Appendix

A. Implementation Details

In our implementation, we build MonoCoP upon the MonoDETR framework [80]. All experiments are conducted on a single NVIDIA A6000 GPU. We train the model for 250 epochs using a batch size of 16 and a learning rate of $2 \times 10^{,-4}$. The AdamW optimizer is adopted with a weight decay of $10^{,-4}$. Additional hyperparameters and implementation details are provided in Tab. A1.

Item	Value
optimizer	AdamW
learning rate	2e-4
weight decay	1e-4
scheduler	Step
decay rate	0.5
decay list	[85, 125, 165, 205]
number of feature scales	4
hidden dim	256
feedforward dim	256
dropout	0.1
nheads	8
number of queries	50
number of encoder layers	3
number of decoder layers	3
encoder npoints	4
decoder npoints	4
number of queries	50
number of group	11
class loss weight	2
α in class loss	0.25
bbox loss weight	5
GIoU loss weight	2
3D centor loss weight	10
dim loss weight	1
depth loss weight	1
depth map loss weight	1
class cost weight	2
bbox cost weight	5
GIoU cost weight	2
3D centor cost weight	10

Table A1. **Main hyperparameters of MonoCoP.**

B. Visualization

We evaluate our MonoCoP on three well-known datasets: KITTI, Waymo, and nuScenes. MonoCoP achieves SoTA performance across these datasets. We finally visualize the detection results on these three datasets. Fig. A2 presents the 3D and BEV detection results. By predicting 3D attributes conditionally to mitigate the instability and inaccuracy arising from their inter-correlation, MonoCoP improves detection accuracy, particularly for farther away objects, consistent with the results in Fig. 4c. Similarly, Fig. A3 demonstrates that, despite the larger variation in 3D size in Waymo compared to KITTI, MonoCoP reliably predicts more accurate 3D size and depth for large objects. Moreover, as shown in Fig. A4, our method also delivers more precise angle and depth estimations.

C. Ablations

C.1. Things We Tried That Did Not Make it into the Main Algorithm

- **Using DINOv2 [48] as a Backbone.** We attempted to replace the conventional ResNet backbone in Mono3D with DINOv2, a powerful vision foundation model known for its depth perception capabilities. We experimented with both freezing and fine-tuning DINOv2 but found no performance improvement. We attribute this to (1) the relatively small scale of the Mono3D dataset, which may not fully leverage DINOv2's capacity, and (2) the substantial domain gap between DINOv2's pre-training data and Mono3D.
- **Splitting Images into Sub-Images.** We also explored splitting the original image into four sub-images (shown in A1) and extracting features from each separately, motivated by the high resolution of the input images (e.g., 1280×340 in KITTI). Unfortunately, this approach led to inferior performance compared to using the entire image at once.
- **Relation Encoding.** We additionally experimented with modeling pairwise relations between queries by incorporating their relative spatial positions. The goal is to enhance the detector's geometric reasoning by providing explicit relational cues. However, we did not observe performance gains from this design.

C.2. Different Backbones

In Tab. A2, we evaluate MonoCoP with different image backbones on the KITTI Val split and observe that it consistently surpasses MonoDGP [52] under all configurations. Among the evaluated backbones, ResNet-50 yields the strongest overall detection performance.

Figure A1. **Image Splitting.** The high-resolution original image is divided horizontally into four sub-images.

Methods	Image Backbone	AP_{3D}, 0.7 Easy	Mod.	Hard
MonoDGP [52]	ResNet-18	25.32	19.62	16.89
MonoCoP	ResNet-18	**27.78**	**21.03**	**17.98**
MonoDGP [52]	ResNet-34	27.96	20.13	17.19
MonoCoP	ResNet-34	**28.32**	**22.32**	**19.23**
MonoDGP [52]	ResNet-50	29.41	21.12	18.11
MonoCoP	ResNet-50	**32.06**	**23.98**	**20.64**
MonoDGP [52]	ResNet-101	27.02	19.92	17.07
MonoCoP	ResNet-101	**30.14**	**21.75**	**18.56**

Table A2. **Performance on Image backbone.** MonoCoP consistently outperforms MonoDGP [52] across all backbones.

AttributeNet	AP_{3D}, 0.7 Easy	Mod.	Hard
LR	29.72	22.68	19.59
LR + ReLU	30.62	22.94	19.91
2LR + ReLU	**32.06**	**23.98**	**20.64**
3LR + ReLU	30.82	23.19	19.82

Table A3. **Performance comparison of different AttributeNet (AN) designs.** We examine a single linear layer, our default two-layer MLP, and deeper variants. The two-layer configuration consistently delivers the best results, demonstrating its effectiveness in balancing representational capacity and computational cost.

C.3. Design of AttributeNet

MonoCoP leverages an AttributeNet (AN) to capture attribute-specific features. Inspired by the MLP-based projector in vision-language models [35], we initially design AN as two linear layers with ReLU activation. This simple, two-layer structure strikes a balance between representational capacity and computational cost, allowing the model to effectively learn attribute representations without excessive overfitting. We then explore alternative AN configurations, such as a single linear layer or deeper MLP variants with additional linear layers and ReLU activations. As shown in Tab. A3, however, our original two-layer configuration consistently yields the strongest overall performance, underscoring its efficacy in learning robust and discriminative attribute-specific features.

Number of chain	AP_{3D}, 0.7 Easy	Mod.	Hard
One chain	**32.06**	**23.98**	**20.64**
Two chains	30.26	23.35	20.10
Three chains	30.67	23.11	19.90

Table A4. **Performance of MonoCoP when varying the number of appended chains**. While adding extra chains can lead to marginal gains, the results demonstrate diminishing returns beyond the first chain, indicating that a single chain already captures most of the essential inter-attribute correlations.

C.4. Number of Chain

MonoCoP leverages a Chain-of-Prediction (CoP), which sequentially and conditionally predicts attributes by **learning**, **propagating**, and **aggregating** attribute-specific features along the chain. This design helps mitigate inaccuracies and instabilities arising from inter-correlations among 3D attributes. In this subsection, we investigate how varying the number of chains in MonoCoP affects performance. First, we incorporate one additional chain and average the outputs across both chains. Next, we add two additional chains and average the outputs of all three. Our experimental findings (see A4) indicate that, although appending extra chains slightly increases computational complexity, it does not consistently yield notable performance gains. One plausible explanation is that the network may have already learned sufficient inter-attribute correlations from a single chain, causing further additions to become redundant. Another possible reason is that the increased complexity could introduce noise into the learning process, offsetting any potential benefits from extra chains. As a result, increasing the number of chains beyond one does not appear to offer further improvements in predictive accuracy.

D. KITTI Results

Tab. A5 presents the image-only 3D detection results on the KITTI Test for the Cyclist and Pedestrian categories. MonoCoP achieves SoTA performance across all metrics for the challenging Cyclist category and attains second-best results in the Moderate and Hard settings for the Pedestrian.

Method	Ped AP_{3D} % (↑)			Cyc AP_{3D} % (↑)		
	Easy	Mod.	Hard	Easy	Mod.	Hard
MonoFlex [81]	11.89	8.16	6.81	3.39	2.10	1.67
GUP Net [43]	14.72	9.53	7.87	4.18	2.65	2.09
DEVIANT [28]	15.04	9.89	8.38	5.28	2.82	2.65
MonoCon [75]	13.10	8.41	6.94	2.80	1.92	1.55
MonoUNI [23]	**15.78**	**10.34**	**8.74**	<u>7.34</u>	<u>4.28</u>	<u>3.78</u>
MonoDGP [52]	15.04	9.89	8.38	5.28	2.82	2.65
MonoCoP (Ours)	<u>15.61</u>	<u>10.33</u>	<u>8.53</u>	**8.89**	**5.08**	**5.25**

Table A5. **KITTI Test Results for Pedestrians and Cyclists** at $IoU_{3D} \geq 0.5$. MonoCoP achieves SoTA performance across most metrics among image-only methods. [Key: **First**, <u>Second</u>]

E. Limitations.

While MonoCoP models the interdependencies among 3D attributes and improves both accuracy and stability, it does not explicitly consider the influence of camera parameters. For instance, variations in camera focal length can introduce a zoom effect that alters the apparent scale of objects and may confuse the detector. Developing methods that remain robust under varying camera intrinsics is an important direction for future work. Moreover, recent advances in multimodal learning and recognition systems have demonstrated strong capabilities in visual reasoning and multimodal fusion [9, 12, 13, 19, 64, 87–89]. However, how to effectively leverage these models for Mono3D remains largely unexplored.

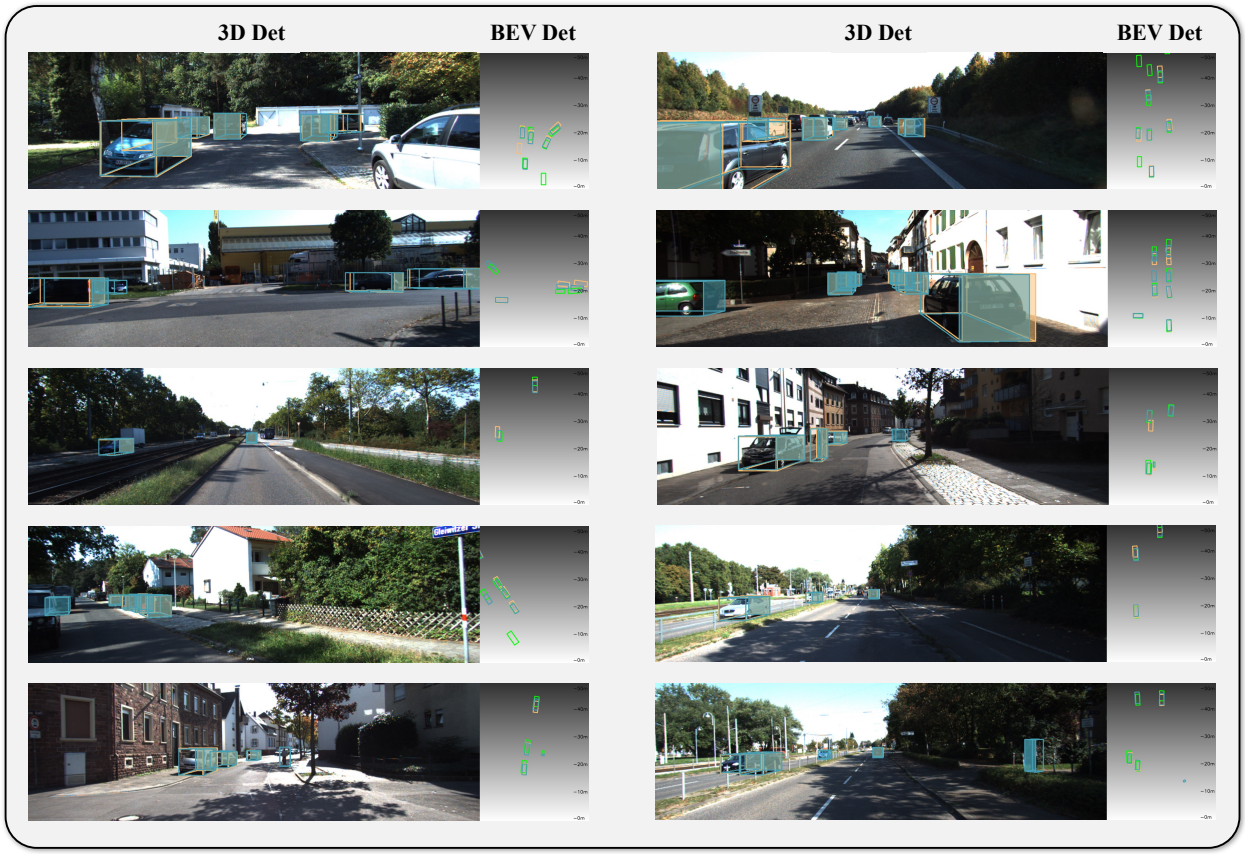

Figure A2. **KITTI Qualitative Results.** MonoCoP demonstrates superior performance in both 3D and BEV detection over the baseline [80]. By predicting 3D attributes conditionally to mitigate the instability and inaccuracy arising from their inter-correlation, MonoCoP improves detection accuracy, particularly for farther away objects, consistent with the results in Fig. 4c. [Key: MonoCoP, Baseline, Ground Truth]

Figure A3. **Waymo Qualitative Results.** MonoCoP demonstrates superior performance in both 3D and BEV detection over the baseline [80]. By predicting 3D attributes conditionally to mitigate the instability and inaccuracy arising from their inter-correlations, MonoCoP predicts more accurate 3D size and depth for large object, demonstrating the effectiveness of MonoCoP. [Key: MonoCoP, Baseline, Ground Truth]

Figure A4. **nuScenes frontal Visualization.** MonoCoP demonstrates superior performance in both 3D and BEV detection over the baseline [80]. By predicting 3D attributes conditionally to mitigate instability and inaccuracy arising from their inter-correlations, MonoCoP predicts more accurate 3D angle and depth, demonstrating effectiveness of MonoCoP. [Key: MonoCoP, Baseline, Ground Truth]