Calibration-Free View-Agnostic Monocular 3D Object Detection for Urban Scenes

Mehmet Kerem Turkcan, Devika Gumaste, Zoran Kostic
Columbia University, New York, USA
{mkt2126,dg3370,zk2172}@columbia.edu

Abstract

Cooperative vehicle-to-everything (V2X) perception requires 3D object detection across heterogeneous cameras whose intrinsic parameters may be unavailable, imprecise, or drifting. We present UrbanOmniDetect, a calibration-free monocular 3D object detection framework that predicts ordered 2D projections of 3D bounding box vertices from a single RGB image. By formulating 3D detection as keypoint regression within a backbone-agnostic single-stage architecture, a single model generalizes across ego-vehicle, infrastructure, and aerial viewpoints without camera intrinsics or scene priors. We construct the UrbanOmniView dataset by unifying KITTI, DAIR-V2X, and high-fidelity Unreal Engine 5 synthetic data spanning ground-level, traffic-surveillance, and drone perspectives. A homography-based bird's-eye-view head maps predicted ground-contact keypoints to a top-down plane, enforcing geometric consistency without camera parameters. We experiment with YOLO11 backbone variants at multiple scales and augmented feature pyramid levels. On the KITTI benchmark, our best model achieves $AP_{3D} = 30.71$ *(Moderate) and* $AP_{BEV} = 35.19$ *at IoU≥0.7, outperforming calibration-dependent baselines on the Moderate and Hard splits, with an* $mAP_{50:95}$ *of 0.751 and 10 ms inference on an A100 GPU. Calibration-dependent baselines degrade catastrophically under small intrinsic perturbations, whereas our formulation is invariant by construction. UrbanOmniDetect provides a deployment-ready framework for autonomous driving, drone surveillance, and V2X cooperative perception.*

1. Introduction

3D object detection is a challenging computer vision problem with wide-ranging applications in autonomous driving, robotics, and surveillance systems. While significant progress has been made using specialized hardware like LiDAR and multi-camera setups especially in autonomous driving settings, these solutions introduce substantial cost, calibration complexity, and computational overhead. Monocular 3D detection from standard RGB images is a compelling alternative but faces fundamental challenges in accurately recovering depth information and object pose from a single viewpoint.

Traditional monocular 3D detection approaches have predominantly focused on specific camera perspectives, particularly ego-vehicle viewpoints, limiting their versatility across diverse deployment scenarios. These methods typically rely on explicit camera intrinsics, depth estimation modules, or ground-plane assumptions that fail to generalize across varied viewing angles. Further, existing frameworks struggle with generalization across ground-level, elevated traffic camera, and drone perspectives.

We address these limitations by introducing UrbanOmniDetect, **a calibration-free framework for monocular 3D object detection across arbitrary camera perspectives**. Unlike prior keypoint-based 3D detectors such as RTM3D [11], PerspectiveNet [7], and Tekin et al. [26], all of which require camera intrinsics at inference and are validated only on single-viewpoint ego-vehicle datasets, our method operates entirely in 2D image space. The backbone-agnostic architecture takes a raw RGB frame as input and directly predicts, for each object, the eight ordered 2D projections of its 3D bounding box corners, encoding 6D object pose without camera parameters. We experiment with the YOLO11 family [9] and augmented feature pyramid levels.

A key technical contribution is our homography-based bird's-eye-view (BEV) head, which maps predicted ground-contact keypoints to a top-down plane by optimizing a geometric orthogonality constraint across all detections in a frame. This produces calibration-free BEV layouts without requiring camera extrinsics, depth estimation, or precomputed ground surfaces.

To enable cross-view generalization, we constructed the UrbanOmniView dataset by unifying KITTI [5], DAIR-V2X [31], and a new synthetic dataset generated in Unreal Engine 5 [4] with ray-traced rendering at 4K resolution. This combination spans ego-vehicle, infrastructure-mounted, and aerial drone viewpoints representative of the

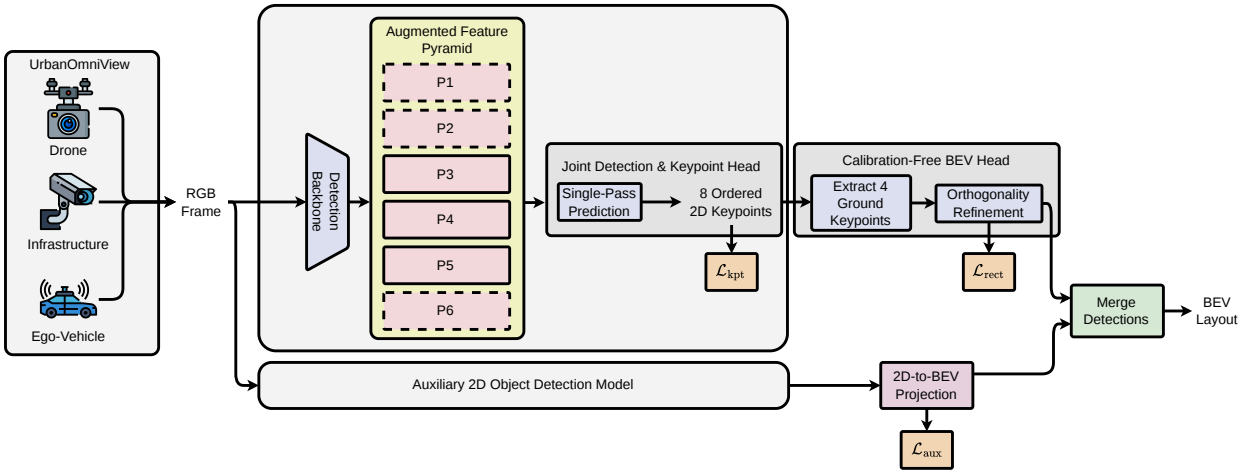
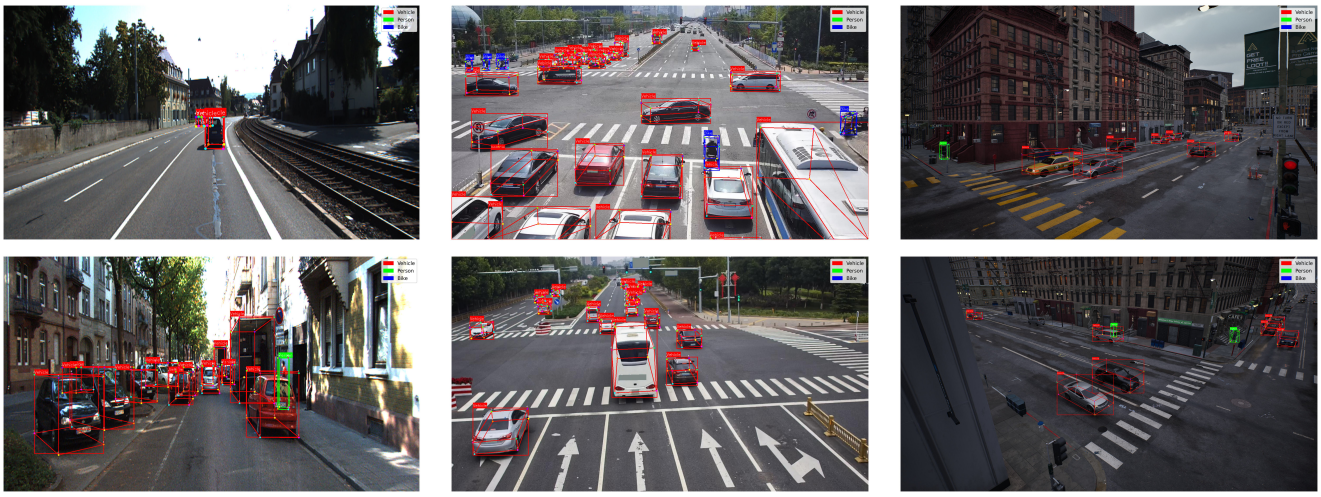


Figure 1. Overview of the UrbanOmniDetect pipeline. **Input:** a single RGB image from any viewpoint (ego-vehicle, infrastructure, or aerial). **Backbone and Neck:** a detection backbone with an augmented feature pyramid spanning levels P1 to P6; the extreme levels P1, P2 and P6 are optional and shown dashed. **Joint Head:** predicts eight ordered 2D keypoints per object, supervised by $\mathcal{L}_{\text{kpt}}$ (Eq. 1). **BEV Head:** maps ground-contact keypoints to a bird's-eye-view plane via a learned homography refined with an orthogonality constraint $\mathcal{L}_{\text{rect}}$ (Eq. 3), without camera intrinsics. An optional frozen auxiliary 2D detector maps box features to BEV centers via $\mathcal{L}_{\text{aux}}$ (Eq. 5).



(a) KITTI (b) DAIR (c) UE5

Figure 2. Samples from the three datasets that make up the UrbanOmniView dataset. (a) KITTI, (b) DAIR, (c) UE5.

sensor diversity encountered in real-world V2X cooperative perception deployments.

UrbanOmniDetect achieves real-time inference (<11 ms on an A100 GPU) while matching or outperforming calibration-dependent baselines on standard KITTI 3D IoU metrics, particularly on the challenging Moderate and Hard splits where depth ambiguity is greatest. Existing monocular 3D detectors fail entirely when deployed on infrastructure cameras with different intrinsics from their training set. In V2X cooperative perception, where infrastructure and vehicle sensors must share detections in real time, per-camera recalibration is a deployment bottleneck that our approach eliminates.

Our contributions are:

- A calibration-free formulation of monocular 3D detection as ordered keypoint regression, enabling a single model to generalize across ego-vehicle, infrastructure, and aerial views. To our knowledge this is the first such unified approach.
- A homography-based BEV projection head that enforces geometric consistency without camera parameters (Equations 2 and 3).
- The UrbanOmniView dataset with high-fidelity UE5 synthetic data spanning diverse urban viewpoints, which we

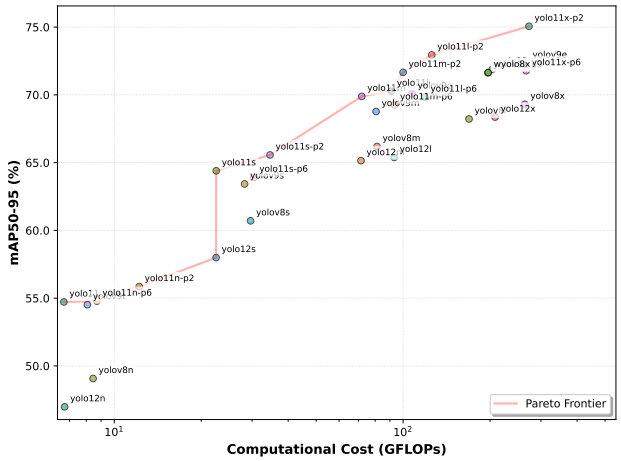


Figure 3. GFLOPs vs. **mAP**$_{50:95}$ performance of different model backbone and head choices on our dataset.

will release publicly.

- Comprehensive evaluation including standard 3D IoU metrics, ablation studies, calibration sensitivity analysis, and cross-domain generalization.

2. Related Work

2.1. Monocular 3D Object Detection

Ego-centric view. Traditional monocular 3D object detection has focused primarily on autonomous driving scenarios with ego-vehicle perspectives, with datasets like KITTI [5] driving most early progress. A large body of methods relies on accurate camera intrinsics and assumes a fixed viewpoint, often regressing depth or 3D box parameters directly from image appearance. Representative approaches include MonoCon [14], which introduces constraints on the monocular 3D box space, DEVIANT [10], which combines geometric priors with volumetric attention, and more recent methods such as MONODGP [22] and MONOLSS [12] that incorporate depth-guided priors and shape regularization. However, recent studies have shown that monocular depth estimation remains unreliable at long range and under diverse outdoor conditions [16], which limits the accuracy and generalizability of such depth-driven approaches, especially when camera parameters change or the deployment viewpoint differs from training.

Infrastructure-based view. More recently, datasets such as DAIR-V2X [31] and V2X-Seq [32] have expanded research into traffic-surveillance viewpoints, offering broader scene coverage from elevated static cameras. Methods designed for these perspectives, including MonoUNI [8] and BEVHeight [30], typically incorporate a pre-estimated ground plane and rely heavily on camera parameters to convert 2D predictions into metric 3D space. These methods show that leveraging a ground plane can stabilize monocular detection, but the reliance on explicit calibration and ground-plane estimation makes them difficult to deploy in real-world infrastructure scenarios where camera height, tilt, and orientation may be unknown or may change over time. Moreover, many of these approaches are tuned for a specific infrastructure viewpoint and are not evaluated under large viewpoint shifts.

Aerial view. The aerial viewpoint introduces even larger viewpoint variability, leading to dramatic changes in object scale and orientation. The CARLA Drone (CDrone) dataset [18] highlights that existing monocular 3D detectors exhibit poor cross-view generalization: models trained on ego-centric images fail under top-down drone viewpoints, and vice versa. Many aerial-view methods also focus only on 2D detection, as in VisDrone-style benchmarks, or make restrictive assumptions about viewpoint range. Recent benchmarks [6, 18] further demonstrate that current monocular 3D detectors rely heavily on calibration and viewpoint-specific cues, limiting applicability to diverse real-world camera placements. In contrast, our setting targets a unified model that can be trained and evaluated across ego-centric, infrastructure-based, and drone viewpoints without explicit camera parameters.

2.2. Pose Estimation

Keypoint-based and pose estimation techniques have been widely explored in human pose prediction and 6D indoor object pose estimation but remain underutilized in outdoor urban 3D detection. Tekin et al. [26] estimate 6D pose from nine keypoints for indoor objects, demonstrating the potential of geometric lifting from 2D structure. PerspectiveNet [7] similarly estimates perspective keypoints and proposes a two-stage detection pipeline, while YOLO-Pose [1] uses unordered keypoints with a transformer architecture for 6D pose estimation. The YOLO pose extensions [17] show that single-stage detection and keypoint estimation can be combined efficiently, but these works typically focus on human or indoor object pose, and not on cross-view urban scenes.

Recent efforts toward perspective-agnostic object detection such as Omni3D [2] and Cube R-CNN [21] show promise for unified cross-dataset 3D detection. However, these systems typically rely on camera intrinsics during inference and often exhibit unstable training dynamics or high computational cost, making them less suitable for real-time, calibration-free deployment. More generally, several works have shown that pose-aware or keypoint-based lifting can be more robust than pure monocular depth regression for outdoor 3D detection, since depth estimates alone are often noisy and uncertain at long range. AutoShape further demonstrates that 2D/3D keypoints and shape priors provide stronger geometric constraints than direct 3D box or depth regression [15]. Depth-centric methods such

as CaDDN and related work explicitly model depth uncertainty because naive depth regression degrades detection performance [23], and recent studies show that depth prediction errors propagate into 3D detection, especially for distant or occluded objects [20]. These observations support our choice to rely on keypoint-based lifting rather than monocular depth estimation.

Nonetheless, keypoint-based monocular 3D detectors have rarely been applied to multi-viewpoint urban scenarios where object scale, orientation, and ground geometry vary strongly across cameras. Existing keypoint or shape-based methods such as RTM3D and AutoShape are evaluated almost exclusively on ego-vehicle datasets like KITTI [11, 15], assuming a single calibrated forward-facing camera. In contrast, recent cross-view benchmarks including Omni3D and CARLA Drone highlight that current monocular 3D detectors tend to perform well either on ego-centric car views or on traffic/drone views, but rarely across all perspectives, and are typically built around depth- or BEV-based architectures that rely on camera intrinsics [2, 18]. To our knowledge, our work is among the first to apply a keypoint-based, calibration-free framework to monocular 3D detection across ego, infrastructure, and aerial viewpoints in a unified training setup.

3. Method

3.1. Overview

Unlike PerspectiveNet, our method uses a single-stage architecture that jointly estimates all bounding boxes and their keypoints in one pass. Unlike YOLOPose, we predict ordered keypoints, which enables consistent orientation estimation for both pedestrians and vehicles and is important for downstream tasks such as trajectory forecasting. For infrastructure-based and aerial views, where metric 3D boxes can be unreliable or inconsistent (especially in synthetic drone data), we introduce a scale-free bird's-eye-view (BEV) estimation head that jointly optimizes our 3D predictions for ground-plane consistency. We further incorporate auxiliary object detectors during training to improve coverage of underrepresented classes in datasets without pose labels. An overview of the full model is shown in Figure 1.

Given a single RGB image as input, our method produces, for each detected object: (i) a 2D bounding box with class label and confidence score, and (ii) an ordered set of eight 2D keypoints corresponding to the projections of the object's 3D bounding box corners onto the image plane. We stress that these *keypoints are not salient feature points*; they are the eight vertices of an axis-aligned 3D cuboid surrounding each object, projected into pixel coordinates. The keypoints follow a fixed ordering convention: indices 0 to 3 correspond to the four top corners and indices 4 to 7 to the four bottom (ground-contact) corners.

The pipeline, illustrated in Figure 1, proceeds as follows. First, the input image is processed by a detection backbone that extracts multi-scale features; we default to YOLO11 [9] (Section 3.2). A feature pyramid neck aggregates features at multiple resolutions; we augment the standard P3 to P5 hierarchy with an additional high-resolution P2 level to improve detection of small and distant objects common in infrastructure and aerial views. A joint detection and keypoint head then predicts, for each object, the 2D bounding box parameters and the eight ordered keypoints simultaneously in a single forward pass (Section 3.3). Finally, a bird's-eye-view (BEV) projection head maps the predicted ground-contact keypoints to a top-down plane using a learned homography, without requiring camera intrinsics or extrinsics (Section 3.4).

Because the network predicts pixel-coordinate keypoints rather than metric 3D coordinates, it requires no camera calibration at inference.

3.2. Backbone and Feature Pyramid

To identify the best speed/accuracy tradeoff, we evaluate four backbone families: YOLOv8 [25], YOLOv9 [29], YOLO11 [9], and YOLO12 [27], each at five scales (nano through extra-large). We also test open-vocabulary and feature-extraction backbones, specifically YOLO-World [3], YOLOE [28], and the DINO family [19, 24, 33], as frozen feature extractors.

Standard detection backbones produce feature maps at scales P3 to P5. We introduce two feature pyramid augmentations that are independent of the backbone choice: (a) **P2**, a higher-resolution feature level that captures fine-grained spatial detail for small objects appearing at distance in infrastructure and aerial views; and (b) **P6**, a lower-resolution level for large, nearby objects. We denote these variants with +P2 and +P6 suffixes (e.g., YOLO11+P2) and P1 to P6 when all levels are included. Higher input resolutions (1280×1280 and 1920×1920) further improve keypoint localization for distant objects and align with HD/4K deployment formats.

3.3. Keypoint-Based 3D Box Prediction

The detection head predicts, for each object, a 2D bounding box and eight ordered 2D keypoints representing the projected corners of the object's 3D bounding box. Training supervision is provided by projecting ground-truth 3D box annotations onto the image plane using the camera parameters available in each training dataset. Crucially, these camera parameters are used *only during data preparation* to generate 2D keypoint labels; they are *not* provided to the network at training or inference time.

Scale ambiguity is inherent to monocular 3D detection without intrinsics. We mitigate it through (1) class-specific dimension priors learned from the training data and (2) the

BEV head's orthogonality constraint (Section 3.4).

We supervise keypoint predictions using a loss derived from the OKS (Object Keypoint Similarity) [13] metric, adapted for 3D corner prediction:

$$\mathcal{L}_{\text{kpt}} = \frac{1}{N}\sum_{i=1}^{N} f_i \sum_{j=1}^{K}\left(1-\exp\left(-\frac{d_{ij}}{2\sigma_j^2\cdot a_i+\epsilon}\right)\right)\cdot m_{ij} \tag{1}$$

where d_{ij} is the squared Euclidean distance between predicted and ground-truth keypoints, f_i normalizes for the number of visible keypoints, a_i is the object's bounding-box area, σ_j is a per-keypoint scale factor, m_{ij} masks occluded keypoints, and $\epsilon = 10^{-9}$ ensures numerical stability.

3.4. Bird's-Eye-View Projection

To produce a top-down spatial layout from 2D keypoint predictions, we estimate a global homography that maps image-plane ground-contact points to a BEV plane, without requiring camera parameters. For each detected instance n, the model predicts eight keypoints; we retain the four ground-contact keypoints $G^n = \{g_i^n\}_{i=1}^4 \subset \mathbb{R}^2$ (indices 4 to 7). A homography $H \in \mathrm{PGL}(3)$ maps image points to BEV coordinates via $\pi(H\tilde{x})$, where $\tilde{x} = [x, y, 1]^\top$ denotes homogeneous coordinates.

We initialize H_0 using a normalized Direct Linear Transform (DLT) fit from one instance's ground keypoints to a canonical rectangle, then refine H by minimizing a geometric orthogonality loss over all instances:

$$H^\star = \arg\min_H \textstyle\sum_n \mathcal{L}_{\text{rect}}\big(\hat{G}^n(H)\big),$$
$$\hat{G}^n(H) = \{\pi(H\tilde{g}_i^n)\}_{i=1}^4, \tag{2}$$

$$\mathcal{L}_{\text{rect}}(Q) = \sum_{i=1}^{4}\left(\frac{q_{i+1}-q_i}{\|q_{i+1}-q_i\|}\cdot\frac{q_{i+2}-q_{i+1}}{\|q_{i+2}-q_{i+1}\|}\right)^2 \tag{3}$$

with cyclic indexing. This loss penalizes non-right angles in the projected ground footprints, enforcing rectangular BEV shapes without access to camera calibration.

3.5. Auxiliary Detection Head

To improve recall for object classes that are underrepresented in datasets with 3D pose annotations, we optionally include a frozen auxiliary 2D detector (pre-trained on COCO). For box-only detections from this auxiliary model, we learn a closed-form linear mapping from box features $\phi(b) = [c_x, c_y, w, h, w/h]^\top$ to the BEV ground-footprint center:

$$\mathcal{L}_{\text{aux}} = \textstyle\sum_n \left\|c^n - W\,\phi(b^n)\right\|_2^2 + \lambda\|W\|_F^2, \tag{4}$$
$$W^\star = \arg\min_{W\in\mathbb{R}^{2\times 5}} \mathcal{L}_{\text{aux}}, \tag{5}$$

yielding predicted centers $\hat{c} = W^\star\phi(b)$ for detections lacking keypoints. The final BEV layout is produced by warping keypoint-based ground footprints and auxiliary predicted centers through $H^\star$.

Table 1. Auxiliary detector evaluation on VisDrone val (548 images, 24,195 instances). Both models use a YOLO11x backbone at 640×640. The COCO-trained model provides higher out-of-distribution recall, motivating its use as the auxiliary head (Section 3.5).

Class	Ours (XL)	Auxiliary (COCO)
All	0.0905	0.1920
Person	0.0312	0.1340
Vehicle	0.0027	0.0264
Bike	0.2370	0.4150

4. Experiments

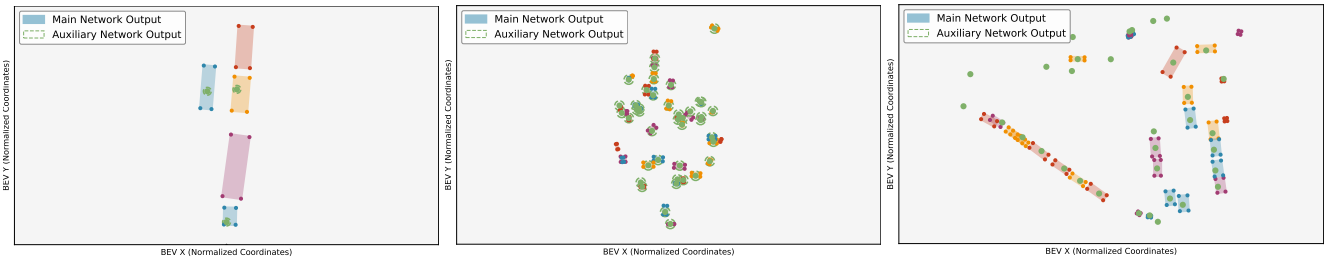
Dataset Our dataset is constructed by combining KITTI, DAIR-V2X [31], and a synthetic dataset generated using the UE5 City Sample demo (denoted UE5-Synthetic). This combination is intended to cover a broad range of camera elevations and orientations relevant to infrastructure-based and aerial sensing. KITTI provides 15,022 frames, the infrastructure split of DAIR-V2X provides 12,424 frames, and UE5-Synthetic adds 10,000 frames. The merged dataset, UrbanOmniView, contains 37,446 samples. Representative examples are shown in Figure 2. As part of this study, we release the UE5-Synthetic portion of UrbanOmniView. We allocate 10% of UrbanOmniView for testing.

Synthetic Data. Our synthetic data component is built upon the Unreal Engine 5 [4] "City Sample" urban scene. We implemented a dynamic environment pipeline with weather and lighting variation (rain, snow, day/night cycles), randomized traffic and pedestrian assets, and custom camera rig scripting to sample diverse viewpoints (ground-level, infrastructure poles, drones). UE5's ray-traced rendering produces high-fidelity RGB frames at 4K resolution. 3D bounding box annotations are generated automatically by exporting each object's transform metadata and collision bounds from the engine and projecting them into image space, yielding pixel-accurate 3D box corners.

4.1. Implementation Details

All models are implemented in PyTorch and trained on a node with 8×A100 GPUs and an AMD EPYC 7J13 CPU. Unless otherwise stated, we train for 100 epochs with the Adam optimizer, an initial learning rate of 1e-2, cosine decay, and a batch size of 64 images per step (8 images per GPU). We apply standard data augmentations including random horizontal flipping, random scaling, and color jitter. For keypoint supervision, we use the loss in Section 3.3 with $K = 8$ keypoints and OKS scale factors

(a) RGB image with predictions, showing 4 Ground Keypoints and auxiliary detection boxes



(b) Estimated Bird's-Eye View

Figure 4. Predictions from our XL model at 1920×1920 with COCO-trained auxiliary detector. Left to right: KITTI, DAIR, UE5. (a) Ground keypoints and auxiliary boxes. (b) Estimated bird's-eye view.

Table 2. **mAP**$_{50:95}$ for different backbone choices. Suffixes +P2 and +P6 indicate our augmented feature pyramid levels (Section 3.2). Bottom rows show frozen open-vocabulary/feature-extraction backbones. Column headers (n, s, m, l, x) denote backbone scale from nano to extra-large.

Backbone	**n**	**s**	**m**	**l**	**x**
YOLO11	0.547	0.644	0.699	0.703	0.719
YOLO11 + P6	0.548	0.639	0.693	0.698	0.718
YOLO11 + P2	**0.559**	**0.656**	**0.717**	**0.729**	**0.751**
YOLO12	0.470	0.580	0.651	0.654	0.684
YOLOv9	0.545	0.634	0.688	0.701	0.716
YOLOv8	0.549	0.607	0.662	0.682	0.693
YOLOEv8-L			0.695		
YOLO-World			0.717		
DINOv3			0.261		

$\{\sigma_j\}_{j=1}^{8} = 0.125$ fixed. The training of the BEV head is performed jointly with the detection backbone, and the auxiliary detector (when used) is frozen and pre-trained on COCO.

4.2. Architecture Design

We evaluate the tradeoff between inference time and accuracy across several backbone families, each tested at five scales (Table 2). We also study the effect of adding P1 to P6 feature levels and test YOLOE, YOLO-World, and DINOv3 as frozen backbones.

We report the relative performance of all models and the resulting Pareto frontier in Figure 3, with numerical results summarized in Table 2. Mean inference times on A100 GPUs are measured using TensorRT over 100 frames from UrbanOmniView and shown in Table 3. Tested feature augmentations yield consistent improvements. Open-vocabulary and feature-extraction backbones show lower performance for this setting.

We also examine the effect of increasing input resolution beyond the default 640×640. Models are trained at 640×640, 1280×1280, and 1920×1920, with results re-

Table 3. Mean inference time (ms) for different backbone variants at 640×640 input resolution on an A100 GPU, compiled with TensorRT.

Backbone	**n**	**s**	**m**	**l**	**x**
YOLO11	1.815	2.281	3.510	4.743	6.891
YOLO11 + P6	2.507	3.236	4.846	6.747	9.632
YOLO11 + P2	2.759	3.534	5.285	7.136	10.069
YOLO12	2.553	3.256	4.549	7.126	9.979
YOLOv9	3.218	3.770	4.447	4.634	10.633
YOLOv8	1.544	2.058	3.494	4.925	7.144

Table 4. mAP$_{50:95}$ vs. input resolution for P2 and P1 to P6 heads (YOLO11x backbone). GFLOPs in gray.

Config.	**640**^{2}	**1280**^{2}	**1920**^{2}
XL, P1 to P6	**0.762** 341	0.797 1363	0.743 4066
XL, P2	0.751 273	**0.800** 1091	**0.808** 2454

Table 5. Comparison of 3D object detection models on the KITTI val dataset.

Model	Precision ↑	Recall ↑	F-1 ↑	AP@OKS = 0.50 ↑	mAP@OKS ↑
DEVIANT	0.9316	0.6658	0.7766	0.6458	0.5520
MonoCon	0.8661	0.6445	0.7391	0.0013	0.0002
MONODGP	0.7170	0.5333	0.6116	0.4911	0.4643
MONOLSS	0.9155	0.7205	0.8064	0.7048	0.6060
Ours (Nano, 640^2)	0.9434	0.9020	0.9222	0.7976	0.5740
Ours (XL, 1920^2)	**0.9853**	**0.9759**	**0.9806**	**0.9429**	**0.8205**

Table 6. Comparison of 3D object detection models on the DAIR-V2x val dataset.

Model	Precision↑	Recall↑	F-1↑	AP@OKS = 0.50↑	mAP@OKS↑
MonoCon	0.2565	0.2579	0.2572	0.0000	0.0000
DEVIANT	0.5925	0.5581	0.5748	0.0003	0.0001
Ours (Nano, 640^2)	0.9404	0.9147	0.9274	0.8265	0.6478
Ours (XL, 1920^2)	**0.9826**	**0.9794**	**0.9810**	**0.9377**	**0.8035**

ported in Table 4. We observe steady gains in mAP with higher resolution. This trend is relevant for deployments using high-resolution infrastructure cameras. Figure 4 shows sample results from our XL model at 1920×1920 with a COCO-trained auxiliary detector.

4.3. Comparison on KITTI and DAIR

To evaluate applicability on standard benchmarks, we compare against open-source implementations of MonoCon [14], DEVIANT [10], MONODGP [22], and MONOLSS [12]. All models are evaluated on KITTI and DAIR using Precision, Recall, F1 score, and average precision averaged across OKS [13] thresholds (AP@OKS). Results are shown in Table 5 and Table 6. Among the baselines, DEVIANT is the only model that produces detections on the infrastructure-based DAIR dataset. Our best model (XL with P2) achieves the strongest results across datasets; note that baselines train on single-source, single-viewpoint data, while ours uses the full multi-view UrbanOmniView corpus by design.

The near-zero OKS scores for MonoCon and DEVIANT on DAIR-V2X (Table 6) result from their reliance on KITTI-specific camera intrinsics: when applied to DAIR's infrastructure cameras with different focal lengths and mounting geometry, depth estimates become geometrically inconsistent and 3D box predictions collapse.

Standard 3D IoU evaluation. To directly compare with calibration-dependent baselines on their native metric, we also report standard KITTI AP_{3D} and AP_{BEV} at IoU≥0.7 for the Car category using the R40 evaluation protocol (Table 7). For our method, we recover metric 3D boxes by combining predicted keypoints with class-specific dimension priors and solving a Perspective-n-Point (PnP) problem using the KITTI camera intrinsics at evaluation time only; this post-hoc step is not part of the model and is used solely to produce boxes in KITTI's metric coordinate system for a fair comparison. Our method achieves the best results on the Moderate and Hard splits for both AP_{3D} and AP_{BEV}, where depth ambiguity is greatest and calibration-dependent methods are most challenged. On the Easy split, which contains predominantly large, nearby objects whose 3D corners project outside the image, PnP recovery is less constrained; this does not affect infrastructure and aerial deployments where objects are typically fully visible.

Table 7. AP_{3D} and AP_{BEV} (R40, IoU≥0.7) on the KITTI val set for the **Car** category.

Method	Easy	Mod.	Hard
AP_{3D}			
DEVIANT	24.63	16.54	14.52
MonoDGP	**30.76**	22.34	19.02
MonoLSS	25.91	18.29	15.94
MonoCon	26.33	19.01	15.98
Ours (XL)	29.61	**30.71**	**27.76**
AP_{BEV}			
DEVIANT	32.60	23.04	19.99
MonoDGP	**39.40**	28.20	24.42
MonoLSS	34.70	25.36	21.84
MonoCon	34.65	25.39	21.93
Ours (XL)	33.86	**35.19**	**31.38**

4.4. Ablation and Analysis

To isolate the contribution of each proposed component, we conduct an ablation using the XL backbone (YOLO11x) at

Table 8. Ablation study on the UrbanOmniView val set. Each row adds one component to the XL baseline (YOLO11x backbone, P3 to P5, real data only, 640×640).

Configuration	$\mathbf{mAP}_{50:95}$
XL baseline (P3 to P5, real only)	0.695
+ Synthetic data (UrbanOmniView)	0.719 (+0.024)
+ P2 feature level	0.751 (+0.032)
+ Resolution 1280×1280	0.800 (+0.049)
+ Resolution 1920×1920	0.808 (+0.008)

640×640 resolution. Table 8 reports the cumulative effect of adding the P2 feature level, synthetic training data, and higher input resolution.

The P2 feature level provides +3.2 mAP points by capturing fine-grained detail for small and distant objects. Synthetic data contributes +2.4 mAP; a model trained on synthetic data alone achieves 0.659, confirming its value even in isolation. Resolution scaling yields the largest individual gain (+4.9 from 640 to 1280), with diminishing returns at 1920 (+0.8). The BEV head is evaluated qualitatively in Figure 4.

Ground-plane adherence. We measure the mean angular deviation between predicted ground-contact edges and the inferred ground plane: 8.79° (KITTI), 11.16° (DAIR-V2X), and 3.57° (UE5), reflecting real-world surface variation across viewpoints.

Auxiliary detector. On VisDrone (Table 1), the COCO-trained auxiliary detector achieves substantially higher recall than our pose-trained model, confirming its value for broadening detection coverage in out-of-distribution settings.

4.5. Calibration Sensitivity

A core motivation for calibration-free detection is robustness to imprecise or unavailable camera intrinsics. To quantify this, we perturb the KITTI P2 projection matrix by scaling the focal length (f_x, f_y) or shifting the principal point (c_x, c_y) and re-evaluate MonoCon and DEVIANT on the val split. Because our method never ingests intrinsics during inference, its predictions are invariant to these perturbations by construction. Table 9 reports AP_{3D} (Moderate, IoU≥0.7).

Even a 2% focal-length error degrades MonoCon by 26% and DEVIANT by 16%; at 5% both lose over 80% of their AP_{3D}; at 10% or with a 50-pixel principal-point shift, performance collapses to near zero. Our method retains its full accuracy under every condition, confirming that the keypoint-based formulation eliminates calibration as a failure mode.

Table 9. Calibration sensitivity on KITTI val (AP_{3D} Mod., IoU≥0.7). Focal length is scaled by the listed factor; principal point is shifted by ±50 px. Our method does not use intrinsics at inference, so its score is constant.

Method	GT	$0.98f$	$0.95f$	$0.9f$	pp ±50
MonoCon	18.51	13.75	3.57	0.00	0.00
DEVIANT	16.71	13.97	2.92	0.01	0.00
Ours	**30.71**	**30.71**	**30.71**	**30.71**	**30.71**

5. Conclusion

We presented UrbanOmniDetect, a calibration-free monocular 3D object detection framework that formulates 3D detection as ordered keypoint regression within a single-stage, backbone-agnostic architecture. By combining KITTI, DAIR-V2X, and high-fidelity UE5 synthetic data into the UrbanOmniView dataset, a single model generalizes across ego-vehicle, infrastructure, and aerial viewpoints without camera intrinsics. This property is critical for scalable V2X cooperative perception across heterogeneous sensor networks. On KITTI, our method outperforms calibration-dependent baselines on the Moderate and Hard splits using standard 3D IoU metrics, while maintaining real-time inference. We further show that calibration-dependent baselines degrade catastrophically under even small intrinsic perturbations, whereas our method is invariant by construction—making it directly applicable to autonomous driving, drone surveillance, and infrastructure-based V2X cooperative perception.

Limitations. The BEV head assumes a dominant ground plane, which may fail on steep ramps or multi-level structures. Performance degrades when ground-contact keypoints are heavily occluded, and the formulation does not enforce temporal consistency in video streams. For large, nearby objects whose 3D corners project outside the image, metric 3D box recovery via PnP is under-constrained; handling out-of-frame keypoints is a direction for future work.

Future work. We plan to extend the framework to LiDAR/camera fusion, incorporate temporal cues, generalize BEV projection to handle multiple ground surfaces, and apply domain adaptation to further improve cross-domain generalization. All code, trained model checkpoints, and datasets will be released publicly.

Acknowledgements

This work was supported by the NSF Engineering Research Center for Smart Streetscapes under Award EEC-2133516, NSF Grants CNS-2450567 and CNS-2038984, and by computing resources provided by the NVIDIA Academic Grant Program and the Empire AI Consortium.

References

[1] Arash Amini, Arul Selvam Periyasamy, and Sven Behnke. Yolopose: Transformer-based multi-object 6d pose estimation using keypoint regression. In *International Conference on Intelligent Autonomous Systems*, pages 392–406. Springer, 2022. 3

[2] Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. Omni3d: A large benchmark and model for 3d object detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13154–13164, 2023. 3, 4

[3] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16901–16911, 2024. 4

[4] Epic Games. Unreal Engine 5. https://www.unrealengine.com, 2022. Accessed: 2025. 1, 5

[5] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 32(11):1231–1237, 2013. 1, 3

[6] Derek Gloudemans, Yanbing Wang, Gracie Gumm, William Barbour, and Daniel B Work. The interstate-24 3d dataset: a new benchmark for 3d multi-camera vehicle tracking. *arXiv preprint arXiv:2308.14833*, 2023. 3

[7] Siyuan Huang, Yixin Chen, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. Perspectivenet: 3d object detection from a single rgb image via perspective points. *Advances in neural information processing systems*, 32, 2019. 1, 3

[8] Jia Jinrang, Zhenjia Li, and Yifeng Shi. Monouni: A unified vehicle and infrastructure-side monocular 3d object detection network with sufficient depth clues. *Advances in Neural Information Processing Systems*, 36:11703–11715, 2023. 3

[9] Rahima Khanam and Muhammad Hussain. Yolov11: An overview of the key architectural enhancements. *arXiv preprint arXiv:2410.17725*, 2024. 1, 4

[10] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. Deviant: Depth equivariant network for monocular 3d object detection. In *European Conference on Computer Vision*, pages 664–683. Springer, 2022. 3, 7

[11] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. In *European Conference on Computer Vision*, pages 644–660. Springer, 2020. 1, 4

[12] Zhenjia Li, Jinrang Jia, and Yifeng Shi. Monolss: Learnable sample selection for monocular 3d detection. In *2024 International Conference on 3D Vision (3DV)*, pages 1125–1135. IEEE, 2024. 3, 7

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 5, 7

[14] Xianpeng Liu, Nan Xue, and Tianfu Wu. Learning auxiliary monocular contexts helps monocular 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1810–1818, 2022. 3, 7

[15] Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15641–15650, 2021. 3, 4

[16] Shujie Luo, Hang Dai, Ling Shao, and Yong Ding. M3dssd: Monocular 3d single stage object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6145–6154, 2021. 3

[17] Debapriya Maji, Soyeb Nagori, Manu Mathew, and Deepak Poddar. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2637–2646, 2022. 3

[18] Johannes Meier, Luca Scalerandi, Oussema Dhaouadi, Jacques Kaiser, Nikita Araslanov, and Daniel Cremers. Carla drone: Monocular 3d object detection from a different perspective. *arXiv preprint arXiv:2408.11958*, 2024. 3, 4

[19] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 4

[20] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3142–3152, 2021. 4

[21] Jens Piekenbrinck, Alexander Hermans, Narunas Vaskevicius, Timm Linder, and Bastian Leibe. Rgb-d cube r-cnn: 3d object detection with selective modality dropout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1997–2006, 2024. 3

[22] Fanqi Pu, Yifan Wang, Jiru Deng, and Wenming Yang. Monodgp: Monocular 3d object detection with decoupled-query and geometry-error priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6520–6530, 2025. 3, 7

[23] Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8555–8564, 2021. 4

[24] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. 4

[25] Mupparaju Sohan, Thotakura Sai Ram, and Ch Venkata Rami Reddy. A review on yolov8 and its advancements. In *International Conference on Data Intelligence and Cognitive Informatics*, pages 529–545. Springer, 2024. 4

[26] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *Proceed-*

ings of the IEEE conference on computer vision and pattern recognition, pages 292–301, 2018. 1, 3

[27] Yunjie Tian, Qixiang Ye, and David Doermann. Yolov12: Attention-centric real-time object detectors. *arXiv preprint arXiv:2502.12524*, 2025. 4

[28] Ao Wang, Lihao Liu, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yoloe: Real-time seeing anything. *arXiv preprint arXiv:2503.07465*, 2025. 4

[29] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. Yolov9: Learning what you want to learn using programmable gradient information. In *European conference on computer vision*, pages 1–21. Springer, 2024. 4

[30] Lei Yang, Kaicheng Yu, Tao Tang, Jun Li, Kun Yuan, Li Wang, Xinyu Zhang, and Peng Chen. Bevheight: A robust framework for vision-based roadside 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21611–21620, 2023. 3

[31] Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo, Hanyu Li, Xing Hu, Jirui Yuan, et al. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21361–21370, 2022. 1, 3, 5

[32] Haibao Yu, Wenxian Yang, Hongzhi Ruan, Zhenwei Yang, Yingjuan Tang, Xu Gao, Xin Hao, Yifeng Shi, Yifeng Pan, Ning Sun, et al. V2x-seq: A large-scale sequential dataset for vehicle-infrastructure cooperative perception and forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5486–5495, 2023. 3

[33] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 4