VLC FUSION: VISION-LANGUAGE CONDITIONED SENSOR FUSION FOR ROBUST OBJECT DETECTION

Aditya Taparia[1*] **Noel Ngu**[1] **Mario Leiva**[2] **Joshua Shay Kricheli**[1]
John Corcoran[3] **Nathaniel D. Bastian**[4] **Gerardo Simari**[2]
Paulo Shakarian[5] **Ransalu Senanayake**[1]

[1]Arizona State University, Tempe, AZ, USA
[2]Dept. of Computer Science and Engineering, Universidad Nacional del Sur,
Institute for Computer Science and Engineering, Bahía Blanca, Argentina
[3]U.S. Department of Defense, Arlington, VA, USA
[4]United States Military Academy, West Point, NY, USA
[5]Syracuse University, Syracuse, NY, USA

ABSTRACT

Although fusing multiple sensor modalities can enhance object detection performance, existing fusion approaches often overlook subtle variations in environmental conditions and sensor inputs. As a result, they struggle to adaptively weight each modality under such variations. To address this challenge, we introduce Vision-Language Conditioned Fusion (VLC Fusion), a novel fusion framework that leverages a Vision-Language Model (VLM) to condition the fusion process on nuanced environmental cues. By capturing high-level environmental context such as darkness, rain, and camera blurring, the VLM guides the model to dynamically adjust modality weights based on the current scene. We evaluate VLC Fusion on real-world autonomous driving and military target detection datasets that include image, LiDAR, and mid-wave infrared modalities. Our experiments show that VLC Fusion consistently outperforms conventional fusion baselines, achieving improved detection accuracy in both seen and unseen scenarios. **Code:** `https://github.com/aditya-taparia/VLCFusion`

1 INTRODUCTION

Reliable object detection is critical for many real-world autonomous systems such as autonomous vehicles and surveillance platforms. Since different sensor modalities offer distinct advantages, multi-modal fusion techniques aim to integrate object detectors trained on these different modalities. For example, since RGB images provide high-resolution detail while LiDAR offers depth perception despite its sparse point cloud, sensor fusion can provide a high-resolution image with some depth information.

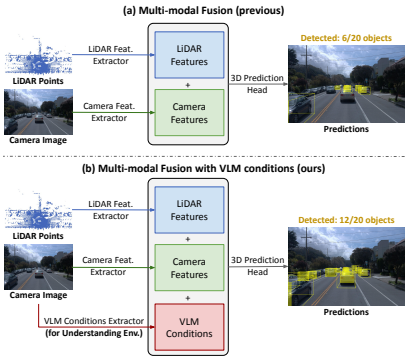


Figure 1: Overview of **VLC Fusion**. Compared to (a) standard fusion, (b) our method modulates modality-specific features with environment-specific meta-information, called *conditions*, improving the resilience of object detection.

A key limitation of current fusion methods is that they overlook how the performance of each modality varies with external environmental conditions. Since object detection models are optimized individually for specific sensor modalities, each excels under certain environmental conditions but can exhibit vulnerabilities under others. For instance, even state-of-the-art RGB-based object detectors such as Detection Transformer (DETR) Carion et al. (2020)

*Corresponding author: `ataparia@asu.edu`.

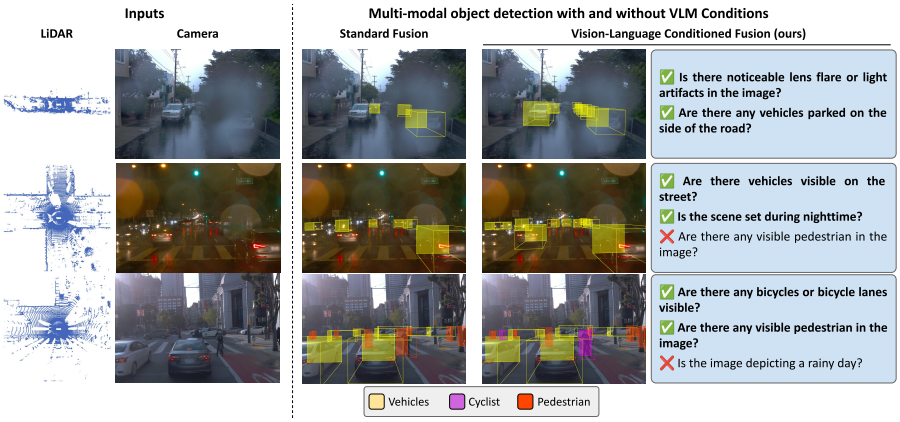


Figure 2: Comparision of predictions from multi-modal and VLC Fusion: from left to right, raw LiDAR point clouds and camera views, predictions from a multi-modal fusion baseline, and predictions from our VLC Fusion conditioned on VLM-extracted environmental cues (✓/✗prompts shown at right). Conditioning on high-level context improves detection performance in recovering occluded cars (Example 1), detecting more vehicles under nighttime glare (Example 2), and correctly identifying the cyclist (magenta) and pedestrians (orange) (Example 3) where the baseline misclassifies.

perform well in clear, well-lit conditions but degrade considerably in low-light or adverse weather scenarios such as fog Bijelic et al. (2020); Pathiraja et al. (2024). Conversely, LiDAR-based object detectors such as PointPillars Lang et al. (2019) and SECOND Yan et al. (2018) provide robust performance under varied lighting levels but can deteriorate in weather conditions such as rain due to light scattering and other sensor-specific limitations Delecki et al. (2022). The problem of environment dependence becomes even more pronounced when the system is deployed in unseen environments.

To address this challenge, we propose Vision-Language Conditioned Fusion (VLC Fusion), a novel approach to incorporate environmental meta-information obtained through Vision-Language models (VLMs) into the fusion process. Since current state-of-the-art VLMs have demonstrated impressive scene-understanding capabilities Li et al. (2023a); Liu et al. (2023), we use them offline to reliably extract detailed environmental cues. We then proposed an architecture to incorporate the VLM conditions into the fusion network. At test-time, in addition to raw sensor inputs, the VLM provides an analysis of the scene, making our method robust to both seen (in-distribution) and unseen (out-of-distribution) scenarios. Fig. 1 illustrates the difference between (a) standard multi-modal fusion and (b) our VLC Fusion. Furthermore, Fig. 2 presents qualitative examples where this conditioning on VLM cues leads to better detection. The primary contributions of the paper are:

1. We propose a novel fusion approach, called VLC Fusion, that automatically weighs fusion of features based on environment-specific meta-information.
2. We introduce an automated framework for offline extraction and integration of relevant environmental cues from raw datasets.
3. We empirically demonstrate the usefulness of environmental conditions in multi-modal sensor fusion on two real-world object detection tasks, autonomous driving and military target detection. We also demonstrate how lightweight, fast, small-scale VLMs can be used realistically during the online object detection phase.

2 Related Work

Multi-Modal Sensor Fusion for Object Detection. LiDAR-camera fusion methods demonstrated clear benefits over single-modality approaches by combining precise geometric measurements with dense visual context. Previous works such as MV3D Chen et al. (2017) and AVOD Ku et al. (2018) project LiDAR point clouds into the image plane to jointly learn features. Fusion SSD Bijelic et al. (2020) concatenates feature maps from both modalities and applies convolutional layers for joint detection, and Learnable Align Li et al. (2022) uses a cross-attention block to align and integrate

modality-specific features. More recent works such as TransFusion Bai et al. (2022) use cross-modal attention to further improve alignment at multiple scales. Although these methods achieve good performance, they generally *apply static fusion rules* that do not adjust to changing environments Bijelic et al. (2020). This lack of adaptability leads to degraded performance when encountering conditions not well-represented in the training data, such as unusual weather patterns Delecki et al. (2022); Pathiraja et al. (2024).

Condition-Aware Fusion Approaches. To handle diverse lighting and weather scenarios, subsequent work recognized the need for adaptability, introducing condition-aware mechanisms that adapt fusion weights based on environment estimates. Switchable Branch Networks Zhao et al. (2019) learn separate experts for day and night. More recently, RGB-X Deevi et al. (2024) proposed the use of scene agnostic switch to switch between detection heads based on particular scenario. Despite these advances, most methods *rely on a fixed taxonomy of conditions and require annotated examples* for each. This reliance on predefined categories restricts their ability to handle ambiguous or varying conditions and requires potentially expensive data annotation efforts for every new condition, limiting their flexibility Brödermann et al. (2024). In contrast, we create application-specific conditions and also provide a way to identify these relevant conditions.

Vision–Language Models for Context-Awareness. Building on condition-aware mechanisms, recent work explores the use of large pre-trained vision–language models (VLMs) for extracting semantic context. Models like CLIP Radford et al. (2021) support matching image regions to text. More integrated approaches, such as PaLM-E Driess et al. (2023) and RoboFlamingo Li et al. (2023b), combine vision, language, and robot state for downstream reasoning. These models enable systems to understand not just what objects are present, but also how they relate to tasks or environmental cues. However, leveraging VLMs to infer environmental cues rather than relying on fixed, discrete categories and using these insights to guide real-time sensor fusion remains an open challenge and through our work we address this gap.

3 METHODOLOGY

Our methodology comprises two major components: 1) identification of application-specific environmental conditions and querying VLM to obtain the corresponding responses, and 2) integrating these conditional information into the sensor fusion architecture.

3.1 OFFLINE CONDITION EXTRACTION AND GENERATION

Extracting meaningful environmental conditions is crucial for guiding sensor fusion in our method. To this end, we explored two ways by which one can extract (or define) conditions from a dataset.

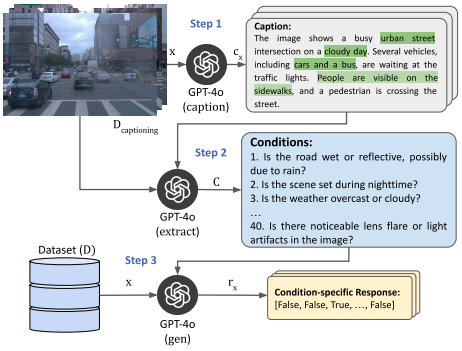


Figure 3: Overview of the 3-step automated pipeline for extracting env. conditions.

Human-Defined Conditions: Leveraging prior domain knowledge and metadata from dataset, experts can manually define relevant conditions based of the task. While straightforward, this approach can be subjective and may not generalize effectively across diverse datasets.

Automated Condition Extraction: To overcome limitations associated with manual definitions, we introduce an automated framework for offline extraction of rich environmental cues from dataset via VLM. For this purpose, as shown in Fig. 3, we introduce a three step process:

Step 1 (Captioning): Let the training dataset be D with N images, and a randomly selected subset of that be $D_{\text{captioning}}$ with M images. With $\text{caption}()$, we first generate descriptive captions, c_x:

$$c_x \leftarrow \text{caption}(x; p_{\text{captioning}}), \quad \forall x \in D_{\text{captioning}}$$

for text prompt $p_{\text{captioning}}$. Specifically, a pre-trained VLM is queried with $p_{\text{captioning}}$ and a system prompt described in Appendix D. It gives us M image-caption pairs (x, c_x).

Step 2 (Extraction): After captioning, we use $\mathrm{extract}()$ to generate a set of environmental conditions, C, using the M image-caption pairs:

$$C \leftarrow \mathrm{extract}(\{(x_m, c_{x_m})\}_{m=1}^{M}, p_{\text{extraction}})$$

for prompt $p_{\text{extraction}}$ and a system prompt described in Appendix D. This step helps us to derive structured, application-specific environmental conditions. This automated process robustly captures both high-level semantics and fine-grained contextual cues.

Step 3 (Generation): Once environmental conditions have been identified, the next step involves generating condition-specific responses for each data point in training dataset D, by querying a pre-trained VLM. Specifically, we utilize GPT-4o to query responses, $r \in \{\mathrm{True}, \mathrm{False}\}$, for evaluating the presence of extracted conditions for each image in the full training dataset:

$$r_{x,c} = \mathrm{gen}(x, c) \quad \forall x \in D, c \in C,$$

where a condition c, acts as a prompt for generation. We obtain the presence of conditions for a given image as $r_x = [\mathrm{True}, \mathrm{False}, \ldots, \mathrm{True}]$. We use these N responses to train the fusion model.

3.2 Sensor Fusion with Environmental Conditions

Integrating environmental conditions, C, into multi-modal sensor fusion is important to improve the robustness of detection models under real-world distribution shifts. We build on the concept of Feature-wise Linear Modulation (FiLM) Perez et al. (2018) to condition fusion on environmental context. The architecture of the proposed method, VLC Fusion, is illustrated in Fig. 1 and consists of feature-level fusion via the Convolutional Block Attention Module (CBAM) Woo et al. (2018), followed by conditional feature reweighting using FiLM.

We first fuse the concatenated features from multiple modalities such as LiDAR and RGB camera using CBAM. CBAM emphasizes significant spatial and channel-wise information from concatenated multi-modal inputs using two attention operations, channel attention and spatial attention. Formally, let $F_{\text{modality1}} \in \mathbb{R}^{B\times C\times H\times W}$ and $F_{\text{modality2}} \in \mathbb{R}^{B\times C'\times H\times W}$ be the two modality feature maps. We first concatenate

$$F \;=\; \big[F_{\text{modality1}};\, F_{\text{modality2}}\big] \;\in\; \mathbb{R}^{B\times C''\times H\times W},$$

where, $C'' = C + C'$. These concatenated features are passed through CBAM:

$$F' = M_c(F) \;\otimes\; F,$$
$$F'' = M_s(F') \;\otimes\; F',$$

where "$\otimes$" is element–wise multiplication, and

$$\begin{aligned} M_c(F) &= \sigma\big(W_1\,\mathrm{AvgPool}(F) + W_2\,\mathrm{MaxPool}(F)\big), \\ &\in \mathbb{R}^{B\times C''\times 1\times 1}, \\ M_s(F') &= \sigma\big(f^{7\times 7}\big([\mathrm{AvgPool}(F');\,\mathrm{MaxPool}(F')]\big)\big), \\ &\in \mathbb{R}^{B\times 1\times H\times W} \end{aligned}$$

where, σ denotes the sigmoid activation, W_i denotes linear layer weights, $f^{7\times 7}$ denotes convolution operation with 7×7 kernel, and $\mathrm{AvgPool}$ and $\mathrm{MaxPool}$ are average and max pooling operations, respectively.

After fusing the two modalities using CBAM, VLC Fusion leverages FiLM to explicitly modulate the fusion process based on environmental cues. FiLM dynamically adjusts the multi-modal feature representations through condition-dependent affine transformations. This enables the model to adapt to diverse scenarios by tailoring the importance of each modality based on the environmental context. Formally, we condition F'' on the VLM-predicted conditions r_x via a FiLM layer:

$$\hat{F} \;=\; \big(1 + \gamma(r_x)\big) \;\odot\; F'' \;+\; \beta(r_x),$$

where $\gamma(r_x)$ and $\beta(r_x)$ are the scale and shift tensors learned from the condition vector r_x.

Figure 4: Qualitative examples of VLC Fusion in both seen and unseen environments. **Top row:** 3D detections on the Waymo Open Dataset under *seen* and *unseen* conditions, with vehicles (yellow), cyclists (purple) and pedestrians (red) accurately localized. **Bottom row:** 2D detection on the ATR dataset for *seen* and *unseen* scenarios. More qualitative examples are provided in App. G.

Table 1: Performance on the Waymo dataset (Seen: Day and Night). VLC Fusion with extracted conditions outperforms all baselines across most categories under both L1 and L2 difficulties. The best and second best performance is highlighted with bold and underline, respectively.

Fusion Techniques	**L1 Difficulty (3D mAP/mAPH)**				**L2 Difficulty (3D mAP/mAPH)**			
	Vehicle	Pedestrian	Cyclist	Overall	Vehicle	Pedestrian	Cyclist	Overall
Fusion SSD	19.7/19.3	37.2/31.7	21.5/19.9	26.1/23.6	16.9/16.6	32.4/27.6	19.7/18.2	23.06/20.8
Fusion SSD with Self-Attention	18.2/17.9	34.5/29.2	12.7/11.7	21.8/19.6	15.5/15.3	29.9/25.3	11.6/10.7	19.07/17.1
Learnable Align	13.1/12.7	33.07/27.5	9.60/8.68	18.6/16.3	11.2/10.9	28.5/23.7	8.81/7.96	16.2/14.2
RGB-X	21.8/21.5	39.04/33.03	20.6/19.2	27.1/24.5	18.7/18.4	34.1/28.8	18.9/17.6	23.9/21.6
VLC Fusion with Human Defined Conditions (n=3)	**25.28/24.9**	39.6/34.1	20.7/19.5	28.5/26.2	**21.7/21.4**	34.7/29.8	19.1/17.9	25.2/23.08
VLC Fusion with Extracted Conditions (n=10)	25.24/24.8	**41.2/35.02**	**25.3/23.5**	**30.6/27.8**	21.7/21.3	**36.2/30.6**	**23.2/21.6**	**27.06/24.5**

4 EXPERIMENTS

In this section, we empirically evaluate our proposed VLC Fusion methodology using two real-world datasets: the Waymo Open dataset Sun et al. (2020), where we fuse RGB and LiDAR modalities, and the Automated Target Recognition (ATR) dataset DSIAC, where we fuse visible and infrared (IR) imagery. We investigate VLC Fusion's effectiveness in enhancing detection performance under both seen (training distribution) and unseen (out-of-distribution) scenarios. As shown in App. C.1.3 and C.2.3, 85% and 76% of data in Waymo and ATR dataset contain at least one active condition.

The experimental setup in described in App. A where we provide details on dataset splits, evaluation metrics, implementation details, and baseline methods. We also provide ablation studies on the importance of selecting accurate and semantically consistent environmental conditions, and use of light-weight VLM at test-time for faster deployment in App. F. Our primary results are as follows,

Waymo Open Dataset: Tables 1 and 2 clearly shows that VLC Fusion with 10 conditions consistently outperforms other baseline algorithms. Specifically, VLC Fusion achieved a 3D mAP of 30.6 in the Day and Night (seen) scenario and 35.2 in the Dawn/Dusk (unseen) scenario. Interestingly, increasing the number of conditions from 3 to 10 notably improved accuracy for underrepresented classes such as cyclists by approximately 5% in both the seen and unseen scenarios. This improvement suggests that incorporating additional environmental context helps the fusion model more effectively generalize and handle challenging, underrepresented scenarios. Qualitative examples from Fig. 4 further support these findings, illustrating VLC Fusion's enhanced capability to detect vehicles, cyclists, and pedestrians under varied environmental conditions.

ATR Dataset: Tables 3 and 4 reinforce the effectiveness of VLC Fusion. Despite the ATR dataset presents comparatively simpler environmental variations, VLC Fusion consistently improved performance across both seen and unseen scenarios. VLC Fusion with 14 human-defined environmental conditions performed best on the seen test set, achieving a mAP of 61.04, surpassing Fusion SSD's best baseline result of 60.22. Additionally, VLC Fusion utilizing 6 extracted conditions delivered the highest performance on the unseen test set, achieving a mAP of 10.02. These results emphasize

Table 2: Performance on the Waymo dataset (Unseen: Dawn/Dusk). VLC Fusion with extracted conditions outperforms all baselines across most categories under both L1 and L2 difficulties. The best and second best performance is highlighted with bold and underline, respectively.

Fusion Techniques	**L1 Difficulty (3D mAP/mAPH)**				**L2 Difficulty (3D mAP/mAPH)**			
	Vehicle	Pedestrian	Cyclist	Overall	Vehicle	Pedestrian	Cyclist	Overall
Fusion SSD	21.5/21.1	40.4/34.6	31.1/28.5	31.03/28.08	18.8/18.5	37.4/32.02	29.4/27.01	28.6/25.8
Fusion SSD with Self-Attention	19.3/19.09	37.0/31.07	16.2/15.2	24.2/21.8	16.9/16.7	34.2/28.7	15.3/14.4	22.2/19.9
Learnable Align	14.3/13.9	36.9/30.5	17.9/16.01	23.08/20.1	12.5/12.1	34.07/28.1	17.01/15.1	21.2/18.5
RGB-X	22.8/22.4	42.4/36.02	27.6/26.04	31.03/28.1	20.04/19.7	39.4/33.3	26.2/24.6	28.5/25.9
VLC Fusion with Human Defined Conditions (n=3)	**27.1/26.7**	42.5/36.6	27.4/25.9	32.3/29.7	**23.9/23.6**	39.4/33.9	26.05/24.5	29.8/27.3
VLC Fusion with Extracted Conditions (n=10)	26.7/26.3	**45.1/38.3**	**33.6/31.2**	**35.2/32.0**	23.6/23.2	**41.8/35.5**	**31.9/29.6**	**32.4/29.4**

Table 3: Class-wise and overall mAP scores on the ATR dataset (Seen distances). VLC Fusion performs best compared to other methods when using human-defined conditions. The best and second best performance is highlighted with bold and underline, respectively.

Fusion Technique	Pickup	SUV	BTR70	BRDM2	BMP2	T72	ZSU23	2S3	MTLB	D20	**Overall**
Fusion SSD	49.95	58.34	67.34	**63.10**	**73.85**	71.39	69.04	**71.23**	**58.55**	19.36	60.22
Fusion SSD with Self-Attention	44.43	52.3	61.71	58.11	67.26	64.31	64.59	65.85	48.7	**59.69**	53.7
Learnable Align	50.38	56.32	**67.59**	61.67	73.56	**73.35**	**70.96**	70.93	56.31	15.51	59.66
RGB-X	42.97	50.06	63.82	58.64	69.94	66.82	66.31	66.80	53.69	5.41	54.45
VLC Fusion with Human Defined Conditions (n=14)	**51.83**	**59.28**	66.86	61.64	71.75	69.14	69.73	70.85	58.05	31.23	**61.04**
VLC Fusion with Extracted Conditions (n=6)	46.74	57.94	66.64	62.72	72.31	69.28	67.69	67.86	56.20	20.06	58.75

Table 4: Class-wise and overall mAP scores on the ATR dataset (Unseen distances). VLC Fusion performs best compared to other methods when using extracted conditions. The best and second best performance is highlighted with bold and underline, respectively.

Fusion Technique	Pickup	SUV	BTR70	BRDM2	BMP2	T72	ZSU23	2S3	MTLB	D20	**Overall**
Fusion SSD	0.8	7.41	14.89	7.32	17.17	13.44	**7.12**	21.68	**4.51**	2.52	9.69
Fusion SSD with Self-Attention	0.76	5.38	14.51	7.96	16.78	10.46	4.29	20.7	3.69	0.79	8.53
Learnable Align	0.01	0.77	10.38	5.32	**19.21**	8.54	0.98	13.95	4.41	0.92	6.45
RGB-X	0.38	7.51	13.12	6.25	15.49	15.81	3.05	**23.05**	2.93	0.86	8.84
VLC Fusion with Human Defined Conditions (n=14)	0.64	8.16	13.65	8.32	19.07	15.36	6.66	20.97	3.88	**3.36**	10.01
VLC Fusion with Extracted Conditions (n=6)	**1.71**	**8.59**	**14.97**	**8.76**	18.54	**16.04**	5.69	19.46	4.42	2.02	**10.02**

that even datasets with less pronounced environmental variation benefit from incorporating context-specific environmental conditions into the fusion model. Additional results are provided in App. E.

5 Conclusion

In this paper, we introduced Vision-Language Conditioned Fusion (VLC Fusion), a novel sensor fusion framework designed to improve object detection robustness by dynamically conditioning on environmental context queried from VLMs. Our approach addresses the inherent limitations of conventional fusion methods, which often struggle to adaptively weight sensor modalities under diverse and previously unseen environmental conditions. By explicitly leveraging high-level information about the environment, VLC Fusion improves detection accuracy and adaptability.

We demonstrated the effectiveness of VLC Fusion on two distinct real-world datasets: the Waymo dataset for autonomous driving and the ATR dataset for military target recognition. Empirical results confirmed that our method consistently outperforms existing fusion baselines on both seen and unseen scenarios. Our findings underscore the potential of incorporating advanced semantic reasoning from VLMs into sensor fusion architectures, paving the way for more reliable autonomous systems in complex, dynamic environments. Future work includes exploring more sophisticated techniques for condition extraction, further generalizing VLC Fusion across additional modalities and environments, and investigating real-time deployment scenarios.

ETHICAL CONSIDERATIONS

Our work focuses on improving object detection under varying environmental conditions. We do not foresee any direct ethical concerns of our approach. All experiments were conducted using publicly available datasets and standard training frameworks, such as Hugging Face and MMDetection3D.

ACKNOWLEDGMENTS

This research was supported by the Defense Advanced Research Projects Agency (DARPA) under Cooperative Agreement No. HR00112420370 (MCAI). The views expressed in this paper are those of the authors and do not necessarily reflect the official policy or position of the U.S. Military Academy, the U.S. Army, the U.S. Department of Defense, or the U.S. Government. We would also like to thank Caleb Liu for early discussions on fine-tuning object detectors, and Som Sagar for insights into the applications of FiLM.

REFERENCES

Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1080–1089, 2022. doi: 10.1109/CVPR52688.2022.00116.

Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11682–11692, 2020.

Tim Brödermann, Christos Sakaridis, Yuqian Fu, and Luc Van Gool. Cafuser: Condition-aware multimodal fusion for robust semantic perception of driving scenes. *arXiv preprint arXiv:2410.10791*, 2024.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1907–1915, 2017.

Sri Aditya Deevi, Connor Lee, Lu Gan, Sushruth Nagesh, Gaurav Pandey, and Soon-Jo Chung. Rgb-x object detection via scene-specific fusion modules. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 7366–7375, 2024.

Harrison Delecki, Masha Itkina, Bernard Lange, Ransalu Senanayake, and Mykel J Kochenderfer. How do we fail? stress testing perception in autonomous vehicles. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5139–5146. IEEE, 2022.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

DSIAC. ATR Algorithm Development Image Database. `https://dsiac.org/databases/atr-algorithm-development-image-database/`, 2010. Accessed: 2024-08-13.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.

Jason Ku, Muhammad Mozifian, Jungwook Lee, Amirmohammad Harakeh, and Steven L. Waslander. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1–8, 2018.

Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705, 2019.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 2023b.

Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17182–17191, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, et al. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

Bimsara Pathiraja, Caleb Liu, and Ransalu Senanayake. Fairness in autonomous driving: Towards understanding confounding factors in object detection under challenging weather. *arXiv preprint arXiv:2406.00219*, 2024.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and et al. Learning transferable visual models from natural language supervision, 2021.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2446–2454, 2020.

vik. moondream2 (revision 92d3d73), 2024. URL https://huggingface.co/vikhyatk/moondream2.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10), 2018. ISSN 1424-8220. doi: 10.3390/s18103337. URL https://www.mdpi.com/1424-8220/18/10/3337.

Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. M2det: A single-shot object detector based on multi-level feature pyramid network. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, volume 33, pp. 9259–9266, 2019. doi: 10.1609/aaai.v33i01.33019259.

APPENDIX

A EXPERIMENTAL SETUP

A.1 DATASETS

Waymo Open Dataset: We use the San Francisco portion of the Waymo Open dataset, which provides synchronized LiDAR and RGB imagery captured at 10 Hz in a busy urban environment with diverse weather (e.g. rainy, sunny) and lighting conditions (e.g. day time, night time, dawn/dusk time). Each data point contains approximately 200 frames across 20 seconds.

Table 5: Dataset splits for Waymo and ATR datasets across seen and unseen scenarios. The unseen scenarios are used exclusively for evaluating fusion robustness.

Dataset	Variation	Train	Validation	Test
Waymo Open dataset (RGB + LiDAR)	Seen	73,112	9,139	9,139
	Unseen	–	–	7,052
ATR dataset (Visible + MWIR)	Seen	45,207	15,075	15,088
	Unseen	–	–	11,952

For our experiments, we define two scenarios: a seen scenario, which includes data collected under day and night time, and an unseen scenario, comprising data from dawn and dusk. The seen scenario was used for both training and testing, while the unseen scenario was reserved strictly for testing. Prior to training, frames were shuffled to ensure diversity and robustness. The resulting splits for both scenarios are summarized in Table 5.

ATR Dataset: The ATR dataset contains visible and microwave infrared (MWIR) imagery aimed at target recognition applications, and comprehensive metadata detailing object distances, viewing angles, wind speed, and other relevant attributes.

We first synchronized the frames from the two modalities using timestamps and object metadata to ensure proper alignment. Following synchronization, we partitioned the dataset into seen and unseen scenarios based on object distance. The seen set includes distances of 1000m, 2000m, 3000m, 4000m, and 5000m, while the unseen set comprises intermediate distances (1500m, 2500m, 3500m, and 4500m). The resulting splits for both seen and unseen sets are summarized in Table 5.

A.2 METRICS

We evaluated the trained fusion models using dataset-specific metrics, as detailed below:

Waymo Open Dataset: For the Waymo Open dataset, we evaluated the performance of the fused networks using mean Average Precision (mAP) and mean Average Precision with Heading (mAPH). Evaluations were conducted for three object classes—Vehicle, Pedestrian, and Cyclist—at IoU thresholds of 0.7, 0.5, and 0.5, respectively. Performance was reported across two difficulty levels (L1 and L2), which are defined in the dataset itself based on the number of LiDAR points associated with each object.

ATR Dataset: For the ATR dataset, we assessed the fusion network's performance using mean Average Precision ($mAP_{0.5:0.05:0.95}$) and mean Average Recall at 100 proposal (mAR_{100}). Evaluations were conducted on both the seen and unseen test splits, and have reported both overall and per-class scores.

A.3 IMPLEMENTATION DETAILS

Below, we describe training setups for individual object detection models per dataset and modality, followed by the generation of environmental conditions by querying VLM.

A.3.1 OBJECT DETECTORS

Waymo Open Dataset: For the RGB modality, we trained a DETR-based 2D object detector with a ResNet-50 backbone using the Huggingface Trainer. The model was trained for 150 epochs with a batch size of 16 and an initial learning rate of 5×10^{-5}. Best checkpoints were saved every 250 steps using validation mAP.

For the LiDAR modality, we used the SECOND 3D object detection model, trained on Waymo point-cloud data. Training followed the standard MMDetection3D pipeline, with the point-cloud range set from $[-76.8, -51.2, -2]$m to $[76.8, 51.2, 4]$m, targeting the Car, Pedestrian, and Cyclist classes. The model was trained for 100 epochs using the AdamW optimizer with a learning rate of 1×10^{-3} and a batch size of 2. Evaluation was conducted after each epoch using the `WaymoMetric` evaluator, and the best-performing checkpoint was selected for downstream use. Additionally, each fusion model in Waymo dataset was fine-tuned with similar setting as LiDAR but for 40 epochs.

ATR Dataset: We trained two separate DETR-based 2D object detectors with a ResNet-50 backbone, one for visible images and the other for MWIR images. Each model was trained for 140 epochs using the AdamW optimizer with an initial learning rate of 5×10^{-5} and a weight decay of 1×10^{-4}. Training used a batch size of 32 with gradient accumulation over 8 steps. Model checkpoints were evaluated based on validation mean Average Precision (mAP), and the checkpoint achieving the highest mAP was retained for final evaluation. Additionally, each fusion model in ATR dataset was fine-tuned with same configuration but for 100 epochs.

A.3.2 VLM-queried Environmental Conditions

We first generated environmental conditions using the methods described in Section 3.1. Two sets of conditions were defined: human-defined and automatically extracted. After obtaining these conditions, we queried GPT-4o to generate responses for each data point in the dataset.

A.4 Baselines

We explored various fusion strategies, including Fusion SSD Bijelic et al. (2020), Fusion SSD with self-attention, RGB-X Deevi et al. (2024), and Learnable Align Li et al. (2022):

Fusion SSD and Variations: The base Fusion SSD architecture concatenates feature maps from both modalities and applies convolution to reduce them to the appropriate dimensions before passing them to the detection head. In the self-attention variant, an additional attention module is applied after the convolution step to re-weight features based on their importance.

RGB-X: In this approach, feature maps from both modalities are concatenated and passed through a Convolutional Block Attention Module (CBAM). CBAM first applies channel attention by feeding global average-pooled and max-pooled descriptors through a shared two-layer MLP. This is followed by spatial attention, computed using a 7×7 convolution over the concatenated channel-wise max and average maps. This two-step attention mechanism adaptively emphasizes both the most informative channels and spatial regions. After attention is applied, the features are passed through a series of convolutional layers to align the dimensions with the detection head.

Learnable Align: We also evaluated Learnable Align, where a lightweight cross-attention block is used to fuse features from the two modalities. In this method, each spatial cell in one modality's feature map is treated as a query, while the corresponding features from the other modality serve as keys and values. This end-to-end attention mechanism enables the model to align and highlight the most relevant information across modalities.

Each fusion method was trained using the standard detection head for its dataset: a DETR-based 2D head for ATR and a SECOND-based 3D head for Waymo. All fusion methods were evaluated without environmental conditions.

B Computational Resources

All experiments were conducted on a single NVIDIA H100 GPU (96 GB HBM3e) running Ubuntu 22.04, with CUDA 11.8. For training the detection models, we utilized mixed-precision (FP16) via PyTorch's AMP module to reduce GPU memory usage and accelerate kernel execution. Memory consumption varied depending on the fusion technique and task, reaching a maximum of approximately 40 GB. Training each model took roughly 4 days. Inference was also performed on the same NVIDIA H100 GPU, with a maximum memory usage of around 10 GB.

C AUTOMATIC CONDITION EXTRACTION

In this section, we provide the details on conditions extracted from both the dataset. We also evaluate the correctness of captions being generated for both dataset in step 1.

C.1 WAYMO OPEN DATASET

C.1.1 SAMPLE IMAGE-CAPTION PAIRS

The Fig. 5 shown below highlights the sample image-caption pairs created during the automated conditional extraction of Waymo dataset.

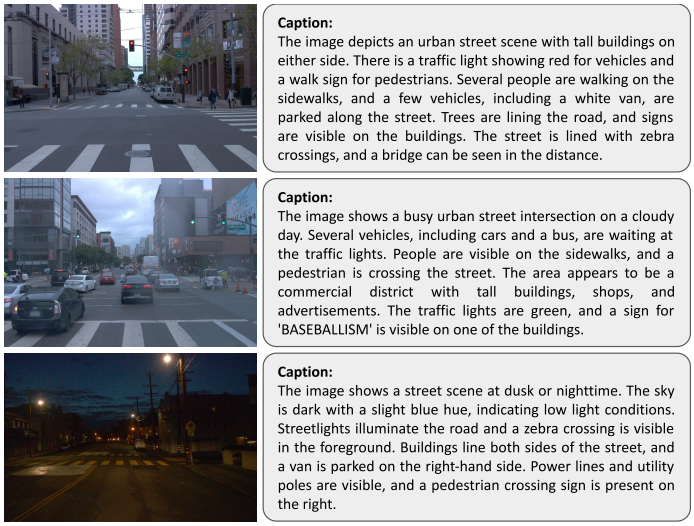


Figure 5: Samples of image-caption pairs generated during automatic condition extraction for Waymo dataset.

C.1.2 LIST OF EXTRACTED ENVIRONMENTAL CONDITIONS

Below, we provide the complete list of extracted environmental conditions extracted from Waymo dataset.

1. Is the road wet or reflective, possibly due to rain?
2. Are there any visible pedestrians in the image?
3. Is there a visible stop sign in the image?
4. Are there any vehicles parked on the side of the road?
5. Is a traffic light visible in the image?
6. Is the image depicting a rainy day?
7. Are there any tall buildings visible?
8. Is there a dedicated lane for buses or taxis?
9. Is the scene set during nighttime?
10. Is there construction work visible?
11. Is there a vehicle in motion in the image?
12. Are street signs or traffic signs visible?
13. Is there greenery or trees lining the street?
14. Is there any advertisement or commercial sign visible?
15. Are there any bicycles or bicycle lanes visible?

16. Is there a body of water visible?
17. Are overhead power lines visible?
18. Is public transportation, like a bus, visible?
19. Is a visible crosswalk present?
20. Are there any orange traffic cones visible?
21. Is the sky clear and blue?
22. Are the roads cracked or uneven?
23. Is there a sense of fog or mist in the image?
24. Is there a notable commercial establishment visible?
25. Is a noticeable hill or incline visible?
26. Is the scene from a residential neighborhood?
27. Is there an indication of a scenic viewpoint?
28. Is the scene taking place at an intersection?
29. Are buildings visible in the scene?
30. Is traffic congestion visible?
31. Is a pedestrian bridge or crossing present?
32. Is there traffic light congestion or light signals visible?
33. Is the street scene located in an urban environment?
34. Are there multiple lanes on the road?
35. Is the weather overcast or cloudy?
36. Are there parked cars visible on the street?
37. Is there a visible neon or illuminated sign?
38. Is the image captured from an elevated perspective?
39. Is the overall atmosphere calm and quiet?
40. Is there noticeable lens flare or light artifacts in the image?

C.1.3 Additional quantitative analysis

Fig. 6 and 7 shows the activation of conditions over Day-night (seen) and Dawn/dusk (unseen) test set. We can see that 85% of the test dataset have at least one active environmental condition.

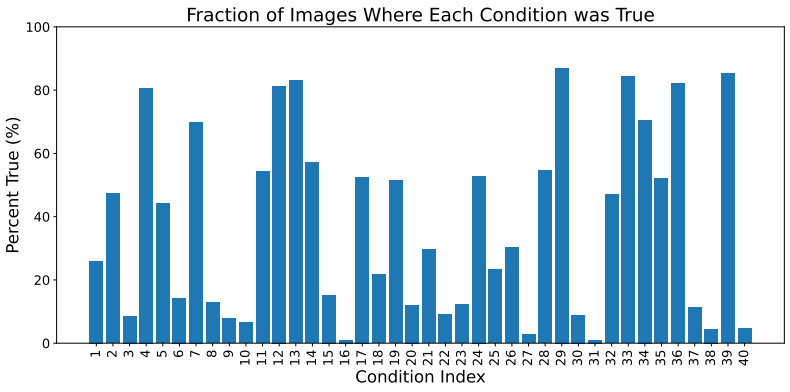


Figure 6: Fraction of images in the Day–Night (seen) test set for which each condition is true in Waymo dataset.

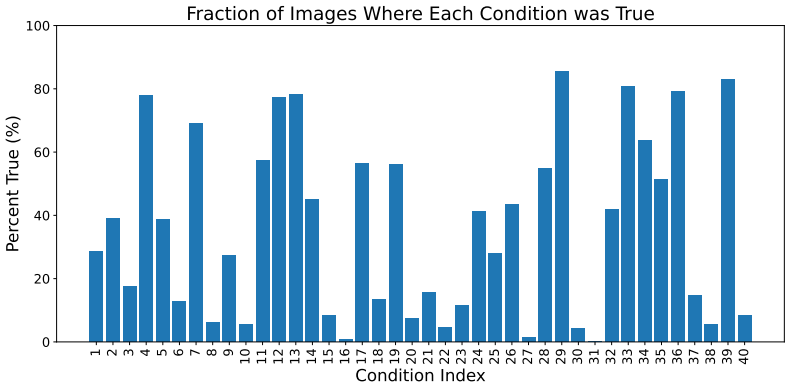


Figure 7: Fraction of images in the Dawn–Dusk (unseen) test set for which each condition is true in Waymo dataset.

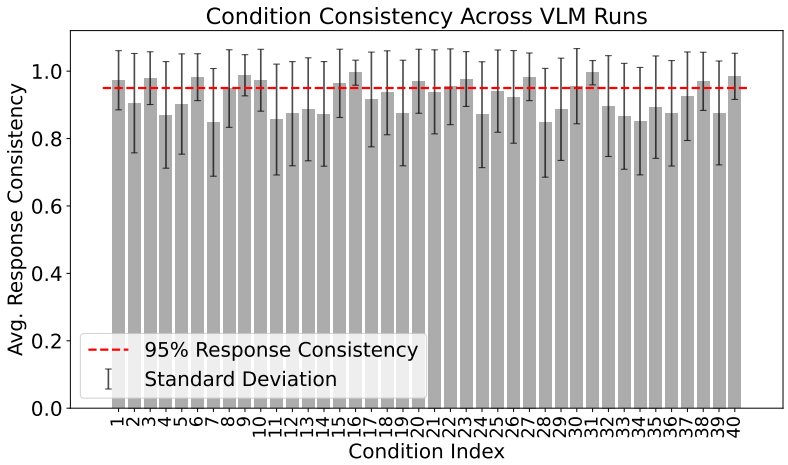


Figure 8: Average condition response consistency over 5 runs for the Waymo dataset.

C.2 ATR Dataset

C.2.1 Sample image-caption pairs

The Fig. 9 shown below highlights the sample image-caption pairs created during the automated conditional extraction of ATR dataset.

C.2.2 List of extracted environmental conditions

Below, we provide the complete list of extracted environmental conditions extracted from ATR dataset.

1. Is there a vehicle present in the image?
2. Is the terrain mostly flat?
3. Are there hills or mountains in the background?
4. Is the sky overcast or cloudy?
5. Is the image in black and white?
6. Is there sparse vegetation present in the image?
7. Does the landscape appear arid or desert-like?
8. Is there a road or path visible in the image?

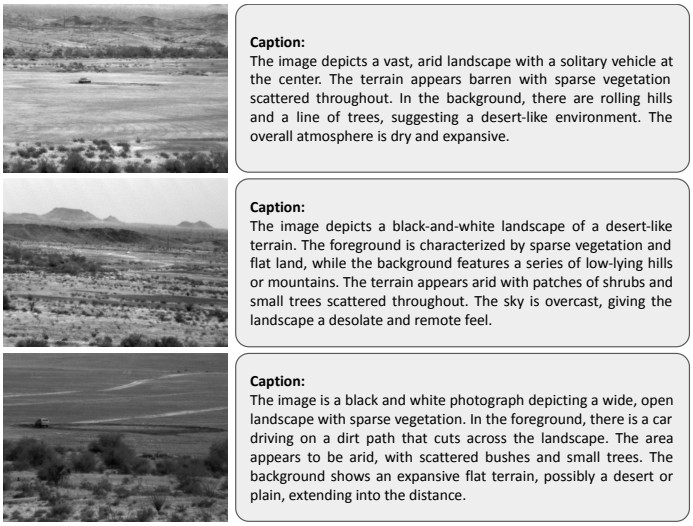


Figure 9: Samples of image-caption pairs generated during automatic condition extraction for ATR dataset.

9. Does the image convey a sense of desolation or remoteness?
10. Is the landscape devoid of human structures?
11. Is there any evidence of movement, such as tire tracks or dust?
12. Does the scene have a sense of barrenness or isolation?
13. Is there a military vehicle like a tank present?
14. Is there any dust or haze present in the scene?
15. Is the image devoid of visible human presence?
16. Is there a single structure visible?
17. Are there rolling hills or mountains in the background?
18. Is the landscape described as barren?
19. Is the lighting subdued or muted?

C.2.3 Additional quantitative analysis

Fig. 10 and 11 shows the activation of conditions over seen distances and unseen distances test set. We can see that 76% of the test dataset have at least one active environmental condition.

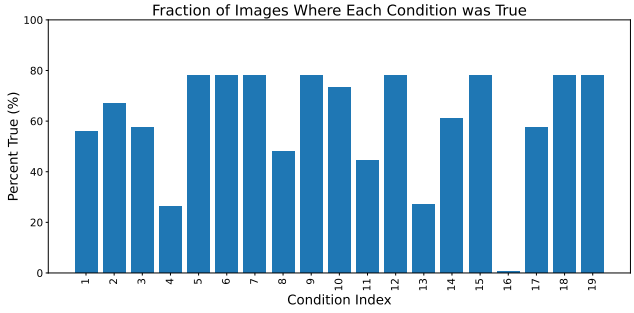


Figure 10: Fraction of images in the seen distances test set for which each condition is true in ATR dataset.

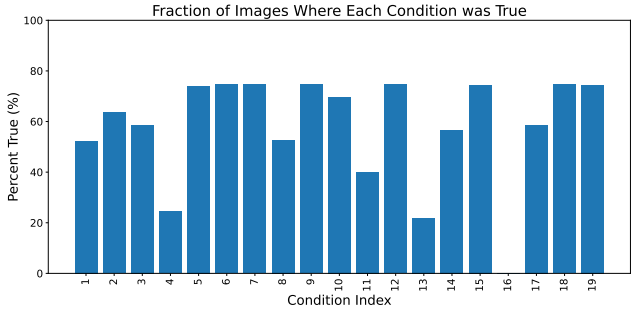


Figure 11: Fraction of images in the unseen distances test set for which each condition is true in ATR dataset.

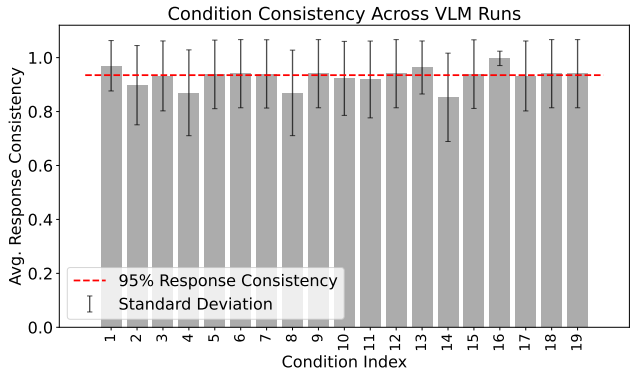


Figure 12: Average condition response consistency over 5 runs for the ATR dataset.

C.3 CLIP-Based Caption Quality Analysis

To quantitatively assess the alignment between generated captions and visual content, we compute the cosine similarity between image and caption embeddings using CLIP (ViT-B/32) Radford et al. (2021). As detailed in Hessel et al. (2021), CLIP similarity scores between 0.3 and 0.4 indicate good alignment in open-world captioning tasks. And as shown in table 6, our results indicate that the generated captions are semantically coherent with the input images, supporting their reliability in condition generation for VLC Fusion.

Table 6: Average CLIP-based cosine similarity between images and their generated captions.

Dataset	**Clip Score**
Waymo Dataset	0.314 ± 0.020
ATR Dataset	0.335 ± 0.023

D Prompt Templates

In this section, we discuss the prompts used for defining our automatic environmental condition extraction framework. For each step, we use a separate set of system and user prompt defined as:

Captioning: In this step, we prompt the VLM to describe the images to create image-caption pairs. The prompt template followed is described in Fig. 13.

Extraction: In this step, we prompt the VLM to provide the set of conditions based on image-caption pairs. The prompt template used is described in Fig. 14.

System and User prompt template

System Prompt:
"You are an assistant that generates consistent, structured descriptions for the provided image(s). Output should be in the following JSON format:"
{ ``Conditions'': ``<description>''}

User Prompt: "Provide a description based on the following image."
[Image]

Figure 13: System and user prompt templates for the VLM "captioning" stage.

Table 7: Class-wise and overall mAR_{100} scores on the ATR dataset (Seen distances).

Fusion Technique	Pickup	SUV	BTR70	BRDM2	BMP2	T72	ZSU23	2S3	MTLB	D20	**Overall**
Fusion SSD	79.87	80.58	84.51	85.53	85.0	87.96	86.18	88.87	66.6	77.3	82.24
Fusion SSD with Self-Attention	78.15	79.87	85.46	84.3	85.68	87.2	86.84	87.38	59.69	74.95	80.95
Learnable Align	81.64	80.01	85.14	85.17	86.73	87.99	84.75	89.27	65.46	78.34	82.45
RGB-X	77.95	80.25	85.46	84.55	86.76	86.44	87.17	86.73	63.09	73.71	81.21
VLC Fusion with Human Defined Conditions (n=14)	82.06	81.53	85.63	83.7	86.98	86.85	85.12	87.75	66.87	76.16	82.27
VLC Fusion with Extracted Conditions (n=6)	76.89	81.77	87.17	85.34	88.0	88.05	85.94	88.53	65.84	76.51	82.41

Generation: In this step, we query the VLM to generate the responses based on the presence and absence of the extracted conditions. The prompt template followed is described in Fig. 15.

E Additional Results from ATR Experiment

In this section, we provide additional results of VLC Fusion and other fusion techniques on ATR dataset. Specifically, we provide the overall and per-class mAR_{100} scores in table 7 and 8. As shown, VLC Fusion with extracted conditions performed best and second best in seen and unseen test scenarios, respectively.

F Ablation Study

To better understand the influence of VLM-based environmental conditions on the performance of our VLC Fusion, we performed ablation studies investigating two critical factors: the scale (capacity) of the VLM used for querying conditions, and the quantity/consistency of the queried conditions.

F.1 Effect of Using Small-scale VLMs for Querying Conditions

In this section, we investigate how the scale and capacity of the Vision-Language Model (VLM) used for querying environmental conditions affect detection performance. Intuitively, we expect larger-scale VLMs to produce more accurate and semantically richer environmental condition predictions, thus enhancing the performance of the fused network. Conversely, smaller-scale VLMs are more practical but provide limited semantic reasoning capabilities and less accurate condition predictions, thus potentially reducing fusion performance.

We compared two smaller-scale VLMs (Moondream2 vik (2024) and SmolVLM Marafioti et al. (2025)) against larger-scale VLM (GPT-4o). As shown in Fig. 16, use of small VLMs slightly reduced the performance compared to the GPT-4o. Specifically, the performance for the "Day-Night (seen)" scenario dropped from the 30.6 to 30.14 with Moondream2 and further to 27.31 with SmolVLM. Similarly, for the "Dawn-Dusk (unseen)" scenario, performance decreased from the 35.2 to 33.91 (Moondream2) and 30.22 (SmolVLM). These results confirm our hypothesis that the scale of VLM influences the accuracy of environmental condition predictions and the overall performance

Table 8: Class-wise and overall mAR_{100} scores on the ATR dataset (Unseen distances).

Fusion Technique	Pickup	SUV	BTR70	BRDM2	BMP2	T72	ZSU23	2S3	MTLB	D20	**Overall**
Fusion SSD	11.19	21.89	28.54	45.73	32.39	39.23	36.77	52.36	8.12	19.14	29.54
Fusion SSD with Self-Attention	14.42	19.46	25.38	34.55	35.43	36.68	22.86	52.58	5.57	18.53	26.54
Learnable Align	2.72	20.64	25.01	31.37	39.41	30.76	18.39	39.26	7.3	12.52	22.74
RGB-X	10.5	18.18	26.14	31.07	34.86	39.68	34.37	44.83	6.88	14.68	26.12
VLC Fusion with Human Defined Conditions (n=14)	4.09	20.88	28.76	40.64	42.39	43.31	27.22	51.16	9.42	22.08	28.99
VLC Fusion with Extracted Conditions (n=6)	17.02	19.51	28.44	42.12	39.84	37.36	38.99	45.24	8.56	16.2	29.33

System and User prompt template

System Prompt:
"You are an assistant that generates consistent, structured conditions for the given image. These conditions are based on various aspects of the image and its description. The conditions should be in the form of questions. Generate as many unique conditions as possible. The questions should be in the form of yes/no questions. Do not include any specific information about the image or description while generating the conditions. Output should be in the following JSON format:"

```
{ "Conditions": [
"<condition_1>",
"<condition_2>"
]}
```

User Prompt: "Provide conditions based on the following images and their captions."
[Images, Captions]

Figure 14: System and user prompt templates for the "extraction" stage.

of VLC Fusion. On a bright side, both the small-scale VLMs achieved performance comparable to large-scale VLM (GPT-4o) making the method useful in practical applications.

F.2 Effect of Condition Quantity and Consistency

We further analyze how varying the number and consistency of queried environmental conditions impacts the fusion model's performance. Fig. 17a and 17b clearly shows the trend observed in our experiments. Initially, increasing the number of conditions leads to improvement in detection performance. However, beyond a certain point, as the number of conditions increases further, we observe a performance decline. This trend can be attributed to incorporating less consistent and potentially noisy conditions. Indeed, we observed that conditions ranked higher in consistency contributed positively to performance, whereas adding less consistent conditions diminished model accuracy.

Thus, our results underscore the critical importance of selecting a carefully curated set of highly consistent conditions, balancing richness of contextual information with the risk of introducing noise or irrelevant context.

G Extended Qualitative Examples

In Fig. 18 we provide an extended qualitative examples on object detection performance of VLC Fusion in both dataset, Waymo dataset and ATR dataset, for seen and unseen scenarios.

System prompt and Input prompt template

System Prompt:
"You are a highly specialized assistant that provides concise answers to specific questions about images, responding to each with either True or False only and returning a JSON object with keys 1 through N corresponding to the question numbers, without any additional context or descriptions."

User Prompt: "Answer the following questions based on the given image by returning a JSON object with exactly N keys (the strings "1" through "N"), each mapped to a boolean (True or False) corresponding to its question and nothing else; the image is provided after these questions."
[Question List]

Figure 15: System and user prompt templates for the "generation" stage.

Model	Param in Billions	Time (sec/image)	Condition Accuracy
GPT-4o (baseline)†	> 100	2 sec	100%
Moondream2	1.9	0.7 sec	79.8%
SmolVLM-Instruct	2.2	1 sec	56.3%

†Parameter count is based on public estimates.

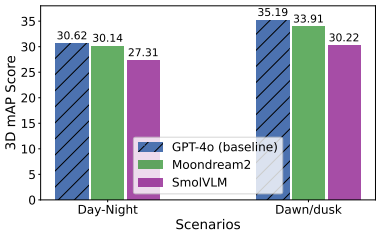
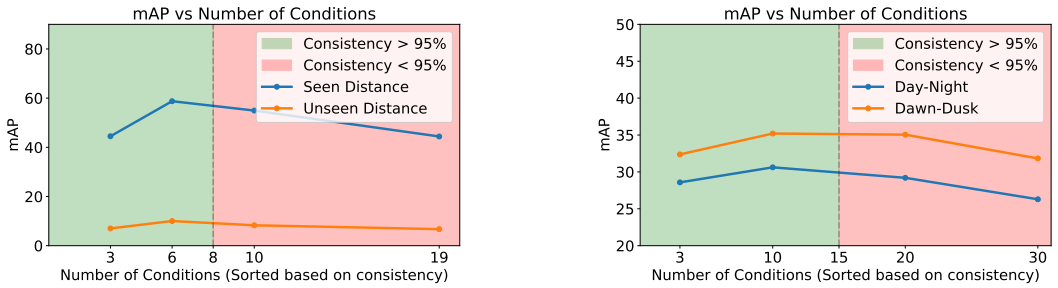


Figure 16: **(a)** As shown in the table, smaller VLMs, such as Moondream2 and SmolVLM-Instruct, deliver faster inference, but yield lower condition-generation accuracy compared to GPT-4o. (Here the accuracy is calculated by considering the response from GPT-4o as ground truth.) **(b)** In zero-shot testing, the corresponding drop in 3D mAP on the Waymo Day-Night and Dawn/Dusk scenarios highlights the importance of condition quality for our method.



(a) ATR dataset (b) Waymo dataset

Figure 17: Performance of VLC Fusion vs. number of environmental conditions. (a) ATR dataset: accuracy peaks at 6 conditions, then declines as less consistent conditions are included. (b) Waymo dataset: accuracy peaks at 10 conditions, followed by a similar decline (see Appendix for condition consistency analysis).

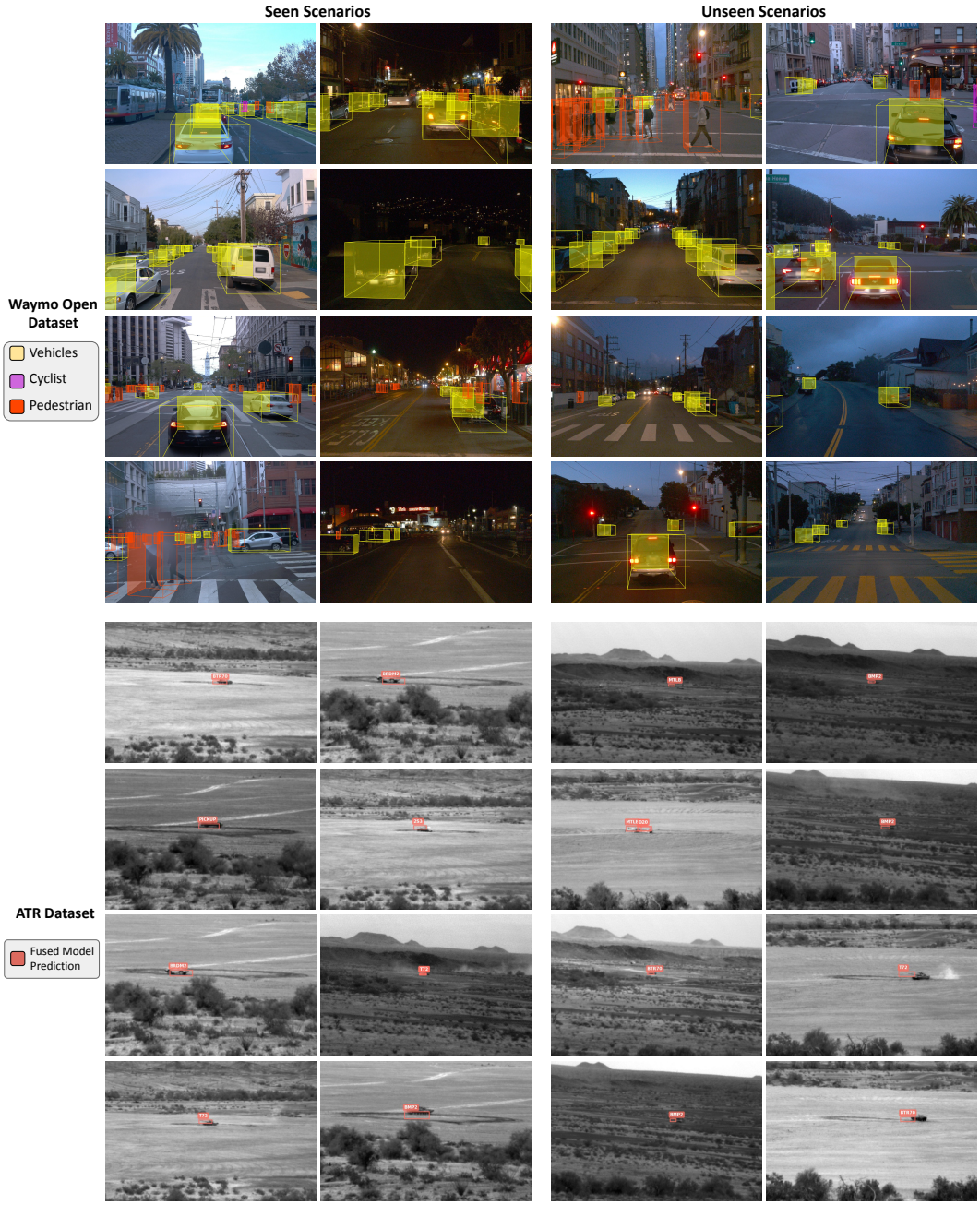


Figure 18: Additional qualitative examples of VLC Fusion for both dataset in seen and unseen scenarios.