TOPORADAR: LEARNING PERSISTENT TOPOLOGICAL STRUCTURE FOR MULTI-VIEW RADAR SEMANTIC SEGMENTATION

ABSTRACT

Radar semantic segmentation requires recovering foreground structure from sparse and fragmented observations. Range–angle, range–Doppler, and angle–Doppler projections provide complementary spatial and motion information, yet their different projection geometries can separate or merge object responses. Pixel-wise supervision does not explicitly constrain the resulting connectivity errors. We formulate multi-view radar semantic segmentation as a topological structural recovery problem, seeking to recover annotation-consistent foreground structure while respecting differences between views. We introduce **TopoRadar**, which couples persistent feature conditioning with component-level structural supervision. Its **DuoPerLay** module computes sublevel and superlevel cubical persistence from learned multi-view features, characterising connectivity and holes across filtration thresholds to condition the segmentation representation. A structural consistency loss uses optimal transport to align persistent component summaries between predictions and annotations within each output view, providing component-level supervision for foreground structure. On CARRADA, TopoRadar achieves 64.87% range–Doppler mIoU, outperforming the published TransRadar and recent HyperRadar results by 0.97 and 1.07 percentage points, respectively; in the range–angle view, it exceeds HyperRadar while remaining below TransRadar and MARSS. Relative to a TransRadar backbone re-trained under our protocol, TopoRadar reduces connected-component error by approximately 30% and spurious-component counts by approximately 35% across both views. A single-view adaptation achieves 83.17% mIoU on RADIal, improving TransRadar by 2.07 percentage points.

1 INTRODUCTION

Radar is an important sensing modality for autonomous driving because it remains informative under adverse illumination and weather conditions that can degrade cameras and LiDAR (Yao et al., 2023). However, dense radar semantic segmentation remains difficult because radar measurements are sparse, noisy, low-resolution, and only weakly semantic (Dalbah et al., 2024; Li et al., 2025). Object evidence is therefore often fragmented, incomplete, and view-dependent across range–angle (RA), range–Doppler (RD), and angle–Doppler (AD) projections. Radar segmentation consequently requires not only local feature discrimination, but also the recovery of coherent foreground structure from incomplete observations.

Existing radar segmentation methods adapt dense prediction architectures to RA, RD, AD, and related radar representations (Ouaknine et al., 2021a; Zhang et al., 2023; Dalbah et al., 2024; Li et al., 2024; Chen et al., 2026). Multi-view fusion and radar-specific refinement have improved segmentation accuracy, but training remains dominated by pixel-wise overlap, local classification, and cross-view consistency. These objectives do not explicitly distinguish a coherent object response from a prediction with similar pixel accuracy but broken connectivity, fragmented regions, or spurious components. We therefore formulate radar segmentation as a *topological structural recovery* problem: predictions should recover annotation-consistent foreground organisation within each supervised view, without assuming identical topology across RA, RD, and AD. Figure 1 illustrates this motivation.

Persistent homology provides a natural representation of structural organisation by tracking connected components, holes, and their stability across filtration levels. Topological supervision has improved dense prediction when geometric consistency is important (Su et al., 2025; Clough et al., 2020; Hu et al., 2019), while differentiable cubical persistence layers such as *CuMPerLay* integrate learnable filtrations and persistence vectorisation into neural networks for grid-structured data (Korkmaz et al.,

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.

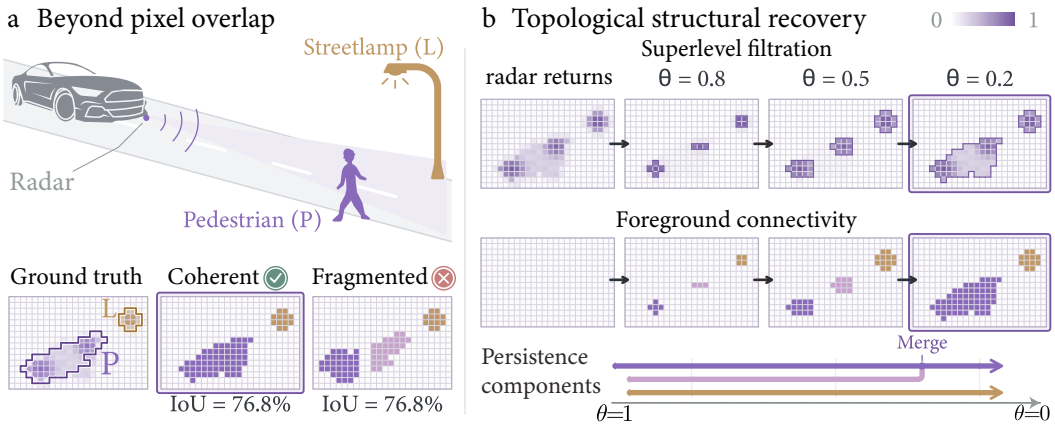


Figure 1: **TopoRadar recovers coherent object structure from fragmented radar evidence.** (a) Two predictions with equal IoU (76.8%): one keeps the pedestrian (P) whole, the other fragments it. (b) Under superlevel filtration, the two peaks merge only at $\theta \approx 0.32$; bars show component lifetimes, arrows mark persisting components. TopoRadar uses this structure to condition features and supervise foreground components.

2025). These developments are relevant to radar because its projected representations are defined on regular grids. Nevertheless, existing persistent-homology segmentation losses mainly constrain output masks, while differentiable persistence layers generally describe a single representation or filtration family. They do not directly address persistent feature conditioning in sparse multi-view radar or flexible alignment between predicted and annotated component sets of unequal size.

We introduce **TopoRadar**, which incorporates persistent structure at both the representation and output levels. First, **DuoPerLay** learns complementary sublevel and superlevel cubical filtrations over the fused radar representation. The two branches describe structures emerging from lower- and higher-valued regions of the learned feature maps, respectively, without assuming that feature magnitude has a fixed physical interpretation. Their persistent descriptors condition the fused representation before adaptive attention and mask decoding. Second, the **Structural Consistency Loss** extracts persistent foreground components from predicted and ground-truth masks within each output view. Each component is represented by its location, foreground-support mass, and persistence. Direct global topology matching would be overly rigid because predicted and annotated masks can contain different numbers of components and may exhibit partial or noisy correspondence. We therefore align their component summaries using unbalanced optimal transport, which permits unmatched mass while penalising missing, fragmented, and spurious foreground structure.

Unlike existing radar segmentation methods, TopoRadar supervises persistent component structure rather than only pixel overlap and cross-view agreement. Unlike prior topological losses, which impose one-to-one feature matching or prior Betti numbers, it transports component mass with relaxed marginals, and complements this training signal with persistence-based conditioning that remains active at inference. Our contributions are:

- We formulate sparse multi-view radar semantic segmentation as a **topological structural recovery** problem, capturing fragmentation, broken connectivity, and spurious foreground components that are not explicitly constrained by pixel-wise objectives.
- We introduce **DuoPerLay**, a cubical persistence module that learns sublevel and superlevel multifiltrations directly from fused radar features, rather than thresholding a fixed output map, and uses their vectorised persistence to condition the segmentation backbone.
- We propose a **Structural Consistency Loss** that aligns persistent prediction and ground-truth component summaries within each output view using unbalanced optimal transport, accommodating unequal component counts and partial correspondence.

2 Related Work

Radar Semantic Segmentation: Automotive radar is robust to adverse weather and illumination, but sparse, low-resolution, and cluttered measurements complicate dense prediction (Palffy et al., 2020; Schumann et al., 2021). Earlier methods used point-cloud or radar–LiDAR representations (Bansal et al., 2022; Feng et al., 2020), while recent approaches exploit radar-native RA/RD projections, raw signals, and RAD tensors (Rebut et al., 2022; Wang et al., 2021; Ouaknine et al., 2021a; Zhang et al., 2021). CARRADA (Ouaknine et al., 2021b) and RADIal (Rebut et al., 2022) have supported architectures including TMVA-Net (Ouaknine et al., 2021a), T-RODNet (Jiang et al., 2022), PeakConv (Zhang et al., 2023), TransRadar (Dalbah et al., 2024), AdaPKC (Li et al., 2024), and MARSS (Chen et al., 2026). HyperRadar (Zia et al., 2026c) models higher-order relations using learnable hypergraphs and UOT-based cross-view alignment, but does not explicitly constrain persistent foreground connectivity, fragmentation, or spurious components. Existing objectives therefore remain dominated by pixel-wise, localisation, and cross-view supervision rather than persistent structural supervision.

Topological Deep Learning. Topological data analysis captures non-local structure through connectivity, holes, and persistence, while topological deep learning integrates such higher-order structure into neural models (Su et al., 2025; Zia et al., 2024; 2026b). Persistent homology and its vectorisations enable topology-aware learning pipelines (Ali et al., 2023), with applications in medical and biological imaging (Crawford et al., 2020; Garside et al., 2019). For dense prediction, persistent-homology losses regularise connectedness and holes (Clough et al., 2020). Stucki et al. (2023) introduced Betti matching based on induced matchings of persistence barcodes to preserve spatial correspondence between topological features, while Wen et al. (2025) further incorporated spatial-domain information into persistent-feature matching. Cubical persistence is well suited to grid data (Rieck et al., 2020; Bleile et al., 2022), and CuMPerLay provides differentiable multiparameter persistence with learnable multifiltrations (Korkmaz et al., 2025).

Optimal Transport for Structural Alignment: Optimal transport (OT) provides a principled mechanism for aligning structured sets under soft correspondence (Peyré & Cuturi, 2019). Entropic regularisation and Sinkhorn solvers make OT practical at modern learning scales (Cuturi, 2013), with applications to domain adaptation (Courty et al., 2016) and topology-aware test-time anomaly segmentation (Zia et al., 2026a). Since classical OT assumes balanced mass, unbalanced OT relaxes exact mass preservation and is better suited to partial, noisy, or unequal supports (Séjourné et al., 2023). Rather than aligning persistence diagrams of a single map with one another, e.g., across thresholds or filtrations, we use UOT only in supervised training to match persistent components of predicted and annotated masks by location, mass, and persistence, with relaxed marginals for components that differ in number or support.

3 Topology-Aware Multi-View Radar Semantic Segmentation

3.1 Problem Formulation: Sparse Radar Segmentation as Topological Structural Recovery

Let $\mathcal{X} = \{X^{RA}, X^{RD}, X^{AD}\}$ denote the multi-view radar input, where each view provides a sparse and noisy projection of the same driving scene. The goal of radar semantic segmentation is to predict view-specific semantic masks $\hat{Y}^v \in [0,1]^{C\times H_v\times W_v}$ for $v \in \{\mathrm{RA}, \mathrm{RD}\}$, given ground-truth masks Y^v. Here, C is the number of semantic classes, $H_v \times W_v$ is the spatial resolution of view v, and $\Omega^v = \{1,\dots,H_v\} \times \{1,\dots,W_v\}$ denotes the corresponding radar grid, with $u \in \Omega^v$ indexing a radar cell. Standard segmentation objectives optimise local agreement between $\hat{Y}^v$ and Y^v, for example through cross-entropy, Dice, or IoU-based losses. However, such objectives mainly compare cell-wise labels or regional overlap and do not explicitly encode whether the predicted foreground forms coherent persistent object structures.

We formulate sparse multi-view radar segmentation as a **topological structural recovery** problem. Two masks can have similar Dice or IoU scores while differing structurally, for example when one preserves a single connected foreground component and the other splits the same object into several pieces. Let P^v_{fg} and Y^v_{fg} denote the predicted and annotated foreground-support maps, i.e., the union

of foreground classes (Sec. 3.4). The thresholded support $\Omega^v_{\mathrm{fg}}(\theta) = \{u \in \Omega^v \mid P^v_{\mathrm{fg}}(u) > \theta\}$ is used only for structural evaluation (Appendix D.1); training operates on the soft maps.

Persistent component measures: For a filtration direction $\xi \in \{\mathrm{sub}, \mathrm{sup}\}$, each pair $c = (b_c, d_c)$ in the 0-dimensional cubical persistence diagram PD^ξ_0 of a foreground map generates a connected support $\Omega_c \subseteq \Omega^v$. We summarise c by $s(c) = [\mathbf{m}(c), a(c), \pi(c)]$: its soft centroid, its foreground-support mass, and its persistence $\pi(c) = d_c - b_c$ (Sec. 3.4). Writing $\{c^{v,\xi}_{P,i}\}_i$ and $\{c^{v,\xi}_{Y,j}\}_j$ for the components of P^v_{fg} and Y^v_{fg}, we represent their structure by the finite measures

$$\mu^{v,\xi} = \sum_i a(c^{v,\xi}_{P,i})\, \delta_{s(c^{v,\xi}_{P,i})}, \qquad \nu^{v,\xi} = \sum_j a(c^{v,\xi}_{Y,j})\, \delta_{s(c^{v,\xi}_{Y,j})}, \tag{1}$$

where δ_s is a Dirac mass at summary s. These measures record where persistent components lie, how much foreground support each carries, and how stable each is across filtration levels, independently of component ordering.

Structural discrepancy: Given the pairwise component cost $C^{v,\xi}$ (Eq. 6), we compare the two measures within each view by unbalanced optimal transport,

$$\mathcal{D}^{v,\xi} = \min_{T \geq 0} \langle T, C^{v,\xi} \rangle + \tau\, \mathrm{KL}\big(T\mathbf{1} \,\|\, \mu^{v,\xi}\big) + \tau\, \mathrm{KL}\big(T^\top \mathbf{1} \,\|\, \nu^{v,\xi}\big), \tag{2}$$

where T is the transport plan, KL denotes the generalised Kullback–Leibler divergence, and $\tau > 0$ controls marginal relaxation. Missed or spurious structures leave unmatched mass, while fragmentation incurs additional component-matching cost. Because $\mathcal{D}^{v,\xi}$ is computed separately in each view, it requires neither one-to-one object–component correspondence nor identical RA/RD topology. Structural recovery minimises $\mathbb{E}\Big[\mathcal{L}_{\mathrm{pix}} + \lambda_{\mathrm{sc}} \sum_{v \in \{\mathrm{RA},\mathrm{RD}\}} \sum_{\xi \in \{\mathrm{sub},\mathrm{sup}\}} \mathcal{D}^{v,\xi}\Big]$, where $\mathcal{L}_{\mathrm{pix}}$ denotes the pixel-level radar losses in Eq. 10.

From objective to method: The Structural Consistency Loss (Sec. 3.4) is a tractable surrogate of Eq. 2. It restricts each measure to its K most persistent components and adds entropic regularisation ε so that the transport can be solved by Sinkhorn iterations (Eq. 7). It also holds component supports and persistence values fixed in the backward pass. Persistence therefore determines which components are compared and biases which pairs are matched, while the unbalanced transport supplies the corrective signal: it pulls matched components towards annotated locations, adjusts their foreground-support mass, and penalises excess or missing mass through the relaxed marginals. We adopt this division because annotated masks are binary, so their component persistence is nearly constant and offers little target signal. Moreover, when several predicted fragments are transported to one annotated component, optimising their persistence against the annotation would stabilise each fragment rather than encourage them to merge. Since $\mathcal{D}^{v,\xi}$ requires annotations, it shapes the model only during training. DuoPerLay (Sec. 3.3), which remains active at inference, supplies the persistent structure of intermediate multi-view features, allowing the backbone to condition on structural evidence when no annotation is available (Fig. 2).

3.2 Topology-Guided Feature Conditioning

TopoRadar consists of three stages: multi-view radar encoding, topology-guided feature conditioning, and dense mask prediction. Following prior multi-view radar segmentation pipelines (Ouaknine et al., 2021a; Dalbah et al., 2024), RA, RD, and AD inputs are encoded into a fused radar representation $X \in \mathbb{R}^{C_X \times H \times W}$ with C_X feature channels. Rather than refining X directly, we condition it with persistent structural information so the downstream attention backbone operates on features informed by persistent component structure and complementary higher-order topology. This is important in sparse radar perception, where object evidence is often fragmented across neighbouring cells, missing in one view, or corrupted by clutter and multipath effects.

To this end, **DuoPerLay** maps X to a dual-direction topology descriptor $v^{\mathrm{dual}} \in \mathbb{R}^D$, whose construction is detailed in Section 3.3. We fuse this descriptor back into the encoded feature space with a gated fusion block (FB), $X^{\mathrm{topo}} = \mathrm{FB}(X, v^{\mathrm{dual}})$, defined in Eq. 3, where X^{topo} denotes the topology-enhanced radar representation. Building on differentiable cubical persistence representations

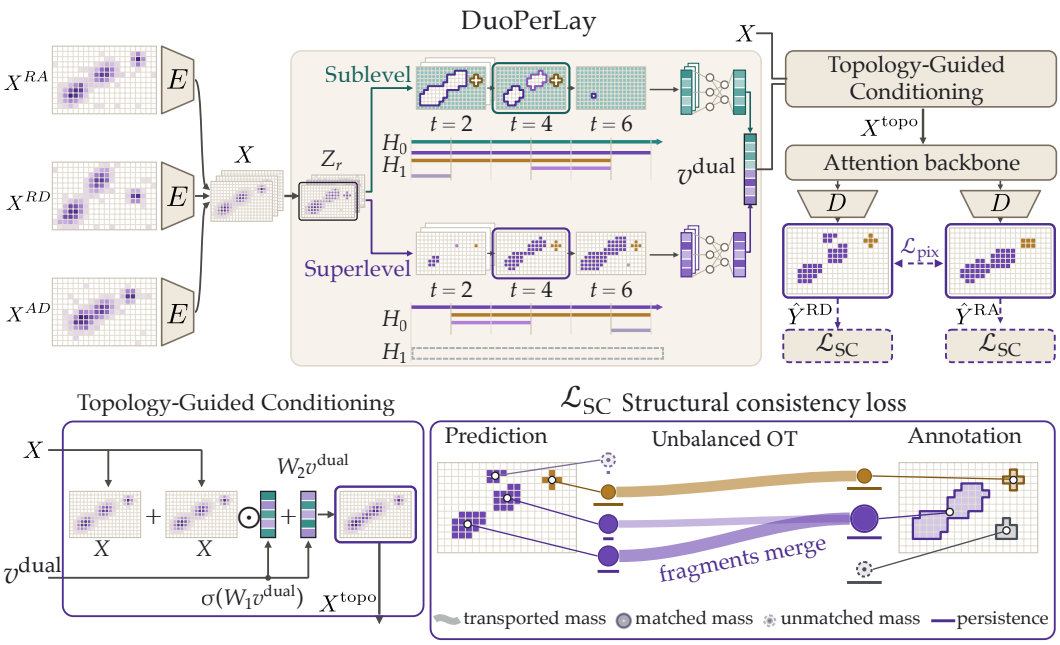


Figure 2: **Overview of TopoRadar.** Radar object evidence is sparse and view-dependent, so TopoRadar makes object shape explicit twice: a layer that measures which responses form stable connected objects (DuoPerLay) guides the features, and a loss that compares objects as whole shapes rather than pixels (Structural Consistency Loss) guides the output. Fused RA/RD/AD features are conditioned on DuoPerLay (Topology-Guided Conditioning, bottom left) and decoded into RD and RA masks, each trained with pixel losses and, per view, with the Structural Consistency Loss via UOT (bottom right).

(Korkmaz et al., 2025), we condition the radar feature tensor using

$$\mathrm{FB}(X, v^{\mathrm{dual}}) = X + X \odot \sigma(W_1 v^{\mathrm{dual}}) + W_2 v^{\mathrm{dual}} \tag{3}$$

where $W_1, W_2 \in \mathbb{R}^{C_X \times D}$ are learnable linear projections, $\sigma(\cdot)$ is the sigmoid function, and $\odot$ denotes channel-wise modulation with broadcasting over spatial dimensions. The gated modulation in Eq. 3 is related to gated linear unit style feature control (Dauphin et al., 2017). The three terms in Eq. 3 respectively preserve the encoded radar features, adaptively reweight channels using topology-aware gates, and inject an additive structural bias from v^{dual}.

The resulting tensor X^{topo} is passed to the adaptive attention backbone (Dalbah et al., 2024) for dense prediction of the RD and RA masks. In parallel, we apply the **Structural Consistency Loss** at the output level to align persistent foreground components between prediction and ground truth.

3.3 DuoPerLay

DuoPerLay learns and vectorises complementary sublevel and superlevel filtrations to produce a compact persistent topology descriptor. Given the encoded radar representation X, DuoPerLay first constructs a learnable multifiltration tensor $Z = [Z_1, \ldots, Z_M]^\top \in [0,1]^{M \times H_z \times W_z}$, where M is the number of learned filtration rows and $H_z \times W_z$ is the topology resolution. Each slice Z_r is produced by a learnable decoder with sigmoid activation and normalisation, following differentiable cubical multipersistence for grid-structured data (Korkmaz et al., 2025). Since RA, RD, and AD radar representations are grid-structured, cubical topology is a natural choice for extracting persistent structure from encoded radar features (Rieck et al., 2020; Bleile et al., 2022). From Z, we construct two complementary branches,

$$Z_r^{\mathrm{sub}} = \lfloor N Z_r \rfloor, \qquad Z_r^{\mathrm{sup}} = \lfloor N(1 - Z_r) \rfloor \tag{4}$$

where N is the number of discrete filtration levels. During backpropagation, the staircase discretisation in Eq. 4 is handled with a straight-through estimator (Bengio, 2013; Korkmaz et al., 2025).

The sublevel branch activates lower-valued responses earlier, whereas the superlevel branch activates higher-valued responses earlier, providing complementary structural views of the same feature tensor. By duality, on a 2D cubical grid the superlevel PD0 of Z_r and the sublevel PD1 describe closely related structures (Bleile et al., 2022), so the two branches are not independent.

For $\xi \in \{\mathrm{sub}, \mathrm{sup}\}$, each discretised slice induces a row-wise cubical filtration $\mathcal{K}^{\xi}_{r,t} = \{u \in \Omega_z \mid Z^{\xi}_r(u) \leq t-1\}$, $t = 1, \ldots, N+1$, where Ω_z denotes the $H_z \times W_z$ topology grid and $r = 1, \ldots, M$ indexes filtration rows. We compute cubical persistence in dimensions 0 and 1, denoted by $PD^{\xi}_{0,r} = PD_0(\mathcal{K}^{\xi}_{r,\cdot})$ and $PD^{\xi}_{1,r} = PD_1(\mathcal{K}^{\xi}_{r,\cdot})$, where PD_0 captures connected components and PD_1 records one-dimensional hole-like structures in the encoded feature map. The resulting diagrams are converted into learnable row-level vectors. For homology dimension $q \in \{0, 1\}$, we define

$$\Psi^{\xi}_q(PD^{\xi}_{q,r}) = \sum_{p \in PD^{\xi}_{q,r}} w^{\xi}_q(p)\, \phi^{\xi}_q(p) \tag{5}$$

where $p = (b, d)$ is a birth–death pair, $w^{\xi}_q(p)$ is a learnable persistence weight, and $\phi^{\xi}_q(p)$ is a learnable vector-valued point transform that maps each persistence pair to a fixed-dimensional feature representation. Following Korkmaz et al. (2025), $\phi^{\xi}_q(p)$ is implemented using a triangle-based point transform with learnable sampling locations, allowing the network to learn informative regions of the birth–death space.

We concatenate the two homology dimensions for each row as $v^{\xi}_r = [\Psi^{\xi}_0(PD^{\xi}_{0,r}); \Psi^{\xi}_1(PD^{\xi}_{1,r})]$, and aggregate row features with a lightweight MLP, $v^{\xi} = \rho([v^{\xi}_1, \ldots, v^{\xi}_M])$, for $\xi \in \{\mathrm{sub}, \mathrm{sup}\}$. This yields v^{sub} and v^{sup}, which are concatenated into the dual-direction topology descriptor $v^{\mathrm{dual}} = [v^{\mathrm{sub}}; v^{\mathrm{sup}}]$ used in Eq. 3. Thus, DuoPerLay learns, vectorises, and aggregates complementary filtration directions to produce a compact structural descriptor for encoded radar features. For completeness, Appendix B provides a brief intuitive summary of the topological concepts used in TopoRadar, including cubical filtrations, persistence diagrams, and the roles of PD_0 and PD_1.

3.4 Structural Consistency and Optimisation Objective

To complement pixel-level radar losses, we introduce a **Structural Consistency Loss** that aligns persistent foreground components between prediction and ground truth. For each view $v \in \{\mathrm{RA}, \mathrm{RD}\}$, we form foreground-support maps as $P^{v}_{\mathrm{fg}} = 1 - \hat{Y}^{v,\mathrm{bg}}$ and $Y^{v}_{\mathrm{fg}} = 1 - Y^{v,\mathrm{bg}}$, where bg denotes the background channel; under multiclass normalisation these correspond to the union of foreground classes.

The superlevel branch captures high-foreground regions, while the sublevel branch provides a complementary ordering; using both, we compute 0-dimensional persistence and retain up to the K most persistent connected components from prediction and ground truth, denoted by $\mathcal{C}^{v,\xi}_P = \{c^{v,\xi}_{P,i}\}_{i=1}^{K_P}$ and $\mathcal{C}^{v,\xi}_Y = \{c^{v,\xi}_{Y,j}\}_{j=1}^{K_Y}$, where $\xi \in \{\mathrm{sub}, \mathrm{sup}\}$ and $K_P, K_Y \leq K$. Each retained component is summarised by $s(c) = [\mathbf{m}(c), a(c), \pi(c)]$ (Sec. 3.1). For a component support Ω_c with foreground weights $w(u)$, given by $P^{v}_{\mathrm{fg}}(u)$ for predicted and $Y^{v}_{\mathrm{fg}}(u)$ for annotated components, we compute $\mathbf{m}(c) = \sum_{u \in \Omega_c} w(u)\, \mathbf{u} \,/\, \big(\sum_{u \in \Omega_c} w(u) + \kappa\big)$ and $a(c) = \sum_{u \in \Omega_c} w(u)$, where $\mathbf{u} \in \mathbb{R}^2$ denotes the grid coordinates of cell u and $\kappa > 0$ is a small numerical stabilisation constant.

Gradient Flow: Persistent-component extraction and top-K ranking are discrete operations, so selected component identities, supports, and persistence values $\pi(c)$ are held fixed during backpropagation. For each selected predicted support Ω_c, the soft centroid $\mathbf{m}(c)$ and foreground-support mass $a(c)$ remain differentiable with respect to the predicted foreground probabilities. The entropically regularised UOT objective is differentiable through the resulting component costs and transport plan. Thus, $\mathcal{L}_{\mathrm{SC}}$ propagates gradients through the soft component summaries and transport objective, but not through component extraction or top-K ordering (Sec. 3.1).

Given a predicted component $c^{v,\xi}_{P,i}$ and a ground-truth component $c^{v,\xi}_{Y,j}$, we define the pairwise structural cost

$$C^{v,\xi}_{ij} = \alpha \left\| \mathbf{m}(c^{v,\xi}_{P,i}) - \mathbf{m}(c^{v,\xi}_{Y,j}) \right\|_1 + \beta \left| a(c^{v,\xi}_{P,i}) - a(c^{v,\xi}_{Y,j}) \right| + \gamma \left| \pi(c^{v,\xi}_{P,i}) - \pi(c^{v,\xi}_{Y,j}) \right| \tag{6}$$

where $\alpha, \beta, \gamma \geq 0$ control the relative contributions of component location, foreground-support mass, and persistence, respectively. As motivated in Sec. 3.1, we use entropic *unbalanced* optimal transport for component alignment under unequal and partially matched supports (Cuturi, 2013; Peyré & Cuturi, 2019; Séjourné et al., 2023). Restricting the measures in Eq. 1 to the retained components gives the marginals $\mu^{v,\xi} \in \mathbb{R}_+^{K_P}$ and $\nu^{v,\xi} \in \mathbb{R}_+^{K_Y}$, whose entries are the normalised foreground-support masses of the retained components plus the constant κ. The entropic counterpart of Eq. 2 is

$$\mathcal{T}^{v,\xi} = \min_{T \geq 0} \langle T, C^{v,\xi} \rangle + \varepsilon \sum_{i,j} T_{ij}(\log T_{ij} - 1) + \tau \,\mathrm{KL}(T\mathbf{1} \,\|\, \mu^{v,\xi}) + \tau \,\mathrm{KL}(T^\top \mathbf{1} \,\|\, \nu^{v,\xi}) \tag{7}$$

where T is the transport plan, $\varepsilon > 0$ is the entropic regularisation parameter, and $\tau > 0$ controls marginal relaxation as in Eq. 2. The Structural Consistency Loss is

$$\mathcal{L}_{\mathrm{SC}} = \sum_{v \in \{\mathrm{RA},\mathrm{RD}\}} \sum_{\xi \in \{\mathrm{sub},\mathrm{sup}\}} \mathcal{T}^{v,\xi} \tag{8}$$

Implementation details of the UOT solver are provided in Appendix C. The objective combines radar segmentation losses, structural consistency, and filtration regularisation. Let $\mathcal{L}_{\mathrm{CA}}$, $\mathcal{L}_{\mathrm{SD}}$, and $\mathcal{L}_{\mathrm{MV}}$ denote class-agnostic object loss, soft Dice, and multi-view consistency losses (Dalbah et al., 2024). Let $\mathcal{L}_{\mathrm{reg}}^{\mathrm{sub}}$, $\mathcal{L}_{\mathrm{reg}}^{\mathrm{sup}}$, $\mathcal{L}_{\mathrm{LE}}^{\mathrm{sub}}$, and $\mathcal{L}_{\mathrm{LE}}^{\mathrm{sup}}$ denote directional filtration regularisers (Korkmaz et al., 2025), combined as

$$\mathcal{L}_{\mathrm{reg}} = \mathcal{L}_{\mathrm{reg}}^{\mathrm{sub}} + \mathcal{L}_{\mathrm{reg}}^{\mathrm{sup}}, \qquad \mathcal{L}_{\mathrm{LE}} = \mathcal{L}_{\mathrm{LE}}^{\mathrm{sub}} + \mathcal{L}_{\mathrm{LE}}^{\mathrm{sup}} \tag{9}$$

The overall training objective is

$$\mathcal{L}_{\mathrm{total}} = \alpha_1 \mathcal{L}_{\mathrm{CA}} + \alpha_2 \mathcal{L}_{\mathrm{SD}} + \alpha_3 \mathcal{L}_{\mathrm{MV}} + \lambda_{\mathrm{sc}} \mathcal{L}_{\mathrm{SC}} + \beta_1 \mathcal{L}_{\mathrm{reg}} + \beta_2 \mathcal{L}_{\mathrm{LE}} \tag{10}$$

where $\alpha_1, \alpha_2, \alpha_3, \lambda_{\mathrm{sc}}, \beta_1$, and β_2 are scalar coefficients. Eq. 10 preserves radar-specific dense supervision while adding component-level structural alignment and regularising the topology branch.

4 Experiments and Results

Datasets and Evaluation Metrics: We evaluate TopoRadar on CARRADA (Ouaknine et al., 2021b) and RADIal (Rebut et al., 2022). CARRADA serves as the multi-class RD/RA segmentation benchmark using RA, RD, and AD projections, whereas RADIal provides a complementary evaluation under a different radar representation and annotation protocol. In addition to **segmentation metrics**, we report topology-oriented **structural metrics** that quantify connected-component error, fragmentation, spurious foreground components, and persistent component matching cost. Additional dataset, preprocessing, and experimental details are provided in Appendix C.

Baselines and Implementation: We compare TopoRadar against recent radar segmentation methods. On CARRADA, the comparison includes TMVA-Net (Ouaknine et al., 2021a), PeakConv (Zhang et al., 2023), TransRadar (Dalbah et al., 2024), AdaPKC (Li et al., 2024), MARSS (Chen et al., 2026), and HyperRadar (Zia et al., 2026c). On RADIal, we follow the benchmark comparison with Pixor (Yang et al., 2018), FFTRadNet (Rebut et al., 2022), C-M DNN (Jin et al., 2023), TransRadar (Dalbah et al., 2024), and HyperRadar (Zia et al., 2026c). TopoRadar is built on a multi-view radar segmentation backbone, with DuoPerLay inserted after multi-view feature encoding and before the adaptive-attention refinement stage. For CARRADA, we follow the same evaluation protocol and metric aggregation convention as TransRadar (Dalbah et al., 2024). Baseline results for methods included in the TransRadar comparison are quoted from (Dalbah et al., 2024); results for later methods are quoted from their original publications. For controlled ablations, we re-train the TransRadar backbone under our implementation and training protocol; therefore, the ablation baseline in Table 3 should not be interpreted as the published TransRadar result in Table 1. The Structural Consistency Loss is applied to the predicted foreground-support masks. Training and implementation details are provided in Appendix C, with computational cost reported in Appendix E.

Quantitative Results: Tables 1 and 2 summarise the results on CARRADA and RADIal. On CARRADA, TopoRadar achieves the highest RD mIoU of **64.87**%, exceeding TransRadar, HyperRadar, and MARSS by **0.97**, **1.07**, and **1.61** points, respectively, with the highest mDice of **76.14**%. It

Table 1: Semantic segmentation performance on CARRADA for RD and RA views. We report class-wise IoU and Dice scores together with their means. Best results are shown in **bold**, and second-best results are shown in **blue bold**.

View	Method	IoU (%)					Dice (%)				
		Bkg.	Ped.	Cycl.	Car	mIoU	Bkg.	Ped.	Cycl.	Car	mDice
RD	MVNet (Ouaknine et al., 2021a)	98.0	0.0	3.8	14.1	29.0	99.0	0.0	7.3	24.8	32.8
	TMVA-Net (Ouaknine et al., 2021a)	99.7	52.6	29.0	53.4	58.7	**99.8**	68.9	45.0	69.6	70.9
	PeakConv (Zhang et al., 2023)	–	–	–	–	60.7	–	–	–	–	72.5
	TransRadar (Dalbah et al., 2024)	**99.9**	57.7	**36.1**	61.9	**63.9**	**99.9**	73.2	**53.1**	76.5	**75.6**
	AdaPKC (Li et al., 2024)	99.7	56.3	32.8	59.6	62.1	**99.9**	72.0	49.4	74.7	74.0
	MARSS (Chen et al., 2026)	–	–	–	–	63.26	–	–	–	–	75.18
	HyperRadar (Zia et al., 2026c)	99.7	**59.2**	32.4	**63.8**	63.8	**99.8**	**74.5**	48.9	**77.6**	75.2
	TopoRadar (DuoPerLay + SC)	**99.80**	**59.80**	**33.10**	**66.80**	**64.87**	**99.90**	**74.84**	**49.74**	**80.10**	**76.14**
RA	MVNet (Ouaknine et al., 2021a)	98.8	0.1	1.1	6.2	26.8	99.0	0.0	7.3	24.8	28.5
	TMVA-Net (Ouaknine et al., 2021a)	**99.8**	26.0	8.6	30.7	41.3	**99.9**	41.3	15.9	47.0	51.0
	PeakConv (Zhang et al., 2023)	–	–	–	–	42.9	–	–	–	–	53.3
	TransRadar (Dalbah et al., 2024)	**99.9**	30.3	**21.5**	**38.2**	**47.5**	**99.9**	46.6	**35.3**	**55.3**	**59.3**
	AdaPKC (Li et al., 2024)	**99.8**	26.2	**19.2**	32.0	44.3	**99.9**	41.5	**32.2**	48.4	55.5
	MARSS (Chen et al., 2026)	–	–	–	–	**46.97**	–	–	–	–	**58.78**
	HyperRadar (Zia et al., 2026c)	**99.9**	**31.8**	8.4	37.5	44.4	**99.9**	**48.2**	14.6	54.4	54.3
	TopoRadar (DuoPerLay + SC)	**99.90**	**31.60**	9.50	**40.04**	45.26	**99.95**	**48.02**	17.35	**57.18**	55.63

Table 2: RADIal results for joint detection and segmentation. AP/AR denote average precision/recall, while R/A denote range/angle errors. Best results are in **bold**, second-best in **blue bold**.

Backbone	AP (%) ↑	AR (%) ↑	R (m) ↓	A (°) ↓	mIoU (%) ↑
Pixor (Yang et al., 2018)	96.6	81.7	**0.10**	0.20	–
FFTRadNet (Rebut et al., 2022)	96.8	82.2	**0.11**	0.17	74.0
C-M DNN (Jin et al., 2023)	96.9	83.5	–	–	80.4
TransRadar (Dalbah et al., 2024)	97.3	98.4	**0.11**	**0.10**	81.1
HyperRadar (Zia et al., 2026c)	**98.5**	**98.9**	**0.10**	**0.08**	**83.4**
TopoRadar	**98.27**	**98.67**	**0.10**	**0.08**	**83.17**

also improves RD pedestrian/car IoU over TransRadar to 59.8/66.8%, while cyclist IoU is lower. In RA, TopoRadar obtains 45.26% mIoU and 55.63% mDice, below TransRadar and MARSS but above HyperRadar overall. This gap is largely driven by cyclists, lowering mIoU by 3.0 points on its own. Low RA cyclist IoU is shared by other methods (8.6 for TMVA-Net, 8.4 for HyperRadar), reflecting weak, Doppler-spread returns that occupy few RA cells and resemble pedestrians; since the Structural Consistency Loss acts on the class-agnostic foreground, it targets connectivity rather than class separation. Beyond pixel accuracy, TopoRadar reduces connected-component error by approximately **30%** and spurious-component counts by approximately **35%** across both RD and RA relative to the re-trained TransRadar backbone, together with lower fragmentation; detailed **structural evaluations** are reported in Appendix D.1. On RADIal, TopoRadar achieves 83.17% mIoU, improving TransRadar by **2.07** points and remaining close to HyperRadar (83.4%), while matching its range and angle errors.

Ablation Studies: Table 3 evaluates component contribution, filtration direction, and structural matching on CARRADA. DuoPerLay and the Structural Consistency Loss each improve the re-trained backbone, while their combination performs best at **64.87**% RD and **45.26**% RA mIoU, supporting their complementarity. Dual-direction filtration outperforms either direction alone, and unbalanced OT gives the strongest matching results, consistent with sparse radar masks containing unequal or fragmented foreground components. Further loss, component-selection, and persistence-specific ablations are provided in Appendices D.2–D.4.

Qualitative Results: Figure 3 shows that TopoRadar more consistently preserves compact, connected foreground support across RD and RA views, whereas competing methods more often fragment object responses, miss foreground structure, or introduce isolated components. These observations are consistent with the intended effect of persistence-aware feature conditioning and component-level structural supervision, and agree with the lower fragmentation and spurious-component errors reported in Appendix D.1. Additional examples and failure cases are provided in Appendix F.

Table 3: **Ablation Study on CARRADA.** We report mIoU and mDice for RD and RA views. The table evaluates component contribution, filtration direction, and structural matching strategy. Best results in each block are shown in **bold**.

Ablation	Variant	RD mIoU	RD mDice	RA mIoU	RA mDice
Component	TransRadar backbone (re-trained)	62.10	73.40	43.10	52.60
	Backbone + DuoPerLay	63.48	74.31	44.18	54.27
	Backbone + Structural Consistency Loss	63.21	74.06	43.86	53.91
	Backbone + DuoPerLay + Structural Consistency Loss	**64.87**	**76.14**	**45.26**	**55.63**
Filtration	Sublevel only	63.72	74.42	44.31	54.63
	Superlevel only	63.58	74.25	44.05	54.38
	Dual-direction	**64.87**	**76.14**	**45.26**	**55.63**
Matching	No structural matching	63.48	74.31	44.18	54.27
	Direct component matching	63.76	74.53	44.46	54.81
	Balanced OT	64.19	74.87	44.79	55.34
	Unbalanced OT	**64.87**	**76.14**	**45.26**	**55.63**

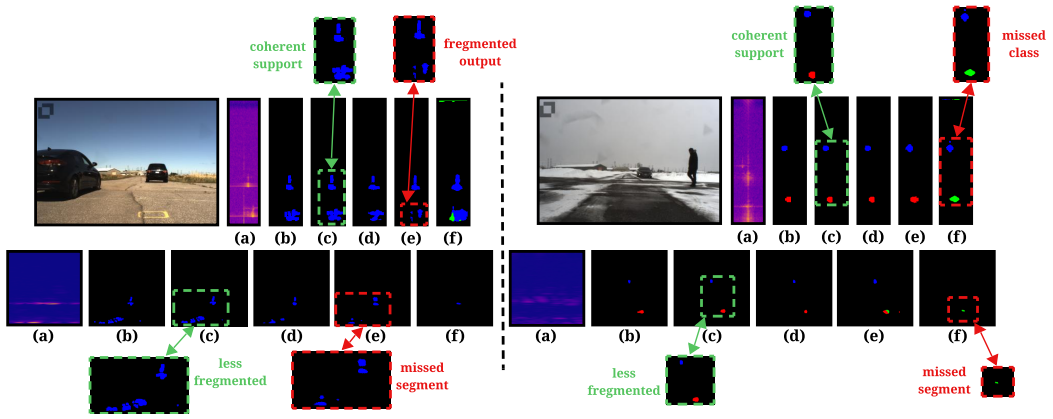


Figure 3: **Qualitative comparison on CARRADA.** Two representative test scenes are shown with RD (top) and RA (bottom) outputs. Columns denote: (a) radar input, (b) ground truth, (c) **TopoRadar**, (d) TransRadar, (e) TMVA-Net, and (f) MVNet. All RD outputs are rotated for visual consistency. Semantic classes are car, cyclist, pedestrian, and black background. Green and red callouts highlight selected structurally coherent and erroneous regions, respectively.

5 Conclusion

We introduced **TopoRadar**, to our knowledge among the first works to treat sparse multi-view radar segmentation as persistent topological structural recovery. Our results indicate that persistence is useful for both representation and supervision i.e., conditioning features on learned persistence and supervising persistent components with unbalanced optimal transport reduce connected-component and spurious-component errors by about 30% and 35% relative to the re-trained backbone. Relaxed mass constraints make this supervision workable, tolerating the unequal, fragmented component sets of radar masks where direct and balanced matching fall short. Although RA performance trails TransRadar and MARSS, mainly because sparse cyclist returns are hard to separate from pedestrians, TopoRadar achieves the highest RD mIoU on CARRADA, outperforms HyperRadar in both views, and improves RADIal mIoU over TransRadar, establishing topological structure as a practical signal for radar perception.

Limitations and Future Directions: TopoRadar remains constrained by weak returns, severe clutter, multipath, and the resolution limits of grid-based radar projections. The Structural Consistency Loss also depends on top-K component selection and transport parameters (ε, τ). Future work will investigate adaptive component selection, uncertainty-aware structural matching, cross-sensor evaluation, and topology-aware attention.

Reproducibility Statement: Implementation details, training settings, loss definitions, evaluation protocols, and hyperparameters are provided in Appendix C. The implementation and evaluation code will be released to support reproducibility.

AI Use Disclosure: Generative AI tools were used during manuscript preparation to assist with language editing, presentation, and methodological discussion and feedback. The authors reviewed and verified all technical decisions, experimental design, implementation, results, analyses, and conclusions and take full responsibility for the paper's content.

REFERENCES

Dashti Ali, Aras Asaad, Maria-Jose Jimenez, Vidit Nanda, Eduardo Paluzo-Hidalgo, and Manuel Soriano-Trigueros. A survey of vectorization methods in topological data analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):14069–14080, 2023.

Kshitiz Bansal, Keshav Rungta, and Dinesh Bharadia. Radsegnet: A reliable approach to radar camera fusion. *arXiv preprint arXiv:2208.03849*, 2022.

Yoshua Bengio. Estimating or propagating gradients through stochastic neurons. *arXiv preprint arXiv:1305.2982*, 2013.

Bea Bleile, Adélie Garin, Teresa Heiss, Kelly Maggs, and Vanessa Robins. The persistent homology of dual digital image constructions. In *Research in Computational Topology 2*, pp. 1–26. Springer, 2022.

Fengyu Chen, Tiao Tan, Teng Li, Yuantian Quan, and Qingmin Liao. Marss: Radar semantic segmentation via modular attention and state space models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17959–17968, 2026.

James R Clough, Nicholas Byrne, Ilkay Oksuz, Veronika A Zimmer, Julia A Schnabel, and Andrew P King. A topological loss function for deep-learning based image segmentation using persistent homology. *IEEE transactions on pattern analysis and machine intelligence*, 44(12):8766–8778, 2020.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

Lorin Crawford, Anthea Monod, Andrew X Chen, Sayan Mukherjee, and Raúl Rabadán. Predicting clinical outcomes in glioblastoma: an application of topological and functional data analysis. *Journal of the American Statistical Association*, 115(531):1139–1150, 2020.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Yahia Dalbah, Jean Lahoud, and Hisham Cholakkal. Transradar: Adaptive-directional transformer for real-time multi-view radar semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 353–362, 2024.

Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *International conference on machine learning*, pp. 933–941. PMLR, 2017.

Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020.

Kathryn Garside, Robin Henderson, Irina Makarenko, and Cristina Masoller. Topological data analysis of high resolution diabetic retinopathy images. *PloS one*, 14(5):e0217413, 2019.

Xiaoling Hu, Fuxin Li, Dimitris Samaras, and Chao Chen. Topology-preserving deep image segmentation. *Advances in neural information processing systems*, 32, 2019.

Tiezhen Jiang, Long Zhuang, Qi An, Jianhua Wang, Kai Xiao, and Anqi Wang. T-rodnet: Transformer for vehicular millimeter-wave radar object detection. *IEEE Transactions on Instrumentation and Measurement*, 72:1–12, 2022.

Yi Jin, Anastasios Deligiannis, Juan-Carlos Fuentes-Michel, and Martin Vossiek. Cross-modal supervision-based multitask learning with automotive radar raw data. *IEEE Transactions on Intelligent Vehicles*, 8(4):3012–3025, 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Caner Korkmaz, Brighton Nuwagira, Baris Coskunuzer, and Tolga Birdal. Cumperlay: Learning cubical multiparameter persistence vectorizations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27084–27094, 2025.

Siru Li, Ziyang Hong, Yushuai Chen, Liang Hu, and Jiahu Qin. Get it for free: Radar segmentation without expert labels and its application in odometry and localization. *IEEE Robotics and Automation Letters*, 2025.

Teng Li, Liwen Zhang, Youcheng Zhang, Zijun Hu, Pengcheng Pi, Zongqing Lu, Qingmin Liao, and Zhe Ma. Adapkc: Peakconv with adaptive peak receptive field for radar semantic segmentation. 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. 136545–136575. Curran Associates, Inc., 2024. doi: 10.52202/079017-4338. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f6b22ac37beb5da61efd4882082c9ecd-Paper-Conference.pdf.

Arthur Ouaknine, Alasdair Newson, Patrick Pérez, Florence Tupin, and Julien Rebut. Multi-view radar semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15671–15680, 2021a.

Arthur Ouaknine, Alasdair Newson, Julien Rebut, Florence Tupin, and Patrick Pérez. Carrada dataset: Camera and automotive radar with range-angle-doppler annotations. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 5068–5075. IEEE, 2021b.

Andras Palffy, Jiaao Dong, Julian FP Kooij, and Dariu M Gavrila. Cnn based road user detection using the 3d radar cube. *IEEE Robotics and Automation Letters*, 5(2):1263–1270, 2020.

Gabriel Peyré and Marco Cuturi. *Computational optimal transport: With applications to data science*. Now Foundations and Trends, 2019.

Julien Rebut, Arthur Ouaknine, Waqas Malik, and Patrick Pérez. Raw high-definition radar for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17021–17030, 2022.

Bastian Rieck, Tristan Yates, Christian Bock, Karsten Borgwardt, Guy Wolf, Nicholas Turk-Browne, and Smita Krishnaswamy. Uncovering the topology of time-varying fmri data using cubical persistence. *Advances in neural information processing systems*, 33:6900–6912, 2020.

Ole Schumann, Markus Hahn, Nicolas Scheiner, Fabio Weishaupt, Julius F Tilly, Jürgen Dickmann, and Christian Wöhler. Radarscenes: A real-world radar point cloud data set for automotive applications. In *2021 IEEE 24th International Conference on Information Fusion (FUSION)*, pp. 1–8. IEEE, 2021.

Thibault Séjourné, Gabriel Peyré, and François-Xavier Vialard. Unbalanced optimal transport, from theory to numerics. *Handbook of Numerical Analysis*, 24:407–471, 2023.

Nico Stucki, Johannes C Paetzold, Suprosanna Shit, Bjoern Menze, and Ulrich Bauer. Topologically faithful image segmentation via induced matching of persistence barcodes. In *International Conference on Machine Learning*, pp. 32698–32727. PMLR, 2023.

Zhe Su, Xiang Liu, Layal Bou Hamdan, Vasileios Maroulas, Jie Wu, Gunnar Carlsson, and Guo-Wei Wei. Topological data analysis and topological deep learning beyond persistent homology: a review. *Artificial Intelligence Review*, 2025.

Yizhou Wang, Zhongyu Jiang, Xiangyu Gao, Jenq-Neng Hwang, Guanbin Xing, and Hui Liu. Rodnet: Radar object detection using cross-modal supervision. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 504–513, 2021.

Bo Wen, Haochen Zhang, Dirk-Uwe G Bartsch, William Freeman, Truong Nguyen, and Cheolhong An. Topology-preserving image segmentation with spatial-aware persistent feature matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5821–5830, 2025.

Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7652–7660, 2018.

Shanliang Yao, Runwei Guan, Xiaoyu Huang, Zhuoxiao Li, Xiangyu Sha, Yong Yue, Eng Gee Lim, Hyungjoon Seo, Ka Lok Man, Xiaohui Zhu, et al. Radar-camera fusion for object detection and semantic segmentation in autonomous driving: A comprehensive review. *IEEE Transactions on Intelligent Vehicles*, 9(1):2094–2128, 2023.

Ao Zhang, Farzan Erlik Nowruzi, and Robert Laganiere. Raddet: Range-azimuth-doppler based radar object detection for dynamic road users. In *2021 18th Conference on Robots and Vision (CRV)*, pp. 95–102. IEEE, 2021.

Liwen Zhang, Xinyan Zhang, Youcheng Zhang, Yufei Guo, Yuanpei Chen, Xuhui Huang, and Zhe Ma. Peakconv: Learning peak receptive field for radar semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17577–17586, 2023.

Ali Zia, Abdelwahed Khamis, James Nichols, Usman Bashir Tayab, Zeeshan Hayder, Vivien Rolland, Eric Stone, and Lars Petersson. Topological deep learning: a review of an emerging paradigm. *Artificial Intelligence Review*, 57(4):77, 2024.

Ali Zia, Usman Ali, Umer Ramzan, Abdul Rehman, Abdelwahed Khamis, and Wei Xiang. Test-time adaptation for anomaly segmentation via topology-aware optimal transport chaining. *arXiv preprint arXiv:2601.20333*, 2026a.

Ali Zia, Usman Ali, Abdul Rehman, Umer Ramzan, Kang Han, Muhammad Faheem, Shahnawaz Qureshi, and Wei Xiang. Learning topology-aware representations via test-time adaptation for anomaly segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026b.

Ali Zia, Muhammad Umer Ramzan, Abdelwahed Khamis, Usman Ali, and Abdul Rehman. Learning structurally consistent representations for multi-view radar semantic segmentation. *arXiv preprint arXiv:2606.31609*, 2026c.

A Technical Appendices and Supplementary Material

This appendix provides supplementary material for TopoRadar, including the topological background, experimental details, additional quantitative analyses, runtime and memory measurements, qualitative examples, and failure cases. The main paper is self-contained; these appendices provide additional details for reproducibility and extended analysis.

Appendix B: Topological Background for TopoRadar. This section summarises the cubical filtration, sublevel/superlevel persistence, persistent homology, persistence vectorisation, and structural consistency concepts used in TopoRadar. It clarifies how persistent topology is instantiated for grid-based RA, RD, and AD radar representations.

Appendix C: Experimental Details. This section provides dataset descriptions, evaluation metrics, training settings, and TopoRadar implementation details. It includes information on CARRADA and RADIal, the reported segmentation and topology-oriented metrics, optimiser settings, batch size, hardware, and topology-specific hyperparameters.

Appendix D: Additional Quantitative Results. This appendix reports structural quality metrics and additional controlled analyses, including loss-function, top-K component-selection, and persistence-specific ablations.

Appendix E: Runtime and Memory Overhead. This section compares TopoRadar and TransRadar in terms of GFLOPs, inference throughput, and peak GPU memory.

Appendix F: Additional Qualitative Results and Failure Cases. This section provides additional qualitative comparisons on CARRADA and discusses representative failure cases. The examples highlight improved foreground coherence under sparse radar observations, while also identifying limitations under extremely weak returns, clutter, multipath, and closely spaced objects.

B Topological Background for TopoRadar

This section summarises the topological concepts used in TopoRadar. The main paper is self-contained; this appendix clarifies how persistent topology is instantiated for grid-based radar representations.

Cubical Filtrations for Radar Maps: RA, RD, and AD radar representations are naturally defined on regular grids. Let $\Omega \subset \mathbb{Z}^2$ denote the spatial grid of a radar view or feature map, and let

$$f : \Omega \to \mathbb{R}$$

be a scalar filtration function. In cubical topology, grid cells and their faces form a cubical complex $\mathcal{K}$. A filtration is a nested sequence of cubical subcomplexes,

$$\mathcal{K}_1 \subseteq \mathcal{K}_2 \subseteq \cdots \subseteq \mathcal{K}_N \tag{11}$$

where cells are activated progressively according to f. In TopoRadar, the filtration functions are not hand-crafted from raw radar intensity; instead, they are learned from encoded radar features.

Sublevel and Superlevel Filtrations: DuoPerLay uses two complementary filtration directions. Given a learned response map $Z_r \in [0,1]^{H_z \times W_z}$ for filtration row r, the sublevel and superlevel discretisations are

$$Z_r^{\mathrm{sub}} = \lfloor N Z_r \rfloor, \qquad Z_r^{\mathrm{sup}} = \lfloor N(1 - Z_r) \rfloor \tag{12}$$

where N controls the filtration discretisation. For $\xi \in \{\mathrm{sub}, \mathrm{sup}\}$, the induced cubical filtration is

$$\mathcal{K}_{r,t}^{\xi} = \{u \in \Omega_z \mid Z_r^{\xi}(u) \leq t - 1\}, \qquad t = 1, \ldots, N+1 \tag{13}$$

The sublevel branch activates lower-valued responses earlier, while the superlevel branch activates higher-valued responses earlier. The sublevel and superlevel branches therefore provide complementary structural orderings of the learned representation, characterising the organisation of lower- and higher-valued regions without assigning a direct physical interpretation to feature magnitude.

Persistent Homology: Persistent homology tracks topological structures across the filtration in Eq. 11. For each filtration direction ξ and row r, TopoRadar computes persistence diagrams

$$PD^{\xi}_{q,r} = PD_q(\mathcal{K}^{\xi}_{r,\cdot}), \qquad q \in \{0, 1\} \tag{14}$$

where PD_0 records connected components and PD_1 records holes. Each topological feature is represented by a birth–death pair

$$p = (b, d) \in PD^{\xi}_{q,r}$$

where b is the filtration level at which the feature appears and d is the level at which it disappears or merges. Its persistence is

$$\pi(p) = d - b \tag{15}$$

Larger persistence indicates a more stable structure across filtration levels, whereas low-persistence features are more likely to correspond to noise or isolated clutter.

Persistence Vectorisation: Persistence diagrams contain variable numbers of birth–death pairs, so they are converted into fixed-dimensional vectors before being used inside the neural network. For homology dimension $q \in \{0, 1\}$, DuoPerLay vectorises a diagram as

$$\Psi^{\xi}_q(PD^{\xi}_{q,r}) = \sum_{p \in PD^{\xi}_{q,r}} w^{\xi}_q(p)\, \phi^{\xi}_q(p) \tag{16}$$

where $w^{\xi}_q(p)$ is a learnable persistence weight and $\phi^{\xi}_q(p)$ is a learnable point transform. The two homology dimensions are concatenated to obtain a row-level descriptor,

$$v^{\xi}_r = \left[\Psi^{\xi}_0(PD^{\xi}_{0,r}); \Psi^{\xi}_1(PD^{\xi}_{1,r})\right] \tag{17}$$

and row descriptors are aggregated as

$$v^{\xi} = \rho\Big([v^{\xi}_1, \ldots, v^{\xi}_M]\Big) \qquad \xi \in \{\mathrm{sub}, \mathrm{sup}\} \tag{18}$$

where $\rho(\cdot)$ is a lightweight learnable aggregator. The final dual topology descriptor is

$$v^{\mathrm{dual}} = [v^{\mathrm{sub}}; v^{\mathrm{sup}}] \tag{19}$$

which is used to condition the radar feature tensor before adaptive attention refinement.

Structural Consistency: The Structural Consistency Loss applies the same topological view at the output level. For each prediction and ground-truth mask, TopoRadar first forms foreground-support maps,

$$P^{v}_{\mathrm{fg}} = 1 - \hat{Y}^{v,\mathrm{bg}}, \qquad Y^{v}_{\mathrm{fg}} = 1 - Y^{v,\mathrm{bg}} \tag{20}$$

Persistent connected components are then extracted from these maps. For a retained component c, TopoRadar uses the structural summary

$$s(c) = \left[\mathbf{m}(c), a(c), \pi(c)\right] \tag{21}$$

where $\mathbf{m}(c) \in \mathbb{R}^2$ is the soft centroid, $a(c)$ is the foreground-support mass, and $\pi(c)$ is its persistence. For component support Ω_c and foreground weight $w(u)$, these quantities are computed as

$$\mathbf{m}(c) = \frac{\sum_{u \in \Omega_c} w(u)\mathbf{u}}{\sum_{u \in \Omega_c} w(u) + \kappa}, \qquad a(c) = \sum_{u \in \Omega_c} w(u), \qquad \pi(c) = d_c - b_c \tag{22}$$

where (b_c, d_c) is the birth–death pair of component c, $\mathbf{u} \in \mathbb{R}^2$ denotes the grid coordinates of cell u, and $\kappa > 0$ is a small constant for numerical stability. These summaries are aligned between prediction and ground truth using unbalanced optimal transport, encouraging predicted masks to preserve object-level foreground structure while remaining tolerant to sparse, fragmented, and unequal radar responses.

C EXPERIMENTAL DETAILS

Datasets: We evaluate TopoRadar on two public automotive radar benchmarks: CARRADA (Ouaknine et al., 2021b) and RADIal (Rebut et al., 2022). CARRADA contains synchronised radar–camera recordings from urban driving scenes with 12,666 annotated frames. Labels are generated semi-automatically and provided directly in the radar domain for the range–angle (RA) and range–Doppler (RD) views. The dataset contains four classes: *pedestrian*, *cyclist*, *car*, and *background*. Radar observations are represented using RA, RD, and angle–Doppler (AD) projections derived from the original RAD tensor, with RA maps of size $1 \times 256 \times 256$ and RD/AD maps of size $1 \times 256 \times 64$. Following prior work, we use the three projections jointly to preserve complementary spatial and kinematic information while keeping computation tractable.

RADIal contains 8,252 annotated automotive radar frames and provides a different sensing and annotation protocol from CARRADA. Unlike CARRADA, it does not provide the same RA/RD/AD multi-view setting and instead uses a single radar representation with resolution $32 \times 512 \times 256$. Its annotations are obtained by projecting labels from aligned RGB images into the radar domain. RADIal contains two semantic categories, *free space* and *occupied space* corresponding to vehicles, and is used for both segmentation and detection evaluation. We report free-space segmentation mIoU alongside detection metrics. RADIal is used to assess whether the topology-aware design remains beneficial under a different radar representation; multi-view evaluation is performed on CARRADA.

Evaluation Metrics: We follow standard radar semantic segmentation protocols and report mean intersection-over-union (mIoU) as the primary metric, together with Dice (F1) and class-wise IoU/Dice. Topology-oriented metrics, including connected-component error, fragmentation rate, spurious component count, and persistent component matching cost, are reported in Appendix D.1.

Training Protocol: All models are implemented in `PyTorch` and trained on a single NVIDIA RTX 5090 GPU. On CARRADA, we use the official train/validation/test split and input sequences of five consecutive radar frames. Models are trained for 300 epochs with batch size 6 using Adam (Kingma & Ba, 2014) with an initial learning rate of 1×10^{-4}, decayed exponentially by a factor of 0.9 every 10 epochs. PyTorch and NumPy seeds are set to 40, and horizontal and vertical flips are applied independently with probability 0.5 during training. Validation is performed every epoch, and the checkpoint with the highest RD validation mIoU is retained for evaluation. Unless otherwise stated, reported results are from a single run with the fixed seed. During inference, the batch size is set to 1 while preserving the same temporal context.

For CARRADA, TopoRadar processes RA, RD, and AD projections through the multi-view radar backbone, applies DuoPerLay to the encoded representation, and then refines the topology-conditioned features through the adaptive attention backbone. The final configuration uses 8 cascaded adaptive-attention blocks. The Structural Consistency Loss is applied during training to the predicted foreground-support masks in the RD and RA views.

For RADIal, which provides a single radar representation rather than the RA/RD/AD multi-view setting, we adapt TopoRadar to a single-view encoder–decoder configuration. Following FFTRadNet (Rebut et al., 2022), we retain the segmentation and detection heads for joint evaluation. Since RADIal does not support multi-view structural alignment, cross-view consistency terms are disabled; the experiment is used to assess the transferability of the topology-aware backbone under a different radar representation and annotation protocol. Unless otherwise stated, the same optimisation strategy is used across datasets.

TopoRadar Implementation Details: The encoded representation derived from RA, RD, and AD inputs is processed by DuoPerLay and fused back into the feature space before adaptive attention refinement. DuoPerLay uses $M = 8$ learnable filtration rows, $N = 16$ discrete filtration levels, a topology resolution of $H_z \times W_z = 64 \times 64$, and a final dual topology descriptor dimension of $D = 256$. The Structural Consistency Loss is applied to foreground-support masks in the predicted RD and RA segmentation views and retains the top $K = 5$ persistent foreground components. We use a numerical stabilisation constant $\kappa = 10^{-8}$ and set the topology regularisation weights to $(\beta_1, \beta_2) = (1 \times 10^{-2}, 1 \times 10^{-4})$. The UOT solver settings are described separately below.

Table 4: **Sensitivity to UOT solver parameters on CARRADA.** The default setting is $\varepsilon = 0.05$, $\tau = 0.8$, and 30 iterations (†). Best results in each metric are shown in **bold**.

ε	τ	Iter.	RD mIoU	RD mDice	RA mIoU	RA mDice
0.02	0.8	30	64.42	74.91	44.88	55.62
0.05	0.8	30†	64.87	**76.14**	45.26	55.63
0.10	0.8	30	64.51	75.02	44.97	55.79
0.05	0.5	30	64.36	74.86	44.71	55.41
0.05	1.0	30	64.61	75.08	45.04	55.88
0.05	0.8	10	64.18	74.64	44.52	55.06
0.05	0.8	20	64.63	75.04	45.03	55.83
0.05	0.8	50	**64.87**	75.25	**45.28**	**56.19**

Unbalanced Optimal Transport Solver Details: We solve the unbalanced optimal transport problem using an entropically regularised Sinkhorn-style iterative solver. The cost matrix is normalised before transport to improve numerical stability. The cost matrix is normalised before transport using a detached maximum for numerical stability, so the normalising scale is treated as constant during backpropagation. We use entropy regularisation ε, marginal relaxation τ, and a fixed number of Sinkhorn iterations. The constant $\kappa = 10^{-8}$ is added to component masses before evaluating KL terms. Since the Structural Consistency Loss is applied only during training, the transport solver does not affect inference time. Unless otherwise stated, we use $\varepsilon = 0.05$, $\tau = 0.8$, and run 30 Sinkhorn iterations.

UOT Parameter Sensitivity: We analyse the sensitivity of the Structural Consistency Loss to the main unbalanced optimal transport parameters: entropy regularisation ε, marginal relaxation τ, and the number of Sinkhorn iterations. The default setting used in TopoRadar is $\varepsilon = 0.05$, $\tau = 0.8$, and 30 Sinkhorn iterations. All variants are evaluated on CARRADA using the same training protocol as the main experiments.

Table 4 shows that TopoRadar is reasonably stable around the default UOT configuration. Very small entropy regularisation can produce sharper transport plans that are more sensitive to noisy components, while the marginal relaxation parameter τ controls the tolerance to unmatched mass. Lower τ allows greater mass relaxation, whereas larger values move the solution closer to balanced transport and reduce tolerance to missing or spurious components. Increasing the number of Sinkhorn iterations from 30 to 50 leaves RD mIoU unchanged and improves RA mIoU by only 0.02 points, while RD mDice decreases by 0.89 and RA mDice increases by 0.56. We therefore keep 30 iterations as a practical trade-off between accuracy and cost.

Structural Component Construction Details: For a finite persistence pair $c = (b_c, d_c)$, Ω_c is taken as the connected support of c at filtration level $d_c - 1$, i.e., immediately before the component merges. The same convention is used for prediction and ground truth.

Structural Loss Implementation Details: For the sublevel branch, low foreground-probability regions enter the filtration first. Therefore, this branch is interpreted as providing a complementary ordering of the foreground-support map rather than as directly extracting high-confidence foreground regions. The superlevel branch activates high foreground-probability regions first. For binary ground-truth masks, equal filtration values are handled deterministically using the row-major ordering of cells in the cubical complex, and zero-persistence pairs are discarded. Essential H_0 components that survive to the end of the filtration are assigned a terminal death level of $d_c = N + 1$. Components are ranked by decreasing persistence, with ties resolved by decreasing support mass and then by the deterministic component order returned by the cubical-complex implementation before top-K selection. Persistence values and discrete component identities are treated as fixed during backpropagation, while predicted centroid and foreground-support mass retain gradients through the selected soft support. Predicted UOT marginal masses also retain gradients through their foreground-support masses, whereas ground-truth masses are constant. Gradients through the transport objective are obtained by backpropagating through the 30 Sinkhorn iterations. If both retained component sets are empty, the corresponding structural term is set to zero. If only one set is empty, the loss reduces to the unmatched-mass penalty τ times the total normalised mass of the non-empty component set. Centroid coordinates

are normalised to $[0,1]^2$ by the spatial dimensions of the corresponding view, component mass is normalised by $H_v W_v$, and persistence is normalised by N. With filtration levels indexed from 1 and essential components assigned the terminal death level $d_c = N + 1$, the maximum persistence is N, so the normalised persistence remains in $[0,1]$. Before UOT, the pairwise component-cost matrix is normalised as $C^{v,\xi} \leftarrow C^{v,\xi}/(\max_{ij} C_{ij}^{v,\xi} + \kappa)$, where $\max_{ij} C_{ij}^{v,\xi}$ is detached from the computational graph and treated as constant during backpropagation. We use equal weights for the normalised component descriptors, $\alpha = \beta = \gamma = 1$, and retain the TransRadar (Dalbah et al., 2024) segmentation-loss weighting with $\alpha_1 = 1$, $\alpha_2 = 10$, and $\alpha_3 = 5$. The Structural Consistency Loss is weighted by $\lambda_{\mathrm{sc}} = 1$.

D Additional Quantitative Results

D.1 Structural Quality Metrics

Since TopoRadar is designed to improve structural consistency in addition to pixel-wise segmentation accuracy, we evaluate topology-oriented structural metrics on the CARRADA test split. We report connected-component error ($\mathrm{B0Err}$), fragmentation ($\mathrm{Frag.}$), merge error (Merge), spurious component count ($\mathrm{Spur.}$), and persistent component matching cost (SCost). The first four metrics provide component-level structural evaluation independently of the training objective, whereas SCost is reported as a training-aligned diagnostic. Lower values indicate better structural agreement between prediction and ground truth.

Structural Evaluation Protocol. Structural metrics are computed independently for each semantic foreground class and for the union of all foreground classes. Predicted class-probability maps are binarised at a fixed threshold of $\theta = 0.5$, while ground-truth masks are binary. Connected components are extracted using 8-connectivity. For each frame, a predicted component and a ground-truth component are considered associated when their component IoU is at least $\eta = 0.10$. Associations are many-to-many, allowing both fragmentation and merging to be measured explicitly.

Let N_P and N_G denote the numbers of predicted and ground-truth components, respectively. $\mathrm{B0Err} = |N_P - N_G|$ measures disagreement in component count. $\mathrm{Frag.}$ measures, per ground-truth component, the average number of additional predicted components associated with it beyond the first, thereby capturing object fragmentation. Conversely, Merge measures, per predicted component, the average number of additional ground-truth components associated with it beyond the first. $\mathrm{Spur.}$ counts predicted components with no ground-truth association. Frames with no components are handled using a denominator of at least one when computing the normalised fragmentation and merge measures.

All metrics are first computed per frame and then averaged over the CARRADA test split. The same threshold, connectivity, association criterion, and aggregation procedure are used for TopoRadar and the re-trained TransRadar backbone. For foreground-union evaluation, all non-background classes are first merged into a single foreground mask and the structural metrics are recomputed on that mask; the reported union values are therefore not sums or averages of the class-wise results.

The persistent component matching cost, SCost, is the unbalanced optimal transport cost used by the Structural Consistency Loss. Since it uses the same matching formulation as $\mathcal{L}_{\mathrm{SC}}$, we report it as a training-aligned diagnostic rather than as independent evidence of structural improvement.

Table 5: **Foreground structural quality on CARRADA.** Metrics are computed on the union of foreground classes. Lower is better.

View	Method	B0Err ↓	Frag. ↓	Spur. ↓	SCost ↓
RD	TransRadar backbone (re-trained)	2.46	0.31	1.84	0.427
	TopoRadar	**1.71**	**0.22**	**1.19**	**0.312**
RA	TransRadar backbone (re-trained)	3.18	0.38	2.27	0.512
	TopoRadar	**2.22**	**0.27**	**1.48**	**0.376**

The TransRadar structural baseline in Tables 5 and 6 is obtained from the same re-trained TransRadar checkpoint used as the baseline in Table 3, evaluated under the identical structural-metric pipeline.

In RD, TopoRadar reduces $\mathrm{B0Err}$ from 2.46 to 1.71, fragmentation from 0.31 to 0.22, and spurious components from 1.84 to 1.19. In RA, the corresponding values decrease from 3.18 to 2.22, 0.38 to 0.27, and 2.27 to 1.48, respectively. These results correspond to approximately 30% lower connected-component error and 35% fewer spurious foreground components in both views. The lower SCost is consistent with these trends but is interpreted only as a training-aligned diagnostic.

Class-wise Structural Analysis. Foreground-union evaluation does not distinguish semantic classes and may therefore hide cases in which foreground connectivity improves while semantic class assignments remain incorrect. We consequently evaluate $\mathrm{B0Err}$, $\mathrm{Frag.}$, Merge, and $\mathrm{Spur.}$ separately for pedestrian, cyclist, and car masks. This analysis is particularly relevant for the RA view, where cyclist segmentation remains substantially more challenging than pedestrian and car segmentation.

Table 6: **Class-wise structural quality on CARRADA.** Structural metrics are computed independently for pedestrian, cyclist, and car classes using the protocol defined above. Lower is better.

View	Class	Method	B0Err ↓	Frag. ↓	Merge ↓	Spur. ↓
RD	Ped.	TransRadar backbone (re-trained)	0.84	0.13	0.07	0.58
		TopoRadar	**0.55**	**0.08**	**0.05**	**0.36**
	Cycl.	TransRadar backbone (re-trained)	0.73	0.11	**0.05**	0.50
		TopoRadar	**0.60**	**0.09**	**0.05**	**0.41**
	Car	TransRadar backbone (re-trained)	0.95	0.15	0.10	0.82
		TopoRadar	**0.59**	**0.08**	**0.06**	**0.47**
RA	Ped.	TransRadar backbone (re-trained)	1.05	0.14	0.09	0.74
		TopoRadar	**0.70**	**0.10**	**0.07**	**0.47**
	Cycl.	TransRadar backbone (re-trained)	**0.92**	**0.12**	**0.08**	**0.64**
		TopoRadar	0.97	0.13	0.09	0.68
	Car	TransRadar backbone (re-trained)	1.26	0.15	0.11	0.96
		TopoRadar	**0.80**	**0.09**	**0.07**	**0.52**

Table 6 provides a more detailed view of the structural behaviour underlying the foreground-union results. TopoRadar reduces component-count error, fragmentation, merge error, and spurious responses for pedestrian and car classes in both RD and RA. For RD cyclist, $\mathrm{B0Err}$ decreases from 0.73 to 0.60, fragmentation from 0.11 to 0.09, and spurious components from 0.50 to 0.41, while merge error remains unchanged at 0.05.

For RA cyclist, however, $\mathrm{B0Err}$ increases from 0.92 to 0.97, fragmentation from 0.12 to 0.13, merge error from 0.08 to 0.09, and spurious component count from 0.64 to 0.68. This trend is consistent with the lower RA cyclist segmentation accuracy reported in Table 1. The class-wise results therefore indicate that the structural improvements are more consistent for pedestrian and car classes, while TopoRadar does not improve every semantic class uniformly. This distinction shows that improved aggregate foreground structure does not necessarily imply uniform improvements in semantic-class discrimination.

D.2 Loss Function Ablation

Table 7: **Loss-function ablation on CARRADA.** We report mIoU and mDice for RD and RA views. $\mathcal{L}_{\mathrm{CA}}$, $\mathcal{L}_{\mathrm{SD}}$, and $\mathcal{L}_{\mathrm{MV}}$ denote class-agnostic object, soft Dice, and multi-view consistency losses, respectively. $\mathcal{L}_{\mathrm{SC}}$ denotes the Structural Consistency Loss, while $\mathcal{L}_{\mathrm{reg}}$ and $\mathcal{L}_{\mathrm{LE}}$ are DuoPerLay-specific filtration regularisers; rows containing either of these terms therefore include the DuoPerLay branch.

$\mathcal{L}_{\mathrm{CA}}$	$\mathcal{L}_{\mathrm{SD}}$	$\mathcal{L}_{\mathrm{MV}}$	$\mathcal{L}_{\mathrm{SC}}$	$\mathcal{L}_{\mathrm{reg}}$	$\mathcal{L}_{\mathrm{LE}}$	RD mIoU	RD mDice	RA mIoU	RA mDice
✓	✗	✗	✗	✗	✗	58.64	69.12	39.18	48.31
✓	✓	✗	✗	✗	✗	60.37	71.04	41.02	50.46
✓	✓	✓	✗	✗	✗	62.10	73.40	43.10	52.60
✓	✓	✓	✓	✗	✗	63.21	74.06	43.86	53.91
✓	✓	✓	✗	✓	✓	63.48	74.31	44.18	54.27
✓	✓	✓	✓	✓	✓	**64.87**	**76.14**	**45.26**	**55.63**

Table 7 analyses the contribution of each loss term. Using only $\mathcal{L}_{\mathrm{CA}}$ provides basic class-agnostic object supervision but yields lower segmentation accuracy. Adding the soft Dice loss improves performance under foreground–background imbalance, while the multi-view consistency term provides further gains by encouraging agreement across radar projections. Introducing $\mathcal{L}_{\mathrm{SC}}$ improves over the $\mathcal{L}_{\mathrm{CA}} + \mathcal{L}_{\mathrm{SD}} + \mathcal{L}_{\mathrm{MV}}$ baseline by incorporating persistent component-level alignment between prediction and ground truth. The filtration regularisers provide additional improvements when combined with the base losses. The full objective achieves the strongest performance, indicating that pixel-level supervision, multi-view consistency, structural component alignment, and topology regularisation provide complementary benefits.

D.3 Effect of Top-K Persistent Component Selection

Table 8: **Effect of top-K persistent component selection on CARRADA.** We report mIoU and mDice for RD and RA views. Best results in each metric are shown in **bold**.

K	RD mIoU	RD mDice	RA mIoU	RA mDice
1	63.92	74.41	44.33	54.72
3	64.37	74.86	44.81	55.41
5	**64.87**	**76.14**	**45.26**	55.63
7	64.61	75.04	45.02	**55.73**
10	64.18	74.69	44.76	55.21

Table 8 studies the effect of retaining the top-K persistent components in the Structural Consistency Loss. Performance generally improves as K increases from 1 to 5, suggesting that retaining multiple dominant components is beneficial for scenes containing several or fragmented foreground structures. The $K = 5$ setting achieves the highest RD mIoU, RD mDice, and RA mIoU, while $K = 7$ gives a marginally higher RA mDice of 55.73 compared with 55.63 at $K = 5$. Increasing K beyond this range generally reduces performance, indicating that lower-persistence components can introduce less reliable structural matches. We therefore use $K = 5$ as the default setting because it provides the strongest overall trade-off across RD and RA segmentation metrics.

D.4 Persistence-Specific Ablations

The preceding ablations establish the contributions of filtration direction, structural matching, and the Structural Consistency Loss, but they do not by themselves determine whether the observed gains arise specifically from persistent topology rather than from additional feature-conditioning capacity or generic component-level supervision. We therefore perform persistence-specific controls using the same re-trained TransRadar backbone, CARRADA split, optimisation settings, and evaluation protocol as the controlled ablations in Table 3. The experiments separately examine non-topological feature conditioning, non-persistent component supervision, the contribution of H_1, and an existing persistent-homology segmentation objective.

Parameter-Matched Non-Topological Conditioning. To control for the additional capacity introduced by DuoPerLay, we replace its persistence computation with a parameter-matched non-topological conditioning module. The encoded radar representation is spatially pooled and passed through a lightweight MLP to produce a $D = 256$ descriptor. Two parallel descriptor branches are used to match the dual-branch structure of DuoPerLay, and their outputs are concatenated before the same gated-additive fusion in Eq. 3. The hidden dimensions are adjusted so that the number of trainable parameters is matched as closely as possible to DuoPerLay. Thus, this baseline preserves descriptor dimensionality, dual-branch capacity, and the feature-fusion mechanism while removing cubical filtrations, persistence diagrams, and persistence vectorisation.

Non-Persistent Component Supervision. We next evaluate whether the Structural Consistency Loss benefits specifically from persistence-based component selection rather than generic connected-component matching. We construct a non-persistent baseline by thresholding the foreground-support map at $\theta = 0.5$, extracting connected components using the same connectivity convention as

Table 9: Persistence-specific ablation on CARRADA. All variants use the same re-trained TransRadar backbone and controlled training protocol. The non-topological conditioning baseline preserves the dual-branch structure, descriptor dimensionality, and fusion mechanism while matching DuoPerLay's trainable capacity as closely as possible.

Variant	RD mIoU	RD mDice	RA mIoU	RA mDice
Feature conditioning				
TransRadar backbone (re-trained)	62.10	73.40	43.10	52.60
Parameter-matched conditioning w/o persistence	62.82	73.76	43.54	53.18
DuoPerLay with H_0 only	63.24	74.05	43.91	53.96
DuoPerLay with $H_0 + H_1$	63.48	74.31	44.18	54.27
Structural supervision				
Connected-component supervision w/o persistence ranking	62.96	73.82	43.61	53.32
Existing PH segmentation loss (Clough et al., 2020)	63.08	73.91	43.72	53.47
Backbone + Structural Consistency Loss	63.21	74.06	43.86	53.91
Complete model				
TopoRadar	**64.87**	**76.14**	**45.26**	**55.63**

Appendix D.1, and retaining at most $K = 5$ components ranked by foreground-support mass instead of persistence. Each component is represented only by its normalised centroid and foreground-support mass. The same unbalanced optimal-transport solver, entropy regularisation, marginal relaxation, and optimisation protocol are retained, while the persistence term is removed from the component cost. This comparison therefore isolates persistence-based component construction and ranking while keeping the remaining structural-matching pipeline unchanged.

H_0 versus $H_0 + H_1$. DuoPerLay uses both connected-component information (H_0) and one-dimensional hole structure (H_1). To determine whether H_1 provides information beyond connectivity alone, we evaluate an H_0-only variant in which the H_1 persistence vectors are removed from both filtration directions. The resulting H_0 descriptor is projected to the same final dimension $D = 256$ before fusion so that the comparison does not simply favour the full model through a larger representation. The $H_0 + H_1$ configuration corresponds to the standard DuoPerLay feature-conditioning model used in the component ablation of Table 3.

Existing Persistent-Homology Segmentation Loss. Finally, we compare our component-level Structural Consistency Loss with a representative persistent-homology segmentation objective following Clough et al. (2020). The existing topology loss is applied to the same re-trained TransRadar backbone and foreground-support predictions, without DuoPerLay, while retaining the same training and evaluation protocol. This control distinguishes the proposed persistence-ranked UOT component alignment from an established output-level topological supervision strategy.

Table 9 shows that the gains cannot be explained solely by additional feature-conditioning capacity or generic component-level supervision. The parameter-matched non-topological conditioning baseline improves over the re-trained TransRadar backbone from 62.10% to 62.82% RD mIoU and from 43.10% to 43.54% RA mIoU, but remains below DuoPerLay with $H_0 + H_1$, which reaches 63.48% and 44.18%, respectively. Using only H_0 yields 63.24% RD mIoU and 43.91% RA mIoU, while incorporating both H_0 and H_1 further improves performance, indicating that higher-order topological information provides complementary information beyond connected components alone. At the output level, connected-component supervision without persistence ranking reaches 62.96% RD mIoU and 43.61% RA mIoU, whereas the proposed Structural Consistency Loss achieves 63.21% and 43.86%, respectively, supporting the usefulness of persistence-based component selection. The existing persistent-homology segmentation loss also improves over the backbone but remains below the proposed structural objective. Finally, combining DuoPerLay with the Structural Consistency Loss produces the strongest overall result of 64.87% RD mIoU and 45.26% RA mIoU. These results indicate that persistent topology contributes beyond parameter-matched feature conditioning and simpler non-persistent structural supervision.

Table 10: Runtime and memory comparison with TransRadar. GFLOPs measure computational cost, FPS measures inference throughput, and peak memory is reported in GB.

Model	GFLOPs ↓	FPS ↑	Memory (GB) ↓
TransRadar (Dalbah et al., 2024)	**1138.57**	**9.60**	**3.25**
TopoRadar	1146.32	9.31	3.34

E Runtime and Memory Overhead

Computational Cost and Runtime: We analyse the runtime and memory footprint of TopoRadar relative to TransRadar using GFLOPs, inference throughput, and peak GPU memory under the same hardware and inference protocol. TopoRadar introduces DuoPerLay for topology-aware feature conditioning, while the Structural Consistency Loss is used only during training and therefore does not affect inference.

As shown in Table 10, TopoRadar introduces only a modest computational overhead compared with TransRadar. GFLOPs increase from **1138.57** to 1146.32, peak GPU memory increases from **3.25** GB to 3.34 GB, and inference throughput decreases slightly from **9.60** FPS to 9.31 FPS. This overhead is expected because DuoPerLay adds topology-aware feature conditioning to the backbone. However, the increase is small relative to the observed gains in segmentation accuracy and structural consistency.

The overhead remains limited because DuoPerLay operates on encoded feature representations rather than full-resolution radar inputs, allowing persistent structure to be extracted at a compact feature resolution. Moreover, the Structural Consistency Loss is used only during training. During inference, no ground-truth component extraction or unbalanced optimal transport matching is performed; only the DuoPerLay feature-conditioning branch remains active. Overall, TopoRadar introduces only a small inference overhead while improving RD segmentation performance and reducing structural errors.

F Additional Qualitative Results

Figure 4 provides an additional qualitative comparison on CARRADA. The scene contains weak and spatially sparse foreground evidence, making it challenging for dense radar segmentation. Compared with the competing methods, **TopoRadar** better preserves the compact car and pedestrian responses while suppressing scattered activations away from the main object support. In the RD view, several baselines either fragment the foreground response or introduce noisy components, whereas TopoRadar remains closer to the ground-truth structure. In the RA view, the foreground targets are small and weakly separated from the background, yet TopoRadar maintains more stable object-level responses with fewer spurious components. These additional examples further support the main qualitative results, showing that topology-aware feature conditioning and structural consistency supervision improve foreground coherence under sparse and cluttered radar observations.

Failure Cases: TopoRadar improves foreground coherence across most scenes, but it can still struggle under extremely weak radar returns, severe clutter, or strong multipath artefacts, as illustrated in Fig. 5. In these challenging cases, the persistent foreground components extracted from the prediction may become incomplete, ambiguous, or noise-dominated, which limits reliable structural recovery. Small or distant objects are especially difficult, as their radar evidence may occupy only a few cells and can therefore be suppressed during feature encoding or confused with spurious responses. Performance may also degrade when multiple foreground objects are spatially close in RA or RD, since their supports can merge into a single persistent component or split inconsistently across views. These failure modes indicate that, while DuoPerLay and the Structural Consistency Loss improve structural consistency, the method remains constrained by the strength, quality, and separability of the underlying radar evidence.

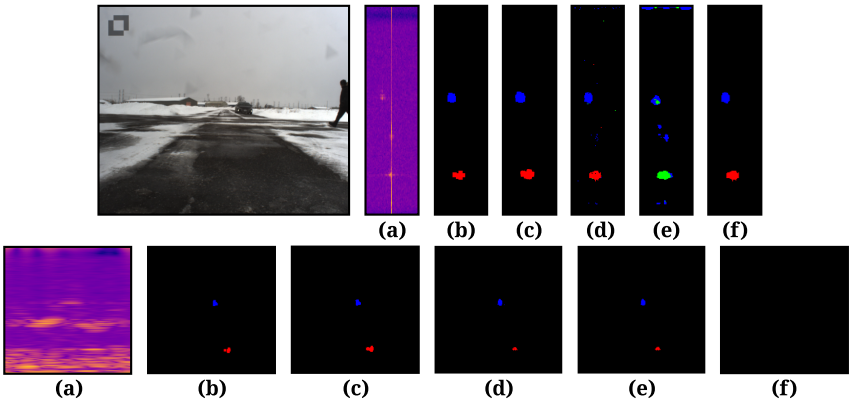


Figure 4: **Additional qualitative comparison on CARRADA.** We show an additional representative test scene with the RGB reference image and semantic segmentation outputs in RD and RA views. Columns denote: (a) radar input, (b) ground truth, (c) **TopoRadar**, (d) TransRadar, (e) TMVA-Net, and (f) MVNet. Colours indicate semantic classes: car, cyclist, pedestrian, and black background.

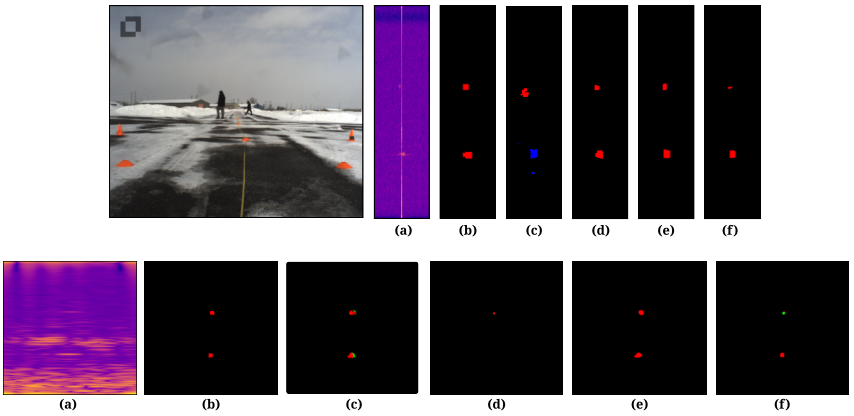


Figure 5: Failure cases on CARRADA. The top row shows the Range–Doppler view and the bottom row shows the Range–Angle view. Columns denote: (a) input radar representation, (b) ground truth, (c) TopoRadar, (d) TransRadar, (e) TMVA-Net, and (f) MVNet. The example highlights challenging scenes with weak radar responses, strong clutter, and spatially sparse foreground evidence, where all methods may partially miss small or distant targets or produce class ambiguity. Although TopoRadar better preserves coherent foreground structure in several regions, its predictions can still be affected when the underlying radar evidence is incomplete, noisy, or poorly separated across views.