per eligible setting: the full and adaptive compiled checkpoints before recovery, then the same pair after matched one-epoch recovery. The metric pair and benchmark are explicit in every row, and every number is a percentage. Differences are compiled minus full. Model-specific matched-recovery evidence remains in the architecture tables, with the GaussianFormer-1 seed control in Appendix B.1.

Table A13: OccFormer per-class sensitivity to budget and recovery recipe

Class	**Matched full** $K = 100$	**Otsu** $K = 21$
car	24.49	24.89
bicycle	1.39	1.32
motorcycle	1.44	1.10
truck	17.75	17.51
other-vehicle	12.15	11.56
person	3.27	3.17
bicyclist	4.34	3.94
motorcyclist	0.00	0.00
road	58.99	58.89
parking	18.96	18.93
sidewalk	25.02	25.66
other-ground	0.30	0.35
building	14.49	14.65
fence	5.15	5.33
vegetation	19.50	19.52
trunk	3.57	3.63
terrain	29.49	30.21
pole	3.96	3.97
traffic-sign	2.63	2.71
SSC mIoU	**12.99**	**13.02**
SC IoU	**36.22**	**36.30**

Primary paired evaluation. Compiled minus full is +0.03 SSC mIoU / +0.08 SC IoU. Other-vehicle has the largest class drop (-0.59); terrain has the largest increase (+0.72); no nonzero full class newly collapses.

Table A14: Released-checkpoint per-class evaluator anchors

Metric / class	**GW released `ckpt_stream`**	**GF1 released solid**
Geometry IoU	32.804	28.779
barrier	21.47	17.86
bicycle	13.53	10.60
bus	27.19	22.43
car	31.18	26.08
construction vehicle	13.77	9.49
motorcycle	16.20	13.35
pedestrian	13.31	11.31
traffic cone	11.85	7.55
trailer	14.85	12.08
truck	23.45	18.73
driveable surface	42.00	37.02
other flat	24.49	19.19
sidewalk	28.44	22.84
terrain	26.32	22.02
manmade	15.60	11.90
vegetation	24.49	18.69
Semantic mIoU	21.759	17.571

Note. The same architecture-specific evaluators are used throughout Appendix A.

Two regularities hold across the eligible model-dataset settings. First, automatic compilation preserves accuracy where the bank has stable row identity: before recovery, semantic mIoU falls by at most 0.179 point and the second reported metric by at most 0.435 point; GaussianFlowOcc instead improves on both metrics. Second, after matched recovery every semantic gap is at most 0.171 point and every geometry/occupancy gap is at most 0.246 point. GaussianFlowOcc remains 0.03 occupied-IoU point above its full control. Where recovery changes the absolute level substantially (GaussianFormer-1 +1.7 mIoU and GaussianFormer3D +2.7), the shift is pipeline adaptation shared by both arms, which is why only matched pairs are compared.

Table A15: Model-dataset compilation and matched-recovery results

Model (benchmark; metric pair)	Eligible $K \rightarrow$ kept (removed)	Full, 0 ep	Ours, 0 ep	Δ 0 ep	Full, matched 1 ep	Ours, matched 1 ep	Δ 1 ep
GaussianWorld (SurroundOcc; mIoU / geometry IoU)	25,600 → 22,427 (12.4%)	21.902 / 33.140	21.723 / 32.850	-0.179 / -0.290	21.874±0.025 / 33.097±0.053	21.792 / 32.978	-0.082 / -0.119
GaussianFormer-1 (SurroundOcc; mIoU / geometry IoU)	25,600 → 17,365 (32.2%)	17.571 / 28.779	17.414 / 28.344	-0.157 / -0.435	19.291±0.002 / 30.050±0.002	19.120±0.003 / 29.804±0.001	-0.171±0.002 / -0.246±0.001
GaussianFormer3D (SurroundOcc; mIoU / geometry IoU)	25,600 → 22,535 (12.0%)	18.400 / 30.056	18.356 / 29.963	-0.044 / -0.093	21.152 / 35.496	21.128 / 35.380	-0.024 / -0.116
OccFormer (SemanticKITTI; SSC mIoU / SC IoU)	100 → 21 (79.0%)	13.50 / 36.42	13.34 / 36.42	-0.16 / 0.00	12.99 / 36.22	13.02 / 36.30	+0.03 / +0.08
GaussianFlowOcc (Occ3D; voxel mIoU / occupied IoU)	10,000 → 8,628 (13.72%)	24.65 / 54.86	24.75 / 55.43	+0.10 / +0.57	24.70 / 54.75	24.71 / 54.78	+0.01 / +0.03

Table A16: Same-K ranking controls before recovery

Model (metric pair)	K	Full	Exact Mean-FLIP	Taylor	Random	Fire/activity	Opacity	Magnitude/query norm
GaussianWorld (mIoU / geometry IoU)	22,427	21.902 / 33.140	**21.723 / 32.850**	21.708 / 32.762	21.512 / 32.342	21.650 / 32.665	21.658 / 32.699	21.576 / 32.388
GaussianFormer-1 (mIoU / geometry IoU)	17,365	17.571 / 28.779	**17.414 / 28.344**	16.349 / 25.961	9.261 / 11.809	8.919 / 11.582	17.388 / 27.566	17.097 / 27.319
GaussianFormer3D (mIoU / geometry IoU)	22,535	18.400 / 30.056	**18.356 / 29.963**	18.325 / 29.873	17.67 / 28.44	18.30 / 29.86	18.295 / 29.915	18.329 / 29.843
OccFormer (SSC mIoU / SC IoU)	21	13.50 / 36.42	**13.34 / 36.42**	5.52 / 7.88	0.82 / 7.89	13.17 / 36.41	—	1.53 / 7.88
GaussianFlowOcc, primary sequential guarded Otsu (voxel mIoU / occupied IoU)	8,628	24.65 / 54.86	**24.75 / 55.43**	24.75 / 55.41	unavailable	unavailable	unavailable	unavailable

A.6 Full ablations and diagnostics

A.6.1 Same-K ranking quality before recovery

The same-K comparison fixes each architecture at the survivor count selected by its adaptive Exact Mean-FLIP/Otsu arm and changes only the ranking. Every number is from the same source checkpoint and complete official validation; no arm receives recovery. This isolates score quality from both budget selection and fine-tuning. Unavailable entries denote controls that are undefined or have not completed under the same strict protocol.

Exact Mean-FLIP matches or exceeds supervised Taylor on the primary semantic metric in all five settings. The largest margins occur on GaussianFormer-1 and OccFormer; GaussianFlowOcc ties in mIoU, with Mean-FLIP 0.02 point higher in occupied IoU. Random selection collapses GaussianFormer-1 and OccFormer. Thus the completed same-K results rule out the explanation that any cut of the selected size works. GaussianFlowOcc's other primary $K = 8{,}628$ proxy controls are unavailable.

Matched recovery controls answer a different question: whether fine-tuning can erase ranking differences. At the GaussianFormer-1 fixed comparison budget $K = 17{,}296$ (Appendix A.2.2), recovery repairs part, but not all, of the random cut: the random arm ends 0.6 mIoU below the exact and opacity arms after the same epoch. We do not extrapolate this result to architectures whose matched recovered proxy arms have not completed.

A.6.2 Decision criterion and thresholding ablation

The exact leave-one-out score can count changed decisions (hard FLIP, the criterion used throughout) or accumulate the total-variation distance between the full and leave-one-out class distributions (TV-FLIP, $s_i^{TV} = \frac{1}{2}\sum_x \|p_f(x) - p_{f,-i}(x)\|_1$); and the Otsu split can be applied once or re-applied to the low mode after each removal round (iterative Otsu). We ran all four combinations on the released GaussianFormer-1 checkpoint with 100 training calibration frames and 50 disjoint held-out training frames, and report the exact set-removal drift of the compiled readout on held-out frames (fraction of voxels the full model predicts occupied whose argmax changes after removing the whole set), with matched-K masks so that every method is read at every budget. This independent 100-frame criterion subset is distinct from the nested 100-frame prefix used in the calibration-size sweep; consequently, their selected K values are not expected to coincide.

Random and fire-rate coincide in this evaluation because all slots exceed the fire threshold on the independent 100-frame subset, leaving fire-rate non-discriminative; the saved masks and resulting drifts are therefore identical.

Both scores are reliable rankings (split-half Spearman 0.925 hard, 0.929 TV) but they rank different slots (Spearman 0.39 between them): TV keeps diffuse slots that move probability mass without changing decisions and drops small decisive ones, so at every budget it is worse on the quantity compilation must preserve. Iterative re-thresholding does not improve the ranking (at $K = 14{,}328$ it reads 0.164 against 0.171 for single-shot on this one held-out split; no variance estimate is available); it only cuts deeper, and the separation gate does not stop it: the TV variant ran eight rounds down to $K = 76$ with η between 0.55 and 0.70 at every round. The separation gate is therefore a plausibility check on one split, not a stopping rule, and the budget of Section 3.5 is the single-shot valley. The

Table A17: Hard and total-variation FLIP with single-shot and iterative Otsu

Held-out occupied-voxel argmax drift	$K = 19{,}830$ **(TV Otsu)**	$K = 18{,}769$ **(hard Otsu)**	$K = 14{,}328$ **(hard, iteration 2)**
Hard FLIP, single-shot	**0.040**	**0.055**	**0.171**
Hard FLIP, iterative Otsu	0.144	0.148	0.164
TV-FLIP, single-shot	0.239	0.296	0.516
TV-FLIP, iterative Otsu	0.515	0.555	0.798
Random ranking	0.596	0.669	0.879
Fire-rate ranking	0.596	0.669	0.879

Table A18: Label supervision and self-distillation during recovery

GaussianWorld, $K = 16{,}461$**, one epoch, 5,700 frames**	**Full**	**Compiled**	Δ
Labels (static-reference recipe)	21.874 / 33.097	21.649 / 32.827	-0.225 / -0.270
Labels + self-distillation	21.843 / 32.870	21.586 / 32.489	-0.257 / -0.381
Self-distillation only (label-free)	—	21.581 / 32.420	—

remaining 5.5% drift at the hard Otsu budget is a joint-removal cost that no re-ranking removes; it is the cost the matched recovery epoch addresses.

A.6.3 Label supervision and self-distillation during recovery

The compiled model is selected to preserve the full model's decisions, whereas the recovery epoch fits ground-truth labels. To test whether the residual gap of the static GaussianWorld reference is an artifact of that mismatch, we repeated recovery with a frozen full-bank teacher (the calibrated source checkpoint, run in lock-step through the same 38-frame streams) and a per-voxel KL term between teacher and student voxel logits (weight 100, temperature 1), either added to the task loss or replacing it, with the unpruned bank receiving the identical recipe as control. Everything else (one epoch, seed, learning rate, evaluator) matches the static reference.

Adding the KL term does not close the gap: both arms end slightly below the label-trained ones (the geometry IoU by 0.2 to 0.4), the compiled-versus-full difference is unchanged (-0.26 / -0.38 against -0.23 / -0.27), and the label-free and mixed variants end within 0.005 mIoU on this run. The residual cost of the 36% cut is therefore not a property of the recovery objective.

A.6.4 Recovery interventions at fixed GaussianWorld budgets

Earlier exploratory runs on the same evaluator varied the recovery itself at fixed budgets (Mean-FLIP ranking, one epoch unless stated; 5,700 frames); Table A19 lists them.

None of these interventions improves the endpoint. A larger anchor learning rate and longer training land where standard recovery lands, progressive pruning lands below it, and at the 21% budget recovery changes nothing at all. Relocating the low-scored anchors instead of deleting them keeps the full bank's cost yet ends 0.17 mIoU below the recovered full control, so moving idle slots is not a free alternative to removing them. Together with the criterion ablation above, this places the residual cost with the capacity of the smaller bank rather than with the optimization of the recovery.

A.6.5 Label-free empty-vote recalibration

The GaussianWorld head adds one fixed empty Gaussian whose vote into the empty class scales with a learned scalar. Compilation removes scene Gaussians but not this vote, so voxels that lost part of their mass tip towards empty; this is why the zero-recovery loss is concentrated in geometry (the Δ 0 ep column of Table A15). Because the empty vote enters the argmax linearly, rescaling the scalar by α is exact and can be replayed from one head pass. We chose α label-free on 100 training frames so that the compiled model's occupied-voxel rate matches the full model's, and read the result on 50 held-out training frames under the local single-frame protocol (absolute values are not comparable with the streaming evaluation). Table A20 reports the result.

The single global rescale returns one third to one half of the geometry loss and costs a little mIoU, because the missing vote mass is spatially uneven: voxels whose Gaussians were removed lose

Table A19: Recovery interventions at fixed GaussianWorld budgets

Lever (GaussianWorld)	K	**mIoU / IoU**
Standard recovery, three seeds	13,364	21.28 / 32.29, 21.38 / 32.41, 21.36 / 32.38
Anchor-position learning rate $\times 4$	13,364	21.37 / 32.34
Standard recovery	20,271	21.72 / 32.99 (compiled before recovery: 21.73 / 32.96)
Relocating instead of deleting: the 5,329 lowest-scored anchors are moved, none removed	25,600	21.31 / 32.42 before recovery, 21.70 / 32.94 after
Progressive pruning in three rounds, last round	13,364	21.15 / 32.07
Three recovery epochs (uncalibrated source)	13,364	20.83 $\rightarrow$ 21.24 $\rightarrow$ 21.26 after epochs 1, 2, 3

Table A20: Label-free empty-vote recalibration

GaussianWorld, $K = 16{,}461$, 0 recovery, 50 held-out train frames	**mIoU / IoU**
Full bank	0.2803 / 0.4476
Compiled, $\alpha = 1$	0.2761 / 0.4244
Compiled, $\alpha^* = 0.976$ (occupancy-matched)	0.2746 / 0.4316
Compiled, $\alpha = 0.95$	0.2726 / 0.4359

more than a global factor restores. It is a partial, recovery-free fix, not a substitute for recovery. Per-Gaussian opacity compensation (a least-squares reweighting of the survivors' opacities, also label-free) is the natural extension and is left for future work.

A.6.6 AUTOMATIC-SELECTOR COMPARISON

Each selector operates on the same fixed Mean-FLIP evidence within an architecture and chooses its own K without validation metrics. The cross-architecture comparison uses zero-recovery semantic mIoU relative to each model's evaluated full source checkpoint, avoiding differences among recovery recipes.

GaussianWorld's Otsu, Kneedle, and largest-gap rules produce the same physical mask; Kneedle and largest-gap likewise coincide on GaussianFlowOcc. The remaining differences illustrate the operating-point trade-off: conservative rules can preserve accuracy by removing almost nothing, whereas aggressive rules can collapse the model or incur a larger loss.

The detailed GaussianFormer-1 comparison fixes its saved 500-frame Mean-FLIP vector and per-frame matrix before any validation result is read. Each non-abstaining mask is physically compiled, loaded under strict shape and key checking, evaluated on all 753 official validation batches, recovered for one matched epoch, and evaluated again. The common full matched-recovery control is 19.293 mIoU / 30.049 geometry IoU.

Largest-gap removes only the nine rows with exactly zero Mean-FLIP ($K = 25{,}591$). Identifiers for the mask, score vector, frame matrix, pinned full-control metrics, and aggregate summary are recorded in a provenance manifest that will be released with the code. An independent run produced byte-identical masks and complete 753-batch evaluations. No validation metric participates in selection. The comparison is not matched in K, and it covers one checkpoint, calibration set, and recovery seed, so it establishes an operating-point trade-off rather than universal selector optimality.

A.6.7 SEPARATION AND CALIBRATION DIAGNOSTICS

The identity eligibility check and score-stability check serve different purposes. Identity is an architectural or lineage property (Section 3.1); split-half statistics ask whether a finite calibration sample estimates a stable ranking or threshold. Table A23 lists the diagnostics; Figure A3 plots the GaussianFormer-1 log-score distribution and Otsu separation curve.

† GaussianWorld split statistics are from an independent strict rerun over the same ten 38-frame training streams; its full $K = 22{,}463$ differs from the primary 22,427 by 36 rows.

All five score arrays in the automatic-selector study reproduce their reported budgets. Split-half mask Jaccard is 0.859–0.960. Rank agreement is lower for the streaming GaussianWorld rerun (0.776) and OccFormer (0.779) than for the other models (0.975–0.985), yet their masks remain stable at 0.859 and 0.955. GF3D illustrates a distinction: the two half-sample masks agree strongly with one another, yet their guarded-Otsu cuts are more conservative than the full-sample cut. We therefore

Table A21: Cross-architecture automatic-selector operating points before recovery

Model	Guarded Otsu (removed / ΔmIoU)	Two-component GMM (removed / ΔmIoU)	Kneedle (removed / ΔmIoU)	Largest-gap (removed / ΔmIoU)
GaussianWorld	12.4% / -0.179	22.0% / -0.544	12.4% / -0.179	12.4% / -0.179
GaussianFormer-1	32.2% / -0.157	45.2% / -0.552	99.6% / -17.531	0.035% / -0.001
GaussianFormer3D	12.0% / -0.044	9.8% / -0.030	99.4% / -17.940	2.4% / -0.006
GaussianFlowOcc	13.7% / +0.10	41.1% / -0.24	12.8% / +0.09	12.8% / +0.09
OccFormer (SemanticKITTI)	79% / -0.16	63% / 0.00	76% / -0.02	3% / +0.06

Table A22: GaussianFormer-1 recovered-selector detail

Selector	Status	K (removed)	Split-half K_A/K_B	Mask Jaccard	Before recovery mIoU / geometry IoU	After matched recovery mIoU / geometry IoU	Recovered Δ vs. full
Guarded Otsu	selected	17,365 (32.168%)	17,442 / 18,019	0.928	17.413 / 28.344	19.120 / 29.804	-0.173 / -0.245
Two-component GMM	selected	14,035 (45.176%)	14,261 / 15,750	0.890	17.019 / 27.313	18.928 / 29.543	-0.365 / -0.507
Kneedle	selected, degenerate operating point	92 (99.641%)	103 / 114	0.428	0.040 / 0.034	2.076 / 2.352	-17.217 / -27.698
Largest-gap	selected	25,591 (0.035%)	25,577 / 25,563	0.998	17.570 / 28.779	19.290 / 30.051	-0.003 / +0.002

report threshold, K, rank, and mask stability separately rather than treating one statistic as an identity gate. The provenance JSON/CSV records source identifiers.

The saved per-frame matrices also permit a prefix calibration-size sweep without rerunning or reading validation labels (Table A24).

Ranking and mask agreement improve with more frames, but K does not converge monotonically: relative to 500 frames, the 100/200-frame selections differ by 1,018/1,016 rows on GF1 and 830/1,541 rows on GF3D. This supports the ranking signal while showing that the automatically chosen operating point is calibration-size dependent.

Exact scoring is vectorized over candidate rows rather than implemented as 25,600 physical forwards; Table A25 gives the wall times.

Times are single-run wall times from the saved A100 scorer logs, except OccFormer, which was scored on one L4. They include model forwarding and exact vectorized readout interventions, but exclude compilation, validation, and recovery. Cross-model timing is therefore descriptive rather than a hardware-normalized comparison.

B Matched recovery seed control

B.1 GaussianFormer-1 three-seed matched recovery

The $K = 17{,}365$ mask is frozen before recovery. Full and compiled arms use the same one-epoch recipe, and every evaluation covers all 753 validation batches.

The mean semantic gap is 0.171 point and the largest observed gap across the three reported seeds is 0.173 point. The matched gap is much larger than its seed spread, so the small reduction is repeatable rather than an artifact of one recovery seed.

C GaussianWorld temporal selection diagnostics

All results below are mechanism diagnostics at fixed $K = 16{,}461$, evaluated on paired 400-frame subsets. They do not replace the 5,700-frame lifecycle-compiled result in Appendix A.2.1.

C.1 Static, lifecycle-integrated, and composite scoring

Lifecycle-aware rankings improve subset mIoU and geometry IoU at the same budget, although occupied-voxel drift does not improve monotonically. This is why the paper uses the direct lifecycle integral and guarded Otsu rather than promoting the exploratory composite.

C.2 Fixed compilation and causal routing

The static mask's large drift shows that a row selected on isolated readout evidence need not retain the same role after recurrent state evolution. Dynamic routers recover much of the oracle ranking signal, but keeping all 25,600 rows in temporal state and routing only before aggregation yields little

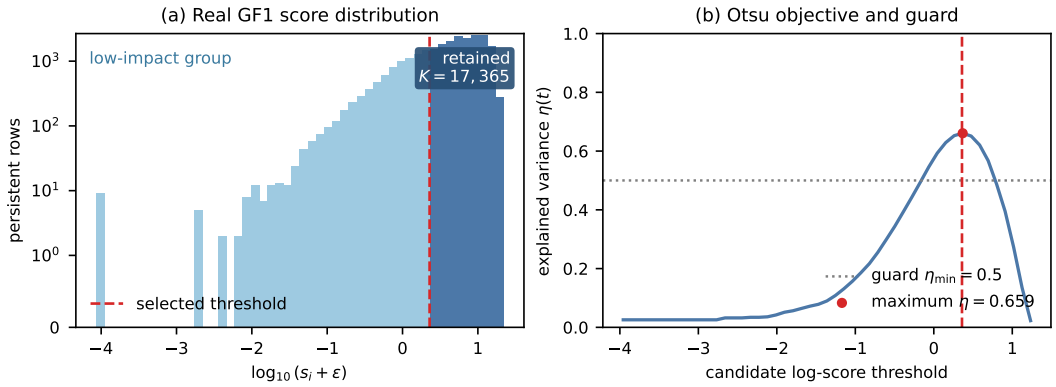


Figure A3: Detailed guarded-Otsu diagnostic on GaussianFormer-1. Left: the saved log MeanFLIP distribution and selected partition. Right: normalized between-group variance over candidate thresholds; guarded Otsu compiles only when the maximizing separation exceeds the fixed gate. This diagnostic is retained as distributional evidence and does not imply universal bimodality or accuracy optimality.

Table A23: Separation and calibration diagnostics

Model	**Calibration**	η	**Split-half** K_A/K_B	**Rank** ρ_s	**Mask Jaccard**	**Full** K
GaussianWorld lifecycle	380 streaming frames	0.8406	20,703 / 21,065†	0.776†	0.859†	22,427 / 25,600
GaussianFormer-1	500 frames	0.6593	17,442 / 18,019	0.983	0.928	17,365 / 25,600
GaussianFormer3D	500 frames	0.5688	24,478 / 24,506	0.975	0.960	22,535 / 25,600
OccFormer	200 frames	0.7950	21 / 22	0.779	0.955	21 / 100
GaussianFlowOcc	300 scored / 8,971 forwarded	0.8556	8,517 / 8,522	0.985	0.959	8,628 / 10,000

end-to-end speedup. These diagnostics motivate lineage-aware *physical compilation* rather than a runtime router.

Table A24: Calibration-size sweep: survivor count and split-half stability versus scored frames

Model	**Frames**	K	η	**Split-half** K_A/K_B	ρ_s	**Jaccard**
GaussianFormer-1	100	18,383	0.601	22,234 / 20,697	0.925	0.877
GaussianFormer-1	200	18,381	0.630	19,700 / 18,770	0.959	0.902
GaussianFormer-1	500	17,365	0.659	17,442 / 18,019	0.983	0.928
GaussianFormer3D	100	23,365	0.794	22,226 / 21,980	0.909	0.892
GaussianFormer3D	200	24,076	0.735	23,103 / 23,239	0.947	0.925
GaussianFormer3D	500	22,535	0.569	24,478 / 24,506	0.975	0.960

Table A25: Scoring wall time per model

Model	**Calibration work**	**Scoring wall time**
GaussianWorld lifecycle	10×38-frame streams	1.8 min
GaussianFormer-1	500 frames	30.9 min
GaussianFormer3D	500 frames	31.5 min
OccFormer	200 frames	3.8 min (L4)
GaussianFlowOcc	300 scored among 8,971 sequential forwards	13.9 min

Table B1: GaussianFormer-1 three-seed matched recovery

Recovery seed	**Full mIoU / geometry IoU**	**Compiled mIoU / geometry IoU**	**Compiled − full**
0	19.2927 / 30.0494	19.1198 / 29.8044	-0.1730 / -0.2450
1	19.2883 / 30.0527	19.1168 / 29.8051	-0.1715 / -0.2476
2	19.2907 / 30.0490	19.1225 / 29.8024	-0.1681 / -0.2466
Mean ± sample std.	**19.2906±0.0022 / 30.0503±0.0021**	**19.1197±0.0029 / 29.8040±0.0014**	**-0.1709±0.0025 / -0.2464±0.0013**

Table C1: Static versus lifecycle-integrated scoring

Fixed-K ranking	**Semantic mIoU**	**Geometry IoU**	**Occupied-voxel argmax drift**
Static Mean-FLIP	20.041	33.626	19.60%
Lifecycle Mean-FLIP	20.304	34.064	20.27%
Lifecycle composite (mean/CVaR/frequency)	20.343	34.086	20.15%

Table C2: Fixed masks and dynamic routers

Readout policy at $K = 16{,}461$	**Δ mIoU vs. paired full**	**Δ Geometry IoU**	**Occupied drift**	**Mean-FLIP mass recall**	**Latency reduction**
Static fixed mask under full history	-10.823	-22.090	77.15%	—	—
Per-frame Decision-FLIP oracle, full history	-0.157	-0.489	1.82%	100%	oracle cost excluded
Vote-mass router, free rollout	-0.417	-1.358	18.27%	—	0.51%
Learned MLP router, free rollout	-0.372	-0.492	14.76%	98.43%	1.82%
Causal selective-SSM router, free rollout	-0.804	-1.092	18.73%	96.86%	0.07%