LB/TH/30/2026
TH6280

VISION-BASED ACCIDENT ANTICIPATION WITH DEEP NEURAL NETWORKS

Vipulananthan Vipooshan

248048G

Master of Science(Major Component of Research)

Department of Computer Science and Engineering
Faculty of Engineering

University of Moratuwa
Sri Lanka

June 2026

VISION-BASED ACCIDENT ANTICIPATION WITH DEEP NEURAL NETWORKS

Vipulananthan Vipooshan

248048G

Thesis submitted in partial fulfilment of the requirements for the degree
Master of Science(Major Component of Research)

Department of Computer Science and Engineering
Faculty of Engineering

University of Moratuwa
Sri Lanka

June 2026

DECLARATION

I declare that this is my own work and this Thesis does not incorporate without acknowledgment any material previously submitted for a Degree or Diploma in any other University or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date: 10/06/2026

The supervisor should certify the Thesis with the following declaration.

The above candidate has carried out research for the Master of Science(Major Component of Research) Thesis under my supervision. I confirm that the declaration made above by the student is true and correct.

Name of Supervisor: Dr. Charith Chitraranjan

Signature of the Supervisor: Date: 10/06/2026

ACKNOWLEDGEMENT

This thesis would not have been possible without the support, guidance, and encouragement of many individuals and institutions, to whom I am deeply grateful.

First and foremost, I would like to express my sincere gratitude to my supervisor, Dr. Charith Chitraranjan. His visionary insights and meticulous attention to detail provided the ideal balance of breadth and depth throughout this work. His patient mentorship, constructive feedback, and unwavering confidence in my abilities continuously motivated me to overcome challenges and develop my ideas into a cohesive and robust system.

I am also thankful to my examiners, Dr. Ranga Rodrigo and Dr. Thanuja D. Ambegoda, for their careful evaluation of my research during progress reviews. Their insightful questions and valuable suggestions encouraged me to strengthen my arguments and clearly articulate my contributions, significantly enhancing the overall quality of this work.

I am sincerely grateful to the academic staff of the Department of Computer Science & Engineering, Faculty of Engineering, University of Moratuwa, for providing a strong academic foundation and a supportive research environment. Their guidance throughout my MSc by Research program has been instrumental in the successful completion of this thesis.

I would also like to extend my heartfelt thanks to my family for their unwavering love, patience, and moral support. Their constant encouragement and understanding made this journey possible.

Finally, I would like to acknowledge all those who contributed, directly or indirectly, to this research, including the contributors and maintainers of open-source datasets, the broader research community whose prior work informed and inspired this study, and all individuals whose efforts laid the foundation for this thesis. I am honoured to build upon their contributions.

ABSTRACT

Traffic accidents remain a leading cause of fatalities and injuries worldwide, making the ability to anticipate hazardous situations in advance critical for improving road safety. Accident anticipation systems enable timely preventive actions, such as driver alerts and collision avoidance maneuvers, and play a key role in advanced driver assistance systems (ADAS) and autonomous driving. While many existing approaches rely on multiple sensors such as LiDAR, radar, and GPS, using dash-cam videos offers a cost-effective and easily deployable alternative, albeit with significant challenges due to the complexity of real-world driving environments and real-time constraints.

In this thesis, we investigate efficient and accurate accident anticipation from dash-cam videos through two complementary approaches. First, we propose an enhanced spatio-temporal graph-based framework (STAGNet), which improves accident prediction by incorporating richer spatio-temporal features and aggregating them using recurrent neural networks. Second, we introduce VAGNet, a novel architecture that leverages global scene representations instead of computationally expensive object-level features, combining transformer and graph modules with the vision foundation model VideoMAE-V2 for efficient feature extraction.

Extensive experiments conducted on multiple benchmark datasets, including DAD, DoTA, DADA, and Nexar, demonstrate that the proposed methods achieve superior performance in terms of average precision and mean time-to-accident compared to existing approaches. Notably, the global feature-based framework significantly improves computational efficiency while maintaining high predictive accuracy. Together, these contributions advance the development of scalable, real-time accident anticipation systems using only dash-cam video, providing a practical pathway toward safer intelligent transportation systems.

Keywords: Advanced driver assistance systems, Computer vision, Transformers, Graph neural networks, Foundation models.

TABLE OF CONTENTS

Declaration of the Candidate & Supervisor i

Acknowledgement ii

Abstract iii

Table of Contents iv

1 Introduction 1

2 Literature Review 3

2.1 Methods based on convolutional neural networks 3

2.2 Convolutional and recurrent neural networks 3

2.3 Incorporating attention mechanisms 4

2.4 Graph neural networks 6

2.5 Reinforcement learning-based methods 7

2.6 Transformer-based and Multi-modal Methods 8

3 Methodology 9

3.1 STAGNet: A Spatio-Temporal Graph and LSTM Framework for Accident Anticipation 9

3.1.1 Spatio-Temporal Object Graph Learning 9

3.1.2 Spatio-Temporal Frame-level Feature Learning 10

3.1.3 Frame Graph Learning 10

3.1.4 Model Pruning 12

3.2 VAGNet: Vision-based Accident Anticipation with Global Features 12

4 Experimental Evaluation 16

4.1 Experimental Settings 16

4.1.1 Datasets 16

4.1.2 Networks and Baselines 17

4.1.3 Evaluation Metrics 19

4.1.4 Results and Discussion 20

5 Conclusion and Future Work 31

5.1 Limitations and Future work 31

5.2 Conclusion 31

References 33

CHAPTER 1

INTRODUCTION

Traffic accidents claim more than one million lives annually and remain a major global public safety concern [1, 2]. Early and accurate anticipation of hazardous situations can significantly mitigate accidents by enabling timely preventive actions, such as driver alerts and collision avoidance maneuvers. Consequently, accident anticipation plays a critical role in advanced driver assistance systems (ADAS) and autonomous vehicles (AVs). In autonomous driving, such predictive capabilities support proactive safety behaviors, including defensive driving strategies and human takeover when required [3–8].

Despite significant advances in autonomous systems, ensuring safety in real-world environments remains challenging. AVs are typically trained using reinforcement or imitation learning, and they cannot be exposed to the full spectrum of rare and hazardous events encountered in real-world driving scenarios [9]. This limitation is commonly referred to as the "long-tail challenge" in autonomous driving [10, 11]. Therefore, accident anticipation systems trained on large-scale real-world driving data, such as dash-cam videos, provide an essential additional layer of safety [12–15].

While many existing accident anticipation systems rely on multiple sensors such as LiDAR, radar, and GPS, these solutions are often expensive and difficult to deploy at scale. In contrast, dash-cam-based approaches offer a cost-effective and easily deployable alternative, with the potential to be retrofitted into existing vehicles [16, 17]. However, relying solely on visual data introduces significant challenges due to the complexity of real-world driving environments, including dynamic interactions among multiple agents, varying lighting conditions, and occlusions [18].

Recent research has explored vision-based accident anticipation using deep learning techniques, particularly graph neural networks (GNNs) and spatio-temporal models [19–25]. Most existing approaches rely on explicit object detection and extract features from individual objects (e.g., using VGG16 [26]), followed by modeling their interactions using graph-based frameworks. While effective, these methods are computationally expensive and may hinder real-time deployment due to the overhead of object-level feature extraction.

To address these challenges, this thesis investigates two complementary approaches for efficient and accurate accident anticipation using dash-cam videos. First, we enhance graph-based accident prediction by incorporating improved spatio-temporal features and explicitly modeling temporal dependencies using recurrent neural networks. Building upon prior GNN-based methods, we introduce a framework that integrates spatio-temporal feature extraction with sequence modeling, enabling more effective representation of dynamic driving scenarios [21, 23]. This approach improves the ability of graph models to capture both spatial interactions and temporal evolution in driving scenes.

Second, we propose a novel paradigm that eliminates the need for explicit object-level feature extraction. Instead, we leverage global scene representations obtained from a vision foundation model, VideoMAE-V2 [27], which captures rich spatio-temporal information directly from video frames. By combining transformer-based modeling with graph structures, the proposed architecture (VAGNet) achieves competitive or superior performance while significantly reducing computational complexity. This shift from object-centric to global feature-based modeling enables more scalable and real-time accident anticipation systems, while maintaining high predictive accuracy.

Another important limitation in existing work lies in dataset biases. Commonly used datasets such as DAD [28] and CCD [29] suffer from issues such as low proportions of ego-involved accidents and video quality biases, which can lead to misleading performance evaluations [22, 30]. To address this, we conduct extensive evaluations on more reliable and diverse benchmark datasets, including DADA [14], DoTA [13], and Nexar [12], with a focus on ego-involved accidents that can be directly mitigated by the ego-vehicle. We also evaluate cross-dataset generalization to assess the robustness of the proposed approaches.

The main contributions of this thesis are summarized as follows:

- We develop an enhanced spatio-temporal graph-based framework that integrates sequence modeling with graph attention for improved accident anticipation.
- We propose a novel global feature-based architecture that eliminates the need for explicit object detection, significantly improving computational efficiency while maintaining high predictive performance.
- We demonstrate the effectiveness and generalization capability of the proposed methods through extensive experiments on multiple benchmark datasets, including cross-dataset evaluations.
- We provide a comprehensive analysis of efficiency-accuracy trade-offs, enabling practical deployment of real-time accident anticipation systems.

CHAPTER 2

LITERATURE REVIEW

Vehicle collision prediction methods utilize deep learning models to estimate the likelihood of a future collision using the current frame along with a sequence of preceding frames. These models are trained on datasets containing both accident and non-accident driving scenarios.

We begin by reviewing methods that primarily rely on convolutional neural networks (CNNs) in supervised learning settings, without incorporating recurrent neural networks (RNNs) or attention mechanisms. Subsequently, We examine more advanced approaches that integrate RNNs, attention mechanisms, graph neural networks (GNNs), and reinforcement learning. Finally, we discuss methods that adopt similar architectures using transformer-based approaches. Although these methods are categorized for clarity, the boundaries between categories are not strict, as certain techniques—such as attention—are employed across multiple approaches. Additionally, some methods aim to predict accidents regardless of ego-vehicle involvement. Nevertheless, they are included in this review because they are trained and evaluated on datasets containing ego-involved accidents, with cameras mounted on the ego vehicle.

2.1 Methods based on convolutional neural networks

Several approaches use CNNs as the primary modeling framework. Jung et al. [31] propose a CNN-based model that directly predicts hazardous situations involving pedestrians and cyclists (Figure 2.1 (a) i). Unlike traditional pipelines that first detect objects and then predict collisions, their method performs end-to-end prediction directly from the current image frame I_t.

Similarly, Kim et al. [32] employ a CNN-based approach (Figure 2.1 (a) ii). In their method, objects are first detected and tracked across frames. The RGB image channels are combined with bounding box representations to form a four-channel input sequence, which is processed by a CNN (a simplified VGG-16 architecture) to predict collision likelihood. Training is conducted using synthetic data generated from a simulator based on GTA V. To improve label quality, trajectory-based models such as CTRA and CCA [33] are used to refine collision annotations. During inference, real images are transformed into synthetic-like representations using CycleGAN before being passed to the prediction model. Final predictions are obtained by averaging outputs from both real and transformed inputs.

2.2 Convolutional and recurrent neural networks

Strickland et al. [34] extend CNN-based models by incorporating convolutional LSTM (ConvLSTM) layers to capture temporal dependencies. Their model processes se-

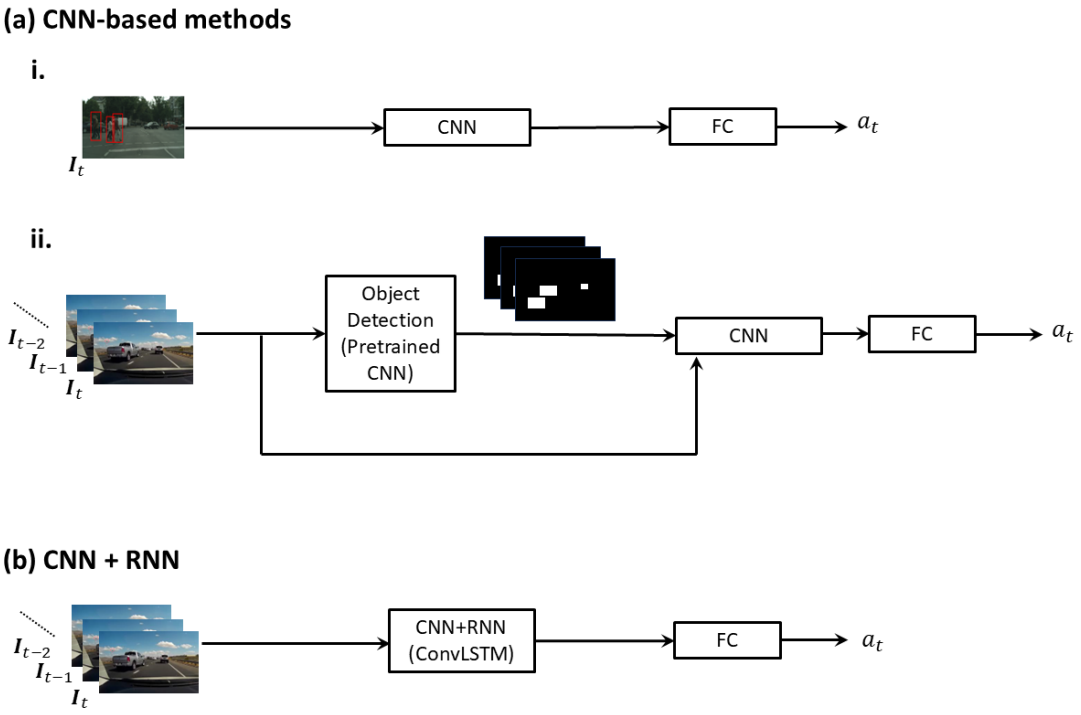


Fig. 2.1: Collision prediction using (a) CNNs and (b) CNNs combined with RNNs. $I_t, I_{t-1}, ...$ denote input frames, and a_t is the predicted collision probability.

quences from multiple ego-mounted cameras along with proprioceptive data and driving commands to estimate collision probability. Image sequences are passed through ConvLSTM layers followed by fully connected layers (Figure 2.1 (b)). To quantify prediction uncertainty, multiple stochastic forward passes are performed using dropout [35]. The model is trained using simulated data generated in V-REP [36], where collision labels are automatically derived from vehicle trajectories.

2.3 Incorporating attention mechanisms

Attention mechanisms enable models to focus on relevant regions or time steps, mimicking human perception. In computer vision, attention can be categorized into channel, spatial, temporal, and branch attention. Hybrid forms, such as spatiotemporal attention, combine multiple types. These mechanisms improve both performance and interpretability.

Spatial attention:

Several works [25, 28, 37] apply spatial attention to emphasize important objects within a scene. These methods extract object-level features O_t and global features f_t at each time step. Attention weights α_t are computed based on object features and the previous hidden state h_{t-1} of an RNN, enabling the model to focus on critical objects. The aggregated representation $\psi(O_t, \alpha_t)$ is then used to predict accident probability a_t.

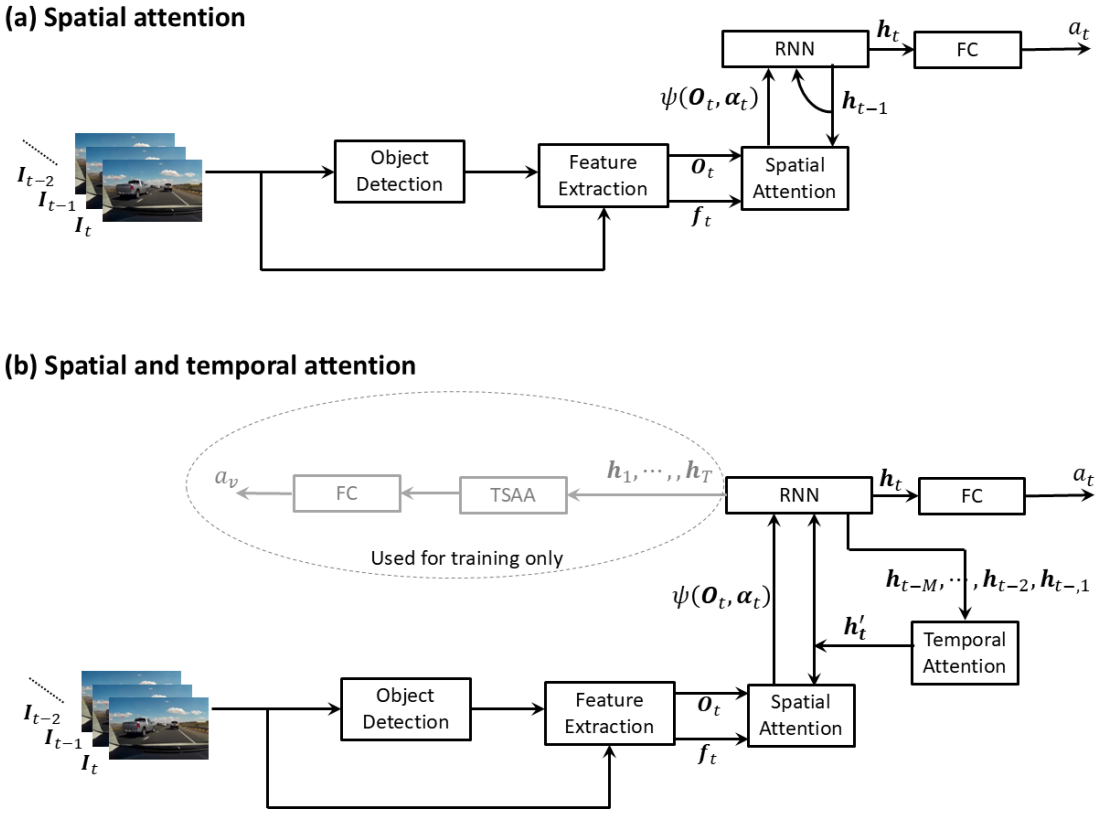


Fig. 2.2: Collision prediction using (a) spatial attention and (b) spatiotemporal attention.

Chan et al. [28] define attention weights as:

$$\alpha_t^i = SoftMax_i \left\{ W^T \rho \left(W_e h_{t-1} + U_e o_t^i + b_e \right) \right\}, \tag{2.1}$$

and compute aggregated features as:

$$\psi(O_t, \alpha_t) = \left[\sum_{i=1}^{N} \alpha_t^i o_t^i; f_t \right]. \tag{2.2}$$

Subsequent works introduce variations in attention computation and network architecture. For instance, Suzuki et al. [25] replace LSTM with QRNN for improved efficiency, while Corcoran et al. [38] simplify attention computation using dot-product attention and incorporate optical flow to model motion dynamics. Fatima et al. [37] extend attention to pairwise object interactions, capturing more complex relationships.

Spatiotemporal attention:

Some methods integrate temporal attention in addition to spatial attention to identify informative frames. Karim et al. [22] compute temporal attention weights over

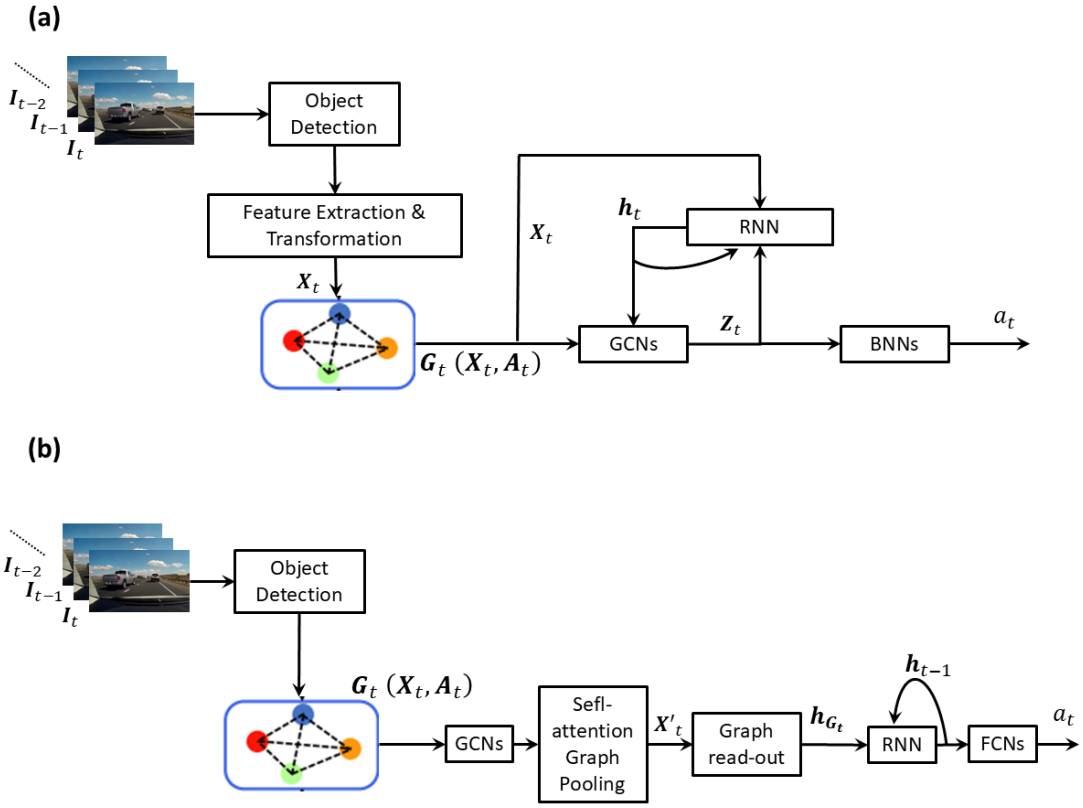


Fig. 2.3: Collision prediction using graph neural networks.

past hidden states:

$$\beta_{t-1}^{k} = SoftMax_k\left\{W_{ta} h_{t-k}\right\}, \quad h'_{t-1} = \sum_{k=1}^{M} \beta_{t-1}^{k} h_{t-k}. \tag{2.3}$$

This refined hidden state is used for both attention computation and recurrent updates. Yi et al. [39] further enhance this approach by introducing bag-level loss and leveraging temporal attention outputs for prediction.

2.4 Graph neural networks

Graph-based methods model relationships between objects explicitly. As shown in Figure 2.3, detected objects are represented as nodes, and their interactions are encoded as edges.

Bao et al. [29] construct graphs where node features combine object and global features:

$$x_t^i = \left[\phi(o_t^i); \phi(f_t)\right]. \tag{2.4}$$

Edge weights are defined based on spatial proximity:

$$a_{ij} = \frac{exp(-d(i,j))}{\sum_{ij} exp(-d(i,j))}. \tag{2.5}$$

The graph is processed using GCN layers and recurrent updates to capture spatio-temporal relationships. Bayesian neural networks are used for prediction, providing both probability estimates and uncertainty measures.

Mahmood et al. [30] extend this approach by incorporating geometric features to improve generalization, while Malawade et al. [23] introduce scene graphs with attention-based pooling and LSTM-based temporal modeling.

2.5 Reinforcement learning-based methods

Some approaches formulate accident prediction as a reinforcement learning problem. Bao et al. [24] model collision probability as an action in a Markov decision process. They employ dynamic attention fusion to focus on salient regions and train the model using a soft actor-critic framework. The reward function encourages early and correct predictions.

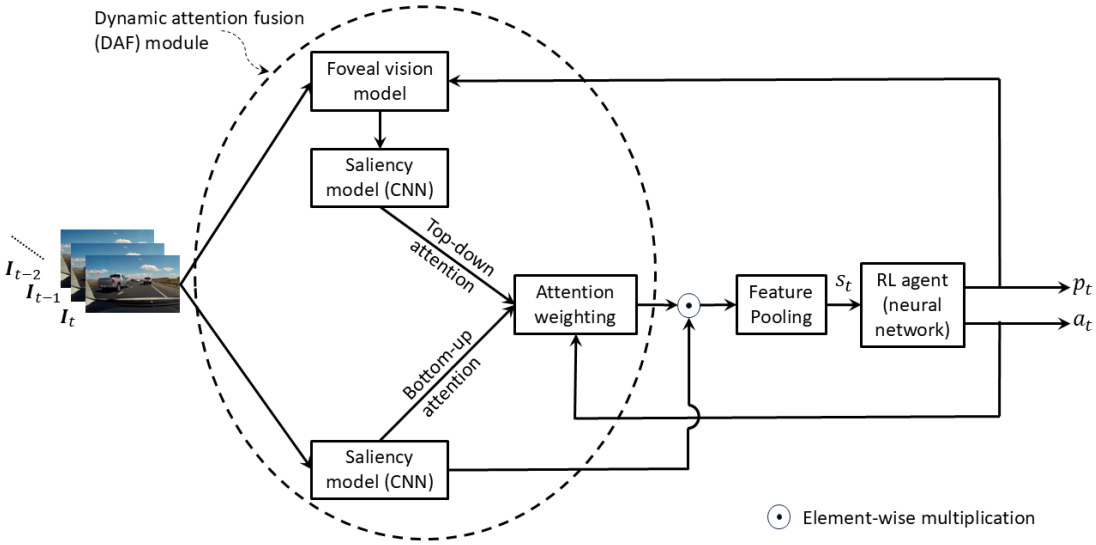


Fig. 2.4: Reinforcement learning framework for collision prediction.

Cho et al. [40] propose a similar framework using double actor regularized critics, achieving improved performance in both accuracy and early prediction.

In recent years, vision-based accident anticipation has attracted significant research attention, resulting in the development of numerous methods. Comprehensive reviews of work published up to 2023 can be found in [18, 41]. In this section, we focus on more recent advancements introduced from 2024 onward.

Most accident anticipation frameworks follow a common pipeline that begins with object detection, followed by feature extraction at both the object level (from bounding boxes) and the global level (from entire frames). Typically, pretrained models are employed for these tasks, such as Faster R-CNN [42] for object detection and VGG16 [26] or I3D [43] for feature extraction. These extracted features are then processed by neural network architectures to estimate the probability of an impending accident.

A notable approach is the nested graph architecture proposed by Thakur et al. [21], where detected objects are represented as nodes within an object-level graph. In this graph, spatial relationships are encoded through connections between objects within the same frame, while temporal relationships are captured by linking objects across consecutive frames. The object-level graph is subsequently aggregated into a frame-level graph, where each node represents a video frame. In addition, a separate frame-level graph is constructed using global features extracted from frames, with I3D [43] enabling the modeling of spatio-temporal dynamics. These graphs are then processed using graph attention mechanisms and fully connected layers to predict accident likelihood. This framework was further enhanced in [44] by incorporating SlowFast networks [45] for improved global feature extraction and integrating long short-term memory (LSTM) units to better capture temporal dependencies.

Another graph-based approach is presented in [20], where object-level and global features are combined using an attention-based fusion mechanism. The fused features are then used to construct a directed graph that models temporal relationships across frames.

2.6 Transformer-based and Multi-modal Methods

In addition to graph-based methods, several studies have explored transformer-based architectures. The AAT-DA model [46] employs a dual-transformer design consisting of separate spatial and temporal transformers to predict accidents. It also incorporates driver attention [47] when computing attention weights for detected objects, allowing the model to account for the influence of each object within the scene. Other approaches [48, 49] utilize transformer architectures to fuse multiple modalities, including video data, textual descriptions, and driver attention maps. However, these methods depend on human-annotated textual descriptions during training, which may limit scalability.

CCAF-Net [19] introduces an alternative multi-modal framework that integrates estimated 3D depth information from RGB images with both object-level and global features. These heterogeneous inputs are combined using a collaborative attention mechanism, enabling more effective modeling of complex spatial relationships. Nevertheless, the additional requirement of depth estimation introduces extra computational overhead.

CHAPTER 3

METHODOLOGY

3.1 STAGNet: A Spatio-Temporal Graph and LSTM Framework for Accident Anticipation

In this work, we present an enhanced framework for accident anticipation from dash-cam videos, built upon the Graph(Graph) architecture [21]. Given a sequence of video frames $(X_1, \ldots, X_t)$, the objective is twofold: (1) to determine whether an accident is likely to occur, and (2) to identify such events as early as possible. At each time step t, the model outputs a probability p_t representing the likelihood of a future accident. For sequences containing accidents, the Time-to-Accident (TTA) is defined as $\tau = a - t$ for $t < a$, where a denotes the start of the accident and t is the earliest frame for which $p_t > \alpha$, with α being a predefined threshold.

The overall architecture of the proposed model is illustrated in Figure 3.1. Similar to the Graph(Graph) framework, our model consists of three main components. Module A models spatio-temporal relationships among detected objects, Module B extracts global contextual information from the video, and Module C integrates these representations to generate the final prediction. The following subsections describe each component and highlight the key improvements introduced in this work.

3.1.1 Spatio-Temporal Object Graph Learning

In real-world driving scenarios, different objects contribute unequally to accident risk. Objects that are spatially closer to each other or to the ego-vehicle are generally more relevant. Although accurate depth estimation is difficult with monocular dash-cam footage, this module captures object relationships both within a frame and across time. A spatio-temporal graph attention mechanism is employed to emphasize the most relevant interactions.

We first detect objects in each frame using YOLO [50] and obtain their bounding boxes. Visual features are extracted for each object using a pretrained VGG16 network [26], while semantic information is incorporated through GloVe embeddings [51]. For each frame X_t, a spatio-temporal object graph G_t^{obj} is constructed, where nodes correspond to detected objects. The node features are initialized by concatenating visual features $f_e^{\text{obj}} \in \mathbb{R}^{S \times d_1}$ and label embeddings $f_l^{\text{obj}} \in \mathbb{R}^{S \times d_2}$.

To model spatial relationships, we define the adjacency matrix:

$$A_s^{\text{obj}}(i,j) = \frac{e^{-d(c_i,c_j)}}{\sum_{ij} e^{-d(c_i,c_j)}}, \tag{3.1}$$

where $d(c_i, c_j)$ represents the Euclidean distance between object centers. This formulation assigns higher importance to nearby objects while reducing the influence of

distant ones.

Temporal relationships are captured using:

$$A_{tm}^{\text{obj}}(i,j) = \begin{cases} s(i,j), & \text{if } l_i = l_j \\ 0, & \text{otherwise} \end{cases}, \tag{3.2}$$

where $s(i,j)$ denotes cosine similarity between object features across frames. This allows information to propagate between instances of the same object class over time.

Node representations are updated using Graph Attention Networks (GATv2) [52]:

$$\begin{aligned} f_{\text{obj}}^{s} &= \text{GAT}([\phi(f_e^{\text{obj}}), \phi(f_l^{\text{obj}})], A_s^{\text{obj}}), \\ f_{\text{obj}}^{tm} &= \text{GAT}([\phi(f_e^{\text{obj}}), \phi(f_l^{\text{obj}})], A_{tm}^{\text{obj}}), \\ f'_{\text{obj}} &= [f_{\text{obj}}^{s}, f_{\text{obj}}^{tm}], \end{aligned} \tag{3.3}$$

where ϕ denotes a fully connected projection layer. Compared to GCN-based approaches, GATv2 allows adaptive weighting of neighboring nodes, enabling the model to focus on the most relevant object interactions for improved accident prediction.

3.1.2 Spatio-Temporal Frame-level Feature Learning

While object-level graphs capture detailed interactions, global contextual understanding of the scene is equally important. Many previous methods rely on image-based models such as VGG16, which lack temporal modeling capabilities. Although I3D captures short-term temporal dynamics, it is limited in representing fast-moving objects and longer temporal dependencies.

To address this, we employ a pretrained SlowFast network [45] to extract spatio-temporal features from video sequences. This architecture processes both slow-changing spatial information and rapid motion patterns, making it well-suited for dash-cam data.

Given a sequence $X_{\text{seq}} = (X_{t-u}, \ldots, X_t)$, we obtain feature representations as:

$$\begin{aligned} f_{\text{fr}} &= \text{SlowFast}(X_{\text{seq}}), \\ f_{\text{fr}}^{\text{fc}} &= \phi(f_{\text{fr}}), \\ f'_{\text{fr}} &= \text{LSTM}(f_{\text{fr}}^{\text{fc}}), \end{aligned} \tag{3.4}$$

where ϕ is a fully connected layer. The LSTM module further enhances temporal modeling by capturing long-range dependencies across frames.

3.1.3 Frame Graph Learning

To explicitly model temporal dependencies between frames, we construct a frame-level graph. Each node represents a frame, and edges connect each frame to its preceding k frames, ensuring a causal structure.

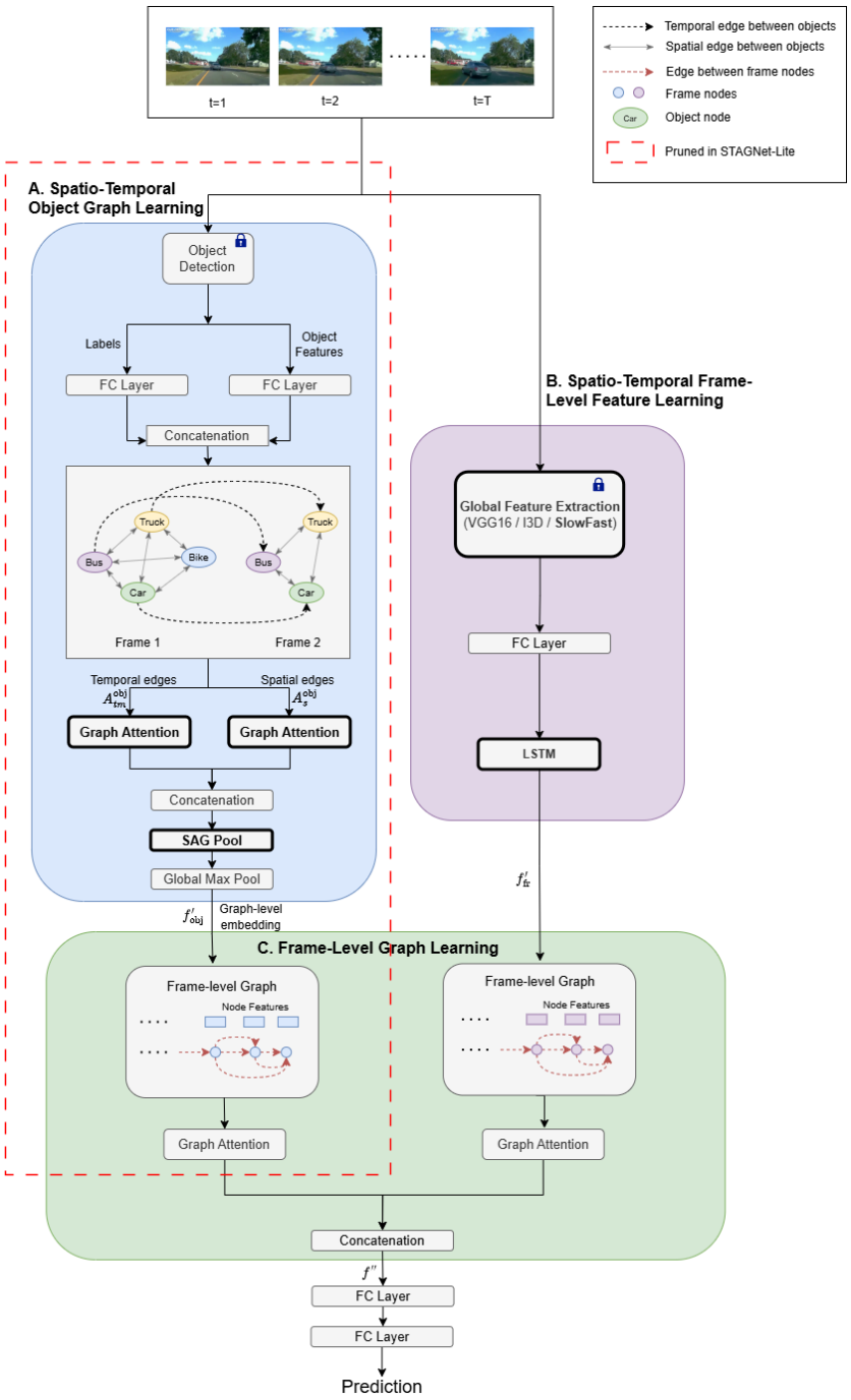


Fig. 3.1: Architecture of the proposed STAGNet accident anticipation framework. Boxes with thicker borders represent the main components introduced in this work. The components within the red dashed rectangle are pruned in the light-weight model, STAGNet-Lite.

The adjacency matrix is defined as:

$$A_{\text{fr}}(i,j) = \begin{cases} 1, & \text{if } i-j \leq k \\ 0, & \text{otherwise} \end{cases}. \tag{3.5}$$

Two parallel GAT layers are applied to capture interactions at both object and frame levels:

$$\begin{aligned} f''_{\text{obj}} &= \text{GAT}(f'_{\text{obj}}, A_{\text{fr}}), \\ f''_{\text{fr}} &= \text{GAT}(f'_{\text{fr}}, A_{\text{fr}}), \\ f'' &= [f''_{\text{obj}}, f''_{\text{fr}}]. \end{aligned} \tag{3.6}$$

The attention mechanism enables the model to identify the most informative frames for predicting future accidents. The combined representation is passed through fully connected layers to produce frame-level probabilities.

The model is trained end-to-end, except for the object detection and feature extraction modules, which remain fixed. The optimization objective is the standard cross-entropy loss:

$$\mathcal{L}(p, y) = -\sum_{m=1}^{M} y_m \log \left(\frac{e^{p_m}}{\sum_j e^{p_j}} \right), \tag{3.7}$$

where $M = 2$ denotes the number of classes.

3.1.4 Model Pruning

The object-level graph construction introduces significant computational overhead. Additionally, since the ego-vehicle is not explicitly detected, its interactions are not directly modeled within the object graph. This motivates a simplified approach focusing on global scene understanding.

We therefore introduce a lightweight variant, **STAGNet-Lite**, by removing the object graph module and relying solely on frame-level representations. This reduces computational complexity while maintaining strong performance for ego-involved accident anticipation. The pruned components are highlighted in Figure 3.1.

3.2 VAGNet: Vision-based Accident Anticipation with Global Features

Understanding dynamic driving environments requires capturing both global contextual information and temporal evolution across video frames. Many existing approaches [22, 29, 30] utilize pretrained image-based networks such as VGG16 [26] to extract global features. However, such models are limited to spatial representations and do not effectively capture temporal dependencies. While methods like I3D [43] incorporate short-term temporal information, they remain insufficient for modeling long-range dependencies and complex interactions over time.

To overcome these limitations, we adopt a pretrained VideoMAE-V2 model [27], which is capable of learning rich spatio-temporal representations directly from video sequences.

Figure 3.2 illustrates the overall architecture of the proposed framework. For a given time step t, we construct an input sequence $X_{\text{seq}} = (X_{t-u}, \ldots, X_t)$ consisting of the current frame and the preceding u frames. This sequence is passed through

the VideoMAE-V2 encoder to obtain a high-dimensional feature representation. The extracted features are then projected into a lower-dimensional space using a fully connected layer, followed by a Transformer Encoder to further model temporal relationships:

$$\begin{aligned} f_{\text{fr}} &= \text{VideoMAE}(X_{\text{seq}}), \\ f_{\text{fr}}^{\text{fc}} &= \text{FC}(f_{\text{fr}}), \\ f'_{\text{fr}} &= \text{TE}(f_{\text{fr}}^{\text{fc}}), \end{aligned} \tag{3.4}$$

Transformer Encoder (TE): The Transformer Encoder follows the standard architecture introduced by Vaswani *et al.* [53]. It consists of $N = 2$ stacked encoder layers with $h = 4$ attention heads. The encoding process is defined as:

$$H_0 = f_{\text{fr}}^{\text{fc}}, \qquad H_i = \text{EncoderLayer}(H_{i-1}),\ i = 1, 2, \ldots, N \tag{3.5}$$

where the final encoded representation is given by $f'_{\text{fr}} = H_N$.

Encoder Layer: Each encoder layer comprises two sub-components: multi-head self-attention (MHA) and a position-wise fully connected layer. Residual connections and layer normalization are applied to stabilize training:

$$\begin{aligned} \tilde{H}_i &= \text{LayerNorm}(H_{i-1} + \text{MHA}(H_{i-1}))\,, \\ H_i &= \text{LayerNorm}\Big(\tilde{H}_i + \text{FC}(\tilde{H}_i)\Big)\,. \end{aligned} \tag{3.6}$$

Multi-Head Self-Attention: Given an input representation $H \in \mathbb{R}^{n \times d_{\text{model}}}$, multi-head attention is defined as:

$$\text{MHA}(H) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\, W^O, \tag{3.7}$$

where h denotes the number of attention heads and W^O is a learnable projection matrix. The sequence length $n = u + 1$ corresponds to the number of input frames.

Each attention head is computed as:

$$\text{head}_j = \text{Softmax}\left(\frac{Q_j K_j^T}{\sqrt{d_s}}\right) V_j, \quad j = 1, \ldots, h, \tag{3.8}$$

where $d_s = d_{\text{model}}/h$. The query, key, and value matrices are obtained through linear projections:

$$Q_j = HW_j^Q, \quad K_j = HW_j^K, \quad V_j = HW_j^V. \tag{3.9}$$

Fully-Connected Layer (FC): The position-wise feed-forward network is defined as:

$$\text{FC}(x) = \sigma(xW_1 + b_1)W_2 + b_2, \tag{3.10}$$

where $\sigma(\cdot)$ is a non-linear activation function (ReLU). The resulting representation encodes both spatial and temporal information and is subsequently used for higher-

level reasoning.

To explicitly capture temporal dependencies between frames, we construct a *frame graph* using the encoded global features. Each node corresponds to a frame at time t, and directed edges connect each frame to its preceding v frames, ensuring a causal structure. The adjacency matrix is defined as:

$$A_{\text{fr}}(i,j) = \begin{cases} 1, & \text{if } i-j \leq v, \\ 0, & \text{otherwise}, \end{cases} \tag{3.11}$$

where i denotes the current frame index.

A Graph Transformer layer [54] is applied to refine the frame-level representations:

$$\begin{aligned} f''_{\text{fr}} &= \text{GT}(f'_{\text{fr}}, A_{\text{fr}}), \\ f'' &= [f'_{\text{fr}}, f''_{\text{fr}}], \end{aligned} \tag{3.12}$$

The attention mechanism within this layer allows the model to identify and prioritize frames that are most informative for accident prediction. The combined representation is then passed through fully connected layers to estimate frame-level probabilities $(p_1, p_2, \ldots, p_N)$.

The entire model is trained end-to-end, except for the VideoMAE-V2 backbone, which is kept fixed. The optimization objective is the standard cross-entropy loss:

$$\mathcal{L}(p,y) = -\sum_{m=1}^{M} y_m \log\left(\frac{e^{p_m}}{\sum_j e^{p_j}}\right), \tag{3.13}$$

where $M = 2$ denotes the number of classes.

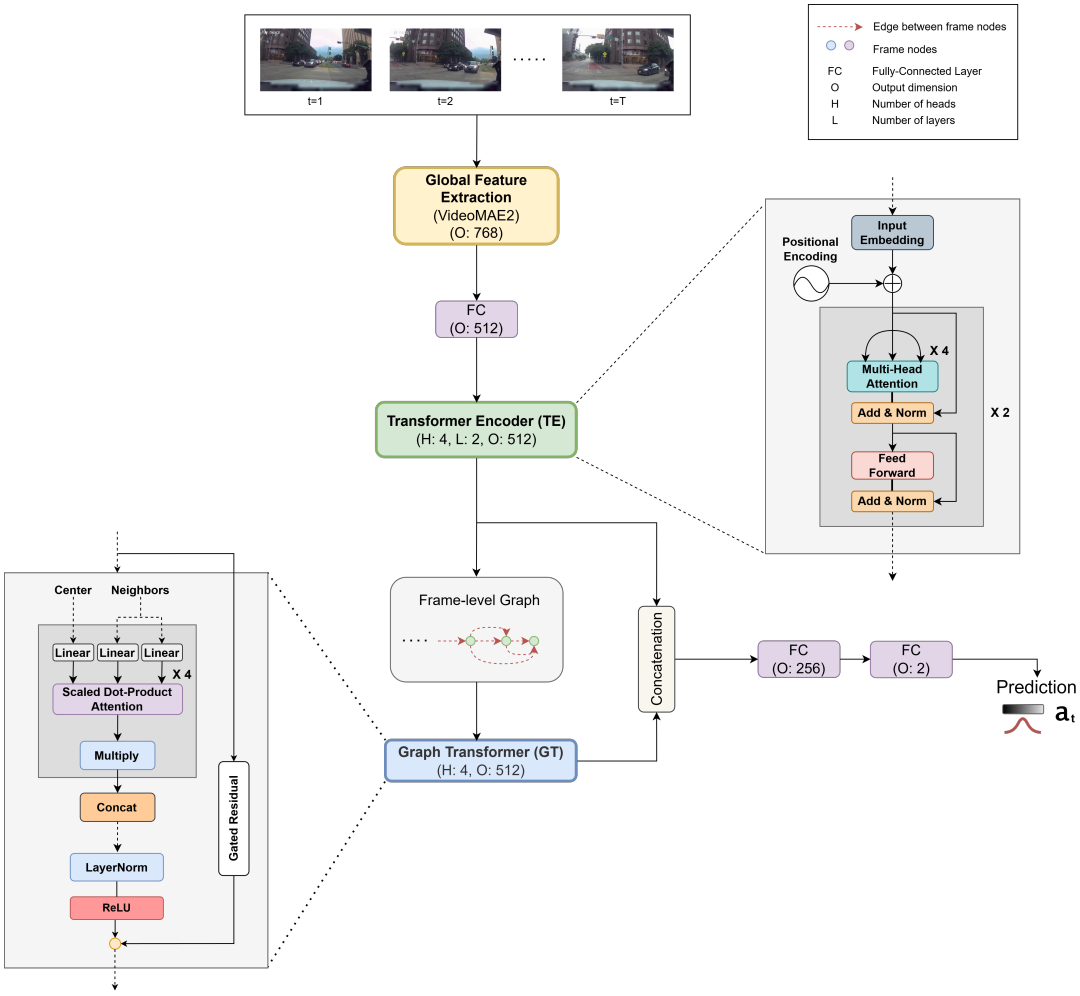


Fig. 3.2: Architecture of the proposed VAGNet accident anticipation framework.

CHAPTER 4

EXPERIMENTAL EVALUATION

4.1 Experimental Settings

4.1.1 Datasets

We evaluate the proposed methods on four widely used benchmark datasets: DAD [28], DADA [14], DoTA [13], and Nexar [12]. For the DAD dataset, we follow the same training and testing splits used in prior work [21, 29, 30]. However, most accident instances in DAD do not involve the ego vehicle. Since ego-involved accidents are of primary importance for Advanced Driver Assistance Systems (ADAS), as they represent scenarios that can be directly prevented or mitigated, our analysis primarily focuses on ego-involved events in the DADA, DoTA, and Nexar datasets.

In the Nexar dataset [12], all accident instances are ego-involved. Therefore, we utilize the dataset in its entirety, following the original training and testing splits. To ensure consistency with other datasets and our implementation, all training videos are trimmed to 5-second clips, such that accident events occur toward the end of the clip for positive samples. Additionally, all videos are downsampled to 10 frames per second.

For the DADA and DoTA datasets, we first filter videos annotated as ego-involved accidents. From these, we extract 5-second clips where the accident occurs within the final 2 seconds, with slight random variation in the exact onset time. These clips are used as positive samples. Since DADA and DoTA contain only accident videos, negative samples are constructed by extracting 5-second segments from portions of the videos that do not contain accidents. This sampling strategy follows the approach used in [30].

The dataset statistics are summarized in Table 4.1. All available ego-involved accident samples from DADA (938 videos) are included in the evaluation. For DoTA, only 745 videos are used, as this is the maximum number of valid negative samples that could be extracted given the available data. To maintain class balance, an equal number of positive samples is selected, consistent with previous studies [30]. Maintaining balanced datasets not only simplifies training but also prevents artificially inflated average precision scores caused by class imbalance [55].

Since predefined training and testing splits are not available for DADA and DoTA, we adopt a five-fold cross-validation strategy. To avoid information leakage, all clips originating from the same source video are assigned to the same fold. Figure 4.1 illustrates the distribution of crash-object categories in the selected DoTA and DADA samples.

For each dataset, negative samples are drawn from the same source as the positive samples rather than from external datasets. This ensures consistency in visual quality and prevents models from exploiting superficial differences between positive and negative samples to achieve misleadingly high performance [22, 30, 41]. For similar

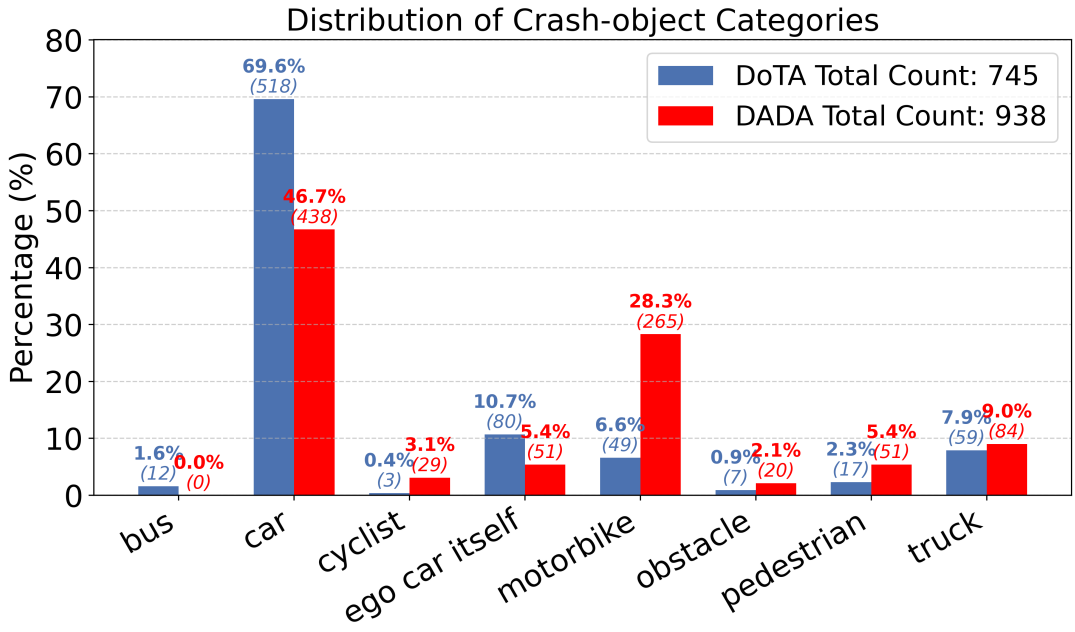


Fig. 4.1: Distribution of crash-object categories in the DoTA and DADA datasets. Crash-object annotations are not available for Nexar.

reasons, we exclude the commonly used CCD dataset [29], where accident videos are typically low-quality clips from online sources, while non-accident videos are high-quality recordings from BDD100k [56]. This discrepancy can lead to unrealistically high performance (often exceeding 99% average precision) due to quality bias rather than genuine accident anticipation capability, making CCD unsuitable for fair evaluation.

TABLE 4.1: Dataset Statistics.

Dataset	**Positive**	**Negative**	**FPS**
DAD [28]	620	1130	20
DoTA [13]	745	745	10
DADA [14]	938	1361	10
Nexar [12]	1422	1422	10

4.1.2 Networks and Baselines

4.1.2.1 STAGNet:

For object detection and feature extraction, we utilize pretrained YOLO [50] and VGG16 [26] models, respectively. Specifically, we consider the top 19 detected objects

in each frame, where each object is represented by a 4096-dimensional feature vector ($d_1 = 4096$). To incorporate semantic information, object class labels are encoded using GloVe embeddings [51] via the spaCy library, resulting in vectors of dimension $d_2 = 300$.

For temporal connectivity in the object graph, each node is linked only to its corresponding instance in the immediately preceding frame ($u = 1$). In the frame-level graph, we set the number of temporal neighbors to $k = 20$, following [21]. The same architectural configuration, as illustrated in Figure 4.2, is consistently applied across all datasets.

To extract global spatio-temporal features, we employ the SlowFast network [45], which produces a feature representation of dimension $h = 2304$ for each frame. All implementations for STAGNet are carried out using the PyTorch framework.

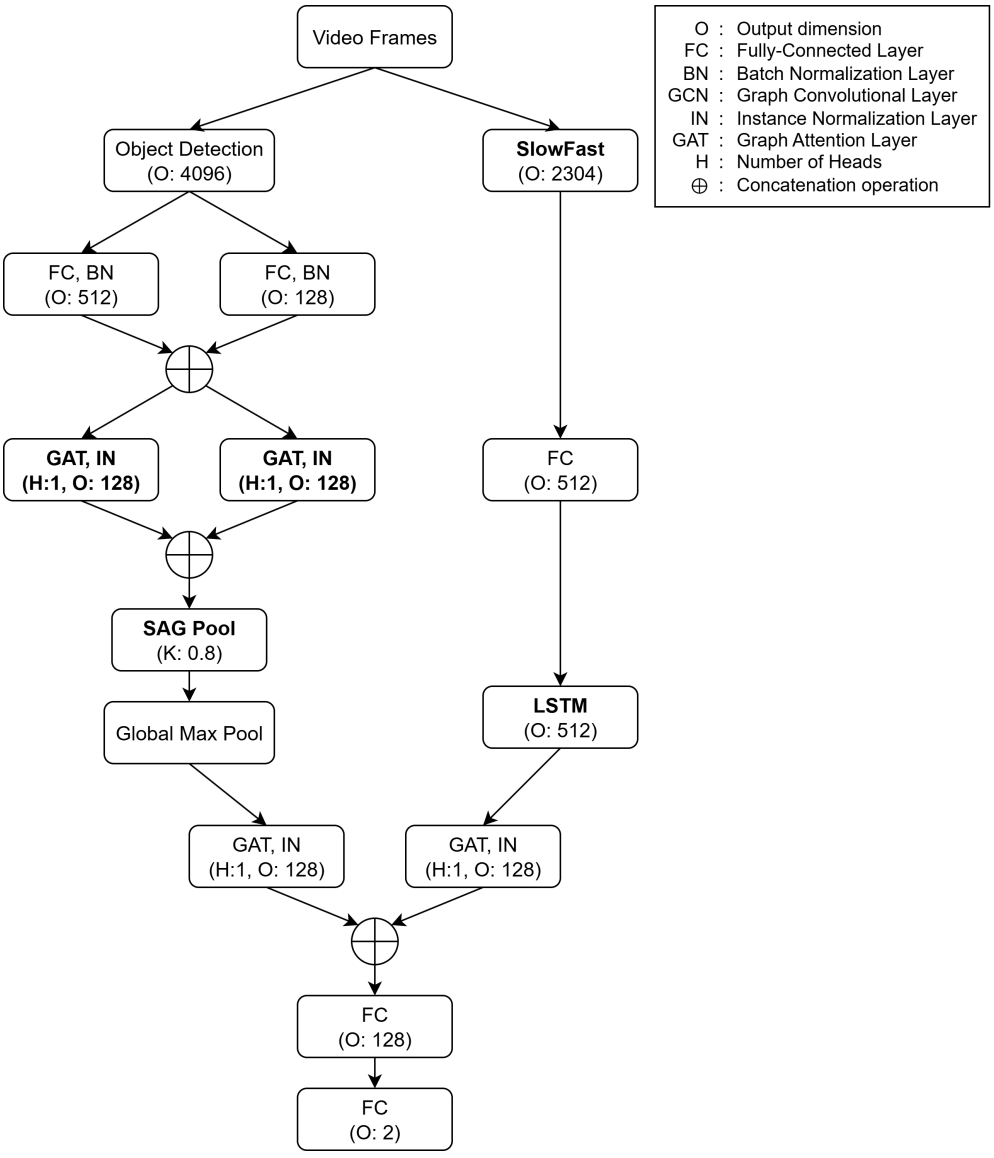


Fig. 4.2: Configuration of the proposed STAGNet accident anticipation framework.

4.1.2.2 VAGNet

For global scene representation, we adopt the VideoMAE-V2 architecture [27], which generates frame-level feature embeddings of dimension $h = 768$. The model takes as input the current frame along with the preceding $u = 15$ frames to capture temporal context. For constructing the frame-level graph, the number of temporal neighbors is set to $v = 20$.

4.1.2.3 Baselines and Experimental Setup

We compare our methods against several state-of-the-art approaches, including USTring [29], UString (Geometric), Graph(Graph) [21], AAT-DA [46], and STAGNet [44], evaluated on the DoTA, DADA, and Nexar datasets.

For UString [29] and Graph(Graph) [21], we use the official implementations provided by the respective authors. The UString (Geometric) variant is implemented as an approximation of GSTL [30], where VGG16-based object features are replaced with geometric features derived from bounding boxes.

Since the source code for AAT-DA is not publicly available, we re-implement the method based on the specifications reported in the original paper. In particular, we follow the commonly used feature extraction pipeline [22, 29], selecting the top 19 detected objects and extracting 4096-dimensional object-level and global features using VGG16. The hidden dimension and number of attention heads for all transformer modules are set to 1024 and 8, respectively. Dropout is applied with rates of 0.3, 0.3, 0.1, and 0.5 for the spatial transformer, object self-attention module, temporal transformer, and fully connected layer, respectively. Our re-implementation achieves comparable performance to the original results on the DAD dataset (63% vs. 64% AP), indicating a faithful reproduction.

All experiments are conducted on a system equipped with an NVIDIA Tesla P100 GPU (16 GB memory), a 2-core Intel Xeon CPU operating at 2.2 GHz, and 29 GB of RAM. VAGNet is implemented in PyTorch 2.0.0 and trained for 20 epochs using the Adam optimizer with a learning rate of 1×10^{-4} and a batch size of 1. This training configuration is consistently applied across all datasets.

4.1.3 Evaluation Metrics

The goal of this work is to anticipate traffic accidents both accurately and as early as possible. Earlier and reliable predictions provide more time for preventive actions, thereby improving road safety. At each frame of a video, the model produces a risk score representing the likelihood of a future accident. When this score exceeds a predefined threshold, the corresponding frame is classified as indicating a potential accident.

To evaluate prediction performance, we use the Average Precision (AP) metric, which summarizes the trade-off between precision and recall. Precision (P) and recall

(R) are defined as:

$$P = \frac{TP}{TP + FP}, \tag{4.1}$$

$$R = \frac{TP}{TP + FN}, \tag{4.2}$$

where TP, FP, and FN represent true positives, false positives, and false negatives, respectively. The Average Precision is computed as the area under the precision-recall curve:

$$AP = \int P_R \, dR, \tag{4.3}$$

where P_R denotes precision at a given recall level R.

In addition to accuracy, we evaluate how early the model can anticipate accidents using the mean Time-to-Accident (mTTA) metric. When the predicted risk score exceeds the threshold, the model assumes that an accident is imminent. The Time-to-Accident (TTA) is defined as the time difference between the first frame where the predicted score crosses the threshold and the actual start of the accident:

$$\text{TTA} = \max\{\tau - t \mid a_t \geq \hat{a},\ 1 \leq t \leq \tau\}, \tag{4.4}$$

where τ denotes the accident onset time, a_t is the predicted risk score at frame t, and $\hat{a}$ is the decision threshold.

Since TTA depends on the chosen threshold $\hat{a}$, multiple TTA values can be obtained by varying this threshold. To provide a more robust measure, we compute the mean Time-to-Accident as:

$$\text{mTTA} = \mathbb{E}[\text{TTA}]. \tag{4.5}$$

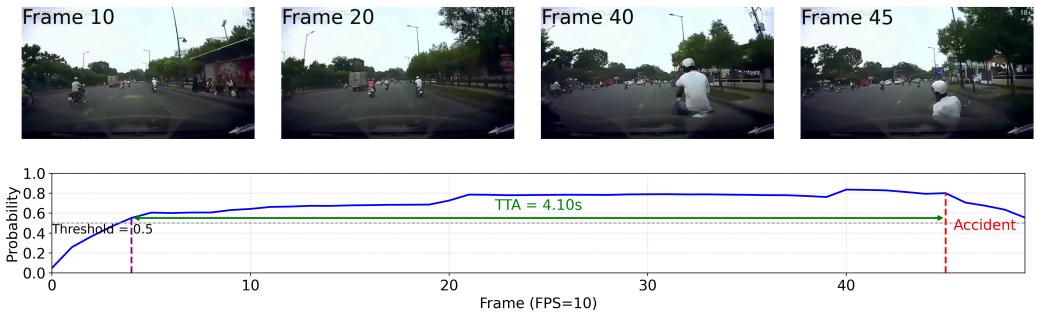
Compared to TTA evaluated at a fixed threshold, mTTA offers a more comprehensive assessment of early prediction capability. Higher mTTA values indicate that the model can anticipate accidents earlier, demonstrating stronger early-warning performance.

4.1.4 Results and Discussion

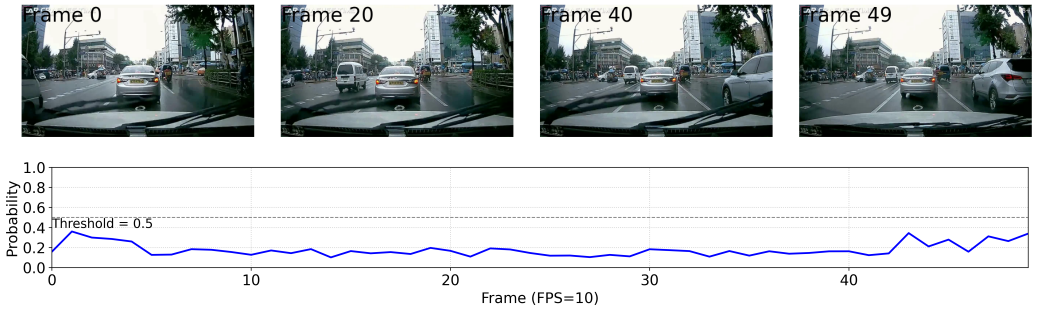
4.1.4.1 Comparison with State-of-the-art Methods

Table 4.2 summarizes the performance of the proposed VAGNet model in comparison with existing approaches across all four datasets. Overall, VAGNet achieves the highest Average Precision (AP) and competitive or superior mean Time-to-Accident (mTTA) in most cases. Minor exceptions are observed where AAT-DA and Graph(Graph) attain slightly higher mTTA on DoTA and DAD, respectively; however, these improvements come at the expense of lower AP.

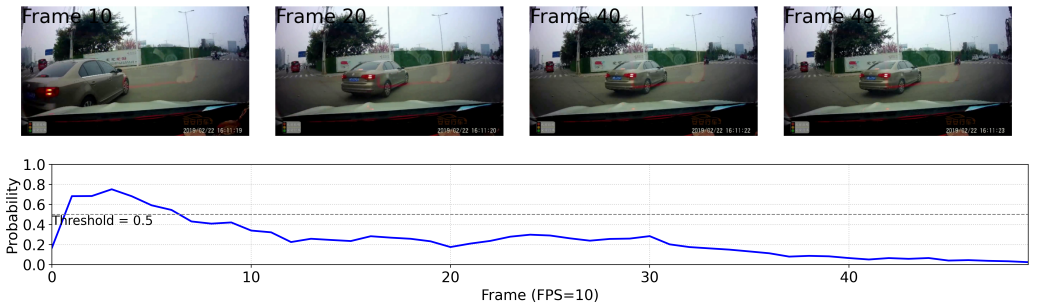
It is important to note that all prior methods, including strong baselines such as Graph(Graph) and STAGNet, rely on explicit object detection and object-level feature extraction. Some methods further incorporate additional cues such as monocular depth



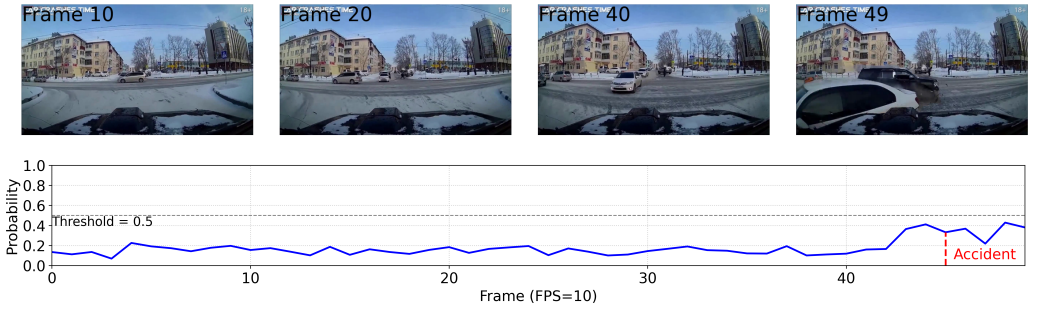
(a) Accident (true positive)



(b) Non-accident (true negative)

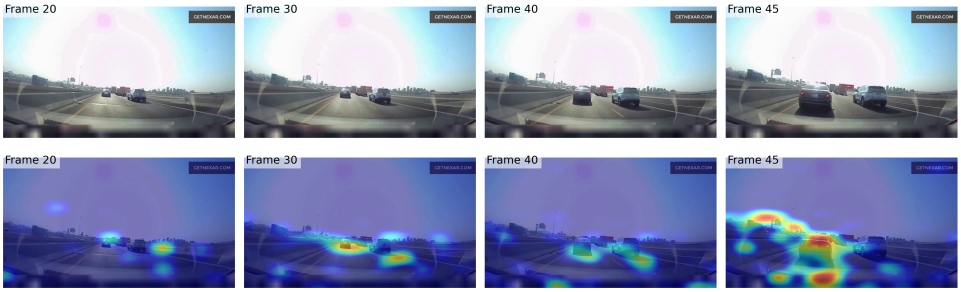


(c) Non-accident (false positive)

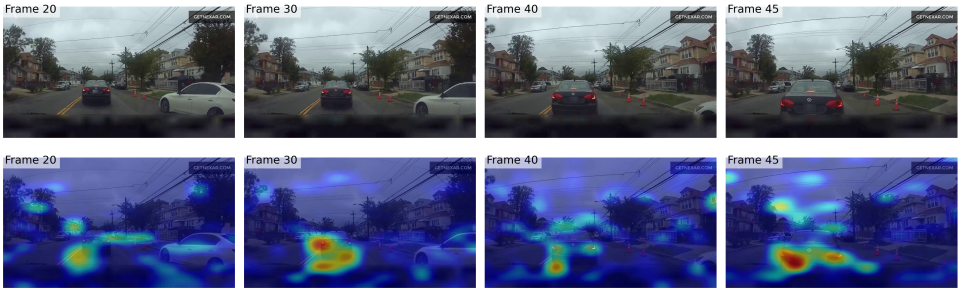


(d) Accident (false negative)

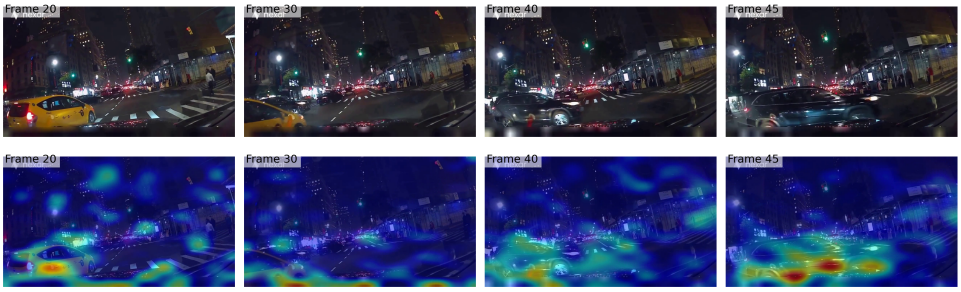
Fig. 4.3: Samples showing the variation of predicted accident probability across time/frames for (a) True positive (b) True negative (c) False positive and (d) False negative cases.



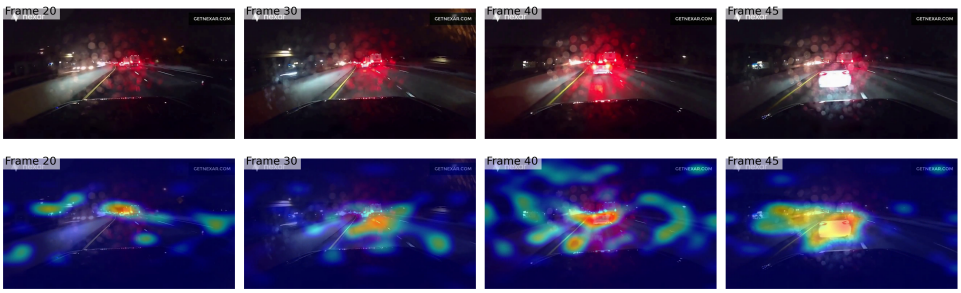
(a) Daytime, clear weather



(b) Daytime, cloudy weather



(c) Nighttime, clear weather



(d) Nighttime, rainy weather

Fig. 4.4: **Samples of visual explanations for accidents across different weather/lighting conditions.** For each condition, the top row shows the original video frames in temporal order, and the bottom row shows the corresponding Grad-CAM [57] maps highlighting areas in the image that contribute to accident anticipation. Higher intensity indicates regions that have a stronger influence on the model's anticipation of a potential accident.

estimation (e.g., CCAF-Net [19]) or driver attention modeling (e.g., AAT-DA [46]), which introduce additional computational overhead. In contrast, VAGNet eliminates the need for object-level processing by leveraging global spatio-temporal representations.

The comparison between STAGNet and VAGNet is particularly insightful. While STAGNet achieves strong performance through explicit modeling of object interactions, VAGNet consistently surpasses it across all datasets. This indicates that global representations learned by vision foundation models can effectively capture both contextual and relational information without requiring explicit object-level reasoning. A likely explanation is that such models implicitly encode object boundaries and interactions, as demonstrated in recent work on foundation models [58].

TABLE 4.2: Performance comparison of different methods.

Dataset	Method	AP (%)	mTTA (s)
DAD [28]	DSA (2017) [28]	48.1	1.34
	adaLEA (2018) [25]	52.3	3.43
	UString (2020) [29]	53.7	3.53
	DSTA (2022) [22]	56.1	3.66
	EGSTL (2023) [30]	56.3	3.23
	GSC (2024) [59]	60.4	3.53
	DAA-GNN (2024) [60]	75.2	1.47
	Graph(Graph) (2024) [21]	63.6	**4.45**
	AAT-DA (2025) [46]	64.0	2.87
	STAGNet (Ours)(2025) [44]	64.36	4.32
	CCAF-Net (2025) [19]	71.8	4.15
	Zhong et. al (2025) [20]	72.7	3.98
	VAGNet (Ours)	**77.98**	4.41
DoTA [13]	UString (Geometrics) [30]	63.2	2.14
	UString (2020) [29]	75.4	2.26
	Graph(Graph) (2024) [21]	83.5	3.14
	AAT-DA (2025) [46]	72.4	**3.28**
	STAGNet (Ours)(2025) [44]	92.73	3.23
	VAGNet(Ours)	**96.66**	3.22
DADA [14]	UString (Geometrics) [30]	56.8	1.25
	UString (2020) [29]	63.9	1.03
	Graph(Graph) (2024) [21]	84.0	2.92
	AAT-DA (2025) [46]	69.4	1.36
	STAGNet (Ours)(2025) [44]	95.39	3.08
	VAGNet (Ours)	**97.82**	**3.09**
Nexar [12]	UString (Geometrics) [30]	71.0	3.33
	UString (2020) [29]	73.3	2.66
	Graph(Graph) (2024) [21]	60.4	4.61
	AAT-DA (2025) [46]	75.1	2.50
	STAGNet (Ours)(2025) [44]	76.5	3.72
	VAGNet (Ours)	**81.25**	**4.69**

Figure 4.3 illustrates the temporal evolution of predicted accident probabilities. In true positive cases, the model successfully identifies hazardous situations well before the actual accident occurs. False positive and false negative examples highlight the inherent difficulty of the task, where near-miss events can appear visually similar to actual collisions from a dash-cam perspective.

To further understand model behavior, Grad-CAM visualizations are analyzed in Figure 4.4. The results show that the model consistently attends to relevant regions, particularly nearby vehicles and potential collision zones, indicating that meaningful spatial cues are being captured.

4.1.4.2 Impact of Global Features

The influence of different global feature extraction backbones is presented in Table 4.3. A clear trend is observed as performance improves when transitioning from image-based models (VGG16) to spatio-temporal architectures (I3D and SlowFast), emphasizing the importance of temporal information in accident anticipation.

Among all evaluated backbones, VideoMAE-V2 achieves the best performance across all datasets, particularly on DAD and Nexar, where improvements are substantial. This can be attributed to its large-scale pretraining as a vision foundation model, enabling it to capture complex spatial and temporal dependencies effectively.

Interestingly, several prior methods that employ relatively simpler feature extractors still achieve competitive AP performance. For example, CCAF-Net and AAT-DA surpass STAGNet on certain datasets, although their performance remains below that of VAGNet. However, these approaches depend on additional components such as depth estimation or driver attention mechanisms, which introduce extra computational overhead. In contrast, VAGNet attains superior performance using only global features, underscoring the effectiveness of its learned representations.

To further assess the impact of improved global representations, we evaluated the USTring framework by replacing its original VGG16-based features with those extracted from SlowFast networks. As shown in Table 4.4, this modification leads to a clear improvement in performance, particularly on the DoTA dataset. These results indicate that incorporating richer spatio-temporal context through SlowFast features enhances the effectiveness of the UString model.

4.1.4.3 Ablation Study - STAGNet

The ablation results in Table 4.6 highlight the contribution of each component—LSTM, GATv2, and SlowFast—to the overall performance of the proposed accident anticipation framework.

From the results, it is clear that removing any of these components leads to a reduction in average precision (AP) across both datasets. However, the use of SlowFast features and the inclusion of an LSTM module have a significantly greater impact compared to the incorporation of GATv2 in the object graph module. For example, on

TABLE 4.3: Performance of VAGNet under different global feature extraction backbones.

Dataset	Global Features	AP (%)	mTTA (s)
DAD	VGG16	54.83	4.15
	I3D	59.27	4.13
	SlowFast	62.90	3.74
	VideoMAE-V2	**77.98**	**4.41**
DoTA	VGG16	88.74	3.25
	I3D	91.71	3.20
	SlowFast	95.06	**3.26**
	VideoMAE-V2	**96.66**	3.22
DADA	VGG16	83.61	3.12
	I3D	90.57	3.09
	SlowFast	95.83	**3.18**
	VideoMAE-V2	**97.82**	3.09
Nexar	VGG16	62.89	4.48
	I3D	65.3	4.43
	SlowFast	73.89	4.67
	VideoMAE-V2	**81.25**	**4.69**

TABLE 4.4: Performance of UString using global features extracted from SlowFast networks.

Dataset	Method	Global Feature	AP (%)	mTTA (s)
DoTA [13]	UString [29]	VGG16 [26]	75.40	2.26
		SlowFast [45]	86.38	1.90
DADA [14]	UString [29]	VGG16 [26]	63.95	1.03
		SlowFast [45]	66.14	1.11

TABLE 4.5: Results of the ablation study on the DADA and DoTA datasets. Each model variant removes one or more components (LSTM, GATv2, or SlowFast) to analyze their individual impact. Note: When GATv2 was not used, GCN layers were used. When SlowFast features were not used, I3D features were used.

Data	LSTM	GATv2	SlowFast	AP (%)	mTTA (s)
DoTA [13]	✗	✗	✗	83.50	3.14
	✓	✗	✗	89.98	3.06
	✗	✓	✗	85.89	3.13
	✗	✗	✓	90.03	2.97
	✓	✓	✗	88.52	2.98
	✓	✗	✓	92.18	3.12
	✗	✓	✓	90.75	3.13
	✓	✓	✓	**92.73**	**3.23**
DADA [14]	✗	✗	✗	84.04	2.92
	✓	✗	✗	89.18	2.89
	✗	✓	✗	84.65	2.96
	✗	✗	✓	92.95	2.89
	✓	✓	✗	89.40	2.92
	✓	✗	✓	94.75	3.04
	✗	✓	✓	93.51	2.79
	✓	✓	✓	**95.39**	**3.08**

the DoTA dataset, the baseline model without any of the three components achieves an AP of 83.50%. Introducing the LSTM alone improves performance substantially to 89.98%, emphasizing the importance of temporal modeling. Similarly, replacing I3D with SlowFast features increases the AP to 90.03%, demonstrating the benefit of stronger spatio-temporal representations. When both LSTM and SlowFast are combined, the performance further improves to 92.18%. The best results are obtained when all three components are included, achieving 92.73% AP and an mTTA of 3.23

seconds.

A similar trend is observed on the DADA dataset. The baseline model achieves 84.04% AP, which increases to 89.18% with the addition of LSTM. Using SlowFast features alone results in a notable improvement to 92.95% AP. The highest performance is again achieved when all components are combined, reaching 95.39% AP with an mTTA of 3.08 seconds.

In terms of early prediction performance, the results indicate that removing any component reduces mTTA. Moreover, applying individual modifications alone does not consistently improve mTTA over the baseline Graph(Graph) model.

Overall, the ablation study suggests that SlowFast features and LSTM contribute significantly to improving prediction accuracy, while GATv2 provides additional but relatively smaller gains. However, achieving improvements in early accident anticipation (mTTA) requires the combined effect of all three components.

4.1.4.4 Ablation Study - VAGNet

The ablation results in Table 4.6 analyze the contributions of the Transformer Encoder (TE) and Graph Transformer (GT) modules. Removing both components reduces the model to a simple pipeline consisting of VideoMAE-V2 features followed by fully connected layers, resulting in a significant drop in both AP and mTTA across all datasets. This confirms the importance of modeling temporal and relational dependencies.

Introducing either TE or GT individually leads to substantial improvements in performance. The TE primarily enhances temporal modeling, while the GT captures relational dependencies across frames. When both components are combined, the model achieves the best overall performance, although the improvement over individual components is relatively modest. This suggests that TE and GT capture overlapping yet complementary aspects of accident dynamics.

4.1.4.5 Analysis of Computational Efficiency

Since real-time performance is essential for practical accident anticipation systems, we analyze computational efficiency in terms of FLOPs and inference speed. Tables 4.7 and 4.8 provide a detailed breakdown.

A key observation is that object detection and object-level feature extraction dominate the computational cost in existing methods. This overhead increases with the number of detected objects and significantly impacts runtime performance. In contrast, VAGNet avoids this cost entirely by relying solely on global features.

Although VideoMAE-V2 requires more FLOPs than some traditional feature extractors, the overall pipeline of VAGNet remains more efficient due to the elimination of object-level processing. Interestingly, despite its higher FLOP count, VideoMAE-V2 achieves faster inference than other spatio-temporal models such as SlowFast and I3D. This discrepancy can be attributed to differences in memory access patterns and implementation optimizations, as reported in prior studies [61, 62].

TABLE 4.6: Results of the ablation study. Each model variant removes one or more components (Transformer Encoder or Graph) to analyze their individual impact.

Data	Transformer Encoder	Graph Transformer	AP (%)	mTTA (s)
DAD [28]	✗	✗	64.15	2.54
	✓	✗	76.58	4.35
	✗	✓	71.37	4.12
	✓	✓	**77.98**	**4.41**
DoTA [13]	✗	✗	87.50	2.45
	✓	✗	95.87	**3.31**
	✗	✓	95.00	3.12
	✓	✓	**96.66**	3.22
DADA [14]	✗	✗	75.43	1.83
	✓	✗	97.07	**3.16**
	✗	✓	97.10	3.10
	✓	✓	**97.82**	3.09
Nexar [12]	✗	✗	76.34	3.11
	✓	✗	78.02	**4.73**
	✗	✓	80.15	4.13
	✓	✓	**81.25**	4.69

Furthermore, VAGNet achieves the highest frame rate (FPS) among all evaluated methods. Another important advantage is that its computational cost is independent of the number of objects in the scene, making it more scalable for complex and crowded environments.

TABLE 4.7: FLOP counts of different methods. Counts for different stages are reported in GFLOPs/frame. The counts were obtained using the Python library THOP [63].

Method	**GFLOPs**			
	Object Detection & Feature Extraction	**Global Feature Extraction**	**Model Inference**	**Total**
UString	322.46 (worst case: 19 objects/frame)	15.47 (VGG16)	10.49	348.42
Graph (Graph)		41.75 (I3D)	2.14	366.35
Zhong [20]		41.75 (I3D)	2.05	366.26
AAT-DA		15.47 (VGG16)	1.05	461.96*
STAGNet		50.58 (SlowFast)	2.20	375.24
UString	123.25 (average case: 6.12 objects/frame)	15.47 (VGG16)	10.49	149.21
Graph (Graph)		41.75 (I3D)	2.14	167.14
Zhong [20]		41.75 (I3D)	2.05	167.05
AAT-DA		15.47 (VGG16)	1.05	262.67*
STAGNet		50.58 (SlowFast)	2.20	176.03
VAGNet	**0**	101.85 (VideoMAE-V2)	**0.033**	**101.88**
		50.58 (SlowFast)		**50.51**
		41.75 (I3D)		**41.78**
		15.47 (VGG16)		**15.50**

* The total includes the 122.98 GFLOPs used for saliency/driver-attention computation.

The proposed accident anticipation frameworks have practical applicability in Advanced Driver Assistance Systems (ADAS), autonomous vehicles, and intelligent transportation systems, where early prediction of hazardous situations can enable timely preventive actions. By providing advance warnings of potential collisions, these systems can assist drivers in avoiding accidents and reducing the severity of road incidents. Furthermore, the proposed VAGNet model is computationally efficient and relies solely on dash-cam video, making it suitable for deployment in cost-sensitive environments and existing vehicles without requiring additional sensors. Its low computational overhead and real-time performance also make it a promising candidate for deployment on edge devices, enabling on-board accident anticipation without depen-

TABLE 4.8: The computational efficiency of the proposed and baseline methods is evaluated in terms of execution time. Module-level performance is reported in seconds per frame (s/f), whereas overall inference efficiency is reported as frames per second (FPS). For AAT-DA, the reported inference time additionally accounts for the generation of driver attention maps. Note that although we do not report the inference time for the method proposed by Zhong et. al [20] due to the unavailability of its implementation, we expect it to operate close to 36 FPS on our platform as it uses VGG16 and I3D as feature extractors.

Method	**Object Detection & Feature Extraction (s/f)**	**Global Feature Extraction (s/f)**	**Inference (s/f)**	**Overall speed (FPS)**
UString	0.032	0.004 (VGG16)	0.024	17
Graph(Graph)		0.01 (I3D)	0.001	23
AAT-DA		0.004 (VGG16)	0.094*	8
STAGNet		0.014 (SlowFast)	0.001	21
VAGNet	0	0.025 (VideoMAE)	0.001	**38**

* 0.009 s for model inference + 0.085 s for generating the saliency (driver-attention) map.

dence on cloud-based processing while preserving privacy and reducing communication latency.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Limitations and Future work

Our approach, similar to most methods in accident anticipation research, relies on dash-cam datasets collected from online sources and vehicle fleets. These datasets enhance model generalization by incorporating data from diverse ego-vehicles, camera setups, and varying environmental and weather conditions. However, further performance improvements could be achieved if the model is fine-tuned after deployment on a specific ego-vehicle. Such post-deployment adaptation would allow the model to better align with the unique viewpoint and physical characteristics of the vehicle.

In practice, this type of fine-tuning is challenging because it would require accident data from the deployed vehicle itself, which contradicts the primary goal of preventing such events. Consequently, current approaches are not designed for post-deployment adaptation. Future research could address this limitation by exploring strategies that enable model fine-tuning using only normal driving data (i.e., negative samples) collected during routine operation.

Additionally, although the proposed method reduces overall computational cost and achieves faster inference compared to existing approaches, the VideoMAE-V2 backbone (base version) still incurs a computational cost of approximately 102 GFLOPs per frame. Future work may investigate knowledge distillation techniques or alternative lightweight architectures to further improve efficiency while maintaining competitive predictive performance.

5.2 Conclusion

In this work, we investigated accident anticipation from dash-cam videos through two complementary approaches, namely STAGNet and VAGNet. STAGNet models accident dynamics by explicitly capturing object-level interactions using spatio-temporal graph representations, while VAGNet introduces a more efficient paradigm based on global scene features without requiring explicit object detection.

The proposed VAGNet framework is computationally efficient, as it implicitly captures object-level information through global representations, eliminating the overhead associated with object-level feature extraction. Experimental evaluations on four benchmark datasets demonstrate that both STAGNet and VAGNet achieve strong performance, with VAGNet consistently outperforming existing methods in terms of both prediction accuracy and earliness.

Ablation studies further reveal that the performance improvements of VAGNet are driven by the integration of transformer-based temporal modeling and graph-based relational learning, along with the use of rich spatio-temporal features extracted using

the VideoMAE-V2 foundation model. Overall, this work highlights the effectiveness of transitioning from explicit object-centric modeling (STAGNet) to global feature-based representations (VAGNet) for scalable and accurate accident anticipation.

REFERENCES

[1] "Road traffic injuries," https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries, 2023, [Online; accessed 24-December-2025].

[2] F. L. Mannering, V. Shankar, and C. R. Bhat, "Unobserved heterogeneity and the statistical analysis of highway accident data," *Analytic methods in accident research*, vol. 11, pp. 1–16, 2016.

[3] M. M. Karim, Y. Li, and R. Qin, "Toward explainable artificial intelligence for early anticipation of traffic accidents," *Transportation research record*, vol. 2676, no. 6, pp. 743–755, 2022.

[4] T. Zhang, Z. Wang, H. Wang, and J. Li, "Intelligent defensive driving for autonomous vehicles: Framework, strategy and verification," *Accident Analysis & Prevention*, vol. 226, p. 108355, 2026.

[5] K. Mattas, G. Albano, R. Donà, M. C. Galassi, R. Suarez-Bertoa, S. Vass, and B. Ciuffo, "Driver models for the definition of safety requirements of automated vehicles in international regulations. application to motorway driving conditions," *Accident Analysis & Prevention*, vol. 174, p. 106743, 2022.

[6] D. Li, Y. Jiang, J. Zhang, and B. Xiao, "SMPC-Based Motion Planning of Automated Vehicle When Interacting With Occluded Pedestrians," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[7] J. Liu, X. Yan, L. Peng, L. Yang, L. Zhang, Y. Luo, Y. Tao, A. Y. X. Tan, M. Li, L. Zhang *et al.*, "Seeing before observable: Potential risk reasoning in autonomous driving via vision language models," *arXiv preprint arXiv:2511.22928*, 2025.

[8] S. Tekkesinoglu, A. Habibovic, and L. Kunze, "Advancing explainable autonomous vehicle systems: A comprehensive review and research roadmap," *ACM Transactions on Human-Robot Interaction*, vol. 14, no. 3, pp. 1–46, 2025.

[9] P. S. Chib and P. Singh, "Recent advancements in end-to-end autonomous driving using deep learning: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 103–118, 2023.

[10] H. X. Liu and S. Feng, "Curse of rarity for autonomous vehicles," *nature communications*, vol. 15, no. 1, p. 4808, 2024.

[11] Y. Zhang, B. Kang, B. Hooi, S. Yan, and J. Feng, "Deep long-tailed learning: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 9, pp. 10 795–10 816, 2023.

[12] D. Moura, S. Zhu, and O. Zvitia, "Nexar dashcam collision prediction dataset and challenge," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 2583–2591.

[13] Y. Yao, X. Wang, M. Xu, Z. Pu, Y. Wang, E. Atkins, and D. Crandall, "DoTA: unsupervised detection of traffic anomaly in driving videos," *IEEE transactions on pattern analysis and machine intelligence*, 2022.

[14] J. Fang, D. Yan, J. Qiao, J. Xue, H. Wang, and 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.

[15] S.-Y. Yu, A. V. Malawade, D. Muthirayan, P. P. Khargonekar, and M. A. Al Faruque, "Scene-graph augmented data-driven risk assessment of autonomous vehicle decisions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7941–7951, 2021.

[16] "Retrofit Collision Warning System Gives Older Vehicles A Safety Boost," https://trid.trb.org/View/1574810, 2018, [Online; accessed 27-October-2025].

[17] D. B. Yoffie, "Mobileye: The future of driverless cars," *Harvard Business School Case*, pp. 715–421, 2014.

[18] J. Fang, J. Qiao, J. Xue, and Z. Li, "Vision-based traffic accident detection and anticipation: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[19] W. Liu, Y. Li, T. Zhang, Y. Gao, L. Wei, and J. Chen, "CCAF-Net: Cascade Complementarity-Aware Fusion Network for traffic accident prediction in dashcam videos," *Neurocomputing*, vol. 624, p. 129285, 2025.

[20] Y. Zhong, G. Yan, R. Zhu, P. Gan, and X. Shen, "Early traffic accident anticipation via feature consistency representation and soft label regression," *ACM Transactions on Multimedia Computing, Communications and Applications*, 2025.

[21] N. Thakur, P. Gouripeddi, and B. Li, "Graph (graph): A nested graph-based framework for early accident anticipation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 7533–7541.

[22] M. M. Karim, Y. Li, R. Qin, and Z. Yin, "A dynamic spatial-temporal attention network for early anticipation of traffic accidents," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9590–9600, 2022.

[23] A. V. Malawade, S.-Y. Yu, B. Hsu, D. Muthirayan, P. P. Khargonekar, and M. A. Al Faruque, "Spatiotemporal scene-graph embedding for autonomous vehicle

collision prediction," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9379–9388, 2022.

[24] W. Bao, Q. Yu, and 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.

[25] T. Suzuki, H. Kataoka, Y. Aoki, and Y. Satoh, "Anticipating traffic accidents with adaptive loss and large-scale incident db," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3521–3529.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[27] L. Wang, B. Huang, Z. Zhao, Z. Tong, Y. He, Y. Wang, Y. Wang, and Y. Qiao, "Videomae v2: Scaling video masked autoencoders with dual masking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 14 549–14 560.

[28] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun, "Anticipating accidents in dashcam videos," in *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part IV 13*. Springer, 2017, pp. 136–153.

[29] W. Bao, Q. Yu, and Y. Kong, "Uncertainty-based traffic accident anticipation with spatio-temporal relational learning," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2682–2690.

[30] F. Mahmood, D. Jeong, and J. Ryu, "A new approach to traffic accident anticipation with geometric features for better generalizability," *IEEE Access*, vol. 11, pp. 29 263–29 274, 2023.

[31] H. Jung, M.-K. Choi, K. Soon, and W. Y. Jung, "End-to-end pedestrian collision warning system based on a convolutional neural network with semantic segmentation," in *2018 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2018, pp. 1–3.

[32] H. Kim, K. Lee, G. Hwang, and C. Suh, "Predicting vehicle collisions using data collected from video games," *Machine Vision and Applications*, vol. 32, no. 4, p. 93, 2021.

[33] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," in *2008 11th international conference on information fusion*. IEEE, 2008, pp. 1–6.

[34] M. Strickland, G. Fainekos, and H. B. Amor, "Deep predictive models for collision risk assessment in autonomous driving," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4685–4692.

[35] Y. Gal *et al.*, "Uncertainty in deep learning," 2016.

[36] E. Rohmer, S. P. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *2013 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2013, pp. 1321–1326.

[37] M. Fatima, M. U. K. Khan, and C.-M. Kyung, "Global feature aggregation for accident anticipation," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 2809–2816.

[38] G.-P. Corcoran and 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.

[39] C. Yi, T.-J. Huang, H.-J. Ye, and D.-C. Zhan, "Improved dynamic spatial-temporal attention network for early anticipation of traffic accidents," in *2023 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*. IEEE, 2023, pp. 81–86.

[40] I. Cho, P. K. Rajendran, T. Kim, and D. Har, "Reinforcement learning for predicting traffic accidents," in *2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2023, pp. 684–688.

[41] C. Chitraranjan, V. Vipulananthan, and T. Sritharan, "Vision-based collision warning systems with deep learning: A systematic review," *Journal of Imaging*, vol. 11, no. 2, p. 64, 2025.

[42] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *Advances in neural information processing systems*, vol. 28, 2015.

[43] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[44] V. Vipulananthan, K. Mohottala, K. Chinthana, N. Paramulla, and C. Chitraranjan, "STAGNet: A Spatio-Temporal Graph and LSTM Framework for Accident Anticipation," *IEEE Access*, vol. 13, pp. 213 769–213 779, 2025.

[45] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6201–6210, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:54463801

[46] "AAT-DA: Accident Anticipation Transformer with Driver Attention, author=Kumamoto, Yuto and Ohtani, Kento and Suzuki, Daiki and Yamataka, Minori and Takeda, Kazuya," in *Proceedings of the Winter Conference on Applications of Computer Vision*, 2025, pp. 1142–1151.

[47] T. Zhao, X. Bai, J. Fang, and J. Xue, "Gated driver attention predictor," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 270–276.

[48] L.-L. Li, J. Fang, and J. Xue, "Cognitive traffic accident anticipation," *IEEE Intelligent Transportation Systems Magazine*, vol. 16, no. 5, pp. 17–32, 2024.

[49] J. Fang, L.-L. Li, K. Yang, Z. Zheng, J. Xue, and T.-S. Chua, "Cognitive accident prediction in driving scenes: A multimodality benchmark," *arXiv preprint arXiv:2212.09381*, 2022.

[50] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "Yolov10: Real-time end-to-end object detection," 2024. [Online]. Available: https://arxiv.org/abs/2405.14458

[51] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[52] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?" *arXiv preprint arXiv:2105.14491*, 2021.

[53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[54] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, "Masked label prediction: Unified message passing model for semi-supervised classification," 2021. [Online]. Available: https://arxiv.org/abs/2009.03509

[55] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PloS one*, vol. 10, no. 3, p. e0118432, 2015.

[56] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2636–2645.

[57] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[58] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa *et al.*, "Dinov3," *arXiv preprint arXiv:2508.10104*, 2025.

[59] T. Wang, K. Chen, G. Chen, B. Li, Z. Li, Z. Liu, and C. Jiang, "Gsc: A graph and spatio-temporal continuity based framework for accident anticipation," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 2249–2261, 2023.

[60] W. Song, S. Li, T. Chang, K. Xie, A. Hao, and H. Qin, "Dynamic attention augmented graph network for video accident anticipation," *Pattern Recognition*, vol. 147, p. 110071, 2024.

[61] H.-I. Liu, M. Galindo, H. Xie, L.-K. Wong, H.-H. Shuai, Y.-H. Li, and W.-H. Cheng, "Lightweight deep learning for resource-constrained environments: A survey," *ACM Computing Surveys*, vol. 56, no. 10, pp. 1–42, 2024.

[62] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.

[63] "THOP: A tool to count the FLOPs of PyTorch model." https://pypi.org/project/thop/, 2022, [Online; accessed 14-November-2025].