

Map-Det3D: Metric Feed-Forward 3D Reconstruction Prior for Multi-view 3D Object Detection from Streaming Inputs

Yung-Hsu Yang¹, Luigi Piccinelli¹, Samuel Rota Bulò², Sunghwan Hong¹,
Denis Rozumny², Johannes Schönberger², Zuria Bauer¹, Hermann Blum³,
Peter Kotschieder², and Marc Pollefeys¹

¹ ETH Zürich, Switzerland

² Meta Reality Labs Zürich, Switzerland

³ University of Bonn, Germany

Abstract. Metric 3D object detection is a core capability for embodied agents, yet most reliable systems lean on depth sensors, trading away cost, power, and integration simplicity. This motivates monocular 3D detection, which avoids additional constraints, yet it faces a major obstacle: from a single image, depth, and especially absolute scale, are underconstrained. As a result, the prevailing pattern of detecting in 2D and then predicting 3D attributes is often brittle, since modest range errors can dominate 3D localization, and the learned scale prior can fail when cameras, motion, or environments undergo domain shifts. To address this, we propose Map-Det3D, an online multi-view 3D object detection model that brings detection directly into a 3D space reconstructed from RGB. We map a short temporal window into multiple views and repurpose a feed-forward metric 3D reconstruction model as our geometric backbone while tuning its object-aware capabilities. Building on this representation, Map-Det3D directly predicts boxes in metric 3D space, without the widely used 2D-to-3D lifting. Experiments across different benchmarks show that this design supports strong online performance and robust transfer without adaptation, suggesting that training reconstruction priors for detection is a practical route to stable metric 3D detection from monocular video. Code and models are available at royyang0714.github.io/Map-Det3D.

Keywords: Multi-view 3D Object Detection · Scene Understanding · Feed-Forward Metric 3D Reconstruction

1 Introduction

3D object detection (3DOD) is a fundamental capability for embodied AI, autonomous navigation, and augmented reality, where agents must reason about objects in metric 3D to act safely and effectively. In many deployed systems, this problem is made substantially easier by active depth sensors such as LiDAR or RGB-D cameras, which directly provide 3D geometry. However, these

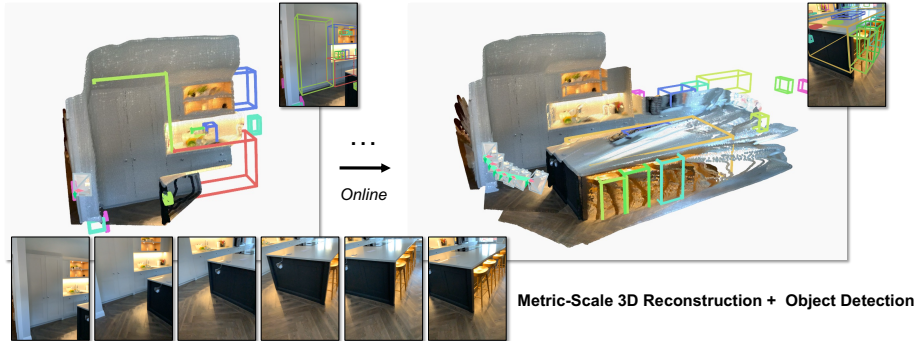


Fig. 1: Map-Det3D. We leverage the metric-scale geometry prior from a multi-view feed-forward 3D reconstruction model and adapts it for 3D object detection from images. In particular, given a streaming video input, Map-Det3D can perform online metric-scale 3D reconstruction and 3D object detection simultaneously in the form of feed-forward inference over a sliding window. Map-Det3D can run on videos alone, but camera poses and intrinsics can be added to further stabilize the geometry estimation.

sensors come with practical drawbacks: prohibitive hardware costs, high power consumption, and restrictive form-factor constraints that are difficult to satisfy on lightweight and consumer platforms. As a result, enabling accurate 3D object detection from images alone has become an increasingly desirable alternative.

A long-standing challenge is that, from a single image, depth and its absolute scale are inherently ambiguous. Consequently, monocular 3DOD methods [2, 46–49] typically follow a 2D-to-3D lifting recipe: they detect objects in the image plane and then regress 3D attributes such as depth, size, and orientation using learned priors. While effective in-distribution, this strategy is fragile in practice because small depth errors can dominate 3D overlap, and the learned scale prior may break under shifts in camera intrinsics, ego-motion patterns, or scene statistics. These issues become particularly pronounced when transferring across devices and environments, precisely the setting where image-only 3DOD is most attractive.

An appealing alternative is to perform detection directly in a 3D representation, where metric reasoning about size, occlusion, and spatial extent is explicit. In principle, operating in 3D mitigates scale ambiguity and reduces the burden on priors tied to the image plane. In real-world scenarios, however, 3D geometry is often noisy, incomplete, and sensor-dependent: point clouds vary widely in density, scale, and noise characteristics, and models trained on one sensing regime can fail to generalize to another. Moreover, compared to image-based pipelines, access to appearance cues is more indirect in 3D-only processing, and designing efficient mechanisms that provide 3D models with rich appearance information remains an active topic [12, 51].

Multi-view geometry offers a natural way to resolve single-view ambiguities by aggregating evidence across viewpoints. Recently, feed-forward 3D reconstruction (FF3R) has emerged as a powerful source of geometric inductive bias [11, 19, 40, 41].

Given a handful of input views, these models can robustly recover camera motion and dense geometry across diverse data, and some explicitly predict a disentangled metric scale factor. Importantly, FF3R enables fast, optimization-free 3D estimation, offering a practical alternative to offline global reconstruction pipelines while producing geometry that is both more stable than monocular lifting and less coupled to a specific depth sensor.

Motivated by these observations, we propose Map-Det3D, as shown in Fig. 1, an *online* multi-view 3D object detection framework that turns a FF3R model into the geometric encoder of a detection transformer [3, 46, 50], and predicts 3D boxes directly in the reconstructed 3D space from RGB images. This design aims to marry image-only systems with the geometric clarity of 3D reasoning, improving robustness to camera and domain shifts that often break the conventional 2D-to-3D lifting pipeline. Concretely, we extend MapAnything [11] for its camera-conditioned modeling and disentangled metric-scale prediction, and repurpose it as the encoder that produces 3D tokens for a detection transformer (DETR) style detection head [3, 50]. On top of this backbone, we introduce an *up-to-scale* 3D box head: it regresses boxes in an unscaled coordinate system and recovers metric outputs via the FF3R scale factor, thereby sidestepping the brittle depth regression inherent in 2D detections. To satisfy online constraints, Map-Det3D operates on a sliding temporal window and aggregates evidence causally with a multi-view transformer; camera intrinsics and, when available, poses provide explicit geometric cues that further stabilize depth and scale.

We focus on class-agnostic detection to isolate the geometric bottleneck and train Map-Det3D on the CA-1M [16] dataset, but the architecture is not inherently class-specific and can be extended to semantic categories using the same geometric backbone. Map-Det3D achieves the new state-of-the-art (SOTA) performance on the held-out scenes of CA-1M. Furthermore, we show the generalization ability of Map-Det3D on the zero-shot ScanNetV2 [5] benchmark.

To conclude, our contributions are threefold. **(1)** We introduce Map-Det3D, an RGB-only online multi-view indoor 3DOD framework that injects feed-forward metric reconstruction priors into a detection transformer to enable direct 3D reasoning from images. **(2)** We propose an up-to-scale, direct 3D bounding-box head that recovers metric scale through FF3R, avoiding the error-amplifying 2D-to-3D lifting pathway. **(3)** a detailed component analysis showing that the gains come from geometry-and-scale design choices (*i.e.* direct 3D prediction, temporal multi-view aggregation, and explicit camera conditioning), together with strong generalization across indoor benchmarks.

2 Related Work

2.1 Monocular 3D Object Detection

Estimating metric 3D bounding boxes from a single RGB image is widely studied in robotics and autonomous driving. Existing approaches span proposal-based pipelines, explicit geometric priors, and BEV-style formulations [7, 8, 17, 21, 23, 25,

27, 39, 43]. A recent trend is to improve robustness across domains, cameras, and label spaces. Cube R-CNN [2] introduces virtual depth to mitigate focal-length variability when mixing datasets, while Uni-MODE [18] uses domain confidence to jointly train BEV detectors across indoor and outdoor scenes. Open-vocabulary variants increasingly leverage 2D foundation models: OVM3D-Det [10] generates pseudo supervision for novel classes, and 3D-MOOD conditions detection on vision-language priors (*e.g.* GroundingDINO [22]) to reduce closed-set dependence. Nevertheless, monocular 3DOD remains fundamentally constrained by single-view ambiguity: depth and, in particular, metric scale are underconstrained, so accurate geometry often relies on learned priors that can be sensitive to domain shifts in intrinsics, motion, and scene statistics [1, 32].

2.2 Feed-Forward 3D Estimation

Feed-forward 3D estimation has improved substantially with large-scale pre-training, yielding single-view models with strong geometric representations and transfer across datasets and camera setups [9, 29, 30, 45]. Multi-view feed-forward estimators further reduce ambiguity by enforcing cross-view consistency and by jointly reasoning about camera motion and depth [11, 19, 40]. However, most of these approaches are scene-centric: they target camera motion and dense geometry, and are not optimized to produce object instances or object-centric representations. In addition, many reconstruction pipelines assume global aggregation over a set of views, whereas streaming perception must update predictions causally from a short temporal context and within limited compute.

2.3 Online 3D Perception

Online 3D perception exploits temporal continuity to reduce ambiguity in streaming settings. For geometry, prior works extend single-view predictors to video via propagation or temporal refinement [4, 31], and others adapt multi-view reconstruction to incremental pipelines that update structure as frames arrive [6, 53]. For object-centric reasoning, BoxFusion [15] combines pretrained 3D detectors with temporal persistence and post-processing, while EFM3D [37] fuses multiple modalities to predict 3D occupancy. Many online systems, however, either rely on additional sensors/modalities or retain a 2D-then-lift design, so it remains unclear how to obtain stable metric 3D boxes from RGB video alone under causal constraints.

Map-Det3D addresses this gap by operating on a sliding window of frames, *i.e.* treating time as additional views, repurposing MapAnything [11] as a geometry-and-scale encoder within a DETR-style detector, and decoding 3D boxes directly in 3D. This avoids depth sensors and a 2D-to-3D lifting stage, while leveraging metric-scale priors learned through feed-forward multi-view reconstruction.

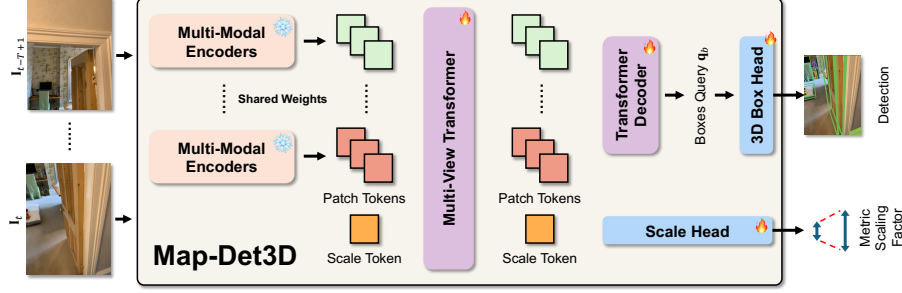


Fig. 2: Architecture overview. Map-Det3D treats a short temporal window as multi-view input, repurposes a metric FF3R backbone, *i.e.* MapAnything, as the encoder of a detection transformer, and decodes up-to-scale 3D boxes that are converted to metric scale via the predicted scale factor.

3 Method

Our goal is online 3D object detection in metric scale from streaming RGB, without depth sensors. One of the obstacles is the geometric component: from a single view, depth and absolute scale are underconstrained, and small range errors quickly dominate 3D overlap. We address this by injecting a metric geometric prior into the detector. Concretely, we repurpose a FF3R model as the geometric encoder of a detection transformer, and we introduce an up-to-scale 3D bounding box head whose outputs are mapped to metric units through a predicted scale factor. Fig. 2 summarizes the architecture. We first formalize the online detection setup and notation (Sec. 3.1), then describe the FF3R prior (Sec. 3.2), the detection architecture (Sec. 3.3), and the up-to-scale 3D head (Sec. 3.4), and finally discuss online inference (Sec. 3.5) and training losses (Sec. 3.6).

3.1 Preliminaries

We address the problem of online 3D object detection (3DOD) from a streaming monocular video $\mathcal{V} := \{\mathbf{I}_1, \mathbf{I}_2, \dots\}$, where we expect the model to produce metric-scale predictions causally, without depth sensors. Each frame $\mathbf{I}_t \in \mathbb{R}^{3 \times H \times W}$ may be accompanied by camera intrinsics $\mathbf{K}_t \in \mathbb{R}^{3 \times 3}$ and extrinsics $\mathbf{P}_t \in \mathbb{R}^{4 \times 4}$. When some of this metadata is unavailable, our FF3R backbone can provide pose and scale estimates, so the same formulation covers settings with complete, partial, or missing camera parameters. At each time step t , the detector observes a sliding window $\mathcal{W}_t := \{\mathbf{I}_{t-T+1}, \dots, \mathbf{I}_t\}$ of T frames. We denote by $\mathcal{C}_t$ the set of camera parameters available for the window. From $\mathcal{W}_t$ and $\mathcal{C}_t$, the model outputs a fixed-size set of M 3D oriented bounding boxes $\mathbf{B}_t^{3D}$ for the current frame. Each box in $\mathbf{B}_t^{3D}$ is parameterized by a 3D center (x, y, z) in the camera coordinate frame, physical dimensions (w, l, h) , an orientation $R \in \text{SO}(3)$, and a confidence score $\sigma \in [0, 1]$ that indicates whether it corresponds to a valid object. A central element of our design is a per-window scale factor ρ_t , predicted by the backbone, that converts up-to-scale predictions (denoted with a tilde, *e.g.* $\tilde{x}$) into metric

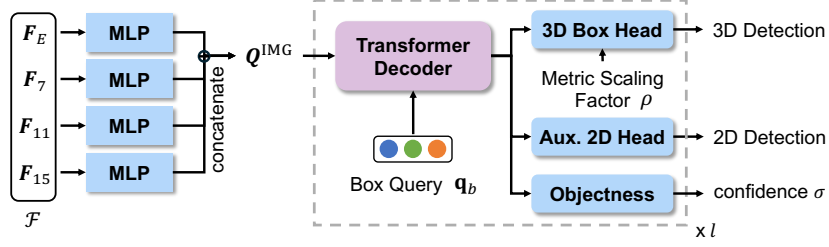


Fig. 3: Detection Architecture. We repurpose the FF3R as the detection transformer [3,50] encoder and project the multi-scale feature maps into a common dimension, then concatenate them to form $\mathbf{Q}^{\text{IMG}}$. We gradually refine bounding box queries and decode 3D bounding boxes using our proposed up-to-scale 3D bounding box head.

quantities (*e.g.* $x = \rho_t \tilde{x}$). We focus on the class-agnostic setting to isolate the geometric challenges of depth and scale from category recognition.

3.2 Metric Feed-Forward 3D Reconstruction Prior

We build on MapAnything [11] as our metric FF3R backbone for two reasons: it produces a rich multi-view representation, and it predicts a disentangled per-window scale factor ρ_t . Given the T frames in the current window $\mathcal{W}_t$ and the available camera parameters $\mathcal{C}_t$, a multi-modal encoder produces per-view patch tokens that are then processed, together with a learnable scale token q_{scale} , by a 16-layer multi-view transformer [40] that fuses information across views. We denote by $\mathbf{F}_E \in \mathbb{R}^{T \times 1536 \times P}$ the fused encoder output and by $\mathbf{F}_7, \mathbf{F}_{11}, \mathbf{F}_{15}$ the intermediate transformer features, each concatenated across all T views, where $P := HW/p^2$ is the number of spatial tokens per view and $p = 16$ is the patch size of the vision encoder. These form the multi-scale feature maps $\mathcal{F}_t := \{\mathbf{F}_E, \mathbf{F}_7, \mathbf{F}_{11}, \mathbf{F}_{15}\}$. An MLP then decodes the transformer-updated token corresponding to q_{scale} into the scale factor ρ_t . When ground-truth extrinsics are not included in $\mathcal{C}_t$, the backbone estimates them internally. In summary, the backbone mapping can be written as:

$$(\mathcal{F}_t, \rho_t) := \Phi_{\text{FF3R}}(\mathcal{W}_t, \mathcal{C}_t), \quad (1)$$

and its output is used by subsequent modules in our architecture. For brevity, we drop the time index t from $\mathcal{F}_t, \rho_t$, and all derived quantities in the following subsections, as the dependency is always through the input window.

3.3 Detection Architecture

To transfer the FF3R geometric prior to 3DOD, we build a detection transformer [3,46,50] on top of the multi-scale feature maps $\mathcal{F}$ as shown in Fig. 3. The T views are split apart, and each is processed independently as a separate batch element. For each view, the feature levels are projected along the channel dimension to a common size of 256 via a per-level MLP and concatenated along

the patch and scale dimensions into a single token sequence, forming per-view image features $\mathbf{Q}^{\text{IMG}} \in \mathbb{R}^{|\mathcal{F}|P \times 256}$. Dense 2D anchor proposals are generated on a regular grid at each feature level with scale-dependent default sizes. A shared classification and regression head scores all proposals. The top- M scoring positions are selected as initial reference points, and the corresponding encoder features, after stop-gradient and linear projection, form the initial object queries $\mathbf{Q}_0 \in \mathbb{R}^{M \times 256}$. This top- M selection determines the fixed output size introduced in Sec. 3.1.

A L -layer deformable decoder [50] then iteratively refines the queries. Each layer applies deformable cross-attention to $\mathbf{Q}^{\text{IMG}}$, self-attention among queries, and updates the reference points via residual 2D box refinement. We denote this encoder-decoder pipeline as Φ_{DET} , which outputs the queries $\mathbf{Q}_k$ and the associated 2D reference boxes $\mathbf{B}_k^{2\text{D}}$ at every layer k :

$$\{(\mathbf{Q}_k, \mathbf{B}_k^{2\text{D}})\}_{k=0}^L := \Phi_{\text{DET}}(\mathcal{F}). \quad (2)$$

Each $\mathbf{Q}_k$ is decoded into an intermediate 3D box prediction (Sec. 3.4), allowing progressive refinement across layers, while the 2D boxes $\mathbf{B}_k^{2\text{D}}$ provide the reference points for deformable attention and are used for bipartite matching during training (Sec. 3.6).

3.4 Up-to-scale 3D Bounding Box Head

As shown in Fig. 3, at every decoder layer k , we apply a layer-specific 3D bounding box head $\Phi_{3\text{D}}^k$ to the M queries $\mathbf{Q}_k$ to produce a set of M 3D bounding boxes with associated confidence scores:

$$(\mathbf{B}_k^{3\text{D}}, \sigma_k) := \Phi_{3\text{D}}^k(\mathbf{Q}_k, \rho), \quad k = 0, \dots, L. \quad (3)$$

The head receives the scale factor ρ because all geometric attributes are regressed in an up-to-scale parameterization and must be converted to metric outputs. The intermediate predictions $\mathbf{B}_0^{3\text{D}}, \dots, \mathbf{B}_{L-1}^{3\text{D}}$ enable deep supervision during training (Sec. 3.6), while the final output of the model is $\mathbf{B}_t^{3\text{D}} := \mathbf{B}_L^{3\text{D}}$ (reinstating the time index from Sec. 3.1). As introduced in Sec. 3.1, each predicted box in $\mathbf{B}_k^{3\text{D}}$ is characterized by a 3D center (x, y, z) in the camera coordinate frame, physical dimensions (w, l, h) , an orientation $R \in \text{SO}(3)$, and a confidence score $\sigma \in [0, 1]$. A dedicated two-layer MLP regresses each geometric attribute in an up-to-scale parameterization, which is then converted to metric scale via the scale factor ρ .

For the 3D center, the head regresses up-to-scale coordinates $\tilde{x}, \tilde{y}$ in camera space and a log-depth $\tilde{d}$, from which the metric center is recovered as $x := \rho \tilde{x}$, $y := \rho \tilde{y}$, and $z := \rho \exp(\tilde{d})$. Similarly, the head regresses up-to-scale log-dimensions $\tilde{s}_w, \tilde{s}_l, \tilde{s}_h$, from which the metric sizes are recovered as $w := \rho \exp(\tilde{s}_w)$, $l := \rho \exp(\tilde{s}_l)$, and $h := \rho \exp(\tilde{s}_h)$. For orientation, following Cube R-CNN [2], we regress a continuous 6D rotation representation [52] in allocentric form and convert it to egocentric using the predicted 3D center direction [14], unlike CuTR [16], which assumes all bounding boxes are gravity-aligned. Finally, a single-layer MLP

predicts a binary objectness logit yielding the confidence score σ . Each query also carries the 2D reference box $\mathbf{B}_k^{2D}$ from the decoder, which is used for bipartite matching during training (Sec. 3.6).

3.5 Online Setting

Inference. Map-Det3D operates causally with a sliding window. At time t , it uses the current frame and the previous $T-1$ frames as multi-view input and outputs detections only for the current frame. This reuses MapAnything’s multi-view fusion for temporal reasoning without modifying the backbone’s architecture.

Training. Following VGGT [40], we randomize the number of views by sampling uniformly from 1 to the maximum window size T , which improves robustness to varying temporal context.

3.6 Training Losses

We combine auxiliary 2D losses for matching with disentangled 3D losses for geometry. We compute Hungarian matching [13] based on the auxiliary 2D box predictions, and supervise matched pairs with focal loss [20] for classification and L_1 +GIoU [33] for 2D box regression. For 3D geometry, we use a disentangled corner loss as in Cube R-CNN [2]. For each attribute, we form the eight 3D box corners using the prediction for that attribute and ground truth for all others. This yields $\mathcal{L}_{xy}$ (center), $\mathcal{L}_z$ (depth), $\mathcal{L}_{\text{dim}}$ (dimensions), and $\mathcal{L}_{\text{rot}}$ (rotation), where $\mathcal{L}_{\text{rot}}$ is a Chamfer distance between corner sets to handle symmetries. The remaining terms use L_1 . We apply deep supervision to the encoder proposal layer and all decoder layers, giving the total per-frame loss:

$$\mathcal{L}_{\text{total}} := \sum_{k=0}^L (\mathcal{L}_{2D}^k + \mathcal{L}_{3D}^k), \quad \mathcal{L}_{3D}^k := \mathcal{L}_{xy}^k + \mathcal{L}_z^k + \mathcal{L}_{\text{dim}}^k + \mathcal{L}_{\text{rot}}^k. \quad (4)$$

The final training objective is the expectation of $\mathcal{L}_{\text{total}}$ over all sampled frames.

4 Experiments

We first discuss the implementation details for model training and testing in Sec. 4.1. We then describe the experimental setup in Sec. 4.2, including the training and testing data and the evaluation metrics. In Sec. 4.3, we take a deep look into Map-Det3D through extensive ablation studies, and in Sec. 4.4, we compare Map-Det3D with other state-of-the-art (SOTA) models on both in-domain and out-of-domain benchmarks. In Sec. 4.5, we extend Map-Det3D with a simple 3D object tracking algorithm to directly compare our method on an online per-scene evaluation benchmark. Finally, we provide the qualitative results in Sec. 4.6.

4.1 Implementation Details

We implement Map-Det3D in PyTorch [28] and CUDA [26]. We train the full model for $100k$ steps with a batch size of 64, *i.e.*, a maximum of 4 samples per GPU on 16 RTX 4090s for 1.5 days. For ablation studies in Sec. 4.3, we train the model for $50k$ steps, also with a batch size of 64. We set the initial learning rate to 0.0001 and apply the Cosine Annealing learning rate scheduler. To fully leverage the multi-view transformer for temporal information for 3DOD, we unfreeze the multi-view transformer and the scale head by setting their learning rate to $1/10$ of the initial learning rate. For input image resolutions, we follow MapAnything by selecting the closest image ratio from the fixed mapping. Instead of resizing the long edge of the image and performing center cropping, we resize the short edge and pad with zeros to preserve global information. During training, we randomly select the number of views from 1 to 5 and the aspect ratio, and randomly select the sampling rate from 2 to 10 FPS as part of the data augmentation. During inference time, we set $T = 5$.

4.2 Experimental Setup

CA1M [16]. We use the dataset to train Map-Det3D and evaluate on validation for an in-domain benchmark. Unlike previous datasets that focus on a limited set of predefined object categories, CA-1M provides exhaustive, class-agnostic 9-DOF 3D bounding box annotations. It features over 400,000 unique 3D objects across more than 1,000 highly accurate, laser-scanned indoor scenes. These high-fidelity 3D annotations are registered to over 3,500 handheld, egocentric video captures, yielding approximately 13 million training frames and 1.8 million validation frames. The pixel-accurate alignment of metric 3D boxes across continuous video sequences makes CA-1M a good fit for training Map-Det3D to learn consistent metric scale and temporal stability under causal, moving-camera conditions.

ScanNetV2 [5]. We further validate the generalization and robustness of our proposed architecture, given its widespread adoption as a benchmark. ScanNetV2 contains 1,513 densely annotated indoor scenes standardized into 1,201 scenes for training and 312 for validation. The dataset provides continuous RGB video streams paired with ground-truth 3D bounding box annotations with 18 core (ScanNetV2) and 198 long-tailed semantic categories (ScanNet200). We follow BoxFusion [15] to use the selected 100 scenes and uniformly sample every 25 frames as the zero-shot benchmark. We use the challenging ScanNet200 settings for per-frame 3DOD evaluation and the original ScanNetV2, *i.e.* 18 classes, as the per-scene evaluation to be aligned with BoxFusion.

Evaluation Metrics. We use the average precision (AP) and average recall (AR) metrics to evaluate the performance of 2D and 3D detection results. We match the predictions and GT by computing the intersection-over-union (IoU_{3D}) of 3D cuboids. The mean 3D AP is reported in a class-agnostic way, while AP_{15} , AP_{25} , and AP_{50} correspond to IoU_{3D} thresholds 0.15, 0.25 and 0.50, respectively. It is worth noting that, unlike in previous benchmark [2, 46], we evaluate *all* ground truth bounding boxes regardless of their visibility and truncation.

Table 1: Model Analysis. We conduct extensive ablation studies on Map-Det3D to analyze the effectiveness of each proposed design on the CA-1M validation set. **Multi-view** stands for using online temporal information, **MV Transformer** stands for multi-view transformer in MapAnything. **K** and **Pose** denote whether encoding the camera intrinsics and poses in the multi-modal encoder, respectively.

	Multi-View	Unfreeze Scale Head	Unfreeze MV Transformer	K	Pose	AP ₁₅
1	-	-	-	-	-	11.7
2	✓	-	-	-	-	11.8
3	✓	✓	-	-	-	11.9
4	-	✓	✓	-	-	14.5
5	✓	✓	✓	-	-	17.2
6	✓	-	-	✓	-	12.6
7	✓	-	-	✓	✓	17.3
8	✓	✓	✓	✓	✓	21.2

4.3 Model Analysis

We first provide a detailed analysis of Map-Det3D in Tab. 1 and Tab. 2 to assess the effectiveness of our model designs. Then, we analyze the efficiency in Tab. 3.

Feed-Forward 3D Reconstruction Prior. We first ablate the different settings for leveraging the FF3R prior. In particular, we analyze the exploitation of the multi-view prior *vs.* single-view and whether the prior components, *i.e.* scale prediction and multi-view feature aggregator, are either frozen or trainable. Enabling multi-view processing without further adapting the FF3R prior yields only a negligible gain (row 1 $\rightarrow$ row 2), which indicates that directly using the multi-view transformer to aggregate temporal information for 3DOD is not straightforward. Unfreezing the scale head alone also has a limited impact (row 2 $\rightarrow$ row 3), indicating that the original scale prediction prior does not require further fine-tuning on CA-1M.

In contrast, fine-tuning the multi-view transformer is important for detection: unfreezing it leads to a clear improvement (row 3 $\rightarrow$ row 4). Under the same fine-tuning setup, adding multi-view input provides further gains (row 4 $\rightarrow$ row 5), supporting the role of temporal context beyond parameter updates. These clear improvements highlight that, although the geometric prior provides strong 3D information, it lacks sufficient object-centric features that must be trained to yield the largest gain.

Camera conditioning. We further show how Map-Det3D can benefit from camera conditioning on FF3R. As shown in row 2 $\rightarrow$ row 6 of Tab. 1, conditioning the camera intrinsics to MapAnything’s multi-modal encoder moderately improves performance by 0.8 AP₁₅. Adding camera poses yields a further substantial boost of 4.7 AP₁₅ (row 6 $\rightarrow$ row 7), which shows that the Map-Det3D can successfully leverage the FF3R prior and perform better 3DOD accordingly.

Finally, enabling all components and training the full model achieves the best result, with a considerable gap of 9.5 AP₁₅ w.r.t. the baseline (row 1 $\rightarrow$ row 8). The final model shows how our design contributed with a different set

Table 2: 3D head ablation. We ablate the proposed 3D head under the multi-view setting while the scale head and multi-view transformer are frozen.

Direct 3D	AP ₁₅ ^{CA1M}	AP ₁₅ ^{ScanNet}
-	15.6	10.8
✓	17.3	12.6

Table 3: FPS and peak GPU memory. * stands few-view settings (offline), *i.e.* strides = 5.

T	GPU Mem (G)	FPS	AP ₁₅ ^{CA1M}	AP ₁₅ ^{ScanNet}
1	5.8	14.3	14.0	11.1
3	6.1	11.1	20.5	13.1
5	6.4	8.3	<i>21.2</i>	<i>13.3</i>
7	6.7	6.3	21.1	<i>13.3</i>
5*	6.4	5.8	22.2	13.5

Table 4: Results on CA-1M [16]. We compare Map-Det3D against different detector types on the challenging CA-1M validation set. The results show that our design can fully leverage the FF3R prior and not just outperform all monocular baselines but also outperform an offline multi-view detector.

Method	Detector Type	AP ₂₅ ↑	AR ₂₅ ↑	AP ₅₀ ↑	AR ₅₀ ↑
FCAF [34]	Point Cloud	29.3	49.5	11.2	22.6
TR3D [36]	Point Cloud	22.0	49.5	4.4	20.0
TR3D + FF [36]	Point Cloud	24.8	52.9	4.7	21.0
ImVoxelNet [35]	Multi-View	10.1	22.8	2.3	6.3
Cube R-CNN [2]	Monocular	4.6	20.1	1.0	4.7
CuTR [16]	Monocular	13.5	35.4	2.4	12.9
Map-Det3D (Ours)	Online Multi-view	16.9	40.3	3.5	14.8

of complementary contributions: metric-scale priors, temporal aggregation, and explicit camera cues.

3D Bounding Box Head. By repurposing MapAnything as the detection transformer encoder, Map-Det3D can leverage the strong geometric prior, thereby bypassing the 2D-to-3D lifting from 2D pixel-space detection. Tab. 2 confirm its effectiveness by improving both AP₁₅ for both in-domain and out-of-domain settings.

Efficiency. We provide the temporal ablation along with the runtime and cost in Tab. 3, which shows that Map-Det3D benefits from the temporal contexts. Given the maximum $T = 5$ during training, testing with $T = 7$ leads to a minor degradation (-0.1 AP₁₅) on CA1M (in-domain), yet we attain the same performance on ScanNet (out-of-domain). We further conduct the few-view setting (last row), *i.e.* strides = 5, which obtains T -frame detection results at once. This offline setting leads to slightly better results, while our online sliding-window design shows competitive performance.

4.4 State-of-the-art Comparisons

In-Domain Benchmark. Tab. 4 reports results on CA-1M validation sets, *i.e.* the scenes are not seen during training. Map-Det3D achieves 16.9 AP₂₅ and 3.5 AP₅₀, improving over monocular baselines such as CuTR [16] (13.5 / 2.4) and Cube R-CNN [2] (4.6 / 1.0). Furthermore, Map-Det3D also surpasses the offline multi-view baseline ImVoxelNet [35] (10.1 / 2.3) while operating online with a sliding window. This demonstrates the effectiveness of our model design in

Table 5: Results on ScanNet200 [5]. We test the generalization ability of Map-Det3D compared to other monocular open-vocabulary 3DOD methods [46–49]. The results demonstrate that, despite being trained on less diverse data, Map-Det3D still outperforms all other methods. The comparison with CuTR further demonstrates that the improved performance is not due to the CA-1M training data, but rather to our model design.

Method	Training Data	AP ₁₅ ↑	AP ₂₅ ↑
OVMono3D [47]	Omni3D [2]	11.7	5.7
LabelAny3D [48]	MS-COCO 3D [48]	11.2	4.9
3D-MOOD [46]	Omni3D [2]	11.2	8.0
DetAny3D [49]	DA3D [49]	11.7	8.0
CuTR [16]	CA-1M [16]	4.3	2.1
Map-Det3D (Ours)		15.2	9.7

leveraging multi-view information via FF3R’s priors. Despite depth/point-cloud methods remaining stronger (*e.g.* FCAF [34]: 29.3 / 11.2), as expected given their access to metric measurements, Map-Det3D closes the gap w.r.t. TR3D (22.0 / 4.4), demonstrating the strengths of our designs.

Zero-shot benchmark. In Tab. 5, we evaluate all methods under the class-agnostic setting on the ScanNet200 validation set without fine-tuning. Relative to open-set monocular baselines trained on other sources, Map-Det3D achieves the best zero-shot performance on AP₁₅ and AP₂₅. This demonstrates the generalization ability even though the model is trained only on a single source dataset, *i.e.* CA-1M. We further test CuTR, the only other 3D object detector trained on CA-1M to our best knowledge, on ScanNetV2. While Map-Det3D reaches 15.2 AP₁₅ and 9.7 AP₂₅, CuTR only achieve 4.3 AP₁₅. This indicates that Map-Det3D’s superior performance is not obtained by CA-1M training data but rather by our model design.

4.5 Per-scene Evaluation

As we conduct per-frame evaluation in Tab. 4 and Tab. 5, we also follow the setting in BoxFusion [15] to have per-scene results. To achieve this, instead of using the optimization process proposed in BoxFusion, we follow the widely used tracking-by-detection paradigm [7, 8] and track the 3D detection results frame by frame in an online manner.

We maintain an *online* track memory $\mathcal{T}$, and for each track $\tau \in \mathcal{T}$, it contains the detected 3D object in its corresponding camera coordinate and the camera pose. For each new frame detection, *i.e.* observation, we first transform both the observation and the detected boxes in track memory to world coordinates, and then compute the 3D IoU between them as the affinity matrix. Then we will run the greedy assignment to determine whether the new detection should be assigned to a new track or associated with an existing one. Finally, we will replace the bounding box of the associated track with the new observation’s bounding box if it is larger. As shown in Fig. 4, we speculate that when two objects are

Table 6: Zero-shot class-agnostic per-scene evaluation on ScanNetV2. We follow BoxFusion [15] and run Map-Det3D with a simple 3D IoU based association algorithm. Our method achieves new SOTA results without using ground-truth depth.

Method	Online	No GT Depth	AP ₁₅ ↑	AP ₂₅ ↑	AP ₅₀ ↑
EmbodiedSAM [42]	-	-	8.2	5.2	1.4
OnlineAnySeg [38]	✓	-	24.2	18.3	5.2
BoxFusion [15]	✓	-	29.2	24.6	8.0
SpatialLM [24]	-	✓	9.0	4.9	0.6
BoxFusion (RGB Only) [15]	✓	✓	18.5	11.8	1.0
Ours	✓	✓	27.6	22.7	6.2

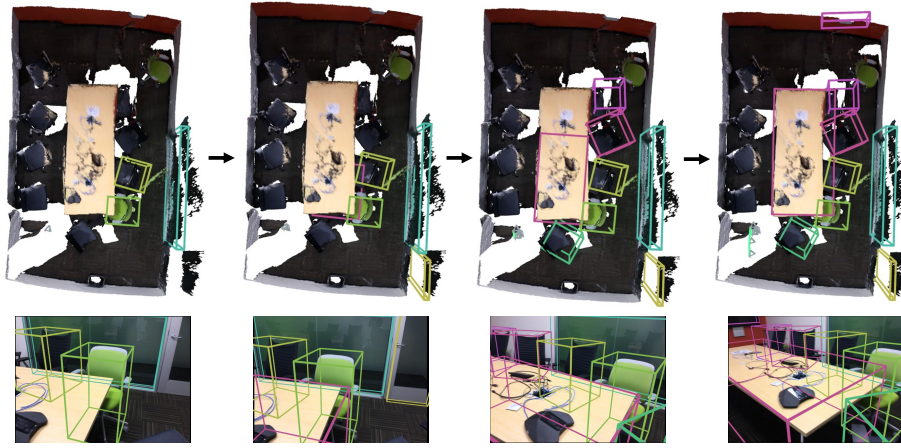


Fig. 4: The effect of the visible part inside the images. The table is only partially observed when $T = 1$. But as the camera rotates, the model can gradually see the full table. Hence, our tracking design can successfully use the better detection results from Map-Det3D as the per-scene results.

associated in 3D space over time, the larger box usually corresponds to the more visible part in the image, and we update the track accordingly.

We report the zero-shot per-scene evaluation results on ScanNetV2 in Tab. 6. Map-Det3D with simple 3D IoU based association yields better performance than the previous one that did not use ground truth depth methods. Moreover, we achieve better results on AP₁₅ than EmbodiedSAM and OnlineAnySeg when they use GT depth information. For stricter evaluation settings, *i.e.* AP₂₅ and AP₅₀, we still have comparable performance against point-cloud-based methods. This demonstrates that Map-Det3D successfully leverages the FF3R prior and achieves new state-of-the-art (SOTA) performance on both per-frame and per-scene evaluations.

4.6 Qualitative Results

In Fig. 5, we qualitatively compare Map-Det3D with several baselines on the ScanNetV2 validation set (out-of-domain). We overlay predicted 3D bounding

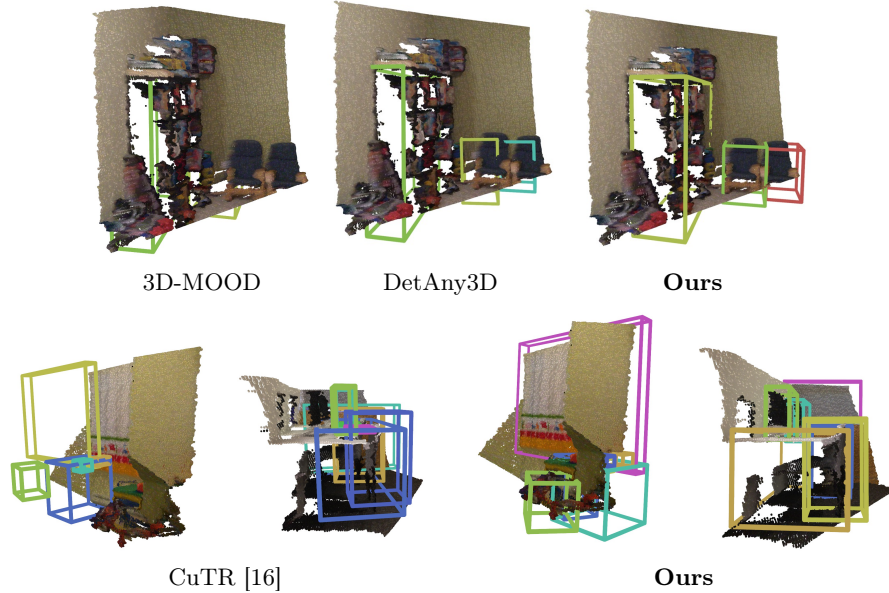


Fig. 5: Qualitative comparison on ScanNet. We qualitatively compare Map-Det3D with, 3D-MOOD [46], DetAny3D [49], and CuTR [16]. We visualize the predicted 3D bounding boxes alongside the ground-truth point cloud to qualitatively assess the accuracy of metric-scale prediction. The results demonstrate that, compared to the baselines, Map-Det3D successfully detects objects given unseen scenes.

boxes and visualize them together with the ground-truth scene geometry (shown as a point cloud) to assess box placement, scale, and orientation in 3D. On CA-1M, CuTR commonly exhibits monocular failure modes such as depth offsets and scale drift, which lead to boxes that are visibly too large/small or shifted along the viewing direction. Map-Det3D produces boxes that better match the scene geometry in these cases, consistent with using a sliding-window view aggregation and an explicit metric scale factor from the FF3R prior. Under zero-shot transfer to ScanNetV2, CuTR’s predictions degrade further, with larger scale and depth errors, whereas Map-Det3D typically preserves more coherent metric scale and maintains plausible 3D alignment under domain, *e.g.* appearance and camera shift. We provide additional qualitative results on ScanNetV2 in Fig. 6 and more in the supplementary material.

5 Limitations

Despite the fact that we are using MapAnything as our detection transformer encoder, given the limitation of resources, we only train Map-Det3D on CA-1M. This leads to the limitation that Map-Det3D works well only for indoor scenes.

Another constraint is the categorization of detected objects: since we focus on the geometric challenge of 3DOD from images, Map-Det3D’s detections are

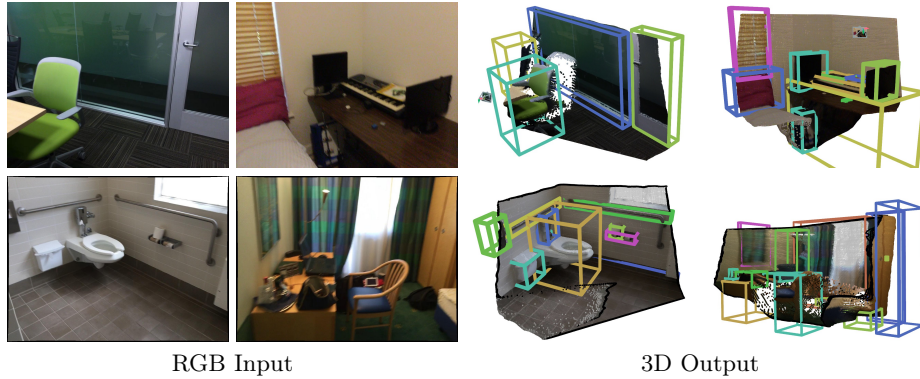


Fig. 6: Qualitative Results on ScanNetV2. We visualize the 3D bounding boxes plus depth estimation from Map-Det3D under the zero-shot settings.

not put in relation to any closed or open-vocabulary queries. We leave open-vocabulary as future work to be explored as a natural extension of our proposed class-agnostic 3DOD model using the FF3R prior. Potentially, the class-agnostic detections from Map-Det3D can also be semantically matched to queries using simple set-of-marks prompting in the input views with a VLM [44].

6 Conclusion

In this paper, we introduce Map-Det3D, a novel framework for online multi-view 3D object detection from streaming monocular inputs. We demonstrate that adapting a metric feed-forward 3D reconstruction as a geometric prior effectively addresses the inherent depth and scale ambiguities that traditionally limit monocular vision. By repurposing FF3R as the detection transformer encoder, we bypass the conventional 2D-to-3D lifting paradigm and propose an up-to-scale 3D detection head that optimally consumes these strong geometric priors to directly predict accurate metric-scale 3D bounding boxes and yield better performance. Map-Det3D achieves new SOTA performance on both in-domain and out-of-domain benchmarks, demonstrating our method’s ability to successfully detect indoor objects using strong FF3R priors.

Acknowledgement

This research was partially funded by the ETH AI Center Postdoc Fellowship (S. Hong), ETH Foundation Project 2025-FS-352 (Z. Bauer), the SNSF Advanced Grant 216260 (Z. Bauer), Swiss AI Initiative from the Swiss National Supercomputing Centre (CSCS) grant under project ID a144 (Y-H. Yang), and the Lamarr Institute for Machine Learning and Artificial Intelligence (H. Blum).

References

1. Bochkovskii, A., Delaunoy, A., Germain, H., Santos, M., Zhou, Y., Richter, S.R., Koltun, V.: Depth pro: Sharp monocular metric depth in less than a second. ICLR (2025)
2. Brazil, G., Kumar, A., Straub, J., Ravi, N., Johnson, J., Gkioxari, G.: Omni3D: A large benchmark and model for 3D object detection in the wild. In: CVPR. IEEE, Vancouver, Canada (June 2023)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV. pp. 213–229. Springer (2020)
4. Chen, S., Guo, H., Zhu, S., Zhang, F., Huang, Z., Feng, J., Kang, B.: Video depth anything: Consistent depth estimation for super-long videos. CVPR (2025)
5. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR (2017)
6. Duisterhof, B.P., Zust, L., Weinzaepfel, P., Leroy, V., Cabon, Y., Revaud, J.: Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. In: 3DV. pp. 1–10 (2025)
7. Fischer, T., Yang, Y.H., Kumar, S., Sun, M., Yu, F.: Cc-3dt: Panoramic 3d object tracking via cross-camera fusion. In: CORL (2022)
8. Hu, H.N., Yang, Y.H., Fischer, T., Darrell, T., Yu, F., Sun, M.: Monocular quasi-dense 3d object tracking. IEEE TPAMI **45**(2), 1992–2008 (2022)
9. Hu, M., Yin, W., Zhang, C., Cai, Z., Long, X., Chen, H., Wang, K., Yu, G., Shen, C., Shen, S.: Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. IEEE TPAMI (2024)
10. Huang, R., Zheng, H., Wang, Y., Xia, Z., Pavone, M., Huang, G.: Training an open-vocabulary monocular 3d detection model without 3d data. In: NeurIPS (2024)
11. Keetha, N., Müller, N., Schönberger, J., Porzi, L., Zhang, Y., Fischer, T., Knapitsch, A., Zauss, D., Weber, E., Antunes, N., Luiten, J., Lopez-Antequera, M., Bulò, S.R., Richardt, C., Ramanan, D., Scherer, S., Kotschieder, P.: MapAnything: Universal feed-forward metric 3D reconstruction. In: 3DV. IEEE (2026)
12. Knaebel, K., Yilmaz, K., de Geus, D., Hermans, A., Adrian, D., Linder, T., Leibe, B.: DINO in the room: Leveraging 2D foundation models for 3D segmentation. In: 2026 International Conference on 3D Vision (3DV) (2026)
13. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)
14. Kundu, A., Li, Y., Rehg, J.M.: 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3559–3568 (2018)
15. Lan, Y., Zhu, C., Gao, Z., Zhang, J., Cao, Y., Yi, R., Wang, Y., Xu, K.: Boxfusion: Reconstruction-free open-vocabulary 3d object detection via real-time multi-view box fusion. In: Computer Graphics Forum. vol. 44, p. e70254. Wiley Online Library (2025)
16. Lazarow, J., Griffiths, D., Kohavi, G., Crespo, F., Dehghan, A.: Cubify anything: Scaling indoor 3d object detection. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 22225–22233 (2025)
17. Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Qiao, Y., Dai, J.: Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers. arXiv preprint arXiv:2203.17270 (2022)

18. Li, Z., Xu, X., Lim, S., Zhao, H.: Unimode: Unified monocular 3d object detection. In: CVPR. pp. 16561–16570 (June 2024)
19. Lin, H., Chen, S., Liew, J.H., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
20. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV. pp. 2980–2988 (2017)
21. Lin, X., Lin, T., Pei, Z., Huang, L., Su, Z.: Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. arXiv preprint arXiv:2211.10581 (2022)
22. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. ECCV (2024)
23. Liu, Y., Wang, T., Zhang, X., Sun, J.: Petr: Position embedding transformation for multi-view 3d object detection. arXiv preprint arXiv:2203.05625 (2022)
24. Mao, Y., Zhong, J., Fang, C., Zheng, J., Tang, R., Zhu, H., Tan, P., Zhou, Z.: Spatiallm: Training large language models for structured indoor modeling. In: NeurIPS (2025)
25. Naiden, A., Paunescu, V., Kim, G., Jeon, B., Leordeanu, M.: Shift r-cnn: Deep monocular 3d object detection with closed-form geometric constraints. In: ICIP. pp. 61–65. IEEE (2019)
26. Nickolls, J., Buck, I., Garland, M., Skadron, K.: Scalable parallel programming with cuda: Is cuda the parallel programming model that application developers have been waiting for? Queue **6**(2), 40–53 (2008)
27. Park, D., Ambrus, R., Guizilini, V., Li, J., Gaidon, A.: Is pseudo-lidar needed for monocular 3d object detection? In: ICCV (2021)
28. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: NeurIPS, pp. 8024–8035. Curran Associates, Inc. (2019)
29. Piccinelli, L., Sakaridis, C., Segu, M., Yang, Y.H., Li, S., Abbeloos, W., Van Gool, L.: UniK3D: Universal camera monocular 3d estimation. In: CVPR (2025)
30. Piccinelli, L., Sakaridis, C., Yang, Y.H., Segu, M., Li, S., Abbeloos, W., Gool, L.V.: UniDepthV2: Universal monocular metric depth estimation made simpler. IEEE TPAMI (2026)
31. Piccinelli, L., Wandel, T., Sakaridis, C., Abbeloos, W., Gool, L.V.: Video depth propagation. In: 3DV (2026)
32. Piccinelli, L., Yang, Y.H., Sakaridis, C., Segu, M., Li, S., Van Gool, L., Yu, F.: UniDepth: Universal monocular metric depth estimation. In: CVPR (2024)
33. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression. In: CVPR. pp. 658–666 (2019)
34. Rukhovich, D., Vorontsova, A., Konushin, A.: Fcaf3d: Fully convolutional anchor-free 3d object detection. In: ECCV. pp. 477–493. Springer (2022)
35. Rukhovich, D., Vorontsova, A., Konushin, A.: Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. In: WACV. pp. 2397–2406 (2022)
36. Rukhovich, D., Vorontsova, A., Konushin, A.: Tr3d: Towards real-time indoor 3d object detection. In: ICIP. pp. 281–285. IEEE (2023)

37. Straub, J., DeTone, D., Shen, T., Yang, N., Sweeney, C., Newcombe, R.: Efm3d: A benchmark for measuring progress towards 3d egocentric foundation models. arXiv preprint arXiv:2406.10224 (2024)
38. Tang, Y., Zhang, J., Lan, Y., Guo, Y., Dong, D., Zhu, C., Xu, K.: Onlineanyseg: Online zero-shot 3d segmentation by visual foundation model guided 2d mask merging. arXiv preprint arXiv:2503.01309 (2025)
39. Tu, T., Chuang, S.P., Liu, Y.L., Sun, C., Zhang, K., Roy, D., Kuo, C.H., Sun, M.: Imgeonet: Image-induced geometry-aware voxel representation for multi-view 3d object detection. In: ICCV. pp. 6996–7007 (2023)
40. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: CVPR. pp. 5294–5306 (2025)
41. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: Pi³: Permutation-equivariant visual geometry learning. In: ICLR (2026)
42. Xu, X., Chen, H., Zhao, L., Wang, Z., Zhou, J., Lu, J.: Embodiedsam: Online segment any 3d thing in real time. arXiv preprint arXiv:2408.11811 (2024)
43. Yang, C., Chen, Y., Tian, H., Tao, C., Zhu, X., Zhang, Z., Huang, G., Li, H., Qiao, Y., Lu, L., et al.: Bevformer v2: Adapting modern image backbones to bird’s-eye-view recognition via perspective supervision. In: CVPR. pp. 17830–17839 (2023)
44. Yang, J., Zhang, H., Li, F., Zou, X., Li, C., Gao, J.: Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. arXiv preprint arXiv:2310.11441 (2023)
45. Yang, L., Kang, B., Huang, Z., Zhao, Z., Xu, X., Feng, J., Zhao, H.: Depth anything v2. arXiv:2406.09414 (2024)
46. Yang, Y.H., Piccinelli, L., Segu, M., Li, S., Huang, R., Fu, Y., Pollefeys, M., Blum, H., Bauer, Z.: 3d-mood: Lifting 2d to 3d for monocular open-set object detection. In: ICCV. pp. 7429–7439 (October 2025)
47. Yao, J., Gu, H., Chen, X., Wang, J., Cheng, Z.: Open vocabulary monocular 3d object detection. arXiv preprint arXiv:2411.16833 (2024)
48. Yao, J., Redoy, R.M., Elbaum, S., Dwyer, M.B., Cheng, Z.: Labelany3d: Label any object 3d in the wild. arXiv preprint arXiv:2601.01676 (2026)
49. Zhang, H., Jiang, H., Yao, Q., Sun, Y., Zhang, R., Zhao, H., Li, H., Zhu, H., Yang, Z.: Detect anything 3d in the wild. arXiv preprint arXiv:2504.07958 (2025)
50. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605 (2022)
51. Zhang, Y., Wu, X., Lao, Y., Wang, C., Tian, Z., Wang, N., Zhao, H.: Concerto: Joint 2d-3d self-supervised learning emerges spatial representations. arXiv preprint arXiv:2510.23607 (2025)
52. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5745–5753 (2019)
53. Zhuo, D., Zheng, W., Guo, J., Wu, Y., Zhou, J., Lu, J.: Streaming 4d visual geometry transformer. ICLR (2026)