EventDrive: Event Cameras for Vision-Language Driving Intelligence

Dongyue Lu[1,6] Rong Li[2] Ao Liang[1] Lingdong Kong[1,3]
Wei Yin[4] Lai Xing Ng[5] Benoit R. Cottereau[6,7] Camille Simon Chane[8] Wei Tsang Ooi[1,6]

[1]NUS [2]HKUST(GZ) [3]CNRS@CREATE [4]Horizon Robotics [5]A*STAR, I^{2}R
[6]IPAL, CNRS IRL 2955, Singapore [7]University Toulouse, CNRS, CerCo, Toulouse, France
[8]ETIS UMR 8051, CY Cergy Paris University, ENSEA, CNRS, France

Project Page: github/EventDrive
Dataset & Toolkit: huggingface/EventDrive

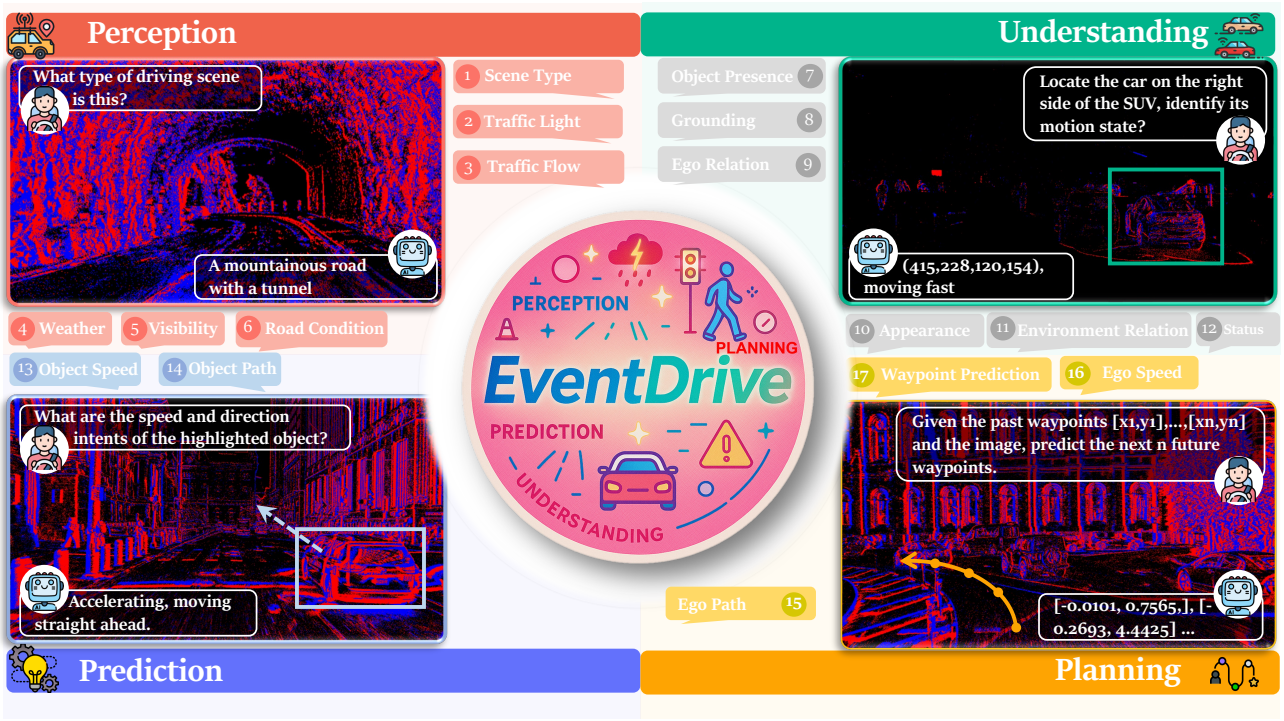

Figure 1. **Overview of the *EventDrive* benchmark.** The dataset contains **471k** event–frame–language samples across four levels of driving reasoning spanning **17** subtasks: **Perception** evaluates *scene-level context* such as scene type, traffic, and illumination. **Understanding** assesses *object-centric semantics*, including presence, motion state, and grounding. **Prediction** infers short-horizon *motion intent of surrounding agents*. **Planning** estimates *ego motion intent and future waypoints* from past trajectories. Each task is formulated through structured, language-grounded queries, enabling unified evaluation of event–frame models across the autonomy stack.

Abstract

*Event cameras sense the world through asynchronous brightness changes with microsecond latency and high dynamic range, offering motion fidelity far beyond frame-based sensors and capturing temporal structure that conventional exposures often miss. These properties make events a powerful complement to RGB in autonomous driving, especially under blur, glare, and rapid motion, where frame-based perception can become unreliable. However, existing event-aware vision–language models remain limited to generic perception and do not reveal how event sensing contributes to reasoning and decision-making across the full driving loop. We present **EventDrive**, a large-*

scale benchmark and model suite that unifies event streams, RGB frames, and language supervision across four core dimensions: ***Perception, Understanding, Prediction, and Planning****, covering captions, structured QA, grounding, motion-state recognition, trajectory forecasting, and planning tasks. Building on this foundation,* ***EventDrive****-VLM introduces a multi-horizon event pyramid and a temporal-horizon mixture-of-experts module to adaptively encode and fuse asynchronous and frame-based information for downstream reasoning. Comprehensive evaluation across diverse tasks shows that event streams provide substantial gains in temporal precision, motion awareness, and robustness, bringing event sensing into the center of driving intelligence.*

1. Introduction

Event cameras have gained increasing attention for their ability to record dynamic scenes with microsecond temporal precision [7, 20]. Unlike RGB sensors that integrate intensity over fixed exposure periods, event cameras report asynchronous brightness changes at the pixel level [40]. This design yields high dynamic range, very low latency, and natural robustness to motion blur. These sensing advantages are especially valuable in autonomous driving, where reliable perception must be maintained across fast ego motion, rapidly moving agents, and challenging illumination conditions [11, 62]. These characteristics position event sensing as a promising complement to frame-based perception in safety-critical driving systems.

Despite this potential, events for driving research remain fragmented. Most existing efforts focus on upstream supervised tasks, *e.g.*, detection [27, 42, 53], segmentation [35, 38, 65], or optical flow estimation [71, 74]. Very few works explore high-level reasoning and decision-making that a complete driving stack requires. In contrast, the RGB community has made rapid progress toward unified vision and language models that couple perception, reasoning, and control within a single architecture [30, 31, 41, 47, 48, 68], which provides interpretability and scalable instruction-driven behavior. Early attempts to incorporate events into vision and language systems, including grounding [39] and caption-based event language models [46, 52, 72], demonstrate encouraging multimodal interactions but remain limited to generic scenes. They do not address the reasoning or decision demands that are central to real-world driving. This gap calls for a framework that integrates events throughout the autonomy pipeline in a unified and scalable manner, rather than treating it as an isolated temporal cue.

To address this need, we introduce ***EventDrive***, a unified dataset and benchmark that integrates event streams, synchronized RGB frames, and language supervision across the **full driving loop**. Our goal is to move from passive sensing to actionable understanding within a coherent multimodal interface. We decompose driving into four sequential reasoning stages: **perception**, **understanding**, **prediction**, and **planning**, each expressed as a language-grounded task probing a complementary aspect of event-based reasoning. Perception assesses robustness under challenging illumination and motion, where events offer stable edges and temporal gradients that compensate for degraded RGB signals. Understanding targets object semantics and spatial relations, with asynchronous event cues helping to disambiguate interactions. Prediction evaluates short-term behavior anticipation, where the high temporal density of events exposes velocity and acceleration. Planning examines ego intent and waypoint estimation, leveraging continuous temporal structure for steadier decisions in dynamic environments. Together, these tasks form a unified evaluation protocol that highlights how temporal cues enhance perception and reasoning across the driving stack while remaining compatible with classical vision models and modern vision–language architectures.

Building upon ***EventDrive***, we develop ***EventDrive*-VLM**, an event-driven vision–language model that integrates asynchronous event cues into unified multimodal reasoning. The framework tackles two core challenges in event-based driving. First, event streams vary widely in temporal density. To capture motion across scales, we introduce a **Dynamic Horizon Encoding module** that voxelizes events at multiple temporal resolutions and selects the most informative representation via a *mixture-of-experts* gate. This preserves high-frequency dynamics during fast motion while ensuring stable aggregation in low-motion regimes. Second, event features must be aligned with the LLM's semantic space. We address this with an **Event Q-Former Alignment module** that performs cross-attention between learnable event queries and temporally encoded representations, enabling motion-focused signals to integrate cleanly with language reasoning. A lightweight two-stage curriculum first learns event–language alignment with the LLM frozen, then performs instruction tuning to produce a coherent event-driven perception-to-action pipeline.

Comprehensive experiments on ***EventDrive*** reveal strong modality contrasts. Frame-only VLMs perform well in perception but degrade sharply under low light and motion blur, and show limited spatial reasoning and weak motion inference. Event-only models excel in speed and direction prediction, highlighting the value of high-frequency temporal cues, yet lack semantic richness for appearance-heavy understanding. Our event–frame fusion model improves performance across all task families, stabilizing perception under adverse conditions, enhancing grounding and relational reasoning, and delivering stronger motion prediction and ego intent estimation. These results show that events and frames provide complementary strengths, and

their integration yields more reliable multimodal reasoning than either alone.

In summary, our main contributions are as follows:

- We introduce ***EventDrive***, the first full-stack event and language benchmark for autonomous driving that unifies perception, understanding, prediction, and planning within a consistent multimodal framework.
- We propose ***EventDrive***-VLM, a general training framework that equips large vision and language models with the ability to interpret, align, and reason over asynchronous event representations.
- We establish a comprehensive evaluation protocol and extensively analyze how temporal cues improve multimodal reasoning, offering a foundation for future event-driven intelligence in real-world driving.

2. Related Work

Event Cameras for Driving. Event sensors offer microsecond latency, high dynamic range, and robustness to motion blur, making them ideal for driving perception. They have been applied to detection [27, 42, 53, 56, 70], segmentation [35, 38, 65], tracking [71], and recognition [16, 24, 36]. Detection methods include graph- or spike-centric models preserving sparsity and asynchrony [17, 18, 22, 55, 59, 63, 71], and dense feed-forward models that convert events into voxel grids or time surfaces for image-based backbones [10, 27, 32, 34, 44, 56, 57]. Beyond detection, event-based segmentation and tracking emphasize temporal continuity and motion cues [25, 39, 65]. However, most systems remain limited to low-level perception without connecting to higher-level reasoning and planning. Our work advances this direction by integrating event sensing into a unified framework that links perception, understanding, and decision-making for end-to-end driving intelligence.

Event-Frame Multimodal Learning. Events and RGB frames are complementary: events capture fine temporal dynamics under extreme lighting, while frames provide rich semantic context. Early pipelines relied on late fusion [9, 43], whereas later designs introduced shared backbones and attention-based gating for cross-modal interaction [5, 53, 66]. Transformer-based approaches further improved temporal reasoning and adaptive weighting under complex motion and illumination [42, 69, 73], benefiting tasks such as deblurring, depth estimation, and segmentation [26, 29, 64]. Yet most frameworks rely on fixed temporal windows, which fail to capture motion cues that unfold over varying temporal scales [6, 23, 42]. Our method addresses this by aligning multi-horizon event streams with frame semantics, enabling frequency-adaptive fusion that maintains both temporal precision and semantic coherence.

Event-based VLMs. While vision-language models (VLMs) have transformed visual understanding, extending them to asynchronous event data remains challenging [38, 39]. The key difficulty lies in embedding sparse, high-temporal-resolution signals into a shared vision-language space. Early contrastive approaches adapt CLIP-style alignment for zero-shot recognition [38, 69], but rely on limited-scale datasets and rasterized inputs, reducing temporal fidelity. Recent works narrow this gap: EventGPT [52] couples an event encoder and temporal aggregator with an LLM; EventVL [46] trains on large-scale event-image-text pairs with spatiotemporal and semantic alignment; and LLaFEA [72] integrates frame-event fusion with cross-attention and duration cues. Despite these advances, most models remain confined to captioning or short QA tasks. On the contrary, our method preserves temporal sparsity and extends event-based VLMs toward comprehensive driving reasoning across diverse tasks.

Table 1. Statistics across four driving reasoning tasks.

Task	Source	#Train	#Test	#Hard
Perception	DSEC [28]	53,196	12,828	780
	M3ED [8]	65,280	32,136	15,396
	PKU [42]	46,860	11,400	3,960
Understanding	DSEC [28]	91,716	28,848	1,812
Prediction	M3ED [8]	7,784	2,810	1,970
Planning	M3ED [8]	46,290	29,526	18,951
Total		**311,126**	**117,548**	**42,869**
Grand Total Samples			**471,543**	

3. *EventDrive*: A Vision–Language Benchmark for Event-Based Driving

In this section, we present the ***EventDrive*** dataset and benchmark in detail. Sec. 3.1 introduces the overall motivation and task hierarchy that define the dataset design. Sec. 3.2 describes the annotation pipeline and data composition, highlighting how multimodal inputs are processed into language-grounded supervision. Sec. 3.3 summarizes the data splits and benchmark statistics, outlining the scale and coverage of the dataset and its comparison with existing event-based vision–language benchmarks.

3.1. Hierarchical Task Framework

Event cameras remain reliable under low light and rapid motion, offering advantages for safety-critical driving where conventional frames often fail. Existing event-based works focus mainly on isolated perception tasks, while RGB-based methods have progressed toward unified vision–language–action frameworks. To close this gap, we build a dataset that extends event understanding across the entire autonomy stack. The driving loop is organized into four sequential steps: environmental **perception**, object-centric **understanding**, forward **prediction**, and ego-driven **planning**. Each step is formulated as a language-grounded

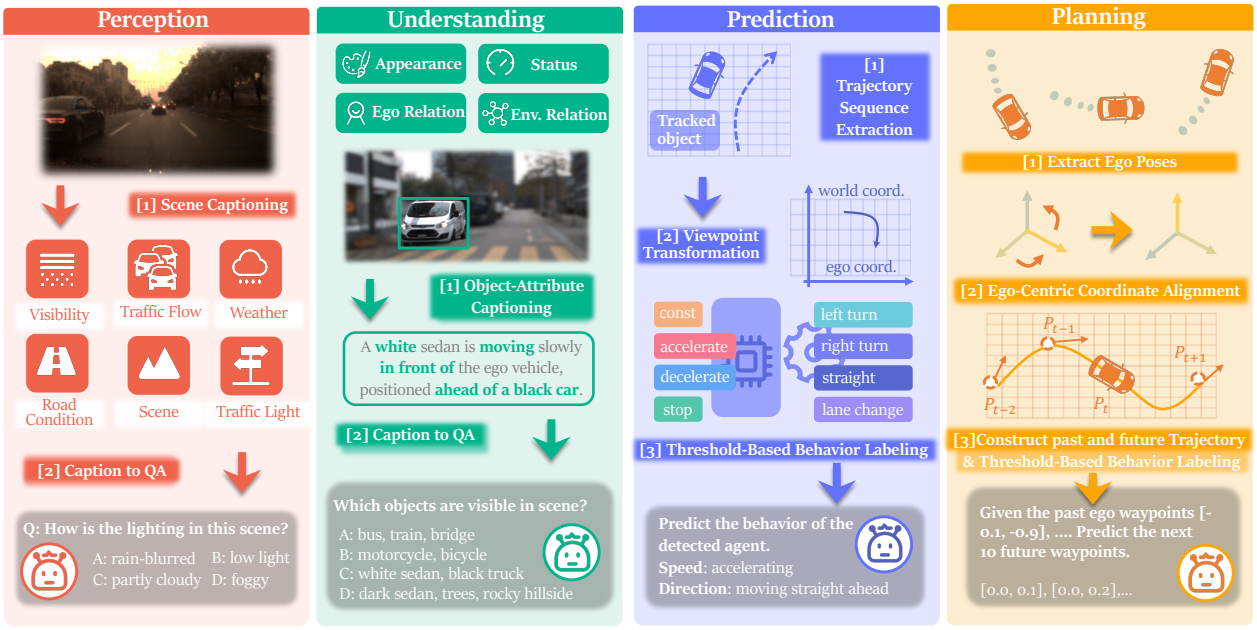

Figure 2. Annotation pipelines of ***EventDrive***. **Perception** converts scene-level attributes into structured QA; **Understanding** generates object-level semantic captions and transforms them into QA; **Prediction** extracts trajectories, applies ego-frame transformation, and assigns motion labels; **Planning** constructs ego-centric waypoints and produces corresponding decision-oriented supervision.

task that aligns event representations with multimodal models, enabling a systematic evaluation of how event signals contribute to high-level reasoning.

Perception describes global scene conditions across six subtasks: [1]*scene type*, [2]*visibility*, [3]*traffic flow*, [4]*weather*, [5]*traffic light*, and [6]*road condition*. Captions and structured QA pairs assess whether models can interpret environmental context from short temporal windows. Event streams preserve edge contrast and temporal gradients under challenging illumination or motion blur, strengthening robustness beyond frame-based perception.

Understanding captures object-level semantics and spatial structure using captions, QA pairs, and bounding boxes spanning [1]*object presence*, [2]*appearance*, [3]*motion state*, [4]*ego relation*, [5]*environment relation*, and [6]*grounding*. The asynchronous event signal supplies fine temporal cues that support reasoning about motion and spatial relations that static frames often miss.

Prediction targets short-horizon behavioral forecasting through indicators of [1]*speed* change and [2]*direction* change. Each instance uses language queries to evaluate whether models can infer future motion trends of surrounding objects from limited observations. The high temporal density of events reveals velocity and acceleration directly, improving the fidelity of motion forecasting.

Planning evaluates ego-centric decision making using [1]*speed intent*, [2]*direction intent*, and [3]*waypoint planning*. Models must convert multimodal observations into coherent driving decisions. Event sensing preserves continuous awareness of dynamic changes, supporting more stable reasoning in rapidly evolving or low-visibility conditions.

3.2. Language-Grounded Data Generation

We construct ***EventDrive*** through a semi-automatic pipeline built upon the event datasets DSEC [28], M3ED [8], and PKU-DAVIS-SOD [42], which span diverse environments and illumination conditions. Using synchronized RGB frames, event streams, bounding boxes, LiDAR, and ego-pose signals, we employ Qwen3-VL [2] to generate linguistically structured supervision across the four task dimensions. The pipeline, shown in Fig. 2, integrates automatic captioning with controlled question–answer generation to ensure both scalability and semantic consistency.

Scene-level **perception** labels are produced by prompting Qwen3-VL to generate global captions describing the six environmental attributes, which are subsequently decomposed into balanced question–answer pairs with explanatory sentences. Object-level **understanding** builds on ground-truth bounding boxes from DSEC. Qwen3-VL first generates descriptions for the five object attributes, and these descriptions are transformed into visual question–answer and grounding tasks that link textual queries to specific spatial regions. For **prediction**, we use ego-pose to project 3D boxes from [49] into the ego frame, extract object trajectories, and convert their motion patterns into natural-language descriptions of speed and direc-

tion intent, paired with corresponding QA items. Ego-level **planning** leverages M3ED trajectory supervision to derive speed intent, path intent, and future waypoints, which are aligned with language queries that evaluate ego decision making. Together, these components yield a unified annotation framework that connects perception, understanding, prediction, and planning through temporally and semantically coherent language-grounded supervision.

3.3. Compositional Structure and Statistics

We adopt standard *training* and *testing* splits and further introduce a **hard** split consisting solely of *low-light* and *motion-blur* sequences. This split enables targeted evaluation of the advantages of event sensing under conditions where frame-based perception degrades. As shown in Tab. 1, the dataset contains **471,543** event–frame–text samples, offering large-scale multimodal supervision across perception, understanding, prediction, and planning. Compared with existing event–language datasets [46, 52, 72], which often rely on simulated data or provide limited real-world coverage below 100k samples, ***EventDrive*** offers a significantly broader and more diverse set of real-world annotations. It supports a wider range of tasks that connect event-driven perception with higher-level reasoning and decision-making, forming a more comprehensive benchmark for evaluating multimodal driving intelligence.

4. *EventDrive*-VLM: A Unified Model for Event-Based Driving Intelligence

Building upon ***EventDrive***, our goal is to develop an event-driven vision–language–action model that can interpret and generate decisions grounded in the spatio-temporal structure of event streams. As shown in Fig. 3, the core of ***EventDrive***-VLM is a multimodal LLM backbone that integrates asynchronous event cues with pre-trained visual–linguistic knowledge to support coherent reasoning across driving tasks. To bridge event dynamics with the LLM embedding space, the model incorporates three components: a **dynamic horizon event encoder** that adapts to varying sampling frequencies and motion patterns (*cf*. Sec. 4.1), an **Event Q-Former** that extracts language-aligned and motion-aware representations (*cf*. Sec. 4.2), and a **two-stage training curriculum** that progressively aligns event, visual, and linguistic pathways for unified multimodal reasoning (*cf*. Sec. 4.3).

4.1. Dynamic Horizon Event Encoder

Event data exhibit large variation in temporal density across datasets and tasks. Sensors operate at different sampling rates, and the required temporal resolution also depends on the objective: perception benefits from broader temporal context, whereas prediction and planning rely on fine-grained motion cues. Conventional voxelization [27] applies a fixed number of temporal bins and thus compresses long exposure windows while blurring fast motion, losing the high-frequency detail essential for motion reasoning.

To address this issue, we propose a **dynamic horizon encoding** strategy that adapts to varying time scales and scene dynamics. Given an event stream $\mathcal{E} = \{e_k\}_{k=1}^{K}$, where each event $e_k = (x_k, y_k, t_k, p_k)$ encodes spatial coordinates, timestamp, and polarity $p_k \in \{-1, 1\}$, standard voxelization maps events into a 4D tensor $\mathbf{E} \in \mathbb{R}^{2\times B\times H\times W}$:

$$\mathbf{E}(p,\tau,x,y) = \sum_{e_k\in\mathcal{E}} \delta(p-p_k)\delta(x-x_k,y-y_k)\delta(\tau-\tau_k), \tag{1}$$

where $\tau_k = \left\lfloor \frac{t_k - t_a}{t_b - t_a} B \right\rfloor$. Instead of relying on a single bin size B, we construct multiple voxel tensors using temporal resolutions $\mathcal{B} = \{b_n\}_{n=1}^{N}$, producing event tensors $\mathbf{E}_n$ that capture short-, medium-, and long-horizon motion patterns.

To adaptively select the most suitable temporal horizon for different motion patterns and task requirements, we employ a *Mixture-of-Experts (MoE)* [60] gating mechanism that dynamically weights multi-scale temporal experts according to scene dynamics. Each expert network [27] specializes in one temporal resolution, processing a voxelized tensor $\mathbf{E}_n$ with bin size b_n to produce an encoded feature

$$\mathbf{F}_n = \boldsymbol{\sigma}(\mathbf{E}_n), \qquad \mathbf{F}_n \in \mathbb{R}^{H\times W\times d}, \tag{2}$$

where d is the channel dimension. The resulting expert features are concatenated into a contextual representation $\mathbf{F}_c \in \mathbb{R}^{H\times W\times (Nd)}$. Global average pooling produces a compact descriptor $\mathbf{f}_c \in \mathbb{R}^{Nd}$ that summarizes multiscale temporal cues. The gating logits are then computed as

$$\mathbf{z} = \mathbf{W}_g \mathbf{f}_c + \texttt{Softplus}\big(\boldsymbol{\epsilon} \odot (\mathbf{W}_{\text{noise}} \mathbf{f}_c)\big), \tag{3}$$

where $\mathbf{W}_g, \mathbf{W}_{\text{noise}} \in \mathbb{R}^{Nd\times N}$ are trainable parameters, and $\boldsymbol{\epsilon} \sim \mathcal{N}(0,1)$ introduces controlled stochasticity to encourage expert diversity. Retaining only the largest k logits focuses computation on the most relevant temporal experts, and softmax normalization yields the weights α_n. The aggregated event representation is computed as $\mathbf{F}^e = \sum_{n=1}^{N} \alpha_n \mathbf{F}_n$. This adaptive formulation allows the encoder to emphasize high-resolution temporal features when motion is fast and to leverage coarse but stable aggregation when dynamics are mild. As a result, the model maintains temporal fidelity across a wide range of driving scenarios while remaining computationally efficient.

4.2. Event Q-Former Alignment

After obtaining the aggregated event representation $\mathbf{F}^e$, we align it with the LLM embedding space using an **Event Q-Former (EQA)**. Rather than concatenating event and frame tokens, which ignores modality asymmetry and leads to high computational cost, the EQA employs cross-attention

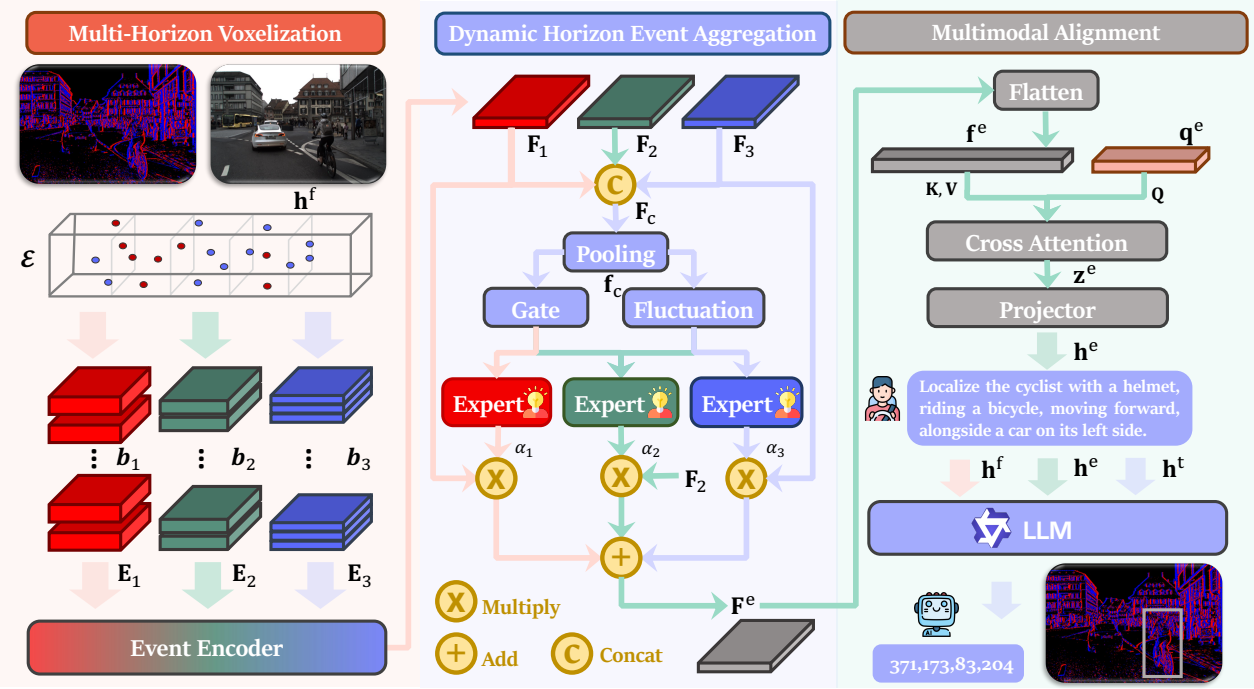

Figure 3. ***EventDrive*-VLM Overview.** We first convert asynchronous events into **multi-horizon voxel tensors** that capture motion at *different temporal scales*. A **dynamic horizon event encoder** then aggregates these representations through a *Mixture-of-Experts* gating mechanism (*cf*. Sec. 4.1). An **Event Q-Former** performs cross-attention to extract language-aligned, motion-aware tokens, enabling coherent fusion of multi-modal features within the LLM for unified driving reasoning (*cf*. Sec. 4.2). A **two-stage training curriculum** further aligns event, visual, and linguistic pathways, strengthening cross-modal grounding and downstream reasoning performance (Sec. 4.3).

to extract the motion-relevant components of the event representation that matter for language-guided reasoning.

Following the Q-Former architecture [45], we introduce a set of learnable event query tokens $\mathbf{q}^e \in \mathbb{R}^{N_q \times d}$, which attend to the event feature map $\mathbf{F}^e \in \mathbb{R}^{H \times W \times d}$. After flattening $\mathbf{F}^e$ into $\mathbf{f}^e \in \mathbb{R}^{(HW) \times d}$, the attended event embeddings are computed as

$$\mathbf{z}^e = \texttt{softmax}\left(\frac{(\mathbf{q}^e\mathbf{W}_Q)(\mathbf{f}^e\mathbf{W}_K)^\top}{\sqrt{d}}\right)(\mathbf{f}^e\mathbf{W}_V), \quad (4)$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d \times d}$ are learnable projections. The output $\mathbf{z}^e \in \mathbb{R}^{N_q \times d}$ forms a compact representation that summarizes the most salient temporal and motion cues.

The Event Q-Former enables each query to attend selectively to temporally informative regions in the event stream, producing motion-aware embeddings suitable for downstream multimodal reasoning. This structured attention preserves the temporal distinctiveness of events while supporting coherent interaction with frame and text representations. A lightweight projection layer maps the attended event features into the LLM embedding space, yielding event tokens $\mathbf{h}^e$ that align with the projected frame tokens $\mathbf{h}^f$ and text embeddings $\mathbf{h}^t$. The concatenated sequence is then used as the input to the LLM decoder, enabling unified multimodal reasoning across all driving tasks.

4.3. Training Curriculum

We adopt a two-stage training curriculum that progresses from event–language grounding to multimodal instruction following, enabling stable and efficient adaptation of the event-driven VLM.

Event–Language Pre-adaptation. In the first stage, the LLM and the frame visual encoder are kept frozen, while the event encoder, Q-Former, and projection layers are trained with a language modeling objective on paired caption data. Although frame data are available, gradients flow only through the event pathway, allowing the model to form a linguistically aligned event representation without altering the pretrained frame semantics. This generative supervision encourages the event encoder and Q-Former to organize temporal cues and motion structure into embeddings compatible with the LLM space, providing a stable form of cross-modal alignment.

Instruction Tuning. In the second stage, we unfreeze the transformer blocks of the LLM and fine-tune the entire event pathway together with these LLM layers on all caption and QA data, while keeping the frame visual encoder frozen. This phase integrates temporal and semantic

Table 2. **Comparison across four driving–reasoning tasks** in the ***EventDrive*** benchmark. For **perception**, we report `QA Accuracy` on three subsets. For **understanding**, we report `QA Accuracy`, grounding Top-1 `Accuracy` at IoU 0.6, and **mIoU**. For **prediction** and **planning**, we report `Speed Accuracy` and `Path Accuracy`, and for planning, we also report the mean **L2 Error** (lower is better). All scores are reported in percentage (%), except for L2 Error in meters. Best results are shown in **bold** and 2nd-best results are underlined.

Method	Perception			Understanding			Prediction		Planning		
	Acc@D	Acc@M	Acc@P	Acc	Acc@60	mIoU	Speed	Path	Speed	Path	L2 Error
Event-based Models											
EventGPT-7B [52]	51.40	54.97	52.25	38.78	5.49	8.24	27.84	65.44	28.99	76.08	11.42
EventDrive-VLM	62.51	67.07	62.39	54.21	43.43	47.82	34.68	82.25	42.56	84.64	6.89
Frame-based Models											
LLaVA-v1.6-Mistral-7B-hf [50]	58.65	59.24	50.05	40.37	15.02	29.0	18.22	29.68	9.89	87.51	8.13
LLaVA-OneVision-1.5-8B [1]	80.81	86.35	72.42	61.62	2.91	12.5	14.38	83.91	56.51	58.50	9.19
InternVL2.5-8B [12]	78.42	83.41	75.86	60.02	0.29	2.5	39.72	86.41	12.52	73.34	10.19
InternVL3-8B [75]	78.61	86.27	74.37	60.60	0.24	2.31	4.41	**91.25**	52.04	84.34	9.84
Qwen2.5-VL-7B-Instruct [3]	77.71	76.92	62.46	49.98	63.35	60.11	8.47	81.64	39.32	87.41	8.78
Qwen2.5-VL-7B-Instruct* [3]	81.52	84.69	75.88	58.44	69.94	67.19	36.84	84.94	53.87	89.44	4.54
Event + Frame Models											
EventDrive-VLM	**85.44**	**86.64**	**78.89**	**65.46**	**72.86**	**72.56**	**42.44**	87.49	**57.03**	**92.35**	**3.66**

signals more tightly, enabling consistent reasoning across perception, understanding, prediction, and planning. Joint optimization under multimodal instructions yields coherent grounding between event dynamics and textual responses, forming a unified event-driven perception-to-action model.

5. Experiment

5.1. Experimental Settings

Implementation Details. We fine-tune Qwen2.5-VL-7B-Instruct [3] with a pretrained RVT [27] backbone as the event encoder. Dynamic horizon encoding uses temporal bins $\mathcal{B} = 20, 50, 100$, and the event projector is a linear layer. The RGB visual tower remains frozen. Training adopts AdamW [37] with a cosine schedule on 16 NVIDIA H20 GPUs, using a learning rate of 1×10^{-4} for the event encoder and projector. Mixed precision bf16 is used throughout. Pre-adaptation and instruction tuning each run for two epochs with a batch size of 128. Packed sequences and flattened multimodal inputs are applied with a sequence length of 4096 tokens, and FlashAttention 2 [19] accelerates all attention layers.

Evaluation Metrics. We establish a unified evaluation protocol across all defined tasks. Perception is evaluated by QA accuracy over six scene attributes. Understanding combines QA accuracy across five object-level aspects with grounding metrics, including Top-1 localization accuracy at IoU 0.6 and mean IoU. Prediction assesses speed and path intent accuracy. Planning evaluates intent accuracy together with mean L2 waypoint error across 1, 3, and 5 seconds. Additional details appear in the supplementary materials.

Baselines. Comparisons include both *frame-only* and *event-only* models. Frame-only evaluation covers Qwen2.5 VL [3], InternVL [12, 75], and LLaVA [1, 50] under zero-shot inference, which tests robustness in low light and motion-blurred scenes. To isolate the contribution of events, we additionally fine-tune a Qwen2.5-VL-7B model using the same instruction tuning protocol as our model. For event-only comparison, we evaluate the open-sourced EventGPT in a zero-shot manner since its training code is unavailable. We further report zero-shot results on the released portion of the Event-Chat [52] dataset, noting that the absence of an official split limits this evaluation to reference use.

Table 3. **Comparative results on the Event-Chat dataset [52]**, which evaluates event-driven description/reasoning through **Detailed Captioning**, **Complex Reasoning**, and **Visual Question Answering**. Higher values indicate better performance.

Models	Params	DC	CR	VQA
Frame-based Models				
LLaVA-7B-v1.5 [51]	7B	2.20	4.04	3.26
Qwen2-VL-7B [67]	7B	2.38	4.02	2.91
InternVL2-8B [14]	8B	2.37	4.00	3.71
Deepseek-vl-7b [54]	7B	2.41	4.10	3.37
Event-based Models				
EventGPT-7B [52]	7B	**3.52**	4.09	**4.29**
EventDrive-VLM	7B	3.43	**4.15**	3.94

5.2. Comparative Studies

Comparisons with Event-Only Methods. Event-only models in Tab. 2 perform well on motion-centric tasks such as speed and path intent, confirming that high-frequency temporal cues are inherently well suited for short-horizon motion reasoning. They also show reasonable grounding performance despite lacking explicit spatial supervision, highlighting the structural information encoded in event

streams. Yet, the absence of appearance cues limits their semantic understanding, resulting in lower perception and object-centric accuracy. Results on the Event-Chat benchmark (Tab. 3) further show that the representations learned from ***EventDrive*** transfer across datasets, indicating that the dataset promotes generalizable event–language alignment rather than overfitting to a specific annotation style.

Comparisons with Frame-Only Methods. Frame-based VLMs achieve high accuracy on perception under normal conditions but degrade sharply in `Acc@P`, understanding `Acc@60`, and speed-related prediction metrics due to their inability to encode motion or handle low-light and blur. These limitations lead to unstable spatial reasoning and poor motion intent inference. Introducing events alleviates these issues: event–frame fusion improves all metrics, especially grounding and speed intent, showing that temporal gradients and motion cues are essential for reliable reasoning and cannot be inferred from RGB imagery alone.

Comparisons across Different Tasks. The four task families reveal complementary strengths of different modalities. Perception is broadly solvable, but event streams offer improved robustness under visual degradations. Understanding exposes significant variance across VLMs, as many struggle with spatial relations and grounding from a single RGB frame; temporal cues help resolve these ambiguities. Prediction presents the largest modality gap: inferring speed or direction from static frames is ill-posed, whereas events directly encode motion, leading to consistently higher accuracy for event-enhanced models. Planning mirrors this pattern, where temporal dynamics from traffic improve speed intent estimation and reduce trajectory L2 error. These trends collectively demonstrate that events provide structural and temporal information that complements semantics from frames across the full driving chain. A qualitative comparison is shown in Fig. 4.

5.3. Ablation Studies

Multi-Horizon Voxelization. We evaluate the impact of using different numbers of temporal resolutions when voxelizing event streams. As shown in Tab. 4, increasing the number of horizons from one to five improves performance across all tasks, particularly in understanding and planning, where temporal diversity helps disambiguate object motion and spatial relations. However, the performance gain saturates beyond three horizons, while computational cost grows proportionally with the number of voxelized tensors. Our choice ($N = 3$), therefore, provides the best trade-off between accuracy and efficiency, capturing both short-range motion cues and longer-term temporal context without incurring unnecessary overhead.

Dynamic Horizon Event Aggregation. We compare different strategies for aggregating multi-horizon event features. A naïve summation ("Add") collapses temporal distinctions and yields the weakest results, confirming that indiscriminate fusion fails to preserve horizon-specific information. Weighted summation ("Wt.sum") improves performance by allowing the model to favor informative horizons, but still underperforms compared with selecting a single expert. Our MoE-based dynamic horizon module, which activates only the top-scoring expert, achieves the best over-

Table 4. Ablation results on the ***EventDrive*** benchmark. We report average `QA Accuracy` for perception, `QA Accuracy` and **mIoU** for understanding, and mean `Speed`/`Path Accuracy` for prediction and planning. Planning additionally includes mean **L2 Error** in meters (lower is better).

Method	Per.	Und.		Pre.	Plan.	
	Acc	Acc	mIoU	Acc	Acc	L2 Error
Voxelization						
N = 1	82.40	62.33	69.52	63.96	71.18	4.11
N = 5	**83.95**	64.97	72.25	62.78	73.49	3.88
Dynamic Horizon Event Aggregation						
Add	76.76	63.89	67.64	62.50	72.49	4.57
Wt.sum	83.84	64.67	70.56	61.08	74.10	3.75
Multimodal Alignments						
Concat	79.35	61.08	71.93	64.76	72.62	4.01
Attention	81.25	65.12	70.23	62.14	**75.85**	3.69
Ours	83.66	**65.46**	**72.56**	**64.96**	74.69	**3.66**

Figure 4. Qualitative results on ***EventDrive*** comparing ***EventDrive***-VLM with Qwen. Events remain reliable under low light and motion, improving scene perception, object understanding, motion prediction, and ego intent estimation.

all results, indicating that event representations benefit from horizon specialization and that suppressing irrelevant resolutions is more effective than blending them.

Multimodal Alignment. We study the alignment between event features and the multimodal embedding space. Simple concatenation of event and frame tokens leads to suboptimal performance due to modality imbalance and increased sequence length. Cross-attention improves grounding and planning by enabling token-level interactions, yet remains inferior to our Event Q-Former. The Q-Former achieves the best results while maintaining a lower computational cost, as learnable queries extract only the most salient motion patterns rather than attending to the full spatiotemporal map. This confirms that structured, query-centric alignment offers both efficiency and stronger motion abstraction.

6. Conclusion

In this work, we introduce ***EventDrive*** and ***EventDrive***-VLM to unify event sensing, frames, and language across the full driving stack. Our results show that high-frequency temporal cues provide complementary strengths to RGB and significantly enhance multimodal perception, reasoning, and trajectory forecasting. Event–frame fusion improves robustness and motion understanding, highlighting events as a key modality for future driving systems.

Acknowledgments

This work is under the programme DesCartes and is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. The authors also gratefully acknowledge Horizon Robotics for its support and computational resources.

The authors would like to sincerely thank the Program Chairs, Area Chairs, and Reviewers for the time and effort devoted during the review process.

Appendix

A. *EventDrive* Dataset

EventDrive is a unified event-frame driving benchmark designed to support the full spectrum of driving reasoning, spanning **perception**, **understanding**, **prediction**, and **planning**. It integrates heterogeneous multimodal data from three complementary sources: M3ED, DSEC, and PKU-DAVIS-SOD, covering diverse illumination regimes, motion patterns, and annotation types. This section first introduces the source datasets and their sensing characteristics, then describes our cross-dataset construction strategy for forming coherent splits for each reasoning level. Finally, we outline the annotation pipeline that converts raw multimodal streams into structured, language-driven supervision.

A.1. Source Datasets

A unified event-frame driving benchmark requires diverse sensing conditions, high-speed motion patterns, and dense multimodal annotations. To construct ***EventDrive***, we draw from three complementary datasets: M3ED Chaney et al. [8], DSEC [28], and PKU-DAVIS-SOD [42], each providing unique combinations of resolution, temporal fidelity, scene diversity, and supervision quality. A comparison of these datasets is shown in Tab. 5. Together, they offer a wide range of illumination and motion regimes and rich annotations spanning 2D/3D perception, tracking, and segmentation, forming a comprehensive foundation for event-driven multimodal reasoning.

A.1.1. M3ED Dataset

The **M3ED dataset** is a large-scale multimodal benchmark designed for high-speed and high-dynamic robotic perception across diverse environments. Unlike prior event-camera datasets that target a single platform or homogeneous driving scenes, M3ED spans **multiple robotic platforms**, including an autonomous ground vehicle, a quadruped robot, and a UAV, all equipped with a unified sensor suite. Each platform streams synchronized Prophesee Gen4 event cameras (1280×720), global-shutter grayscale and RGB cameras (1280×800), a 64-beam Ouster LiDAR, and a high-quality VectorNav IMU, enabling consistent multimodal perception under extreme motion and illumination changes.

The dataset contains **70** sequences collected across urban streets, indoor corridors, forest trails, and aerial environments, totaling more than 3 TB of data. These sequences cover a wide range of dynamic conditions: high-speed car driving in dense traffic and tunnels, aggressive UAV flights through cluttered vegetation and urban canyons, and quadruped locomotion with gait-induced vibrations on uneven terrains. Such diversity produces extremely high event rates, often exceeding 200 MEPS during rapid rotations or high-texture scenes, making M3ED a challenging benchmark for temporal reasoning and motion-aware perception.

M3ED provides rich ground-truth supervision for multimodal learning. This includes **global pose trajectories** estimated by FasterLIO, **dense depth maps** computed from accumulated LiDAR sweeps, **2D semantic segmentation** with eleven categories aligned with the DSEC taxonomy, and **3D instance annotations** for pedestrians, vehicles, buildings, and trees. The dataset additionally supports evaluation of event-based optical flow and odometry, facilitating research across both low-level motion estimation and high-level semantic understanding.

By combining heterogeneous platforms, synchronized multimodal sensors, and geographically diverse environments, M3ED offers a comprehensive and realistic foundation for studying event-driven perception. Its mixture of high-speed motion, complex illumination, and irregular camera trajectories provides a challenging setting that enables robust evaluation of temporal fusion, motion forecasting, and multimodal reasoning algorithms in real-world robotic scenarios.

A.1.2. DSEC Dataset

The DSEC dataset is a large-scale multimodal driving benchmark collected using a tightly synchronized stereo sensor rig composed of two Prophesee Gen3.1 event cameras (640×480) and two FLIR Blackfly S RGB cameras operating at 20 Hz. All four sensors are factory-calibrated, geometrically rectified, and time-synchronized at the microsecond level, enabling pixel-accurate fusion of RGB frames and asynchronous events. The recordings span multiple Swiss cities, including Zurich, Thun, and Interlaken, and cover a wide variety of driving environments such as busy downtown streets, narrow suburban lanes, mountain tunnels, and high-speed highway segments. In total, DSEC contains over one hour of real-world driving with typical sequence durations of 2–8 minutes.

Compared with earlier event-camera driving datasets, DSEC emphasizes geometric accuracy and multimodal alignment. The dataset provides stereo rectification, precise extrinsics, and high-fidelity rolling-shutter compensation for RGB cameras, ensuring consistency between modalities even under high-speed motion. The event stream captures fine-grained temporal structure with microsecond resolution, making DSEC well suited for studying motion blur suppression, optical flow estimation, and low-latency perception in dynamic urban scenes.

Building on DSEC, DAGr [23] introduces **DSEC-Detection**, an enhanced object detection benchmark that augments the original sequences with dense, spatially aligned **2D bounding-box** annotations. A depth-free geometric warping procedure is applied to map RGB frames into the event-camera viewpoint, reducing residual parallax to within 6 px across the image plane. Object trajectories are first generated using QDTrack and subsequently verified and corrected by human annotators. Additional pedestrian-centric scenes are introduced to mitigate the category imbalance inherent in the original dataset.

DSEC-Detection offers high-quality annotations for eight categories across long, continuous sequences. Notably, approximately one-third of the annotated frames contain object appearance or disappearance induced by fast ego-motion, dynamic occlusions, or sudden viewpoint changes, making DSEC-Detection a challenging benchmark for event-driven detection and motion-aware fusion. Together, DSEC and DSEC-Detection provide a geometrically precise, temporally rich, and motion-intensive foundation for evaluating asynchronous perception and event–frame alignment algorithms in real-world driving conditions.

A.1.3. PKU-DAVIS-SOD Dataset

The PKU-DAVIS-SOD dataset is a large-scale multimodal benchmark designed for real-world object detection under challenging illumination and motion conditions. It is collected using a DAVIS346 sensor that streams high-frequency DVS events together with global-shutter RGB frames (346×260 at 25 Hz). The combined sensing pipeline enables robust annotation even in scenarios that traditionally hinder frame-based perception, such as extreme motion blur, low-light urban environments, and sudden illumination transitions. Overall, the dataset consists of **220 continuous driving sequences**, offering long, uninterrupted event–frame streams rather than isolated clips.

Table 5. Comparison of the three source datasets used to construct ***EventDrive***. All datasets provide synchronized RGB frames and event streams with diverse environments and motion characteristics.

Dataset	#Seq	#Time (s)	Resolution	FPS	#Frames	Annotations	Domain
M3ED [8]	70	12,237	1280×720	30 Hz	>300,000	Pose, depth, 2D/3D seg.	Driving, UAV, quadruped
DSEC [23, 28]	60	∼3,600	640×480	20 Hz	78,344	Depth, flow, 2D boxes/seg.	Driving
PKU-DAVIS-SOD [42]	220	∼11,000	346×260	25 Hz	∼276,000	2D boxes	Driving

Table 6. Statistics of the ***EventDrive*** dataset across tasks, sequences, and annotated frames.

Task	Source	Train			Test			Hard		
		#Seq	#Frame	#Data	#Seq	#Frame	#Data	#Seq	#Frame	#Data
Perception	DSEC [28]	47	4,433	53,196	12	1,069	12,828	1	65	780
	M3ED [8]	3	5,440	65,280	2	2,678	32,136	1	1,283	15,396
	PKU [42]	31	3,905	46,860	10	950	11,400	3	330	3,960
Understanding	DSEC [28]	47	4,433	91,716	12	1,069	28,848	1	65	1,812
Prediction	M3ED [8]	3	3,892	7,784	2	1,405	2,810	1	985	1,970
Planning	M3ED [8]	3	15,430	46,290	2	9,842	29,526	1	6,317	18,951
Grand Total Samples						**471,543**				

A distinguishing feature of PKU-DAVIS-SOD is its dense, frame-wise bounding-box supervision. The benchmark provides **276k labeled timestamps** and **1.08M manually annotated bounding boxes** across three object categories: **cars**, **pedestrians**, and **two-wheelers**. To ensure accuracy under fast motion or low-light conditions, annotations are supported by event-reconstructed grayscale images that preserve structural edges where RGB frames may fail. The dataset is split into 671.3k / 194.7k / 214.1k bounding boxes for train/val/test, maintaining balanced distributions of object sizes (small, medium, large) within each split.

The sequences capture a wide spectrum of urban driving conditions. Approximately 92% of data is recorded under normal illumination, while 8% focuses on low-light scenes including dimly lit roads, urban night drives, and shadowed intersections. Motion patterns also vary significantly: normal-speed trajectories account for 87% of frames, whereas 13% correspond to high-speed or motion-blur scenarios where events provide crucial temporal detail unavailable in RGB alone. These diverse conditions make PKU-DAVIS-SOD a valuable testbed for studying multimodal robustness and motion-induced edge cases.

Beyond object detection, PKU-DAVIS-SOD enables research in asynchronous fusion, temporal reasoning, and motion-aware event processing. Its long continuous streams, high annotation density, and broad coverage of real-world driving scenarios offer a strong foundation for evaluating fine-grained perception algorithms that must handle illumination variability, high-speed dynamics, and sparse-but-informative event modality signals.

These three datasets collectively provide (1) heterogeneous viewpoints and motion regimes (ground, aerial, quadruped), (2) multiple sensor generations (Gen3.1, Gen4 event cameras), (3) both short and long temporal horizons, and (4) dense multimodal annotations spanning 2D/3D detection, segmentation, optical flow, and odometry. This diversity allows ***EventDrive*** to unify perception, understanding, prediction, and planning within a single event-frame language model while maintaining broad generalization across motion scales, illumination conditions, and real-world driving scenarios.

A.2. Dataset Construction Strategy

To construct a unified multimodal benchmark that spans perception, understanding, prediction, and planning, we curate data from three complementary event-frame datasets: **DSEC**, **M3ED**, and **PKU-DAVIS-SOD**. These datasets jointly provide high-quality RGB-event pairs, dense 2D annotations, LiDAR-supported 3D motion labels, and long continuous trajectories under diverse daytime, nighttime, and motion-blur conditions. However, their sensing configurations, temporal resolutions, and scene distributions differ substantially. We therefore apply task-specific filtering rules and consistent sequence-level splits to form a balanced, coherent corpus for ***EventDrive***.

Perception Split Construction. Perception focuses on object and scene recognition and relies only on synchronized image-event pairs. We include all three source datasets while enforcing dataset-specific sampling procedures to ensure temporal diversity and

avoid redundancy. For **DSEC**, we adopt the official train/validation/test partition. Because consecutive RGB frames change minimally at 20 Hz, we subsample every eighth frame to reduce duplication while maintaining temporal coverage. The sequence *zurich_city_12_a*, a nearly pitch-black urban drive with extremely low illumination, is designated as the hard split. For **M3ED**, we retain only ground-vehicle recordings, discarding UAV and quadruped runs due to incompatible motion characteristics, along with extremely short clips and parking-lot-only trajectories. The final selection includes three daytime sequences (*urban_day_city_hall*, *urban_day_rittenhouse*, *urban_day_ucity_small_loop*) and three nighttime sequences (*urban_night_city_hall*, *urban_night_rittenhouse*, *urban_night_ucity_small_loop*). We follow a symmetric day/night protocol in which *urban_day_rittenhouse* and *urban_night_rittenhouse* form the test set, while the long sequence *urban_night_ucity_small_loop* is used as the hard split. For **PKU-DAVIS-SOD**, which has lower resolution and noisier appearance, we rely on the official validation split containing 28 normal, 10 motion-blur, and 6 low-light sequences. We select eight normal sequences (21-28), one low-light, and one motion-blur sequence for testing, and designate one low-light and two motion-blur sequences as hard cases. This selection preserves a balanced distribution of illumination and motion-blur conditions.

Understanding Split Construction. Scene understanding requires high-quality 2D bounding boxes. We therefore use DSEC exclusively, leveraging the dense annotations of DSEC-DET together with its high-resolution stereo views. We follow the same sequence-level split and 8-frame subsampling strategy used in perception, while additionally ensuring that each selected frame contains at least one valid instance annotation. This produces a reliable, densely annotated corpus for event-driven scene understanding.

Prediction Split Construction. Trajectory prediction depends on accurate agent motion and ego-motion annotations, which M3ED provides through synchronized Ouster LiDAR sweeps, globally registered odometry, and 3D instance tracking from [49]. We inherit the perception split for sequence selection and further refine it by removing segments with unstable ground truth, including those with severely noisy 3D boxes, prolonged occlusions, or substantial pose drift. The resulting subset offers consistent agent trajectories and diverse velocity profiles suitable for motion forecasting.

Planning Split Construction. Planning supervision requires future ego-vehicle trajectories, which are also provided only by M3ED through FasterLIO-based odometry. We adopt the same train/test/hard split as prediction to ensure consistency across the motion-centric tasks. For each retained sequence, we generate high-level driving intents (speed and path) together with 5-second future waypoint sequences sampled at 0.5-second intervals.

```
You are a helpful assistant

Suppose you are driving, and I'm providing you with
the image captured by the car's front center,
generate a description of the driving scene which
includes the key factors for driving planning,
including the scene type (what type of area, e.g,
urban or rural), the visibility in this scene (e.g,
clear or limited), the density of the traffic flow
(e.g, light or congested), the positions and
movements of vehicles and pedestrians; prevailing
weather conditions (e.g, sunny or rainy); time of
day, distinguishing between daylight and nighttime;
road conditions (e.g, dry or wet), indicating
smooth surfaces or the presence of obstacles; and
the status of traffic lights (e.g, red or not
visible), which influence your decision making,
specifying whether they are red or green. The
description should be concise, providing an
accurate understanding of the driving environment
to facilitate informed decision-making.
```

Figure 5. Prompt used to generate a scene caption capturing six essential attributes of the driving scene.

Across all four reasoning tasks, ***EventDrive*** integrates 471,543 multimodal samples drawn from three datasets. The corpus spans urban daytime and nighttime driving, tunnel environments, motion-blur episodes, rapid rotations, and extreme low-light conditions. Compared with previous event-language datasets, our dataset provides the largest event-frame-language corpus for driving, the only dataset that covers all four reasoning levels in a single benchmark, diverse cross-domain and cross-illumination conditions, and long-horizon trajectory supervision necessary for learning event-driven prediction and planning. Tab. 6 summarizes the curated splits.

A.3. Annotation Pipeline

Beyond curating and splitting raw sequences, ***EventDrive*** requires transforming heterogeneous sensor data into unified, language-driven supervision suitable for multimodal reasoning. Each task demands different forms of annotation, ranging from structured scene attributes and object-centric queries to agent trajectories and ego-motion intentions. To achieve this, we design a modular annotation pipeline that combines rule-based filtering, trajectory processing, and large vision-language model prompting.

The pipeline converts raw RGB-event pairs, 2D/3D detections, and LiDAR-supported trajectories into standardized instruction-following QA formats. For semantics-driven tasks (perception and understanding), we employ caption generation and caption-to-QA transformation with controlled distractors. For motion-centric tasks (prediction

```
Suppose you are driving, and I'm providing you with
the image and corresponding event representation
captured by the car's front center. {question} You
should answer one choice from four answers
{choice_str}. Your response must contain both the
option letter and the corresponding text, separated
by a space. For example, respond exactly as 'A
Urban' or 'C Foggy' — do not add punctuation or
explanations.
```

Figure 6. Prompt used to assess the model's **Perception** capability in driving scenes.

and planning), we process agent and ego trajectories in the ego frame and derive high-level speed and path intents via kinematic rules. All annotations are formatted into consistent two-turn Qwen-style conversations, ensuring that every modality, including event, image, and text, contributes a coherent supervisory signal for training unified event-driven VLMs.

A.3.1. Perception Annotation

To construct perception-level annotations, we design a semi-automated pipeline that transforms raw driving images into six structured multiple-choice QA pairs representing key scene attributes. The full process consists of three stages: scene caption generation, caption-to-QA conversion, and final formatting with controlled distractors.

■ ***Stage 1: Scene Caption Generation.*** We begin by generating a rich and driving-oriented description for each RGB frame. A high-capacity vision–language model (Qwen3-VL-30B-A3B-Instruct [2]) receives the front-view image together with a prompt, as shown in Fig. 5, that instructs it to summarize all scene factors essential for driving. This includes environmental type (e.g., urban, suburban), visibility, traffic density, agent motion, weather, time of day, road surface condition, and traffic-light state. Images are processed in batches for efficiency. These captions serve as a structured semantic foundation for subsequent QA construction.

■ ***Stage 2: Caption-to-QA Generation.*** Given the caption for each image, we use Qwen2.5-VL-7B-Instruct to transform the description into six perception QA pairs, one for each predefined category: [1]*Scene type*, [2]*Visibility*, [3]*Traffic flow*, [4]*Weather*, [5]*Traffic light*, and [6]*Road condition*. The prompt (shown in Fig. 10) instructs the model to: (1) produce natural, self-contained questions grounded in the caption, (2) offer four candidate answers (A–D) with exactly one correct option, (3) ensure wording diversity across samples, (4) avoid repetitive patterns in answer ordering, and (5) provide a short justification sentence tied to the caption. The model outputs a strict JSON array with six entries. Outputs that fail JSON parsing are kept for later inspection but

```
Given the image and the corresponding event
representation from the front camera, identify the
bounding box corresponding to the described
object:{qa['question']}

Output ONLY the four integers x1,y1,x2,y2
separated by commas — where:

x1 and y1 are the coordinates of the top-left
corner of the box, and x2 and y2 are the
coordinates of the bottom-right corner of the box.
For example: 534,197,566,267.

Do NOT include any text, description, explanation,
or line breaks — only the numbers.
```

Figure 7. Prompt used to assess the model's **grounding** capability in driving scenes.

```
Suppose you are driving, and the image and the
corresponding event representation is captured
from the vehicle's front camera. In the image, one
object has been highlighted with a bounding box.
{question}

You should answer one choice from four answers
{choice_str}.
Your response must contain both the option letter
and the corresponding text from the option,
separated by a space, do not add punctuation or
explanations.
```

Figure 8. Prompt used to assess the model's **Understanding** capability in driving scenes.

excluded from final training data.

■ ***Stage 3: Multiple-Choice Construction and Formatting.*** The generated QA objects are then converted into the final format used by our training pipeline. Although the LLM provides candidate answers, we replace all options with a controlled set of answer choices for consistency across the dataset. For each attribute category, we maintain a manually curated distractor pool covering realistic but incorrect alternatives relevant to driving scenes (e.g., various weather types, road-surface states, visibility conditions, or traffic-light statuses). For each QA instance, we: (1) extract the correct answer from the LLM output, (2) sample three plausible distractors from the category's distractor pool, (3) shuffle the four options to randomize the correct label position, (4) construct a two-turn conversation in the Qwen training format. The human turn contains the driving context, the question, and the four answer options, while the as-

```
Suppose you are driving, and the image and the
corresponding event representation is captured
from the vehicle's front camera. {question} You
should answer one choice from four answers
{choice_str}.

Your response must contain both the option letter
and the corresponding text from the option,
separated by a space, do not add punctuation or
explanations.
```

Figure 9. Prompt used to assess the model's **object-awareness** capability in driving scenes.

sistant turn outputs the ground-truth choice in the required "letter text" format (e.g., "B Low light").

Once the QA pairs are constructed, we apply the prompting template illustrated in Fig. 6 to perform inference over RGB-event inputs and generate the final perception results used for evaluation.

This pipeline combines generative models with controlled distractor sampling and strict formatting rules to produce high-quality, diverse, and semantically grounded perception annotations. By leveraging LLM-generated captions and ensuring category-level consistency through curated distractor pools, the resulting QA dataset offers a robust benchmark for evaluating scene-level understanding under both RGB and event-based sensing.

A.3.2. Understanding Annotation

The annotation pipeline for object-level understanding follows a structured, multi-stage process that combines LLM-based object captioning, rule-driven QA construction, bounding-box candidate generation, and final conversion into Qwen-style supervision. This dimension requires fine-grained descriptions for individual objects, strict uniqueness validation, and multiple task types (appearance, grounding, relations, etc.).

■ ***Stage 1: Generating Object-Level Referential Descriptions*** For each ground-truth object, we provide an LLM with (1) the full-frame scene image, (2) a masked view where the target is enclosed by a bounding box, and (3) the object's class label. A specialized prompt, as shown in Fig. 14, instructs the model to first perform a series of validity checks, ensuring that the boxed region contains exactly one object of the correct class and that the target is sufficiently visible, unambiguous, and not overly occluded or truncated. If any of these conditions fail, the model must output the rejection statement: *"There is no describable object within the specified bounding box."*

If the region is valid, the model generates a structured referential description following a constrained five-part template, covering [1]*appearance*, [2]*motion state*, [3]*position in the view*, [4]*relation to the viewer*, and [5]*relation to surrounding objects*. The prompt strictly prohibits hallucination, speculation, uncertain attributes, or treating nearby objects as subjects. All spatial terminology follows a fixed vocabulary (e.g., "top-right", "center-left"), and descriptions must uniquely identify the boxed object without mentioning the bounding box itself. This produces consistent, grounded, and disambiguated object-level captions.

■ ***Stage 2: Converting Captions into Six QA Categories.*** Each structured description is then passed to a second LLM, which generates exactly six multiple-choice QA pairs, one for each object-understanding dimension: [1]*object awareness*, [2]*grounding*, [3]*appearance*, [4]*status*, [5]*relation-to-viewer*, and [6]*relation-to-others*. The prompt, as shown in Fig. 15, enforces uniform formatting: four answer options (A–D) with exactly one correct choice, plausible distractors, randomized correct-answer position, and a short justification sentence. For grounding, the model produces a referring expression but does not invent coordinates, as bounding-box candidates are injected later. The LLM returns a clean JSON array containing the six QA entries.

■ ***Stage 3: Preparing Bounding-Box Candidates for Grounding.*** For grounding questions, we construct four candidate bounding boxes by combining the ground-truth box with up to three distractors drawn from other objects in the same frame. If fewer distractors are available, additional jittered boxes with randomized size and location (remaining within image bounds) are synthesized. The four candidates are then shuffled and reassigned labels, and the correct label is recorded. This ensures reliable localization supervision while preventing positional bias.

■ ***Stage 4: Converting QA Items into Qwen Training Format.*** Finally, each QA pair is converted into a two-turn dialogue compatible with Qwen-style multimodal training. For *non-grounding* tasks, the human turn introduces the driving context, highlights that an object is boxed in the image, presents the question and four answer choices, and instructs the model to answer in the exact "letter text" format (e.g., "C Moving left"). The assistant turns outputs the correct label and answer. *Grounding* questions follow a stricter interface: the human turn requests only the four integers (x_1, y_1, x_2, y_2) corresponding to the selected box, with no additional text, and the assistant turn outputs the correct coordinates. Each sample stores the image path, event path, QA category, and the ground-truth box for later evaluation. All QA pairs are flattened so that each becomes an independent training example. Once the QA pairs are generated, we use the prompting templates shown in Fig. 9 for object-awareness questions, Fig. 8 for attribute-based understanding that requires bounding-box inputs, and Fig. 7 for grounding. These templates are applied to RGB–event inputs to obtain the final understanding results used for evaluation.

This four-stage pipeline transforms raw bounding-box annotations into rich object-level supervision: validated referential descriptions, six structured QA pairs per instance, standardized grounding candidates, and fully formatted Qwen dialogue samples. The result is a consistent and scalable object-understanding dataset covering appearance, localization, motion, and relational reasoning across diverse driving scenes.

A.3.3. Prediction Annotation

The prediction annotation pipeline focuses on generating supervisory labels for short-horizon motion prediction of dynamic agents in driving scenes. Unlike perception or understanding, which rely primarily on static visual cues, prediction requires integrating multi-frame geometric information, ego-vehicle pose, and local object motion. The pipeline proceeds through three major stages: (1) deriving object trajectories in the ego coordinate frame, (2) converting physical motion into discrete intent labels, and (3) constructing image-conditioned QA samples for model training and evaluation.

■ ***Stage 1: Ego-Frame Trajectory Construction.*** For each video sequence, we load the LiDAR-based 3D detection and tracking results [49], obtaining the world-frame positions of surrounding dynamic agents at 10 Hz. For each tracked agent, its center position is transformed into the ego-vehicle coordinate system at every timestamp, producing a short-term trajectory

$$\mathcal{Q} = \{\mathbf{q}_t = (x_t,\, y_t,\, z_t)\}_{t=0}^{10},$$

where y denotes the forward axis, x the lateral axis, and the sampling interval is $\Delta t = 0.1$ s.

To analyze motion relative to the ego vehicle, we normalize the trajectory by subtracting the initial position:

$$\tilde{\mathbf{q}}_t = \mathbf{q}_t - \mathbf{q}_0.$$

This removes global-location bias and ensures that the resulting displacement reflects the agent's motion purely from the ego-vehicle's perspective.

■ ***Stage 2: Deriving Speed and Path Intent Labels.*** Given the 1-second relative trajectory $\{\tilde{\mathbf{q}}_t\}$, we infer the agent's semantic motion intent along two axes: *speed intent* and *path intent*.

1) Speed Intent. Velocity is estimated using finite differences, following [15],

$$v_t = \frac{\|\tilde{\mathbf{q}}_{t+1} - \tilde{\mathbf{q}}_t\|}{\Delta t},$$

and intent is determined by thresholding the velocity change across the trajectory:

- `STOP`: all velocities remain below a small threshold;
- `ACCELERATE`: final velocity significantly exceeds the initial velocity;
- `DECELERATE`: final velocity drops noticeably below the initial velocity;
- `KEEP`: velocity remains approximately constant.

This rule-based design follows conventions in autonomous driving motion-forecasting benchmarks.

2) Path Intent. Directional motion is determined from the final displacement:

$$\Delta x = x_{10}^{\text{future}}, \qquad \Delta y = y_{10}^{\text{future}}.$$

The path intent is assigned by comparing lateral drift against a fixed threshold while ensuring that longitudinal displacement indicates stable forward motion:

- `LEFT`: Δx exceeds a positive threshold, indicating object moving toward the ego-vehicle's left side;
- `RIGHT`: Δx is below a negative threshold, indicating a rightward drift;
- `STRAIGHT`: longitudinal displacement dominates, i.e., $|\Delta x|$ is small relative to $|\Delta y|$;
- `UNKNOWN`: displacement is too small or inconsistent to reliably infer direction.

■ ***Stage 3: Building Prediction QA Samples.*** Each tracked agent contributes one prediction QA instance. Given the RGB frame and event stream with the agent highlighted by a bounding box, the model must output the two-token intent classification

`<SPEED>, <PATH>`

(e.g., "`DECELERATE, LEFT`" or "`STOP, UNKNOWN`"). This yields a unified training format for multimodal intent prediction from RGB–event inputs. Once the QA pairs are constructed, we apply the prompting template shown in Fig. 11 to perform RGB-event inference, producing the final prediction results that are used for evaluation.

The prediction annotation pipeline transforms 3D-detected agent trajectories into semantically meaningful short-horizon motion intents and pairs them with image–event evidence through structured QA formatting. By grounding motion labels in ego-centric geometry and enforcing standardized multiple-choice output, the pipeline provides a clean and scalable supervision scheme that unifies physical trajectory prediction with multimodal language-model training.

A.3.4. Planning Annotation

The planning annotation process generates two types of supervision: (1) high-level driving intent (*speed* and *path* decisions), and (2) low-level ego-trajectory forecasting. The pipeline consists of four main stages, combining ego-pose extraction, ego-frame trajectory normalization, rule-based intent derivation, and Qwen-style QA construction.

■ ***Stage 1: Ego-Frame Trajectory Construction.*** For each sequence, we first load the vehicle's global poses from the ground-truth pose logs, where each record contains a timestamp, position $(t,\, x,\, y,\, z)$ and orientation represented as a

You are an intelligent assistant for autonomous driving visual-language understanding.

Given the following driving scene description: {caption}
Your task is to generate 6 diverse and informative multiple-choice QA pairs that evaluate scene understanding for autonomous driving models.
Each QA pair must belong to one of the following six fixed categories:
1. Scene type – classify the environment
2. Visibility – describe lighting or clarity
3. Traffic flow – assess traffic density
4. Weather – describe weather conditions
5. Traffic light – identify the visible signal state
6. Road condition – describe surface state

For each QA pair, follow these rules strictly:
- Write a natural, self-contained, and diverse question that can be answered directly from the description.
- The question wording must vary across samples — do not simply copy or paraphrase the examples.
- Create four short, distinct, and realistic answer choices, each answer should be less than three words (A–D).
- The answer options must also be diverse — avoid repeating the same order or phrasing across questions.
- Exactly one option must be correct and clearly supported by the caption.
- The other three must be plausible but clearly incorrect distractors that make sense in driving context.
- Randomize the correct answer position (not always 'A').
- Add a short, clear answer_sentence that justifies the correct answer naturally.

Output strictly as a valid JSON array of six QA objects — one per category, matching the input order. Do not include any commentary, explanations, or text outside the JSON.
The structure must look exactly like this example:

```
[
    {
    category: Scene type,
    question: What environment does this scene represent?,
    answer_sentence: It depicts an urban area with multiple lanes and surrounding
    buildings.,
    answer_choices: {{A: Urban, B: Rural, C: Suburban, D: Parking area}},
    correct_choice: A
    ... (total 6 entries)
    }
]
```

Important formatting requirements:
- Return only the JSON array, with exactly six QA entries.
- Each entry must use one of the six categories once.
- Ensure all JSON syntax (quotes, commas, braces) is valid.

Figure 10. Prompt used to generate QA pairs from a scene caption that encodes six essential attributes of the driving scene.

```
Suppose you are driving. In the following driving-
scene image and the corresponding event
representation captured from the vehicle's front
camera, the object highlighted by the bounding box
in the image denotes a detected dynamic agent.

Based on its position and motion cues, predict its
driving behavior over the next 1 second from the
ego-vehicle's perspective.

SPEED: choose one from {speed_choices_str}:

- KEEP means maintaining the current speed,
- ACCELERATE means speeding up,
- DECELERATE means slowing down,
- STOP means the object is stationary.

PATH: choose one from {path_choices_str} from the
ego-vehicle's viewpoint:

- LEFT means moving toward the left side of the
  ego vehicle,
- RIGHT means moving toward the right side of the
  ego vehicle,
- STRAIGHT means moving forward along the ego
  vehicle's direction,
- UNKNOWN means the motion direction cannot be
  clearly determined.

Respond in a single line, e.g., B ACCELERATE, C
STRAIGHT. Do not add punctuation or explanation.
```

Figure 11. Prompt used to assess the model's **Prediction** capability in driving scenes.

quaternion. Given the image timestamp, the nearest pose is identified, and a temporal window of the past 3 seconds and future 5 seconds is collected around this index. Let the global trajectory be:

$$\mathcal{P} = \{\mathbf{p}_t = (x_t, y_t, z_t)\}_{t=-T_{\text{past}}}^{t=T_{\text{future}}}.$$

To remove global-layout bias and express all motion relative to the current vehicle pose, we convert every point into the ego-vehicle coordinate system using the rotation and translation of the current pose:

$$\tilde{\mathbf{p}}_t = (\mathbf{p}_t - \mathbf{p}_0)\,\mathbf{R}_0^{\top},$$

where $\mathbf{R}_0$ is the rotation matrix of the current pose. The past and future ego trajectories are then uniformly sampled at 2 Hz, producing

$$\text{Past: } \{\tilde{\mathbf{p}}_i^{\text{past}}\}_{i=1}^{6}, \qquad \text{Future: } \{\tilde{\mathbf{p}}_j^{\text{future}}\}_{j=1}^{10}.$$

■ ***Stage 2: Deriving High-Level Intent Labels.*** Using the future ego-trajectory, we classify the semantic driving intention along two axes: *speed intent* and *path intent*.

1) Speed Intent. Velocity is estimated via finite differences,

$$v_t = \frac{\|\tilde{p}_{t+1} - \tilde{p}_t\|}{\Delta t},$$

```
You are the autonomous driving system controlling
this vehicle.

In the following driving-scene image and the
corresponding event representation captured from
the vehicle's front camera. The current vehicle
speed is {ego_cur_vel:.2f} m/s.

Based on the observed driving scene and the recent
motion history, predict the future ego-vehicle
waypoints for the next 10 timestamps (sampled every
0.5 seconds, covering the next 5.0 seconds).

Provided below is the past ego-vehicle trajectory
(sampled every 0.5 seconds) over the last 3.0
seconds in meters:
Coordinate definition:
  • x – lateral displacement (rightwards is
positive)
  • y – longitudinal displacement (forward is
positive)
Past waypoints [x, y]: {his_trajs_str}

Output format requirement:
  • Predict exactly 10 future [x, y] pairs.
  • Each pair should contain two floating-point
values with 4 decimal precision.
  • Separate pairs by commas and enclose each in
square brackets.
  • Do NOT include explanations, descriptions, or
extra symbols.

Example output format:
[0.0012, 0.0048], [0.0025, 0.0060], …, [0.0037,
0.0074], [0.0048, 0.0089]
```

Figure 12. Prompt used to assess the model's ego waypoint **Planning** capability in driving scenes.

and intent is determined by thresholding velocity change:

- `STOP`: all velocities below a small threshold;
- `ACCELERATE`: final velocity exceeds initial velocity significantly;
- `DECELERATE`: the reverse case;
- `KEEP`: speed remains approximately constant.

2) Path Intent. Lateral displacement in the ego frame determines the future path:

$$\Delta x = x_{10}^{\text{future}}, \qquad \Delta y = y_{10}^{\text{future}}.$$

Path direction is determined by comparing the magnitude of lateral drift against a fixed threshold, while ensuring the longitudinal component provides sufficient forward motion stability. The labeling rule is:

- `LEFT_TURN`: Δx exceeds a positive lateral threshold, indicating a clear drift toward the ego-vehicle's left side;
- `RIGHT_TURN`: Δx falls below a negative threshold, indicating motion toward the right side;
- `STRAIGHT`: the longitudinal component dominates, i.e., $|\Delta x|$ is small relative to $|\Delta y|$;

```
You are the autonomous driving system controlling this vehicle.

The following driving-scene image and its corresponding event
representation were captured from the vehicle's front camera.
The current vehicle speed is {ego_cur_vel:.2f} m/s.

Based on the visible scene context, road geometry and markings,
dynamic agents (vehicles, pedestrians, cyclists), and
potential traffic controls (lights or signs), determine the
ego vehicle's near-term driving intent.

You must output EXACTLY two tokens, separated by a comma and a
space, with no extra words:

- SPEED plan from {KEEP, ACCELERATE, DECELERATE, STOP}
- PATH plan from {STRAIGHT, RIGHT_TURN, LEFT_TURN, UNKNOWN}

Decision principles:

- SPEED:
Consider the current speed, distance to the lead vehicle, red
lights or stop signs, and any immediate hazards.
- KEEP: maintain current speed
- ACCELERATE: increase speed
- DECELERATE: reduce speed
- STOP: come to a halt

- PATH:
Infer from lane geometry, turn arrows, road curvature, and
surrounding traffic behavior.
- LEFT_TURN: trajectory trends toward the left
- RIGHT_TURN: trajectory trends toward the right
- STRAIGHT: continue forward along the ego vehicle's current
direction
- UNKNOWN: direction cannot be reliably determined

If the visual evidence does not clearly support LEFT_TURN,
RIGHT_TURN, or STRAIGHT, output UNKNOWN.

Return ONLY the two tokens, for example:
KEEP, STRAIGHT; DECELERATE, RIGHT_TURN; STOP, UNKNOWN
```

Figure 13. Prompt used to assess the model's ego path and speed **Planning** capability in driving scenes.

- UNKNOWN: motion is too small or inconsistent to reveal a reliable direction.

■ ***Stage 3: Building Planning QA Samples.*** Two complementary QA tasks are generated for each frame:

- *High-level intent planning*: The model receives the RGB image, event representation, and current speed, output the two-token driving decision

  <SPEED>, <PATH>

  (e.g., "DECELERATE, RIGHT_TURN"). The ground truth comes directly from the rule-based labels above.
- *Trajectory forecasting*: The model is given the past 3 s of ego-motion in ego coordinates and must predict the next 10 waypoints at 0.5 s intervals. The ground-truth future trajectory is formatted as a sequence of

  $$[x_j,\, y_j] \quad (j = 1, \ldots, 10),$$

  with four-decimal precision.

■ ***Stage 4: Conversion into Qwen Training Format.*** Each frame produces two Qwen-style conversation samples. The human message describes the driving context, the required output format, and either *(1) the intent-choice rules* or *(2) the past-waypoint history*. The assistant message contains only the strict ground-truth output, without explanations. All samples reference the corresponding RGB image and its aligned event stream, producing a unified, multimodal planning dataset suitable for both training and evaluation. The prompting template for high-level intent planning inference is shown in Fig. 13, and for trajectory forecasting is shown in Fig. 12.

B. Complete Experimental Details

B.1. Baselines

To comprehensively assess the contributions of event sensing across the driving stack, we compare our model with a broad set of existing vision–language systems, covering both frame-based and event-based paradigms. These baselines allow us to isolate the performance differences arising from sensing modality, temporal fidelity, and instruction tuning strategy under our event–frame benchmark.

B.1.1. Frame-based Methods

For frame-based evaluation, we select several fully open-source VLMs that represent the current frontier in multimodal reasoning. These models differ substantially in visual encoder capacity, instruction-tuning depth, and the ability to handle multiple images or videos, providing a diverse reference for assessing robustness in event-heavy driving scenes.

- **LLaVA-v1.6-Mistral-7B** [50] is a multimodal instruction-following model that aligns a Mistral-7B [33] backbone with a large collection of curated image–text and GPT-generated interaction data. The model integrates a lightweight visual encoder with an autoregressive language model, enabling competitive zero-shot reasoning and general-purpose VQA capabilities. Although trained on diverse imagery and conversational tasks, its perception is primarily frame-centric, offering a useful reference for evaluating robustness in event-heavy or motion-degraded driving scenes.
- **LLaVA-OneVision-1.5** [1] is an open multimodal model family trained on large-scale, high-quality image–text corpora with native-resolution images, improving recognition of fine-grained visual details. Its training framework provides efficient scaling through Megatron-LM [61] and long-sequence optimization, enabling strong zero-shot performance across diverse multimodal benchmarks. As a frame-centric LMM with broad instruction-tuning coverage, it offers a representative baseline for assessing general-purpose visual reasoning under driving scenarios.
- **InternVL 2.5** [12] is a multimodal model family built upon a ViT–MLP–LLM architecture, pairing an incrementally trained InternViT [13, 14, 21] encoder with

Table 7. Per-category comparison in **Perception** task on ***EventDrive*** **DSEC** subset. We report per-category results for **Scene Type**, **Visibility**, **Traffic Flow**, **Weather**, **Traffic Light**, and **Road Condition**, respectively.

Method	Scene Type	Visibility	Traffic Flow	Weather	Traffic Light	Road Condition
Event-based Models						
EventGPT-7B [52]	55.12	52.15	63.26	31.89	47.54	58.42
EventDrive-VLM	72.83	61.25	74.46	43.12	59.74	63.25
Frame-based Models						
LLaVA-v1.6-Mistral-7B-hf [50]	67.63	56.50	70.81	36.30	52.11	68.57
LLaVA-OneVision-1.5-8B [1]	90.46	83.35	91.96	38.45	84.57	96.07
InternVL2.5-8B [12]	87.09	87.56	92.98	35.17	86.81	80.92
InternVL3-8B [75]	90.18	83.63	84.10	31.71	88.59	93.45
Qwen2.5-VL-7B-Instruct [3]	86.62	80.17	84.57	40.13	86.44	88.31
Qwen2.5-VL-7B-Instruct* [3]	93.52	86.73	87.21	36.88	89.87	94.91
Event + Frame Models						
EventDrive-VLM	96.12	89.05	89.93	46.95	94.14	96.42

modern language backbones such as InternLM 2.5 [4] and Qwen2.5 [58]. The model adopts dynamic high-resolution tiling and pixel-unshuffle token reduction, enabling efficient processing of single images, multi-image inputs, and videos. Its training pipeline combines cross-modal warm-up, optional vision-encoder refinement, and large-scale instruction tuning with strict data filtering, yielding strong general-purpose visual and multimodal reasoning. As a frame-based LMM with high-resolution processing, InternVL provides a competitive benchmark for evaluating robustness in event-rich driving scenarios.

- **InternVL 3** [75] extends the ViT–MLP–LLM architecture of InternVL 2.5 with native multimodal pre-training, allowing the vision and language components to learn jointly from interleaved image–text and video–text corpora. It incorporates an incrementally trained InternViT [13, 14, 21] encoder and introduces variable visual position encoding for improved long-context reasoning across multi-image and video inputs. Enhanced supervised fine-tuning and mixed preference optimization further strengthen its multimodal understanding and CoT performance. As a high-capacity frame-based baseline, InternVL3 offers a strong reference for evaluating perception and reasoning under driving scenarios.
- **Qwen2.5-VL** [3] is an upgraded vision–language model that enhances its ViT encoder and temporal modeling to support dynamic-resolution images and variable-rate video inputs. The model demonstrates strong capabilities in structured visual understanding, including text-rich imagery, charts, and layout reasoning, and supports fine-grained localization through bounding boxes and keypoints. Its instruction-tuned design further enables tool-driven visual reasoning and long-video event analysis. As a frame-centric VLM, Qwen2.5-VL provides a strong baseline for assessing semantic and spatial reasoning in driving scenarios.

All frame-based models are evaluated in a zero-shot manner to measure out-of-the-box generalization under low light, motion blur, and rapid agent dynamics. In addition, to explicitly quantify the gain brought by event signals, we fine-tune a Qwen2.5-VL-7B model using the same instruction-tuning protocol as our method, serving as a frame-only counterpart that shares the backbone of our ***EventDrive***-VLM.

B.1.2. Event-based Methods

Event-based multimodal models remain scarce due to the absence of publicly released training frameworks. Systems such as EventVL [46] provide neither checkpoints nor code, leaving EventGPT [52] as the only available event-driven LLM for direct comparison.

EventGPT is designed to extend language-model reasoning to asynchronous event streams through a three-stage training paradigm involving image–text warm-up, large-scale synthetic event–text pre-training, and instruction tuning on the Event-Chat dataset. Its data construction relies heavily on synthetic corpora such as N-ImageNet-Chat and N-ImageNet-Instruction, with only a smaller portion drawn from real event recordings. By aligning sparse spatio-temporal event representations with language, EventGPT demonstrates improved scene summarization and reasoning under low-light and high-motion conditions where RGB-based models deteriorate.

Because EventGPT's training pipeline, evaluation split, and testing code are not publicly released, we can only evaluate the model in a zero-shot manner on the subset of its dataset provided in the official repository, and the resulting numbers should therefore be interpreted with caution. As an

Table 8. Per-category comparison in **Perception** task on ***EventDrive*** **M3ED** subset. We report per-category results for **Scene Type**, **Visibility**, **Traffic Flow**, **Weather**, **Traffic Light**, and **Road Condition**, respectively.

Method	Scene Type	Visibility	Traffic Flow	Weather	Traffic Light	Road Condition
Event-based Models						
EventGPT-7B [52]	70.76	38.66	55.58	57.38	50.01	57.43
EventDrive-VLM	88.54	47.18	67.81	70.01	56.61	72.27
Frame-based Models						
LLaVA-v1.6-Mistral-7B-hf [50]	87.04	41.67	59.90	61.84	32.34	72.67
LLaVA-OneVision-1.5-8B [1]	98.62	77.78	88.09	82.34	74.09	97.20
InternVL2.5-8B [12]	99.59	72.85	86.01	82.45	75.73	83.76
InternVL3-8B [75]	99.29	78.90	78.08	86.15	77.07	98.13
Qwen2.5-VL-7B-Instruct [3]	98.13	53.47	81.18	64.13	79.57	85.14
Qwen2.5-VL-7B-Instruct* [3]	98.02	68.86	89.36	70.59	87.59	93.72
Event + Frame Models						
EventDrive-VLM	98.28	72.45	91.42	72.22	89.61	95.88

event-only model optimized for temporal cues rather than appearance semantics, its performance serves as a qualitative reference rather than a fully standardized comparison within our benchmark.

B.1.3. Event-Frame Fusion Methods

For event–frame fusion approaches, existing multimodal event LLMs such as LLaFEA [72] have not released code or checkpoints, preventing controlled reproduction or fine-tuning. Thus, our ***EventDrive***-VLM serves as the primary event–frame fusion baseline, enabling a structured investigation of how asynchronous event cues complement RGB perception across perception, understanding, prediction, and planning.

B.2. Evaluation Protocol

We adopt a unified evaluation protocol across the four task families in ***EventDrive***. Below, we provide the complete metric definitions and computation procedures.

B.2.1. Perception Evaluation

Perception evaluates scene-level attributes using multiple-choice questions. Each answer consists of two components: a choice identifier $c \in \{A, B, C, D\}$ and a textual label t describing the attribute class (e.g., "Urban", "Tunnel", "Night"). Given a ground-truth answer $y = (c, t)$ and a model prediction $\hat{y} = (\hat{c}, \hat{t})$, we first normalize both components by removing prefixes such as "Answer:", stripping punctuation, collapsing repeated whitespace, and parsing the prediction into its letter and text components. Choice identifiers are converted to uppercase, and textual labels are compared case-insensitively.

A prediction is considered correct only when both the choice identifier and textual label match:

$$\mathrm{Correct}(i) = \mathbf{1}\left(c_i = \hat{c}_i \ \wedge \ t_i = \hat{t}_i\right), \tag{5}$$

where $\mathbf{1}(\cdot)$ is the indicator function, and the overall perception accuracy is:

$$\mathrm{Acc}_{\mathrm{perc}} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Correct}(i). \tag{6}$$

This strict matching prevents degenerate strategies such as predicting only the textual label without choosing the correct option.

Each perception question belongs to a predefined scene category (e.g., scene type, traffic flow, weather). For each category k, accuracy is computed as:

$$\mathrm{Acc}(k) = \frac{\sum_{i \in k} \mathrm{Correct}(i)}{|k|}. \tag{7}$$

Category-wise results allow us to analyze robustness under challenging conditions, such as tunnel transitions, nighttime scenes, or strong motion blur.

B.2.2. Understanding Evaluation

Understanding includes two complementary components: *1) object-level multiple-choice question answering*, and *2) spatial grounding via bounding box localization*. Each sample contains either a categorical QA label or a ground-truth bounding box, depending on the task type.

Each answer consists of a choice identifier $c \in \{A, B, C, D\}$ and a textual label t. Given a ground-truth answer $y = (c, t)$ and a normalized prediction $\hat{y} = (\hat{c}, \hat{t})$, we evaluate correctness using a strict letter match and a soft textual match. The QA accuracy is:

Table 9. Per-category comparison in **Perception** task on ***EventDrive*** **PKU** subset. We report per-category results for **Scene Type**, **Visibility**, **Traffic Flow**, **Weather**, **Traffic Light**, and **Road Condition**, respectively.

Method	Scene Type	Visibility	Traffic Flow	Weather	Traffic Light	Road Condition
Event-based Models						
EventGPT-7B [52]	79.67	36.48	49.11	41.32	32.41	74.50
EventDrive-VLM	85.13	48.56	58.64	54.34	38.70	88.96
Frame-based Models						
LLaVA-v1.6-Mistral-7B-hf [50]	76.32	34.95	47.05	39.58	31.05	71.37
LLaVA-OneVision-1.5-8B [1]	84.84	52.53	78.95	62.21	67.47	88.53
InternVL2.5-8B [12]	93.47	49.47	76.11	74.21	72.84	89.05
InternVL3-8B [75]	93.26	45.05	71.89	69.05	74.04	92.95
Qwen2.5-VL-7B-Instruct [3]	80.63	54.53	61.68	41.58	73.47	62.53
Qwen2.5-VL-7B-Instruct* [3]	93.04	56.31	75.00	65.56	84.34	81.03
Event + Frame Models						
EventDrive-VLM	93.73	61.54	77.98	68.16	87.69	84.24

$$\texttt{Acc}_{\text{under}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\big(c_i = \hat{c}_i \ \wedge \ \texttt{SoftMatch}(t_i, \hat{t}_i)\big)\,, \tag{8}$$

where $\texttt{SoftMatch}(\cdot)$ is a relaxed matching function that: (1) removes articles ("a", "an", "the"), (2) strips punctuation, (3) normalizes whitespace and commas, (4) computes a Jaccard similarity over token sets. A pair $(t_i, \hat{t}_i)$ is considered matched when

$$\text{Jaccard}(t_i, \hat{t}_i) > 0.8. \tag{9}$$

This rule mitigates minor formatting variations frequently produced by language models.

For *grounding* samples, the model predicts a bounding box $\hat{b} = (\hat{x}, \hat{y}, \hat{w}, \hat{h})$, while the ground truth is $b = (x, y, w, h)$. We compute Intersection-over-Union (IoU) as:

$$\text{IoU}(b, \hat{b}) = \frac{|b \cap \hat{b}|}{|b \cup \hat{b}|}. \tag{10}$$

Two grounding metrics are reported, including *Top-1 Localization Accuracy*:

$$\texttt{Acc}_{\text{gd}} = \frac{1}{N_{\text{gd}}}\sum_{i=1}^{N_{\text{gd}}} \mathbf{1}\Big(\text{IoU}(b_i, \hat{b}_i) \geq \tau\Big)\,, \quad \tau = 0.6, \tag{11}$$

and *Mean IoU*:

$$\text{mIoU} = \frac{1}{N_{\text{gd}}}\sum_{i=1}^{N_{\text{gd}}} \text{IoU}(b_i, \hat{b}_i). \tag{12}$$

Each understanding sample also belongs to a semantic category (e.g., status, ego-relation, grounding). For a category k, accuracy is computed as:

$$\texttt{Acc}(k) = \frac{\sum_{i \in k} \text{Correct}(i)}{|k|}. \tag{13}$$

This provides fine-grained insight into how event signals affect semantic understanding and spatial grounding across different object-level reasoning tasks.

B.2.3. Prediction Evaluation

Prediction assesses the model's ability to infer short-term motion tendencies of dynamic agents. Each sample contains two categorical labels: a *speed intent* y^{spd} and a *path intent* y^{path}. Given a prediction $(\hat{y}^{\text{spd}}, \hat{y}^{\text{path}})$, we compute intent accuracy for each component as well as their joint correctness.

Since language models may output free-form text such as "A ACCELERATE, C STRAIGHT" or "the vehicle will keep straight," we normalize predictions by: (1) removing punctuation and repeated whitespace, (2) uppercasing tokens, and (3) extracting the canonical intent words using a predefined vocabulary for speed and path. This ensures robustness to minor formatting variations.

A speed prediction is correct when $\hat{y}^{\text{spd}} = y^{\text{spd}}$. Thus,

$$\texttt{Acc}_{\text{spd}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\Big(\hat{y}_i^{\text{spd}} = y_i^{\text{spd}}\Big)\,. \tag{14}$$

Similarly, path correctness is evaluated as:

$$\texttt{Acc}_{\text{path}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\Big(\hat{y}_i^{\text{path}} = y_i^{\text{path}}\Big)\,. \tag{15}$$

Table 10. **Comparison across four driving–reasoning tasks** in the ***EventDrive*** **Hard** benchmark. For **perception**, we report `QA Accuracy` on three subsets. For **understanding**, we report `QA Accuracy`, grounding Top-1 `Accuracy` at IoU 0.6, and **mIoU**. For **prediction** and **planning**, we report `Speed Accuracy` and `Path Accuracy`, and for planning, we also report the mean **L2 Error** (lower is better). All scores are reported in percentage (%), except for L2 Error in meters. Best results are shown in **bold** and 2nd-best results are underlined.

Method	Perception			Understanding			Prediction		Planning		
	Acc@D	Acc@M	Acc@P	Acc	Acc@60	mIoU	Speed	Path	Speed	Path	L2 Error
Event-based Models											
EventGPT-7B [52]	49.09	55.00	52.83	36.96	2.89	3.67	14.26	59.24	24.82	69.64	12.24
EventDrive-VLM	67.30	66.22	66.93	52.17	34.22	29.50	28.56	76.48	39.23	85.53	7.26
Frame-based Models											
LLaVA-v1.6-Mistral-7B-hf [50]	55.64	61.55	50.45	36.69	15.23	25.3	20.10	23.25	10.43	88.59	7.31
LLaVA-OneVision-1.5-8B [1]	80.00	85.36	76.26	57.09	2.65	16.1	15.23	55.84	46.44	41.12	8.31
InternVL2.5-8B [12]	75.95	86.27	**80.15**	58.28	0.18	2.21	27.05	78.52	34.02	51.75	11.08
InternVL3-8B [75]	74.11	84.09	75.61	58.68	0.16	1.51	13.71	78.58	33.78	83.62	8.97
Qwen2.5-VL-7B-Instruct [3]	72.25	77.14	60.76	43.05	27.15	35.12	11.68	70.15	43.23	82.62	7.91
Qwen2.5-VL-7B-Instruct* [3]	77.94	83.52	68.88	50.77	38.15	40.89	20.76	75.46	46.02	86.44	6.02
Event + Frame Models											
EventDrive-VLM	**82.43**	**86.09**	76.94	**59.04**	**45.86**	**49.66**	**29.48**	**79.32**	**52.45**	**88.64**	**5.41**

Table 11. Per-category comparison in the **Understanding** task on the ***EventDrive*** benchmark. We report per-category results for **Object Awareness**, **Appearance**, **Status**, **Relation-to-Viewer**, and **Relation-to-Others**, respectively.

Method	Object Awareness	Appearance	Status	Relation-to-Viewer	Relation-to-Others
Event-based Models					
EventGPT-7B [52]	41.75	46.03	41.55	34.90	29.69
EventDrive-VLM	58.73	64.95	57.98	47.47	41.92
Frame-based Models					
LLaVA-v1.6-Mistral-7B-hf [50]	31.20	64.31	45.34	27.83	33.15
LLaVA-OneVision-1.5-8B [1]	66.76	73.84	65.93	53.95	47.63
InternVL2.5-8B [12]	66.39	84.82	62.23	43.76	42.89
InternVL3-8B [75]	66.00	85.27	67.09	44.96	39.74
Qwen2.5-VL-7B-Instruct [3]	58.99	57.57	55.74	37.15	40.47
Qwen2.5-VL-7B-Instruct* [3]	68.97	67.31	65.17	43.43	47.32
Event + Frame Models					
EventDrive-VLM	77.30	75.39	73.01	48.68	53.02

A prediction is considered fully correct only when both the speed and path intents match:

$$\mathrm{Acc}_{\mathrm{joint}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left(\hat{y}_i^{\mathrm{spd}} = y_i^{\mathrm{spd}} \ \wedge \ \hat{y}_i^{\mathrm{path}} = y_i^{\mathrm{path}}\right). \tag{16}$$

This metric reflects the model's ability to jointly reason about the agent's longitudinal tendency (speed changes) and lateral evolution (path direction).

B.2.4. Planning Evaluation

Planning evaluates both high-level driving intent and low-level future trajectory prediction. Each sample belongs to one of two categories: *1) Planning-HighLevel*, which requires predicting ego speed intent and path intent, or *2) Planning-Trajectory*, which requires forecasting future ego waypoints over a 5-second horizon.

The model predicts a pair $(\hat{y}^{\mathrm{spd}}, \hat{y}^{\mathrm{path}})$. Because language models may produce free-form text (e.g., "the vehicle will accelerate and turn right"), we normalize predictions by stripping punctuation, collapsing whitespace, uppercasing tokens, and extracting intent words from the predefined vocabularies.

Speed and path intent accuracies are computed as:

$$\mathrm{Acc}_{\mathrm{spd}} = \frac{1}{N_{\mathrm{high}}}\sum_{i=1}^{N_{\mathrm{high}}} \mathbf{1}\left(\hat{y}_i^{\mathrm{spd}} = y_i^{\mathrm{spd}}\right), \tag{17}$$

$$\mathrm{Acc}_{\mathrm{path}} = \frac{1}{N_{\mathrm{high}}}\sum_{i=1}^{N_{\mathrm{high}}} \mathbf{1}\left(\hat{y}_i^{\mathrm{path}} = y_i^{\mathrm{path}}\right). \tag{18}$$

Table 12. Per-category comparison in the **Understanding** task on the ***EventDrive*** **hard** subset. We report per-category results for **Object Awareness**, **Appearance**, **Status**, **Relation-to-Viewer**, and **Relation-to-Others**, respectively.

Method	Object Awareness	Appearance	Status	Relation-to-Viewer	Relation-to-Others
Event-based Models					
EventGPT-7B [52]	36.47	44.22	50.61	24.64	28.85
EventDrive-VLM	58.36	66.48	71.44	45.47	19.10
Frame-based Models					
LLaVA-v1.6-Mistral-7B-hf [50]	35.10	43.05	49.01	23.84	32.45
LLaVA-OneVision-1.5-8B [1]	66.23	50.33	66.23	54.97	47.68
InternVL2.5-8B [12]	66.23	64.24	76.82	46.36	37.75
InternVL3-8B [75]	69.44	63.87	78.55	45.06	36.48
Qwen2.5-VL-7B-Instruct [3]	60.93	39.07	57.62	21.85	35.76
Qwen2.5-VL-7B-Instruct* [3]	69.56	47.31	64.42	25.96	46.59
Event + Frame Models					
EventDrive-VLM	78.67	59.24	71.04	31.98	54.29

A prediction is fully correct when both components match:

$$\mathrm{Acc}_{\mathrm{joint}} = \frac{1}{N_{\mathrm{high}}} \sum_{i=1}^{N_{\mathrm{high}}} \mathbf{1}\Big(\hat{y}_i^{\mathrm{spd}} = y_i^{\mathrm{spd}} \ \wedge \ \hat{y}_i^{\mathrm{path}} = y_i^{\mathrm{path}}\Big). \tag{19}$$

Each trajectory sample provides a sequence of $T = 10$ future waypoints at 0.5-second intervals:

$$W_i = \{(x_{i,t}, y_{i,t})\}_{t=1}^{T}, \qquad \hat{W}_i = \{(\hat{x}_{i,t}, \hat{y}_{i,t})\}_{t=1}^{T}.$$

We compute the per-step Euclidean error:

$$e_{i,t} = \left\| \begin{pmatrix} x_{i,t} \\ y_{i,t} \end{pmatrix} - \begin{pmatrix} \hat{x}_{i,t} \\ \hat{y}_{i,t} \end{pmatrix} \right\|_2. \tag{20}$$

The benchmark reports waypoint errors at 1s, 3s, and 5s, corresponding to indices $t = \{2, 6, 10\}$ under a sampling interval $\Delta t = 0.5$ seconds:

$$\mathrm{L2}(t) = e_{i,t}. \tag{21}$$

The mean trajectory error is:

$$\mathrm{L2}_{\mathrm{mean}} = \frac{1}{T} \sum_{t=1}^{T} e_{i,t}. \tag{22}$$

We report the average of these errors over all trajectory samples.

B.3. Additional Implementation Details

We fine-tune ***EventDrive***-VLM on top of Qwen2.5-VL-7B Instruct, augmented with a pretrained MaxViT–RNN [27] event backbone. The event encoder adopts a multi-horizon voxelization scheme with three temporal bin sizes $\mathcal{B} = \{20, 50, 100\}$ to accommodate varying event sparsity and motion dynamics across datasets. Each voxel grid is processed by a stacked MaxViT-RNN encoder, whose output feature maps are aggregated through a lightweight Event Q-Former, and a single linear layer mapping the 512-d event features into the LLM hidden space (2048-d). The RGB vision tower of Qwen2.5-VL is kept frozen throughout all experiments.

Training Strategy. Training is performed using the `transformer` framework with several infrastructure optimizations. We adopt AdamW [37] with a cosine learning schedule and a warmup ratio of 0.03. The learning rate is decoupled across modules: the event encoder and event Q-former use 1×10^{-4}, and the LLM layers use a conservative 2×10^{-7}. All experiments use **bf16**-precision and enable gradient checkpointing to reduce memory cost.

Packed Multimodal Training. To support long multimodal sequences at 4096 tokens, we employ a packed-sequence data pipeline. All modalities (RGB frames, events) are flattened into a unified token stream following Qwen2.5-VL's format. During batching, visual tokens are concatenated across samples and re-located by per-sample `position_ids`, computed using an extended 2D RoPE scheme that also supports events.

Fused FlashAttention. We replace the default attention kernels in Qwen2.5-VL with FlashAttention 2 [19] for both text-only and multimodal tokens. Because event patches greatly increase the local sequence length, we adopt the variable-length attention API and modify the causal mask update in Qwen2.5-VL to support packed multimodal inputs. This reduces attention overhead by over 40% and enables training at the full 4096-token context.

Token-Level Training Masking. To prevent the LLM from predicting pixels or events, all labels at positions corresponding to `<|image_pad|>`,

Table 13. Per-category comparison in **Perception** task on ***EventDrive*** **DSEC hard** subset. We report per-category results for **Scene Type**, **Visibility**, **Traffic Flow**, **Weather**, **Traffic Light**, and **Road Condition**, respectively.

Method	Scene Type	Visibility	Traffic Flow	Weather	Traffic Light	Road Condition
Event-based Models						
EventGPT-7B [52]	55.70	59.47	57.59	16.05	50.03	55.70
EventDrive-VLM	76.36	81.53	78.95	22.00	68.59	76.36
Frame-based Models						
LLaVA-v1.6-Mistral-7B-hf [50]	69.23	86.15	56.92	7.69	50.77	63.08
LLaVA-OneVision-1.5-8B [1]	90.77	96.92	93.85	26.15	81.54	90.77
InternVL2.5-8B [12]	86.15	98.78	95.38	4.62	90.77	80.00
InternVL3-8B [75]	92.31	87.69	84.62	4.62	86.15	89.23
Qwen2.5-VL-7B-Instruct [3]	86.15	98.14	75.38	7.69	86.15	80.00
Qwen2.5-VL-7B-Instruct* [3]	90.94	98.87	83.31	18.29	89.94	86.30
Event + Frame Models						
EventDrive-VLM	95.29	98.56	85.99	28.17	95.29	91.27

`<|event|>`, and `<|video_pad|>` tokens are replaced by `IGNORE_INDEX`. Loss is computed only on natural language tokens:

$$\mathcal{L} = \mathrm{CE}\big(\mathrm{shift}(\mathbf{y}^{\mathrm{logits}}), \mathrm{shift}(\mathbf{y}^{\mathrm{target}})\big) .$$

Inference. At test time, only the first forward pass processes visual tokens. Subsequent decoding steps remove all image/event features to reduce overhead. Inference uses the same 2D-RoPE indexing code as training, ensuring exact consistency.

Overall, these design choices allow ***EventDrive***-VLM to fuse asynchronous event streams with RGB context at scale, while maintaining full compatibility with Qwen2.5-VL's instruction-following behavior.

B.4. Additional Quantitative Results

Perception Per-category Results. Across all three datasets, the per-category comparisons in Tabs. 7 to 9 reveal several consistent patterns. First, pure event-based models (EventGPT-7B) struggle with appearance-heavy categories such as Scene Type, Traffic Light, and Road Condition, where semantic cues depend substantially on spatial textures and color information, confirming that event streams alone are insufficient for fine-grained semantic reasoning.

Frame-based VLMs demonstrate strong improvements in most categories, particularly Scene Type, Traffic Flow, and Road Condition, where global illumination and texture structure are crucial. Large models such as InternVL3-8B and Qwen2.5-VL-7B-Instruct consistently exceed 80% on these categories across datasets. However, they degrade notably on event-favored categories, highlighting their weakness under motion blur and extreme lighting.

Our ***EventDrive***-VLM shows the strongest and most balanced performance across all six categories. The Event + Frame fusion consistently outperforms both single-modality baselines, with substantial gains on Visibility and Weather. At the same time, performance on appearance-dominant categories (e.g., Road Condition, Traffic Light) matches or exceeds the best frame-based VLMs. These results confirm that event–frame fusion enables robust scene recognition across both texture-dependent and motion/illumination-sensitive conditions.

Perception Per-dataset Results. A cross-dataset comparison highlights the complementary strengths of each source domain and further demonstrates the robustness of ***EventDrive***-VLM, as shown in Tabs. 7 to 9, and Tabs. 13 to 15.

DSEC exhibits the highest illumination variation, including tunnels, nighttime sequences, and rapid exposure changes. Frame-based VLMs perform well on bright daytime scenes but drop significantly in low-visibility categories. Event-only models capture high-speed motion but fail on semantic cues requiring RGB textures. Our method closes this gap: it achieves strong and stable performance across all categories, demonstrating resilience to both dark scenes and motion blur.

M3ED, collected with a higher-resolution sensor suite, contains fast egomotion, extreme rotations, and diverse weather/visibility shifts across day/night conditions. Frame-only models excel when the RGB signal is clean, but degrade on nighttime or glare-heavy scenes, as seen in their instability on Visibility. Event-only models handle motion but fail on semantics. In contrast, our method consistently provides top accuracy across all categories, where motion cues and structural cues must be jointly exploited.

PKU-DAVIS-SOD, with much lower spatial resolution and relatively noisy RGB frames, is the most challenging dataset for all frame-based methods. While event-only

Table 14. Per-category comparison in **Perception** task on ***EventDrive*** **M3ED hard** subset. We report per-category results for **Scene Type**, **Visibility**, **Traffic Flow**, **Weather**, **Traffic Light**, and **Road Condition**, respectively.

Method	Scene Type	Visibility	Traffic Flow	Weather	Traffic Light	Road Condition
Event-based Models						
EventGPT-7B [52]	69.75	33.30	61.82	47.66	52.69	64.78
EventDrive-VLM	78.45	43.24	75.08	58.73	65.21	76.61
Frame-based Models						
LLaVA-v1.6-Mistral-7B-hf [50]	90.02	44.82	57.05	47.23	52.77	77.40
LLaVA-OneVision-1.5-8B [1]	98.05	65.63	88.78	74.05	87.76	97.90
InternVL2.5-8B [12]	96.23	62.47	94.78	90.01	89.82	84.36
InternVL3-8B [75]	97.43	62.82	74.12	82.77	88.23	99.14
Qwen2.5-VL-7B-Instruct [3]	95.64	57.21	81.14	52.46	85.97	90.41
Qwen2.5-VL-7B-Instruct* [3]	96.18	69.42	88.39	58.13	92.44	96.57
Event + Frame Models						
EventDrive-VLM	99.52	73.36	91.18	59.69	94.87	97.93

Table 15. Per-category comparison in **Perception** task on ***EventDrive*** **PKU hard** subset. We report per-category results for **Scene Type**, **Visibility**, **Traffic Flow**, **Weather**, **Traffic Light**, and **Road Condition**, respectively.

Method	Scene Type	Visibility	Traffic Flow	Weather	Traffic Light	Road Condition
Event-based Models						
EventGPT-7B [52]	52.62	28.07	67.62	39.77	50.28	78.62
EventDrive-VLM	65.40	42.95	78.70	55.47	66.96	92.10
Frame-based Models						
LLaVA-v1.6-Mistral-7B-hf [50]	53.94	28.18	66.36	40.30	43.33	70.61
LLaVA-OneVision-1.5-8B [1]	80.30	52.42	90.61	63.94	73.03	97.27
InternVL2.5-8B [12]	87.88	52.73	92.73	69.39	86.36	91.82
InternVL3-8B [75]	79.09	55.15	83.64	53.33	86.97	95.45
Qwen2.5-VL-7B-Instruct [3]	51.21	51.52	78.79	25.45	84.24	73.33
Qwen2.5-VL-7B-Instruct* [3]	63.52	61.43	89.47	34.84	94.03	69.98
Event + Frame Models						
EventDrive-VLM	71.56	73.62	93.45	52.44	95.24	75.32

models show some robustness in low-light scenes, their semantic performance remains limited. Our event–frame fusion model again yields the best balanced results, showing that event signals substantially mitigate the resolution limitations of PKU's RGB modality.

Overall, results across the three datasets indicate that neither modality alone is sufficient in diverse real-world driving scenarios. Event–frame fusion, when performed with structured alignment as in our method, offers significantly improved generality and stability across varied environments.

Understanding Per-category Results. The per-category results in Tab. 11 and Tab. 12 reveal clear and consistent trends across the five object-level understanding attributes. Overall, purely event-based models struggle with fine-grained semantic cues such as Appearance, Status, and Relation-to-Others, where texture, color, and spatial detail are critical. EventGPT-7B, for instance, achieves low results on Status and Relation-to-Others, confirming that asynchronous events alone are insufficient for reasoning about detailed object properties or multi-agent relationships.

Frame-based VLMs show significantly stronger performance on appearance-heavy categories. Large models such as InternVL2.5-8B and InternVL3-8B exceed 80% on Appearance, benefiting from high-resolution RGB information. However, their performance decreases on categories requiring temporal awareness and spatial consistency under motion (e.g., Object Awareness, Relation-to-Viewer). This drop is most notable for LLaVA-v1.6-Mistral, which reflects the limitations of frame-only perception under high-speed or blurred scenarios typical in DSEC.

Our ***EventDrive***-VLM achieves the strongest and most balanced results across all five categories. The fusion of high-frequency event cues and rich RGB semantics delivers substantial gains over both event-only and frame-only baselines. Notably, on Object Awareness, the model outperforms the best pure-frame model by a large margin. It also provides notable boosts on temporal–geometric categories (Status, Relation-to-Others), where event signals help disambiguate subtle motion cues, occlusion patterns, and relative positioning.

Although frame-based models remain competitive on highly appearance-driven categories, our method matches or surpasses them while simultaneously providing superior robustness on motion-critical attributes. These results highlight the necessity of integrating both temporal and spatial modalities for comprehensive object-level understanding in real-world driving scenes.

Hard split Results. The Hard split in Tab. 10 represents the challenging subsets in ***EventDrive***, characterized by extremely low illumination, aggressive high-speed motion, heavy occlusions, and substantial appearance degradation. While performance differences relative to the standard split vary across tasks and models, the Hard results provide a clearer picture of robustness under adverse conditions.

For **perception**, most models maintain relatively stable accuracy across the Hard subsets, especially on datasets where global scene semantics remain largely intact despite darker illumination. Frame-based VLMs show modest degradation on night-heavy sequences. The fused method achieves the highest scores, suggesting that combining event edges with frame semantics yields more consistent predictions under challenging visibility.

In **understanding**, the performance gap becomes more pronounced. Tasks involving grounding (`Acc@60`) and spatial consistency (**mIoU**) are especially sensitive to blur, glare, and sparse textures, leading to noticeable drops for most frame-based models. Event-driven cues mitigate some of this degradation, as high-temporal-resolution edges remain informative even when RGB details fade. The fused model again provides the most stable results, improving both grounding and mIoU by leveraging complementary spatiotemporal signals.

For **prediction**, Hard scenes introduce irregular target motion and heavy occlusions, which challenge both event- and frame-based models. Here, event cues contribute positively to short-term motion reasoning: event-only and fused models obtain higher `Speed` accuracy than most RGB-based VLMs. Path intent also benefits from the temporal consistency present in event streams, leading the fused model to reach top performance. Although the improvements are not universally large, the trend shows that motion-asynchronous signals help stabilize intent inference.

In **planning**, the difficulty of the Hard split is more evident. Nighttime ego-motion, complex turns, and GPS-denied tunnel segments create additional uncertainty for long-horizon forecasting. Event-driven temporal cues provide a more stable basis for predicting ego-vehicle dynamics, especially for acceleration and turning behavior. The fused model achieves the best results across all metrics, reducing L2 error and improving intent consistency without relying solely on appearance cues.

Across all four tasks, Hard split results highlight several general trends:

- Event-only models tend to be more stable than frame-only models in conditions dominated by low light, motion blur, or rapid dynamics.
- Frame-based models perform strongly on standard splits but may show larger variance in grounding and spatial reasoning under degraded visibility.
- The fused ***EventDrive***-VLM consistently provides the most balanced and robust performance, benefiting from complementary strengths of event and RGB modalities.

The Hard split amplifies the value of fine-grained temporal cues, demonstrating that event streams meaningfully contribute to robustness, especially for understanding and motion-centric tasks. Overall, the Hard evaluation confirms that event signals are not merely auxiliary but play a meaningful role in maintaining reliable perception and decision-making under adverse, real-world conditions.

C. Broader Impact & Limitations

C.1. Broader Impact

EventDrive introduces the first full-stack event–frame multimodal benchmark for driving perception, understanding, prediction, and planning. By unifying asynchronous event streams with RGB images and language supervision, our framework pushes event-based research beyond low-level sensing toward higher-level reasoning, decision making, and explainability.

Event cameras provide microsecond temporal fidelity and high dynamic range, enabling robust perception under motion blur, low light, or extreme illumination, where conventional sensors degrade. EventDrive therefore has the potential to support safer autonomous navigation and more reliable robotic systems in challenging real-world environments. Furthermore, EventDrive demonstrates how language grounding can be systematically incorporated into event-driven pipelines, offering a path toward interactive driving agents capable of interpreting, explaining, and justifying their decisions.

We expect that the proposed dataset and pre-training tasks will stimulate research in temporally aligned multimodal fusion, reasoning under high-frequency signals, and efficient multimodal large models. These insights may extend beyond autonomous driving to domains such as mobile

robotics, AR/VR, and high-speed industrial automation.

C.2. Societal Influence

EventDrive contributes to the broader vision–language community by providing a new resource that emphasizes temporal precision, sensor efficiency, and structured multimodal reasoning. Event-based vision has inherent advantages: low latency, low power consumption, and resilience to high-speed motion, which align with long-term goals of sustainable and dependable AI deployment.

From a societal perspective, improved multimodal understanding can help autonomous systems behave more predictably and transparently, promoting trust in safety-critical applications. EventDrive is curated to avoid identifiable biometric information and focuses on object-level and scene-level semantics rather than personal identity. All annotations are task-oriented and non-sensitive, targeting driving behavior analysis rather than surveillance or personal profiling.

We believe this dataset will support community-driven progress in designing interpretable, robust, and efficient event-based learning systems, while also encouraging responsible use of multimodal sensor data.

C.3. Potential Limitations

Despite its breadth and contributions, ***EventDrive*** has several limitations:

- **Geographic and sensor constraints.** The dataset covers diverse driving sequences but still originates from a limited set of cities and camera configurations. This may introduce distributional bias and reduce transferability to unseen regions, weather patterns, or sensor setups.
- **Automatic annotation dependency.** While we enforce strict validation and consistency checks, many annotations are derived using model-assisted pipelines. These may inherit biases from the underlying VLMs or from rule-based heuristics, particularly in high-level planning intent or motion interpretation.
- **Focus on 2D grounding and ego-centric reasoning.** The current benchmark emphasizes 2D bounding boxes, event–frame fusion, and ego-centric coordinate systems. World-coordinate 3D grounding and multi-agent joint planning remain out of scope, though they are natural extensions.

These limitations highlight opportunities for future expansion, for example, incorporating world-level 3D grounding, extending geographic coverage, or exploring ultra-long horizon reasoning.

C.4. Ethical Considerations

EventDrive is designed with careful attention to ethical use and responsible data handling. All source data are from existing open-sourced datasets. All recordings are captured in public roadway environments where individuals are not identifiable, and no biometric, demographic, or personally sensitive attributes are annotated or inferable. The dataset focuses strictly on object-level and scene-level semantics relevant to autonomous driving, such as vehicles, road layout, and motion cues, rather than human identity or personal behavior patterns.

We explicitly avoid annotating or enabling tasks that could lead to privacy-invasive applications such as facial recognition, pedestrian re-identification, or profiling. Language annotations are task-specific and do not describe individuals beyond generic categories required for driving safety (e.g., "pedestrian," "cyclist"). Furthermore, any language generation that involves LLMs is manually filtered to ensure that no harmful or inappropriate content is introduced into the dataset.

Despite these safeguards, autonomous driving remains a safety-critical domain. Models trained on EventDrive should not be deployed directly in real-world systems without appropriate verification, robustness testing, and compliance with regional safety regulations. The work is intended solely for research purposes, and any downstream use in commercial or operational systems should incorporate additional validation and ethical review.

D. Public Resources Used

In this section, we acknowledge the use of the following public resources during the course of this work.

D.1. Public Datasets Used

We acknowledge the use of the following public datasets during the course of this work:

- M3ED[1] . . . . . CC BY-SA 4.0
- DSEC[2] . . . . . CC BY-SA 4.0 License
- PKU-DAVIS-SOD[3] . . . . . Unknown

D.2. Public Implementations Used

- EventGPT[4] . . . . . Apache License 2.0
- RVT[5] . . . . . MIT License
- Qwen2.5-VL-7B-Instruct[6] . . . . . Apache License 2.0
- InternVL3-8B[7] . . . . . Apache License 2.0
- InternVL2_5-8B[8] . . . . . MIT License

[1] https://m3ed.io.
[2] https://dsec.ifi.uzh.ch.
[3] https://git.openi.org.cn/LiDianze/PKU-DAVIS-SOD
[4] https://github.com/XduSyL/EventGPT.
[5] https://github.com/uzh-rpg/RVT.
[6] https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct.
[7] https://huggingface.co/OpenGVLab/InternVL3-8B.
[8] https://huggingface.co/OpenGVLab/InternVL2_5-8B.

- LLaVA-OneVision-1.5-8B-Instruct[9] Apache License 2.0
- LLaVA-v1.6-mistral-7b-hf[10] ..... Apache License 2.0
- Impromptu-VLA[11] ..... CC-BY-SA-4.0 license
- Pi3DET[12] ..... MIT License
- PyTorch[13] ..... BSD License

[9] https://huggingface.co/lmms-lab/LLaVA-OneVision-1.5-8B-Instruct.
[10] https://huggingface.co/llava-hf/llava-v1.6-mistral-7b-hf.
[11] https://github.com/ahydchh/Impromptu-VLA.
[12] https://huggingface.co/datasets/Pi3DET/data.
[13] https://pytorch.org/.

You are an assistant specialized in visual scene understanding, responsible for generating precise and unambiguous referential descriptions for individual target objects in views, for use by subsequent localization or grounding models.

Input
Image A: Original view (containing all objects, used for referencing target's surrounding relationships and confirming description uniqueness)
Image B: Target mask view, aligned with Image A, with the target enclosed within a green bounding box
Category label: {class_label}

Task
Must describe the **unique target** within the mask region of Image B, ignoring objects outside Image B as subjects. May reference surrounding object positions in Image A, but the described subject must be the masked target.

Validation Steps (Highest Priority)
1. If the target in the mask region is not {class_label}, directly output: "There is no describable object within the specified bounding box."
2. If the mask region contains multiple objects (especially objects of the same class), directly output: "There is no describable object within the specified bounding box.“
3. If the mask region is empty, severely occluded, severely truncated, unclear, too far, or too small, directly output: "There is no describable object within the specified bounding box.“
4. Final uniqueness verification after generation: After generating the description, simulate "finding the target in Image A based on the description": if there exists "objects other than the bounding box target" that completely match the description (including features, position, orientation, surrounding relationships), output: "There is no describable object within the specified bounding box.“

Hard Rules
1. Subject Lock: All descriptions target only the Image B masked object, surrounding objects serve only as references, not as subjects.
2. No Hallucination: Do not speculate about brands, models, text/numbers, speed; colors and features must be clearly visible in Image A/B, omit if uncertain.
3. Occlusion Description: Must describe when target is partially occluded, boundary truncated, or blurred; omit if none.
4. Position vocabulary in view: Use specified terms (may add modifiers for refinement): top-left | top | top-right | left | center | right | bottom-left | bottom-center | bottom-right | left-center | right-center (e.g., "top-right, slightly right, near the right edge of the view").
5. Relative relationship with observer (categorized, select 1-2 most accurate):
- Pedestrians: back to observer | side profile facing observer | front facing observer | cannot clearly determine
- Vehicles: side facing observer | rear facing observer | front facing observer | cannot clearly determine
- If uncertain, write "cannot clearly determine".
6. Motion state (select 1): stationary | moving toward camera | moving away from camera | moving left in frame | moving right in frame | cannot clearly determine; if no clear visual evidence, write "cannot clearly determine".
7. Relationship with surrounding objects: Reference Image A, use left/right/front/back (based on observer's relative coordinate system) to describe up to 3 objects closest to the target. Only describe when surrounding objects have distinctive features (such as unique colors, types) and clear positions.
8. Appearance description standards:
- Vehicles: [vehicle type (if distinguishable)], [color (if visible)], [distinctive feature 1 (if visible)], [distinctive feature 2 (if visible)]. Example: "truck, white, with roof rack". If color is uncertain, omit color description.
- Pedestrians: [wearing [upper garment color (if visible)] top and [lower garment color (if visible)] bottom, [other distinctive features (if visible)]]. Example: "person, wearing black top and blue jeans, with a hat". Omit attributes that are not visible.

Important Reminders
Category priority: If not matching {class_label}, immediately output invalid statement. Uniqueness priority: If mask region contains multiple objects or is unclear, immediately output invalid statement.
Subject lock: Subject must be the masked target, surrounding objects can only serve as references.
Omit uncertain: When unclear due to lighting/occlusion, directly omit.
Unified terminology: Use "view" instead of "image", use position/orientation vocabulary from the rules above.
Only describe very clearly visible features, do not describe uncertain features.
All information must come from visible content in Image A/B, not from "common sense speculation" (e.g., "cars by the roadside are likely family sedans", but cannot determine from the image, so don't write "family").
Avoid unusual and unreasonable vehicle colors, such as green cars.
Do not mention green bounding box in descriptions.

Output Format (Strictly Follow)
1. Appearance: [Describe object type, color, distinctive features according to rule 8; omit attributes when uncertain]
2. Motion state: [Motion state terms from rule 6]
3. Position in view: [Position terms from rule 4 + modifiers (if any) + occlusion/truncation/blur conditions (if any)]
4. Relative relationship with observer: [Orientation terms from rule 5 + 1-2 visual evidence; use "cannot clearly determine" if no evidence]
5. Relationship with surrounding objects: [Content from rule 7, such as "a pedestrian to the left front of the target, a white car to the right", or "a phone booth to the left of the target, target is on a crosswalk", or "target is under a red light"; omit this section if no distinctive surrounding objects]
Or There is no describable object within the specified bounding box.

Figure 14. Prompt used to generate an object caption capturing essential object attributes of the driving scene.

```
You are a vision-language assistant that generates high-quality multiple-choice QA data
for object-level understanding in autonomous driving scenes.

You will be provided with a structured description of a target object, its bounding box,
and information about other objects in the same scene (if any).{object_context}

Generate exactly six QA pairs, each belonging to one of the following fixed categories:

1. Object Awareness – identifying which objects are present in the scene (using the caption and other object classes)
2. Grounding – given the structured description of the object, selecting which of four candidate bounding boxes corresponds to
the object
3. Appearance – describing the visual appearance of the object inside the given bounding box
4. Status – describing the motion or action state of the object inside the bounding box
5. Relation-to-Viewer – describing where the object inside the bounding box is located relative to the camera or viewer
6. Relation-to-Others – describing the spatial relation between the object in the bounding box and other nearby objects

Important task details
- All six QA pairs must be multiple-choice questions (A–D) with exactly one correct answer.
- The correct answer must be directly supported by the description.
- The other three answers should be plausible distractors within the driving context but clearly incorrect.
- The correct answer position should be randomized (not always A).
- Each question must include a short `answer_sentence` explaining why the correct answer is correct.
- For all question types, refer to the target only as "the object inside the bounding box" — never reveal its true class name.
-  Use natural, diverse, and fluent English wording.

Category-specific instructions
1. Object Awareness
- Question should ask what objects are present in the scene (target + others).
- Example: Which objects are visible in this driving scene?
- Answer choices should list combinations of plausible objects.
2. Grounding
- The model will be provided four candidate bounding boxes (A–D); you only write the question. This question tests whether the
model can localize the described object among four bounding box candidates (A–D).
- Your first step: rephrase the caption into one referring expression that describes the target object unqiuely.
- Then, based on that description, write a question that asks: * Which bounding box corresponds to the (your object
description, including appearance, status, relation to viewer, and relation to other objects)?*

In the following four question categories, the image and bounding box are given.
3. Appearance
- Ask about the visual appearance (color, type, texture, size, etc.) of the object inside the box.
- Example: What best describes the appearance of the object inside the box?
- Provide four concise answers, one correct.
4. Status
- Ask about motion or state (moving, parked, stopped, turning, etc.).
- Example: What is the motion state of the object inside the box?
5. Relation-to-Viewer
- Ask about where the object is relative to the viewer (front, behind, left, right, above, below, etc.).
- Example: Where is the object inside the box relative to the camera?
6. Relation-to-Others
- Ask about the spatial relationship between the object in the box and another object (e.g., traffic sign, billboard,
pedestrian).
- Example: What is the spatial relationship between the object inside the box and the traffic sign?

Output format
Return a valid JSON array with six entries, one per category:

[
    {
    category: Appearance,
    question: ...,
    answer_choices: {{A: ..., B: ..., C: ..., D: ...}},
    correct_choice: B,
    answer_sentence: ...
    },
    ... (total six entries)
]

Formatting rules
- Return only the JSON array, nothing else.
- Ensure valid JSON syntax (no trailing commas, proper quotes, braces, etc.).
```

Figure 15. Prompt used to generate QA pairs from an object caption that encodes essential object attributes in the driving scene.

References

[1] Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Chunsheng Wu, et al. LLaVA-OneVision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025.

[2] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[4] Zheng Cai, Maosong Cao, Haojiong Chen, et al. InternLM2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

[5] Hu Cao, Guang Chen, Zhijun Li, Yingbai Hu, and Alois Knoll. NeuroGrasp: multimodal neural network with euler region regression for neuromorphic vision-based grasp pose estimation. *IEEE Transactions on Instrumentation and Measurement*, 71:1–11, 2022.

[6] Hu Cao, Zehua Zhang, Yan Xia, Xinyi Li, Jiahao Xia, Guang Chen, and Alois Knoll. Embracing events and frames with hierarchical feature refinement network for object detection. In *European Conference on Computer Vision*. Springer, 2024.

[7] Bharatesh Chakravarthi, Aayush Atul Verma, Kostas Daniilidis, Cornelia Fermuller, and Yezhou Yang. Recent event camera innovations: A survey. In *European Conference on Computer Vision Workshops*. Springer, 2024.

[8] Kenneth Chaney, Fernando Cladera, Ziyun Wang, Anthony Bisulco, M. Ani Hsieh, Christopher Korpela, Vijay Kumar, Camillo J. Taylor, and Kostas Daniilidis. M3ED: Multi-robot, multi-sensor, multi-environment event dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 4016–4023, 2023.

[9] Guang Chen, Hu Cao, Canbo Ye, Zhenyan Zhang, Xingbo Liu, Xuhui Mo, Zhongnan Qu, Jörg Conradt, Florian Röhrbein, and Alois Knoll. Multi-cue event information fusion for pedestrian detection with neuromorphic vision sensors. *Frontiers in Neurorobotics*, 13:10, 2019.

[10] Nicholas FY Chen. Pseudo-labels for supervised learning on dynamic vision sensor data, applied to object detection under ego-motion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 644–653, 2018.

[11] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023.

[12] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[13] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to GPT-4V? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024.

[14] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024.

[15] Haohan Chi, Huan ang Gao, Ziming Liu, Jianing Liu, Chenyu Liu, Jinwei Li, Kaisen Yang, Yangcheng Yu, Zeda Wang, Wenyi Li, Leichen Wang, Xingtao Hu, Hao Sun, Hang Zhao, and Hao Zhao. Impromptu VLA: Open weights and open data for driving vision-language-action models. *arXiv preprint arXiv:2505.23757*, 2025.

[16] Hoonhee Cho, Hyeonseong Kim, Yujeong Chae, and Kuk-Jin Yoon. Label-free event-based object recognition via joint learning with image reconstruction from events. In *IEEE/CVF International Conference on Computer Vision*, pages 19866–19877, 2023.

[17] Loïc Cordone, Benoît Miramond, and Philippe Thierion. Object detection with spiking neural networks on automotive event data. In *International Joint Conference on Neural Networks*, pages 1–8, 2022.

[18] Javier Cuadrado, Ulysse Rançon, Benoit R. Cottereau, Francisco Barranco, and Timothée Masquelier. Optical flow estimation from event-based cameras and spiking neural networks. *Frontiers in Neuroscience*, 17:1160034, 2023.

[19] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

[20] Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J Davison, Jörg Conradt, Kostas Daniilidis, et al. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180, 2022.

[21] Zhangwei Gao, Zhe Chen, Erfei Cui, Yiming Ren, Weiyun Wang, Jinguo Zhu, Hao Tian, Shenglong Ye, Junjun He, Xizhou Zhu, et al. Mini-InternVL: A flexible-transfer pocket multimodal model with 5% parameters and 90% performance. *arXiv preprint arXiv:2410.16261*, 2024.

[22] Daniel Gehrig and Davide Scaramuzza. Pushing the limits of asynchronous graph-based object detection with event cameras. *arXiv preprint arXiv:2211.12324*, 2022.

[23] Daniel Gehrig and Davide Scaramuzza. Low-latency automotive vision with event cameras. *Nature*, 629(8014):1034–1040, 2024.

[24] Daniel Gehrig, Antonio Loquercio, Konstantinos G. Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *IEEE/CVF International Conference on Computer Vision*, pages 5633–5643, 2019.

[25] Daniel Gehrig, Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. EKLT: asynchronous photometric feature tracking using events and frames. *International Journal of Computer Vision*, 128(3):601–618, 2020.

[26] Daniel Gehrig, Michelle Rüegg, Mathias Gehrig, Javier Hidalgo-Carrió, and Davide Scaramuzza. Combining events and frames using recurrent asynchronous multimodal networks for monocular depth prediction. *IEEE Robotics and Automation Letters*, 6(2):2822–2829, 2021.

[27] Mathias Gehrig and Davide Scaramuzza. Recurrent vision transformers for object detection with event cameras. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13884–13893, 2023.

[28] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. DSEC: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 6(3): 4947–4954, 2021.

[29] Ryuhei Hamaguchi, Yasutaka Furukawa, Masaki Onishi, and Ken Sakurada. Hierarchical neural memory network for low latency event processing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22867–22876, 2023.

[30] Tianshuai Hu, Xiaolu Liu, Song Wang, Yiyao Zhu, Ao Liang, Lingdong Kong, Guoyang Zhao, Zeying Gong, Jun Cen, Zhiyu Huang, Xiaoshuai Hao, Linfeng Li, Hang Song, Xiangtai Li, Jun Ma, Shaojie Shen, Jianke Zhu, Dacheng Tao, Ziwei Liu, and Junwei Liang. Vision-language-action models for autonomous driving: Past, present, and future. *arXiv preprint arXiv:2512.16760*, 2025.

[31] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023.

[32] Massimiliano Iacono, Stefan Weber, Arren Glover, and Chiara Bartolozzi. Towards event-driven object detection with off-the-shelf deep learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1–9, 2018.

[33] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

[34] Zhuangyi Jiang, Pengfei Xia, Kai Huang, Walter Stechele, Guang Chen, Zhenshan Bing, and Alois Knoll. Mixed frame-/event-driven fast pedestrian detection. In *IEEE International Conference on Robotics and Automation*, pages 8332–8338, 2019.

[35] Linglin Jing, Yiming Ding, Yunpeng Gao, Zhigang Wang, Xu Yan, Dong Wang, Gerald Schaefer, Hui Fang, Bin Zhao, and Xuelong Li. HPL-ESS: Hybrid pseudo-labeling for unsupervised event-based semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23128–23137, 2024.

[36] Junho Kim, Jaehyeok Bae, Gangin Park, Dongsu Zhang, and Young Min Kim. N-ImageNet: Towards robust, fine-grained object recognition with event cameras. In *IEEE/CVF International Conference on Computer Vision*, pages 2146–2156, 2021.

[37] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference for Learning Representations*, 2015.

[38] Lingdong Kong, Youquan Liu, Lai Xing Ng, Benoit R. Cottereau, and Wei Tsang Ooi. OpenESS: Event-based semantic scene understanding with open vocabularies. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15686–15698, 2024.

[39] Lingdong Kong, Dongyue Lu, Ao Liang, Rong Li, Yuhao Dong, Tianshuai Hu, Lai Xing Ng, Wei Tsang Ooi, and Benoit R Cottereau. Talk2Event: Grounded understanding of dynamic scenes from event cameras. In *Advances in Neural Information Processing Systems*, 2025.

[40] Lingdong Kong, Dongyue Lu, Xiang Xu, Lai Xing Ng, Wei Tsang Ooi, and Benoit R. Cottereau. EventFly: Event camera perception from ground to the sky. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1472–1484, 2025.

[41] Lingdong Kong, Xiang Xu, Jiawei Ren, Wenwei Zhang, Liang Pan, Kai Chen, Wei Tsang Ooi, and Ziwei Liu. Multimodal data-efficient 3D scene understanding for autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3748–3765, 2025.

[42] Dianze Li, Yonghong Tian, and Jianing Li. SODFormer: Streaming object detection with transformer using events and frames. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):14020–14037, 2023.

[43] Jianing Li, Siwei Dong, Zhaofei Yu, Yonghong Tian, and Tiejun Huang. Event-based vision enhanced: A joint detection framework in autonomous driving. In *IEEE International Conference on Multimedia and Expo*, pages 1396–1401, 2019.

[44] Jianing Li, Jia Li, Lin Zhu, Xijie Xiang, Tiejun Huang, and Yonghong Tian. Asynchronous spatio-temporal memory network for continuous event-based object detection. *IEEE Transactions on Image Processing*, 31:2975–2987, 2022.

[45] 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*, pages 19730–19742. PMLR, 2023.

[46] Pengteng Li, Yunfan Lu, Pinghao Song, Wuyang Li, Huizai Yao, and Hui Xiong. EventVL: Understand event streams via multimodal large language model. *arXiv preprint arXiv:2501.13707*, 2025.

[47] Rong Li, Shijie Li, Lingdong Kong, Xulei Yang, and Junwei Liang. SeeGround: See and ground for zero-shot open-

vocabulary 3D visual grounding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3707–3717, 2025.

[48] Rong Li et al. 3EED: Ground everything everywhere in 3D. *arXiv preprint arXiv:2511.01755*, 2025.

[49] Ao Liang, Lingdong Kong, Dongyue Lu, Youquan Liu, Jian Fang, Huaici Zhao, and Wei Tsang Ooi. Perspective-invariant 3D object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 27725–27738, 2025.

[50] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023.

[51] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, pages 34892–34916, 2023.

[52] Shaoyu Liu, Jianing Li, Guanghui Zhao, Yunjian Zhang, Xin Meng, Fei Richard Yu, Xiangyang Ji, and Ming Li. Event-GPT: Event stream understanding with multimodal large language models. *arXiv preprint arXiv:2412.00832*, 2024.

[53] Dongyue Lu, Lingdong Kong, Gim Hee Lee, Camille Simon Chane, and Wei Tsang Ooi. FlexEvent: Towards flexible event-frame object detection at varying operational frequencies. In *Advances in Neural Information Processing Systems*, 2025.

[54] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. DeepSeek-VL: Towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024.

[55] Nico Messikommer, Daniel Gehrig, Antonio Loquercio, and Davide Scaramuzza. Event-based asynchronous sparse convolutional networks. In *European Conference on Computer Vision*, pages 415–431. Springer, 2020.

[56] Yansong Peng, Hebei Li, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. Scene adaptive sparse transformer for event-based object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16794–16804, 2024.

[57] Etienne Perot, Pierre De Tournemire, Davide Nitti, Jonathan Masci, and Amos Sironi. Learning to detect objects with a 1 megapixel event camera. *Advances in Neural Information Processing Systems*, 33:16639–16652, 2020.

[58] Qwen, An Yang, Baosong Yang, Beichen Zhang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[59] Simon Schaefer, Daniel Gehrig, and Davide Scaramuzza. AEGNN: Asynchronous event-based graph neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12371–12381, 2022.

[60] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

[61] Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-LM: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

[62] Lea Steffen, Daniel Reichard, Jakob Weinland, Jacques Kaiser, Arne Roennau, and Rüdiger Dillmann. Neuromorphic stereo vision: A survey of bio-inspired sensors and algorithms. *Frontiers in Neuroscience*, 13:28, 2019.

[63] Daobo Sun and Haibo Ji. Event-based object detection using graph neural networks. In *IEEE Conference on Data Driven Control and Learning Systems*, pages 1895–1900, 2023.

[64] Lei Sun, Christos Sakaridis, Jingyun Liang, Qi Jiang, Kailun Yang, Peng Sun, Yaozu Ye, Kaiwei Wang, and Luc Van Gool. Event-based fusion for motion deblurring with cross-modal attention. In *European Conference on Computer Vision*, pages 412–428. Springer, 2022.

[65] Zhaoning Sun, Nico Messikommer, Daniel Gehrig, and Davide Scaramuzza. ESS: Learning event-based semantic segmentation from still images. In *European Conference on Computer Vision*, pages 341–357. Springer, 2022.

[66] Abhishek Tomy, Anshul Paigwar, Khushdeep S Mann, Alessandro Renzaglia, and Christian Laugier. Fusing event-based and RGB camera for robust object detection in adverse conditions. In *IEEE International Conference on Robotics and Automation*, pages 933–939, 2022.

[67] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[68] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. PARA-Drive: Parallelized architecture for real-time autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15449–15458, 2024.

[69] Ziyi Wu, Xudong Liu, and Igor Gilitschenski. EventCLIP: Adapting CLIP for event-based object recognition. *arXiv preprint arXiv:2306.06354*, 2023.

[70] Ziyi Wu, Mathias Gehrig, Qing Lyu, Xudong Liu, and Igor Gilitschenski. LEOD: Label-efficient object detection for event cameras. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16933–16943, 2024.

[71] Jiqing Zhang, Bo Dong, Haiwei Zhang, Jianchuan Ding, Felix Heide, Baocai Yin, and Xin Yang. Spiking transformers for event-based single object tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8801–8810, 2022.

[72] Hanyu Zhou and Gim Hee Lee. LLaFEA: Frame-event complementary fusion for fine-grained spatiotemporal understanding in LMMs. In *IEEE/CVF International Conference on Computer Vision*, pages 22294–22304, 2025.

[73] Zhuyun Zhou, Zongwei Wu, Rémi Boutteau, Fan Yang, Cédric Demonceaux, and Dominique Ginhac. RGB-event fusion for moving object detection in autonomous driving. In *IEEE International Conference on Robotics and Automation*, pages 7808–7815, 2023.

[74] Alex Zihao Zhu and Liangzhe Yuan. EV-FlowNet: Self-supervised optical flow estimation for event-based cameras. In *Robotics: Science and Systems*, 2018.

[75] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.