# WHAT MOTION COMPENSATION REMOVES FROM ACCUMULATED RADAR AND WHEN IT STILL HELPS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Accumulating radar sweeps requires choosing where to place past returns before spatial rendering. Doppler-based motion compensation can concentrate returns around moving objects, but it also alters spatial trails that may support velocity estimation. We examine this trade-off through geometric measurements, linear probes of frozen encoders, and controlled detector comparisons on nuScenes and TruckScenes. On nuScenes, compensation reduces the correlation between normalised trail extent and object speed from $0.764$ to $0.297$ over ten sweeps, while increasing in-box return counts for fast-moving objects by $54\%$. A displacement-matched permutation reproduces neither effect. Object speed is more linearly accessible from raw-placement features, with an $R^2$ advantage of $0.019$ to $0.032$ across eight frozen encoders. In detector comparisons differing only in placement, raw placement yields lower velocity error in eight of nine paired comparisons across three systems and three training seeds, whereas compensation provides less consistent detection benefits. On TruckScenes, compensation reduces aggregate in-box concentration across moving objects at every evaluated accumulation window. Near-radial objects are the exception and the locus of both effects: there concentration instead improves and the trail-speed correlation is removed, an association with radial motion fraction that we do not explain. We further decompose the end-to-end gain of a detector retaining both placements. Along the evaluated sequence of controlled comparisons, correcting a Doppler coordinate-frame inconsistency in the reference pipeline evaluated here accounts for $69\%$ and $81\%$ of the mAP gain at two seeds; a displacement-matched, correspondence-breaking control recovers part of the remaining gain. These findings show that spatial concentration and access to motion cues can favour different placements, and that attributing downstream gains to placement requires controlling for changes in input features and architecture.

## 1 INTRODUCTION

Radar returns are sparse enough that detectors accumulate several sweeps before doing anything else (Yao et al., 2025), and accumulating forces a choice the measurements do not resolve. A return recorded $t$ seconds ago has to be given a position in the current frame, and for a moving object there are two defensible answers: leave it where it was measured, or move it by its own Doppler to where that Doppler implies it has since travelled. Both are computable from the measurements alone, and they produce different tensors. The second of them, motion compensation, systematically removes motion-dependent spatial evidence, while the benefit obtained in exchange is neither universal nor guaranteed to reach the detector.

Those tensors carry different information (Figure 1), and the difference is measurable before any network is trained. Left in place, a moving object's returns form a trail whose extent correlates with its speed at $+0.76$ over ten sweeps; compensating each return removes most of that correlation while concentrating returns inside fast movers. Both effects grow with accumulation depth, both are near zero at a single sweep, and a displacement-matched control that breaks the correspondence between a displacement and its own return reproduces neither. The localisation gain is conditional in a way the loss of the trail-borne cue is not: on a second dataset, where much of the motion is not radial, that gain reverses at every depth (§5).



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.

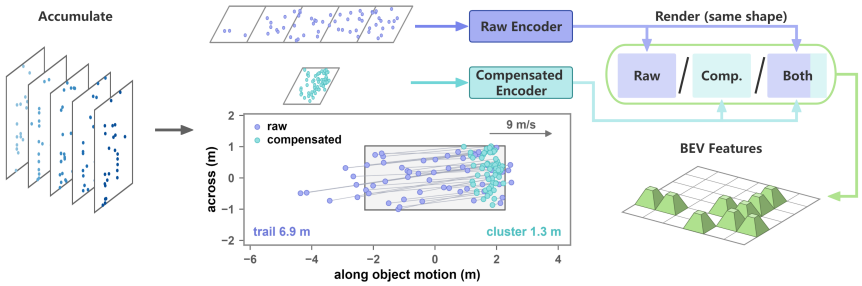


Figure 1: The placement choice accumulation forces, and what each answer preserves. The inset shows one nuScenes vehicle at 9 m/s, its 63 returns drawn at both addresses and joined by a segment; raw placement spreads them over 6.9 m, compensation collapses them to 1.3 m. All three renderings produce a tensor of the same shape and width.

Placement affects velocity estimation more consistently than detection accuracy. In comparisons whose arms differ only in which placement they render, the velocity direction holds in eight of the nine paired comparisons and the detection direction in six: a representation-level gain does not guarantee downstream use.

Representation changes are ordinarily evaluated end to end, against the detector the new one was built from (Palmer et al., 2023; Haitman & Bialer, 2025); that comparison gives $+2.27$ and $+2.16$ mAP at two seeds and does not isolate the placement. Training each step as its own arm attributes most of it to a Doppler frame defect in the reference, and the speed-dependent signature that seemed to confirm the placement account belongs to that defect too (§6.3, §6.4).

This paper makes two contributions. The first characterises what placement costs and what it buys, across the geometry, a frozen encoder's linearly extractable features, a detector's output and a second dataset. The second separates an end-to-end gain into controlled increments and shows that a speed-dependent improvement does not by itself identify the placement as the mechanism. Our evidence is radar, and the attribution problem it exposes is not obviously specific to that setting.

## 2 Spatial Rendering

### 2.1 Radar accumulation and motion compensation

The detectors we study discretise a point cloud into pillars, encode each, and scatter the result to a bird's-eye-view map on which a backbone and head operate (Lang et al., 2019; Yin et al., 2021). Fusion methods scatter every modality into the same grid (Liu et al., 2023), and radar detectors inherit the pipeline (Bang et al., 2024), adapting pillar encoders built for LiDAR (Shi et al., 2022). The accumulated representation is consumed downstream for tracking as well (Xu et al., 2026). Radar variants accumulate several sweeps first, fixing two things: how many, and where to put a return measured earlier. The sweep count is reported as a hyper-parameter; the placement is treated as a correction rather than a choice with a cost. Palmer et al. (2023) study that placement directly, separating dynamic motion in an accumulated radar cloud with a learned instance-motion estimate and reporting the resulting detection gain.

Motion compensation is established practice and we claim no part of it. Haitman & Bialer (2025) shift returns from earlier frames radially according to their Doppler component to remove the scatter that moving objects otherwise produce, and additionally give each return its own aggregation duration to limit the tangential scatter a radial correction cannot address, reporting improved detection across detectors and datasets. The compensated placement of §2.2 is this construction. That evaluation and Palmer et al.'s both use average precision alone, so a cost appearing in velocity and not in precision would be invisible to them, and we know of no evaluation that isolates the placement-dependent velocity information compensation removes. That full velocity needs more than radar Doppler alone is known: Long et al. (2021) resolve the tangential component by fusing radar with camera, consistent with §3.5.

The choice arises wherever accumulated radar points are rendered to a grid, which includes the distillation pipeline we evaluate, whose radar branch is trained to match a LiDAR teacher's features (Bang et al., 2024), and query-based fusion, whose radar branch pillar-renders to bird's-eye view before its queries sample it (Chu et al., 2025). Two designs avoid it (Appendix L): fully sparse detectors that keep encoded points (Wolters et al., 2025), and occupancy models that consume the raw radar tensor rather than a point cloud (Ding et al., 2024). Dual-stream radar designs are dual in a different sense: Lin et al. (2024a;b) pair a point-based with a transformer-based encoder over the same placement, so our distinction is between placements, not encoder types. Nor are we aware of prior work evaluating a radar rendering choice by decomposing its end-to-end effect into matched single-factor arms.

### 2.2 Placement and rendering operators

Write a radar measurement as $m_i = (p_i, d_i, t_i, f_i)$: the position at which the return was recorded, in the reference frame but not corrected for object motion; the ego-motion-compensated radial Doppler measured with it; the lag between its sweep and the reference time; and the per-point features the encoder receives, which repeat some of these and differ by pipeline (Appendix G). A placement assigns each measurement a location in the reference frame, and accumulation makes two available without any oracle or learned component,

$$\varphi_{\text{raw}}(m_i) = p_i, \qquad \varphi_{\text{comp}}(m_i) = p_i + d_i\, t_i\, \hat{u}_i, \tag{1}$$

the position at which the return was measured, and the position a radial constant-velocity extrapolation of its own Doppler implies for the reference time. The compensated position is the object's later position only where that assumption holds, and §5 reports a dataset where it often does not. Here $\hat{u}_i$ is the unit sensor-to-return ray and $t_i \geq 0$ the lag. The signed radial Doppler $d_i$ is the only component of velocity a single return measures, which is what makes $\hat{u}_i$ the only direction available: over the validation split the reported Doppler vector is parallel to $\hat{u}_i$ for every return checked.

A rendering assigns each measurement to a cell of a fixed grid and pools what lands there. Write $\kappa(x)$ for the index of the cell containing $x$, and $\mathcal{I}_\varphi(\kappa) = \{i : \kappa(\varphi(m_i)) = \kappa\}$ for the returns a placement sends to that cell. Then

$$R_\varphi[\kappa, :] = \max_{i \in \mathcal{I}_\varphi(\kappa)} \text{enc}\big(f_i \| \delta_i\big) \in \mathbb{R}^C, \qquad R_{\text{both}}[\kappa, \mathcal{G}_g] = \max_{i \in \mathcal{I}_{\varphi_g}(\kappa)} \text{enc}_g\big(f_i \| \delta_i\big), \tag{2}$$

element-wise maxima over encoded returns, with $\delta_i$ each return's offsets to its cell's mean and centre, taken within its own group. The second operator keeps both placements, $\varphi_1 = \varphi_{\text{raw}}$ and $\varphi_2 = \varphi_{\text{comp}}$: every measurement is rendered twice, once into each group, and one $m_i$ may reach different cells under the two. The channel axis is partitioned, $\mathcal{G}_1 \sqcup \mathcal{G}_2 = \{1, \ldots, C\}$, an empty group pooling to zero. Same grid, same total width $C$, same encoder family; what differs is which returns reach which encoder and which channels they are written to (Appendix G gives the widths per system).

### 2.3 Scope

Whether the choice can be undone depends on what $f_i$ keeps, and pipelines differ. Inverting compensation needs the lag; our CenterPoint bed retains it and the PillarNet-radar student does not, a difference §3.5 settles. Wherever $\varphi_{\text{raw}}(m_i) \neq \varphi_{\text{comp}}(m_i)$, retaining both requires a different rendering operator, which §6 evaluates. Ground-truth object velocity enters this paper only in object-level analysis and evaluation, never inside a placement.

We distinguish three levels of evidence. What a rendering makes present can be answered on the geometry alone, before an encoder exists. What is extractable from the tensor the detector receives takes a probe fitted on a frozen encoder. What a learner uses takes training and a comparison. The first two are the distinction probing studies draw between information being present in a representation and being recoverable from it (Alain & Bengio, 2018; Pimentel et al., 2020; Voita & Titov, 2020); radar is unusual in that presence can be settled with no probe at all. A detector-level difference is evidence about use only to the extent that the placement is the sole thing that changed between the arms compared.

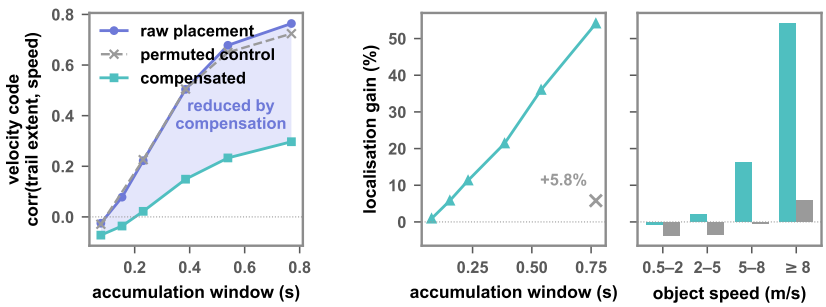


Figure 2: The two quantities against accumulation depth, with the permuted control. Compensation removes most of the velocity code; the permutation of the same field does not. Left: the velocity code under each placement. Centre: localisation gain for fast movers ($\geq 8$ m/s) against accumulation depth, where the control reaches $+5.8\%$. Right: the same gain against speed, the other factor of $d_i t_i$. Appendix B tabulates both.

## 3 The Representation-Level Phenomenon

This section measures two cues each placement leaves in the geometry, and what a probe reads of one of them, on the nuScenes validation split (Caesar et al., 2020). §3.1 to §3.3 compute every quantity on the geometry of the accumulated point cloud, with no network trained and no reference detector involved; §3.4 then asks what survives into the rendered tensor, holding a trained encoder fixed, and §3.5 what the input no longer determines and a cell cannot resolve.

### 3.1 The two placements disagree, and most where it matters

Accumulation creates the choice without making the tensor dense. At the detector's $0.2$ m cell size, $83.9\%$ of occupied cells hold exactly one return. The two placements disagree most where it matters: near fast movers ($\geq 8$ m/s), $73.2\%$ of occupied cells contain returns from only one of them, and compensation displaces a return by one cell on average. This is not a subtle perturbation of one tensor; it is a different set of occupied cells.

### 3.2 The two placements carry different information

We measure two quantities over $13{,}007$ moving-vehicle boxes at six accumulation windows, using the same boxes at every window so that the trend cannot be a composition artefact. The velocity code correlates each object's along-motion spread, normalised by its box extent along that direction, with its speed; the localisation gain is the relative increase in in-box returns for fast movers. Appendix G defines both as computed.

Object membership is defined by time rather than by a spatial window: a return belongs to the object whose box contained it when it was measured (Appendix A). Ground-truth velocity labels membership and supplies the motion direction; it constructs neither rendering, and both are scored on the identical point set.

At ten sweeps the raw placement's trail extent correlates with object speed at $+0.764$. Compensation reduces that to $+0.297$, removing $61\%$ of it, while raising the concentration of an object's returns inside its own box by $54.1\%$ for objects moving at $8$ m/s or more. That $61\%$ is a reduction in one measured correlation, not in the velocity information the tensor holds: the per-point Doppler survives unchanged, and what weakens is the trail-borne cue. Both quantities are monotone in accumulation depth on this dataset, both driven by how far the object travels while the window is open. The coupling is empirical, not definitional: §5 reports a dataset where the velocity cost still appears at every depth while the localisation gain does not.

### 3.3 THE EFFECT REQUIRES EACH DISPLACEMENT TO BELONG TO ITS OWN MEASUREMENT

A natural objection is that any redistribution of points would produce this, and that Doppler correspondence is doing no work. We test it with a third placement that gives every return another return's displacement, permuted within strata of time lag and range so the displacement distribution is preserved exactly (Appendix B): a matched control in the spirit of control tasks (Hewitt & Liang, 2019).

The permuted placement, plotted alongside the other two in Figure 2 and tabulated at every window in Table 6, shows no rise in localisation gain with accumulation depth and leaves the velocity code intact: $+0.724$ against the raw placement's $+0.764$, where real compensation leaves $+0.297$. Stratified by object speed at the deepest window, real compensation reaches $+54.1\%$ localisation gain above $8$ m/s while the permutation reaches $+5.8\%$. Moving points is not what produces either effect: with the displacements matched and only their correspondence broken, both effects are claims about that correspondence rather than perturbation. §6 returns to it in a form that trains a detector.

### 3.4 THE DIFFERENCE SURVIVES INTO THE RENDERED TENSOR

Between the geometry measured above and the trained detector of §4 sits the tensor the detector receives. We fit a ridge probe (Alain & Bengio, 2018) from that tensor to object speed, holding the encoder fixed and varying only the placement of its input, so the contrast has the single-factor form of §4.1.

Speed is more linearly extractable from the raw-placed tensor. Across eight frozen encoders, two systems by two encoder trainings by two seeds, the margin runs $+0.019$ to $+0.032$ $R^2$, over all boxes and on movers alike, and every interval from a scene-level cluster bootstrap excludes zero. The sign does not turn on either system when the encoder was itself trained on the compensated placement, which is what separates the placement from the weights, and it holds across encoders as different as a dense pillar scatter and a sparse residual backbone.

Both arms reach $R^2 \approx 0.6$: the margin is what the placement adds on top of the Doppler both renderings retain, not the velocity code alone, and linear extractability is not use. Appendix D gives the eight encoders, a crop-window sweep and the protocol.

### 3.5 WHAT THE INPUT NO LONGER DETERMINES, AND WHAT A CELL CANNOT RESOLVE

The probe measures accessibility. What follows asks two different questions: whether the raw address survives anywhere in the detector's input, and what a single cell constrains about motion.

**The placement itself, where the lag is discarded.** Undoing compensation requires the lag, which the PillarNet-radar student's loader consumes and drops (§2.3). Absence of the direct inverse is weak; the stronger claim, that no function of that input recovers the raw address, rests on a collision. The same target measured at a different lag in the window, under the constant radial motion the compensated placement already assumes, yields an identical detector input and a raw address displaced by $d_i(t_i - t'_i)\hat{u}_i$, non-zero whenever the Doppler is (Appendix G). This is a property of that pipeline, not of compensation: where the lag is kept, it does not apply.

**The tangential component, everywhere.** A return measures only the projection of motion onto the line of sight, so recovering two-dimensional velocity within a cell needs its returns to span directions, and near fast movers they do not. Returns from one object at one range share essentially a single bearing, the direction covariance is rank-one, and an observability gate admits $0.000\%$ of multi-return cells. What compensation removes is not present in the cell as a local constraint a better decoder could use, and the unmeasured component is the larger one, median magnitude $2.4$ times the measured radial error (Appendix C).

## 4 PLACEMENT REACHES THE DETECTOR

§3 showed the two placements carry different information; this section asks whether a detector responds. Table 1 reports every rendering on every nuScenes system, so each system's contrast reads on its own terms rather than against whichever placement its implementation happened to adopt.

Metrics are the nuScenes ones: mAP, mean velocity error mAVE, and NDS, a weighted composite of mAP and five true-positive errors including mAVE (Appendix H). All three are reported $\times 100$ throughout, so a difference of $0.07$ is $0.0007$ in the units the evaluator writes; the remaining true-positive errors are left in their native units.

Within each system all arms share one frozen OpenPCDet configuration (Team, 2020) (Appendix H); we make no absolute state-of-the-art claim. Every difference is formed within a seed and judged against the arms' own spreads (Table 22), and single-seed and post hoc results are labelled as such.

Table 1: Every rendering on every system, nuScenes validation, mean $\pm$ sample standard deviation over three seeds; "both" renders both placements into two channel groups; per-seed values in Table 22. Bold marks the best mean, which need not lead at every seed (§4.2). Blue marks a single placement that beats the other at every seed by more than the larger arm spread.

| system | rendering | mAP ↑ | NDS ↑ | mAVE ↓ |
|---|---|---|---|---|
| CenterPoint (Yin et al., 2021) | raw | $9.67 \pm 0.11$ | $35.21 \pm 0.14$ | $\mathbf{32.04 \pm 0.67}$ |
| | compensated | $9.83 \pm 0.07$ | $35.13 \pm 0.17$ | $33.47 \pm 0.25$ |
| | both | $\mathbf{10.10 \pm 0.16}$ | $\mathbf{35.54 \pm 0.35}$ | $32.33 \pm 0.40$ |
| PillarNet-radar (Shi et al., 2022) student | raw | $4.21 \pm 0.06$ | $29.24 \pm 0.24$ | $39.19 \pm 0.49$ |
| | compensated | $\mathbf{4.49 \pm 0.05}$ | $29.16 \pm 0.07$ | $44.35 \pm 0.15$ |
| | both | $4.46 \pm 0.02$ | $\mathbf{29.73 \pm 0.18}$ | $\mathbf{39.00 \pm 0.25}$ |
| RadarDistill (Bang et al., 2024) | raw | $14.35 \pm 0.11$ | $38.78 \pm 0.32$ | $\mathbf{31.11 \pm 0.63}$ |
| | compensated | $13.98 \pm 0.30$ | $38.30 \pm 0.17$ | $32.61 \pm 1.05$ |
| | both | $\mathbf{14.65 \pm 0.12}$ | $\mathbf{39.16 \pm 0.16}$ | $31.78 \pm 0.50$ |

The single-factor arms are tightly matched: on the student they are parameter-identical, their configurations differing by one line (Appendix H).

### 4.1 The single-factor contrast

The raw and compensated rows of Table 1 are that contrast, one placement each. On the student, where the loader expresses Doppler in the correct frame throughout, choosing the trail rather than the compensated placement improves velocity error by $4.78$ to $5.62$ and costs $0.18$ to $0.39$ mAP, over three seeds. All six differences exceed the larger of the two arms' spreads and run in the direction §3.2 predicts: the trail-preserving placement gives better velocity estimates, the concentrating placement better detections. The experiment identifies the placement as the cause; it does not establish that the along-motion spread and in-box concentration of §3.2 are what mediate the effect. The velocity advantage also grows with object speed, the pattern a placement account predicts, which §6.4 shows is supporting evidence rather than a demonstration of mechanism; Appendix E gives the bins.

The other two systems reproduce the raw placement's velocity advantage only in part. On CenterPoint raw gives lower velocity error at every seed, but two of the three margins sit inside that arm's spread, so the direction there is a sign result rather than a spread-clearing one. On RadarDistill the advantage is not consistent across seeds, raw leading at two of the three. Compensation's detection benefit is absent at every RadarDistill seed and does not replicate on CenterPoint, where two of the three margins sit inside the pair's spread (Appendix K). The systems differ in ways that change how much of a trail a cell resolves (Appendix H); we do not isolate which. The asymmetry is itself the result: the cost agrees at all three levels we can observe, the geometry of §3.2, the tensor of §3.4 and the detector here, while the benefit is measured only at the first and the last and disagrees between them.

### 4.2 KEEPING BOTH PLACEMENTS

The both-placement rows of Table 1 answer a different question: whether a detector given both renderings does better than one given either. On NDS the both-placements arm leads in sixteen of the eighteen same-seed paired differences, both exceptions at third seeds (Table 22). Only its NDS margin over the compensated placement clears the larger arm spread at every seed of a system, and only on the student and on RadarDistill, $+0.40$ to $+1.06$; over the raw placement it clears no system's spread. Keeping both is the better choice against a compensated reference, and not reliably better than a raw one.

It is not additive. On the student the both-placements arm does not improve detection accuracy over the compensated placement at any seed (within $0.08$ mAP), and on velocity error it beats the trail alone at moderate speeds but loses above $10$ m/s. Keeping both avoids committing early to a placement whose benefit is conditional; it is not evidence that a detector exploits the two as complementary information, since §6.2 shows that a large share of the apparent gain survives controls that destroy the placement correspondence entirely.

## 5 WHAT TRANSFERS ACROSS DATASETS, AND WHAT REVERSES

Re-measured on TruckScenes (Fent et al., 2024), a dataset with a different sensor layout and a much longer range, the weakening of the trail-borne cue appears at every depth while the localisation gain is negative at every window (Table 16).

With exact Doppler and constant velocity, compensation cannot increase a return's Euclidean error against its own current position: $-vt$ raw-placed becomes $-(I - \hat{u}\hat{u}^\top)vt$, never longer. A smaller error does not guarantee higher in-box concentration, because a box is narrow across its heading and long along it, so a shortened error can still cross the narrow face. The measured returns show this mismatch between Euclidean error and box membership (Appendix I).

The localisation gain correlates with how radial the motion is: where motion is essentially all radial, $64\%$ of boxes here, compensation consolidates returns as on nuScenes; where it is only partly radial, in-box counts fall. The velocity cost concentrates in that bin too, which qualifies the asymmetry: outside it compensation removes at most $11\%$ of the code and in one bin raises it, while inside it the correlation changes sign, $+0.128$ to $-0.037$, against the $61\%$ removed on nuScenes. That stratification associates rather than explains, and the vertical component of the displacement is ruled out (Appendix I). The depth relationship of §3.2 therefore holds within a measurement geometry, and radar datasets differ widely in it (Palffy et al., 2022; Paek et al., 2022; Zhang et al., 2025; Zheng et al., 2026).

Table 2: Every rendering on both TruckScenes systems, mean ± sample standard deviation over two seeds; official v1.2-trainval split, 12 classes. The distillation system is absent for want of a teacher pretrained on this dataset (Appendix I). Highlighting follows Table 1; spreads in Appendix K.

| system | rendering | mAP ↑ | NDS ↑ | mAVE ↓ |
|---|---|---|---|---|
| CenterPoint | raw | $14.98 \pm 0.25$ | $31.46 \pm 0.53$ | $96.95 \pm 3.36$ |
| | compensated | $13.58 \pm 0.06$ | $26.86 \pm 0.44$ | $98.69 \pm 1.97$ |
| | both | $\mathbf{16.05 \pm 0.42}$ | $\mathbf{32.95 \pm 0.22}$ | $\mathbf{91.05 \pm 3.15}$ |
| PillarNet-radar student | raw | $16.26 \pm 0.13$ | $31.92 \pm 0.25$ | $86.22 \pm 3.71$ |
| | compensated | $16.45 \pm 0.22$ | $29.48 \pm 1.13$ | $90.42 \pm 6.08$ |
| | both | $\mathbf{17.04 \pm 0.88}$ | $\mathbf{33.78 \pm 0.18}$ | $\mathbf{83.27 \pm 1.03}$ |

The same reversal reaches the detector (Table 2). On CenterPoint compensation is worse on every metric at both seeds, on detection at $3.6$ and $4.4$ times that reference's spread. The student agrees on NDS and velocity, though only its first-seed NDS margin clears the arms' own spread, and its two seeds disagree in sign on mAP.

Keeping both placements is the more stable claim: on both systems that arm beats either single placement on NDS at both seeds and above the larger arm spread, so the composite result transfers to a geometry where the single-placement benefit reverses. The mAP and velocity margins do not, and on CenterPoint the mAP gain rests on a single class (Appendix I). What an end-to-end comparison would credit to the placement is a separate question (§6).

# 6 Attributing End-to-End Gains

Returning to nuScenes, this section decomposes the gain of CenterPoint's both-placements arm over the original reference, the uncorrected arm at the top of Table 3. That end-to-end comparison puts a detector rendering both placements against the detector it was built from (Palmer et al., 2023; Haitman & Bialer, 2025), and gives $+2.27$ and $+2.16$ mAP at the two seeds. Its arms differ in more than the placement, so a placement conclusion drawn from it holds only if placement is the factor actually compared. The compensated-against-raw contrast is a different comparison, evaluated in §4.1 with the frame correction applied to both of its arms, and is not decomposed here.

## 6.1 The reference and its frame defect

One of the factors separated below is a defect in the reference arm rather than a property of the intervention. The nuScenes devkit rotates radar positions into the reference frame but not the Doppler vectors accompanying them, and the reference rendering inherits that, so it trains on velocity features that differ from their corrected values for $99.9\%$ of returns (Appendix G); being a rotation, the difference is bounded by twice a return's own speed. We establish this for the pipeline evaluated here and make no claim about how widely it is shared. Positions are byte-identical, so this is purely a change of point features and cannot move a return into a different cell.

## 6.2 Controlled comparisons and placebos

That end-to-end comparison is therefore decomposed into a sequence whose arms differ from their neighbours in exactly one factor: feature correction, then architecture, then changed support, then placement correspondence. Parameter matching does not substitute for it: our width-matched control, with fewer parameters than the reference, keeps almost none of its apparent gain once that reference is corrected (Appendix J).

Two of the rungs of Table 3 are controls, randomising the component the method claims to exploit while matching everything else (Adebayo et al., 2018; Zhang et al., 2016).

The identity duplicate gives the second channel group a byte-identical copy of the first: two groups, two encoders, twice the points, no new information whatsoever. It controls the architecture: over the corrected single placement the duplicate adds $+0.04$ and $+0.18$ mAP, almost nothing.

The matched placebo gives that group the real displacement field with the correspondence destroyed, each displacement reassigned to a different return within strata of time lag and range. It is the construction of §3.3, and it controls the change of support: it reaches new cells to the same degree the real placement does, $+30.7\%$ against $+30.3\%$. Over the corrected single placement it adds $+0.18$ and $+0.28$ mAP, a quarter of the gain the correction leaves at one seed and two thirds at the other.

## 6.3 The decomposition

At both seeds the frame correction dominates the detection gain, $+1.57$ then $+1.74$ mAP, which is $69\%$ then $81\%$ of the end-to-end gain over the uncorrected reference. It accounts for that comparison's velocity change as well, $99\%$ then $121\%$ of it, passing $100\%$ at the second seed because every later step makes velocity worse there. Restoring each displacement to its own return does not replicate on detection: over the placebo it adds $+0.52$ then $+0.14$ mAP, $23\%$ then $6\%$ of that same gain, the second value half the placebo arm's own spread of $0.26$. On the composite it does, $+0.48$ and $+0.58$ NDS, roughly four times the larger spread. These are sequential single-factor increments along one ordering, not independent additive components (Appendices J and L).

Table 3: The ladder, at both seeds as $s_0/s_1$, each arm differing from the one above it in exactly one factor (§6.2). Read at the two seeds every rung has; two rungs carry a third in Table 1. Bold marks the best arm where both seeds agree, which velocity does not.

| arm | factor added | mAP ↑ | NDS ↑ | mAVE ↓ |
|---|---|---|---|---|
| original reference | none | 7.98 / 7.96 | 32.32 / 32.46 | 37.60 / 37.33 |
| corrected, one placement | feature correction | 9.55 / 9.70 | 35.20 / 35.36 | 31.93 / 31.44 |
| identity duplicate | architecture | 9.59 / 9.88 | 35.39 / 35.66 | 32.34 / 31.86 |
| matched placebo | changed support | 9.73 / 9.99 | 35.20 / 35.22 | 31.90 / 31.88 |
| both placements | placement correspondence | **10.25/10.12** | **35.68/35.81** | 31.88 / 32.46 |

Tracking divides the same end-to-end gain over three arms rather than five: the frame correction takes $+0.0105$ and $+0.0153$ AMOTA, and going from the corrected single placement to both placements takes $+0.0109$ and $+0.0049$. Stratified post hoc by condition and by range at both seeds, the placement step keeps no stable sign at night and worsens velocity beyond 20 m, while the frame correction improves every range band (Appendix F).

### 6.4 The speed-stratified signature

Velocity error in the original comparison improved more for faster objects at both seeds, from $-8.3\%$ and $-7.2\%$ in the slowest moving bin to $-18.3\%$ and $-17.3\%$ above 10 m/s, and least on stationary objects: the pattern a placement account predicts. The frame correction reproduces this pattern. Across the moving bins the correction's profile runs from $-9.5\%$ and $-11.6\%$ to $-18.0\%$ and $-17.0\%$, while the placement's own contribution is small and mostly adverse, worse in three of five bins at one seed and four at the other (Appendix E). Why the correction has that profile is not measured. A speed-stratified pattern discriminates between the two accounts only if the competing one does not also produce it, and here it did.

## 7 Discussion and Conclusion

Where a multi-sweep radar detector places a past measurement determines what its tensor makes available. Compensation weakens the trail-borne cue whose extent carries object speed, in the geometry, in what a frozen encoder makes linearly extractable, and in velocity error at every seed on two of three systems. What it buys is less consistent: it turns on the measurement geometry, reverses where much of the motion is not radial, and reaches detection accuracy on one of three. Availability does not imply exploitation, and where the lag is discarded the choice is irreversible as well. End to end, that arm gains far more than the placement is responsible for, most of it a Doppler frame defect in the reference.

An end-to-end comparison leaves joined three things the arms above separate, and the separations are not specific to radar. Matching capacity does not match the input: the factor that dominates here is a point feature in the wrong frame, which a width- and parameter-matched arm carries untouched. Nor does a capacity control serve as a placebo: an identity duplicate and a correspondence-breaking permutation each reproduce most of the gain measured against the uncorrected reference (§6.2), and the control that discriminates randomises the component the method claims to exploit rather than matching its size. And a speed-stratified pattern does not confirm a placement account, because the frame correction produces the same profile (§6.4).

The evidence for the separations is uneven. The decomposition and both placebos exist on CenterPoint alone, so that share is measured on one system and assumed on none. Its increments are sequential contrasts along one ordering that telescope by construction, and factor interactions are not evaluated. The conditional analyses are post hoc strata over matched detections. Across datasets the both-placements result transfers on NDS while the single-factor benefit reverses; these are two results, not one (§5). Nothing here shows object velocity to be unrecoverable: both renderings keep the radial Doppler, so these are results about exploitation, not possibility. They concern accumulated automotive radar; the difficulty they illustrate is not obviously specific to it (Appendix L).

### AI USE STATEMENT

In this work we used generative AI tools for research methodology and experiment design and for feedback on it, for method implementation, for refining hypotheses, for dataset reformatting, for formulating the mathematical claim of Eq. 3 and its accompanying control, and for interpreting results. We have not used generative AI tools for synthetic data generation, for proof writing, or for translation, and the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for software code, for creating and editing figures, for drafting and editing the paper, for literature identification and summarisation, for brainstorming, and for proposing the title and structure.

We have reviewed all AI-assisted work, and the review is mechanical wherever it can be. Every table cell and every four-decimal figure in the text is regenerated from the stored evaluation outputs and checked against them by scripts that fail on any disagreement; the numbers quoted in this paper are not transcribed by hand or by a model. Claims about what the measurements show were checked against the measurements, and where a quoted figure could not be reproduced it was replaced by one that can be. Numerical kernels obtained with AI assistance were validated against reference implementations before use. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

### ETHICS STATEMENT

The work re-analyses two public driving datasets, nuScenes (Caesar et al., 2020) and TruckScenes (Fent et al., 2024), under their licences. It introduces no new data collection and no human subjects. Its subject is a preprocessing choice in detectors intended for road vehicles; where such a detector is deployed, the velocity evidence this paper shows compensation weakens is the evidence a downstream tracker uses, so the choice has a safety dimension that a detection metric alone does not expose.

### REPRODUCIBILITY STATEMENT

Every arm within a system shares one frozen configuration differing only in the factor its comparison names; Appendix H gives the configurations, optimiser, schedule, seeds and evaluation protocol, and Appendix G gives the placement operators and the per-point features as implemented. The geometric measurements of §3.2, the tensor probe of §3.4 and the controls of §5 are defined in Appendices G, D and I in terms sufficient to reimplement them from the datasets alone. The scripts named in those appendices, and the evaluation outputs every table is generated from, are released with the paper.

## REFERENCES

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. 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/294a8ed24b1ad22ec2e7efea049b8737-Paper.pdf.

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes, 2018. URL https://arxiv.org/abs/1610.01644.

Geonho Bang, Kwangjin Choi, Jisong Kim, Dongsuk Kum, and Jun Won Choi. Radardistill: Boosting radar-based object detection performance via knowledge distillation from lidar features. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15491–15500, 2024. doi: 10.1109/CVPR52733.2024.01467.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Nazanin Mohammadi Sepahvand, Edward Raff, Kanika Madan, Vikram Voleti, et al. Accounting for variance in machine learning benchmarks. *Proceedings of machine learning and systems*, 3:747–769, 2021.

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.

Xiaomeng Chu, Jiajun Deng, Guoliang You, Yifan Duan, Houqiang Li, and Yanyong Zhang. Racformer: Towards high-quality 3d object detection via query-based radar-camera fusion. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17081–17091, 2025. doi: 10.1109/CVPR52734.2025.01592.

Fangqiang Ding, Xiangyu Wen, Yunzhou Zhu, Yiming Li, and Chris Xiaoxuan Lu. Radarocc: Robust 3d occupancy prediction with 4d imaging radar. *Advances in Neural Information Processing Systems*, 37:101589–101617, 2024.

Felix Fent, Fabian Kuttenreich, Florian Ruch, Farija Rizwin, Stefan Juergens, Lorenz Lechermann, Christian Nissler, Andrea Perl, Ulrich Voll, Min Yan, and Markus Lienkamp. Man truckscenes: A multimodal dataset for autonomous trucking in diverse conditions. 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. 62062–62082. Curran Associates, Inc., 2024. doi: 10.52202/079017-1982. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/71ac06f0f8450e7d49063c7bfb3257c2-Paper-Datasets_and_Benchmarks_Track.pdf.

Yuval Haitman and Oded Bialer. Doppdrive: Doppler-driven temporal aggregation for improved radar object detection. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26085–26094, 2025. doi: 10.1109/ICCV51701.2025.02420.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1275. URL https://aclanthology.org/D19-1275/.

Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12689–12697, 2019. doi: 10.1109/CVPR.2019.01298.

Zhiwei Lin, Zhe Liu, Yongtao Wang, Le Zhang, and Ce Zhu. Rcbevdet++: Toward high-accuracy radar-camera fusion 3d perception network, 2024a. URL https://arxiv.org/abs/2409.04979.

Zhiwei Lin, Zhe Liu, Zhongyu Xia, Xinhao Wang, Yongtao Wang, Shengxiang Qi, Yang Dong, Nan Dong, Le Zhang, and Ce Zhu. Rcbevdet: Radar-camera fusion in bird's eye view for 3d object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14928–14937, 2024b. doi: 10.1109/CVPR52733.2024.01414.

Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L. Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2774–2781, 2023. doi: 10.1109/ICRA48891.2023.10160968.

Yunfei Long, Daniel Morris, Xiaoming Liu, Marcos Castro, Punarjay Chakravarty, and Praveen Narayanan. Full-velocity radar returns by radar-camera fusion. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16178–16187, 2021. doi: 10.1109/ICCV48922.2021.01589.

Dong-Hee Paek, Seung-Hyun Kong, and Kevin Tirta Wijaya. K-radar: 4d radar object detection for autonomous driving in various weather conditions. *Advances in Neural Information Processing Systems*, 35:3819–3829, 2022.

Andras Palffy, Ewoud Pool, Srimannarayana Baratam, Julian F. P. Kooij, and Dariu M. Gavrila. Multi-class road user detection with 3+1d radar in the view-of-delft dataset. *IEEE Robotics and Automation Letters*, 7(2):4961–4968, 2022. doi: 10.1109/LRA.2022.3147324.

Patrick Palmer, Martin Krueger, Richard Altendorfer, and Torsten Bertram. Ego-motion estimation and dynamic motion separation from 3d point clouds for accumulating data and improving 3d object detection. In *AmE 2023–Automotive meets Electronics; 14. GMM Symposium*, pp. 86–91. VDE, 2023.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. Information-theoretic probing for linguistic structure. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4609–4622, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.420. URL https://aclanthology.org/2020.acl-main.420/.

Guangsheng Shi, Ruifeng Li, and Chao Ma. Pillarnet: Real-time and high-performance pillar-based 3d object detection. In *European conference on computer vision*, pp. 35–52. Springer, 2022.

OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020.

Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 183–196, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.14. URL https://aclanthology.org/2020.emnlp-main.14/.

Philipp Wolters, Johannes Gilg, Torben Teepe, Fabian Herzog, Felix Fent, and Gerhard Rigoll. Sparc: Sparse radar-camera fusion for 3d object detection, 2025. URL https://arxiv.org/abs/2411.19860.

Bingxue Xu, Emil Hedemalm, Ajinkya Khoche, and Patric Jensfelt. Radar-informed 3d multi-object tracking under adverse conditions, 2026. URL https://arxiv.org/abs/2604.13571.

Shanliang Yao, Runwei Guan, Zitian Peng, Chenhang Xu, Yilu Shi, Weiping Ding, Eng Gee Lim, Yong Yue, Hyungjoon Seo, Ka Lok Man, Jieming Ma, Xiaohui Zhu, and Yutao Yue. Exploring radar data representations in autonomous driving: A comprehensive review. *IEEE Transactions on Intelligent Transportation Systems*, 26(6):7401–7425, 2025. doi: 10.1109/TITS.2025.3554781.

Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11784–11793, June 2021.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Xinyu Zhang, Li Wang, Jian Chen, Cheng Fang, Guangqi Yang, Yichen Wang, Lei Yang, Ziying Song, Lin Liu, Xiaofei Zhang, et al. Dual radar: A multi-modal dataset with dual 4d radar for autononous driving. *Scientific data*, 12(1):439, 2025.

Lianqing Zheng, Long Yang, Qunshu Lin, Wenjin Ai, Minghao Liu, Shouyi Lu, Jianan Liu, Hongze Ren, Jingyue Mo, Xiaokai Bai, et al. Omnihd-scenes: A next-generation multimodal dataset for autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

## A Membership definition and the margin sweep

§3.2 defines object membership by time: a return belongs to an object if and only if it lay inside that object's box at the moment it was measured, i.e. inside the box translated back by $v_o^{\mathrm{gt}} t$. The alternative, a fixed spatial margin around the box, is not a neutral choice.

A spatial margin fails in both directions. Too small, and it clips the trail: an object moving at $10\,\mathrm{m/s}$ travels $7.7\,\mathrm{m}$ during a $0.77\,\mathrm{s}$ accumulation window, far outside a $4\,\mathrm{m}$ window, so the measured trail extent saturates for exactly the fast objects that carry the velocity code. Too large, and it admits returns from neighbouring objects and static structure, which are present under both placements and

dilute the difference between them. Table 4 sweeps the margin on the population and window of Table 5, with only the membership rule changing.

Time-based membership has neither failure mode: it cannot clip a trail however long, and it admits a return only where the box, translated back by the object's ground-truth velocity, contained it. That is membership under a ground-truth constant-translation approximation rather than membership observed at each measurement time, and acceleration, rotation, annotation error and overlapping boxes each violate it. The cost is using ground-truth box velocity to label which returns belong to which object. That use is confined to labelling: it does not enter either rendering, and both renderings are scored on the identical labelled point set.

No margin recovers the time-based measurement. The raw velocity code never exceeds $+0.239$ against $+0.764$, the raw-minus-compensated difference peaks at $0.151$ against $0.467$ and collapses to zero by $8$ m, and the localisation gain reaches at most $+43.7\%$ against $+54.1\%$. All numbers in the main text use time-based membership.

Table 4: The geometric measurement under each membership rule: 13,007 moving-vehicle boxes, $0.77$ s window, both placements scored on the identical member set. A spatial margin tests the raw position against the reference-time box grown by the margin on each side.

| membership | velocity code (raw) | velocity code (comp.) | raw − comp. | localisation gain |
|---|---|---|---|---|
| time-based | $+0.764$ | $+0.297$ | $0.467$ | $+54.1\%$ |
| margin 1 m | $+0.175$ | $+0.039$ | $0.137$ | $-0.7\%$ |
| margin 2 m | $+0.239$ | $+0.088$ | $0.151$ | $+13.6\%$ |
| margin 4 m | $+0.229$ | $+0.135$ | $0.094$ | $+32.7\%$ |
| margin 8 m | $+0.176$ | $+0.162$ | $0.014$ | $+43.5\%$ |
| margin 15 m | $+0.156$ | $+0.161$ | $-0.005$ | $+43.7\%$ |

## B THE DEPTH LAW AND THE PERMUTATION CONTROL IN FULL

Figure 2 plots these numbers. They are given here at every window, over the same 13,007 moving-vehicle boxes throughout.

Two populations are in play and must not be read against each other. Table 5's localisation column and Figure 2's centre and right panels measure fast movers, reaching $+54.1\%$ at the deepest window against the permutation's $+5.8\%$. Table 6's localisation columns measure all moving boxes, which is why its real column reaches $+15.1\%$ rather than $+54.1\%$. The velocity code is the same measurement in both tables.

Table 5: Both columns are monotone in accumulation depth and both are approximately zero at a single sweep. The localisation gain is for fast movers ($\geq 8$ m/s).

| window (s) | ≈sweeps | pts/box | vel. code raw | vel. code comp. | loc. gain |
|---|---|---|---|---|---|
| 0.077 | 1 | 7.2 | $-0.025$ | $-0.072$ | $+0.9\%$ |
| 0.154 | 2 | 11.6 | $+0.078$ | $-0.036$ | $+5.8\%$ |
| 0.231 | 3 | 16.0 | $+0.222$ | $+0.022$ | $+11.3\%$ |
| 0.385 | 5 | 24.6 | $+0.504$ | $+0.149$ | $+21.4\%$ |
| 0.539 | 7 | 33.0 | $+0.678$ | $+0.233$ | $+36.0\%$ |
| 0.770 | 10 | 42.0 | $+0.764$ | $+0.297$ | $+54.1\%$ |

The localisation gain is a pooled ratio, and its denominator falls with speed: in-box raw returns per box run 37.6, 39.1, 31.0 and 16.5 across the four bins of Table 7, which is the trail doing what §3.2 describes. The gain for fast movers is therefore measured on the smallest base, and should be read with that in mind.

Table 6: The displacement-matched permutation reproduces neither effect. Localisation over all moving boxes.

| window (s) | loc. real | loc. permuted | vel. raw | vel. comp. | vel. permuted |
|---|---|---|---|---|---|
| 0.077 | $+0.4\%$ | $-0.3\%$ | $-0.025$ | $-0.072$ | $-0.029$ |
| 0.231 | $+4.3\%$ | $-0.5\%$ | $+0.222$ | $+0.022$ | $+0.226$ |
| 0.385 | $+7.6\%$ | $-0.9\%$ | $+0.504$ | $+0.149$ | $+0.505$ |
| 0.539 | $+11.1\%$ | $-1.1\%$ | $+0.678$ | $+0.233$ | $+0.650$ |
| 0.770 | $+15.1\%$ | $-0.9\%$ | $+0.764$ | $+0.297$ | $+0.724$ |

Table 7: Localisation gain by object speed, at the deepest window, over the same paired population. The velocity code is not stratified here because it is a correlation taken over speed. Objects below $0.5$ m/s are excluded; the four bins sum to the $13{,}007$ boxes of Table 5.

| speed (m/s) | boxes | in-box returns, raw | loc. real | loc. permuted |
|---|---|---|---|---|
| 0.5–2 | 1,907 | 71,645 | $-0.7\%$ | $-3.6\%$ |
| 2–5 | 3,170 | 123,879 | $+2.1\%$ | $-3.5\%$ |
| 5–8 | 3,827 | 118,713 | $+16.1\%$ | $-0.5\%$ |
| $\geq 8$ | 4,103 | 67,680 | $+54.1\%$ | $+5.8\%$ |

The displacement distribution is preserved exactly by construction: real and permuted fields both have median $0.025$ m, mean $0.209$ m and 99th percentile $4.094$ m. A non-zero permuted localisation value is expected and is not a failure, because moving points changes concentration mechanically; what the control tests is whether the value rises with depth and concentrates on fast movers, and it does neither.

## C Local observability in full

§3.5 states the result; this is the measurement behind it.

A Doppler return measures only the projection of motion onto the line of sight. In this pipeline that is exact rather than approximate: the reported Doppler vector is parallel to the sensor-to-return ray for all $4{,}082$ returns checked. Recovering full two-dimensional velocity within a cell requires the returns in that cell to span directions, since the least-squares decode $v = (\sum \hat{u}\hat{u}^\top)^{-1} \sum d\hat{u}$ needs $\sum \hat{u}\hat{u}^\top$ to be well conditioned.

Measured on $420{,}869$ cells near fast movers, restricted to the $12.1\%$ containing at least two returns because single-return cells are rank-one by construction, the direction covariance has median anisotropy $5.9 \times 10^{-7}$ against $0.5$ for an isotropic cell, and returns within a cell span a median of $0.092°$ of bearing. A gate admitting the decode only where full velocity is observable, at either $0.15$ or $0.05$, admits $0.000\%$ of cells.

The reason is geometric rather than statistical: returns from one object at one range share essentially a single bearing. The component compensation removes is not present in the cell in a form a better local decoder could reach. That component is also the larger one: Doppler recovers the radial part of vehicle motion with a median error of $0.373$ m/s, while the tangential part it does not measure at all has median magnitude $0.890$ m/s, roughly $2.4$ times the error in the part that is measured.

## D The tensor probe

§3.4 asks what the rendered tensor carries. The tensor is the VFE output, one feature vector per occupied pillar at the cell the placement sent its returns to, which is $R_\varphi[\kappa, :]$ of Eq. 2; it is 32-d on the student and $64$-d on CenterPoint, the widths those systems deploy. Around each ground-truth box we crop a square, ego-aligned window of the bird's-eye plane, pool it into an $8 \times 8$ grid of

mean features, and fit ridge regression from that to the box's ground-truth speed. The crop is square and ego-aligned rather than rotated onto the object's motion: a motion-aligned window would hand the probe the direction along which the trail extends, which is the label dependence Appendix G discloses for the geometric measurement. Only the box centre is used, not its velocity.

**The encoder is held fixed.** Each row pair of Table 8 runs one frozen set of weights over both placements, so the arms differ in where returns were sent and in nothing else. Comparing each placement under its own separately trained encoder would confound the two: the pillar layers of the two trained students differ by up to $5.4$ in absolute weight.

Table 8: Linear probe from the rendered tensor to object speed, 16 m crop. Each row pair is one frozen encoder run over both placements. Intervals are $2{,}000$ scene-level cluster bootstrap resamples. The probe reads the VFE output, before either backbone: 32 channels on the student, $64$ on CenterPoint.

| system | frozen encoder | subset | $R^2$ raw | $R^2$ comp. | $\Delta$ (CI) |
|---|---|---|---|---|---|
| student | trained on raw, $s_0$ | all boxes | 0.6156 | 0.5865 | $+0.0291$ $[+0.0211, +0.0375]$ |
| | | movers | 0.5470 | 0.5150 | $+0.0319$ $[+0.0167, +0.0457]$ |
| | trained on raw, $s_1$ | all boxes | 0.6179 | 0.5992 | $+0.0187$ $[+0.0118, +0.0256]$ |
| | | movers | 0.5480 | 0.5198 | $+0.0282$ $[+0.0140, +0.0410]$ |
| | trained on comp., $s_0$ | all boxes | 0.6183 | 0.5923 | $+0.0260$ $[+0.0179, +0.0340]$ |
| | | movers | 0.5469 | 0.5175 | $+0.0294$ $[+0.0144, +0.0428]$ |
| | trained on comp., $s_1$ | all boxes | 0.6173 | 0.5961 | $+0.0212$ $[+0.0132, +0.0292]$ |
| | | movers | 0.5505 | 0.5248 | $+0.0257$ $[+0.0135, +0.0368]$ |
| CenterPoint | trained on raw, $s_0$ | all boxes | 0.5816 | 0.5495 | $+0.0321$ $[+0.0175, +0.0470]$ |
| | | movers | 0.5767 | 0.5478 | $+0.0289$ $[+0.0156, +0.0419]$ |
| | trained on raw, $s_1$ | all boxes | 0.5798 | 0.5550 | $+0.0248$ $[+0.0115, +0.0372]$ |
| | | movers | 0.5683 | 0.5484 | $+0.0199$ $[+0.0048, +0.0340]$ |
| | trained on comp., $s_0$ | all boxes | 0.5789 | 0.5533 | $+0.0256$ $[+0.0105, +0.0395]$ |
| | | movers | 0.5735 | 0.5503 | $+0.0232$ $[+0.0067, +0.0385]$ |
| | trained on comp., $s_1$ | all boxes | 0.5809 | 0.5559 | $+0.0250$ $[+0.0111, +0.0385]$ |
| | | movers | 0.5718 | 0.5495 | $+0.0223$ $[+0.0088, +0.0349]$ |

**Grouping.** nuScenes validation is $150$ scenes of about $40$ keyframes, and boxes within a scene are the same objects seen again, so scenes rather than boxes are the unit throughout: $105$ scenes train and $45$ test, the inner split that selects the ridge penalty holds out training scenes, and the bootstrap resamples scenes. Treating boxes as independent narrows the interval by roughly a fifth. The penalty is chosen on held-out training scenes only, and the feature scaler is fitted on training scenes only, so no test statistic enters either.

**The crop window.** Appendix A shows a fixed spatial margin is not neutral for the geometric measurement, where the raw-minus-compensated difference falls from $0.151$ at 2 m to $-0.005$ at 15 m. The probe is far less sensitive: on the first encoder the margin runs $+0.0341$, $+0.0291$ and $+0.0219$ $R^2$ at 12, 16 and 24 m over all boxes, and $+0.0414$, $+0.0319$ and $+0.0320$ on movers, every interval excluding zero. A fitted readout can down-weight the cells a wider window admits; a pooled count cannot.

**What it does not show.** Both placements support $R^2$ near $0.6$ on the student and near $0.56$ on CenterPoint, because both retain the per-point Doppler. The probe therefore bounds what the placement adds over that, and says nothing about whether the detector uses it. Both systems are nuScenes. Scripts: `code/probes/tensor_probe_{extract,fit}.py`.

## E VELOCITY ERROR STRATIFIED BY OBJECT SPEED

Two speed stratifications are referenced in the body. Both use matched detections and post hoc bins at both seeds, and both select the evaluation population; neither is a fitted trend or a test. Bins hold from about $2{,}500$ matched detections above $10$ m/s to $30{,}000$–$40{,}000$ on stationary objects.

In Table 10 the original column is the end-to-end comparison's profile, reference to both; correction is reference to corrected, and placement is corrected to both. The speed pattern that appeared to confirm a placement mechanism sits in the correction column at both seeds, while the placement column is small and adverse in three of the five bins at one seed and four at the other. Table 9 repeats the stratification on the student, whose arms differ only in placement.

Table 9: Velocity error by ground-truth speed on the PillarNet-radar student (§4.1), as $s_0/s_1$. Raw vs comp. is the single-factor contrast; both vs raw compares keeping both placements with the trail alone. The absolute improvement of raw over compensated rises monotonically with speed at both seeds, from $0.09$ to $28.42$ and from $0.38$ to $17.23$.

| GT speed | comp. | raw | both | raw vs comp. | both vs raw |
|---|---|---|---|---|---|
| 0.0–0.5 | 9.07/9.28 | 8.98/8.90 | 8.94/9.14 | $-1.0\%/-4.1\%$ | $-0.4\%/+2.7\%$ |
| 0.5–2.0 | 69.44/69.91 | 68.48/68.03 | 67.00/65.98 | $-1.4\%/-2.7\%$ | $-2.2\%/-3.0\%$ |
| 2.0–5.0 | 94.19/92.44 | 87.73/87.42 | 82.43/84.44 | $-6.9\%/-5.4\%$ | $-6.0\%/-3.4\%$ |
| 5.0–10.0 | 125.00/121.34 | 110.94/113.14 | 110.35/109.76 | $-11.2\%/-6.8\%$ | $-0.5\%/-3.0\%$ |
| 10.0+ | 185.55/174.41 | 157.13/157.19 | 162.02/165.80 | $-15.3\%/-9.9\%$ | $+3.1\%/+5.5\%$ |

Table 10: The same stratification on CenterPoint with the corrected arm inserted (§6.4), at both seeds.

| seed | GT speed (m/s) | reference | corrected | both | original | correction | placement |
|---|---|---|---|---|---|---|---|
| $s_0$ | 0.0–0.5 | 9.28 | 9.38 | 9.37 | $+1.0\%$ | $+1.0\%$ | $0.0\%$ |
| | 0.5–2.0 | 69.82 | 63.19 | 64.00 | $-8.3\%$ | $-9.5\%$ | $+1.3\%$ |
| | 2.0–5.0 | 95.12 | 83.96 | 85.18 | $-10.4\%$ | $-11.7\%$ | $+1.5\%$ |
| | 5.0–10.0 | 117.62 | 101.78 | 104.79 | $-10.9\%$ | $-13.5\%$ | $+3.0\%$ |
| | 10.0+ | 179.12 | 146.90 | 146.34 | $-18.3\%$ | $-18.0\%$ | $-0.4\%$ |
| $s_1$ | 0.0–0.5 | 9.40 | 9.28 | 9.29 | $-1.1\%$ | $-1.3\%$ | $+0.1\%$ |
| | 0.5–2.0 | 70.47 | 62.30 | 65.37 | $-7.2\%$ | $-11.6\%$ | $+4.9\%$ |
| | 2.0–5.0 | 97.28 | 85.45 | 86.04 | $-11.6\%$ | $-12.2\%$ | $+0.7\%$ |
| | 5.0–10.0 | 115.29 | 101.25 | 102.52 | $-11.1\%$ | $-12.2\%$ | $+1.3\%$ |
| | 10.0+ | 181.42 | 150.58 | 150.02 | $-17.3\%$ | $-17.0\%$ | $-0.4\%$ |

## F STRATIFIED AND DOWNSTREAM RESULTS IN FULL

§6.3 reports the range, condition and tracking analyses in prose; the underlying tables are given here. The speed stratifications are in Appendix E.

Table 11 is the condition stratification. On the $602$ night samples neither step keeps its sign across seeds: the placement step is $-0.10$ then $+0.81$ mAP, the frame correction $+0.58$ then $-0.21$. The correction's largest gain is in rain at both seeds, $+2.05$ and $+2.46$ mAP.

In Table 12, ground truth and predictions are both restricted to the band, so each band is a self-contained detection problem. At both seeds the frame correction improves mAP and mAVE in every band. The placement step's mAP change is non-negative in every band but unstable in size, and beyond $20$ m it worsens mAVE at both seeds.

Tracking runs from identical detection dumps through CenterPoint's greedy tracker (Yin et al., 2021), which has no learned parameters and uses published per-class gates. Table 13 decomposes

Table 11: Condition stratification, three arms, both seeds. Cells are mAP / NDS / mAVE. Strata are post hoc.

| seed | condition | $n$ | reference | corrected | both |
|---|---|---|---|---|---|
| $s_0$ | all | 6,019 | 7.98 / 32.32 / 37.61 | 9.55 / 35.20 / 31.93 | 10.25 / 35.68 / 31.88 |
| | night | 602 | 6.96 / 20.95 / 69.64 | 7.54 / 23.29 / 63.05 | 7.44 / 22.87 / 65.73 |
| | rain | 1,088 | 8.12 / 29.56 / 44.65 | 10.17 / 35.02 / 28.62 | 10.38 / 35.24 / 28.59 |
| $s_1$ | all | 6,019 | 7.96 / 32.46 / 37.33 | 9.70 / 35.36 / 31.44 | 10.12 / 35.81 / 32.46 |
| | night | 602 | 7.43 / 21.31 / 68.39 | 7.23 / 20.91 / 65.65 | 8.04 / 23.79 / 65.71 |
| | rain | 1,088 | 8.02 / 29.65 / 43.57 | 10.48 / 35.21 / 25.59 | 10.40 / 35.60 / 27.07 |

the original contrast. The frame correction takes $+0.0105$ and $+0.0153$ AMOTA and the placement $+0.0109$ and $+0.0049$, so the division is even at one seed and three to one at the other. The identity-switch reduction belongs to the correction, which accounts for $104\%$ and $98\%$ of it; the placement step moves switches by $+44$ and $-19$. Table 14 repeats the contrast with velocity withheld from association.

Table 12: Range-band evaluation, three arms, both seeds. The first three columns are mAP; the $\Delta$ columns split the change into the frame correction (reference to corrected) and the placement (corrected to both).

| | | | | | $\Delta$ mAP | | $\Delta$ mAVE | |
|---|---|---|---|---|---|---|---|---|
| seed | band | reference | corrected | both | correction | placement | correction | placement |
| $s_0$ | 0–20 m | 11.53 | 13.67 | 14.33 | $+2.14$ | $+0.66$ | $-4.39$ | $-1.33$ |
| | 20–35 m | 7.55 | 8.89 | 9.63 | $+1.35$ | $+0.74$ | $-6.32$ | $+0.71$ |
| | 35–55 m | 3.60 | 3.98 | 4.42 | $+0.38$ | $+0.44$ | $-8.38$ | $+1.27$ |
| $s_1$ | 0–20 m | 11.74 | 13.42 | 15.06 | $+1.68$ | $+1.64$ | $-3.97$ | $+0.16$ |
| | 20–35 m | 7.45 | 9.18 | 9.18 | $+1.73$ | 0.00 | $-6.92$ | $+1.54$ |
| | 35–55 m | 3.52 | 4.07 | 4.20 | $+0.55$ | $+0.13$ | $-6.25$ | $+1.37$ |

Table 13: The tracking contrast decomposed, both seeds, score threshold $0.2$.

| seed | arm | AMOTA ↑ | MOTA ↑ | recall ↑ | IDS ↓ | FP ↓ | TP ↑ | FA/fr. ↓ |
|---|---|---|---|---|---|---|---|---|
| $s_0$ | reference | 0.0938 | 0.0900 | 0.2047 | 1,759 | 44,181 | 34,858 | 127.9 |
| | corrected | 0.1043 | 0.1134 | 0.2098 | 654 | 10,261 | 33,035 | 40.8 |
| | both | 0.1152 | 0.1234 | 0.2258 | 698 | 10,796 | 34,432 | 42.7 |
| $s_1$ | reference | 0.0907 | 0.0954 | 0.2065 | 1,769 | 47,968 | 34,149 | 138.2 |
| | corrected | 0.1060 | 0.1103 | 0.2062 | 692 | 10,963 | 34,433 | 40.4 |
| | both | 0.1109 | 0.1123 | 0.2161 | 673 | 10,260 | 34,437 | 40.4 |

## G THE RENDERING, SPECIFIED

§2.2 states the two placements abstractly. This appendix gives the exact operations, so that what is and is not discarded can be checked rather than inferred.

**Accumulation and the reference frame.** Ten sweeps from each of the five nuScenes radars are accumulated into the LIDAR_TOP frame at the sample timestamp. Positions are transformed by the composed sensor-to-ego, ego-to-global and global-to-reference transforms. The nuScenes devkit rotates positions but not Doppler vectors; we rotate the ego-motion-compensated Doppler vector by

Table 14: Velocity zeroed before tracking, so association proceeds on position alone; threshold $0.2$. Without velocity the both-placements arm has more identity switches than the original reference at both seeds ($+7.8\%$ and $+7.2\%$), while $79\%$ and $66\%$ of its AMOTA gap remain.

| | | AMOTA | | ID switches | |
|---|---|---|---|---|---|
| seed | arm | velocity used | zeroed | velocity used | zeroed |
| $s_0$ | reference | 0.0938 | 0.0705 | 1,759 | 5,632 |
| | corrected | 0.1043 | 0.0777 | 654 | 5,867 |
| | both | 0.1152 | 0.0874 | 698 | 6,073 |
| $s_1$ | reference | 0.0907 | 0.0700 | 1,769 | 5,706 |
| | corrected | 0.1060 | 0.0800 | 692 | 5,761 |
| | both | 0.1109 | 0.0833 | 673 | 6,115 |

the same rotation, which the reference detector's original rendering does not do (§6.1). The two differ for $99.9\%$ of returns, by a median of $0.12$ m/s and $13.7$ at the ninety-ninth percentile.

**The compensated placement.** nuScenes reports Doppler as an ego-motion-compensated vector $(v_x, v_y)$ per return. That vector is parallel to the sensor-to-return ray for every return checked (Appendix C), so it is exactly $d_i\hat{u}_i$ with $d_i$ the radial Doppler. The compensated placement adds $d_i t_i \hat{u}_i$ to the measured position, with $t_i = t_{\text{ref}} - t_{\text{sweep}} \geq 0$; verified against the generated renderings, the applied displacement equals $d_i t_i \hat{u}_i$ to float32 precision. A return measured in the past moves along its own ray, outward where its Doppler is positive and inward where it is negative, and a return with zero Doppler or zero lag does not move at all.

**Per-point features.** $f_i = (x, y, z, \sigma, v_x, v_y, t_i)$: position, radar cross-section, the Doppler vector and the lag. The position features are the coordinates after that arm's placement; the pre-placement coordinates are not retained as additional features. Doppler and lag are retained per point under both placements, so in this bed the second address is computable from the per-point input before pooling, and the non-invertibility of §3.5 does not apply here.

**Scatter.** The plane is discretised into $0.2 \times 0.2$ m pillars over $[-51.2, 51.2]^2$ m, a $512 \times 512$ grid, at most $32$ returns per pillar and at most $30{,}000$ occupied pillars. Pooled over all $6{,}019$ validation samples, an occupied cell holds $1.266$ returns on average and $83.9\%$ hold exactly one. Each return is described by $f_i$ concatenated with its offset from its pillar's mean point and its offset from the pillar centre, and passed through two layers. The first is a linear map with batch normalisation and a ReLU whose per-return output is concatenated with its element-wise maximum over the pillar; the second repeats the map, and the pillar's $64$-d vector is the element-wise maximum of its output. That vector is written to the pillar's cell, so two returns a placement sends to the same cell cannot afterwards be told apart. We do not use this as an irreversibility argument: the raw placement pooled the same way is equally many-to-one, so what it would establish is a property of pooling, not of the placement choice.

**The two-group rendering.** The channel partition of §2.2 is $(|\mathcal{G}_1|, |\mathcal{G}_2|) = (48, 16)$ at $C = 64$ on CenterPoint and $(24, 8)$ at $C = 32$ on the student, $\mathcal{G}_1$ carrying the raw placement in both. Each group has its own pillar encoder, followed by a bias-free linear projection to the group's width. The total width is the reference's in both cases, so the arm renders two placements into the tensor the single-placement arm already had.

**What the representation-level quantities are.** §3.2 names two measurements; this is how they are computed, over moving vehicles whose speed is at least $0.5$ m/s and which have at least three member returns at every window, so that the population is identical in every row. A return belongs to object $o$ iff it lay inside $o$'s box at the moment it was measured, the box translated back by $v_o^{\text{gt}} t_i$, with a $0.5$ m vertical tolerance because radar elevation is poorly resolved. Let $\hat{w}_o$ be the unit heading of $o$'s ground-truth velocity, $c_o$ its box centre and $L_o$ the box's extent along $\hat{w}_o$. The velocity code is then the correlation, taken over objects, between speed and the along-motion spread of the object's own returns,

$$\operatorname{corr}_o\Big( \, |v_o^{\text{gt}}|, \ \operatorname{sd}_{i \in o}\big[(\varphi(m_i) - c_o) \cdot \hat{w}_o\big] / L_o \, \Big),$$

a standard deviation along the direction of travel, normalised by how long the object is in that direction, not an overhang beyond the box. Both quantities use labels: $\hat{w}_o$ and $c_o$ come from the annotation, and membership is decided by the time-propagated box. They therefore measure what a rendering makes present given the object's motion direction, which is more than a detector is given. We use them only comparatively, which makes the comparison paired on an identical point set and conditional on this membership construction and its constant-translation approximation: the same construction, the same $\hat{w}_o$ and the same membership rule are applied to every placement. That makes the dependence common to all arms without eliminating it. The sweep in Appendix A moves the raw-minus-compensated difference from $0.151$ to $-0.005$, so the measured gap is itself a function of the membership rule, and a large raw velocity code is not a result on its own. §3.4 is the reading that does not take the motion direction as given. The localisation gain is a ratio of pooled in-box counts over the fast-mover population, $|v_o^{\text{gt}}| \geq 8$ m/s,

$$\Big(\textstyle\sum_o |\{i \in o : \varphi_{\text{comp}}(m_i) \in \text{box}_o\}| \;/\; \sum_o |\{i \in o : \varphi_{\text{raw}}(m_i) \in \text{box}_o\}| - 1\Big) \times 100\%,$$

so it is a relative increase in returns landing inside their own box, pooled, rather than a mean over per-object fractions. Ground-truth velocity labels membership and supplies $\hat{w}_o$; it never constructs either rendering, and both are scored on the identical point set. The permutation control replaces each return's displacement with another return's, drawn within strata of time lag ($0.02$ s) and range ($10$ m) so the displacement distribution is preserved conditional on both; the permutation is seeded, so the control is reproducible.

**Anisotropy.** §3.5 reports the direction covariance $\sum_i \hat{u}_i \hat{u}_i^\top$ of a cell's returns as rank-one to within a median anisotropy of $5.9 \times 10^{-7}$". The statistic is $\lambda_{\min}/(\lambda_{\min} + \lambda_{\max})$ of that matrix: $0.5$ for an isotropic cell, $0$ for an exactly rank-one one.

**How the two evaluated systems differ.** The specification above is CenterPoint's. The PillarNet-radar student accumulates six sweeps per radar rather than ten, renders at $0.075$ m rather than $0.2$ m, does not cap returns per cell, and differs additionally in pillar encoder, point-cloud range, batch size, schedule length, class-balanced resampling (Appendix H) and the setting of the devkit's radar point filter, which the two loaders set oppositely. Any of these could change how much of a trail a single cell resolves, which is why §4.1 does not attribute the difference in the size of the velocity effect to architecture.

**The lag, and the audit behind §3.5.** The two beds also differ in whether the sweep lag survives into the detector, which is what makes the placement invertible in one and not the other. CenterPoint's point features are $(x, y, z, \sigma, v_x, v_y, t)$. The student's loader computes the lag, applies the displacement with it, and never appends it: its features are $(x, y, z, \sigma, v_x, v_y)$, six wide. A claim of non-invertibility is only as good as the paths it rules out, so the others are closed here. The batch its network receives is $(N, 7)$: a batch index and those six features, with no timing column. Sweeps are concatenated in a fixed order, but point order is shuffled at training and test, so order carries no sweep identity. Voxel coordinates are derived from $x, y$ alone. The augmentation matrix is a geometric transform. The frame identifier and sample metadata travel in the batch dictionary but are written rather than read, and only under a test-time-augmentation branch this arm does not enable. The student's both-placements arms carry one further feature, a group flag marking which of the two renderings a return came from; it identifies the channel group, not the sweep, and does not reintroduce the lag.

**The construction.** The audit shows the lag is not transmitted. It does not by itself show that no function of the input recovers the raw address, since the unordered point set might in principle encode sweep identity implicitly. That gap is closed by a collision. Write the student's retained features for one return as

$$\pi(m_i) \;=\; \big(\; \underbrace{p_i + d_i t_i \hat{u}_i}_{\text{compensated position}} \;,\; z_i,\; \sigma_i,\; \underbrace{d_i \hat{u}_i}_{\text{Doppler vector}} \;\big),$$

and note that the second block does not depend on $t_i$. Now take $m_i'$ to be the same physical target measured at a different lag $t_i'$ in the same accumulation window: moving radially at constant speed $d_i$ along $\hat{u}_i$, it was at $p_i' = p_i + d_i(t_i - t_i')\hat{u}_i$ at that moment, with the same radial Doppler and the same cross-section. Then

$$\pi(m_i') = \pi(m_i) \qquad \text{while} \qquad \varphi_{\text{raw}}(m_i') - \varphi_{\text{raw}}(m_i) = d_i(t_i - t_i')\,\hat{u}_i \neq 0$$

for every return with $d_i \neq 0$ and $t_i \neq t'_i$. Let $M'$ be $M$ with $m_i$ replaced by $m'_i$; the two measurement sets render to the identical student input and to different raw placements, so the raw placement is not a function of that input. The argument runs in the other direction as well: the compensated placement is likewise not a function of the raw rendering, since recovering it needs the same $t_i$.

**Admissibility.** Three conditions make $M'$ a measurement set the pipeline could have produced. The lag $t'_i$ must be one the window actually contains, which holds because a nuScenes sample carries returns at many distinct radar timestamps across the accumulation window. The displaced position must survive the range filter and fall in the grid, which holds for $|d_i(t_i - t'_i)|$ small relative to the $108$ m extent of the student's grid. And the target must move radially at constant speed over the interval, which is what the compensated placement already assumes. Since $\varphi_{\text{comp}}$ is exactly the position that assumption implies, a construction that relies on it grants the rendering no less than the rendering grants itself. Two remarks on scope. This is a statement about the operator over admissible measurement sets, not about the $6{,}019$ realised validation tensors, on which a lookup table would trivially be a function. And it establishes non-invertibility of the spatial assignment: $\pi(m_i)$ retains $d_i\hat{u}_i$, so nothing here bounds what a detector may infer about motion.

Under placement-preserving rendering the only change is that each return additionally carries the index of the placement that produced it, returns are routed to their group's encoder, and the two pooled vectors are written to disjoint channel slices of the same cell. The routing index stays in the tensor and reaches the encoder, where it is constant within a group and carries no within-group information: the first linear has no bias and is followed by batch normalisation, so a constant column shifts every point alike and is absorbed. The encoder's input width is $14$, the seven point features plus that index plus the six pillar-relative offsets. Each group has its own encoder in every reported system and in both placebo arms; the width-matched $32{+}32$ arm and the $48{+}16$ shared-encoder arm of Table 18 share one. In the identity duplicate the two groups hold the same returns at the same positions, so the index is the only input that differs between them.

## H Training and evaluation protocol

Within a system every arm shares one configuration; the arms of a comparison differ only in the factor the comparison names: on the student the raw and compensated arms are parameter-identical at $16{,}164{,}780$ weights, their configurations differing by one line. Table 15 gives each system's configuration. All five use Adam with a one-cycle learning-rate schedule peaking at $10^{-3}$ (warm-up over the first $40\%$ of iterations, initial rate a tenth of the peak, momentum cycled between $0.95$ and $0.85$), weight decay $0.01$ and gradient-norm clipping at $10$. The arms of Table 1 are trained from seeds $0$, $1$ and $2$; the remaining arms from seeds $0$ and $1$, except the two marked single-seed in Appendix J and the accumulation-depth pair, which use seed $0$. The TruckScenes arms of Table 2 use seeds $0$ and $1$. Every arm is evaluated at its final epoch; no checkpoint is selected on a validation metric. The metrics are the official ones: mAP averaged over four centre-distance thresholds, the true-positive errors mATE, mASE, mAOE, mAVE and mAAE, and NDS, which combines mAP with those five errors in a single weighted score. A change in mAVE therefore moves NDS on its own, which is what §3.2's two codes make separable. Detection is scored with the official nuScenes and TruckScenes evaluators on the validation split of each dataset, $6{,}019$ and $2{,}995$ keyframes. Training uses one NVIDIA RTX PRO 6000 Blackwell GPU.

## I TruckScenes: detection arms and measurement geometry

§5 reports both in prose: the detection arms of Table 2, and the qualification that the depth relationship of §3.2 holds within a measurement geometry rather than across geometries. This appendix gives the detail behind each.

**Two systems here, three on nuScenes.** The distillation system of Table 1 is absent from Table 2 because it requires a LiDAR teacher pretrained on the target dataset. On nuScenes that teacher is the one released with the system. TruckScenes provides LiDAR, but no teacher pretrained on it is released, so the system would first require pretraining one, which is outside the scope of this study.

**The depth sweep in full.** Table 16 is the measurement behind the claim that the cost appears at every depth. Every quantity in this appendix comes from one probe over one population, the $42{,}277$

Table 15: Per-system configuration. "Rotation" and "scaling" are the ranges of global augmentation; GT sampling pastes ground-truth objects from a database of training samples.

| nuScenes | CenterPoint | student | RadarDistill |
|---|---|---|---|
| accumulation | 10 sweeps / radar | 6 sweeps / radar | 6 sweeps / radar |
| range (m) | $\pm 51.2$ | $\pm 54.0$ | $\pm 54.0$ |
| cell | 0.2 m, $\leq$32 returns | 0.075 m, uncapped | 0.075 m, uncapped |
| pillar encoder | 2 layers, 64-d | 1 layer, 32-d | 1 layer, 32-d |
| backbone | dense 2D | sparse residual | sparse residual |
| output stride | 4 | 8 | 8 |
| class-balanced resampling | yes | no | yes |
| GT sampling | no | yes | yes |
| rotation | $\pm 22.5^\circ$ | $\pm 45^\circ$ | $\pm 45^\circ$ |
| scaling | 0.95–1.05 | 0.9–1.1 | 0.9–1.1 |
| translation | no | yes | yes |
| batch size | 8 | 16 | 16 |
| epochs | 30 | 20 | 12 |
| initialisation | random | random | released LiDAR teacher |
| **TruckScenes** | **CenterPoint** | **student** | |
| accumulation | 0.5 s | 0.5 s | |
| range (m) | $\pm 153.6$ | $\pm 153.6$ | |
| cell | 0.4 m, $\leq$32 returns | 0.4 m, $\leq$32 returns | |
| pillar encoder | 2 layers, 64-d | 1 layer, 32-d | |
| backbone | dense 2D | sparse residual | |
| output stride | 4 | 8 | |
| class-balanced resampling | yes | yes | |
| GT sampling | no | no | |
| rotation | $\pm 22.5^\circ$ | $\pm 22.5^\circ$ | |
| scaling | 0.95–1.05 | 0.95–1.05 | |
| translation | no | no | |
| batch size | 8 | 8 | |
| epochs | 30 | 30 | |
| initialisation | random | random | |

moving-vehicle boxes of the validation split that are measured at all six windows, so the depth comparison varies the window and nothing else. The trail-speed correlation behaves as it does on nuScenes, several times weaker under the compensated placement at every window; the localisation gain does not, being negative at every window with no crossing.

Table 16: The trade against accumulation time on TruckScenes val, 2,995 keyframes. The same 42,277 moving-vehicle boxes are measured at every window, so the population is fixed and only the window varies. The trail-speed correlation is the velocity code of Table 5, given here for both placements in one column, and the localisation gain is defined as it is there.

| $T_{\text{acc}}$ (s) | in-box raw | in-box comp. | loc. gain | trail-speed corr. raw / comp. |
|---|---|---|---|---|
| 0.10 | 242,298 | 211,967 | $-12.5\%$ | $+0.089$ / $+0.036$ |
| 0.20 | 417,990 | 342,269 | $-18.1\%$ | $+0.173$ / $+0.051$ |
| 0.30 | 540,559 | 433,340 | $-19.8\%$ | $+0.184$ / $+0.055$ |
| 0.40 | 631,625 | 505,283 | $-20.0\%$ | $+0.168$ / $+0.050$ |
| 0.50 | 698,312 | 567,239 | $-18.8\%$ | $+0.155$ / $+0.046$ |
| 0.70 | 776,777 | 678,664 | $-12.6\%$ | $+0.144$ / $+0.039$ |

**Why the counts fall, measured rather than argued.** Under constant velocity and exact Doppler, radial compensation cannot increase a return's Euclidean displacement from its current position:

that displacement is $-vt$ raw-placed and $-(I - \hat{u}\hat{u}^\top)vt$ after compensation, and the second is never longer. This does not guarantee preservation or improvement of in-box membership. The control of §5 measures both quantities on the same returns at $T_{\text{acc}} = 0.5\,\text{s}$ over the $42{,}277$ moving-vehicle boxes of the validation split, $947{,}537$ returns that one placement or the other puts inside a box.

Neither quantity is a distance to the box centre. Both are errors against the return's own current position under constant translation, in which an offset $q$ from the centre cancels, so the comparison is the point-target one. Because the applied displacement lies along the ray, the compensated error splits exactly,

$$\|e_{\text{comp}}\|^2 = \left(\hat{u}^\top\Delta - \hat{u}^\top vt\right)^2 + \left\|(I - \hat{u}\hat{u}^\top)vt\right\|^2, \tag{3}$$

the first term the disagreement between the displacement applied and the radial projection of the object's motion, the second what a correction restricted to that ray cannot remove even with an exact Doppler. The identity holds numerically to $3 \times 10^{-8}\,\text{m}^2$.

Compensation shortens the median error from $4.022$ to $1.824\,\text{m}$, the split at the median being $0.189\,\text{m}$ radial against $0.877\,\text{m}$ tangential. It is strictly worse exactly where $(\hat{u}^\top\Delta - \hat{u}^\top vt)^2 > (\hat{u}^\top vt)^2$, the tangential term cancelling, which is $29.9\%$ of returns. That subset is over-represented among departures, $43.4\%$ of the returns that leave the box against $20.8\%$ of those that stay.

Box membership moves the other way: $698{,}312$ in-box under the raw placement against $567{,}239$ under the compensated one, $-18.8\%$. Of the $380{,}298$ returns that leave, $56.6\%$ had the smaller error of the two. Membership does not follow from that magnitude: a box is anisotropic and rotated, narrow across its heading and long along it, so an error that shrinks while turning across the short axis crosses a face where a larger one along the long axis would not. No offset of the scatterer from the box centre is required for this, though such an offset adds to it. They leave across the heading in $61.3\%$ of cases, along it in $37.3\%$ and vertically in $1.4\%$.

Two limits on reading this. The radial term is a disagreement with a ground-truth velocity that is itself a finite difference between neighbouring annotations, so it absorbs annotation error, acceleration and rotation as well as any error in the Doppler; it is not a measurement of sensor error. And the tangential term is irreducible only under a correction confined to the ray, not in principle. What the control establishes is that neither displacement magnitude nor object extent accounts for the fall on its own, and that the radial-fraction stratification identifies neither. Probes: `truckscenes/depth_and_error_control.py` for the sweep, the stratification and the decomposition, `point_target_control.py` for the faces crossed; both report the same $42{,}277$ boxes and $947{,}537$ returns.

**The placement operator differs in one respect from the nuScenes construction of Appendix G.** TruckScenes returns carry elevation, so $\hat{u}_i$ is a three-dimensional ray and the compensated placement displaces $z$ as well as $x$ and $y$; on nuScenes the displacement is planar and $z$ is untouched. The consequence is small but not nothing. Radar elevation is poorly resolved, so a large Doppler on a steep ray produces a large vertical displacement, up to 22 m against a grid that admits $z \in [-5, 3]\,\text{m}$, and $0.19\%$ of the returns the raw placement keeps are pushed outside the vertical range, against $0.09\%$ pushed outside the lateral one. Both TruckScenes bin sets rotate the ego-motion-compensated Doppler vector into the reference frame along with the positions, so neither arm here carries the frame defect of §6.1. Points move into the grid as well as out, and the compensated rendering ends with $0.13\%$ more returns than the raw one. This is not a population asymmetry between the arms but a difference in the operator, to be read alongside the geometry results below.

On CenterPoint the both-against-raw margins are $+0.95$ and $+1.20$ mAP, $+1.70$ and $+1.28$ NDS, and $-10.50$ and $-1.29$ mAVE. Detection transfers and velocity does not. Both detection metrics improve at both seeds, above the larger arm spread in every case. The velocity margin improves in sign at both seeds, but the second is under a third of the spread it must clear, so the seed-0 figure should not be read as a velocity result: at one seed it was $2.2\times$ that spread, at the other $0.3\times$. The per-class pattern moves the same way: seven of the nine classes that score velocity improve at seed 0, four of nine at seed 1, and the two largest seed-0 class margins, motorcycle at $-28.80$ and bus at $-23.98$, fall to $-1.74$ and $-8.16$.

What does replicate is where the detection gain comes from. It is concentrated in one class, motorcycle, at $+15.18$ and $+14.39$ AP, contributing $+1.27$ and $+1.20$ to a twelve-class mean whose total margins are $+0.95$ and $+1.20$. On both seeds that single class accounts for the whole of the mean gain, so the aggregate figure should not be read as a broad improvement.

**The single-factor contrast runs the other way here.** On CenterPoint, compensation against raw, the same comparison §4.1 draws on the three nuScenes systems, costs $1.26$ and $1.53$ mAP and $4.54$ and $4.66$ NDS, at 3.6–4.4 and 6.1–6.3 times the larger arm spread. It is worse on every metric at both seeds, so on this dataset the compensated placement is not a trade but a loss. The aggregate velocity margin, $+0.76$ and $+2.73$, understates the velocity damage: it is computed over classes the compensated arm sometimes fails to detect at all, which the protocol charges the maximum error, $100.00$ in these units, rather than the larger error a detected-but-mistimed class would earn. On the five classes with non-trivial AP under both placements the velocity error rises at both seeds without exception, by $17.85$ to $39.48$ at the first seed and $2.67$ to $41.19$ at the second.

**One of the three results replicates on the student.** Its both-placements arm is the best of the three renderings on every metric at both seeds, and on NDS the margin also clears the spread: $+1.57$ and $+2.17$ over raw at $4.5$ and $6.2$ times the larger arm spread, $+4.98$ and $+3.63$ over compensated at $3.1$ and $2.3$ times. Keeping both placements therefore beats either single placement above that spread here as it does on CenterPoint, which on nuScenes holds for the comparison against the compensated placement on the student and on RadarDistill and for none against the raw one. The other two margins do not survive their own noise. On mAP the arm gains $+1.52$ then $+0.07$ over raw, the second $0.06$ times the $1.25$ spread it must clear, which the both-placements arm itself sets; on mAVE it improves at both seeds, $-4.84$ and $-1.06$, against $5.25$ from the raw arm, so it clears that spread at neither seed where CenterPoint cleared it at one. The single-factor contrast likewise points the way CenterPoint's does without settling it: raw ahead of compensated on NDS by $3.41$ and $1.46$ against a spread to clear of $1.60$, and on velocity by $5.87$ and $2.52$ against $8.60$, while on mAP the sign flips, $+0.44$ then $-0.06$. This system's arms are the loosest we report on this dataset, wider than CenterPoint's on every metric ($1.25$ against $0.60$ mAP, $1.60$ against $0.74$ NDS, $8.60$ against $4.76$ mAVE), which is what its two weaker margins run into. We read one result from it, the NDS margin for keeping both placements, and none from its detection or velocity margins.

Re-measuring the same two quantities on TruckScenes, a dataset with a different sensor layout and a much longer range, the cost transfers in aggregate: the compensated rendering's trail-speed correlation is several times weaker at every accumulation depth, $+0.036$ to $+0.055$ against $+0.089$ to $+0.184$ (Table 16). It is substantially reduced rather than absent, as on nuScenes, where compensation leaves $+0.297$ of $+0.764$. The localisation gain does not transfer at all: it is negative at every window rather than rising with depth.

Both sides of the aggregate are carried by the same subset, and that qualifies the asymmetry this paper reports. Table 17 stratifies by radial fraction: only in the top bin, $64\%$ of boxes, does compensation both consolidate returns and remove the code, $+0.128$ to $-0.037$. In the $36\%$ whose motion is only partly radial it scatters returns and barely touches the code, removing at most $11\%$ of it and in one bin raising it by $14\%$, against the $61\%$ it removes on nuScenes. So on this dataset the cost is consistent in the aggregate and concentrated where the benefit is, rather than being independent of geometry the way the nuScenes measurements alone would suggest.

The difference resolves when objects are stratified by the radial fraction of their motion, $f = |\cos\angle(v_o, r)|$, the alignment between an object's velocity and the sensor-to-object ray. Where motion is essentially all radial, which covers $64\%$ of boxes in this dataset, the behaviour matches nuScenes. Where motion is only partly radial, in-box counts fall. The stratification associates the two; it does not establish that a ray-bound correction displacing returns away from the object is what produces the fall, which is the question the control above addresses.

Table 17 gives the stratification. The aggregate is negative because the four non-radial bins lose more returns than the radial bin gains, and the velocity code survives compensation in exactly the bins where it fails to consolidate.

**The sign is not an artefact of displacing $z$.** Since the operator here moves returns vertically and membership is tested in three dimensions, a natural objection is that compensation loses in-box returns through the axis radar resolves worst. Repeating the measurement with the vertical component of the displacement zeroed, so the correction is planar as on nuScenes, moves the localisation gain at $T_{\text{acc}} = 0.5\,\text{s}$ from $-18.8\%$ to $-16.5\%$. The sign and most of the magnitude survive, so the effect belongs to the radial-only compensation and not to elevation.

This is a limit on the generality of the depth relationship, not a contradiction of it. The displacement accumulated during the window still governs the conflict; how that conflict manifests depends on

Table 17: Localisation gain and velocity code by radial fraction $f = |\cos \angle(v_o, \hat{u})|$, $T_{\text{acc}} = 0.5\,\text{s}$, over the same 42,277 boxes as Table 16. Essentially radial means the top bin, $f > 0.98$, which holds 27,098 of them. Only there does compensation consolidate returns and remove the velocity code, as it does on nuScenes; every other bin scatters, and in one the code is slightly stronger after compensation than before.

| radial fraction $f$ | boxes | localisation gain | velocity code, raw | velocity code, comp. |
|---|---|---|---|---|
| $< 0.33$ | 1,265 | $-30.3\%$ | $+0.241$ | $+0.223$ |
| 0.33–0.66 | 1,776 | $-58.5\%$ | $+0.207$ | $+0.192$ |
| 0.66–0.90 | 3,801 | $-56.1\%$ | $+0.196$ | $+0.223$ |
| 0.90–0.98 | 8,337 | $-29.7\%$ | $+0.193$ | $+0.172$ |
| $> 0.98$ | 27,098 | $+44.3\%$ | $+0.128$ | $-0.037$ |

the geometry through which the displacement is measured. It is also the reason §3.2 reports the relationship on one dataset rather than asserting it as a law.

## J THE COMPLETE CENTERPOINT RESULT SET

§6 reports the arms and metrics that carry the argument. This appendix gives every CenterPoint arm, with detection and composite metrics in Table 18 and the remaining true-positive errors in Table 19. The reference row is the original, uncorrected arm, the one whose Doppler features are expressed in the wrong frame; the corrected single-placement arm and the two placebo arms are the ladder of Table 3 and are not repeated here. All values are read from the evaluation outputs by `code/probes/centerpoint_full_table.py`; two seeds are shown as $s_0/s_1$ and a dagger marks a single-seed arm. "sep. enc." means each channel group has its own pillar encoder, "shared enc." that one encoder is applied to both.

Table 18: All ten CenterPoint arms, detection and composite metrics. Total bird's-eye-view width is the representation size the arm renders into; the reference is 64. The body reports the both-placements (48+16, sep. enc.) arm, which leads on mAP at both seeds but not in every column. No row is marked.

| arm | BEV | mAP ↑ | NDS ↑ | mAVE ↓ |
|---|---|---|---|---|
| single placement (uncorrected ref.) | 64 | 7.98 / 7.96 | 32.32 / 32.46 | 37.60 / 37.33 |
| compensated + point features | 64 | 8.28 / 8.23 | 31.63 / 31.44 | 44.33 / 44.23 |
| placement-preserving 32+32 | 64 | 9.76 / 9.78 | 35.21 / 34.98 | 33.94 / 34.69 |
| placement-preserving 48+16, shared enc. | 64 | 9.75 / 9.88 | 35.14 / 35.57 | 33.54 / 32.81 |
| both placements (48+16, sep. enc.) | 64 | 10.25 / 10.12 | 35.68 / 35.81 | 31.88 / 32.46 |
| sum + 2 occupancy bits | 66 | 9.78 / 9.98 | 35.25 / 35.37 | 31.77 / 32.12 |
| placement-preserving 64+16, sep. enc.† | 80 | 10.10 | 35.71 | 32.52 |
| placement-preserving 64+16, trail-wide† | 80 | 9.97 | 35.35 | 32.63 |
| placement-preserving 16+64, comp.-wide | 80 | 10.02 / 9.83 | 35.29 / 34.95 | 32.67 / 34.52 |
| placement-preserving 64+64 | 128 | 8.73 / 8.69 | 33.35 / 33.12 | 38.83 / 37.74 |

Three readings follow that the body has room only to assert. First, the improvement is not confined to the three reported metrics: against the reference the both-placements arm improves translation by $0.034/0.047$, scale by $0.006/0.012$, orientation by $0.094/0.091$ and attribute error by $0.030/0.028$, at both seeds. Second, no arm wider than 64 channels exceeds the both-placements arm on mAP: the three 80-channel variants land between 9.83 and 10.10, and the 128-channel arm is markedly lower at $8.73/8.69$. Two of the three 80-channel arms are single-seed, so no width trend is claimed. Third, a composite score can conceal a change that both of its components register: at the first seed compensation's detection gain and velocity cost leave NDS unchanged, 35.21 against 35.20, well inside both arms' NDS spreads, 0.28 and 0.31, while at the second, with no gain left to offset the cost, NDS simply falls.

Table 19: The same ten arms, named as in Table 18, remaining nuScenes true-positive errors, all lower-is-better. All four improve for the both-placements arm against the reference at both seeds; it is not the lowest arm in every column, and the 80-channel arms are lower on some of them.

| arm | mATE | mASE | mAOE | mAAE |
|---|---|---|---|---|
| reference | 0.6207 / 0.6277 | 0.2989 / 0.3025 | 0.7226 / 0.7050 | 0.1483 / 0.1436 |
| comp. + point features | 0.6255 / 0.6233 | 0.2989 / 0.2997 | 0.7462 / 0.7490 | 0.1375 / 0.1533 |
| 32+32 | 0.5859 / 0.5941 | 0.2925 / 0.2935 | 0.6266 / 0.6402 | 0.1231 / 0.1163 |
| 48+16, shared enc. | 0.5853 / 0.5805 | 0.2926 / 0.2928 | 0.6443 / 0.6089 | 0.1157 / 0.1268 |
| 48+16, sep. enc. | 0.5869 / 0.5803 | 0.2925 / 0.2909 | 0.6283 / 0.6142 | 0.1180 / 0.1153 |
| sum + 2 occ. bits | 0.5886 / 0.5864 | 0.2924 / 0.2937 | 0.6452 / 0.6441 | 0.1204 / 0.1163 |
| 64+16, sep. enc.† | 0.5742 | 0.2914 | 0.6266 | 0.1169 |
| 64+16, trail-wide† | 0.5858 | 0.2927 | 0.6436 | 0.1147 |
| 16+64, comp.-wide | 0.5923 / 0.5845 | 0.2958 / 0.2930 | 0.6397 / 0.6477 | 0.1176 / 0.1258 |
| 64+64 | 0.6074 / 0.6112 | 0.3018 / 0.3009 | 0.6736 / 0.6893 | 0.1305 / 0.1436 |

**The width-matched control.** The placement-preserving 32+32 arm renders both placements at the reference's total width and with 3,360 fewer parameters than it. Against the uncorrected reference it gains $+1.78$ and $+1.81$ mAP; against the corrected single placement, $+0.21$ and $+0.07$. Matching capacity is therefore not a substitute for the single-factor ladder of §6.2: most of what the matched arm's apparent gain measured was the point-feature defect left in its denominator.

**A cheaper variant.** A further arm bears on what the intervention needs to be rather than on whether placement matters. Summing the two groups into one feature map and appending two bits recording which placements were present costs 1,184 parameters against 8,864, and reaches 9.78 and 9.98 mAP at two seeds. Its detection gap to the separate-group arm is asymmetric: 0.47 at the first seed, which exceeds either arm's observed spread, and 0.14 at the second, which is inside this arm's own spread of 0.20. On velocity it is the better arm at both seeds. Measured from the corrected single placement, a two-bit record of which placements were present recovers a third of the separate-group arm's mAP gain at the first seed and two thirds at the second, at under a seventh of the parameter cost.

### PER-CLASS RESULTS, AND WHERE THE ACCOUNT FAILS

In Table 21, displacement, the $> 0.5$ m fraction and pillar change are point-weighted, while the Jaccard overlap of a class's occupied-pillar sets and mean speed are box-weighted, so those columns are not directly comparable. Returns claimed by more than one box are 0.58% and are left in.

Table 20: Per-class AP and velocity error, reference against the both-placements arm, both seeds. All ten classes improve in AP at both seeds. Barrier and traffic cone carry no velocity score in the nuScenes protocol.

| class | ref AP | both AP | Δ | ref AVE | both AVE | Δ |
|---|---|---|---|---|---|---|
| car | 40.05/40.06 | 43.22/43.78 | $+3.20/+3.70$ | 35.70/36.17 | 32.38/31.96 | $-3.30/-4.20$ |
| bus | 13.11/12.58 | 13.90/13.04 | $+0.80/+0.50$ | 54.85/57.50 | 52.05/52.49 | $-2.80/-5.00$ |
| truck | 7.59/7.56 | 8.69/8.59 | $+1.10/+1.00$ | 35.15/35.17 | 32.44/30.95 | $-2.70/-4.20$ |
| pedestrian | 6.58/5.97 | 8.15/8.60 | $+1.60/+2.60$ | 55.90/55.86 | 47.22/48.10 | $-8.70/-7.80$ |
| barrier | 6.07/6.39 | 14.60/15.63 | $+8.50/+9.20$ | not defined by nuScenes | | |
| traffic cone | 2.54/3.16 | 3.99/3.58 | $+1.50/+0.40$ | not defined by nuScenes | | |
| trailer | 2.34/2.19 | 5.82/5.51 | $+3.50/+3.30$ | 27.51/22.49 | 16.28/18.44 | $-11.20/-4.10$ |
| motorcycle | 1.47/1.71 | 3.34/1.79 | $+1.90/+0.10$ | 55.48/54.91 | 43.07/47.93 | $-12.40/-7.00$ |
| bicycle | 0.04/0.03 | 0.69/0.60 | $+0.70/+0.60$ | 28.35/27.34 | 21.15/20.33 | $-7.20/-7.00$ |
| constr. | 0.00/0.00 | 0.09/0.09 | $+0.10/+0.10$ | 7.89/9.23 | 10.47/9.52 | $+2.60/+0.30$ |

**The mAP gain is concentrated even though every class improves.** Table 20 gives the per-class breakdown. nuScenes mAP weights all ten classes equally, so barrier's $+8.50$ and $+9.20$ contribute $+0.85$ and $+0.92$ mAP of aggregate gains of $+2.27$ and $+2.16$, which is $38\%$ and $43\%$ from one class. The improvement is broad in sign and concentrated in magnitude.

Table 21: How much the intervention actually changes each class's own returns. Full validation split; the columns are not directly comparable, see text.

| class | mean disp. | $> 0.5$ m | pillar change | Jaccard | speed | $\Delta$ AP |
|---|---|---|---|---|---|---|
| bus | 0.894 m | 39.9% | 59.1% | 0.441 | 2.90 | +0.60 |
| car | 0.803 m | 29.2% | 48.2% | 0.533 | 2.19 | +3.50 |
| motorcycle | 0.708 m | 30.7% | 51.3% | 0.547 | 1.97 | +1.00 |
| truck | 0.583 m | 23.0% | 42.9% | 0.582 | 1.51 | +1.10 |
| bicycle | 0.391 m | 18.0% | 37.8% | 0.619 | 0.87 | +0.60 |
| trailer | 0.361 m | 16.2% | 36.6% | 0.634 | 0.71 | +3.40 |
| pedestrian | 0.251 m | 17.0% | 55.1% | 0.473 | 0.86 | +2.10 |
| traffic cone | 0.144 m | 5.2% | 24.2% | 0.732 | 0.07 | +0.90 |
| constr. veh. | 0.124 m | 5.1% | 26.3% | 0.687 | 0.16 | +0.10 |
| barrier | 0.112 m | 3.6% | 20.9% | 0.724 | 0.07 | +8.90 |

Rank correlations between how much a class's returns move and how much that class gains are mildly negative: $-0.32$ on absolute and $-0.55$ on relative AP gain, over the nine classes with non-zero reference AP. Excluding bicycle as well, whose reference AP of $0.04$ makes a relative gain meaningless, they are $-0.38$ and $-0.64$. Euclidean displacement understates the change the detector sees, because pillarisation is discontinuous: barrier's sub-pillar mean shift still relocates $20.9\%$ of its returns and alters $27.6\%$ of its occupied cells. Barrier remains the smallest representation change of any class and the largest gain. The two measures also order classes differently. Pedestrian has near-barrier displacement but the second-highest pillar churn, small objects sitting near cell boundaries, which is why we report pillar change as the primary quantity.

**Foreign-trail contamination does not explain it either.** If small static objects were harmed by other objects' trails sweeping their cells, the compensated group would clean those foreign returns away and barrier would gain without its own returns moving. Counting returns owned by objects moving at least 2 m/s that land in each box, at four dilations up to 4 m so that the backbone's receptive field is covered, barrier has almost no such contamination to remove, and what little it has compensation increases at every scale. The only classes genuinely decontaminated are trailer and bus, and bus is among the smallest AP gains of any class. Neither displacement magnitude nor foreign-trail contamination accounts for the per-class pattern.

### Alternatives to keeping both placements

**A fixed blend.** Displacing every return by a fraction $\alpha$ of its own compensation keeps each return once. Scored as in Table 5 at the deepest window, with each quantity expressed as a fraction of its value at the better endpoint, a single blended placement keeps $1.17$, $1.24$ and $1.29$ of the two codes together at $\alpha = 0.25$, $0.5$ and $0.75$, against $1.39$ for full compensation and $2.00$ for keeping both placements. Adding two separately normalised quantities is a descriptive summary, not a measure of information or of complementarity. Localisation gain rises from $+14.2\%$ to $+44.1\%$ over those fractions while the raw-normalised velocity code falls from $0.90$ to $0.48$.

**A coarser grid.** Enlarging the cell shrinks the difference between the placements without removing it. At $0.8$ m pillars compensation moves a return by $0.3$ cells on average, against $1.0$ at $0.2$ m, and near fast movers the share of occupied cells reached by only one placement falls from $73.2\%$ to $55.7\%$.

## K SEED SPREADS

Tables 1 and 2 summarise each arm as mean ± sample standard deviation over its seeds, and Table 22 gives the seeds themselves, which is what the paired differences in the text are formed from. We quote each arm's own seed spread rather than a global noise floor, because the metrics differ by an order of magnitude in seed sensitivity. Seed spread means the range across the seeds an arm has on file, used as the descriptive scale against which a margin is judged (Bouthillier et al., 2021).

The original CenterPoint reference spreads $0.01$ mAP, $0.14$ NDS and $0.27$ mAVE, and the student's compensated reference $0.09$, $0.14$ and $0.27$. Over the three seeds of Table 1 the CenterPoint arms spread $0.21/0.28/1.32$ (raw), $0.13/0.31/0.49$ (compensated) and $0.32/0.67/0.76$ (both) on mAP, NDS and mAVE, the student's $0.12/0.45/0.87$, $0.09/0.14/0.27$ and $0.04/0.36/0.44$, and RadarDistill's $0.20/0.63/1.13$, $0.59/0.32/2.08$ and $0.22/0.29/0.88$. Velocity is where the third seed changed most: RadarDistill's raw arm spreads $1.13$ there against $0.09$ over its first two. RadarDistill's compensated arm is the widest of them at $0.59$ mAP, which is why every difference is formed within a seed and judged against these spreads rather than against the summary in Table 1, and why the third seed changes which margins clear their own arm's variation. On TruckScenes the reference spreads $0.35$ mAP, $0.74$ NDS and $4.76$ mAVE and the intervention $0.60$, $0.32$ and $4.46$; velocity is an order of magnitude noisier there than on nuScenes, which is why its margin clears that spread at one seed only. The student's arms on that dataset are looser still on every metric, to $1.25$ mAP, $1.60$ NDS and $8.60$ mAVE, which is why only its NDS margins, judged against arm spreads of $0.25$ and $0.35$, are read as results.

Table 22: Per-seed values behind Tables 1 and 2, as $s_0/s_1$ or $s_0/s_1/s_2$. The main tables summarise these as mean ± sample standard deviation; the paired differences quoted in the text are formed within a seed, which a mean and a deviation cannot express.

| nuScenes | rendering | mAP ↑ | NDS ↑ | mAVE ↓ |
|---|---|---|---|---|
| CenterPoint | raw | 9.55 / 9.70 / 9.76 | 35.20 / 35.36 / 35.08 | 31.93 / 31.44 / 32.76 |
| | compensated | 9.90 / 9.77 / 9.83 | 35.21 / 34.94 / 35.25 | 33.22 / 33.46 / 33.71 |
| | both | 10.25 / 10.12 / 9.93 | 35.68 / 35.81 / 35.14 | 31.88 / 32.46 / 32.64 |
| PillarNet-radar student | raw | 4.24 / 4.14 / 4.26 | 29.14 / 29.06 / 29.51 | 39.45 / 38.63 / 39.49 |
| | compensated | 4.51 / 4.53 / 4.44 | 29.09 / 29.23 / 29.16 | 44.52 / 44.25 / 44.28 |
| | both | 4.48 / 4.45 / 4.44 | 29.92 / 29.71 / 29.56 | 39.13 / 39.15 / 38.71 |
| RadarDistill | raw | 14.28 / 14.30 / 14.48 | 38.74 / 38.49 / 39.12 | 30.70 / 30.79 / 31.83 |
| | compensated | 13.70 / 14.29 / 13.95 | 38.23 / 38.17 / 38.49 | 33.58 / 32.75 / 31.50 |
| | both | 14.79 / 14.56 / 14.59 | 39.27 / 39.24 / 38.98 | 31.49 / 31.48 / 32.36 |
| TruckScenes | rendering | mAP ↑ | NDS ↑ | mAVE ↓ |
| CenterPoint | raw | 14.80 / 15.15 | 31.08 / 31.83 | 99.33 / 94.57 |
| | compensated | 13.54 / 13.62 | 26.55 / 27.17 | 100.09 / 97.30 |
| | both | 15.75 / 16.35 | 32.79 / 33.11 | 88.82 / 93.28 |
| PillarNet-radar student | raw | 16.16 / 16.35 | 32.09 / 31.74 | 88.85 / 83.60 |
| | compensated | 16.60 / 16.29 | 28.68 / 30.28 | 94.72 / 86.12 |
| | both | 17.67 / 16.42 | 33.66 / 33.91 | 84.00 / 82.54 |

Evaluation is itself not deterministic, because the pipeline shuffles points before a fixed per-cell cap: re-scoring one checkpoint moves mAP, NDS and mAVE by $0.01$, $0.00$ and $0.03$.

## L LIMITS

§7 states these in summary.

The single-factor placement contrast is measured on three nuScenes systems at three seeds each and on two TruckScenes systems at two. The decomposition assigns placement correspondence $23\%$ of

the apparent mAP gain at one seed and $6\%$ at the other. The conditional analyses use post hoc strata over matched detections, which select the evaluation population.

The five-arm decomposition is exploratory rather than pre-registered. Its increments are sequential single-factor contrasts along one evaluated ordering; they telescope to the total by construction, which is arithmetic rather than causal independence, and alternative orderings and interactions between the factors are not evaluated. The decomposition and both placebos exist on CenterPoint alone. Whether an identity duplicate or a displacement-matched permutation would reproduce as much of the apparent gain on the student or on RadarDistill is untested, so the share attributable to correspondence is measured on one system and assumed on none.

Cross-dataset evidence is partial in a different way. For the both-placements arm on TruckScenes, NDS transfers on both systems and at every seed and velocity does on neither; mAP transfers on CenterPoint alone, where it rests on a single class, and stays inside its own arm spread on the student. The single-factor contrast on the same dataset runs the other way, so the two should not be read as one result. No absolute state-of-the-art claim is made.

Three claims here are easy to run together and should not be. §3.5 establishes that where the lag is discarded the alternative placement is not a function of the detector's input, and separately that the tangential component is not observable within a rendered cell. Neither establishes that object velocity is unrecoverable: the tensor retains the radial Doppler, and a detector may infer motion from it and from context. And where the lag is retained, the placement is reconstructible in principle, so the CenterPoint velocity result is evidence about exploitation, not about possibility. Downstream the two sides are uneven, and uneven in one direction: trail-borne motion evidence is more extractable from the rendered tensor under every frozen encoder and reaches velocity estimation at every seed on two of the three systems, while the concentration the compensated placement provides reaches detection accuracy on one of the three nuScenes systems and reverses on TruckScenes. Exploitation was partial and metric-specific, not absent, and the two sides are not partial to the same degree. The problem also requires a rendering step at all: detectors that keep encoded points (Wolters et al., 2025), and occupancy models that consume the raw radar tensor rather than a point cloud (Ding et al., 2024), never make the assignment we study, so the loss is a property of an operator they need not apply. Finally, the results concern accumulated automotive radar, while the difficulty they illustrate is not obviously specific to it.