

End-to-End Traffic Accident Anticipation and Localization for Autonomous Driving

Haicheng Liao^{a,b,1}, Chengyue Wang^{a,c,1}, Bin Rao^{a,c}, Jiaxun Zhang^{a,c}, Haoyu Sun^d,
Zhenning Li^{a,b,c,*}, Chengzhong Xu^{a,b}, Hai Yang^e and Keqiang Li^{f,*}

^aState Key Laboratory of Internet of Things for Smart City, University of Macau, Macau SAR, China

^bDepartment of Computer Science, University of Macau, Macau SAR, China

^cDepartment of Civil and Environmental Engineering, University of Macau, Macau SAR, China

^dDepartment of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China

^eDepartment of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

^fSchool of Vehicle and Mobility, Tsinghua University, Beijing, China

ARTICLE INFO

Keywords:

Traffic Accident Anticipation
Accident Localization
Autonomous Driving
Spatio-Temporal Analysis

ABSTRACT

Traffic accidents impose substantial human, societal, and economic costs worldwide, creating an urgent need for systems that can identify hazardous events early enough for preventive action. However, most existing methods end with estimating whether an accident will occur and provide limited support for locating the threat, informing passengers, or documenting the event. To address this limitation, we present CRASH++, a unified framework that extends accident anticipation into a coherent safety pipeline encompassing risk perception, real-time interaction with passengers, and accident documentation. From dashcam video, CRASH++ predicts accident risk over time and identifies the road users most likely to be involved. Its perception backbone combines road-user interactions, human-attention cues, global scene context, and temporal reasoning, while frequency-domain modeling helps reveal subtle risk cues in complex traffic environments. These structured predictions are translated into concise passenger alerts and retrieval-grounded reports that summarize the event, involved road users, driving conditions, and likely contributing factors. Evaluations on three accident datasets and four driver-attention benchmarks show that CRASH++ achieves state-of-the-art accident anticipation and competitive driver-attention prediction. It also remains robust when 10%–50% of observations are missing and when training data are limited. In a user study involving 43 participants, its alerts received higher ratings for clarity, trust, comfort, usefulness, and situation awareness. Overall, CRASH++ recasts accident anticipation as an end-to-end safety framework, providing a new pathway toward safer and more trustworthy autonomous driving systems.

1. Introduction

Road traffic crashes remain a major global public health and engineering challenge. Despite a 21% decline in the global road traffic death rate between 2011 and 2025, crashes claimed approximately 1.16 million lives in 2025 and remained the leading cause of death among people aged 5 to 29 years [1]. The burden remains particularly severe for vulnerable road users and in settings with elevated risk, such as complex urban intersections and rural roads [2, 3]. Meeting the target of halving road deaths and serious injuries by 2030, and ultimately realizing the “Vision Zero” ambition by 2050, requires a shift from responding to crashes toward recognizing and addressing danger before impact [4, 5, 6]. Autonomous vehicles (AVs) provide an important opportunity for this transition by combining continuous environmental perception with intelligent decision-making [7, 8]. Within this safety loop, accident anticipation aims to identify emerging hazards from onboard observations and provide sufficient lead time for preventive action [9, 10].

Turning this potential into a reliable real-world capability remains challenging. Crashes are rare and diverse, and their early signs are often weak and meaningful only within the surrounding traffic context. A slight lane deviation or a pedestrian hesitating near a crosswalk may become dangerous only when considered together with nearby road users and the wider environment. Recent deep learning methods have improved anticipation by modeling the movements and interactions of detected road users [9, 10]. However, many approaches still represent traffic scenes mainly through detected objects and bounding boxes. Such representations indicate what is present and where it is located, but cannot

*Corresponding author; ¹Equally Contributed

✉ zhenningli@um.edu.mo (Z. Li); likq@tsinghua.edu.cn (K. Li)

fully explain why a particular behavior may be risky. Road geometry, traffic controls, visibility, and surrounding infrastructure can substantially change the meaning of the same movement. Real driving videos are also frequently affected by occlusion, adverse weather, sensor noise, communication interruptions, and missing frames [11, 12]. Models developed from clean and complete observations may therefore lose reliability when important evidence is unavailable [13]. A practical anticipation system must consequently integrate road user interactions, visually salient regions, global scene context, and temporal information, while remaining robust to incomplete observations [14].

Another gap in current systems lies in passenger experience and human-AV interaction [15, 16]. Most existing studies focus on prediction accuracy, while paying limited attention to how passengers perceive and understand vehicle behavior. This creates a “black box” dilemma: without real-time feedback or transparent explanations, passengers remain unaware of the vehicle’s intentions and maneuvers, especially in high-risk situations, fueling passenger distrust and concerns about loss of control in AV decisions. We argue that trustworthy driving systems must transcend mere accident prediction. They need to explain the when, where, and why of a potential event. AV must answer more than a binary question of whether an accident will occur. It must provide interpretable insights into when it is imminent, where it will happen, which agents are involved, and crucially, why it may unfold. Generating structured reports and natural-language feedback is thus essential to bridging this communication gap, ensuring accountability, and facilitating broader societal acceptance of AVs. Multimodal large language models (MLLMs) provide a promising interface for translating such structured evidence into understandable alerts and reports, thereby improving transparency, accountability, and public trust. However, unconstrained generation may introduce latency, unsupported statements, or errors propagated from upstream perception modules. MLLMs should therefore serve as communication and documentation interfaces, while the decision to trigger a critical warning should remain with the systems [17].

To this end, we propose CRASH++, a unified end-to-end framework that connects accident anticipation and localization of involved road users with timely passenger alerts and structured accident reporting. CRASH++ estimates the likelihood and timing of an accident, identifies the road users most likely to be involved, and uses MLLMs to translate these predictions into intelligible alerts and reports grounded in evidence. It therefore extends accident anticipation beyond isolated risk prediction to support communication between passengers and vehicles, accident analysis, and future safety improvement. Overall, the main contributions of this study can be summarized as follows:

- Expanding beyond the traditional scope of accident anticipation, this study develops a full-stack framework that integrates accident-involved object localization, precise and timely passenger alerts, human-centered feedback, and automated accident reporting. Our model not only predicts *whether* an accident will occur, but also *when* it might happen, *where* it will take place, and *why* it occurs. This comprehensive framework addresses a critical gap in accident prevention while advancing both safety and human-AI interaction in autonomous driving systems.
- We introduce CRASH++, a context-aware and temporally attentive risk perception model. Fast Fourier Transform (FFT)-based representations and Context-aware Attention Blocks (CAB) capture global scene cues that are often overlooked by methods based mainly on object bounding boxes, while a separate saliency encoder highlights visually informative regions. Object Focus Attention (OFA) models short- and long-term interactions among road users, and Temporal Focus Attention (TFA) aggregates informative historical observations, improving object-level risk assignment and supporting earlier and more stable accident anticipation.
- We develop human-centered interfaces for timely feedback and structured accident reporting. A lightweight MLLM converts risk predictions into concise verbal alerts, while a larger MLLM with retrieval-augmented generation (RAG) produces reports grounded in timestamped video evidence, object trajectories, and retrieved accident records. These components demonstrate the applicability of CRASH++ across systems by extending its predictive outputs to support interpretable communication and post-event documentation.
- We evaluate CRASH++ on three accident datasets (DAD [18], A3D [19], and CCD [20]) and four driver-attention datasets (DADA-2000 [21], BDD-A [22], DrFixD-rainy [23], and TrafficGaze [24]). CRASH++ achieves state-of-the-art (SOTA) performance in both accuracy and anticipation timeliness. Notably, it maintains strong results under challenging conditions such as data-missing observation input (10%-50%) and reduced training data (25%-75%), demonstrating robustness and strong generalization across diverse real-world scenes.

The following sections of this paper are structured as follows: Section 2 reviews related work. In Section 3, we delve into the formulation of accident anticipation and the details of CRASH++. In Section 4, we report the experimental results and comparisons. Finally, we conclude the paper in Section 5 with a summary of our findings and future work.

2. Related Work

Accident-centered video understanding encompasses accident classification, recognition, anticipation, localization, driver-attention modeling, and risk communication. These tasks require a system to interpret scene semantics and road-user interactions from dashcam video and, ultimately, translate the resulting evidence into actionable safety information. A fundamental challenge is the sparsity of accident evidence: accident-related content commonly occupies only 20–60 frames, or approximately 1–2 seconds, and follows a long-tail distribution [25]. Early cues are also easily obscured by poor illumination, adverse weather, motion blur, and partial occlusion. Research has therefore progressed from conventional statistical analysis toward deep learning methods that model richer scene semantics, agent interactions, and uncertain observations [26]. We review this literature from five complementary perspectives: accident classification, accident recognition, salient object detection, accident anticipation, and accident feedback.

Accident Classification. Accident classification determines the category of an observed traffic event. Progress in ego-view classification has been constrained by the limited diversity of available accident datasets, leading many studies to rely on surveillance footage with static viewpoints that may not fully represent onboard driving observations [27]. Deep learning methods have improved this setting by learning more discriminative scene representations. Kang et al. [28], for example, introduced the Vision Transformer-based ViT-TA model, which uses attention maps to emphasize objects relevant to accident classification. Graph Neural Networks (GNNs) [19, 29] further model relationships among road users and scene elements. Nevertheless, classification from monocular video often provides only a coarse description of the event and may not fully capture the spatiotemporal causality of dynamic traffic conflicts [30]. This limits its ability to support early intervention or identify the agents responsible for an emerging risk.

Accident Recognition. Accident recognition aims to determine whether an accident is present and to locate its spatial and temporal extent in an observed video. Early approaches commonly employed recurrent neural networks (RNNs) or Long Short-Term Memory (LSTM) units [20] to encode how traffic scenes evolve, often using attention to emphasize relevant regions or road users. Transformer-based models [31] and more advanced attention mechanisms [32] subsequently improved the modeling of long-range dependencies and complex visual interactions. Related studies have also explored reinforcement learning and task-specific objectives [33] to balance recognition accuracy with the need for earlier responses. Safety-oriented learning strategies further encourage models to identify accident evidence before it becomes visually dominant [7, 34]. More recent methods explore richer visual representations to improve motion and distance reasoning under challenging conditions [35]. Karim et al. [36] and Song et al. [37], for example, combine spatial and temporal attention to prioritize informative video segments and scene regions. Despite this progress, methods that depend heavily on detected bounding boxes [30] may overlook weak contextual cues outside predefined object regions, limiting recognition when accident evidence is small, occluded, or visually ambiguous.

Salient Object Detection. Driver-attention prediction and salient object detection provide a complementary perspective by identifying regions that are likely to influence human driving decisions. Predicting where drivers direct their attention can reveal cues associated with hazard perception and situation awareness, although driving attention is more context-dependent than generic visual saliency. Early studies approximated driver focus by manually labeling important scene elements, including traffic lights, vehicles, and pedestrians [38]. The introduction of large-scale eye-tracking datasets such as DR(eye)VE [39] enabled CNNs and fully convolutional networks to learn attention distributions directly from fixation data. Subsequent datasets broadened the range of driving conditions. TrafficGaze [24] provides gaze observations under clear-weather conditions, whereas DrFixD-rainy [23] extends evaluation to rain, nighttime driving, and reduced visibility. Recent work combines attention supervision with semantic, motion, and depth information. The DADA-2000 [21] has supported the development of multimodal methods for accident-related attention, while models that jointly reason about spatial and semantic relationships [40] have improved attention prediction in complex scenes. These advances emphasize the importance of attention modeling and salient object detection as an intermediate representation, which improves anticipation and provides insights into human perception.

Accident Anticipation. Accident anticipation differs from recognition by requiring a model to infer risk before a collision or other decisive visual evidence appears. RNN-based methods with attention [41] capture temporal dynamics and emphasize precursors such as abrupt braking or unexpected pedestrian motion. Spatiotemporal GNNs [35] extend this reasoning by explicitly representing interactions among vehicles, pedestrians, and other traffic participants. Generative approaches, including GANs and VAEs [42, 20], model possible future developments of the scene, while adaptive timing strategies such as AdaLEA [43] adjust learning according to accident imminence. Probabilistic representations, including Gaussian-mixture heatmaps [44], provide another means of expressing uncertainty about future risk. Recent work further incorporates specialized objectives and richer sensory cues to improve motion reasoning and balance

anticipation accuracy against warning timeliness [45]. Nevertheless, many methods prioritize object dynamics while treating global scene context less explicitly, even though road geometry, lane markings, crosswalks, traffic controls, and visibility can alter the risk associated with the same road-user behavior. Robustness to missing or degraded observations also remains comparatively underexplored. Accident anticipation is additionally related to the broader use of naturalistic driving data for safety testing and validation. Scenario-based safety testing organizes, generates, and evaluates representative or safety-critical situations for connected and automated vehicles [46]. From this perspective, CRASH++ does not aim to replace scenario generation methods; instead, it can complement scenario-driven safety analysis by identifying critical temporal segments, high-risk agents, and contextual risk cues from raw driving videos. The resulting frame-wise accident probabilities, time-to-accident estimates, and agent-wise risk scores may serve as structured evidence for scenario mining, critical case selection, and downstream safety validation.

Accident Feedback. Risk predictions are most useful when they can be communicated in a form that passengers and other stakeholders can understand. Explainable AI (XAI) methods improve transparency by exposing evidence associated with model decisions [47]. Monjurul et al. [48], for example, apply Grad-CAM to GRU-based models to visualize influential semantic regions. Attention-based explanations [49] and driver-behavior signals [50, 51] provide additional information about model focus and driver state. Sathyanarayana et al. [50] combine distraction detection with contextual scene cues to infer driver intent. However, visual explanations and technical attention maps may still be difficult for non-expert passengers to interpret during time-critical events. Language and multimodal models create new opportunities to translate perception and planning evidence into more accessible explanations [52]. DriveMLM [53] aligns visual and LiDAR observations with planning modules to generate contextual verbal rationales, while Talk2Drive [54] expresses autonomous-driving actions through natural language and supports interactive queries. These systems advance human-vehicle communication, but real-time accident feedback introduces additional requirements. Alerts must be concise, timely, grounded in observable evidence, and separated from the safety-critical decision of whether a warning should be triggered. Post-event documentation must additionally connect event descriptions with time-stamped video evidence, involved road users, trajectories, and environmental conditions. CRASH++ addresses these complementary needs by using risk predictions to support passenger-facing alerts and retrieval-grounded accident reports, while leaving the warning decision under the control of the risk-perception model.

3. Methodology

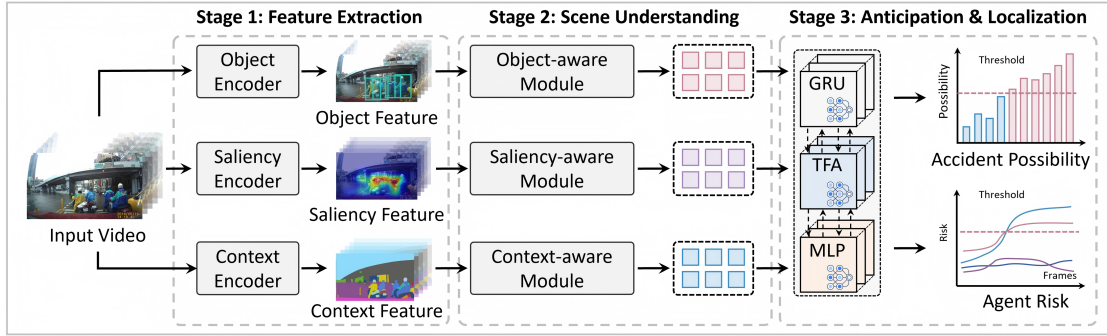
3.1. Problem Formulation

This study aims to propose a unified framework that jointly addresses three interrelated tasks: (1) accident anticipation and localization, (2) human-centered feedback, and (3) accident report generation.

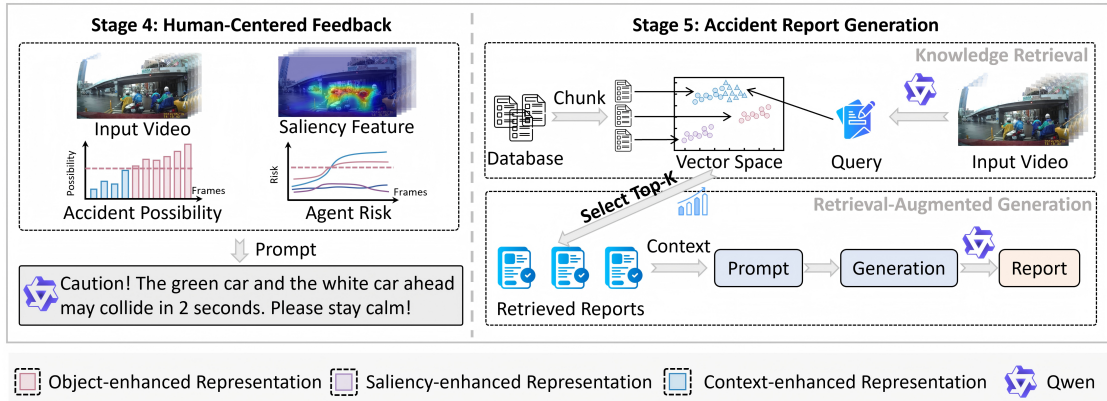
Accident Anticipation and Localization. The framework is designed to accurately and proactively anticipate traffic accidents at the earliest possible time. Given a dashboard camera video sequence of length T , denoted as $V = \{V_1, V_2, \dots, V_T\}$, the objective is to produce a corresponding sequence of accident probability scores $P = \{p_1, p_2, \dots, p_T\}$, $p_t \in [0, 1]$, where each p_t represents the estimated likelihood of an accident at frame t . To enable accident localization, CRASH++ further generates an agent-wise risk sequence at each frame: $R_t = \{r_t^1, r_t^2, \dots, r_t^{N_t}\}$, $r_t^j \in [0, 1]$, where N_t denotes the number of detected traffic agents in frame t , and each r_t^j reflects the estimated risk score of agent j . An agent is considered as potentially involved in the accident if $r_t^j > 0.5$. This formulation allows the system not only to predict whether an accident will occur but also to identify which agents are most likely to be involved. In cases where an accident occurs, we define the ground-truth accident time as $\tau \in [1, T]$. To evaluate how early a system can detect a potential accident, we introduce the Time-to-Accident (TTA), formulated as $\Delta t = \tau - t^0$, where t^0 is the earliest frame at which the predicted probability p_t exceeds a pre-established decision threshold p^0 . A video is classified as containing an accident (positive) if any $p_t \geq p^0$ for $t \leq \tau$, and as non-accident (negative) if no such threshold crossing occurs or if $\tau = 0$.

Human-Centered Feedback. The second task focuses on generating real-time feedback that is both informative and emotionally considerate for passengers. The inputs to this module include: (1) the dashboard video stream up to frame t ; (2) the accident probability sequence and corresponding agent-level risk scores up to frame t ; and (3) a structured prompt designed to contextualize potential risks. Based on these inputs, the system outputs human-centered verbal feedback that not only communicates the presence and nature of imminent risks but also alleviates passenger anxiety through clear, concise, and reassuring messages. This design integrates accident anticipation with empathetic interaction, thereby enhancing passenger trust and situational awareness.

(a) Risk-Perception Backbone



(b) Human-Centered Interfaces



Object-enhanced Representation
 Saliency-enhanced Representation
 Context-enhanced Representation
 Qwen

Figure 1: Overall architecture of our proposed CRASH++ model. The framework comprises a two-level, five-stage pipeline. Panel (a) presents the core risk-perception backbone (Stages 1-3). Stage 1 extracts complementary object features F_o , saliency features F_s , and context features F_c . Stage 2 refines them through the Object-aware, Saliency-aware, and Context-aware modules, producing $\bar{F}_o$, $\bar{F}_s$, and $\bar{F}_c$. Stage 3 performs attention-guided temporal fusion using GRU, TFA, and MLP layers to estimate frame-wise accident probabilities and agent-wise risk scores for early anticipation and involved-agent localization. Panel (b) presents the downstream human-centered interfaces (Stages 4-5). Stage 4 uses Qwen2-VL to translate the structured risk predictions into real-time passenger alerts, while Stage 5 employs retrieval-augmented generation to produce structured post-accident reports grounded in multimodal evidence and retrieved accident records.

Accident Report Generation. The third task focuses on generating comprehensive and structured accident reports, providing first-hand information for emergency responders, and creating valuable resources for traffic safety research. The inputs to this module include the complete dashboard video sequence, the corresponding accident probability scores, the agent-level risk sequences across all frames, and a structured prompt that guides the reporting process. Based on these inputs, the system produces a structured accident report that contains three essential components: (1) the timeline and location of the accident event, (2) the traffic agents involved and their estimated risk contributions, and (3) a detailed natural language description of the accident causes and context.

3.2. Overall Architecture

As illustrated in Figure 1, our proposed CRASH++ is a unified, end-to-end framework that can be conceptually understood through five sequential stages: (1) feature extraction, (2) scene understanding, (3) accident anticipation and localization, (4) human-centered feedback, and (5) accident report generation.

To clarify the methodological focus of CRASH++, we organize the framework into two levels. The first level is the core risk perception backbone, which consists of Stages 1-3 and represents the primary technical contribution of this work. Its goal is to learn context-aware spatio-temporal risk representations for early accident anticipation and accident-involved agent localization. This backbone integrates object-level agent interactions, human-attention-inspired saliency cues, and global scene context, with the FFT-based Context-aware Module and attention-guided temporal fusion serving as the key mechanisms for capturing subtle accident precursors beyond bounding-box-level

object dynamics. Building upon these structured risk representations, the second level consists of downstream human-centered interfaces, including real-time verbal feedback and structured accident report generation. These modules consume the outputs of the core risk perception backbone, such as accident probabilities, time-to-accident estimates, saliency cues, and agent-wise risk scores, and translate them into passenger-facing alerts or post-event reports. Together, these components extend the predictive capability of CRASH++ toward explainable communication and post-event documentation, enabling a more complete accident understanding and response pipeline.

In the feature extraction stage, the input video is processed through three complementary streams. The object encoder captures object features F_o that describe the identities and basic motion state of each traffic participant. The saliency encoder, inspired by human visual attention, extracts saliency features F_s that prioritize safety-critical regions over redundant details. The context encoder derives contextual features F_c that characterize the global context of the traffic scene. Unlike the feature extraction stage, which focuses on static representations, the scene understanding stage emphasizes the temporal evolution patterns of different feature streams. By modeling how feature dynamics unfold over time, it enables the identification of safety-critical behaviors such as abrupt lane changes and produces enhanced representations $\bar{F}_i, i \in \{o, s, c\}$ for each stream. In the third stage, the model fuses the three complementary representations and generates a frame-wise accident probability sequence $P = \{p_1, p_2, \dots, p_T\}$. Beyond conventional accident anticipation frameworks that end with risk prediction, CRASH++ extends its capability to provide human-centered support and structured reporting. In the fourth stage, it employs a lightweight MLLM to generate human-centered responses that help alleviate passenger anxiety during accident scenarios. In the final stage, the framework produces structured accident reports through the MLLM, offering valuable data resources for traffic safety research.

3.3. Stage 1: Feature Extraction

Accurate accident anticipation requires a multifaceted understanding of the traffic scene, as accidents may be triggered by a variety of factors, such as aggressive driving behaviors of traffic participants or extreme weather conditions. This diversity makes it challenging for any single source of features to capture all risk-relevant cues. To address this challenge, CRASH++ adopts three parallel encoders: an object encoder, a saliency encoder, and a context encoder, which together provide complementary perspectives of the scene.

Object Encoder. Accurate accident anticipation is predicated on object-level representations of traffic participants, whose identities, spatial positions, and motion states play a decisive role in the emergence of risky interactions. To the end, we employ Cascade R-CNN [55] as the visual backbone due to its high-precision detection capabilities, particularly in crowded or occluded scenes. Cascade R-CNN selects the top- n objects in each frame based on confidence scores, ensuring prioritization of agents most relevant to the traffic context. Each selected object is then processed by a VGG-16 network [56], which encodes its visual appearance into a d -dimensional embedding, yielding object features F_o .

Saliency Encoder. Treating all regions equally not only obscures risk-relevant cues but also introduces unnecessary computational overhead. To address this, we design a lightweight saliency encoder inspired by human visual attention [57]. This encoder adopts a dual-branch architecture for processing each video frame: one branch extracts multi-scale visual patterns from the raw image through an efficient Mamba backbone, while the other branch encodes high-level semantic cues with convolutional layers. These features are then fused by a spatial-channel attention module, yielding saliency features F_s that provide fine-grained representations of safety-critical regions in the scene.

Context Encoder. Accurate accident anticipation also requires holistic representations of the driving environment, since road geometry, weather conditions, and other background semantics provide essential context for interpreting the behaviors of traffic participants. To capture these scene-level characteristics, we process each frame with a VGG-16 network, followed by a lightweight Multi-Layer Perceptron (MLP) to encode global semantics with low computational cost. This yields context features F_c , which complement object and saliency cues to enable holistic scene understanding.

3.4. Stage 2: Scene Understanding

The key to accident anticipation lies in deeply understanding how the traffic scene evolves, since hazardous situations often emerge from the gradual accumulation of risky behaviors. In Stage 2, CRASH++ addresses this by modeling the temporal evolution of the object features F_o , saliency features F_s , and context features F_c extracted for each video frame, producing enhanced representations $\bar{F}_i, i \in \{o, s, c\}$ that make safety-critical temporal patterns explicit. This stage is implemented through three dedicated functional modules: an Object-aware Module, a Saliency-aware Module, and a Context-aware Module. The following subsections provide an overview of each proposed module.

3.4.1. Object-aware Module

The Object-aware Module aims to identify high-risk behaviors among dynamic traffic agents, such as vehicles, pedestrians, and motorcyclists, by modeling their spatiotemporal interactions, such as abrupt lane changes or sudden accelerations. These interactions are critical precursors to traffic incidents, and this module provides refined object-level representations to support accurate accident anticipation. The design leverages a novel attention-based mechanism, Object Focus Attention (OFA), inspired by multi-head attention in Transformer architectures [58]. The innovation of OFA lies in its use of dual-layer queries to capture both short-term and long-term interaction dynamics, optimizing the detection of accident-prone behaviors in traffic scenarios.

The module takes as input the object features $F_o \in \mathbb{R}^{n \times d}$ from Stage 1, where n is the number of detected objects and d is the embedding dimension, along with dual-layer hidden states $\bar{H}_{t-1} = \{\bar{H}_{t-1}^1, \bar{H}_{t-1}^2\}$ generated by a two-layer Gated Recurrent Unit (GRU). The hidden state $\bar{H}_{t-1}^1$ encodes short-term dynamics, such as immediate changes in an agent's position or velocity, while $\bar{H}_{t-1}^2$ captures long-term patterns, such as sustained trajectory trends. This dual-layer structure enables the module to model multi-scale temporal dependencies, addressing the diverse temporal characteristics of accident precursors. To quantify the interaction criticality between traffic entities i and j , OFA computes attention scores using similarity. Specifically, query vectors Q_t^1 and Q_t^2 are derived from the hidden states using independent MLPs and projection weights:

$$Q_t^1 = W_Q^1 \cdot \phi_{MLP}(\bar{H}_{t-1}^1), \quad Q_t^2 = W_Q^2 \cdot \phi_{MLP}(\bar{H}_{t-1}^2) \quad (1)$$

where W_Q^1 and W_Q^2 are learnable matrices. The queries Q_t^1 and Q_t^2 enable the module to focus on short-term and long-term interaction patterns, respectively, enhancing its ability to detect critical behaviors. Simultaneously, the object features F_o are mapped to key-value pairs (K_t, V_t) to establish spatial-semantic relationships:

$$K_t = W_K \cdot \phi_{MLP}(F_o), \quad V_t = W_V \cdot \phi_{MLP}(F_o) \quad (2)$$

where W_K and W_V are learnable projection matrices. The key K_t computes similarity with queries, while the value V_t aggregates relevant feature information. The interaction criticality score α_{ij} between agents i and j is computed as:

$$\alpha_{ij} = \phi_{Softmax} \left(\frac{W_\alpha Q_{t,i}^1 K_{t,j}^\top + W_\beta Q_{t,i}^2 K_{t,j}^\top}{\sqrt{d_k}} \right) \quad (3)$$

where d_k prevents gradient saturation, and W_α, W_β balance the contributions of short-term and long-term temporal patterns. This score prioritizes interactions with high accident potential, such as rapidly decreasing distances between agents, by combining multi-scale temporal information. The final object-enhanced representation $\bar{F}_o$ is obtained by aggregating value vectors weighted by these scores. Formally,

$$\bar{F}_o = \sum_j \alpha_{ij} V_{t,j} \quad (4)$$

This dynamic weighting focuses computational resources on agents exhibiting highly collision-conductive kinematics, such as vehicles cutting in, while suppressing irrelevant background actors. By integrating short-term and long-term dynamics, OFA provides a robust representation for downstream accident anticipation and localization, tailored to the stringent real-time operational requirements of autonomous driving.

3.4.2. Saliency-aware Module

The Saliency-aware Module aims to enhance the saliency features F_s extracted in Stage 1 by modeling their temporal evolution while preserving computational efficiency. Since safety-critical regions often exhibit temporal consistency (e.g., pedestrians crossing behavior will attract sustained attention), modeling the evolution of saliency over time is essential for reliable accident anticipation.

Let the input be a sequence of M consecutive saliency features, denoted as $\{F_s^{t-M+1}, \dots, F_s^t\}$, where F_s . The saliency features F_s of each frame are first spatially pooled and projected into saliency tokens X_s :

$$X_s = \phi_{proj}(\phi_{pool}(F_s)) \quad (5)$$

where ϕ_{proj} denotes a convolutional layer. Next, the saliency tokens X_s from all M frames are concatenated and processed with a transformer-based encoder to form a saliency tensor $\tilde{X}_s$.

To capture their temporal evolution, we treat the M axis as a temporal sequence and feed $\tilde{X}_s$ into a lightweight transformer encoder that performs two operations. First, a self-attention block models long-range temporal dependencies among salient regions, enabling the module to strengthen entities that exhibit consistent temporal patterns while down-weighting short-lived noise. Second, a feed-forward refinement block enriches token interactions and stabilizes the learned temporal context. This process can be formally represented as follows:

$$\hat{X}_s = \phi_{\text{attn}}(\tilde{X}_s) \quad (6)$$

where ϕ_{attn} denotes the transformer-based saliency refinement. To derive temporally enhanced saliency features, we further aggregate the refined sequence across the M frames using an attention-weighted pooling ϕ_{AttnPool} :

$$Z_s = \phi_{\text{AttnPool}}(\hat{X}_s) \quad (7)$$

which emphasizes frames with higher temporal saliency scores. Finally, the pooled representation yields the saliency-enhanced feature $\bar{F}_s$, which preserves the spatial distinctiveness of salient regions while capturing their temporal dynamics, thereby providing a compact and reliable signal for downstream accident anticipation.

3.4.3. Context-aware Module

Beyond inter-agent dynamics, global scene context provides essential information for distinguishing hazardous situations from visually similar but safe events. Road geometry, crosswalks, traffic controls, sidewalks, and surrounding infrastructure can substantially alter the risk associated with the same road-user behavior. The Context-aware Module therefore refines the scene-level representation by combining latent frequency-domain filtering with channel-wise attention. The module processes context features F_c from Stage 1 to capture dynamic scene patterns for accident anticipation. Unlike conventional time-domain approaches, which struggle to detect transient visual signals critical for risk assessment, it employs a frequency-domain transformation to uncover periodic or transient patterns, such as flashing lights or oscillating traffic signals, as missing these patterns may lead to undetected accident precursors and reduced prediction accuracy. Concretely, we first apply a one-dimensional convolution $\psi(\cdot)$ along the temporal axis of the feature sequence $F_c = (F_c^{t-M+1}, \dots, F_c^t)$. This operation expands the channel dimension from C to C' to both capture short-term dynamic patterns and create a richer feature representation for the subsequent Fast Fourier Transform analysis. The resulting temporally-enriched feature sequence is $F'_c = \psi(F_c)$. Following this, we apply the Fast Fourier Transform to the temporally-expanded features F'_c to convert them from the time domain to the spectral domain, which is computed as $S_c = \phi_{FFT}(F'_c)$. This yields a frequency-domain representation S_c , where each frequency component captures the rate of change in visual features across time.

Following this, we introduce a Spectral Gating Unit (SGU) to adaptively weight the frequency components. The SGU is implemented as a learnable set of parameters W_f that act as a frequency filter. W_f assigns an importance weight to each frequency channel, and we apply it via element-wise multiplication in the frequency domain: $\tilde{S}_c = W_f \odot S_c$. Here, the spectral gating operation amplifies frequencies that carry useful signal for accident anticipation and suppresses those that are likely noise, resulting in $\tilde{S}_c$, a spectrally-enhanced feature representation that highlights visual discontinuities, contours, and transient cues relevant for accident anticipation.

To integrate the frequency-domain insights back into our model, we perform an Inverse Fast Fourier Transform (IFFT) that brings the data back into the spatial domain. The IFFT of $\tilde{S}_c$ yields a refined time-domain feature sequence which carries frequency-filtered information. We denote this inverse-transformed feature as $Y_c = \phi_{IFFT}(\tilde{S}_c)$, corresponding to the same dimensionality as the earlier expanded features. Intuitively, Y_c is like an enhanced version of the original context features F_c , but augmented with global scene cues highlighted by the spectral analysis.

While the preceding Spectral Gating Unit (SGU) operates in the frequency domain to identify critical temporal patterns, it is equally crucial to determine the spatial origin of these dynamics within the scene. To this end, we introduce the Context-aware Attention Block (CAB). The CAB operates on the refined features Y_c and is designed to attend to important spatial regions and channels, effectively pinpointing which parts of the scene contribute most to a potential accident scenario. We implement the CAB using a combination of pooling operations and gating, inspired by attention modules in CNNs. First, for each Y_c we compute both a global average pooling and a global max pooling, yielding two statistic vectors: $p_{\text{avg}} = \phi_{\text{AvgPool}}(Y_c)$ and $p_{\text{max}} = \phi_{\text{MaxPool}}(Y_c)$. The average-pooled vector summarizes the overall presence of features, while the max-pooled vector captures the strongest activation of each feature channel.

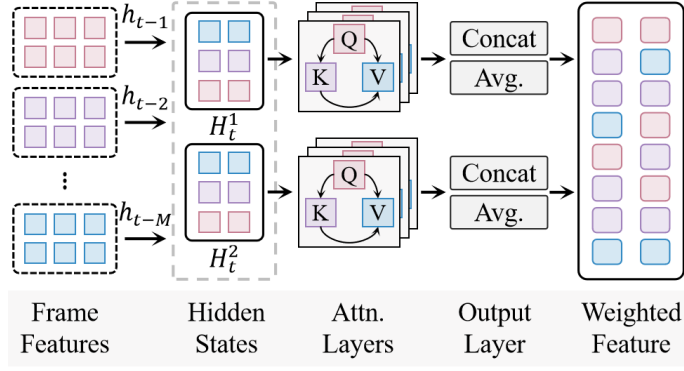


Figure 2: Overview of the proposed Temporal Focus Attention mechanism. Given frame-level features from the past M time steps, the corresponding hidden states H_t^1 and H_t^2 are constructed. These hidden states are processed by multiple attention layers, where queries (Q), keys (K), and values (V) are jointly learned to capture temporal dependencies across frames. The outputs of the attention layers are concatenated and averaged to produce temporally weighted features, enabling the model to selectively emphasize informative frames while suppressing irrelevant or redundant temporal signals.

We then multiply these two vectors element-wise to obtain an attention mask $\mathbf{m} = \mathbf{p}_{\text{avg}} \odot \mathbf{p}_{\text{max}}$. This mask $\mathbf{m}$ highlights channels that are consistently important (high average activation) and salient (high peak activation) in the scene. To apply this channel-wise attention, the mask vector $\mathbf{m}$ is first expanded to match the spatial dimensions of the feature map $\mathbf{Y}_c$. The resulting attention mask is then multiplied element-wise with $\mathbf{Y}_c$. This modulation effectively re-weights the feature map, amplifying the channels most informative for accident anticipation while suppressing the rest. Finally, we concatenate the original $\mathbf{Y}_c$ and the modulated version ($\mathbf{m} \odot \mathbf{Y}_c$) along the channel dimension, and pass this through a small feed-forward network (implemented as a point-wise MLP) to produce the context-enhanced representation $\tilde{F}_c$.

$$\tilde{F}_c = \phi_{\text{CAB}}(\phi_{\text{IFFT}}(\tilde{\mathbf{S}}_c)) = \phi_{\text{MLP}}(\phi_{\text{Softmax}}[(\phi_{\text{AvgPool}}(\mathbf{Y}_c) \odot \phi_{\text{MaxPool}}(\mathbf{Y}_c)) \odot \mathbf{Y}_c \oplus \mathbf{Y}_c]) \quad (8)$$

where ϕ_{IFFT} denotes the IFFT function, and ϕ_{AvgPool} and ϕ_{MaxPool} are the average and maximum pooling layers, respectively. Overall, this output serves as a context-enhanced scene descriptor, combining frequency-domain signal awareness with spatial attention. It complements the object-level reasoning by supplying a globally informed, structurally enriched view of the various dynamic traffic environments. Notably, the FFT-based operation does not assume that traffic accidents exhibit periodic patterns. Instead, it provides an efficient means of decomposing the temporal evolution of scene-level context features into components with different rates of variation. Low-frequency components primarily capture relatively stable or slowly varying contextual information, such as road layout, illumination conditions, and persistent scene structure, whereas higher-frequency components are more responsive to rapid changes in the visual context. For example, at an urban intersection, most of the scene may remain relatively stable over time, while safety-critical events, such as a pedestrian entering the roadway, a vehicle abruptly changing its motion, or a sudden change in traffic-state cues, can induce more pronounced temporal variations in the feature representation. By separating these temporal patterns, frequency-domain modeling enables the network to distinguish slowly varying background context from rapidly evolving cues that may be relevant to accident anticipation. The Spectral Gating Unit subsequently learns to emphasize informative frequency responses while suppressing irrelevant or noisy components. After the filtered representation is transformed back to the temporal domain through IFFT, the CAB further reweights informative feature channels to obtain a more discriminative context representation for accident anticipation.

3.5. Stage 3: Accident Anticipation and Localization

In real-world driving, accidents can occur abruptly and often represent only a small fraction of a driving video, with a long-tail distribution. Most previous approaches focus primarily on individual anomalous frames, using top-layer frame features passed through a linear layer to estimate accident probabilities. However, frames preceding an incident—even if not visually abnormal—may contain valuable contextual clues. Isolating “key frames” risks missing these early warnings, as anomalies may only become apparent through cumulative context. To address this, we propose

a multi-layer fusion mechanism that employs a two-step architecture to systematically aggregate spatio-temporal representations and enhance sensitivity to gradual, risk-indicative changes.

The first step of our mechanism is a temporal encoder, implemented as a robust dual-layer GRU module. It processes the fused feature vector, including the context vector F_c , context-enhanced representations $\bar{F}_c$, saliency-enhanced representations $\bar{F}_s$, and object-enhanced representations $\bar{F}_o$, to effectively model local temporal dependencies and subsequently generate a coherent sequence of hidden states. This process can be formally represented as follows:

$$h_t^i = \phi_{\text{GRU}}^i (F_c \parallel \bar{F}_c \parallel \bar{F}_s \parallel \bar{F}_o) \quad (9)$$

where $\parallel$ denotes vector concatenation and h_t^i is the hidden state at time t for the i -th GRU layer. Each hidden state h_t^i encapsulates information from the current frame and its immediate temporal neighborhood.

While the GRU captures local sequential patterns, its ability to model long-range dependencies remains limited. To address this, we introduce a historical aggregator based on the Temporal Focus Attention (TFA) mechanism, as shown in Figure 2. At each time step t , a sliding window $H_t = \{H_t^1, H_t^2\}$ is constructed, containing the most recent M hidden states, i.e., $H_t^i = \{h_{t-M+1}^i, \dots, h_{t-1}^i, h_t^i\}$. This design enables the aggregator to explicitly model pairwise relationships across all moments within the window, allowing the network to dynamically emphasize the most informative precursors to an accident, regardless of their temporal distance. In this way, the TFA extends beyond short-term context modeling and adaptively focuses on historical cues that carry high predictive value for accident anticipation.

Next, the aggregation process consists of k stacked attention layers. Within each layer, the sequence H_t^i is linearly projected into query, key, and value matrices. This can be formally denoted as follows:

$$\bar{Q}_t^i = W_Q^i H_t^i, \quad \bar{K}_t^i = W_K^i H_t^i, \quad \bar{V}_t^i = W_V^i H_t^i \quad (10)$$

Multi-head attention is then used to compute frame-wise relevance scores over the value vectors based on query-key similarity, scaled by the dimensionality $\sqrt{d_k}$:

$$R_y^i = \sum_{j=1}^m \text{head}_j^i = \sum_{j=1}^m \phi_{\text{Softmax}} \left(\frac{\bar{Q}_t^i (\bar{K}_t^j)^T}{\sqrt{d_k}} \right) \bar{V}_t^j \quad (11)$$

To enhance learning stability and improve gradient flow, especially in earlier layers, we incorporate Gated Linear Units (GLUs) and residual connections:

$$\bar{R}_y^i = \phi_{\text{Softmax}} \left(\phi_{\text{MLP}}(\phi_{\text{GLU}}(R_y^i)) \right) + R_y^i. \quad (12)$$

Each attention block output $\bar{R}_y^i$ is stacked across the k layers and combined using learnable weights $\gamma_1^i, \gamma_2^i, \dots, \gamma_k^i$ to capture aggregated temporal dependencies. These weights are unconstrained, unnormalized, and unregularized during training, allowing the model to adjust their values to optimize temporal feature integration dynamically. Formally,

$$\bar{R}_{\text{weighted}}^i = \sum_{y=1}^k \gamma_y^i \cdot \bar{R}_y^i \quad (13)$$

Finally, the outputs from each layer are fused through learnable weighted aggregation followed by average pooling, yielding the final temporally enriched representation $\bar{H}_t$. This can be represented as follows:

$$\bar{H}_t = \phi_{\text{AvgPool}}(\bar{R}_{\text{weighted}}^1) \oplus \phi_{\text{AvgPool}}(\bar{R}_{\text{weighted}}^2) \quad (14)$$

The GLU's sigmoid gate $\sigma(\cdot)$ filters out noise while preserving salient features, especially in the early layers where gradients are weakest. Next, the output $\bar{H}_t$ serves as a temporally refined representation and is passed to the GRU-MLP network for final crash probability and agent-wise risk score prediction within the CRASH++ framework. By dynamically integrating frame-level cues over time, the multi-layer fusion mechanism enhances the model's ability to detect early indicators of potential collisions, particularly in long-tail, rare-event, and visually ambiguous scenarios.

Accident Report	Vehicle	Environment	Roadway	Collision
1. Accident Number	Vehicle Information	Weather	Roadway Condition	Collision Type
2. Time of Accident	1. Manufacturer	1. Clear	1. Usual	1. Head-On
Vehicle	2. Vehicle Year	2. Cloudy	2. Holes	2. Side Swipe
3. Vehicle Information	3. Vehicle Mode	3. Rainy	3. Obstruction	3. Rear End
4. Movement	Movement	4. Snowing	4. Reduced Width	4. Broadside
Environment	1. Stopped	5. Fog	5. Flooded	5. Hit Object
5. Weather	2. Straight	6. Windy	6. Construction	6. Overturned
6. Lighting Condition	3. Ran Off Road	7. Stormy	7. Cracked	7. Veh/Pedestrian
Roadway	4. Right Turn	Lighting Condition	Roadway Surface	8. Other
7. Roadway Condition	5. Left Turn	1. Daylight	1. Dry	Outcomes
8. Roadway Surface	6. Backing	2. Dusk-Dawn	2. Wet	1. Property Damage
Collision	7. Slowing	3. Dark-Street Light	3. Snowy-Icy	2. Injury
9. Collision Type	8. Merging	4. Dark-No Light	4. Muddy	3. Fatalities
10. Outcomes				

Figure 3: Structured factors in traffic accident reports. The contributing factors are organized into four categories: vehicle, environment, roadway, and collision. Vehicle factors include vehicle information (e.g., manufacturer, year, model) and movement states (e.g., turning, stopping, merging). Environmental factors consist of weather conditions and lighting states, such as clear, rainy, foggy, or dusk. Roadway factors describe roadway conditions (e.g., obstruction, construction, flooding) and surface states (e.g., wet, icy, slippery). Collision factors summarize the accident outcome, including collision type (e.g., rear-end, side swipe, pedestrian-involved) and accident consequences such as property damage, injury, or fatalities.

3.6. Stage 4: Human-Centered Feedback

Recent autonomous driving studies increasingly recognize that prediction accuracy alone is insufficient for passenger trust, particularly when passengers cannot understand imminent risks or vehicle intentions during sudden maneuvers [9, 17, 59]. CRASH++ therefore augments accident anticipation with real-time verbal feedback, which translates accident probabilities, time-to-accident estimates, saliency cues, and agent-wise risk scores into concise passenger-facing alerts. With the development of LLMs and MLLMs [60, 61], complex multimodal risk representations can be converted into natural-language explanations. We adopt Qwen2-VL-2B [62] for lightweight alert generation, using it only as a communication interface that consumes structured risk outputs and video context. The safety-critical warning decision remains handled by the accident anticipation and localization modules.

To contextualize these visual features, the system incorporates key outputs from the accident anticipation stage, including frame-wise accident probabilities and predicted TTA. These signals are used to construct structured prompts to guide the MLLM. In particular, the prompt strategy is tailored to reflect the sudden and high-stakes nature of road accidents. The generated verbal prompts are concise yet precise, communicating the essential “when”, “where”, and “what” of a potential accident. For example, in the case of an imminent rear-end collision, the system may issue a real-time warning: “Watch out! A red car and truck ahead may collide in 2 seconds.”. These alerts are crafted for clarity and immediacy, ensuring they are easily understood by non-expert passengers. In contrast, during routine driving conditions, the system shifts to a more conversational and reassuring tone to maintain passenger comfort and engagement. For instance, the system might be friendly enough to say: “All clear. We’re enjoying a smooth ride through downtown.”. This adaptive communication strategy not only enhances transparency but also ensures that passengers remain informed and comfortable throughout the journey. As illustrated in Figure 1 (b), the system uses these prompts to integrate multimodal information, producing human-centered dialogue that clearly narrates the risk’s timing, location, and underlying cause. This addresses a pivotal gap in AD by bridging perception with intuitive explanation, thereby fostering the passenger’s trust and operational transparency, which are pivotal for the broad adoption of fully AV.

3.7. Stage 5: Accident Report Generation

The final stage of the CRASH++ framework addresses the question of “why” an accident occurs. We adopt the MLLM Qwen2-VL-7B [62], generating a structured accident report that integrates perceptual data from the model with external, real-world accident knowledge. To enhance MLLM’s comprehension of complex scenes and minimize hallucinations, we employ the Retrieval-Augmented Generation (RAG) technology. This technology dynamically retrieves relevant information from an external knowledge base during inference, ensuring the generated reports are context-aware and anchored in authentic accident data, enhancing the reliability and applicability of the output.

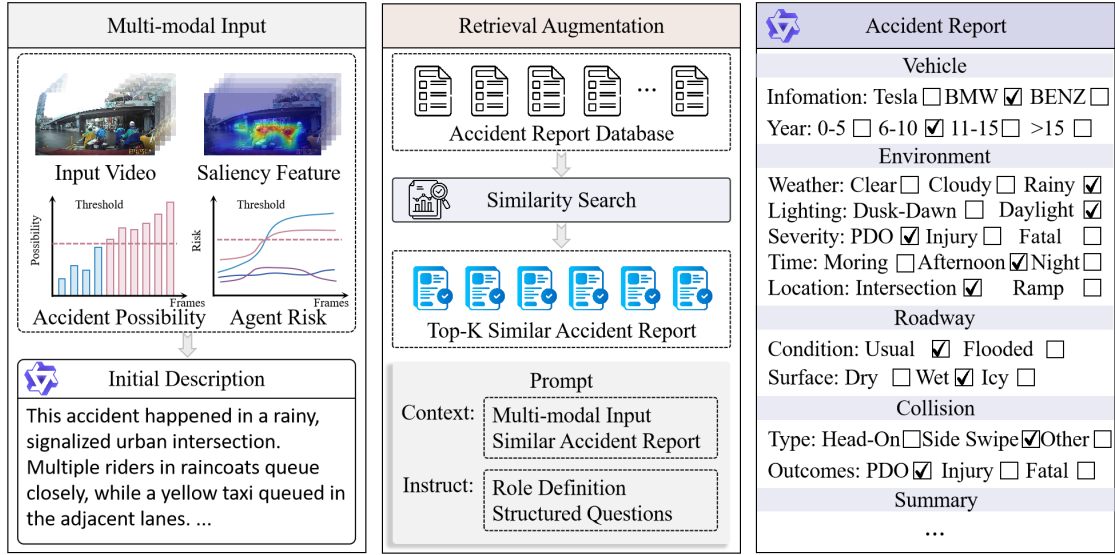


Figure 4: Retrieval-augmented accident report generation in our proposed CRASH++ model. The framework consists of three stages. First, multi-modal inputs, including the raw video stream, saliency features, accident probabilities, and agent-level risk scores, are processed to generate an initial accident description. Second, the description is compared with a database of official accident reports through similarity search, retrieving the top- k most similar reports as external evidence. Finally, a prompt is constructed that integrates the multi-modal context, retrieved reports, and structured instructions and is passed to Qwen2-VL to produce a structured accident report aligned with standardized formats.

The external knowledge base is constructed from over 100 real-world driving reports published by the California Department of Motor Vehicles (DMV) between 2019 and 2024. As illustrated in Figure 3, these reports are annotated according to a standardized schema that captures the principal factors involved in a crash. This schema includes four key categories: vehicle, environment, roadway, and collision. Specifically, the vehicle-related fields include information such as manufacturer, year, and movement states (turning, stopping, merging). Furthermore, the environmental factors encompass weather and lighting conditions. Roadway descriptors include pavement conditions, surface states, and construction activity. Collision-related attributes describe the type of crash and its consequences in terms of property damage, injuries, or fatalities. Aligning generated accident reports with this structure ensures interpretability, interoperability with existing reporting standards, and relevance for traffic safety analysis.

As depicted in Figure 4, the report generation pipeline operates in three sequential steps. During inference, the Qwen2-VL-7B first generates a preliminary scene annotation for the multimodal input (accident video sequence, saliency feature, possibility of an accident), which summarizes the event, including involved agents, their interactions, and the predicted accident type. Then, this initial annotation is used as a query to retrieve the most relevant cases from the DMV report database. Each report in the database is pre-encoded into a dense vector representation, enabling an efficient similarity search. The top- k most semantically similar reports are selected to serve as contextual evidence. Finally, an augmented prompt that integrates the initial description, the retrieved evidence, and instructions for structured reporting guides the model to produce a finalized report $\mathcal{A}$ that details the event sequence and root causes.

Overall, this retrieval-grounded generation ensures high factual accuracy and significantly mitigates hallucinations, thereby producing actionable outputs that benefit both real-time emergency response and long-term safety analytics.

3.8. Learning Strategy

To address the challenges of early prediction and robust feature learning, we formulate a multi-task loss $\mathcal{L}_f$ comprising three components: (1) anticipation loss $\mathcal{L}_a$, (2) localization loss $\mathcal{L}_o$, and (3) enhancement loss $\mathcal{L}_e$.

3.8.1. Anticipation Loss with Temporal Prioritization

The primary objective (timely accident anticipation) is guided by a temporally-weighted cross-entropy loss $\mathcal{L}_a$. Unlike conventional loss functions that treat all frames equally, $\mathcal{L}_a$ incorporates an exponential decay term $e^{-\frac{1}{2} \max(\frac{t-l}{f}, 0)}$

that progressively increases penalty weights as frames approach the ground-truth accident time τ . Mathematically,

$$\mathcal{L}_a = \frac{1}{B} \sum_{v=1}^B \left[-l_v \sum_{t=1}^T e^{-\frac{1}{2} \max(\frac{\tau-t}{f}, 0)} \log(p^t) - (1-l_v) \sum_{t=1}^T \log(1-p^t) \right] \quad (15)$$

where B is the batch size, l_v is the binary label of accident occurrence within each video (1 for an accident, 0 for none), while T is the total number of frames per video, and f denotes the video frame rate.

3.8.2. Localization Loss with Object Prioritization

This loss function is responsible for supervising the model's localization of objects involved in the accident. It supervises per-object accident localization by assigning a binary label to each tracked agent over time. For video $v \in [1, V]$, frame $t \in [1, T]$, and object $n \in [1, N]$ that appears in the video, let $\mathcal{L}_o^{p,t} \in \{0, 1\}$ indicate whether the object is accident-involved (1) or not (0); for brevity, $\mathcal{L}_o^{p,t}$ denotes the label of object n at frame t in video v . Let $l_o^{t,n} \in (0, 1)$ be the predicted involvement probability for object n at frame t . The localization loss $\mathcal{L}_o$ averages binary cross-entropy over videos, frames, and objects. It can be formally represented as follows:

$$\mathcal{L}_o = \frac{1}{V} \frac{1}{T} \frac{1}{N} \sum_{v=1}^V \sum_{t=1}^T \sum_{n=1}^N \left[-\mathcal{L}_o^{p,t} \log(l_o^{t,n}) - (1 - \mathcal{L}_o^{p,t}) \log(1 - l_o^{t,n}) \right]. \quad (16)$$

Here, V is the number of dashcam videos, and N is the number of objects per frame.

3.8.3. Enhancement Loss for Feature Stability

To mitigate error propagation in temporal modeling, particularly in early GRU hidden states, we introduce an enhancement loss $\mathcal{L}_e$ that reinforces feature consistency. Formally,

$$\mathcal{G}_e = \phi_{\text{MLP}}(\phi_{\text{MHA}}(\phi_{\text{PE}}(h_1^2, h_2^2, \dots, h_T^2))) \quad (17)$$

where ϕ_{MHA} and ϕ_{PE} denote the multi-head attention and position encoding mechanisms, respectively. The enhancement loss is computed using binary cross-entropy over $\mathcal{G}_e$, promoting its alignment with the accident occurrence label:

$$\mathcal{L}_e = \frac{1}{B} \sum_{v=1}^B \left[-l_v \log(\mathcal{G}_e) - (1 - l_v) \log(1 - \mathcal{G}_e) \right] \quad (18)$$

3.8.4. Uncertainty-Aware Loss Fusion

The final objective $\mathcal{L}_f$ optimally balances anticipation loss $\mathcal{L}_a$, localization $\mathcal{L}_o$, and enhancement loss $\mathcal{L}_e$ through homoscedastic uncertainty weighting. It can be mathematically expressed as follows:

$$\mathcal{L}_f = \frac{\mu_1}{2\rho_1^2} \mathcal{L}_a + \frac{\mu_2}{2\rho_2^2} \mathcal{L}_o + \frac{\mu_3}{2\rho_3^2} \mathcal{L}_e + \log(\rho_1 \rho_2 \rho_3) \quad (19)$$

Here, μ_1 , μ_2 , and μ_3 are fixed task-specific weights, while ρ_1 , ρ_2 and ρ_3 are initialized to 1 and learned during training. Overall, this multi-task objective encourages both early and accurate accident anticipation while enhancing the model's sensitivity to relevant temporal and semantic features. Together, these losses guide the model toward robust, context-aware decision-making under real-world driving conditions.

4. Experiment

4.1. Experiment Setups

4.1.1. Datasets

As shown in Table 1, we evaluate CRASH++ on two groups of datasets according to the task setting. For accident anticipation, we use three dashcam accident datasets, including DAD [18], A3D [19], and CCD [20]. These datasets provide video-level accident labels and accident timestamps and are used to evaluate early accident prediction with AP, mTTA, and TTA@R80. Among them, DAD further provides object-level annotations, including bounding boxes, object IDs, and accident-involved object labels, and is therefore also used for accident-involved object localization.

Table 1

Task-specific use of datasets and supervision signals. Accident labels/timestamps are used only for anticipation/localization, while gaze or fixation maps are used only for saliency evaluation.

Task	Datasets	Supervision signal	Metrics
Accident anticipation	DAD, A3D, CCD	Video label; accident time	AP, mTTA, TTA@R80
Object localization	DAD	Boxes; object IDs; involved-object labels	AOLA
Saliency evaluation	DADA-2000, BDD-A, DrFixD-rainy, TrafficGaze	Fixation / gaze-density / attention maps	KLD, CC, SIM, NSS, AUC-J

To evaluate whether the saliency branch aligns with human visual attention, we further use four driver-attention datasets, including DADA-2000 [21], BDD-A [22], DrFixD-rainy [23], and TrafficGaze [24]. These datasets provide fixation maps, gaze-density maps, or aggregated attention maps, and are used only for saliency evaluation with standard attention metrics, such as KLD, CC, SIM, NSS, and AUC-J. We emphasize that accident labels and timestamps are used only for accident anticipation/localization, while gaze or fixation annotations are used only for saliency evaluation. Detailed dataset statistics, supervision signals, and task-specific usage are provided in **Appendix A**.

4.1.2. Data Segmentation and Missing Data Simulation

This study reports standard train, validation, and test splits and follows each dataset’s protocols [4, 57]. To simulate real-world sensor failures, we benchmark model robustness under data insufficiency by introducing synthetic sensor dropouts and frame losses into the DAD, CCD, and A3D datasets. Specifically, we create scenarios with randomly missing frames at rates of 10%, 20%, and 50%, and structured patterns such as one frame missing out of every five. A stochastic selection mechanism ensures that missing data patterns are unbiased and realistic. Furthermore, we simulate data scarcity conditions by restricting the training data to 75%, 50%, and 25% subsets. This enables evaluation of the model’s generalizability under limited supervision. We then evaluate the performance of our proposed model on both *complete* and *missing* datasets. These evaluations aim to assess the model’s adaptability to unfamiliar data and its ability to deal with missing data, providing an assessment of the robustness of our model. All methods are evaluated using the same data-sampling protocol, training-data ratios, and frame-drop settings. Because data subsampling and random frame removal are stochastic, absolute results may vary across random realizations; accordingly, the reported comparisons are intended to assess relative performance under the common protocol used in this study.

4.1.3. Evaluation Metrics

We use task-specific metrics to evaluate accident anticipation, accident-involved object localization, and saliency alignment. For accident anticipation, we report AP, mTTA, and TTA@R80 to measure classification quality and early-warning timeliness. For accident object localization, we use AOLA to evaluate whether the model identifies the correct accident-related agents. For saliency evaluation, we follow standard driver-attention protocols and report KLD, NSS, SIM, CC, AUC-J, and AUC-B to measure the alignment between predicted saliency maps and human gaze or fixation distributions. All compared methods follow the same operating protocol. Unless otherwise specified, the accident decision threshold is fixed as $p^o = 0.5$, and the agent-involvement threshold is fixed as $r_t^j = 0.5$. TTA and mTTA are computed by the first threshold-crossing rule, and TTA@R80 is reported at the dataset-level operating point corresponding to 80% recall. No additional score calibration or temporal smoothing is applied. Overall, these metrics assess the following aspects of the model: when it anticipates accidents (mTTA, TTA@R80); how reliably it classifies them (AP); which objects it attends to (AOLA); and whether its visual focus aligns with that of human drivers (KLD, NSS, SIM, CC, AUC-J, AUC-B). Together, they provide a thorough evaluation of accident anticipation, localization, and salient object detection in CRASH++. More metric definitions are provided in **Appendix B**.

4.1.4. Implementation Details

CRASH++ is implemented in PyTorch and trained on four NVIDIA A100 GPUs with 80GB memory. All models are trained for 80 epochs with a batch size of 10 using the Adam optimizer. The initial learning rate is set to 1×10^{-4} , and a ReduceLROnPlateau scheduler is used to adjust the learning rate according to validation performance. The input video is represented as a tensor of shape (B, T, C, W, H) , where B , T , C , W , and H denote the batch size, number of frames, number of channels, image width, and image height, respectively. Following the dataset settings, we use $T = 100$ for DAD and A3D, and $T = 50$ for CCD. The feature dimensions are set to $D = 4096$ and $d = 512$. When predicting the accident probability at frame t , the model only uses frames from 1 to t , without accessing future frames.

Table 2

Comparison of models seeking **balance** between mTTA and AP on the *complete* datasets. **Bold** and underlined values represent the best and second-best performance. Instances where values are not available are marked with a dash ("-").

Model	DAD			CCD		A3D	
	AP (%)↑	mTTA (s)↑	AOLA↑	AP (%)↑	mTTA (s)↑	AP (%)↑	mTTA (s)↑
DSA [18]	48.1	1.34	-	98.7	3.08	92.3	2.95
AdaLEA [43]	52.3	3.43	-	99.2	3.45	92.9	3.16
UString [20]	53.7	3.53	-	99.5	3.74	93.2	3.24
DSTA [36]	59.2	2.90	-	<u>99.6</u>	3.87	93.5	2.87
GSC [29]	60.4	2.55	-	<u>99.4</u>	3.68	94.9	2.62
AccNet [4]	60.8	<u>3.58</u>	0.78	99.5	3.78	<u>95.1</u>	<u>3.26</u>
Ours	72.7	4.28	0.92	99.7	3.94	96.5	3.43

Table 3

Performance comparison on the DAD dataset in terms of peak AP and TTA@80. TTA@80 denotes the mTTA measured at 80% recall. **Bold** and underlined values indicate the best and second-best results in each category, respectively.

Model	Backbone	Publication	AP (%)↑	mTTA (s)↑	TTA@R80 (s)↑
DSA [18]	VGG-16	ACCV'16	63.50	1.67	1.85
UniFormerv2 [63]	Transformer	ICCV'23	65.24	-	-
VideoSwin [64]	Transformer	CVPR'22	65.45	-	-
DSTA [36]	VGG-16	IEEE TITS'22	66.70	1.52	2.39
UString [20]	VGG-16	ACMMM'20	68.40	1.63	2.18
XAI-Accident [48]	ResNet50	TRR'22	64.32	1.80	0.68
GSC [29]	VGG-16	IEEE TIV'23	68.90	1.33	2.14
AccNet [4]	VGG-16/ZoeDepth	AAP'24	<u>70.10</u>	1.73	2.23
Ours	TFA/SAE	-	79.28	4.08	4.23

4.2. Evaluation Results

4.2.1. Evaluation Performance Comparison on Overall Real-world Datasets

Table 2 summarizes performance on three real-world benchmarks. Specifically, our approach achieves SOTA results while maintaining a favorable AP-mTTA trade-off. On the DAD dataset, which spans diverse urban traffic scenarios, our model attains an AP of 72.7% and an AOLA of 0.92, while providing longer lead times (mTTA 4.28 s) within the 5 s anticipation horizon, outperforming the second-place AccNet by 19.57%, 19.55%, and 17.95%, respectively. These improvements indicate earlier and more reliable anticipation, along with stronger accident-object localization. On CCD and A3D datasets, the method likewise sets the best overall balance, surpassing competing approaches on both AP and mTTA. These results underscore the model's robustness across complex and heterogeneous traffic environments, including varying congestion levels and urban road conditions. As shown in Table 3, our model not only delivers the highest AP (79.28%) but also provides the longest anticipation window, with an mTTA of 4.08 s. Within a 5-second horizon, this extended lead time is highly valuable for safety-critical driving, as it provides additional time for preventive actions. While mTTA alone may not fully capture prediction quality, the combination of high AP and long anticipation lead time offered by our model represents a significant advancement in accident anticipation.

4.2.2. Evaluation Performance Comparison on Missing Datasets

The results in Table 4 show that our model remains robust under missing observations. Across data loss rates of 10%, 20%, and 50%, it consistently outperforms all baselines on the evaluated datasets. For example, under 10% missing observations, our model achieves 70.6% AP and 4.22 s mTTA on DAD, both higher than the competing methods. Even when the missing rate increases to 50%, it still maintains clear advantages. Under periodic dropout, where one out of every five frames is missing, our model also shows strong adaptability. It achieves 62.3% AP on DAD, outperforming the second-best GSC model by 12.1%, and reaches 3.36 s mTTA on A3D. These results indicate that CRASH++ can preserve reliable accident anticipation and reasonable lead time even when the input video is incomplete.

As shown in Table 5, consecutive burst dropout causes a more pronounced performance degradation than random or periodic frame loss. This result is intuitive because random missing frames mainly remove isolated observations,

Table 4

Robustness comparison under missing-frame conditions. “Drop-10%”, “Drop-20%”, and “Drop-50%” indicate that 10%, 20%, and 50% of the video frames are randomly removed during evaluation, respectively. Accordingly, “1 in 5 Frames” denotes periodic frame loss, with one frame removed every five frames. No interpolation or frame reconstruction is applied. For each dataset and condition, all methods use identical sampled training subsets and missing-frame masks. Absolute values may differ from those obtained under other random realizations, whereas the within-table comparisons are conducted under the same controlled conditions. **Bold** and underlined values indicate the best and second-best results, respectively.

Dataset	Model	Drop-10%		Drop-20%		Drop-50%		1 in 5 Frames	
		AP↑	mTTA↑	AP↑	mTTA↑	AP↑	mTTA↑	AP↑	mTTA↑
DAD	UString [20]	53.5	<u>3.48</u>	52.2	<u>3.46</u>	52.4	<u>3.11</u>	52.6	<u>3.36</u>
	DSTA [36]	56.7	2.50	<u>55.9</u>	2.47	<u>54.8</u>	1.62	55.2	2.14
	GSC [29]	57.2	2.56	54.8	2.35	51.4	1.81	55.6	2.21
	Ours	70.6	4.22	68.8	4.18	64.3	4.10	62.3	4.02
A3D	UString [20]	92.6	<u>3.14</u>	<u>92.1</u>	3.05	80.3	2.76	<u>92.5</u>	<u>3.10</u>
	DSTA [36]	<u>92.8</u>	2.82	90.3	<u>2.61</u>	<u>81.8</u>	<u>2.34</u>	91.5	2.77
	Ours	96.1	3.40	94.5	3.23	90.7	3.01	95.4	3.36
CCD	UString [20]	98.7	3.70	96.2	3.34	94.5	3.39	96.8	3.64
	DSTA [36]	<u>98.8</u>	<u>3.79</u>	<u>97.2</u>	<u>3.51</u>	<u>94.7</u>	3.01	<u>97.9</u>	<u>3.73</u>
	Ours	99.2	3.85	98.8	3.74	96.3	3.60	98.5	3.80

Table 5

Robustness evaluation under consecutive burst dropout without reconstruction. Burst-0.5s and Burst-1.0s indicate that a continuous temporal segment of 0.5 and 1.0 seconds is masked from the input sequence, respectively. For accident videos, the missing segment is sampled from the two-second pre-accident interval $[\tau - 2\text{s}, \tau]$, where τ denotes the ground-truth accident time. For non-accident videos, it is sampled from the entire clip. Missing observations are represented by blank frames or zero-valued feature placeholders to preserve the original sequence length. Results are reported as mean $\pm$ standard deviation over three random trials. **Bold** and underlined values indicate the best and second-best results, respectively.

Dataset	Model	Burst-0.5s		Burst-1.0s	
		AP↑	mTTA↑	AP↑	mTTA↑
DAD	UString [20]	49.8 $\pm$ 0.5	<u>2.93 $\pm$ 0.06</u>	46.4 $\pm$ 0.6	<u>2.56 $\pm$ 0.08</u>
	DSTA [36]	51.7 $\pm$ 0.4	1.39 $\pm$ 0.05	48.5 $\pm$ 0.5	1.13 $\pm$ 0.06
	GSC [29]	49.2 $\pm$ 0.6	1.52 $\pm$ 0.07	45.4 $\pm$ 0.7	1.25 $\pm$ 0.07
	Ours	61.4 $\pm$ 0.4	3.84 $\pm$ 0.05	57.8 $\pm$ 0.5	3.48 $\pm$ 0.07
A3D	UString [20]	78.4 $\pm$ 0.6	2.58 $\pm$ 0.05	74.2 $\pm$ 0.7	2.26 $\pm$ 0.07
	DSTA [36]	<u>79.3 $\pm$ 0.5</u>	2.16 $\pm$ 0.06	75.7 $\pm$ 0.6	1.87 $\pm$ 0.06
	Ours	88.5 $\pm$ 0.4	2.86 $\pm$ 0.04	85.1 $\pm$ 0.5	2.54 $\pm$ 0.06
CCD	UString [20]	93.1 $\pm$ 0.4	3.18 $\pm$ 0.05	90.3 $\pm$ 0.5	2.80 $\pm$ 0.06
	DSTA [36]	<u>93.6 $\pm$ 0.3</u>	2.78 $\pm$ 0.06	<u>90.9 $\pm$ 0.5</u>	2.42 $\pm$ 0.07
	Ours	95.2 $\pm$ 0.3	3.43 $\pm$ 0.04	93.5 $\pm$ 0.4	3.15 $\pm$ 0.05

whereas burst dropout breaks a continuous segment of the temporal risk-evolution process. When the missing interval increases from 0.5 s to 1.0 s, both AP and mTTA decrease further, indicating that longer sensor blackout not only weakens accident/non-accident discrimination but also delays the first reliable warning. In particular, mTTA is more sensitive to burst dropout because early accident anticipation depends heavily on temporally coherent precursor cues, such as gradual changes in inter-agent distance, lane position, and contextual risk accumulation. The degradation pattern also varies across datasets. The performance drop is more evident on DAD, where accident scenes are more diverse, visually cluttered, and often contain subtle precursors that may be fully hidden by a continuous missing segment. In contrast, CCD remains relatively stable because many accident patterns are more visually separable, allowing the model to recover from the remaining observations before and after the dropout interval. Despite the more challenging setting, CRASH++ consistently outperforms the baselines across all datasets and burst-dropout lengths. These results suggest that the proposed temporal fusion mechanism can exploit the remaining pre- and post-gap observations by integrating object interactions, saliency cues, and contextual features. The mTTA advantage

Table 6

Comparison of models trained with limited training sets on evaluation metrics for *missing* video frames. “Drop-10%”, “Drop-20%”, and “Drop-50%” indicate that 10%, 20%, and 50% of video frames are randomly treated as missing during evaluation, respectively. The “1 in 5 Frames” setting denotes a periodic frame-loss pattern in which one frame is missing every five frames, simulating regular frame-rate degradation. No interpolation, frame synthesis, or reconstruction-based preprocessing is applied to recover missing content. “DAD (75%)” means only 75% of the videos in the DAD dataset are used for training. **Bold** and underlined values represent the best and second-best performances in each category, respectively.

Dataset	Model	Drop-10%		Drop-20%		Drop-50%		1 in 5 Frames	
		AP (%)↑	mTTA (s)↑	AP (%)↑	mTTA (s)↑	AP (%)↑	mTTA (s)↑	AP (%)↑	mTTA (s)↑
DAD (75%)	UString [20]	51.2	3.41	48.5	3.31	50.9	<u>3.02</u>	50.5	3.25
	DSTA [36]	54.1	2.40	<u>53.1</u>	2.39	<u>50.5</u>	<u>1.54</u>	<u>53.2</u>	2.10
	GSC [29]	54.3	2.43	51.5	2.30	49.1	1.74	52.3	2.15
	Ours	68.4	4.17	65.9	4.09	62.7	4.04	66.2	4.13
A3D (75%)	UString [20]	91.4	3.05	89.3	2.94	85.8	2.69	89.6	2.95
	Ours	95.2	3.33	93.7	3.25	90.1	3.18	93.8	3.28
CCD (75%)	UString [20]	96.5	<u>3.61</u>	95.1	3.47	92.4	3.24	95.2	<u>3.49</u>
	DSTA [36]	96.9	3.64	95.7	<u>3.41</u>	93.3	2.88	<u>95.2</u>	<u>3.40</u>
	Ours	98.7	3.82	97.9	3.69	95.6	3.51	97.5	3.67
DAD (50%)	UString [20]	46.2	3.16	44.1	<u>3.05</u>	41.3	2.24	45.8	<u>3.11</u>
	DSTA [36]	52.0	<u>2.32</u>	50.8	2.29	<u>48.2</u>	1.49	51.3	<u>2.34</u>
	GSC [29]	<u>50.7</u>	2.29	<u>49.1</u>	2.17	46.5	1.62	48.3	2.20
	Ours	64.5	4.08	63.4	3.97	60.3	3.87	63.7	4.01
A3D (50%)	UString [20]	90.2	<u>2.98</u>	<u>88.2</u>	<u>2.83</u>	<u>83.5</u>	<u>2.58</u>	<u>87.8</u>	<u>2.88</u>
	Ours	93.7	3.27	93.0	3.13	88.4	3.02	92.7	3.21
CCD (50%)	UString [20]	93.2	3.37	91.9	3.23	88.1	3.04	92.3	3.30
	DSTA [36]	<u>94.4</u>	3.41	<u>92.7</u>	3.29	<u>90.5</u>	3.02	<u>92.1</u>	3.20
	Ours	97.0	3.74	96.1	3.57	94.2	3.42	96.4	3.60
DAD (25%)	UString [20]	38.1	2.92	34.6	2.74	29.8	2.19	35.2	2.80
	DSTA [36]	48.5	2.17	46.9	2.10	43.4	1.92	<u>47.6</u>	2.13
	GSC [29]	44.9	2.13	41.5	2.04	37.4	1.87	48.9	2.10
	Ours	60.7	3.87	60.2	3.80	58.1	3.65	60.3	3.82
A3D (25%)	UString [20]	80.4	<u>2.25</u>	<u>75.3</u>	<u>2.14</u>	<u>69.6</u>	<u>1.97</u>	<u>76.4</u>	<u>2.18</u>
	Ours	89.4	3.18	86.7	3.04	84.5	2.98	87.1	3.07
CCD (25%)	UString [20]	81.1	<u>3.04</u>	76.5	<u>2.75</u>	67.4	<u>2.43</u>	77.2	<u>2.81</u>
	DSTA [36]	<u>83.6</u>	3.16	<u>77.1</u>	2.92	69.0	2.59	79.7	3.01
	Ours	94.2	3.60	92.9	3.48	88.2	3.26	93.1	3.51

indicates stronger early-warning capability under continuous missing observations, while the degradation under Burst-1.0s shows that long-duration sensor blackout remains challenging. Overall, the burst-dropout results provide a more conservative assessment of our proposed CRASH++ model under imperfect sensing conditions.

4.2.3. Evaluation under Limited Training and Missing-Frame Conditions

To evaluate robustness under limited supervision and incomplete observations, we compare CRASH++ with the strongest available baselines for each dataset. As summarized in Table 6, the models are trained using 75%, 50%, or 25% of the original training data and evaluated under both random and periodic missing-frame conditions. CRASH++ achieves the highest AP and the longest mTTA in every reported setting across DAD, A3D, and CCD. More importantly, its advantage becomes increasingly pronounced as the amount of training data decreases. When macro-averaged across the three datasets and four missing-frame patterns, the relative gains of CRASH++ in AP and mTTA metrics over the strongest available baseline increase from 10.5% and 15.1%, respectively, with 75% training data to 20.9% and 35.5% with only 25% training data. Under the most challenging combined condition, in which only 25% of the training data are retained and 50% of the test frames are missing, CRASH++ improves AP and mTTA by an average of 27.7% and 47.9%, respectively. These results suggest that CRASH++ benefits not only from improved predictive accuracy but also from more data-efficient representation learning and more stable temporal reasoning. Its consistent performance under both random frame loss and periodic frame-rate degradation further demonstrates robustness to different forms of incomplete observation. Overall, Table 6 and Fig. 5 show that CRASH++ is able to maintain reliable and timely accident anticipation under increasingly constrained real-world driving conditions.

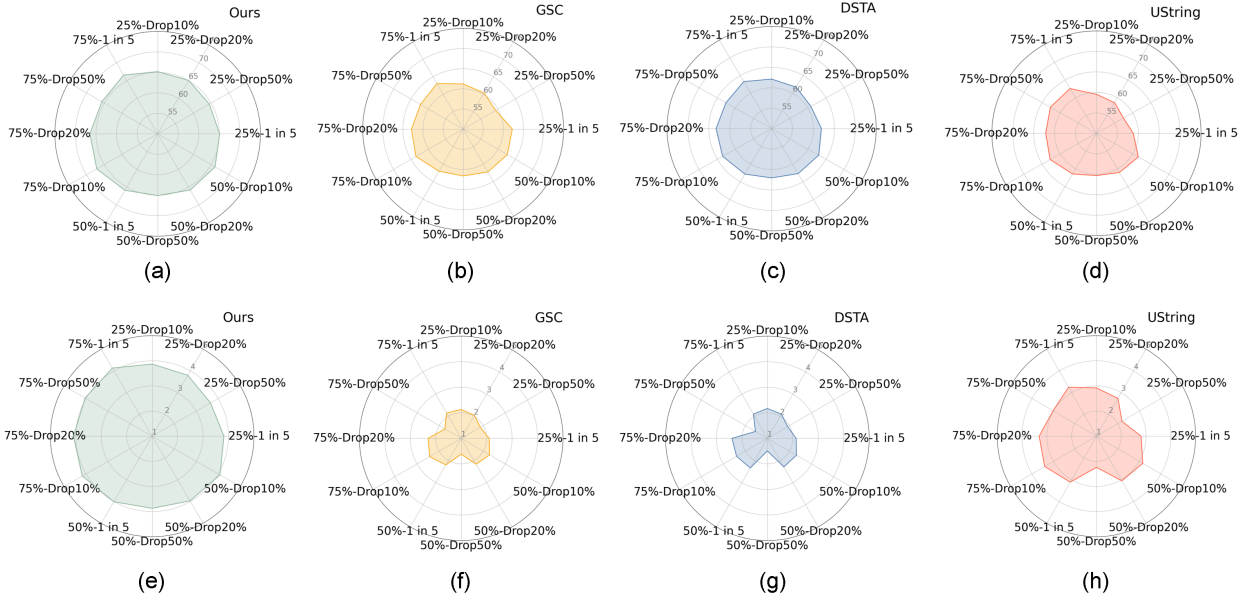


Figure 5: Best AP for our model, GSC, DSTA, and UString across diverse paradigms. Panels (a), (b), (c), and (d) report the highest AP results when training on the complete dataset and testing on missing datasets; Panels (e), (f), (g), and (h) display the best AP when these models are trained on limited training sets and evaluated on missing datasets.

4.3. Comparative Analysis of Real-time Performance

To evaluate the real-time performance of CRASH++, we benchmark it against representative SOTA methods, including GSC, DSTA, and AccNet, on a single NVIDIA A40 GPU with batch size 1. To further stress-test latency, we also evaluate a challenging DAD subset consisting of multi-agent scenes with more than 10 traffic agents under adverse conditions such as rain, fog, or night. As summarized in Table 7, CRASH++ achieves a latency of 31 ms for accident anticipation, with 19.82 GFLOPs and 2.07M parameters. This is 42.6% faster than the strongest baseline DSTA and uses 38.2% fewer parameters than the next-lightest model GSC. Adding the localization head incurs only an additional 12 ms, resulting in 43 ms for joint accident anticipation and involved-agent localization, which remains faster than the anticipation-only baselines with latencies of 54–62 ms. Overall, these results show CRASH++ maintains a lightweight online warning path while retaining the ability to generate more detailed post-event accident reports.

4.4. Deployment-Oriented System Integration

Practical deployment requires separating the safety-critical perception loop from the downstream communication modules. In CRASH++, Stages 1–3 form the on-board risk-perception backbone. These stages produce frame-wise accident probabilities, time-to-accident estimates, and agent-wise risk scores and should therefore run locally on vehicle-side computing platforms without relying on network connectivity. Since this backbone is responsible for accident anticipation and localization, it remains independent of open-ended language generation. This part can be deployed on automotive GPUs, Neural Processing Units (NPUs) and TPU-like inference units, such as NVIDIA DRIVE AGX Orin/Thor-class systems and Tesla FSD-class automotive computing platforms.

Stage 4 is treated as a soft real-time passenger-interface module rather than part of the low-level vehicle-control loop. It consumes the outputs of the perception backbone, such as accident probability, saliency cues, and agent-wise risk scores, and converts them into concise verbal alerts. In practice, it can be implemented using a lightweight MLLM when sufficient lead time is available, while template-constrained alerts can be used as a fallback for highly urgent cases to ensure deterministic wording and bounded latency. This design keeps the warning decision itself under the perception backbone, allowing the passenger-facing message to be more context-aware than a fixed warning template.

Stage 5 is designed for post-event documentation and accident reconstruction. After an accident or near-miss, the system can generate a structured report using the recorded video segment, predicted risk scores, involved-object information, saliency cues, and retrieved accident cases. This module may run as an on-board background process or off-board on an edge/cloud or fleet-management server, depending on connectivity and privacy requirements.

Table 7

Efficiency comparison of accident anticipation and localization for our model and other baselines. A dash (—) indicates that a model lacks a localization module. **Bolded** values denote the best performance in each column.

Model	Accident Anticipation			Accident Localization			Total Latency (s)
	Param (M)	GFLOPs	Latency (s)	Param (M)	GFLOPs	Latency (s)	
GSC [29]	3.342	51.07	0.062	-	-	-	0.062
DSTA [36]	4.821	44.34	0.054	-	-	-	0.054
AccNet [4]	5.532	52.42	0.061	-	-	-	0.061
Ours	2.065	19.82	0.031	0.313	2.63	0.012	0.043

Table 8

Component-wise ablation study on the DAD, A3D, and CCD datasets. The complete configuration is listed first in each dataset group, followed by variants in which one component is removed at a time. ✓ and ✗ indicate that a component is included and excluded, respectively. **Bold** values indicate the best performance in each category.

Dataset	Components						Performance		
	OFA	CAB	FFT	TFA	$\mathcal{L}_e$	SAE	AP (%)↑	mTTA (s)↑	@80 (s)↑
DAD [18]	✓	✓	✓	✓	✓	✓	72.7	4.28	4.30
	✗	✓	✓	✓	✓	✓	67.4	3.76	3.78
	✓	✗	✓	✓	✓	✓	62.3	3.11	3.58
	✓	✓	✗	✓	✓	✓	61.7	3.35	3.63
	✓	✓	✓	✗	✓	✓	68.5	3.71	4.14
	✓	✓	✓	✓	✗	✓	70.2	4.01	4.03
	✓	✓	✓	✓	✓	✗	63.9	3.25	3.32
A3D [19]	✓	✓	✓	✓	✓	✓	96.5	3.43	3.65
	✗	✓	✓	✓	✓	✓	89.3	3.19	3.24
	✓	✗	✓	✓	✓	✓	86.1	3.22	3.30
	✓	✓	✗	✓	✓	✓	91.2	3.29	3.41
	✓	✓	✓	✗	✓	✓	90.6	2.79	2.93
	✓	✓	✓	✓	✗	✓	91.4	3.32	3.45
	✓	✓	✓	✓	✓	✗	87.2	3.00	3.12
CCD [20]	✓	✓	✓	✓	✓	✓	99.7	3.94	4.07
	✗	✓	✓	✓	✓	✓	95.3	3.69	3.78
	✓	✗	✓	✓	✓	✓	92.1	3.74	3.85
	✓	✓	✗	✓	✓	✓	93.2	3.80	3.91
	✓	✓	✓	✗	✓	✓	97.0	3.77	3.88
	✓	✓	✓	✓	✗	✓	98.1	3.84	3.93
	✓	✓	✓	✓	✓	✗	93.5	3.72	3.79

To reduce unsupported generation, the report-generation stage is grounded in structured model outputs and RAG-retrieved evidence, and uncertain fields are conservatively marked as unknown rather than inferred beyond the available evidence. Therefore, the MLLM-based modules are used as controlled communication and documentation interfaces, while the safety-critical anticipation and localization tasks are handled by the real-time perception backbone.

4.5. Ablation Studies

4.5.1. Ablation Studies of Different Modules

In this section, we present the results of the ablation studies on the six core components in the CRASH++ model: Object Focus Attention (OFA), Context-aware Attention Block (CAB), Fast Fourier Transform (FFT), Temporal Focus Attention (TFA), enhancement loss $\mathcal{L}_e$, and Saliency Encoder and Saliency-aware Encoder (SAE). The impact of each component is evaluated across the DAD, A3D, and CCD datasets, and the results are summarized in Table 8.

The ablation studies reveal that each core component plays a crucial role in the CRASH++ model's performance. When all components are included, the model consistently outperforms any configuration where a component is removed. Notably, excluding CAB and FFT results in significant performance drops across all datasets. For instance,

Table 9

Performance of our model on the DADA-2000 benchmark. Arrows ($\uparrow$, $\downarrow$) indicate the expected direction of improvement for each metric. The leading scores are marked in **bold**, while the runner-up scores are underlined.

Model	KLD $\downarrow$	CC $\uparrow$	SIM $\uparrow$	NSS $\uparrow$	AUC-J $\uparrow$
DR(eye)VE [39]	2.27	0.45	0.32	2.92	0.91
BDD-A [22]	3.32	0.33	0.25	2.15	0.86
TwoStream [65]	2.85	0.23	0.14	1.48	0.84
ASIAF-Net [66]	1.66	0.49	0.36	<u>3.39</u>	0.93
SCAFNet [67]	2.19	<u>0.50</u>	0.37	3.34	<u>0.92</u>
TASEDNet [68]	1.78	0.46	0.31	3.20	<u>0.92</u>
Gate-DAP [69]	1.65	0.52	<u>0.36</u>	3.14	0.93
Ours	1.60	0.47	<u>0.36</u>	3.41	0.93

Table 10

Evaluation of different methods on the BDD-A dataset under saliency-based metrics. A larger value ($\uparrow$) corresponds to better performance, whereas a smaller value ($\downarrow$) indicates improvement for error-based measures.

Model	CC $\uparrow$	SIM $\uparrow$	KLD $\downarrow$	Params (M)	FLOPs (G)
MLNet [70]	0.61	0.43	1.20	15.45	34.46
TASED-Net [68]	0.55	0.42	1.24	21.26	91.48
CDNN [24]	0.62	0.45	1.14	0.68	7.10
SCAFNet [67]	0.64	0.48	1.09	54.51	85.86
DrFixD-rainy [23]	0.64	<u>0.47</u>	1.09	39.57	9.74
FBLNet [71]	0.64	<u>0.47</u>	1.40	87.48	20.31
SCOUT [72]	0.63	0.48	1.04	4.96	17.72
SalM ² [57]	0.64	<u>0.47</u>	<u>1.08</u>	<u>0.08</u>	<u>4.45</u>
Ours	0.64	0.48	1.04	0.06	4.21

on the DAD dataset, the removal of CAB leads to a 14.3% decrease in AP, while removing FFT reduces AP by 15.1%. These results emphasize the importance of CAB and FFT in enhancing the model's ability to capture contextual and frequency-based features, which are essential for accurate accident prediction. Particularly in complex traffic scenarios, these components help the model capture more nuanced traffic patterns, significantly boosting its predictive accuracy.

Moreover, excluding OFA or TFA mainly impacts timeliness. On average, mTTA and TTA@R80 drop by 10%-20% once these components are removed, with the largest effect on A3D. This highlights their complementary roles: OFA directs the model to accident-prone agents, while TFA emphasizes critical frames, together enabling earlier and more stable anticipation. Notably, dropping the Saliency Encoder and the Saliency-aware Encoder leads to consistent degradation across all datasets. On average, AP falls by nearly 10%, while mTTA and TTA@R80 are shortened by 10%-20%. This highlights the critical role of saliency supervision in guiding the model's attention toward human-like focus regions, which in turn strengthens both accuracy and early anticipation. Finally, the enhancement loss $\mathcal{L}_e$ contributes to improving model performance, particularly in enhancing TTA@R80. While its impact is smaller compared to other components, removing $\mathcal{L}_e$ results in a significant decline in TTA@R80, indicating that the enhancement loss helps optimize hidden states, leading to more accurate and timely predictions.

4.5.2. Case Study for Salient Encoder

To evaluate the ability of the saliency branch to identify driver-relevant regions, we conduct experiments on four driver attention benchmarks. As shown in Table 9, our model achieves leading performance on several metrics on DADA 2000. Specifically, it obtains the lowest KLD of 1.60, the highest NSS of 3.41, and a joint best AUC J of 0.93. It also achieves the second highest SIM of 0.36 and a competitive CC of 0.47. The low KLD indicates close agreement between the predicted and reference saliency distributions, while the high NSS demonstrates accurate responses at human fixation locations. Together, these results show that the model provides a favorable balance between distribution-level alignment and fixation localization. As shown in Table 10, the proposed saliency branch also achieves a favorable balance between accuracy and efficiency on BDD-A. It matches the best CC of 0.64 and SIM of 0.48 and ties the lowest

Table 11

Comparison of model performance on the TrafficGaze dataset. A dash ("-") indicates that the corresponding metric was not reported in the official release. The best results are marked in **bold**, and the second-best are underlined.

Model	AUC-B $\uparrow$	AUC-J $\uparrow$	NSS $\uparrow$	CC $\uparrow$	SIM $\uparrow$	KLD $\downarrow$
MLNet [70]	0.87	0.90	5.69	0.87	0.45	0.87
TASED-Net [68]	<u>0.92</u>	<u>0.97</u>	5.73	<u>0.94</u>	0.79	1.43
CDNN [24]	0.93	<u>0.97</u>	5.83	0.95	<u>0.78</u>	<u>0.29</u>
SCAFNet [67]	-	0.98	<u>6.10</u>	<u>0.94</u>	0.77	0.66
DrFixD-rainy [23]	<u>0.92</u>	0.98	6.01	<u>0.94</u>	<u>0.78</u>	0.28
FBLNet [71]	0.87	<u>0.97</u>	6.50	0.90	0.69	0.46
SCOUT [72]	0.91	<u>0.97</u>	5.35	0.91	0.72	0.39
Ours	0.93	0.98	5.88	<u>0.94</u>	0.79	<u>0.29</u>

Table 12

Comparison of model performance on DrFixD-rainy dataset. Evaluation metrics with $\uparrow$ indicate that higher values are better, while metrics with $\downarrow$ prefer lower values. The best results are marked in **bold**, and the second-best are underlined.

Model	AUC-B $\uparrow$	AUC-J $\uparrow$	NSS $\uparrow$	CC $\uparrow$	SIM $\uparrow$	KLD $\downarrow$
MLNet [70]	0.89	0.93	3.90	0.79	0.63	3.69
TASED-Net [68]	0.88	<u>0.95</u>	4.21	0.84	0.59	0.85
CDNN [24]	<u>0.90</u>	<u>0.95</u>	4.11	0.82	0.63	0.52
SCAFNet [67]	<u>0.90</u>	0.94	4.17	0.84	<u>0.67</u>	1.87
DrFixD-rainy [23]	0.91	0.96	4.19	0.85	<u>0.67</u>	<u>0.47</u>
FBLNet [71]	0.88	<u>0.95</u>	<u>4.29</u>	0.85	0.69	0.50
SCOUT [72]	<u>0.90</u>	<u>0.95</u>	3.95	0.84	0.65	0.50
Ours	<u>0.90</u>	0.96	4.30	0.85	0.69	0.46

KLD of 1.04, while requiring only 0.06M parameters and 4.21G FLOPs. Following the SalM² evaluation protocol [57], model complexity is measured on an NVIDIA RTX 4090. Under the same setting, our method uses 25% fewer parameters than SalM² at 0.06M compared with 0.08M and approximately 5.4% fewer FLOPs at 4.21G compared with 4.45G. It also requires approximately 83 times fewer parameters than SCOUT at 4.96M and 1,458 times fewer parameters than FBLNet at 87.48M. These complexity results refer specifically to the saliency model evaluated under the driver attention benchmark protocol rather than the complete CRASH++ pipeline. These results on TrafficGaze and DrFixD further demonstrate the generalization of the saliency branch. As reported in Table 11, our model matches the best AUC B of 0.93, AUC J of 0.98, and SIM of 0.79 on TrafficGaze while remaining competitive on the other metrics. On DrFixD rainy, Table 12 shows that the model achieves the best NSS of 4.30 and the lowest KLD of 0.46 while tying the best AUC J of 0.96, CC of 0.85, and SIM of 0.69. Collectively, these results demonstrate reliable saliency prediction across standard driving scenes and challenging conditions involving rain and reduced visibility.

4.5.3. Ablation Discussion

The ablation results in Table 8 show that the main components of CRASH++ provide complementary benefits. The context-related modules, including FFT and CAB, mainly improve contextual discrimination by capturing scene-level cues such as road layout, crosswalks, lane markings, and spatial relations between agents and infrastructure. Removing them weakens the model's ability to distinguish accident-prone scenes from visually similar safe scenes, leading to larger drops in AP. The interaction and temporal reasoning modules affect different aspects of anticipation. OFA improves agent-level risk reasoning under multi-agent interactions and therefore supports both localization and stable risk assignment. TFA strengthens historical evidence aggregation, which is especially useful when accident precursors are weak or temporally dispersed, explaining its impact on mTTA and TTA@R80. The saliency branch further helps suppress visual clutter by guiding the model toward safety-critical regions, while the enhancement loss $\mathcal{L}_e$ contributes to smoother and more stable temporal risk predictions. Overall, FFT and CAB reduce contextual ambiguity, OFA improves accident-agent reasoning, TFA enhances temporal evidence aggregation, saliency modeling focuses

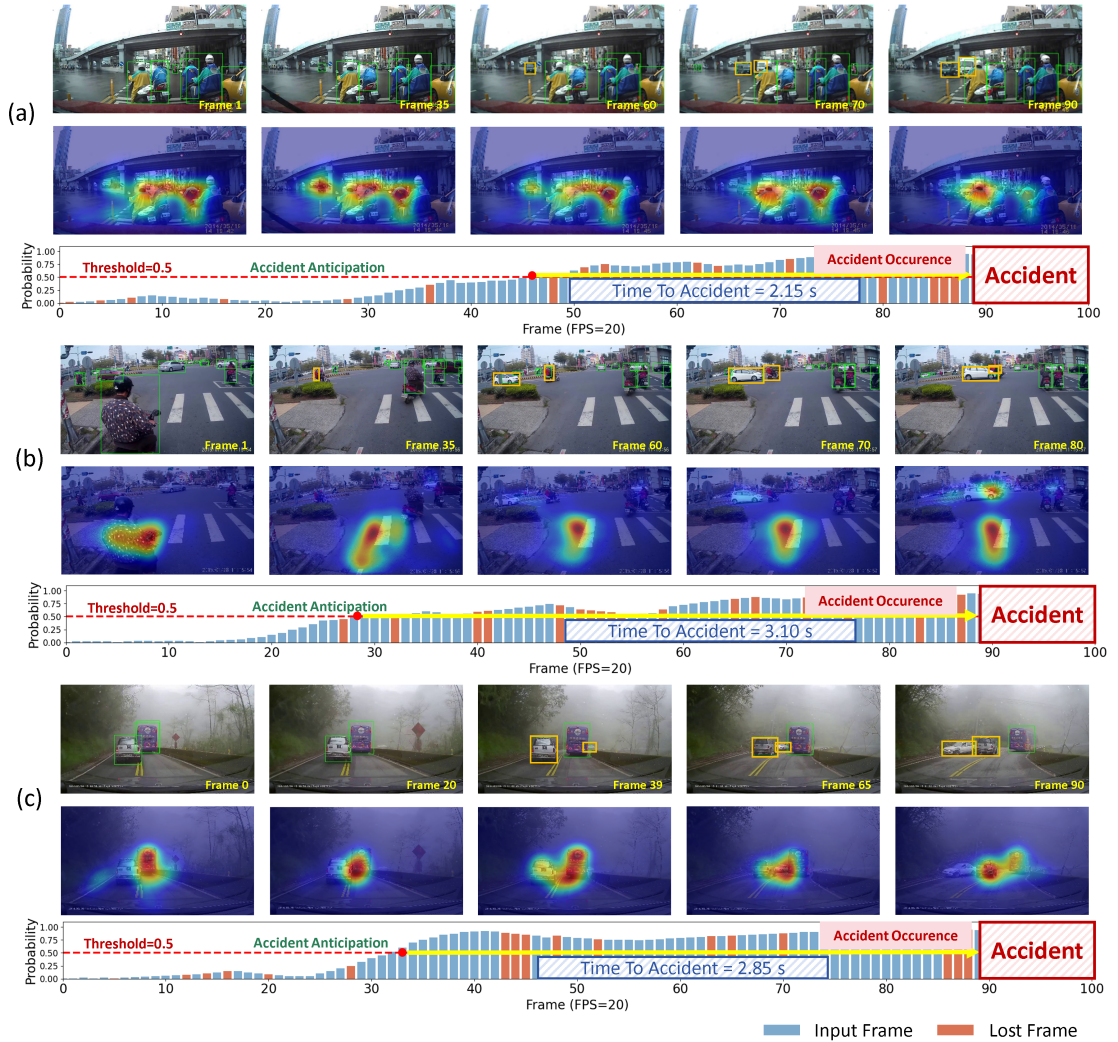


Figure 6: Qualitative results of CRASH++ in (a) rain, (b) multi-agent, and (c) heavy fog scenes. In each sequence, green boxes mark detected traffic agents and orange boxes mark the accident-involved agents localized by the model. The middle row overlays attention maps (warmer colors denote closer attention). The bottom panel plots per-frame accident probability (blue), with missing observations highlighted in orange. The red dashed line indicates the fixed decision threshold (0.5).

attention on risk-relevant regions, and $\mathcal{L}_e$ stabilizes weak precursor learning. These complementary effects explain why the full model achieves the best balance among accuracy, anticipation timeliness, and robustness.

4.6. Qualitative Results

4.6.1. Qualitative Results for Accident Anticipation and Localization

We analyze the performance of the CRASH++ system on non-accident (negative) and accident (positive) videos by conducting a comparative analysis of the experimental results on the DAD and CCD datasets. The qualitative results are illustrated in Fig. 6, Fig. 7, and Fig. 8, highlighting the model's effectiveness across different scenarios.

Fig. 6 visualizes the accident-anticipation behavior of CRASH++ in three challenging conditions: (a) rain, (b) dense multi-agent traffic at a crosswalk, and (c) heavy fog. Across all settings, the model raises risk smoothly and crosses the threshold more than two seconds before impact ($TTA \approx 2.1\text{-}3.1$ s), enabling timely warnings. In the rainy scene (a), attention concentrates early on the eventual collision pair despite reflections and motion blur. In the multi-agent scene (b), the model suppresses distractors and allocates attention to the interacting vehicles near the crosswalk, correctly localizing the accident-involved agents. In heavy fog (c), where contrast is low and visibility is limited, the

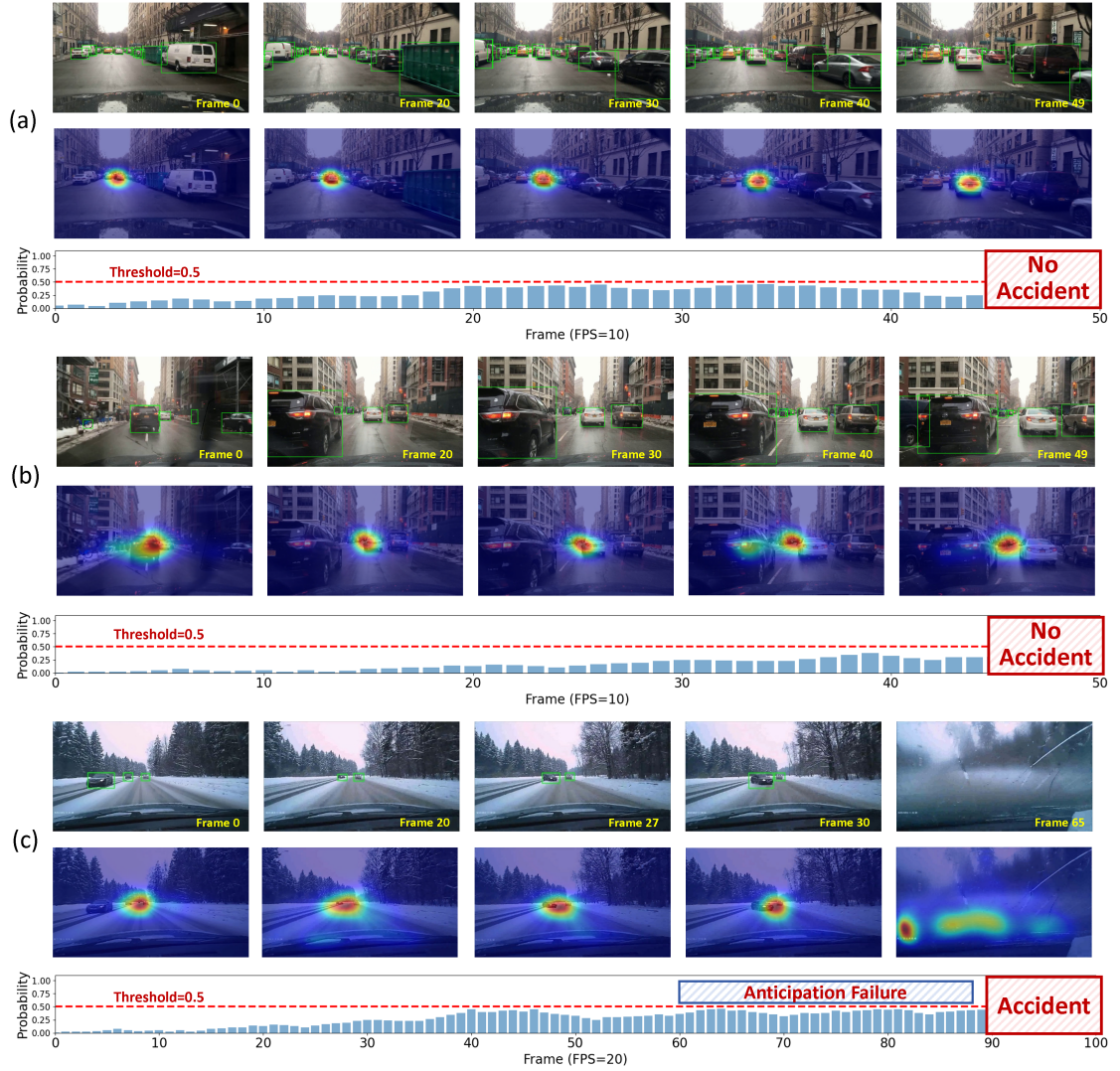


Figure 7: Qualitative results of CRASH++ across three scenarios: (a) dense urban traffic, (b) rainy urban conditions, and (c) snowy weather. Sequences (a)-(b) contain no accidents; sequence (c) is a failure case in which an accident occurs but is not anticipated in time. In each sequence, the top row shows RGB frames with detected agents (green boxes), the middle row overlays attention maps (warmer colors denote closer attention), and the bottom panel plots per-frame accident probability (blue bars). The red dashed line indicates the fixed decision threshold (0.5).

model still tracks risk cues and maintains an early lead time. Notably, these results hold even under partial observations and reduced supervision: the probability curves remain stable with up to 20% frame dropouts, and comparable behavior is observed when the model is trained on only 50% of the original training set.

Moreover, Fig. 7 presents qualitative results on non-accident sequences and a failure case. In Fig. 7(a)-(b), vehicles weave through a rainy, congested urban street. Although nearby agents make repeated approaches toward the ego vehicle, the frame-wise accident probability (bottom blue bars) consistently remains below the 0.5 decision threshold (red dashed line). The attention maps concentrate on the most conflict-prone agents and shift as headway changes; as gaps close, the score increases, and as vehicles separate, it recedes. These patterns indicate that the model remains stable in complex scenes, avoids false alarms, and allocates attention to the correct regions even under adverse weather. Notably, Fig. 7(c) indicates a failure case of our model in snowy conditions: prolonged visual occlusion, such as spray or blockage on the windshield, suppresses informative cues, leading to an anticipation failure despite an ensuing crash. This case underscores the need for stronger occlusion handling like cross-sensor fusion (radar/LiDAR), temporal

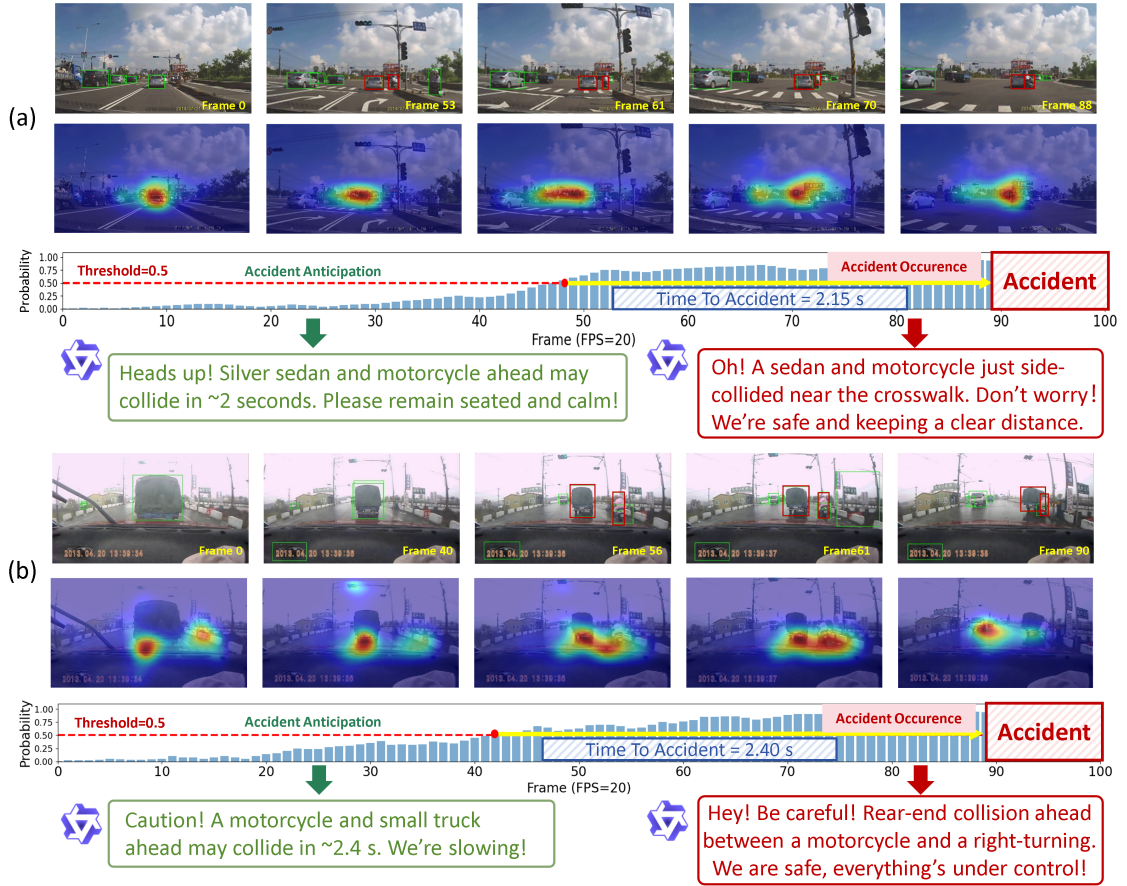


Figure 8: Visualization of human-centered feedback produced by CRASH++. Green boxes mark detected agents; red boxes highlight accident-involved agents. For each sequence, the middle row overlays attention maps (warmer colors denote higher attention), and the bottom panel plots per-frame accident probability with a fixed threshold (0.5) and the predicted TTA. Speech callouts illustrate the spoken output. (a) Zebra-crossing scene: anticipated side collision between a silver sedan and a motorcycle (TTA ≈ 2.15 s); the system issues a pre-incident alert and a post-incident reassurance. (b) Urban mixed traffic: anticipated rear-end collision between a motorcycle and a right-turning light truck (TTA ≈ 2.40 s).

imputation, or memory-based priors. Collectively, these results demonstrate CRASH++’s capabilities in temporal anticipation, spatial grounding, and agent localization, while maintaining robust performance under adverse weather, crowded scenes, and observation-missing driving conditions.

4.6.2. Qualitative Results for Human-centered Feedback

Fig. 8 and Fig. 9 illustrate the human-centered feedback produced by CRASH++ in multi-agent, rainy, and low-light night scenes. In all cases, the system issues early, concise alerts (TTA ≈ 2.1 – 2.4 s) and adjusts tone, length, and content to the detected risk and urgency, aiming to inform without alarming and to reduce opacity in algorithmic decisions. In Fig. 8 (a), at a zebra crossing with mixed traffic, the model anticipates a side collision between a silver sedan and a motorcycle (TTA ≈ 2.15 s). It first provides a calm cautionary prompt and, after impact, follows with a short reassurance that we are stopped at a safe distance. In Fig. 8(b), under urban rain with many agents, the system forecasts a rear-end crash between a motorcycle and a right-turning light truck (TTA = 2.40 s), prompting smooth braking and then confirming post-incident safety. The attention maps in both sequences concentrate on the interacting agents, and the probability traces rise smoothly past the 0.5 threshold before impact. Correspondingly, Fig. 9 demonstrates operation in low-light night scenes. In panel (a), the model detects a fast cut-in by a black sedan from the front-left and predicts a side impact with a white car (TTA = 2.35 s); the alert escalates from a neutral “heads-up” to a clear warning as risk increases, followed by a brief, reassuring status update after the collision. In panel (b), a right-turning truck crosses

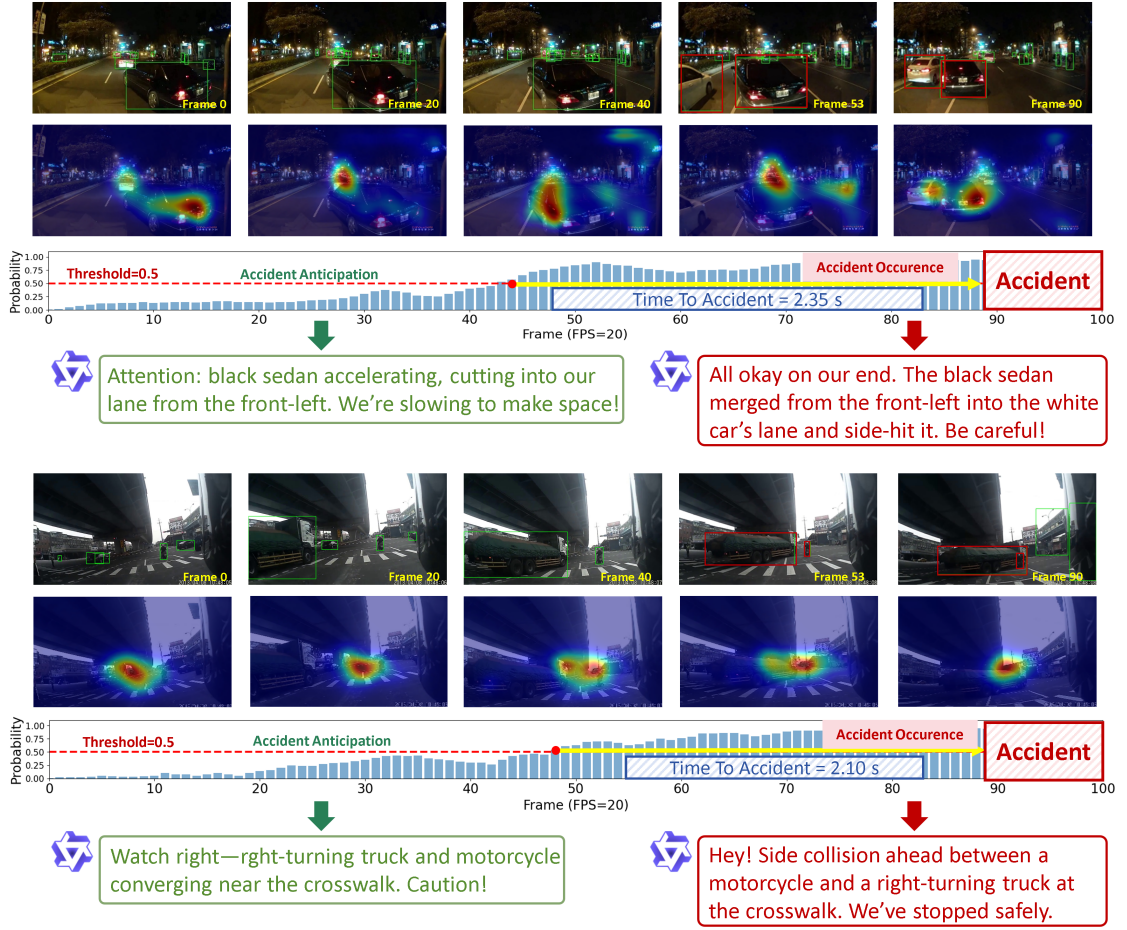


Figure 9: Visualization of CRASH++ human-centered feedback in low-light scenes. Green boxes mark detected agents, and red boxes highlight accident-involved agents. (a) Sudden cut-in scene: a black sedan accelerates from the front-left, merges into the adjacent lane, and side-collides with a white car; the system issues a pre-incident alert and a post-incident reassurance. (b) Right-turn conflict scenes: a truck turns across a through-moving motorcycle near the crosswalk, leading to a side impact; voice prompts guide safe deceleration and communicate collision risk.

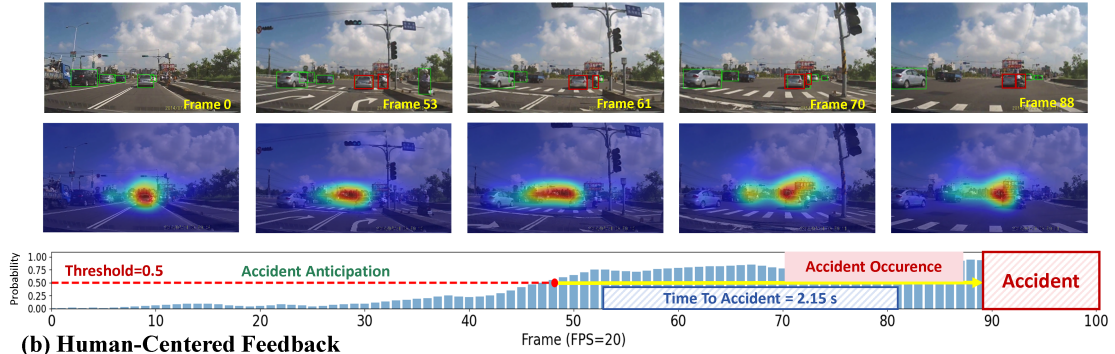
the path of a through-moving motorcycle near a crosswalk ($TTA = 2.10$ s); the model issues a targeted “watch right” prompt and then confirms that we are stationary and safe.

Furthermore, across all visualizations in Fig. 8 and Fig. 9, our model is able to provide the alerts that are short (typically fewer than 18 words) and delivered with low latency, enabling actionable guidance within a 2-3 s anticipation window. Prompts encode what may happen, when it may happen (imminence), and where/who it may affect (spatial reference and agents). After impact, the tone of the messages deliberately shifts to a calm, reassuring one to ease passenger anxiety while confirming safety status and next steps like “stopped safely” and “holding distance”.

4.6.3. Qualitative Results for End-to-End Performances

Fig. 10 presents the end-to-end outputs of CRASH++ on a representative case, linking accident anticipation and localization, human-centered feedback, and accident report generation. To begin with, the model detects a developing interaction between a silver sedan and a motorcycle at a signalized crosswalk. Detected agents are shown in green, and the accident-involved agents in red; attention maps highlight the risk region. The frame-wise probability rises smoothly and crosses the 0.5 threshold with a lead time of $TTA \approx 2.15$ s, providing sufficient margin for intervention. Then, when the probability first exceeds the threshold, the Qwen2-VL-2B module issues a concise spoken alert that specifies who and where (i.e. “silver sedan and motorcycle ahead”), while advising passengers to remain seated.

(a) Accident Anticipation & Localization



(b) Human-Centered Feedback

Real-Time Human-Centered Feedback with Qwen2-VL-2B	
Heads up! Silver sedan and motorcycle ahead may collide in about 2 seconds. Please remain seated and calm!	

(c) Accident Report Generation

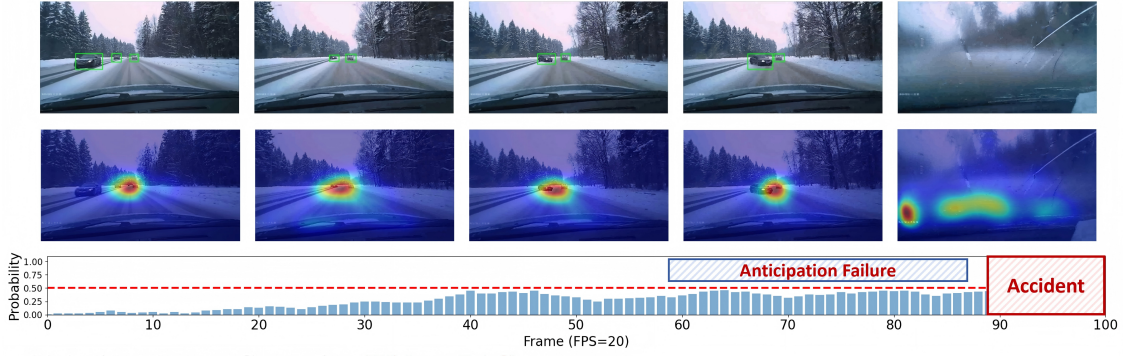
Accident Report	Vehicle	Environment	Roadway	Collision
1. Accident Number	Vehicle Information	Weather	Roadway Condition	Collision Type
2. Time of Accident	1. Manufacturer ☒	1. Clear ☒	1. Usual ☒	1. Head-On ☐
3. Vehicle Information	2. Vehicle Year ☒	2. Cloudy ☐	2. Holes ☐	2. Side Swipe ☐
4. Movement	3. Vehicle Mode ☒	3. Rainy ☐	3. Obstruction ☐	3. Rear End ☐
5. Weather	Movement	4. Snowing ☐	4. Reduced Width ☐	4. Broadside ☒
6. Lighting Condition	1. Stopped ☐	5. Fog ☐	5. Flooded ☐	5. Hit Object ☐
7. Roadway Condition	2. Straight ☐	6. Windy ☐	6. Construction ☐	6. Overturned ☐
8. Roadway Surface	3. Ran Off Road ☐	7. Stormy ☐	7. Cracked ☐	7. Veh/Pedestrian ☐
9. Collision Type	4. Right Turn ☒	Lighting Condition	Roadway Surface	8. Other ☐
10. Outcomes	5. Left Turn ☐	1. Daylight ☒	1. Dry ☒	Outcomes
	6. Backing ☐	2. Dusk-Dawn ☐	2. Wet ☐	1. PDO ☐
	7. Slowing ☐	3. Dark-Light ☐	3. Snowy-Icy ☐	2. Injury ☒
	8. Merging ☐	4. Dark-No Light ☐	4. Muddy ☐	3. Fatalities ☐

RAG-Grounded Accident Report with Qwen2-VL-7B	
This accident occurred during daylight under clear weather conditions on a roadway with usual conditions and a dry surface . Unit 1 (silver vehicle) initiated a right turn at an intersection and entered the path of Unit 2 (motorcycle) traveling straight through. The collision type was classified as a broadside crash (side-impact). As a consequence of the collision, injuries were reported, though no fatalities occurred.	

Figure 10: Visualization of CRASH++. (a) The model anticipates a collision between a silver sedan and a motorcycle about 2.15 s before impact, with saliency maps highlighting the high-risk regions. (b) Qwen2-VL-2B converts the predicted risk into a concise passenger-facing alert. (c) Qwen2-VL-7B with retrieval augmentation generates a structured accident report, describing the maneuver, weather and lighting conditions, collision type, and injury outcome.

Finally, to explain *why* the crash occurs, Qwen2-VL-7B generates a structured report using retrieval-augmented generation. The pipeline proceeds in three steps: (i) the model composes an initial scene anticipation from the video, attention features, and prediction trace (agents, maneuvers, likely crash type); (ii) this anticipation is used to retrieve top-*k* semantically similar cases from an external knowledge base of officially documented accidents (California DMV), indexed with dense embeddings; and (iii) an augmented prompt fuses the scene summary, retrieved evidence, and a standardized schema to produce the final report. The schema follows four categories—vehicle, environment, roadway, and collision—and yields checkbox-style fields and a short narrative. In this example, the report identifies a right-turn maneuver by Unit 1 (silver vehicle), clear daylight and a dry, usual roadway, a broadside (side-impact) configuration, and injuries without fatalities. Grounding generation in retrieved cases and time-stamped visual evidence improves factual consistency, reduces hallucinations, and ensures compatibility with traffic-safety documentation.

(a) Failed Accident Anticipation under Snowy Occlusion



(b) Accident Report Generation (Without RAG)

	Accident Report without RAG with Qwen2-VL-7B
This sequence occurred during daylight under snowy weather conditions on a snowy-icy roadway. The lead vehicle ahead appears to be traveling straight. Due to the low accident probability and severe windshield obstruction in the late frames, no confirmed collision is identified . The event is reported as a visibility-related hazard rather than an accident , leaving the collision type and outcome unreported.	

(c) Accident Report Generation (With RAG)

Accident Report	Vehicle	Environment	Roadway	Collision
1. Accident Number	Vehicle Information	Weather	Roadway Condition	Collision Type
2. Time of Accident	1. Manufacturer ☒	1. Clear ☐	1. Usual ☒	1. Head-On ☐
Vehicle	2. Vehicle Year ☒	2. Cloudy ☐	2. Holes ☐	2. Side Swipe ☐
3. Vehicle Information	3. Vehicle Mode ☒	3. Rainy ☐	3. Obstruction ☐	3. Rear End ☐
4. Movement	Movement	4. Snowing ☒	4. Reduced Width ☐	4. Broadside ☐
Environment	1. Stopped ☐	5. Fog ☐	5. Flooded ☐	5. Hit Object ☐
5. Weather	2. Straight ☒	6. Windy ☐	6. Construction ☐	6. Overturned ☐
6. Lighting Condition	3. Ran Off Road ☐	7. Stormy ☐	7. Cracked ☐	7. Veh/Pedestrian ☐
Roadway	4. Right Turn ☐	Lighting Condition	Roadway Surface	8. Other ☒
7. Roadway Condition	5. Left Turn ☐	1. Daylight ☒	1. Dry ☐	Outcomes
8. Roadway Surface	6. Backing ☐	2. Dusk-Dawn ☐	2. Wet ☐	1. PDO ☐
Collision	7. Slowing ☐	3. Dark-Light ☐	3. Snowy-Icy ☒	2. Injury ☒
9. Collision Type	8. Merging ☐	4. Dark-No Light ☐	4. Muddy ☐	3. Fatalities ☐
10. Outcomes				

	RAG-Grounded Accident Report with Qwen2-VL-7B
This accident likely occurred during daylight under snowy weather conditions on a snowy-icy roadway . Unit 1 (dark lead vehicle ahead) and Unit 2 (another partially visible lead vehicle) were moving straight before visibility became severely degraded by snow spray and windshield blockage. Although the upstream accident probability remains below the warning threshold, the sampled video frames and retrieved snowy-road accident records suggest a possible forward-road conflict between Unit 1 and Unit 2 under reduced visibility and low road traction. The collision type is classified as other/uncertain vehicle conflict, and property damage is possible, while the injury outcome cannot be reliably determined from the video alone.	

Figure 11: Failure case for report generation under imperfect upstream anticipation. (a) In this snowy-weather case, snow spray and windshield occlusion prevent the anticipation module from triggering before the crash. (b) Without RAG, the MLLM produces an incomplete visibility-related description. (c) With RAG, retrieved snowy-road records and multimodal evidence help generate a better-grounded report, although severe upstream errors may still affect the final output.

Overall, these results show that CRASH++ reliably assesses accident risk in complex scenes by tracking interactions among road users and their context. The model localizes the agents likely to be involved and issues threshold-crossing alerts more than two seconds before impact, supporting timely intervention and demonstrating predictive validity under adverse weather and partial observations. Guided by structured prompts, the feedback module fuses visual cues with semantic context to deliver brief, passenger-facing messages that specify *who*, *where*, and *when* risk

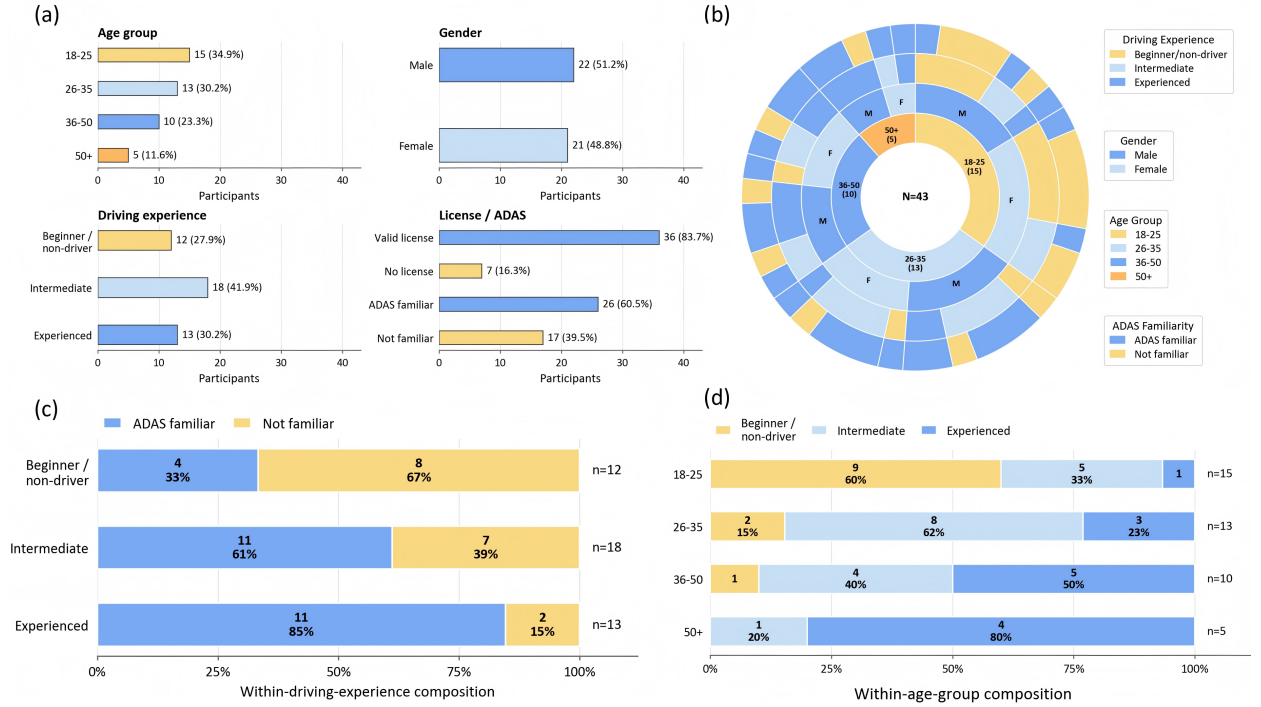


Figure 12: Participant demographics and cross-dimensional composition of the user study ($N = 43$). (a) Marginal distributions of participants by age group, gender, driving experience, valid driving license, and familiarity with ADAS functions. (b) Nested donut chart illustrating the hierarchical distribution across four dimensions: Age Group (inner ring), Gender (second ring), Driving Experience (third ring), and ADAS Familiarity (outer ring). (c) Distribution of ADAS familiarity within each driving-experience group, with each row normalized to 100%. (d) Distribution of driving experience within each age group, with each row normalized to 100%. Counts and percentages are shown in the corresponding bars.

arises, then shifts to a calm, reassuring tone after an incident. Finally, the accident-report stage synthesizes retrieval-grounded, structured reports aligned with standard fields (vehicle, environment, roadway, collision), explaining *why* the crash occurred and providing traceable, interoperable documentation for downstream safety analysis.

4.6.4. Qualitative Results of Accident Report Generation under Upstream Recognition Failure

To further examine the reliability of the report generation module, we analyze a challenging failure case in which the upstream accident anticipation and localization outputs are imperfect. As shown in Fig. 11(a), the sequence is captured under snowy conditions, where snow spray and windshield occlusion severely degrade the visual evidence in the late frames. As a result, the accident probability remains below the warning threshold before the crash, leading to an anticipation failure. As shown in Fig. 11(b), without RAG, the generated report is strongly influenced by the low upstream accident probability. Although the video sequence eventually contains an accident, the report describes it as a visibility-related hazard rather than a confirmed collision, and leaves the collision type and outcome unspecified. This suggests that directly relying on anticipation and localization outputs may cause upstream errors to propagate into the final textual report, especially when the model treats low accident probability as strong negative evidence.

By contrast, As shown in Fig. 11(c), the RAG-grounded report treats accident probabilities and agent-wise risk scores as soft evidence rather than deterministic facts. Before filling the structured report fields, the generator cross-checks these signals with sampled video frames, saliency cues, object-level evidence, and retrieved snowy-road accident records. In this case, the RAG-grounded report correctly identifies the snowy daytime environment, snowy-icy road surface, and straight movement of the visible vehicles. Meanwhile, it avoids over-specifying uncertain details: the collision is conservatively described as an other/uncertain vehicle conflict, and the injury outcome is marked as not reliably determinable from the video alone. This comparison indicates that RAG can improve grounding and reduce unsupported claims under moderate upstream noise. However, it also reveals an important limitation: when

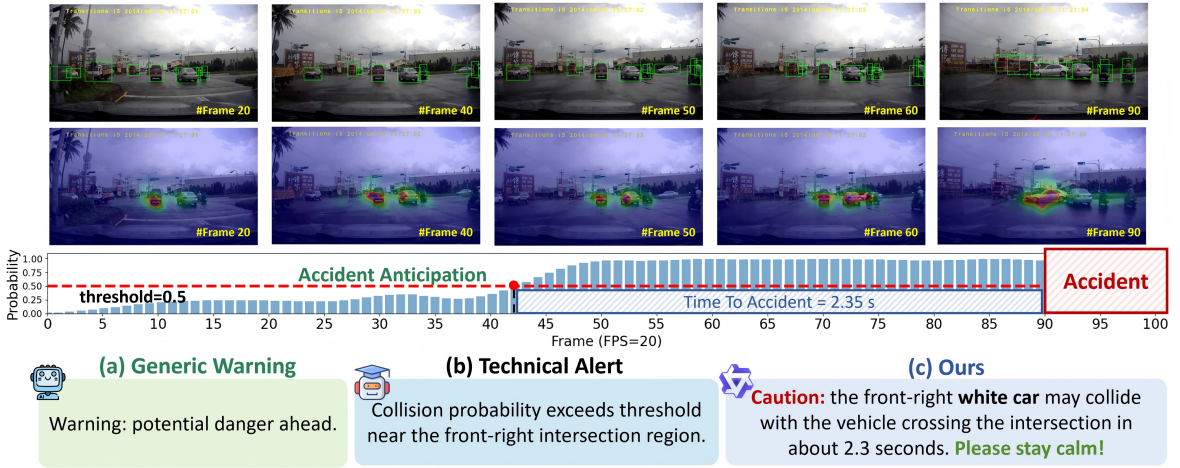


Figure 13: Comparison of three passenger-facing feedback strategies in a rainy intersection scenario. The top part shows representative frames, saliency cues, and the accident probability curve, where the warning threshold is crossed about 2.35 seconds before the accident. The bottom part shows alerts triggered at the same time: (a) a generic warning gives only a coarse danger notice; (b) a technical alert reports the threshold-crossing risk region; and (c) the CRASH++'s human-centered alert identifies the likely conflicting agents and time window, and provides a concise, calming message.

the anticipation module fails severely or the key involved objects are heavily occluded, errors may still affect the final report, especially in the estimated collision configuration and outcome description.

4.7. User Study

Standard anticipation metrics, such as AP, AUC, mTTA, and TTA@R80, evaluate prediction accuracy and warning timeliness, but do not directly assess the quality of user-facing communication. To complement these benchmark results, we conduct a user-centered evaluation of two downstream interfaces in CRASH++: real-time passenger alerts and structured accident reports. For alerts, we compare CRASH++ with alternative feedback strategies and baseline model-based alerts, evaluating clarity, trust, comfort, annoyance, preference, and situation awareness. For report generation, we compare MLLM-only, CRASH++ without RAG, and CRASH++ with RAG to assess factual correctness, evidence support, completeness, readability, and practical usefulness.

4.7.1. Participant Demographics

We recruit $N = 43$ participants to evaluate the human-centered feedback and report-generation components of CRASH++. The participants cover different age groups, gender groups, driving-experience levels, valid driving-license status, and ADAS familiarity, as summarized in Fig. 12. This diversity helps collect feedback from both novice passengers and experienced drivers with different levels of exposure to ADAS functions. Detailed demographic statistics and cross-dimensional distributions are provided in **Appendix D**.

4.7.2. User Study for Real-Time Accident Alerts

User Study Protocol. We conduct a user study with 43 participants to evaluate the perceived quality of passenger alerts generated by the human-centered feedback module. This user study comprises an evaluation of feedback strategies and a comparison of model-generated alerts. Participants view short dashcam clips representing accident, near-miss, and normal driving scenarios and imagine themselves as passengers in an autonomous vehicle. Under each condition, each participant evaluates three clips, yielding 129 ratings for each condition and metric. Within each feedback condition, the order of the three clips is randomized independently for each participant, whereas the order of the feedback conditions is counterbalanced across participants. Participants rate each clip on a five-point Likert scale from 1 to 5, with higher scores indicating more favorable evaluations for all metrics except annoyance. After each clip, participants answer two situation-awareness questions that require them to identify the risky road user and its location.

Feedback Strategy Comparison. We first compare four feedback strategies: no feedback, a generic warning, a technical alert, and the proposed CRASH++ human-centered alert. To focus on the effect of feedback style, all

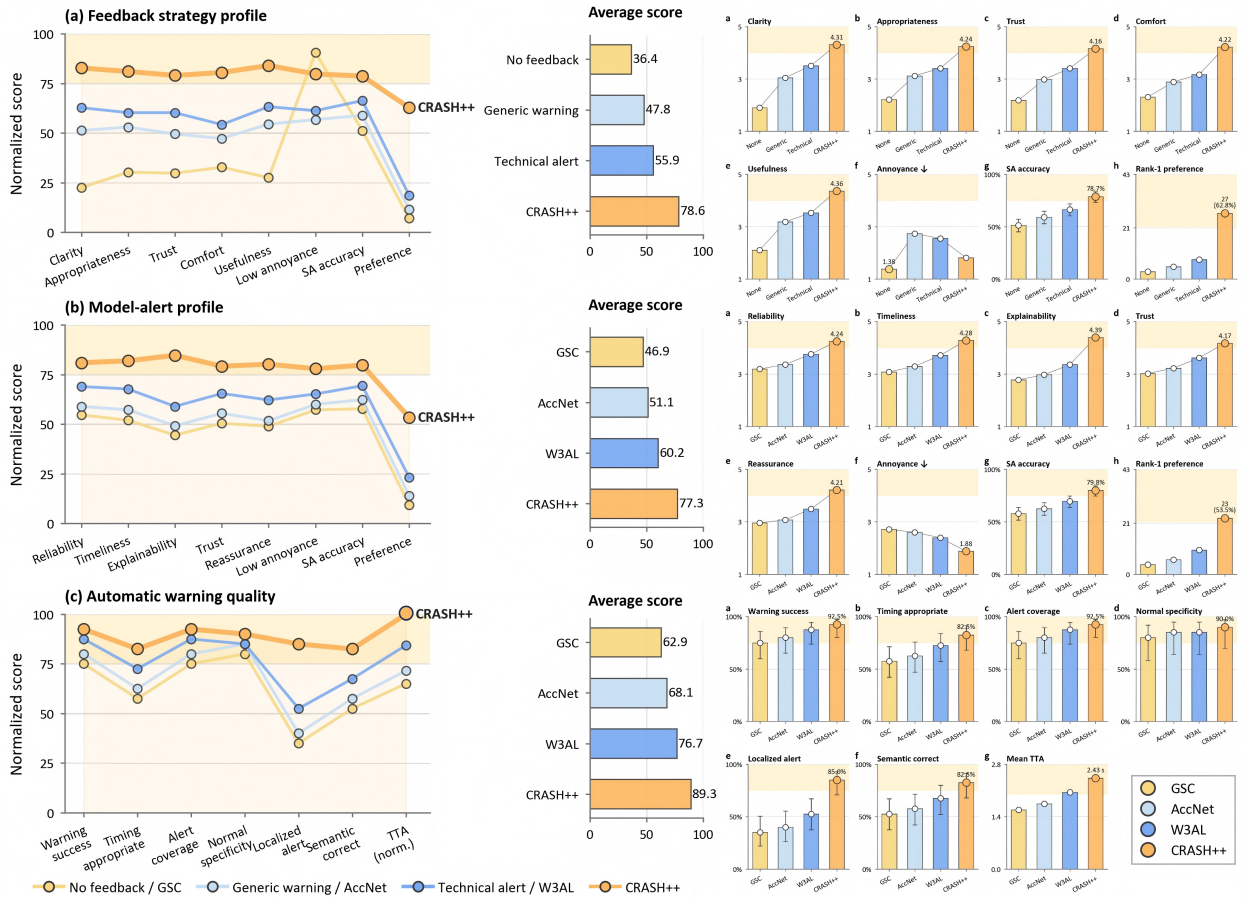


Figure 14: User-centered evaluation of human-centered feedback. Left: the normalized user-study profiles compare feedback strategies, model-based alerts, and automatic warning quality. Our proposed CRASH++ model consistently forms the strongest profile, while low-annoyance scores are normalized so that higher values indicate less intrusive feedback. Middle: composite scores summarize the overall performance of each strategy or model. CRASH++ achieves the highest composite scores in feedback strategy evaluation, model-based alert comparison, and automatic warning quality. Right: detailed metric-level results show the raw 1-5 Likert scores for subjective ratings, situation-awareness accuracy, rank-1 preference counts, and automatic warning metrics. CRASH++ improves clarity, reliability, timeliness, explainability, reassurance, and situation awareness over generic feedback, technical alerts, and model-based baselines (GSC, AccNet, W3AL).

warning-based strategies are triggered at the same timestamp. Fig. 13 illustrates a representative rainy-intersection case. The generic warning only provides a coarse danger notice; the technical alert reports the threshold-crossing risk region, while the CRASH++ alert identifies the likely conflicting agents and provides a concise calming message. This example highlights the difference between simply notifying of danger and communicating risk in a passenger-oriented manner. As shown in Fig. 14(a), CRASH++ receives higher ratings in clarity, appropriateness, trust, comfort, and usefulness than the other feedback strategies. It obtains scores of 4.31, 4.24, 4.16, 4.22, and 4.36 on these five metrics, respectively. No feedback has the lowest annoyance score, as expected, but it provides limited situational support, with a situation-awareness accuracy of 51.2%. CRASH++ reaches 78.7%, compared with 58.9% for generic warnings and 66.3% for technical alerts. This suggests that the proposed feedback helps participants better identify the risky object and its location while maintaining a relatively low annoyance score. In the final preference question, 27 out of 43 participants select our proposed CRASH++ model as their preferred feedback strategy.

Baseline Alert Comparison. We compare CRASH++ with SOTAs, including GSC, AccNet, and W3AL [9]. Since these baselines do not directly generate human-centered verbal feedback, we convert their predicted accident scores into standardized alerts for user-facing comparison. The evaluation covers reliability, timeliness, explainability, trust, reassurance, annoyance, situation-awareness accuracy (SA accuracy), and preference. As shown in Fig. 14(b),

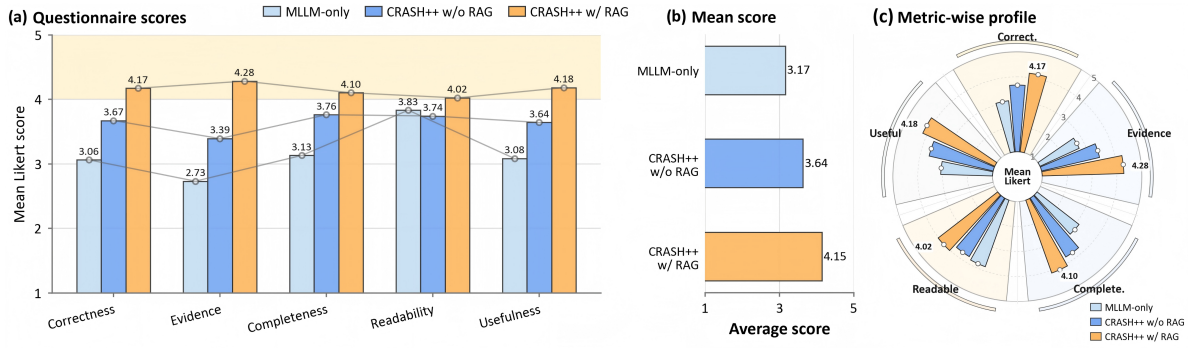


Figure 15: Evaluation of accident report generation for three strategies. (a) Mean questionnaire scores on a 1-5 Likert scale for correctness, evidence support, completeness, readability, and practical usefulness. (b) Average score across the five questionnaire metrics. (c) Metric-wise profile of the three report-generation settings. MLLM-only uses the video and prompt as input; CRASH++ w/o RAG further incorporates accident probability and agent-wise risk scores, and CRASH++ w/ RAG additionally uses retrieved accident reports and a structured prompt. CRASH++ w/ RAG obtains higher scores on most metrics, especially evidence support and factual correctness, while MLLM-only remains competitive in readability.

CRASH++ obtains higher ratings across the evaluated dimensions, with scores of 4.24 in reliability, 4.28 in timeliness, 4.39 in explainability, 4.17 in trust, and 4.21 in reassurance. The difference is most visible in explainability, where CRASH++ benefits from using accident probability, time-to-accident estimates, saliency cues, and agent-wise risk scores to form more specific alerts. CRASH++ also achieves a situation-awareness accuracy of 79.8%, compared with 69.4% for W3AL, 62.4% for AccNet, and 57.8% for GSC. In the preference ranking, CRASH++ also receives the highest rank-1 preference count, with 23 out of 43 participants selecting it as the preferred model-based alert.

Automatic Warning Quality Analysis. In addition to subjective user ratings, we perform an automatic warning-quality check on a separate set of driving clips, including accident, near-miss, and normal-driving scenarios. This evaluation examines whether the generated warning is triggered successfully, whether the timing is appropriate, whether the alert covers the relevant risk, whether normal scenes are correctly handled, and whether the warning is localized and semantically consistent with the visual evidence. As shown in Fig. 14(c), CRASH++ successfully triggers warnings before accidents in 92.5% of accident clips, achieves 82.5% timing appropriateness, and reaches 85.0% localized-alert accuracy and 82.5% semantic correctness. It also provides the longest mean time-to-accident among the compared methods, 2.43 seconds, indicating that its alerts are not only more interpretable but also sufficiently early. These automatic results explain the user-study findings: participants prefer CRASH++ because its warnings are more specific, better grounded in risky agents and locations, and less likely to be perceived as vague or intrusive.

4.7.3. User Study for Accident Report Generation

As shown in Fig. 15, we select 30 accident clips and generate one report for each clip under each setting, resulting in 90 reports in total. Three annotators independently rate all generated reports on a 1 to 5 Likert scale. No consensus procedure is used to reconcile differences among their ratings, and no individual rating is discarded or modified. Instead, all independent ratings are retained, and the final score for each method and evaluation metric is calculated as the mean of the 90 ratings obtained from 30 reports and three annotators. The evaluation covers five dimensions: factual correctness, evidence support, completeness, readability, and practical usefulness. Specifically, CRASH++ w/ RAG achieves the highest average score across the five metrics. Compared with MLLM-only, it improves factual correctness from 3.06 to 4.17 and evidence support from 2.73 to 4.28. CRASH++ w/o RAG also improves correctness and completeness, suggesting that accident probabilities and agent-wise risk scores provide useful structure for identifying relevant agents and describing the event process. MLLM-only remains competitive in readability, but its lower correctness and evidence-support scores indicate weaker grounding. These results suggest that structured risk outputs improve report informativeness, while RAG further helps ground the generated reports in retrieved evidence. This trend is also consistent with the qualitative failure case in Fig. 11. When upstream anticipation is unreliable under snowy occlusion, the report without RAG tends to follow the low accident probability and produces an incomplete visibility-related description. With RAG, retrieved snowy-road records and multimodal evidence help the model generate a more cautious and better-grounded report, although severe upstream errors may still affect uncertain fields. Overall, these

results indicate that CRASH++ benefits from both structured risk cues and retrieval grounding, producing reports that are more accurate, better supported, and more useful for post-accident analysis.

5. Conclusion

This study presented CRASH++, a unified framework that extends accident anticipation from an isolated prediction task into a broader autonomous-driving safety pipeline. Its risk-perception backbone jointly models road-user interactions, visual saliency, global scene context, and temporal evidence to estimate frame-wise accident risk and identify the agents most likely to be involved. These structured predictions are further translated into concise passenger alerts and retrieval-grounded accident reports, while the decision to trigger a safety-critical warning remains under the control of the risk-perception model. Experiments on three accident datasets and four driver-attention benchmarks show that CRASH++ achieves state-of-the-art accident anticipation and competitive driver-attention prediction, while maintaining reliable performance under missing-frame conditions and limited training data. The localization and saliency results further demonstrate its ability to associate scene-level risk with relevant road users and visual regions. In addition, the user study indicates that the generated alerts improve perceived clarity, trust, comfort, usefulness, and situation awareness. Independent assessment of the generated reports also supports their factual correctness, evidence grounding, completeness, readability, and practical usefulness. Together, these findings demonstrate the value of connecting accurate risk perception with interpretable communication and structured post-event documentation. Several limitations provide directions for future research. The current passenger evaluation is conducted in a controlled video-based setting, and future studies should examine in-vehicle or simulator-based interactions involving realistic warning timing, voice delivery, user workload, and repeated exposure. Report quality may also be affected by upstream perception errors and incomplete retrieved evidence. Future work will therefore investigate uncertainty-aware generation, stronger cross-modal consistency checks, and adaptive retrieval. Further optimization of model size, latency, and multisensor robustness will also be required for deployment in open-world driving environments. Overall, CRASH++ provides a practical step toward accident anticipation systems that not only identify emerging risks but also communicate and document them in a transparent and human-centered manner.

CRedit authorship contribution statement

Haicheng Liao: Conceptualization, Methodology, Experiment, Writing. **Chengyue Wang:** Methodology, Experiment, Writing. **Bin Rao:** Methodology, Experiment. **Jiaxun Zhang:** Methodology, Experiment. **Haoyu Sun:** Methodology, Experiment. **Zhenning Li:** Conceptualization, Methodology, Funding. **Chengzhong Xu:** Methodology, Funding. **Hai Yang:** Methodology, Review. **Keqiang Li:** Methodology, Review.

Acknowledgements

This work was supported by the Science and Technology Development Fund of Macau [0007/2025/RIC, 0122/2024/RIB2, 0215/2024/AGJ, 0074/2025/AMJ, 001/2024/SKL, 0002/2025/EQP], the Research Services and Knowledge Transfer Office, University of Macau [SRG2023-00037-IOTSC, MYRG-GRG2024-00284-IOTSC], the Shenzhen-Hong Kong-Macau Science and Technology Program Category C [SGDX20230821095159012], the Science and Technology Planning Project of Guangdong [2025A0505010016], National Natural Science Foundation of China [52572354], the State Key Lab of Intelligent Transportation System [2024-B001], and the Jiangsu Provincial Science and Technology Program [BZ2024055].

References

- [1] World Health Organization, Road deaths fall by 21% globally but stronger action is needed to save lives, 2026. URL: <https://www.who.int/news/item/20-07-2026-road-deaths-fall-by-21--globally-but-stronger-action-is-needed-to-save-lives>, accessed 20 August 2026.
- [2] B. Noh, H. Yeo, A novel method of predictive collision risk area estimation for proactive pedestrian accident prevention system in urban surveillance infrastructure, Transportation research part C: emerging technologies 137 (2022) 103570.
- [3] S. Chen, T. U. Saeed, S. Labi, Impact of road-surface condition on rural highway safety: A multivariate random parameters negative binomial approach, Analytic methods in accident research 16 (2017) 75–89.
- [4] H. Liao, Y. Li, Z. Li, Z. Bian, J. Lee, Z. Cui, G. Zhang, C. Xu, Real-time accident anticipation for autonomous driving through monocular depth-enhanced 3d modeling, Accident Analysis & Prevention 207 (2024) 107760.

- [5] Z. Li, H. Liao, R. Tang, G. Li, Y. Li, C. Xu, Mitigating the impact of outliers in traffic crash analysis: A robust bayesian regression approach with application to tunnel crash data, *Accident Analysis & Prevention* 185 (2023) 107019.
- [6] Z. Li, Z. Cui, H. Liao, J. Ash, G. Zhang, C. Xu, Y. Wang, Steering the future: Redefining intelligent transportation systems with foundation models, *CHAIN 1* (2024) 46–53.
- [7] S. Feng, H. Sun, X. Yan, H. Zhu, Z. Zou, S. Shen, H. X. Liu, Dense reinforcement learning for safety validation of autonomous vehicles, *Nature* 615 (2023) 620–627.
- [8] H. Liao, Z. Li, H. Shen, W. Zeng, D. Liao, G. Li, S. Li, C. Xu, Bat: Behavior-aware human-like trajectory prediction for autonomous driving, in: *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024, pp. 10332–10340.
- [9] H. Liao, Y. Li, C. Wang, Y. Guan, K. Tam, C. Tian, L. Li, C. Xu, Z. Li, When, where, and what? a benchmark for accident anticipation and localization with large language models, in: *ACM International Conference on Multimedia (ACM MM)*, Oral Presentation, 2024, pp. 8–17.
- [10] Y. Zhu, S. Wang, W. Zhong, N. Shen, Y. Li, S. Wang, Z. Li, C. Wu, Z. He, L. Li, A survey on large language model-powered autonomous driving, *Engineering* (2025).
- [11] H. Liao, C. Wang, K. Zhu, Y. Ren, B. Gao, S. E. Li, C. Xu, Z. Li, Minds on the move: Decoding trajectory prediction in autonomous driving with cognitive insights, *IEEE Transactions on Intelligent Transportation Systems* (2025).
- [12] B. Wijaya, M. Yang, K. Jiang, W. Zhang, D. Yang, Exploring the application of blockchain technology in crowdsource autonomous driving map updating, *Communications in Transportation Research* 4 (2024) 100140.
- [13] H. Liao, H. Shen, B. Wang, Y. Li, Y. Tang, C. Wang, D. Zhuang, K. Chen, H. Yang, C. Xu, et al., Think before you drive: World model-inspired multimodal grounding for autonomous vehicles, *arXiv preprint arXiv:2512.03454* (2025).
- [14] Z. Li, C. Wang, H. Liao, G. Li, C. Xu, Efficient and robust estimation of single-vehicle crash severity: A mixed logit model with heterogeneity in means and variances, *Accident Analysis & Prevention* 196 (2024) 107446.
- [15] J. Wu, Z. Huang, Z. Hu, C. Lv, Toward human-in-the-loop ai: Enhancing deep reinforcement learning via real-time human guidance for autonomous driving, *engineering* 21, 75–91 (2023), 2023.
- [16] H. Liao, Z. Li, C. Wang, B. Wang, H. Kong, Y. Guan, G. Li, Z. Cui, C. Xu, A cognitive-driven trajectory prediction model for autonomous driving in mixed autonomy environment, *arXiv preprint arXiv:2404.17520* (2024).
- [17] J. Zhang, Y. Guan, C. Wang, H. Liao, G. Zhang, Z. Li, Latte: A real-time lightweight attention-based traffic accident anticipation engine, *Information Fusion* (2025) 103173.
- [18] F.-H. Chan, Y.-T. Chen, Y. Xiang, M. Sun, Anticipating accidents in dashcam videos, in: *Asian Conference on Computer Vision*, Springer, 2016, pp. 136–153.
- [19] Y. Yao, M. Xu, C. Choi, D. J. Crandall, E. M. Atkins, B. Dariush, Egocentric vision-based future vehicle localization for intelligent driving assistance systems, in: *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019, pp. 9711–9717.
- [20] W. Bao, Q. Yu, Y. Kong, Uncertainty-based traffic accident anticipation with spatio-temporal relational learning, in: *Proceedings of the 28th ACM International Conference on Multimedia (MM '20)*, 2020.
- [21] J. Fang, D. Yan, J. Qiao, J. Xue, H. Wang, S. Li, Dada-2000: Can driving accident be predicted by driver attention analyzed by a benchmark, in: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, IEEE, 2019, pp. 4303–4309.
- [22] Y. Xia, D. Zhang, J. Kim, K. Nakayama, K. Zipser, D. Whitney, Predicting driver attention in critical situations, in: *Asian conference on computer vision*, Springer, 2018, pp. 658–674.
- [23] H. Tian, T. Deng, H. Yan, Driving as well as on a sunny day? predicting driver's fixation in rainy weather conditions via a dual-branch visual model, *IEEE/CAA Journal of Automatica Sinica* 9 (2022) 1335–1338.
- [24] T. Deng, H. Yan, L. Qin, T. Ngo, B. Manjunath, How do drivers allocate their potential attention? driving fixation prediction via convolutional neural networks, *IEEE Transactions on Intelligent Transportation Systems* 21 (2019) 2146–2154.
- [25] J. Fang, J. Qiao, J. Xue, Z. Li, Vision-based traffic accident detection and anticipation: A survey, *IEEE Transactions on Circuits and Systems for Video Technology* (2023).
- [26] Z. Sheng, Z. Huang, Y. Qu, Y. Leng, S. Chen, Talk2traffic: Interactive and editable traffic scenario generation for autonomous driving with multimodal large language model, in: *Proceedings of the Computer Vision and Pattern Recognition Conference, 2025*, pp. 3788–3797.
- [27] S. Ghosh, S. J. Sunny, R. Roney, Accident detection using convolutional neural networks, in: *2019 international conference on data science and communication (IconDSC)*, IEEE, 2019, pp. 1–6.
- [28] M. Kang, W. Lee, K. Hwang, Y. Yoon, Vision transformer for detecting critical situations and extracting functional scenario for automated vehicle safety assessment, *Sustainability* 14 (2022) 9680.
- [29] T. Wang, K. Chen, G. Chen, B. Li, Z. Li, Z. Liu, C. Jiang, Gsc: A graph and spatio-temporal continuity based framework for accident anticipation, *IEEE Transactions on Intelligent Vehicles* (2023).
- [30] J. Yang, S. Gao, Y. Qiu, L. Chen, T. Li, B. Dai, K. Chitta, P. Wu, J. Zeng, P. Luo, et al., Generalized predictive model for autonomous driving, *arXiv preprint arXiv:2403.09630* (2024).
- [31] D. Yang, Y. Wu, F. Sun, J. Chen, D. Zhai, C. Fu, Freeway accident detection and classification based on the multi-vehicle trajectory data and deep learning model, *Transportation research part C: emerging technologies* 130 (2021) 103303.
- [32] H. Liao, H. Sun, H. Shen, C. Wang, K. Tam, C. Tian, L. Li, C. Xu, Z. Li, Crash: Crash recognition and anticipation system harnessing with context-aware and temporal focus attentions, in: *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 11041–11050.
- [33] W. Bao, Q. Yu, Y. Kong, Drive: Deep reinforced accident anticipation with visual explanation, in: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7619–7628.
- [34] S. Chen, Y. Leng, S. Labi, A deep learning algorithm for simulating autonomous driving considering prior knowledge and temporal information, *Computer-Aided Civil and Infrastructure Engineering* 35 (2020) 305–321.
- [35] Y. Guan, H. Liao, C. Wang, X. Liu, J. Zhang, Z. Li, World model-based end-to-end scene generation for accident anticipation in autonomous driving, *Communications Engineering* 4 (2025) 144.

- [36] M. M. Karim, Y. Li, R. Qin, Z. Yin, A dynamic spatial-temporal attention network for early anticipation of traffic accidents, *IEEE Transactions on Intelligent Transportation Systems* 23 (2022) 9590–9600.
- [37] W. Song, S. Li, T. Chang, K. Xie, A. Hao, H. Qin, Dynamic attention augmented graph network for video accident anticipation, *Pattern Recognition* 147 (2024) 110071.
- [38] Y. Xia, J. Kim, J. Canny, K. Zipser, T. Canas-Bajo, D. Whitney, Periphery-fovea multi-resolution driving model guided by human attention, in: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 1767–1775.
- [39] A. Palazzi, D. Abati, F. Solera, R. Cucchiara, et al., Predicting the driver's focus of attention: the dr (eye) ve project, *IEEE transactions on pattern analysis and machine intelligence* 41 (2018) 1720–1733.
- [40] R. Fu, T. Huang, M. Li, Q. Sun, Y. Chen, A multimodal deep neural network for prediction of the driver's focus of attention based on anthropomorphic attention mechanism and prior knowledge, *Expert Systems with Applications* 214 (2023) 119157.
- [41] G.-P. Corcoran, J. Clark, Traffic risk assessment: A two-stream approach using dynamic-attention, in: *2019 16th Conference on Computer and Robot Vision (CRV)*, IEEE, 2019, pp. 166–173.
- [42] X. Liu, B. Rao, Y. Guan, C. Wang, H. Liao, J. Zhang, C. Lin, M. Zhu, Z. Li, Predict and resist: Long-term accident anticipation under sensor noise, *arXiv preprint arXiv:2511.08640* (2025).
- [43] T. Suzuki, H. Kataoka, Y. Aoki, Y. Satoh, Anticipating traffic accidents with adaptive loss and large-scale incident db, in: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3521–3529.
- [44] L. Neumann, A. Zisserman, A. Vedaldi, Future event prediction: If and when, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.
- [45] H. Liao, B. Rao, H. Sun, C. Wang, Q. Chang, S. E. Li, C. Xu, Z. Li, Chain-of-thought guided multimodal large language models for scene-aware accident anticipation in autonomous driving, *IEEE Transactions on Intelligent Transportation Systems* (2025).
- [46] Y. Zhang, X. Shi, Y. Shao, N. Shiwakoti, J. Zhang, Z. Pu, Z. Ye, A review of scenario cases for autonomous transportation system: Insights from CAV safety testing and scenario generation, *Accident Analysis & Prevention* 215 (2025) 107994.
- [47] J. Dong, S. Chen, M. Miralinaghi, T. Chen, P. Li, S. Labi, Why did the ai make that decision? towards an explainable artificial intelligence (xai) for autonomous driving systems, *Transportation research part C: emerging technologies* 156 (2023) 104358.
- [48] M. M. Karim, Y. Li, R. Qin, Toward explainable artificial intelligence for early anticipation of traffic accidents, *Transportation research record* 2676 (2022) 743–755.
- [49] K. V. Thakare, D. P. Dogra, H. Choi, H. Kim, I.-J. Kim, Rareanom: a benchmark video dataset for rare type anomalies, *Pattern Recognition* 140 (2023) 109567.
- [50] A. Sathyanarayana, P. Boyraz, J. H. Hansen, Information fusion for robust 'context and driver aware' active vehicle safety systems, *Information Fusion* 12 (2011) 293–303.
- [51] Z. Huang, Z. Sheng, C. Ma, S. Chen, Human as ai mentor: Enhanced human-in-the-loop reinforcement learning for safe and efficient autonomous driving, *Communications in Transportation Research* 4 (2024) 100127.
- [52] S. Kuang, Y. Liu, X. Wang, X. Wu, Y. Wei, Harnessing multimodal large language models for traffic knowledge graph generation and decision-making, *Communications in Transportation Research* 4 (2024) 100146.
- [53] W. Wang, J. Xie, C. Hu, H. Zou, J. Fan, W. Tong, Y. Wen, S. Wu, H. Deng, Z. Li, et al., Drivemlm: Aligning multi-modal large language models with behavioral planning states for autonomous driving, *arXiv preprint arXiv:2312.09245* (2023).
- [54] C. Cui, Z. Yang, Y. Zhou, Y. Ma, J. Lu, Z. Wang, Large language models for autonomous driving: Real-world experiments, *CoRR* (2023).
- [55] Z. Cai, N. Vasconcelos, Cascade r-cnn: Delving into high quality object detection, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6154–6162.
- [56] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: *International Conference on Learning Representations*, 2015.
- [57] C. Zhao, W. Mu, X. Zhou, W. Liu, F. Yan, T. Deng, Salm²: An extremely lightweight saliency mamba model for real-time cognitive awareness of driver attention, in: *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2025, pp. 1647–1655.
- [58] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, *Advances in neural information processing systems* 30 (2017).
- [59] H. Liao, H. Kong, B. Wang, C. Wang, K. Y. Wang, Z. He, C. Xu, Z. Li, Cot-drive: Efficient motion forecasting for autonomous driving with llms and chain-of-thought prompting, *IEEE Transactions on Artificial Intelligence* 7 (2025) 625–641.
- [60] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altschmidt, S. Altman, S. Anadkat, et al., Gpt-4 technical report, *arXiv preprint arXiv:2303.08774* (2023).
- [61] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al., The llama 3 herd of models, *arXiv preprint arXiv:2407.21783* (2024).
- [62] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, et al., Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, *arXiv preprint arXiv:2409.12191* (2024).
- [63] K. Li, Y. Wang, Y. He, Y. Li, Y. Wang, L. Wang, Y. Qiao, Uniformerv2: Spatiotemporal learning by arming image vits with video uniformer, *arXiv preprint arXiv:2211.09552* (2022).
- [64] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, H. Hu, Video swin transformer, in: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 3202–3211.
- [65] K. Zhang, Z. Chen, Video saliency prediction based on spatial-temporal two-stream network, *IEEE Transactions on Circuits and Systems for Video Technology* 29 (2019) 3544–3557.
- [66] Q. Li, C. Liu, F. Chang, S. Li, H. Liu, Z. Liu, Adaptive short-temporal induced aware fusion network for predicting attention regions like a driver, *IEEE Transactions on Intelligent Transportation Systems* 23 (2022) 18695–18706.
- [67] J. Fang, D. Yan, J. Qiao, J. Xue, H. Yu, Dada: Driver attention prediction in driving accident scenarios, *IEEE Transactions on Intelligent Transportation Systems* 23 (2022) 4959–4971.

- [68] K. Min, J. J. Corso, Tased-net: Temporally-aggregating spatial encoder-decoder network for video saliency detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 2394–2403.
- [69] T. Zhao, X. Bai, J. Fang, J. Xue, Gated driver attention predictor, in: 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), 2023, pp. 270–276. doi:10.1109/ITSC57777.2023.10422543.
- [70] M. Cornia, L. Baraldi, G. Serra, R. Cucchiara, A deep multi-level network for saliency prediction, in: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 3488–3493.
- [71] Y. Chen, Z. Nan, T. Xiang, Fblnet: Feedback loop network for driver attention prediction, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 13371–13380.
- [72] I. Kotseruba, J. K. Tsotsos, Scout+: Towards practical task-driven drivers' gaze prediction, in: 2024 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2024, pp. 1927–1932.