

Article

Not peer-reviewed version

TU-DAT: A Computer Vision Dataset on Road Traffic Anomalies

[Pavana Pradeep Kumar](#)^{*} and [Krishna Kant](#)^{*}

Posted Date: 28 April 2025

doi: 10.20944/preprints202504.2344.v1

Keywords: Intelligent Transport Systems; Anomaly Detection in Road Traffic



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

TU-DAT: A Computer Vision Dataset on Road Traffic Anomalies

Pavana Pradeep Kumar ^{1,*}, Krishna Kant ^{2,*}

Temple University

* Correspondence: pavana.pradeep@temple.edu (P.P.K.); kkant@temple.edu (K.K.)

[†] These authors contributed equally to this work.

Abstract: This paper introduces a novel, freely downloadable computer vision dataset for analyzing traffic accidents using roadside cameras. Our objective is to address the lack of public data for research concerning the automatic detection and prediction of road anomalies and accidents to enhance traffic safety. The Temple University Data on Anomalous Traffic (TU-DAT) dataset comprises various accident videos from news reporting and documentary websites. To guarantee the applicability of our method to roadside edge devices, we exclusively utilize footage and images from traffic CCTV cameras. We have collected approximately 210 videos, ranging from 24 to 30 FPS, depicting road accidents, comprising 17,255 accident keyframes and 505,245 standard frames. Analysis of the TU-DAT dataset revealed a significant finding. Due to the challenges in acquiring real-world traffic videos to analyze aggressive driving, we used a game simulator to produce road traffic video data that emulates aggressive driving behaviors, including speeding, tailgating, weaving through traffic, and disregarding red lights. We collected around 40 videos of positive instances and 25 videos of negative cases. We have already used this dataset in several contexts where we integrate deep learning with explicit spatiotemporal logic reasoning and demonstrate substantial outperformance over pure deep-learning methods in accuracy and running time. We hope it will be used innovatively for computer vision research.

Dataset: The TU-DAT dataset is accessible via GitHub at the following URL <https://github.com/pavana27/TU-DAT>. The dataset should be used only for research purposes and may not be used for profit either as it stands or with repackaging or modifications. The dataset is offered without any liability regarding any consequences resulting from using this data.

Keywords: intelligent transport systems; anomaly detection in road traffic

1. Introduction

1.1. Need for Accident Datasets

The National Highway Traffic Safety Administration (NHTSA) estimates that about 29,135 people died in car crashes in the first nine months of 2024 [1]. The department-wide adoption of the safe system approach is the foundation of the NRSS's implementation and is essential for resolving the fatality crisis on our roads. Roadside camera can nonintrusively monitor the traffic and their real-time analysis followed by alerts to the driver (including desired maneuvers in cases of high risks of accidents) can make the roads safer.

Unfortunately, datasets for accident detection and prediction are limited by several factors: (i) traffic accidents are rare, making it impractical to gather sufficient data through prolonged recording at intersections, and (ii) legal and privacy restrictions complicate access to traffic camera footage. Due to the challenges in collecting real-world traffic accident videos—such as the rarity of events concerning accidents—many traffic cameras are operated by government agencies or private entities, making it challenging to obtain permissions or licenses to access the data.

Accident datasets are invaluable assets that underpin various aspects of road safety analysis, traffic management, and the advancement of intelligent transportation systems (ITS). The increasing

complexity of modern traffic systems necessitates a robust understanding of road safety dynamics. Accident datasets, which encapsulate detailed information about traffic collisions—including environmental conditions, human behavior, and vehicle performance—play a pivotal role in discerning patterns that contribute to road incidents. Accident datasets provide empirical evidence that aids researchers and policymakers in uncovering high-risk locations and common causes of accidents. By examining patterns in large-scale data, stakeholders can devise targeted interventions to increase road safety. For instance, these datasets can reveal environmental factors, such as weather conditions or road quality, that influence collision frequency. The work of [2] emphasizes that without these data, strategies to improve traffic safety would rely heavily on theoretical assumptions rather than solid evidence derived from real-world occurrences.

In recent years, the integration of machine learning techniques with accident datasets has gained significance. By training predictive models on historical accident data, researchers can identify accident-prone areas and recommend preventive strategies. Several studies explore how large-scale accident data and deep learning enhance autonomous vehicle safety. These datasets help train machine learning models to predict and mitigate hazards, improving decision-making in self-driving systems. By analyzing crash patterns, autonomous vehicles can adapt to risk factors like road conditions, weather, and driver behavior. Authors in [3] introduces the Integrated Ensemble Learning-Logit Model (IELLM) to improve accident severity prediction. This model integrates multiple machine learning techniques to assess risk levels and anticipate crashes in real time. Leveraging accident datasets, it enhances autonomous systems' ability to navigate complex traffic scenarios by considering factors like speed, traffic density, and environmental conditions. These advancements help self-driving cars make data-driven decisions, reducing accidents and improving road safety.

Further, accident datasets are integral to fostering a culture of safety on the roads. By facilitating evidence-based decision-making, they support advancements in technology, policy formulation, and urban planning. Without comprehensive and high-quality accident datasets, safety improvements would rely primarily on theoretical models rather than empirical evidence, limiting their real-world effectiveness [4].

This paper introduces TU-DAT, a dataset annotated with spatiotemporal data, specifically designed for analyzing traffic accidents. TU-DAT is sourced from a variety of channels, including surveillance camera footage, publicly available accident videos on YouTube, and crash scenarios generated synthetically using a high-fidelity game simulation environment. By combining a diverse range of accident recordings, TU-DAT provides a comprehensive and multi-modal dataset that captures both real-world and simulated traffic incidents, making it an invaluable resource for traffic safety research, accident prediction modeling, and training for autonomous vehicles.

A key feature of TU-DAT is its spatiotemporal annotation, which allows for precise tracking of accident dynamics, vehicle interactions, and environmental conditions over time. The dataset includes detailed metadata such as vehicle trajectories, collision points, impact severity levels, road types, and weather conditions. This information offers researchers critical insights into the causes and progression of accidents. The surveillance camera footage provides an objective, fixed-angle view of real-world crashes, while the YouTube-sourced videos present a variety of accident scenarios captured from dash-cams and street cameras. The simulated crashes, created with a physics-accurate game engine, enable controlled scenario testing and augment real-world data with synthetic examples to help train machine-learning models for accident detection and severity estimation. We also illustrate 3 different ways in which we have already used this dataset in our past work.

2. Related Work

2.1. Datasets for Modeling Road Accidents

With the advancement of autonomous driving concepts and smart cities, recent research has focused on monitoring traffic safety using computer vision/ deep learning techniques. In [5], the authors present a novel large-scale benchmark dataset named Detection of Traffic Anomaly (DoTA) to assess

traffic accident detection and anomaly detection across nine distinct action categories, comprising 4,677 videos with temporal, spatial, and categorical annotations. A metric for prediction consistency was developed to calculate anomaly scores resilient to inaccurate object detection and tracking in driving videos. However, this dataset highlights a scarcity of extensively annotated, real-life accident datasets.

Authors in [6] propose a novel CCTV traffic camera-based accident analysis (CADP) dataset. It comprises 1416 accident recordings captured at different times of the day and under various weather conditions from countries worldwide. A total of 7000 frames were utilized for training the model, comprising approximately 3500 accident frames and 3500 non-accident frames from these videos. This paper also introduces a novel and efficient approach to detecting road accidents using Random Forest Classifiers and DETR (Detection Transformers). The DETR is employed to identify objects in the CCTV footage, including cars, bikes, and individuals. The features are then passed to a Random Forest Classifier for frame-wise classification. Each frame of the video is categorized as either an accident frame or a non-accident frame.

Ref [7] provides a compilation of vehicular collision footage from various geographical regions sourced from YouTube. The surveillance videos are evaluated at a frame rate of 30 frames per second (FPS). In order to incorporate the frames containing accidents, the video clips are compressed to approximately 20 seconds. The dataset encompasses accidents captured from CCTV videos at road intersections that occurred in a variety of ambient conditions, including harsh sunlight, daylight hours, snow, and nighttime.

QMUL Junction Dataset [8], collected concentrating on a singular surveillance camera located on a suburban street, consisting of a surveillance video situated at a busy traffic intersection. The duration of videos is approximately 52 minutes, comprising 78,000 frames with a resolution of 360×288 pixels. The video's content is rather complex, with the background comprising four roads, two sidewalks, and numerous adjacent buildings. Each scene features numerous vehicles, bicycles, and pedestrians, accompanied by significant variations in lighting. Simultaneously, several activities occur, such as vehicles proceeding straight, executing turns, halting for red lights, and pedestrians traversing the roadway. Authors in [9] have utilized simulated game videos integrating various weather and environmental conditions, producing results analogous to actual traffic videos from platforms such as YouTube. The video game GTA V, featuring realistic graphics, varied environments, and traffic models, was utilized to produce training data. Utilizing a free-range camera modification, using a free-range camera mod authors customized the point of view and altered the AI of bot vehicles to refine their driving behavior. Game footage was annotated to denote accidents, with accident sequences delineated from the frame in which a collision becomes inevitable to the frame where the vehicles cease movement.

There are also datasets for modeling accidents, which are collected using Dashboard cameras. (1) Car Crash Dataset (CCD) [10] is a collection of traffic accident videos captured via dashcam that are retrieved from YouTube with diversified environmental attributes (day/night, snowy/rainy/good weather conditions), the temporal annotations indicates accident beginning time is labeled at the time when a car crash actually happens. The accident start times were randomly placed within the last 2 seconds of each clip, creating trimmed videos of 5 seconds in length, resulting in 1,500 traffic accident video clips. The dataset was then divided into 3,600 training videos and 900 testing videos, (2) Car Accident Detection and Prediction (CADP) dataset [6] comprises 4,675 dashcam videos sourced from YouTube, including 1,150 accident videos and 3,525 normal videos. Each video has a resolution of 1280×720 pixels, with an average length of 6.4 seconds. The accident videos were annotated with accident occurrence times, marking when accidents happen within the clips, and (3) Dashcam Accident Dataset (DAD) [11] contains dashcam videos collected from real-world scenarios in six cities across Taiwan. The dataset includes 620 accident videos and 1,130 regular (non-accident) videos. Each video is trimmed to a fixed duration of 5 seconds, sampled into 100 frames, with accidents positioned within the last 10 frames of the accident clips. The dataset is divided into training and testing subsets, with 1,284 videos (455 accident videos and 829 regular videos) used for training and 466 videos (165 accident

videos and 301 regular videos) for testing. These dashcam datasets are specifically collected to evaluate and advance methods for early accident anticipation in traffic environments. There is also an extensive research on datasets for intrusive driver behavior detection [12,13] that relies on in-vehicle sensors and video data. Studies have used smartphone sensors, vehicle-mounted accelerometers, and dashboard cameras to detect behaviors like drunk driving, aggression, fatigue, and distraction using Machine Learning and pattern recognition techniques. In contrast, non-intrusive methods [14], such as roadside camera-based systems with deep learning, offer accurate driver behavior classification and valuable insights for traffic safety and efficiency.

2.2. Accident Detection and Prediction

In [15], the authors propose a single-class neural network technique using a Convolutional Auto Encoder (CAE) to extract robust spatiotemporal features for detecting abnormal events in crowded scenes. This technique is part of a broader trend in anomaly detection, where several deep learning methods have emerged as non-parametric alternatives for anomaly prediction. Comprehensive surveys of deep learning techniques in traffic flow analysis and prediction can be found in [16] and [17]. Additionally, [18] offers an in-depth discussion on deep learning methods for anomaly detection in surveillance videos, including an analysis of supervised and unsupervised approaches and open problems. Another challenge in deploying hybrid deep learning models that integrate spatial and temporal components is the discrepancy between training and prediction time horizons. While some studies, such as [19], show that hybrid deep learning architectures can improve performance under certain conditions, there is ongoing debate regarding the necessity and effectiveness of fine-tuning these models for real-world applications. The scaling of these models remains a significant concern for widespread deployment in traffic monitoring and anomaly detection systems.

3. Dataset Description

The TU-DAT dataset plays a crucial role in traffic accident analysis by enhancing spatio-temporal reasoning and anomaly prediction in road traffic. Unlike conventional neural networks, TU-DAT leverages explicit logic-based reasoning, providing high accuracy and explainability in traffic monitoring. TU-DAT has proven instrumental in predicting and resolving traffic irregularities through the Compositional Framework for Anomaly Resolution (C-FAR). This framework combines deep learning-based object detection (YLLO) and logical reasoning (RTEC) to monitor real-time traffic conditions, anticipate potential collisions, and resolve inconsistencies in road behavior. By employing an event-driven approach, TU-DAT enables continuous risk assessment by analyzing vehicle movement patterns, inter-object distances, and road safety constraints. The epsilon-DDS algorithm, integrated into C-FAR, dynamically adjusts key variables to restore logical consistency and mitigate accident risks. Through these capabilities, TU-DAT not only enhances accident detection and prevention but also strengthens traffic management strategies and improves autonomous vehicle safety, making it a critical asset in intelligent transportation systems. Furthermore, TU-DAT has proven highly effective in enhancing Vision-Language Models (VLMs) by integrating traditional computer vision techniques with logical reasoning. This improves situational awareness in traffic monitoring systems, enabling the detection of rare but critical incidents such as near-miss collisions and unauthorized lane changes. With these advancements, TU-DAT stands out as a comprehensive dataset for intelligent traffic monitoring, autonomous vehicle safety, and real-time accident prediction.

Additionally, this dataset can be used in driving education based on the analysis of traffic incident patterns like aggressive driving. This can provide insights for developing effective training programs that assist drivers in recognizing hazards and comprehending risky behaviors. Other related uses include focused awareness initiatives, advocating for safe driving practices, enhancing driver conduct, and decreasing accident rates.

3.1. Dataset Creation

3.1.1. Data Collection

To create the dataset, we first created a Python-based crawler to extract accident videos from news and documentary websites. We also conducted searches on YouTube for various types of anomalies using text search queries with slight variations (e.g., "unexpected object on the road," "pedestrian accident," etc.). To ensure our approach is suitable for roadside edge devices, we exclusively use footage and images from traffic CCTV cameras. Additionally, we also utilized the BeamNG.drive [20] game simulator to generate road traffic video data. This approach allowed us to simulate aggressive driving behaviors, including speeding, tailgating, weaving through traffic, and running red lights. Using this method, we collected approximately 40 videos of positive examples (aggressive driving) and 25 videos of negative examples (non-aggressive driving).

3.1.2. Data Annotation

We used the Computer Vision Annotation Tool (CVAT) [21] to annotate the video frames. The anomalous situation time is labeled at the time of the anomaly in temporal annotations. The dataset provides diverse real-world examples, capturing variations in visibility, road surface conditions, and vehicle interactions. This comprehensive coverage allows for more robust anomaly detection and predictive modeling by accounting for the impact of weather on traffic incidents.

Fig. 2 presents various crash scenarios captured under different weather conditions in our TU-DAT dataset. Figs. 2 (a) to (d) depict accident scenarios from TU-DAT, captured by roadside cameras under various day/night and weather conditions. Figure 2 (e) illustrates a scenario where a car collides with a stationary object, such as an electric pole, while Figure 2 (f) shows an accident involving a pedestrian being struck by a motorcycle. The details of our dataset are shown in Table 1.

Table 1. Statistics of TU-DAT Dataset.

Conditions	#Frames	Accident Types	#Frames
Day Light	9796	Weaving thru traffic	2417
Night/low light	1487	X-section accidents	6566
Foggy	445	Tailgating/Driving Maneuvers	1452/305
Rainy/Snowy	128/274	Highway/Rear-end Accidents	1254/1215
Camera too far	211	Pedestrian accidents	447

3.2. Statistics of TU-DAT Dataset

Fig. 1 (a) illustrates the distribution of detected objects in a dataset across six categories: Bus, Car, Person, MotorCycle, and Trucks where the y-axis represents the total count of objects, while the x-axis denotes the object categories. The Car category comprises most objects, with approximately 140,000 detections, suggesting a substantial emphasis on traffic environments dominated by cars. This distribution signifies the dataset’s primary focus on urban or suburban traffic, making it suitable for applications like pedestrian safety analysis, accident prediction, or traffic monitoring.

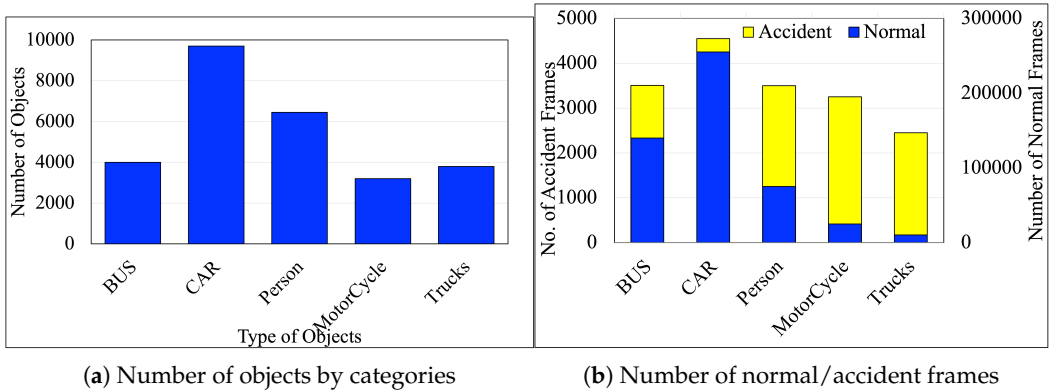


Figure 1. The Statistics of TU-DAT Dataset

Fig. 1 (b) illustrates the distribution of several accidents and regular frames in the dataset across six categories: Bus, Car, Person, MotorCycle, and Trucks where the y-axis represents the number of frames, while the x-axis denotes the object categories. We have collected around 210 videos of road accidents varying around 24-30 FPS through these steps, with 17255 accident frames and 505245 regular frames.

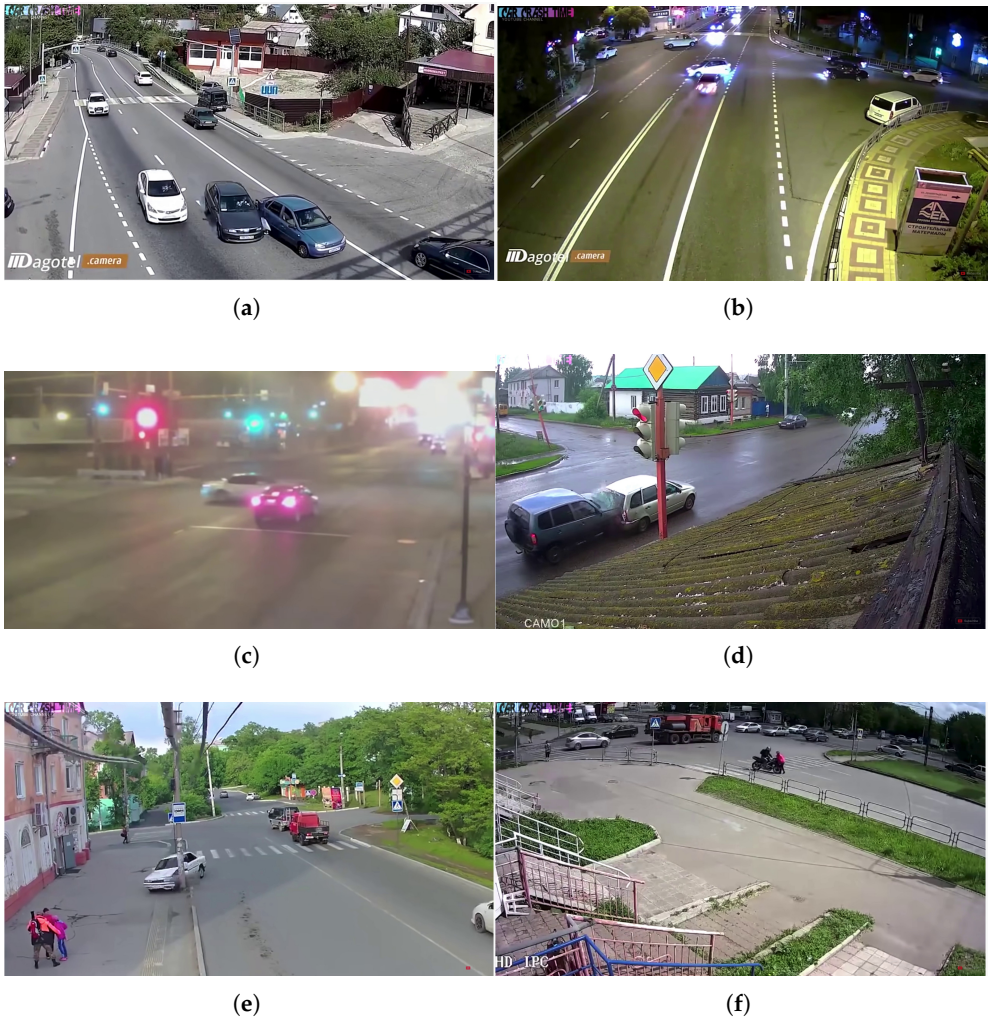


Figure 2. (a)-(e) Some frames of TU-DAT dataset of accident scenarios.(a) shows a front-end accident scenario in a low light condition, (b) shows a rear-end accident on a rainy day, (c) shows car hitting a pole, and (d) shows a motorcycle hitting a pedestrian

3.3. BeamNG.drive Simulator

Accident simulations are essential for interpreting vehicle collisions, enhancing road safety, and facilitating research in automotive engineering and forensic accident analysis. BeamNG.drive, featuring a sophisticated soft-body physics engine, is an optimal platform for producing realistic accident videos. These videos serve multiple purposes, including driver education, insurance fraud detection, law enforcement investigations, and vehicle safety research [22]. Manufacturers and safety organizations, including the National Highway Traffic Safety Administration (NHTSA) and the European New Car Assessment Programme (Euro NCAP), depend on physical crash tests to assess vehicle safety. However, actual crash tests are costly and labor-intensive [23]. BeamNG.drive provides an economical method for virtually pre-evaluating vehicle designs before crash testing. The capacity to produce authentic accident videos with BeamNG.drive markedly improves accident analysis, safety research, driver education, and insurance fraud detection. Experts in various sectors can enhance vehicle design, road safety, and forensic accident analyses using sophisticated physics-based simulations, thereby decreasing traffic-related fatalities and injuries.

3.3.1. BeamNG.drive: Features, Physics, and Applications

BeamNG.drive is a sophisticated vehicle simulation game recognized for its realistic soft-body physics and intricate vehicle dynamics. Created by BeamNG GmbH, this simulator offers an open-world environment where players can drive, crash, and customize a variety of vehicles across different scenarios. Unlike conventional racing or driving games, BeamNG.drive prioritizes realism, making it a favored choice among automotive enthusiasts, game modders, and researchers exploring vehicle dynamics and crash scenarios. Fig. 3 (a) to (d) shows some of the frames from videos generated using BeamNG.drive simulator from TU-DAT dataset. The key features include,

- **Soft-Body Physics Model:** BeamNG.drive's most defining feature is its groundbreaking soft-body physics engine, which revolutionizes how vehicles behave in a virtual environment. Unlike traditional rigid-body physics in most racing games, this technology allows a vehicle's structure to deform accurately upon impact. Each part of the vehicle, from crumple zones to body panels, reacts independently and dynamically in real-time to collisions. As a result, players experience a highly immersive driving simulation where vehicle physics closely mirrors real-life dynamics.
- **Realistic Material Properties:** At the core of BeamNG.drive's realism is its node-beam structure, which forms the foundation of every vehicle model. Nodes represent individual physical points, while beams simulate the connections between these nodes, enabling detailed mechanical responses to external forces. This sophisticated system models materials such as steel, aluminum, plastic, and rubber, each exhibiting unique properties that influence vehicle behavior under stress. Whether it's the flex of a plastic bumper in a minor collision or the resilience of a steel frame in a high-speed crash, these material properties contribute significantly to the game's authenticity.
- **Deformation Mechanics:** Deformation mechanics in BeamNG.drive enhance the impact of crashes and accidents in ways that traditional games cannot replicate. When a vehicle crashes, the soft-body physics engine meticulously calculates stress and strain on each component, resulting in lifelike damage representation. For instance, a high-speed collision may cause metal panels to crumple significantly, glass windows to shatter, or wheels to bend at unnatural angles. This intricate simulation also accounts for crucial factors such as impact angle, vehicle speed at the moment of collision, and the material strength of the affected components, creating a truly realistic driving experience.
- **Suspension and Tire Dynamics:** Beyond vehicle deformation, BeamNG.drive incorporates highly accurate suspension and tire physics, further enhancing realism. The suspension system mimics real-world behavior, effectively simulating weight transfer, body roll, and compression in actual vehicles. This attention to detail affects vehicle handling and influences how cars respond to different terrains and driving conditions. Tires interact with various surfaces—such as asphalt,

- gravel, and mud—realistically affecting grip levels, skidding behavior, and rolling resistance, ultimately making vehicle control challenging and immersive.
- **Crash Testing Scenarios:** BeamNG.drive is a powerful tool for conducting controlled crash tests, allowing players to experiment with various vehicles in different environments. Users can simulate common crash scenarios, including head-on collisions, side impacts, rollovers, and rear-end crashes. The platform also supports AI-controlled vehicles, enabling multi-vehicle collision simulations replicating complex crash dynamics. This feature has drawn interest from engineers and researchers, who often utilize BeamNG.drive as a cost-effective way to visualize crash physics and validate safety measures before conducting real-world testing.
 - **Influence of Environment on Vehicle Damage:** The environment plays a crucial role in BeamNG.drive's approach to crash physics, considering multiple external factors that influence collision outcomes. For example, terrain type, surrounding obstacles, and weather conditions all impact how a vehicle behaves during a crash. A car colliding with a tree will sustain a distinctly different deformation pattern than a concrete wall impact. Additionally, off-road terrains introduce vehicle wear and tear, challenging players to navigate conditions that test their driving skills and vehicle endurance. By combining these elements, BeamNG.drive delivers an engaging gaming experience and a rich platform for exploring vehicle physics, making it a unique offering in the world of vehicular simulation.

4. Methods

4.1. Spatio-Temporal Reasoning

Before discussing the applications, we briefly talk about the spatio-temporal reasoning that we have used extensively in all these applications instead of the routine method of training a neural net. The explicit logic reasoning used in these applications has many advantages including high accuracy, explainability, and highly efficient detection appropriate for real-time monitoring of the traffic.

The spatio-temporal reasoning works by expressing all relevant events, rules of operation, rules of inference, relationships, and constraints in form logic assertions and then using deductions to arrive at the conclusions. In particular, anomalies to be detected such as a vehicle following too closely, speeding near a stop sign, or approaching pedestrians unsafely are represented as logic assertions. We then check if any of these assertions are satisfied based on the observed behavior from the video frames or "groundings" (e.g., a car behind another one with distance decreasing) and the rules/constraints for drawing conclusions.

Basic logic reasoning can be modeled as the Boolean satisfiability problem, leveraging SMT (Satisfiability Modulo Theory) tools to expand the scope of reasoning to various other relevant "theories" such as arithmetic, motion, etc. [24]. For temporal reasoning, we employ Event Calculus (EC), which defines Events (actions at specific times) and Fluents (states affected by events). EC provides a formal framework to represent and reason about dynamic systems where the state of the world changes over time due to the occurrence of events; events cause changes in fluents, which are properties that can be true or false at different time points. This structure enables EC to accurately capture cause-and-effect relationships and temporal dependencies, which are essential for understanding and managing dynamic systems.

Event Calculus for Run-Time Reasoning (RTEC) [25] is an open-source Prolog implementation for efficient real-time event recognition, which utilizes Linear Time Logic (LTL) with integer time points to incorporate real-time aspects. RTEC extends traditional EC by supporting complex event processing, incremental reasoning, and temporal aggregation, making it well-suited for dynamic environments such as real-time traffic monitoring and anomaly detection. With incremental reasoning, it allows for continuous updates as new events occur and optimizes performance through caching and indexing strategies. RTEC also provides mechanisms for event pattern recognition, temporal aggregation, and reasoning over sliding time windows, enabling it to detect temporal trends and persistent conditions

critical for proactive traffic management. This allows timely detection of essential traffic conditions and enhances decision-making in dynamic environments.

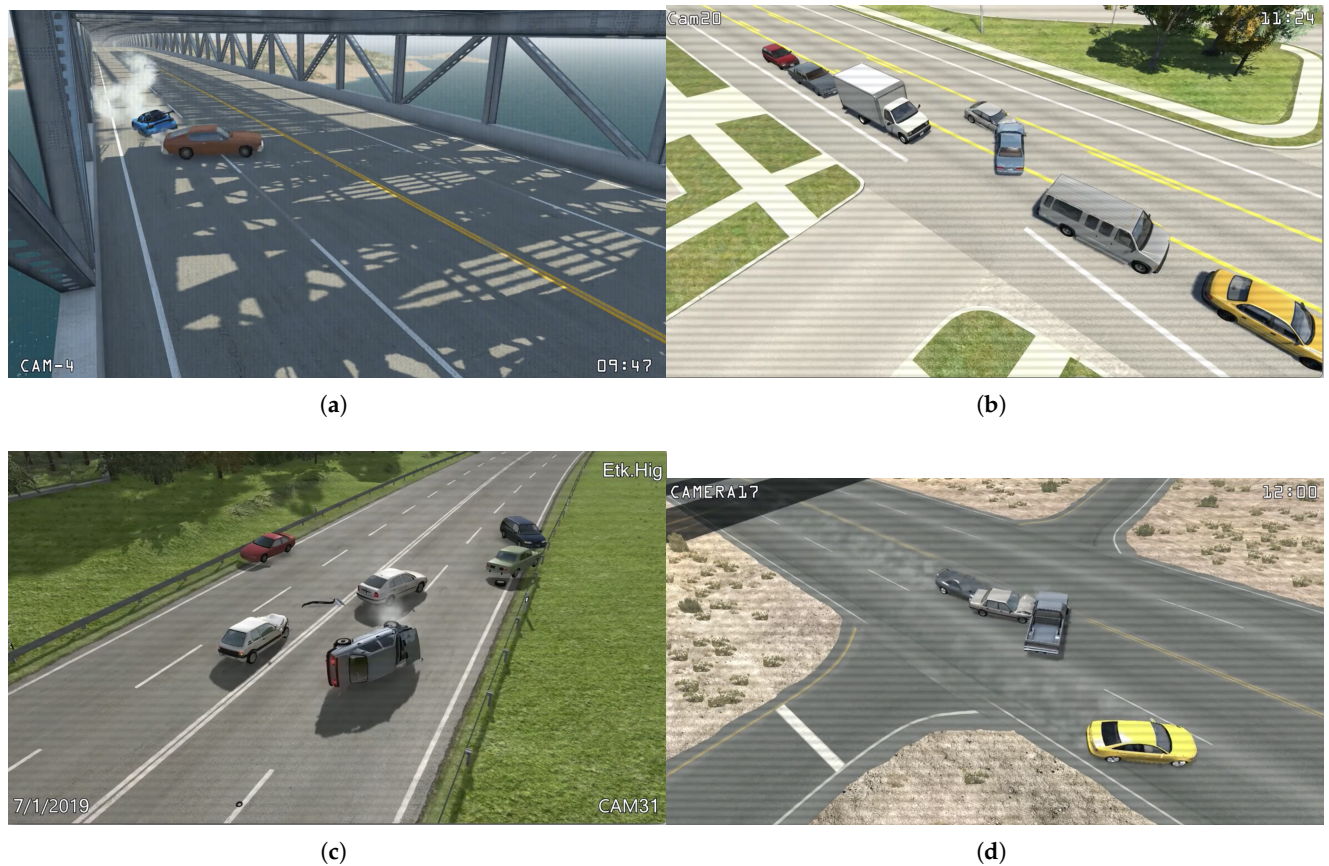


Figure 3. (a)-(d) BeamNG.drive video frames from TU-DAT dataset of accident scenarios.

4.2. Predicting Anomalies in Road Traffic

The primary application of the TU-DAT dataset is the prediction and resolution of anomalies in road traffic monitored through roadside camera networks. We propose a Compositional Framework for Anomaly Resolution in Intelligent Transportation Systems (C-FAR) [26]. C-FAR focuses on predicting potential anomalies rather than detecting specific activities, accounting for uncertainty in future actions and inaccuracies in estimates like speed or distance. We handle these using weighted assertions and model the problem with Weighted Partial Maxsat (WPM) optimization. C-FAR consists of two stages, as explained in the following sections,

4.2.1. Stage 1

In C-FAR, stage 1 is a CNN-based lightweight object detection framework named YLLO, based on YOLOv4 and optimized for continuous video streams by utilizing redundancy to identify the “only” essential frames. YLLO is a three-stage process that begins with a scene change detection algorithm and progresses to object detection via YOLOv4 or any single-shot detector. The Simple Online and Real-time Tracking (SORT) algorithm assigns a tracker to each detected object or multiple objects. YLLO decouples classification and regression tasks to eliminate redundant objects between the frames. Additionally, before sending frames to object detection, for the scene change detection, it generates Color Difference Histograms (CDH) for edge orientations, where edge orientations are determined using the Laplacian-Gaussian edge detection framework.

4.2.2. Stage 2

In Stage 2 of the C-FAR framework, we integrate an Event Calculus engine, RTEC, which processes EC predicates representing time-stamped primitive activities detected in individual video frames. These predicates encode object attributes such as bounding box coordinates, orientation, and movement direction. Each event is time-stamped, with tracked objects' positions recorded as X and Y pixel coordinates at each time step. When dynamic dependencies are significant, such as modeling accidents in which traffic conditions interact dynamically (e.g., a braking maneuver is executed in response to danger), developing a system capable of representing dynamic relationships is necessary. We define derived events as occurrences resulting from changes in state or interactions between entities. The framework detects and predicts anomalies in real-time by analyzing object trajectories and spatial relationships. For instance, potential collisions are forecasted by computing inter-object distances and comparing them against predefined thresholds.

4.2.3. Resolving Anomalies

After predicting an anomaly in Stage 2 of the C-FAR framework, the ϵ -DDS algorithm adjusts key variables to resolve inconsistencies. It identifies unsatisfiable constraints and modifies selected variables within a predefined threshold to restore logical consistency. This optimization process ensures that critical constraints (hard clauses) are always satisfied while a limited number of soft constraints may be adjusted. The algorithm balances minimizing the overall perturbation and ensuring feasibility, with different penalty functions (e.g., absolute or squared differences) influencing how adjustments are distributed across variables. We develop a modified ϵ -DDS algorithm called *situation-based ϵ -DDS*, which compares and selects optimal solutions based on both objective function values and constraint violations. If two solutions are feasible or have similar constraint violations, the one with a better objective value is preferred. Otherwise, solutions are ranked by the degree of constraint violation, ensuring minimal disruption while effectively resolving anomalies.

4.3. Enhancing VLMs for Situational Awareness

We have tested our TU-DAT in our recent work [27] where we propose a novel consistency-driven fine-tuning framework for Vision-Language Models (VLMs) [28] that integrates traditional computer vision (TCV) techniques for detailed visual recognition with explicit logical reasoning to enhance model performance. The proposed approach significantly reduces the dependency on large labeled datasets during fine-tuning by employing an intelligent input selection mechanism, resulting in substantially higher accuracy than approaches that use random or uninformed input selection.

4.3.1. Fine-Tuning VLMs

Fine-tuning vision-language models (VLMs) is crucial for customizing pre-trained models to particular tasks, enhancing their precision, and honing their comprehension of multi-modal data. These models, which combine textual and visual inputs, are increasingly employed in applications such as image captioning, visual question answering, and autonomous navigation [29]. Fine-tuning is essential because general-purpose VLMs may not perform optimally for particular applications despite being trained on extensive datasets. Pre-trained models such as CLIP [29], Minigt4-Video [30], and Video-LLama2 [31] are constructed using varied image-text datasets; however, they frequently lack the requisite domain-specific knowledge for specialized tasks like medical image analysis or satellite imagery interpretation [32]. Fine-tuning facilitates the adaptation of these models to specific contexts by modifying their weights through focused training on selected datasets. This process improves their capacity to identify domain-specific objects, comprehend contextual relationships, and produce more precise textual descriptions from images. These models may generate generic or erroneous outputs without fine-tuning, constraining their applicability in critical domains such as healthcare diagnostics or autonomous driving [33].

A significant limitation of fine-tuning vision-language models is the requirement for extensive labeled data. In contrast to natural language processing models that predominantly depend on textual

corpora, vision-language models necessitate extensive, varied, and meticulously annotated image-text datasets, which can be costly and labor-intensive to assemble [29]. Training a medical imaging model requires millions of images and expert-annotated descriptions, necessitating domain expertise [32]. The deficiency of high-quality, annotated data in specialized domains can impede the efficacy of fine-tuning, complicating the attainment of substantial enhancements over the base model. Moreover, biased or inadequate training data may result in overfitting, wherein the model excels on the training dataset but falters in real-world generalization.

Another challenge of fine-tuning Vision-Language Models is the computational expense. Training large-scale multimodal models necessitate robust GPUs or TPUs, frequently rendering the process unattainable for smaller research teams or organizations with constrained resources [33]. In contrast, to transfer learning in exclusively text-based models, where parameters can be efficiently fine-tuned, Vision-Language Models (VLMs) necessitate intricate interactions between visual and linguistic elements, demanding more comprehensive optimization [29]. Despite these challenges, fine-tuning is crucial for enhancing the accuracy and applicability of vision-language models to real-world issues. Researchers persist in investigating techniques such as prompt tuning, adapter-based fine-tuning, and data-efficient learning strategies to reduce reliance on extensive labeled datasets. In the next section, we discuss an intelligent fine-tuning mechanism designed and tested using TU-DAT to address some of the challenges mentioned.

4.3.2. TU-DAT in Automated Situational Understanding

Automated situational understanding is essential for video-based monitoring of cyber-physical systems and anomalous situations, such as safety issues, security breaches, policy violations, or unusual events. For instance, in traffic monitoring, key activities include identifying accidents, near-accidents, and vehicle-related criminal activities. These critical events, termed main activities (we denote the set of these “main” activities as $\mathcal{A}^m$), are predefined or learnable for situational awareness. The set $\mathcal{A}^m$ represents key activities/events of interest in situational awareness, described at a high level by VLMs. However, finer details, such as object poses, movements, and relative locations, are crucial and are extracted using traditional computer vision (TCV) techniques. By integrating VLM for context and TCV for detailed analysis, supported by efficient, logical reasoning, high-level activities, and low-level details in $\mathcal{A}^m$ can be effectively captured.

Two VLMs, a main VLM (VLM^m) and an auxiliary VLM (VLM^a), are essential for effective situational awareness. While VLM^m provides high-level activity descriptions for $\mathcal{A}^m$, TCV extracts finer details like object poses, movements, and relative locations. VLM^a complements VLM^m by identifying an auxiliary activity set ($\mathcal{A}^a$) for comparison. TCV aids in quickly identifying proxy activities ($\mathcal{A}^p$) that approximate $\mathcal{A}^m$ for efficient data selection, while inconsistencies between VLM^m , VLM^a , and TCV outputs highlight areas needing fine-tuning. This targeted approach reduces resource demands, eliminates the need for labeled data during evaluation, and enables ongoing consistency checks during inference, ensuring accurate recognition of rare but critical activities in $\mathcal{A}^m$. Logical reasoning integrates these components efficiently. By analyzing inconsistencies between the outputs of VLM^m , VLM^a , and proxy activity recognition, we can precisely identify areas where fine-tuning is required, ensuring more targeted and efficient improvements.

The fine-tuning loop begins by evaluating a batch of inputs from ED called the eval-batch. This batch is processed through VLM^m , VLM^a (if used), and TCV inferencing to generate outputs. These outputs are used to ground relevant assertions in the logic representation of detected classes and proxy activities, leveraging a prebuilt Logic rules database for consistency checks.

Following the grounding, SMT checks the consistency between VLM outputs and grounded assertions. If no inconsistency is found, the eval-batch is removed from ED (based on monotonicity assumptions). If inconsistencies arise, the SMT framework identifies offending assertions, pinpointing VLM classes needing further fine-tuning. The next step is selecting a batch from FTD (FT-batch), processing it, and removing it from FTD. Termination occurs if eval-batches or fine-tuning batches are exhausted, the fine-tuning time limit is exceeded, or the consistency measure stabilizes. Initially, fine-

tuning uses randomly selected labeled inputs. For video-based VLMs, longer videos are segmented to focus on single-class interactions, with captions or labels assigned per video or frame as needed. Composite classes may be defined for overlapping activities when necessary.

5. Technical Validation

5.1. Results of Predicting Anomalies in Road Traffic

We evaluate our proposed model by assessing its ability to anticipate future accidents or anomalies. The method computes anomaly confidence at each time slot using metrics like object distance or orientation, defined in terms of fluents or events. If the confidence at time slot t exceeds a threshold δ , the C-FAR framework predicts an anomaly will occur. A correct prediction for an accident video is a True Positive (TP), where the model anticipates the accident $t' - t$ time slots before it happens at t' . For non-accident videos, such predictions are False Positives (FP). The model predicts no anomaly if the confidence remains below δ across time slots. This results in a False Negative (FN) if the video contains an accident or a True Negative (TN) if it does not.

5.1.1. Comparison with state-of-art methods

We compare our proposed model to three other existing accident detection and prediction models proposed in [11,34,35], and the performance results are shown in Fig. 4(b)-(c). In [34], the authors proposed a three-stage framework for automatic accident detection in videos. The first stage employs a car detection algorithm based on the YOLOv3 deep convolutional neural network; the second stage is a tracking algorithm based on the discriminative correlation filter method, and the final stage employs the Violent Flows (ViF) descriptor to highlight the magnitude changes in the motion vectors that are computed using an optical flow algorithm to detect car crashes. [35] is a framework for detecting anomalies, a 3D neural network architecture based on the EfficientNet 2D classifier for detecting aggressive driving, specifically car drifting. The model proposed in [11] is a Dynamic-Spatial-Attention (DSA) based model that uses RNN along with Long Short-Term Memory (LSTM) cells to model the long-term dependencies of all cues to anticipate accidents in dash-cam videos.

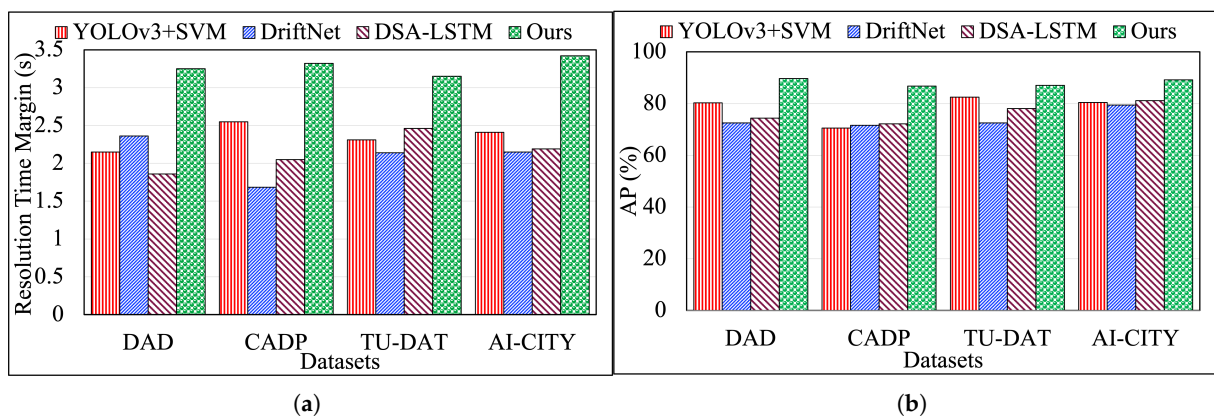


Figure 4. (a) and (b): Performance results showing RTM and AP Values for all Datasets from our previous work [26].

We calculate Average Precision (AP) from precision and recall pairs to evaluate the accuracy of our C-FAR framework. In Fig. 4 (a), the x-axis represents datasets, and the y-axis represents Resolution Time Margin (RTM), which is the time required to anticipate each type of anomaly in seconds, showing that C-FAR performs best on the DAD dataset with optimal AP and RTM. As seen in Fig. 4 (b), our model anticipates accidents 3.42 seconds earlier with an AP of 89.27%, outperforming three other methods. The YOLOv3 + SVM method struggles with parameter setting and orientation issues, while the DSA+LSTM model is resource-intensive and inefficient. Comparing C-FAR with DriftNet, we find that C-FAR captures a broader range of driving behaviors (e.g., speeding, weaving) and achieves 95% accuracy, surpassing DriftNet's 92.5%.

5.2. Results on Enhancing VLMs for Situational Awareness

The proposed directed fine-tuning framework introduced in Section 4.3 evaluates consistency using the TU-DAT dataset. Consistency is quantified using the Consistency Improvement Factor (CIF), defined as $(n_b - n_e)/n_b$, where n_b and n_e are the inconsistencies recorded before and after fine-tuning, respectively. Directed and undirected fine-tuning methods are compared based on CIF over an equivalent number of iterations for fairness. Table 2 shows the achieved CIF for the TU-DAT dataset using X-CLIP, Video-MAE, and MiniGPT4 (image-based), MiniGPT4-Video, Video-Llama, and VideoMamba respectively.

Table 2. CIF results on TU-DAT dataset.

VLM Models	Undirected		Directed	
	VLM ^m	VLM ^a	VLM ^m	VLM ^a
X-CLIP	54.5	55.15	74.25	73.65
VideoMAE	52.04	52.41	72.65	73.25
MiniGPT4	59.78	60.41	75.51	74.35
MiniGPT4-Video	71.45	71.8	86.35	85.125
Video-Llama	72.16	72.41	86.85	87.32
VideoMamba	61.95	61.41	80.85	80.4

6. Conclusions and Future Work

In this paper, we present TU-DAT, a novel and publicly available dataset designed for analyzing road traffic accidents using computer vision techniques. TU-DAT fills a crucial gap in accessible, annotated, and diverse accident video datasets captured from roadside surveillance perspectives. By gathering real-world accident footage from news and documentary sources and augmenting it with simulated videos that model aggressive driving behavior, TU-DAT provides a rich and versatile resource for researchers. The dataset features spatiotemporal annotations, extensive modality coverage, and structured metadata, which support detailed analysis of vehicle interactions, crash dynamics, and environmental contexts. Our initial experiments with TU-DAT demonstrate the benefits of combining deep learning with logic-based spatiotemporal reasoning frameworks, yielding improvements in both accuracy and computational efficiency compared to conventional models.

We believe that TU-DAT will facilitate significant advancements in intelligent transportation systems, particularly in areas such as accident detection, predictive safety analytics, and the training of autonomous vehicle systems. By making this dataset openly available to the research community, we aim to foster innovation and collaboration toward the shared objective of improving road safety and reducing traffic-related fatalities through data-driven approaches. Future work can build upon TU-DAT by exploring several promising directions to enhance its utility for intelligent transportation research. One area of focus could be the integration of multimodal sensor data, such as LiDAR, radar, or vehicle telemetry, to provide a richer contextual understanding of accident scenarios. Furthermore, the dataset presents an opportunity to develop real-time accident prediction models that can be deployed on edge devices embedded in roadside infrastructure.

References

1. NHTSA. <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813561>. Accessed: 1 September 2024.
2. Ou, J.; Xia, J.; Wang, Y.; Wang, C.; Lu, Z. A data-driven approach to determining freeway incident impact areas with fuzzy and graph theory-based clustering. *Computer-Aided Civil and Infrastructure Engineering* **2020**, *35*, 178–199.
3. Zhao, Z.; Liu, T.; Zhang, L.; Xie, S.; Jin, H. Enhancing Autonomous Vehicle Safety: An Integrated Ensemble Learning-Logit Model for Accident Severity Prediction and Analysis. *International Journal of Transportation Science and Technology* **2025**.

4. Wang, J.; Zhao, C.; Liu, Z. Can historical accident data improve sustainable urban traffic safety? A predictive modeling study. *Sustainability* **2024**, *16*, 9642.
5. Yao, Y.; Wang, X.; Xu, M.; Pu, Z.; Wang, Y.; Atkins, E.; Crandall, D.J. Dota: Unsupervised detection of traffic anomaly in driving videos. *IEEE transactions on pattern analysis and machine intelligence* **2022**, *45*, 444–459.
6. Shah, A.; Lamare, J.B.; Anh, T.N.; Hauptmann, A. CADP: A Novel Dataset for CCTV Traffic Camera based Accident Analysis. *arXiv preprint arXiv:1809.05782* **2018**.
7. Ijjina, E.P.; Chand, D.; Gupta, S.; Goutham, K. Computer vision-based accident detection in traffic surveillance. In Proceedings of the 2019 10th International conference on computing, communication and networking technologies (ICCCNT). IEEE, 2019, pp. 1–6.
8. QMUL. https://personal.ie.cuhk.edu.hk/~ccloy/downloads_qmul_junction.html. Accessed: 30 September 2021.
9. Bortnikov, M.; Khan, A.; Khattak, A.M.; Ahmad, M. Accident recognition via 3d cnns for automated traffic monitoring in smart cities. In Proceedings of the Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 2 1. Springer, 2020, pp. 256–264.
10. Bao, W.; Yu, Q.; Kong, Y. Uncertainty-based traffic accident anticipation with spatio-temporal relational learning. In Proceedings of the Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 2682–2690.
11. Chan, F.; et al. Anticipating accidents in dashcam videos. In Proceedings of the Asian Conference on CV. Springer, 2016, pp. 136–153.
12. Dai, J.; Teng, J.; Bai, X.; Shen, Z.; Xuan, D. Mobile phone based drunk driving detection. In Proceedings of the IC-PCT Health, 2010, pp. 1–8.
13. Devi, M.S.; Bajaj, P.R. Driver Fatigue Detection Based on Eye Tracking. In Proceedings of the ICETET, 2008, pp. 649–652.
14. Pradeep, P.; Kant, K.; Pal, A. Non-Intrusive Driver Behavior Characterization From Road-Side Cameras. *IEEE IoT Journal* **2023**.
15. Bouindour, S.; et al. Enhanced CNNs for abnormal event detection in video streams. In Proceedings of the 2019 IEEE Second AIKE. IEEE, 2019, pp. 172–178.
16. Nguyen, H.; et al. DL methods in transportation domain: a review. *IET ITS* **2018**, *12*, 998–1004.
17. Ali, U.; et al. Using DL to predict short term traffic flow: A systematic literature review. In Proceedings of the ITSC. Springer, 2017, pp. 90–101.
18. Zhu, S.; Chen, C.; Sultani, W. Video anomaly detection for smart surveillance. *arXiv preprint arXiv:2004.00222* **2020**.
19. Wang, J.; et al. Traffic speed prediction and congestion source exploration: A DL method. In Proceedings of the ICDM. IEEE, 2016, pp. 499–508.
20. BeamNG.Drive. <https://www.beamng.com/game/>. Accessed: 15 September 2021.
21. CVAT. <https://github.com/openvinotoolkit/cvat>. Accessed: 1 September 2021.
22. Underwood, G. *Traffic and transport psychology: theory and application*; Elsevier, 2005.
23. Organization, W.H. *Global status report on road safety 2018*; World Health Organization, 2019.
24. De Moura, L.; Bjørner, N. Satisfiability Modulo Theories: Introduction and Applications. *Commun. ACM* **2011**, *54*, 69–77.
25. Artikis, A.; et al. An event calculus for event recognition. *IEEE TKDE* **2014**, *27*, 895–908.
26. Pradeep, P.; Kant, K.; Pal, A. C-FAR: A Compositional Framework for Anomaly Resolution in Intelligent Transportation System. *IEEE Trans. on Intelligent Transportation Systems* **2022**. <https://doi.org/10.1109/TITS.2022.3196548>.
27. Pradeep, P.; Kant, K.; You, S. Enhancing Visual Language Models with Logic Reasoning for Situational Awareness. *Submitted for publication. Available at https://www.kkant.net/papers/Pavana_VLM_paper.pdf* **2024**.
28. Li, Z.; Wu, X.; Du, H.; Nghiem, H.; Shi, G. Benchmark evaluations, applications, and challenges of large vision language models: A survey. *arXiv preprint arXiv:2501.02189* **2025**.
29. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International conference on machine learning. PmLR, 2021, pp. 8748–8763.
30. Ataallah, K.; Shen, X.; Abdelrahman, E.; Sleiman, E.; Zhu, D.; Ding, J.; Elhoseiny, M. Minigpt4-video: Advancing multimodal llms for video understanding with interleaved visual-textual tokens. *arXiv preprint arXiv:2404.03413* **2024**.

31. Cheng, Z.; Leng, S.; Zhang, H.; Xin, Y.; Li, X.; Chen, G.; Zhu, Y.; Zhang, W.; Luo, Z.; Zhao, D.; et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476* **2024**.
32. Li, J.; Li, D.; Savarese, S.; Hoi, S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In Proceedings of the International conference on machine learning. PMLR, 2023, pp. 19730–19742.
33. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research* **2020**, *21*, 1–67.
34. Arceda, V.M.; Riveros, E.L. Fast car crash detection in video. In Proceedings of the 2018 CLEI. IEEE, 2018, pp. 632–637.
35. Noor, A.; Benjdira, B.; Ammar, A.; Koubaa, A. DriftNet: Aggressive Driving Behaviour Detection using 3D CNNs. In Proceedings of the 2020 SMARTTECH. IEEE, 2020, pp. 214–219.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.