

AMGF3D: Adaptive Multi-scale Gated Fusion for Robust Indoor RGB-D 3D Object Detection

Faisal Ahmed
faisal.ahmed@utsa.edu

John Quarles
John.Quarles@utsa.edu

Kevin Desai
kevin.desai@utsa.edu

Department of Computer Science
University of Texas at San Antonio
San Antonio, TX, USA

Abstract

Fully convolutional sparse-voxel detectors are efficient for indoor 3D object detection, but RGB-D detection pipelines can degrade when geometry, appearance, or camera-depth alignment is imperfect. Projection-based 2D-3D fusion can help, yet simple fusion baselines often assume that projected correspondences are reliable and that one fixed image-feature scale is suitable for all object sizes. These assumptions do not hold for indoor scenes with depth discontinuities, occlusion, and large scale variation. We propose Adaptive Multi-scale Gated Fusion (AMGF3D), a selective RGB-D fusion framework for sparse-voxel detectors. AMGF3D aggregates features from a 2D Feature Pyramid Network (P_2 – P_5) using per-voxel scale weights conditioned on local 3D context, and uses a lightweight feature-conditioned gate to modulate projected image evidence before injecting it into the 3D stream. Fusion is applied at two backbone stages to combine lower-level appearance detail with higher-level semantic cues while retaining single-pass inference. Beyond clean accuracy, we evaluate robustness under controlled, multi-seed perturbations. On SUN RGB-D and ARKitScenes, AMGF3D is significantly more robust than additive fusion under the two pre-specified primary perturbations, RGB patch dropout and calibration rotation (six matched seeds, with Wilcoxon and 95% CI support), with gains that persist at mAP@0.50. Across all three benchmarks we observe no statistically significant underperformance within the tested suite. In a SUN RGB-D cross-method comparison, it is best or tied-best on the perturbations that meaningfully degrade accuracy. AMGF3D remains competitive on clean ScanNet, SUN RGB-D, and ARKitScenes evaluation, improves most clearly on small objects, and adds +0.6% parameters and a modest ~9% runtime cost over additive fusion. Code and models are available at <https://github.com/UTSA-VIRLab/AMGF3D>.

1 Introduction

Indoor 3D object detection underpins autonomous robotics, augmented reality, and scene understanding. Sparse 3D convolutional frameworks [6, 30, 31] have made it efficient enough for practical use, but their accuracy depends strongly on input geometry, and in real deployments that geometry is rarely clean: depth sensors are noisy, point clouds are sparse and

unevenly sampled, and RGB-D alignment can drift. Combined with the occlusions, scale variation, and depth ambiguity typical of cluttered indoor scenes, these imperfections degrade single-stream pipelines and motivate detectors that stay reliable when the input is not.

RGB-D fusion [14, 15, 16] is a natural remedy, since image appearance can compensate for missing or unreliable geometry. However, simple projection-based fusion baselines often depend on a fixed-projection assumption: that a 3D voxel maps reliably to a single, fixed-scale 2D feature. This assumption breaks down in two critical ways. First, *geometric misalignment*: under depth noise, sensor parallax, or calibration error, a foreground 3D point may project onto a background image pixel, injecting misaligned appearance features that can corrupt the 3D representation. Second, *scale mismatch*: small objects need fine-grained, high-resolution features whereas large objects benefit from coarser, semantically richer levels, so a single fixed FPN level is unlikely to serve all object sizes equally well. Unconditional fusion can compound both problems by injecting appearance indiscriminately.

We propose Adaptive Multi-scale Gated Fusion (AMGF3D), a selective RGB-D fusion framework that mitigates both failure modes within sparse-voxel detectors, replacing rigid single-scale additive fusion with three complementary choices (Fig. 3). *Adaptive multi-scale aggregation* predicts per-voxel weights across the FPN hierarchy (P_2 – P_5) from local 3D context, so the network learns a scale mixture per voxel instead of sampling one fixed layer. *Feature-conditioned residual gating* modulates per-voxel appearance injection through a sigmoid gate conditioned jointly on the 3D geometry and the sampled 2D descriptor, rather than adding projected features unconditionally. *Dual-stage fusion* applies the module at an early and a later backbone stage, injecting lower-level detail and higher-level semantics.

Beyond clean accuracy, we center our evaluation on *robustness*, comparing the full AMGF design against an additive-fusion baseline under controlled, multi-seed input perturbations, the regime where adaptive fusion should matter most.

The key contributions of our work are as follows:

1. *Adaptive Multi-scale Gated Fusion*: We propose a per-voxel, multi-scale, gated RGB-D fusion module that injects 2D pyramid features into a sparse-voxel backbone at two stages, conditioned on local 3D context.
2. *Multi-seed robustness study*: Through a statistically paired evaluation across all three benchmarks (six matched seeds per dataset, pre-specified primary perturbations, an exact-Wilcoxon and 95%-CI significance criterion), AMGF3D is significantly more robust than additive fusion under calibration rotation and RGB dropout on SUN RGB-D and ARKitScenes, with gains that persist at mAP@0.50. In a SUN RGB-D cross-method comparison, it is best or tied-best on the perturbations that meaningfully degrade accuracy and avoids the sharpest baseline collapses, and we observe no significant underperformance within the tested perturbation suite.
3. *Component-level attribution*: A per-component ablation under perturbation on six matched seeds isolates its source: adaptive multi-scale aggregation and dual-stage placement each carry a significant share of the misalignment robustness, gating a smaller one, while on clean input all differences stay within seed noise.
4. *Clean competitiveness, small-object gains, and efficiency*: AMGF3D stays competitive on clean evaluation at both mAP@0.25 and mAP@0.50 across the three benchmarks, with its clearest gains on small objects under a data-driven volume-based size split, at low parameter (+0.6%) and modest runtime (~9%) overhead.

Code, configurations, the perturbation harness, and the synthetic calibration testbed tooling are publicly available at <https://github.com/UTSA-VIRLab/AMGF3D>.

2 Related Work

2.1 Indoor 3D Object Detection

Early indoor detectors relied on raw-point representations: PointNet++ [25] introduced hierarchical point feature extraction, and VoteNet [26] formulated detection as a Hough voting process from seed points. Transformer-based detectors such as 3DETR [22] and GroupFree3D [19] use attention to model global context, but can incur nontrivial computational and memory overhead in dense indoor scenes. Sparse voxel methods [8] instead compute only on occupied locations, enabling efficient high-resolution processing. It underpins the anchor-free detectors FCAF3D [60] and TR3D [61], which offer a strong accuracy-efficiency trade-off and form the backbone we build on.

More recent indoor detectors push clean accuracy along distinct axes: SPGroup3D [44] groups superpoints in a one-stage detector, V-DETR [52] adds a vertex relative position encoding to a DETR framework, and UniDet3D [13] trains one transformer jointly across six indoor datasets (the latter two report strong ScanNet results). A parallel line changes the input regime: GO-N3RDet [16] detects from multi-view posed RGB without depth, and Cubify Anything [15] scales class-agnostic detection on a large RGB-D benchmark. All target clean accuracy, multi-dataset generalization, or alternative input regimes, largely orthogonal to our focus: a single-pass RGB-D fusion module emphasising perturbation robustness.

2.2 Robustness to Input Perturbations in 3D Perception

Robustness-specific evaluation has become an important concern in 3D perception. Dong *et al.* [9] and Robo3D [14] benchmark common corruptions and sensor failures for outdoor 3D detection and segmentation, while ModelNet-C [28] shows that strong clean accuracy need not imply robustness under corrupted geometry. Closest to our fusion setting, Yu *et al.* [40] and MultiCorrupt [9] benchmark LiDAR-camera fusion under camera failure, sensor degradation, temporal misalignment, and spatial calibration error, and analyse which fusion designs stay robust. These studies target outdoor driving perception or object-level recognition, with corruption taxonomies tailored to those settings; robustness of indoor RGB-D sparse-voxel detection, where geometric, appearance, and calibration perturbations interact in cluttered rooms, is comparatively less explored. We therefore apply perturbations that directly stress the RGB-D fusion pathway: geometry (point-coordinate noise and sparsity), appearance (RGB noise and dropout), and RGB-D alignment (calibration rotation). AMGF3D is not proposed as a corruption benchmark; we use controlled, multi-seed perturbations to ask whether adaptive RGB-D fusion improves robustness in this indoor setting.

2.3 Multimodal RGB-D Fusion

RGB images provide appearance cues that complement the structural information of depth geometry. Fusion strategies are mostly divided into input-level and feature-level methods.

Input-level fusion. Point augmentation methods such as PointPainting [66] attach 2D semantic cues to 3D points before 3D processing. These are simple and efficient, but once applied, the detector has little opportunity to revise unreliable image evidence, leaving performance dependent on the quality and alignment of the projected predictions.

Feature-level fusion. Feature-level methods inject visual cues during 3D representation learning. PointFusion [69] uses a two-stream design for 3D box estimation. 3D-SIS [11] back-projects 2D features into 3D voxels for joint reasoning. ImVoteNet [27] lifts image evidence through 2D-guided voting, and EPNet++ [20] performs point-wise bidirectional

fusion. Closest to us, IIFNet3D [64] and the concurrent GC-DETR [8] improve clean indoor detection with instance-level attention or geometry-color decoupling.

These methods demonstrate the value of appearance for 3D perception, but projection-based fusion can fail when depth discontinuities, occlusion, or calibration error shift the projected feature, causing simple additive or concatenative fusion to mix conflicting appearance with useful geometry. Two properties are shared by essentially all of this prior work and are the ones we change. First, each 3D location is tied to a *single, fixed-scale* 2D descriptor: PointPainting [56] attaches one semantic vector per point, 3D-SIS [11] back-projects one feature map, and ImVoteNet [27] lifts votes from one image resolution. Second, that descriptor is injected *unconditionally*: every voxel receives its projected appearance with the same weight, whether or not the projection is trustworthy. AMGF3D removes both assumptions (Fig. 3). It predicts per-voxel weights over four FPN scales from local 3D context, so the effective descriptor resolution varies with the voxel rather than being fixed by the architecture, and it gates the injection on the joint geometry–appearance feature, so a voxel whose projected evidence disagrees with its geometry can suppress that evidence. Sec. 5.5 shows that these choices, not fusion *per se*, carry the robustness advantage under misalignment.

2.4 Adaptive and Multi-Scale Fusion

Multi-scale aggregation. Indoor objects exhibit large scale variation, from small tabletop items to large furniture. 2D detection frameworks address such variation through feature pyramids [11] and learned cross-scale weighting [35], but many 2D-to-3D transfer pipelines associate each 3D location with a single projected 2D descriptor [11, 27], which is unlikely to suit both small objects requiring high-resolution texture and large objects requiring broader semantic context. AMGF3D therefore predicts per-voxel weights over the P_2 – P_3 FPN levels, producing a 2D descriptor whose scale mixture is conditioned on local 3D context. Recent indoor detectors also use adaptive fusion and aggregation: AVIFF [21] dynamically weights image versus voxel features, and SGCDet [41] adaptively constructs multi-view 3D volumes. Both target clean accuracy. We instead condition the scale weights and a per-voxel gate on local geometry, and evaluate under controlled perturbations rather than for peak accuracy.

Progressive fusion across stages. Building on multi-stage point-based fusion [27], we apply a dual-stage scheme within the sparse-voxel backbone: an early stage ($s=1$) for fine appearance detail and a later stage ($s=3$) for higher-level semantics. This preserves single-pass inference and isolates the effect of adaptive scale selection and stage placement.

3 Methodology

3.1 Problem Formulation and Overview

We address multimodal indoor 3D object detection from posed RGB-D data. We represent each sample by a 3D point cloud $\mathcal{P}$ (back-projected from a depth map for single-view inputs, or given as a scene reconstruction otherwise) and one paired RGB image $I \in \mathbb{R}^{H \times W \times 3}$. The dataset supplies camera calibration that maps a 3D point to the image plane. For a point or voxel center $\mathbf{x}_i$ in the dataset’s depth-coordinate system, we write this projection as

$$\tilde{\mathbf{u}}_i = \mathbf{M}_{\text{depth} \rightarrow \text{img}} [\mathbf{x}_i^\top, 1]^\top, \quad \mathbf{u}_i = (\tilde{u}_{i,x}/\tilde{u}_{i,z}, \tilde{u}_{i,y}/\tilde{u}_{i,z}), \quad (1)$$

where $\mathbf{M}_{\text{depth} \rightarrow \text{img}}$ includes the dataset-provided intrinsics, extrinsics, and alignment transforms. Resizing, padding, and flipping use the same projection utilities as TR3D-style fusion. The detector predicts a set of 3D boxes $\mathcal{B} = \{b_k\}$ and labels $\{y_k\}$ from the fused 3D representation.

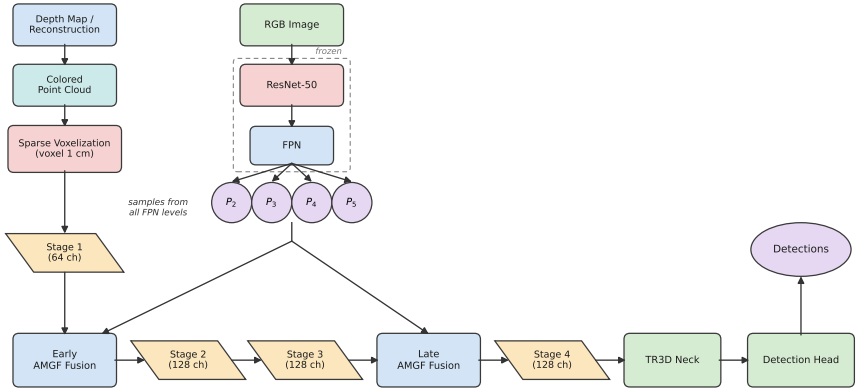


Figure 1: AMGF3D overview. A sparse 3D stream processes colored point clouds, while a frozen ResNet-FPN extracts P_2 – P_5 image features. Projected FPN descriptors are fused into active voxels at backbone layers 1 and 3 using adaptive scale weighting and per-voxel gating before the TR3D neck and detection head.

As illustrated in Fig. 1, AMGF3D uses a sparse 3D stream and a frozen ResNet-FPN image stream. The AMGF module learns per-voxel FPN-scale weights and a residual gate for appearance injection. For SUN RGB-D each sample is a single posed RGB-D frame back-projected into a point cloud; for ScanNet and ARKitScenes we follow the standard scene-level reconstruction protocol [80, 81], pairing the reconstructed point cloud with a single posed RGB frame from the same capture (the first valid frame in the metadata). Voxel centers project into this image to sample 2D features; out-of-image projections are zero-padded, though the residual branch may still apply a geometry-conditioned update. All fusion baselines share this pairing and projection policy, so appearance coverage is matched.

3.2 Multi-Modal Feature Extraction

Following TR3D [82], the 3D stream uses a Minkowski ResNet-34 [8]. The input point cloud contains XYZ coordinates and RGB color. XYZ determines sparse voxel coordinates, while RGB forms the initial per-voxel feature. The backbone outputs sparse tensors at stages $s \in \{1, 2, 3, 4\}$ with channel dimensions $\{64, 128, 128, 128\}$. Active voxels at stage s are denoted by $\mathcal{X}^{(s)} = \{(\mathbf{p}_i^{(s)}, \mathbf{f}_i^{(s)})\}_{i=1}^{N_s}$, where $\mathbf{p}_i^{(s)} \in \mathbb{Z}^3$ is the sparse voxel index and $\mathbf{f}_i^{(s)} \in \mathbb{R}^{C_s}$ is the learned 3D feature.

The 2D stream is a ResNet-50 [44] with FPN [45], kept frozen during training (gradients disabled, batch-normalization layers in inference mode) so that it acts as a fixed pretrained image descriptor rather than a jointly-optimized branch. We initialize it from the ImVoteNet R50-FPN detector [47]. Per-dataset initialization and training configurations are detailed in Sec. 4. We use the first four FPN levels, $\mathcal{P} = \{P_2, P_3, P_4, P_5\}$ (the calligraphic $\mathcal{P}$ denotes this level set throughout; $\mathcal{L}$ is reserved for the training loss, Sec. 3.4), each with $C_{\text{img}} = 256$ channels. For each active voxel we convert its sparse-tensor coordinate to metric coordinates using the same voxel-size and projection convention as TR3D+FF (TR3D with additive fusion), project this center to the image plane, and bilinearly sample each FPN level:

$$\mathbf{z}_i^{(l)} = \text{BilinearSample}\left(\mathbf{F}^{(l)}, \mathbf{u}_i/s_l\right) \in \mathbb{R}^{C_{\text{img}}}, \quad l \in \mathcal{P}, \quad (2)$$

where $\mathbf{F}^{(l)}$ is the FPN feature map at level l and s_l is its stride. Out-of-image projections are zero-padded by sampler. Unlike the TR3D+FF additive-fusion baseline, which samples one image feature level, AMGF3D samples all four FPN levels before adaptive aggregation.

3.3 Adaptive Multi-scale Gated Fusion

Fig. 3 contrasts the module with the additive baseline; Fig. 2 details it. Given a 3D feature $\mathbf{f}_i^{(s)}$ and its projected multi-scale image descriptors $\{\mathbf{z}_i^{(l)}\}_{l \in \mathcal{P}}$, AMGF first constructs a scale-adaptive 2D descriptor, then applies a gated residual update to the 3D feature.

Adaptive Multi-scale Aggregation. Each sampled 2D descriptor is first projected to the stage-specific 3D feature dimension:

$$\tilde{\mathbf{z}}_i^{(s,l)} = \text{ReLU}\left(\mathbf{W}_{\text{proj}}^{(s,l)} \mathbf{z}_i^{(l)}\right), \quad \mathbf{W}_{\text{proj}}^{(s,l)} \in \mathbb{R}^{C_s \times C_{\text{img}}}. \quad (3)$$

We then predict per-voxel scale weights from the local 3D feature:

$$\alpha_i^{(s)} = \text{Softmax}\left(\mathbf{W}_\alpha^{(s)} \text{ReLU}\left(\mathbf{W}_\beta^{(s)} \mathbf{f}_i^{(s)}\right)\right) \in \mathbb{R}^{|\mathcal{P}|}, \quad (4)$$

Here $\alpha_i^{(s)} \in \mathbb{R}^4$ holds the per-voxel scale weights over the four FPN levels P_2 – P_5 : a softmax, so its entries are non-negative and sum to one, predicted from the voxel’s own feature $\mathbf{f}_i^{(s)}$ alone. The matrices form a bottleneck: $\mathbf{W}_\beta^{(s)} \in \mathbb{R}^{(C_s/2) \times C_s}$ halves the feature width and $\mathbf{W}_\alpha^{(s)} \in \mathbb{R}^{|\mathcal{P}| \times (C_s/2)}$ maps it to the four logits. The multi-scale image descriptor is

$$\bar{\mathbf{z}}_i^{(s)} = \sum_{l \in \mathcal{P}} \alpha_i^{(s,l)} \tilde{\mathbf{z}}_i^{(s,l)} \in \mathbb{R}^{C_s}. \quad (5)$$

This lets the detector mix high-resolution features for small structures with coarser semantic features for larger or context-dependent regions, conditioned on local 3D state.

Feature-Conditioned Gated Fusion. Projected image descriptors can be unreliable when the paired RGB frame does not observe the voxel, when depth discontinuities cause foreground-background mismatch, or when calibration error perturbs the projection. AMGF therefore learns a scalar gate from the geometry-appearance pair:

$$g_i^{(s)} = \sigma\left(\mathbf{w}_g^{(s)\top} [\mathbf{f}_i^{(s)}; \bar{\mathbf{z}}_i^{(s)}] + b_g^{(s)}\right), \quad g_i^{(s)} \in (0, 1), \quad (6)$$

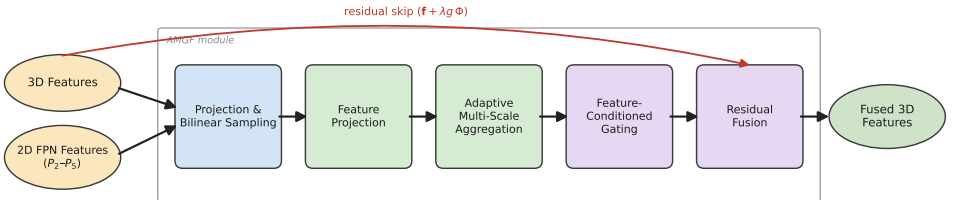


Figure 2: AMGF module. Projected multi-scale FPN descriptors are mapped to the 3D channel dimension, combined with 3D-conditioned scale weights, and injected through a gated residual update; a small gate down-weights the update and preserves the original sparse-voxel feature.

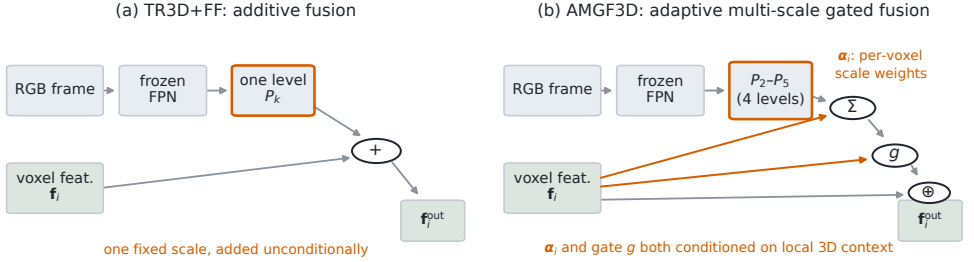


Figure 3: Additive fusion versus AMGF3D at one fusion stage. The additive baseline samples a single fixed FPN level and adds it to every active voxel with the same weight. AMGF3D samples P_2 – P_5 , mixes them with per-voxel weights α_i predicted from the voxel’s own 3D feature, and gates the resulting residual on the joint geometry–appearance pair. Orange marks the two components that differ; Sec. 5.5 isolates their contributions.

where $[\cdot; \cdot]$ denotes concatenation and $\mathbf{w}_g^{(s)} \in \mathbb{R}^{2C_s}$. The final update is a scaled residual:

$$\mathbf{f}_i^{(s),\text{out}} = \mathbf{f}_i^{(s)} + \lambda g_i^{(s)} \Phi^{(s)}\left([\mathbf{f}_i^{(s)}; \mathbf{z}_i^{(s)}]\right), \quad (7)$$

where $\Phi^{(s)}$ is a two-layer MLP mapping $\mathbb{R}^{2C_s}$ to $\mathbb{R}^{C_s}$ and $\lambda = 0.1$ is a fixed residual scale used for stable training. The gate is not supervised by projection-error labels and is not an explicit visibility or calibration estimator, but a learned feature-quality modulator: as $g_i^{(s)} \rightarrow 0$ the residual is suppressed and the original voxel feature preserved. This limits the influence of unreliable projected descriptors rather than injecting appearance unconditionally.

Dual-Stage Progressive Fusion. We apply AMGF at two backbone stages, each with its own (non-shared) parameters: an early stage ($s = 1$, $C_s = 64$) and a later stage ($s = 3$, $C_s = 128$). At each insertion point, the fused tensor replaces the stage activation in place. So the appearance-conditioned update propagates through subsequent sparse-convolution layers and into the TR3D neck. This dual-stage design injects both lower-level appearance and higher-level semantic cues while retaining single-pass inference.

3.4 Detection and Training

Detection Head. Following [40], a lightweight sparse 3D neck aggregates multi-stage backbone features, and the detection head predicts class logits and box parameters with $1 \times 1 \times 1$ sparse convolutions. For oriented SUN RGB-D boxes, the head predicts center offsets, positive box dimensions, and TR3D’s two-channel yaw/shape encoding. For axis-aligned datasets, the yaw terms are omitted. Positive locations are assigned to ground-truth boxes using the TR3D assignment strategy [40].

Training Objective. We train the sparse 3D backbone, detection head, and AMGF modules end-to-end while keeping the 2D backbone frozen. The loss is

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}, \quad \lambda_{\text{reg}} = 1. \quad (8)$$

For classification we use Sigmoid Focal Loss [48] with $\gamma = 2$ and $\alpha = 0.25$. For box regression we follow the TR3D IoU-family objective, computed on decoded boxes: rotated DIoU is used for oriented SUN RGB-D boxes, while axis-aligned IoU/DIoU variants are used for ScanNet and ARKitScenes. This can be written compactly as

$$\mathcal{L}_{\text{reg}} = \frac{1}{N_{\text{pos}}} \sum_{i \in \text{pos}} \left(1 - \text{IoU}_{3D}(\mathbf{b}_i, \mathbf{b}_i^{\text{gt}}) + \eta \frac{\rho^2(\mathbf{c}_i, \mathbf{c}_i^{\text{gt}})}{d^2} \right), \quad (9)$$

where $\rho(\cdot)$ is Euclidean distance, d is the diagonal of the smallest enclosing box, and $\eta \in \{0, 1\}$ indicates whether the DIoU center penalty is enabled for a given TR3D configuration.

Computational Complexity. At each fusion stage, AMGF samples $|\mathcal{P}| = 4$ projected FPN descriptors per active voxel and applies small per-voxel linear layers for scale weighting, gating, and residual fusion. The 2D FPN is computed once per input image and reused at both fusion stages. On SUN RGB-D (single-view inference, batch size 1, one NVIDIA L40S GPU), AMGF3D runs at 23.8 FPS with 0.38 GB peak memory, versus 26.1 FPS and 0.32 GB for the additive-fusion baseline TR3D+FF. The dual-stage gated fusion adds +0.25M parameters (+0.6%, 41.78M vs. 41.53M) and a modest $\sim 9\%$ runtime overhead.

4 Experimental Setup

4.1 Datasets

We evaluate AMGF3D on three indoor 3D detection benchmarks, using each dataset’s standard validation split and format: ScanNet [6], SUN RGB-D [53], and ARKitScenes [10].

ScanNet V2 provides 1,513 reconstructed indoor scans over 18 categories. Following the standard protocol [26, 50, 51] we train on 1,201 and evaluate on 312, using axis-aligned boxes derived from instance masks.

SUN RGB-D is a single-view RGB-D benchmark of 10,335 samples. Following VoteNet [26] we use the 10 common furniture categories with oriented boxes (5,285 train / 5,050 val), where each sample’s RGB and depth come from one camera observation.

ARKitScenes provides mobile-captured room scans over 17 categories, with the MMDetection3D [23] split (4,498 train / 549 val). As in ScanNet, detection uses the reconstruction paired with one posed RGB frame (Sec. 3.1), a scene-level, not per-frame, pairing.

4.2 Implementation Details

Clean-evaluation protocol. We report mAP at IoU 0.25 and 0.50, with mAP@0.25 the primary metric, following prior indoor 3D detection work [26, 50, 51]. For rows with repeated runs, Table 1 uses the FCAF3D/TR3D Best (Mean) convention (best first, mean in parentheses). Following the TR3D/MMDetection3D pipeline, SUN RGB-D and ARKitScenes are sampled to 100k points at validation, while ScanNet is globally axis-aligned and evaluated on all scene points (few ScanNet scenes exceed 100k).

Significance testing. Because Best (Mean) is not a significance test, claims that require statistical support use matched seeds. For the clean ARKitScenes comparison and the robustness study, we compare AMGF3D against TR3D+FF over six paired training seeds per dataset, with the seed fixed within each pair so both models see identical (perturbed) inputs. We report per-seed deltas with two complementary tests: the exact two-sided Wilcoxon signed-rank test and the paired two-sided t -test (95% confidence interval of the deltas excluding zero). At $n=6$ the exact Wilcoxon attains its minimum, $p = 0.03125$, only when all six deltas share sign. We call an effect significant when this holds *and* the 95% CI excludes zero. To limit multiple comparisons we pre-specify calibration rotation and RGB dropout as the *primary* hypotheses (the two failure modes AMGF3D targets, Sec. 1) and treat the remaining perturbations as secondary, descriptive checks.

Size-based split. To probe performance when object geometry is limited, we split categories by per-class median ground-truth box volume on the training split: $V_c = \text{median}_{b \in \mathcal{B}_c} (w_b \cdot h_b \cdot l_b)$, threshold $T = \text{median}_c (V_c)$. Classes with $V_c \leq T$ are **Small**, the rest **Large**; Small- and Large-mAP@0.25 are macro averages over each subset.

Training details. All reproduced models use MMDetection3D [23] with a 0.01m base voxel size on NVIDIA L40S GPUs. The RGB branch is a frozen Faster R-CNN ResNet-50-FPN initialized from the MMDetection3D ImVoteNet checkpoint and shared identically by AMGF3D and TR3D+FF, so the two differ *only* in the fusion operator. Unless the TR3D recipe specifies otherwise, we train 12 epochs with AdamW ($\eta = 10^{-3}$, weight decay 10^{-4}), step decay at epochs 8 and 11, and gradient clipping at norm 10. Augmentation, batch sizes (8–16 samples/GPU) and RGB resizing follow the per-dataset TR3D configs we release.

Robustness perturbations. Perturbations are applied at test time only (training data and checkpoints unchanged), and the multi-seed matrix is run on all three benchmarks, which span near-complete (SUN RGB-D) to sparse (ScanNet, ARKitScenes) RGB coverage. We use five families at three severities: point-coordinate Gaussian noise ($\sigma = 0.01/0.05/0.10\text{m}$); point subsampling (keeping 75/50/25% of points); RGB Gaussian noise ($\sigma = 0.05/0.10/0.20$ after normalization); RGB patch dropout (one square of side $\sqrt{f} \cdot \min(H, W)$, $f \in \{0.25, 0.5, 0.75\}$); and calibration rotation, a zero-mean point-cloud yaw ($\sigma = 1^\circ/3^\circ/5^\circ$) with the RGB image and projection matrix fixed. The last is a lightweight proxy for 2D-3D extrinsic mismatch (ground-truth boxes remain in the original frame), so we read it via paired deltas against TR3D+FF on identical inputs, not as absolute AP. The two RGB families corrupt the *image* passed to the 2D backbone and leave the per-point colour of the coloured point cloud intact. This isolates the projected-appearance pathway in which AMGF3D and TR3D+FF differ, since both models then receive identical point colour. The RGB results accordingly describe camera-side appearance failure, not a total loss of colour (Sec. 6). Table 2 reports representative mid-severity results. Fig. 4 summarizes paired deltas across all three benchmarks at both IoU thresholds.

5 Results

We organize our evaluation around the paper’s central question: *when does multimodal fusion improve robustness as inputs degrade?* We first establish clean competitiveness (Sec. 5.1); our central contribution is then the multi-seed robustness study (Sec. 5.2), a regime-dependent characterization of where adaptive fusion helps. We then break performance down by object size (Sec. 5.3), and close with qualitative results and ablations.

5.1 Overall Results for 3D Object Detection

Table 1 compares AMGF3D with established and recent detectors. It reaches 76.1% (75.6%), 69.7% (69.1%), and 77.8% (77.4%) mAP@0.25 on ScanNet, SUN RGB-D, and ARKitScenes: *on par* with the multimodal TR3D+FF baseline on SUN RGB-D (paired-seed test, no significant difference) and modestly ahead on ScanNet (+0.8) and ARKitScenes (+1.1). At the stricter mAP@0.50 it ties on ScanNet (60.1 vs. 60.1), is on par on SUN RGB-D (+0.2), and leads on ARKitScenes (+1.4) (Table 1). On ARKitScenes this clean-evaluation advantage is statistically supported: across the six matched training seeds used for the paired test (Sec. 4), the per-seed difference $\Delta = \text{AMGF3D} - \text{TR3D+FF}$ is positive at both IoU thresholds for all six seeds and significant under our paired-seed criterion. These clean-evaluation results establish that the adaptive gated fusion does not sacrifice accuracy; the more distinctive advantage emerges under input degradation (Sec. 5.2).

Method	Venue	Inputs	ScanNet		SUN RGB-D		ARKitScenes	
			@0.25	@0.50	@0.25	@0.50	@0.25	@0.50
VoteNet [12]	ICCV'19	PC	58.6	33.5	57.7	–	–	–
GSDN [9]	ECCV'20	PC	62.8	34.8	–	–	–	–
ImVoteNet [13]	CVPR'20	PC+RGB	–	–	63.4	–	–	–
EPNet [14]	ECCV'20	PC+RGB	–	–	64.6	–	–	–
TokenFusion [15]	CVPR'22	PC+RGB	–	–	64.9 (64.4)	48.3	–	–
EPNet++ [16]	TPAMI'23	PC+RGB	–	–	65.3	–	–	–
MMTC [17]	BMVC'21	PC+RGB	–	–	65.3 (64.7)	48.6	–	–
BRNet [9]	CVPR'21	PC	66.1	50.9	61.1	43.7	–	–
3DETR [18]	ICCV'21	PC	65.0	47.0	59.1	32.7	–	–
H3DNet [19]	ECCV'20	PC	67.2	48.1	60.1	39.0	–	–
GroupFree [19]	ICCV'21	PC	69.1 (68.6)	52.8 (51.8)	63.0 (62.6)	45.2 (44.4)	–	–
RBGNet [20]	CVPR'22	PC	70.6 (69.9)	55.2 (54.7)	64.1 (63.6)	47.2 (46.3)	–	–
HyperDet3D [21]	CVPR'22	PC	70.9	57.2	63.5	47.3	–	–
FCAF3D [6a]	ECCV'22	PC	71.5 (70.7)	57.3 (56.0)	64.2 (63.8)	48.9 (48.2)	72.3 (71.7)	60.5 (59.6)
TR3D [6b]	ICIP'23	PC	72.9 (72.0)	59.3 (57.4)	67.1 (66.3)	50.4 (49.6)	75.4 (74.9)	61.3 (60.7)
TR3D+FF [6b]	ICIP'23	PC+RGB	75.3 (74.5)	60.1 (58.9)	69.4 (68.7)	53.4 (52.2)	76.7 (76.1)	62.5 (61.7)
SPGroup3D [19]	AAAI'24	PC	74.3 (73.5)	59.6 (58.3)	65.4 (64.8)	47.1 (46.4)	–	–
V-DETR [8a]	ICLR'24	PC	77.8	65.9	67.5	50.4	–	–
UniDet3D [†] [8b]	AAAI'25	PC	77.5 (77.0)	66.1 (65.0)	–	–	–	–
AMGF3D (Ours)	–	PC+RGB	76.1 (75.6)	60.1 (59.0)	69.7 (69.1)	53.6 (52.3)	77.8 (77.4)	63.9 (62.7)

Table 1: Clean detection comparison, mAP@0.25/@0.50. The sparse-voxel rows FCAF3D, TR3D, TR3D+FF and AMGF3D are *our own reproductions* under one matched pipeline (same frozen 2D backbone, splits, schedule and evaluation code), in the FCAF3D/TR3D Best (Mean) convention; they are not quotations of those papers’ printed numbers, so small differences are expected. Other rows are quoted from their source papers. **Bold** marks the best entry per column. [†]For UniDet3D we quote its *single-dataset* ScanNet result, trained on ScanNet alone (best from its Tab. 5; 25-trial mean from its Tab. 3), for a like-for-like comparison; its headline numbers use joint training over six datasets, and its ARKitScenes protocol is not comparable to ours.

Against recent detectors, AMGF3D leads all listed methods on SUN RGB-D and ARKitScenes and stays competitive on ScanNet, surpassing SPGroup3D [19] on ScanNet (+1.8/+0.5) and SUN RGB-D (+4.3/+6.5). The point-only transformers V-DETR [8a] and UniDet3D [8b] lead on ScanNet: V-DETR is lower on SUN RGB-D (67.5 vs. 69.7), and UniDet3D reports no SUN RGB-D result while leading us by 1.4 mAP@0.25 on ScanNet alone. Our aim is efficient, robust RGB-D fusion, not peak clean accuracy.

5.2 Robustness to Input Perturbations

We subject AMGF3D and the strongest multimodal baseline, TR3D+FF, to the five test-time perturbation families of Sec. 4. TR3D+FF shares AMGF3D’s exact fusion inputs (same paired RGB frame and projection), so the comparison isolates the fusion *design*, not the inputs or backbone. Table 2 reports absolute mAP@0.25 for both, averaged over six matched seeds per dataset. Significance uses the paired-seed criterion of Sec. 4 on the per-seed difference $\Delta = \text{AMGF3D} - \text{TR3D+FF}$.

On **ARKitScenes**, both pre-specified primaries are significant: *RGB dropout* (+0.7 mAP@0.25) and, by a larger margin, *calibration rotation* (+2.6 at 3°, a controlled proxy for 2D-3D misalignment, Sec. 4). Both persist and partly strengthen at mAP@0.50 (RGB dropout +1.8, calibration rotation +1.0), so neither is an artifact of the coarse metric.

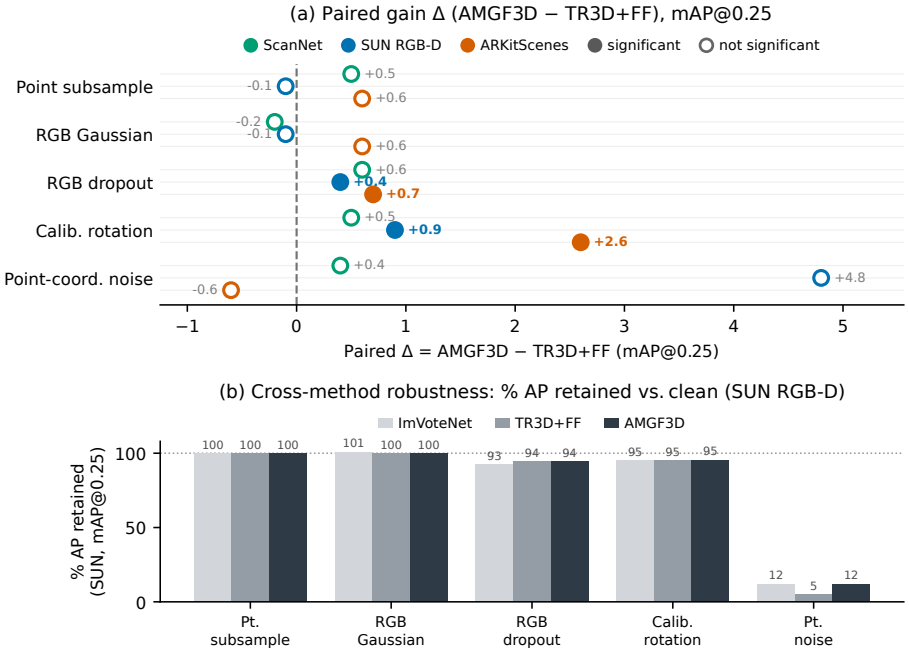


Figure 4: Robustness summary. **(a)** Paired difference $\Delta = \text{AMGF3D} - \text{TR3D+FF}$ in mAP@0.25 at representative perturbation severities, averaged over six matched training seeds. Filled markers are significant under the paired-seed criterion (Sec. 4), hollow markers are not. **(b)** SUN RGB-D cross-method robustness, shown as AP retained relative to clean. For the perturbations that degrade accuracy, AMGF3D is best or tied-best. Stricter mAP@0.50 results are reported in the supplementary material.

AMGF3D is also more robust on average to point subsampling and RGB noise, though not significantly, and on par under point-coordinate noise. **SUN RGB-D** follows the same direction. Both primaries are again significant, calibration rotation (+0.9) and RGB patch-dropout (+0.4), and both persist at mAP@0.50 (+0.7 and +0.5). Secondary point-coordinate noise shows a large average gain under severe noise (+4.8 at $\sigma=0.10$, $3.3 \rightarrow 8.1$), but the per-seed deltas are not unanimous, so we report it as descriptive. Full @0.50 deltas for all three benchmarks are in the supplementary material. It is on par with additive fusion under point subsampling and RGB Gaussian noise. On **ScanNet**, the deltas are small and within seed-level variance (Table 2, where RGB Gaussian is marginally negative at -0.2), consistent with its high-quality aligned reconstruction and sparse single-frame coverage, leaving less for fusion to recover. Both primaries are therefore significant on SUN RGB-D and ARKitScenes, and no significant underperformance appears anywhere in the tested suite; we read the latter as a limit of what six seeds can detect (Sec. 6).

Fig. 4(a) visualizes these paired deltas at mAP@0.25. Beyond this within-design comparison, a cross-method comparison on SUN RGB-D (Fig. 4b, full table in the supplementary) shows that AMGF3D is best or tied-best on perturbations that degrade accuracy. It avoids TR3D+FF’s sharpest point-noise collapse and ImVoteNet’s strongest RGB-dropout degradation (point subsampling and RGB Gaussian noise barely affect any method).

Perturbation (level)	ScanNet		SUN RGB-D		ARKitScenes	
	TR3D+FF	AMGF3D	TR3D+FF	AMGF3D	TR3D+FF	AMGF3D
Clean	74.5	75.6	68.7	69.1	76.1	77.4
Point subsample (50%)	69.6	70.1	68.6	68.5	75.2	75.8
RGB Gaussian ($\sigma=0.1$)	71.0	70.8	68.6	68.5	75.3	75.9
RGB dropout (50%)	71.1	71.7	64.5	64.9	75.3	76.0
Calib. rotation (3°)	66.7	67.2	65.0	65.9	61.7	64.3
Point-coord. noise ($\sigma=0.10$ m)	0.7	1.1	3.3	8.1	4.1	3.5

Table 2: Robustness to representative test-time perturbations: absolute mAP@0.25 for the additive-fusion baseline TR3D+FF and AMGF3D, averaged over six matched training seeds per dataset. **Bold** marks AMGF3D entries significantly more robust than TR3D+FF under the paired-seed criterion (Sec. 4). RGB dropout and calibration rotation are the pre-specified primary perturbations. Point-coordinate noise is a secondary descriptive check.

Method	SUN RGB-D			ScanNet			ARKitScenes		
	Small	Large	Overall	Small	Large	Overall	Small	Large	Overall
FCAF3D	70.5 (69.8)	56.2 (55.7)	64.2 (63.8)	69.4 (68.7)	72.2 (71.6)	71.5 (70.7)	63.8 (63.1)	82.0 (81.2)	72.3 (71.7)
TR3D	71.8 (71.1)	60.0 (59.2)	67.1 (66.3)	71.1 (70.5)	74.1 (73.5)	72.9 (72.0)	65.7 (64.8)	85.6 (84.4)	75.4 (74.9)
TR3D+FF	75.2 (74.7)	63.4 (62.7)	69.4 (68.7)	76.1 (75.3)	77.7 (77.0)	75.3 (74.5)	69.0 (67.7)	86.1 (83.9)	76.7 (76.1)
AMGF3D (Ours)	75.8 (75.1)	62.9 (62.3)	69.7 (69.1)	76.6 (76.0)	78.4 (77.8)	76.1 (75.6)	70.7 (69.4)	85.3 (84.2)	77.8 (77.4)

Table 3: Object-size analysis at mAP@0.25 using volume-based category split. Small-mAP and Large-mAP are macro averages over category subsets. Overall repeats the full-category Best (Mean) mAP@0.25 from Table 1 and is not a size-weighted average of Small and Large. AMGF3D improves Small-mAP on all benchmarks, while large-object results are mixed.

5.3 Object-Size Based Comparisons

Table 3 provides a detailed performance breakdown using our volume-based split. AMGF3D achieves the strongest performance on small-object categories, improving Small-mAP@0.25 (peak) by +0.6%, +0.5%, and +1.7% over TR3D+FF on SUN RGB-D, ScanNet, and ARKitScenes, respectively, consistent with AMGF3D helping most where geometric evidence is sparse. These aggregates are macro averages, and the gain behind them is category-concentrated rather than a uniform lift across the small classes: on SUN RGB-D it is carried almost entirely by *night stand* and, among the large classes, by *dresser*, with the remaining categories essentially at parity. On large objects it is comparable overall, with the small per-dataset trade-offs reported in Table 3, consistent with geometry-rich detections already being well served by the sparse-voxel backbone.

5.4 Qualitative Results

Figure 5 illustrates small or partially observed objects recovered by AMGF3D and baseline false positives removed on ScanNet; quantitative results are in Secs. 5.3 and 5.2.

5.5 Design Analysis

Because TR3D+FF is AMGF3D with adaptive multi-scale aggregation, dual-stage placement, and gating removed, the robustness comparison of Sec. 5.2 is a *cumulative* ablation. To attribute that cumulative effect to individual components, we train and evaluate four intermediate variants on ARKitScenes over the same six matched seeds and the same protocol (Sec. 4): *single-scale* replaces the adaptive P_2 – P_3 mixture with the single fixed FPN level



Figure 5: Qualitative comparison on ARKitScenes, SUN RGB-D, and ScanNet. Columns: paired RGB frame, bird’s-eye view ground truth, TR3D+FF, and AMGF3D. Box colors denote TP/FP/FN at $\text{IoU} \geq 0.25$. On ScanNet and ARKitScenes, the single posed RGB frame covers only part of the reconstructed scene, and the BEV is cropped for legibility. AMGF3D recovers small or partially observed objects and removes baseline false positives.

TR3D+FF uses; *early-only* and *late-only* retain one of the two fusion stages; and *no-gate* injects the multi-scale residual with $g_i^{(s)} \equiv 1$.

Table 4 reports the outcome. Under calibration rotation every appearance-side component contributes: removing multi-scale aggregation costs 1.6 mAP@0.25, removing the early stage 1.8, and removing the late stage 2.4. For all three, the six per-seed deltas share one sign and the 95% CI excludes zero (exact Wilcoxon $p=0.03125$). The gate adds a further 0.9 under calibration rotation and stays within noise elsewhere, matching the role it was designed for (Sec. 3.3), down-weighting appearance that the local geometry contradicts. Under clean input and RGB dropout, by contrast, all component deltas fall within seed noise (≤ 0.6), which is why we motivate these choices by robustness and not by clean accuracy. Adaptive multi-scale aggregation and dual-stage placement, not gating alone or fusion *per se*, are what buy robustness to 2D-3D misalignment, and every variant’s calibration-rotation score falls back toward the additive baseline.

Appearance, capacity, and a synthetic check. AMGF3D adds +0.6% parameters, so we test whether its gains are an appearance or a capacity effect. Where RGB coverage is dense (SUN RGB-D), the model clearly relies on appearance: zeroing all projected image features at test time drops AMGF3D from 68.5 to 45.8 mAP@0.25, and TR3D+FF comparably ($68.6 \rightarrow 49.3$). The sparse scene-level case is less favourable. On ARKitScenes a capacity-matched geometry-only variant, trained with zeroed image descriptors, matches the identically trained full model on clean input to within 0.5 mAP over six seeds, so we attribute the ARKit *clean* gain to capacity and not to appearance. The robustness margin is not: additive fusion has the same parameter count yet falls to 61.7 under calibration rotation (Table 4) by injecting misaligned appearance unconditionally. Finally, because calibration

Variant	Clean	Calib. 3°	Drop. 50%
TR3D+FF (additive)	76.1	61.7	75.3
early-only (−late stage)	77.1	61.9	75.7
late-only (−early stage)	76.8	62.5	75.4
single-scale (−multi-scale)	77.2	62.7	75.9
no-gate (−gate)	77.1	63.4	75.8
AMGF3D (full)	77.4	64.3	76.0

Table 4: Per-component ablation on ARKitScenes, mAP@0.25 over six matched training seeds. Each row removes one component from the full model. **Bold** marks the best entry per column. The corresponding mAP@0.50 results are in the supplementary material.

rotation perturbs the points and not the projection, we replicate the misalignment result on a synthetic testbed with exact calibration (Hypersim [29] with Omni3D [9] labels), injecting extrinsic-rotation error into the projection alone: AMGF3D stays significantly more robust (+1.6 mAP@0.25, +2.2 mAP@0.50 at 3°, six seeds, all deltas of one sign) at clean parity, while under severe synthetic depth noise both collapse with no advantage.

6 Limitations

AMGF3D depends on useful, reasonably aligned appearance and does not guarantee a geometry-only fallback: under failures more severe than tested (near-total RGB loss, large miscalibration) the appearance branch may become unreliable. Our RGB perturbations corrupt the image given to the 2D backbone while leaving per-point colour intact (Sec. 4), so they model camera-pathway degradation and not a complete loss of colour. Component attribution (Sec. 5.5) is evaluated on ARKitScenes alone, where a capacity-matched geometry-only variant also matches the full model on clean input, so we do not claim that clean gain as an appearance effect, and the balance between components may differ where RGB coverage is dense. Our claims rest on six matched seeds per dataset. This supports the pre-specified primary effects but leaves the secondary perturbations descriptive, and non-significance there should not be read as evidence of equivalence. Finally, calibration rotation is a controlled proxy and not a physical sensor model, and the testbed that isolates extrinsic error is itself synthetic. We do not cover motion blur, missing frames, or correlated drift, and since ScanNet and ARKitScenes pair each reconstruction with one posed RGB frame, multi-view appearance aggregation remains future work.

7 Conclusion

We introduced AMGF3D, a selective RGB-D fusion module for sparse-voxel indoor detectors combining per-voxel multi-scale FPN aggregation with gated residual appearance injection at two backbone stages. Rather than a universal clean-accuracy gain, our central result is robustness: AMGF3D is significantly more robust than additive fusion under both pre-specified primary perturbations, RGB dropout and calibration rotation, on SUN RGB-D and ARKitScenes, with gains that persist at mAP@0.50.

On clean evaluation it stays competitive on all three benchmarks and improves most clearly on small objects, adding only +0.6% parameters and ~9% runtime over additive fusion, positioning adaptive appearance fusion as a practical robustness mechanism indoors.

Acknowledgements

This work was supported by a grant from the National Science Foundation (IIS 2316240).

References

- [1] G. Baruch et al. ARKitScenes: A Diverse Real-World Dataset for 3D Indoor Scene Understanding Using Mobile RGB-D Data. In *Proc. NeurIPS Datasets and Benchmarks*, 2021.
- [2] T. Beemelmans, Q. Zhang, C. Geller, and L. Eckstein. MultiCorrupt: A Multi-Modal Robustness Dataset and Benchmark of LiDAR-Camera Fusion for 3D Object Detection. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*, pages 3255–3261, 2024.
- [3] G. Brazil, A. Kumar, J. Straub, N. Ravi, J. Johnson, and G. Gkioxari. Omni3D: A Large Benchmark and Model for 3D Object Detection in the Wild. In *Proc. CVPR*, pages 13154–13164, June 2023.
- [4] B. Cheng, L. Sheng, S. Shi, M. Yang, and D. Xu. Back-Tracing Representative Points for Voting-Based 3D Object Detection in Point Clouds. In *Proc. CVPR*, pages 8963–8972, 2021.
- [5] C. Choy, J. Gwak, and S. Savarese. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In *Proc. CVPR*, 2019.
- [6] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *Proc. CVPR*, 2017.
- [7] Y. Dong, C. Kang, J. Zhang, Z. Zhu, Y. Wang, X. Yang, H. Su, X. Wei, and J. Zhu. Benchmarking Robustness of 3D Object Detection to Common Corruptions in Autonomous Driving. In *Proc. CVPR*, pages 1022–1032, 2023.
- [8] Y. Fang, Q. Wu, S. Li, H. Xu, K. Jiang, J. Cai, and J. Wang. GC-DETR: Geometry–Color Decoupled Transformer for 3D Indoor Object Detection in Smart Consumer Devices. *IEEE Transactions on Consumer Electronics*, 2026. doi: 10.1109/TCE.2026.3695613. Early Access.
- [9] J. Gwak, C. Choy, and S. Savarese. Generative Sparse Detection Networks for 3D Single-Shot Object Detection. In *Proc. ECCV*, 2020.
- [10] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proc. CVPR*, 2016.
- [11] J. Hou, A. Dai, and M. Nießner. 3D-SIS: 3D Semantic Instance Segmentation of RGB-D Scans. In *Proc. CVPR*, 2019.
- [12] T. Huang, Z. Liu, X. Chen, and X. Bai. EPNet: Enhancing Point Features With Image Semantics for 3D Object Detection. In *Proc. ECCV*, pages 35–52, 2020. doi: 10.1007/978-3-030-58555-6_3.
- [13] M. Kolodiaznyi, A. Vorontsova, M. Skripkin, D. Rukhovich, and A. Konushin. UniDet3D: Multi-dataset Indoor 3D Object Detection. In *Proc. AAAI*, 2025.
- [14] L. Kong, Y. Liu, X. Li, R. Chen, W. Zhang, J. Ren, L. Pan, K. Chen, and Z. Liu. Robo3D: Towards Robust and Reliable 3D Perception against Corruptions. In *Proc. ICCV*, pages 19994–20006, 2023.

- [15] J. Lazarow, D. Griffiths, G. Kohavi, F. Crespo, and A. Dehghan. Cubify Anything: Scaling Indoor 3D Object Detection. In *Proc. CVPR*, 2025.
- [16] Z. Li, H. Yu, Y. Ding, J. Qiao, B. Azam, and N. Akhtar. GO-N3RDet: Geometry Optimized NeRF-enhanced 3D Object Detector. In *Proc. CVPR*, 2025.
- [17] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature Pyramid Networks for Object Detection. In *Proc. CVPR*, 2017.
- [18] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal Loss for Dense Object Detection. In *Proc. ICCV*, 2017.
- [19] Z. Liu, Z. Zhang, Y. Cao, H. Hu, and X. Tong. Group-Free 3D Object Detection via Transformers. In *Proc. ICCV*, 2021.
- [20] Z. Liu, T. Huang, B. Li, X. Chen, X. Wang, and X. Bai. EPNet++: Cascade Bi-Directional Fusion for Multi-Modal 3D Object Detection. *IEEE TPAMI*, 45(7):8324–8341, 2023. doi: 10.1109/TPAMI.2022.3228806.
- [21] Z. Liu, Z. Fu, G. Li, and S. Zhang. A Novel Multi-Model 3D Object Detection Framework with Adaptive Voxel-Image Feature Fusion. *IET Computer Vision*, 18(5):640–651, 2024.
- [22] I. Misra, R. Girdhar, and A. Joulin. An End-to-End Transformer Model for 3D Object Detection. In *Proc. ICCV*, 2021.
- [23] MMDetection3D Contributors. MMDetection3D: OpenMMLab Next-generation Platform for General 3D Object Detection. <https://github.com/open-mmlab/mmdetection3d>, 2020.
- [24] J. Park, X. Weng, Y. Man, and K. Kitani. Multi-Modality Task Cascade for 3D Object Detection. In *Proc. BMVC*, 2021.
- [25] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Proc. NeurIPS*, 2017.
- [26] C. R. Qi, O. Litany, K. He, and L. J. Guibas. Deep Hough Voting for 3D Object Detection in Point Clouds. In *Proc. ICCV*, 2019.
- [27] C. R. Qi, X. Chen, O. Litany, and L. J. Guibas. ImVoteNet: Boosting 3D Object Detection in Point Clouds With Image Votes. In *Proc. CVPR*, pages 4404–4413, 2020. doi: 10.1109/CVPR42600.2020.00446.
- [28] J. Ren, L. Pan, and Z. Liu. Benchmarking and Analyzing Point Cloud Classification under Corruptions. In *Proc. ICML*, pages 18559–18575, 2022.
- [29] M. Roberts, J. Ramapuram, A. Ranjan, A. Kumar, M. A. Bautista, N. Paczan, R. Webb, and J. M. Susskind. Hypersim: A Photorealistic Synthetic Dataset for Holistic Indoor Scene Understanding. In *Proc. ICCV*, pages 10912–10922, October 2021.
- [30] D. Rukhovich, A. Vorontsova, and A. Konushin. FCAF3D: Fully Convolutional Anchor-Free 3D Object Detection. In *Proc. ECCV*, 2022.

- [31] D. Rukhovich, A. Vorontsova, and A. Konushin. TR3D: Towards Real-Time Indoor 3D Object Detection. In *Proc. ICIP*, pages 281–285, 2023. doi: 10.1109/ICIP49359.2023.10222644.
- [32] Y. Shen, Z. Geng, Y. Yuan, Y. Lin, Z. Liu, C. Wang, H. Hu, N. Zheng, and B. Guo. V-DETR: DETR with Vertex Relative Position Encoding for 3D Object Detection. In *Proc. ICLR*, 2024.
- [33] S. Song, S. P. Lichtenberg, and J. Xiao. SUN RGB-D: A RGB-D Scene Understanding Benchmark Suite. In *Proc. CVPR*, 2015.
- [34] Z. Sun, Z. Fan, B. Fan, and H. Liu. IIFNet3D: Instance-to-Instance Fusion with Dual Attention for Indoor RGB-D 3D Object Detection. *Pattern Recognition*, 179:113541, 2026. doi: 10.1016/j.patcog.2026.113541.
- [35] M. Tan, R. Pang, and Q. V. Le. EfficientDet: Scalable and Efficient Object Detection. In *Proc. CVPR*, 2020.
- [36] S. Vora, A. H. Lang, B. Helou, and O. Beijbom. PointPainting: Sequential Fusion for 3D Object Detection. In *Proc. CVPR*, 2020.
- [37] H. Wang, S. Shi, Z. Yang, R. Fang, Q. Qian, H. Li, B. Schiele, and L. Wang. RBGNet: Ray-Based Grouping for 3D Object Detection. In *Proc. CVPR*, pages 1110–1119, 2022.
- [38] Y. Wang, X. Chen, L. Cao, W. Huang, F. Sun, and Y. Wang. Multimodal Token Fusion for Vision Transformers. In *Proc. CVPR*, pages 12186–12195, 2022.
- [39] D. Xu, D. Anguelov, and A. Jain. PointFusion: Deep Sensor Fusion for 3D Bounding Box Estimation. In *Proc. CVPR*, 2018.
- [40] K. Yu, T. Tao, H. Xie, Z. Lin, T. Liang, B. Wang, P. Chen, D. Hao, Y. Wang, and X. Liang. Benchmarking the Robustness of LiDAR-Camera Fusion for 3D Object Detection. In *Proc. CVPR Workshops*, pages 3188–3198, 2023.
- [41] R. Zhang, Z. Yu, S.-Y. Cao, L. Zhu, G. Zhang, X. Bai, and H.-L. Shen. Boosting Multi-View Indoor 3D Object Detection via Adaptive 3D Volume Construction. In *Proc. ICCV*, 2025.
- [42] Z. Zhang, B. Sun, H. Yang, and Q. Huang. H3DNet: 3D Object Detection Using Hybrid Geometric Primitives. In *Proc. ECCV*, 2020.
- [43] Y. Zheng, Y. Duan, J. Lu, J. Zhou, and Q. Tian. HyperDet3D: Learning a Scene-Conditioned 3D Object Detector. In *Proc. CVPR*, pages 5585–5594, 2022.
- [44] Y. Zhu, L. Hui, Y. Shen, and J. Xie. SPGroup3D: Superpoint Grouping Network for Indoor 3D Object Detection. In *Proc. AAAI*, 2024.