Application of Outlier Detection for Volvo Trucks Safety Scoring

Classification of drivers based on driving behaviour and assign safety score using unsupervised learning and object detection.

Master's thesis in Systems, Control and Mechatronics

AROKIA SHALINI ALOYSIUS

SWADESH GANDHI

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Application of Outlier Detection for Volvo Trucks Safety Scoring

Classification of drivers based on driving behaviour and assign safety score using unsupervised learning and object detection.

Arokia Shalini Aloysius
Swadesh Gandhi

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2024

Application of Outlier Detection for Volvo Trucks Safety Scoring
Classification of drivers based on their driving behaviour and assigning safety score using unsupervised learning and object detection.
Arokia Shalini Aloysius & Swadesh Gandhi

Supervisor: Lucas Oliveira, Volvo Trucks
Examiner: Martin Fabian, Electrical Engineering

Master's Thesis 2024
Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Scoring of a Driver based on the Speeding, Braking, Acceleration and Idling, indicating how safe the driver is driving.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Application of Outlier Detection for Volvo Trucks Safety Scoring
Classification of drivers based on their driving behavior and assigning safety score using unsupervised learning and object detection.
AROKIA SHALINI ALOYSIUS & SWADESH GANDHI
Department of Electrical Engineering
Chalmers University of Technology

Abstract

Improving safety and preventing accidents is a pressing concern in today's growing demand for increasing transportation services. To ensure road safety protocols and minimize traffic-related incidents, Volvo Trucks has committed to evaluating and improving driver behaviour through advanced data analysis and machine learning techniques. This thesis explores the application of outlier detection methods to evaluate and improve safety scoring of Volvo truck drivers based on the driver's behavior, braking and acceleration patterns, and contextual traffic conditions. The data from the truck's Advanced Driver Assistance Systems, including brake pedal position, longitudinal acceleration, and longitudinal velocity, is used to examine the braking behaviour of the drivers during Pre-brake and Full-brake events from the CW-EB. These behaviors are then clustered using unsupervised learning and vector quantization techniques to classify them into different driving risk levels and assign safety scores. Additionally, YOLOv8, an object detection model, is introduced to determine whether the event was caused by the driver or the surrounding environment.

Keywords: Outlier detection, Safety score, Time Series Data analysis, Clustering, Vector Quantization, Machine Learning, Pre-Brake, Full-Brake, CW-EB, K-means, YOLO.

Acknowledgements

We would like to express our deepest gratitude to Martin Fabian, our academic supervisor, whose expert guidance, invaluable insights, and unwavering support have been instrumental in shaping this research.

The thesis work was carried out at Volvo Group, Göteborg under the Services Team from February 2024 to July 2024.

We are deeply grateful to Lucas Oliveira, our supervisor at Volvo Group, whose expertise, patience, and constructive feedback have greatly enhanced the quality of the work and helped us resolve the hindrance at each level as we progressed in the thesis. Despite becoming a father very recently, he has consistently been there to guide us, balancing his family duties with his professional responsibilities.

Additionally, we extend our heartfelt gratitude to Malin Larking, Manager at Service Team, Volvo Group, for her encouragement, support, and for providing the opportunity to delve into this research within the company. Their collective contributions have profoundly enriched our learning experience and have been pivotal in navigating the complexities of this master thesis.

The authors also extend their heartfelt thanks to the ADAS team and those who have supported us along the way by providing input and ideas throughout the thesis work.

Swadesh Gandhi and Arokia Shalini Aloysius, Göteborg, September 2024.

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed here:

ADAS	Advanced Driver Assistance Systems
ABS	Anti-lock Braking System
AEBS	Advanced Emergency Braking System
AI	Artificial Intelligence
CNN	Convolutional Neural Networks
CW-EB	Collision Warning with Emergency Brake
DAS	Driver Assistance System
FCW	Forward Collision Warning
IQR	Interquartile Range
LBG	Linde–Buzo–Gray
LKS	Lane Keeping System
LSTM	Long Short-Term Memory
RID	Reading-ID
R-CNN	Regions with Convolutional Neural Networks
SQL	Structured Query Language
TTC	Time To Collision
TTR	Time To React
TSKM	Time Series K-Means
V8	Version 8
VID	Vehicle-ID
VQ	Vector Quantization
WCSS	Within-Cluster Sum of Square
SSD	Single Shot MultiBox Detector
YOLO	You Only Look Once

Nomenclature

Below is the nomenclature of variables that have been used throughout this thesis.

Variables

Ego Vehicle	The following vehicle from which the data is extracted for analysis.
Leading Vehicle	The vehicle followed by the ego vehicle
ϵ	Tuning parameter in LGB algorithm
Segment	Period of unsafe event of individual drivers
$\|\|$	OR
X	Feature vector
VID	Each vehicle is identified by a unique id called Vehicle-ID
RID	Each VID can have multiple drivers and they are identified by a unique id called Reading-ID
V_{d}	Velocity difference between the following and leading vehicles
V_{E}	Velocity of the Ego vehicles
V_{L}	Velocity of the leading vehicles
a_{min}	Minimum Acceleration
a_{average}	Average Acceleration
η_{E}	Percentage reduction in Kinetic Energy
t_0	Triggering point to braking
t_1	Maximum deceleration point
S_{final}	Final score
w	Weights to evaluation techniques
S_{brake}	Score from the braking pattern
S_{risk}	Score from the driving risk levels
S_{warnings}	Score from the TTC warning
S_{object}	Score from the object detection results

$f_{safe}(t)$	Acceleration curve of the safest pattern
$f_{seg}(t)$	Acceleration curve of the segment of individual RIDs
A	Area between the segment and the safest pattern
$D(t)$	Distance between the following and the leading vehicle in time
Braking	Fault status from the object detection algorithm as 1
Normal	Fault status from the object detection algorithm as 0
$S_{\text{intentional}}$	Intentional Score of the safety scoring system
$S_{\text{contextual}}$	Contextual score of the safety scoring system
tp	time period in which the acceleration data is obtained.

Contents

List of Figures

List of Tables

1 Introduction

Transportation plays a huge role in the global economy, with trucks being a critical source for moving goods. Millions of trucks travel through cities and countries every day, ensuring timely delivery of products. However, the safety of truck drivers and other road users is a pressing concern. In the US, 5,837 heavy trucks were involved in fatal crashes in 2022, which is 1.8% increase from 2021 and 49% over the last ten years [1]. Furthermore, 120,200 heavy trucks were involved in accidents in 2022 that resulted in injuries, a 2.5% increase over 2021. Involvement rates per 100 million miles traveled by large trucks have increased by 24% in the last ten years and by 3% in 2021 [1]. Nearly 95% of serious traffic collisions are due to human error, with over 70% of commercial fleet collisions involving distracted drivers [2]. These statistics demand an urgent attention in improving the safety measures within the truck industry to prevent the risks of collision, injuries or causalities [1].

With a commitment of achieving 100% safety, Volvo trucks monitors driver behaviour such as harsh braking and acceleration, as well as intervention by active safety systems to avoid an incident or accident. This helps fleet managers and drivers identify areas of improvement and contribute towards safer transports. Applying machine learning models to driver data can provide a solid solution for enhanced driver behaviour analysis by examining driver's responses to particular situations and identifying anomalies in driving patterns.

1.1 Background and Motivation

Volvo trucks has a service called "Safety Reports" that gets data from the active safety systems on the trucks and presents them in a reports that can be easily assessed by fleet managers. This provides information to fleet managers regarding the safety behavior of drivers in their fleets. However, this existing safety report does not offer any insights on the driving style, behaviour of the driver during critical events, traffic density, dynamic road conditions, driver attention or other external elements that influence driver behaviour and safety during an event.

To address these limitations, a system was created to assign scores to the drivers based on their driving style and driving behaviour. This can help fleets understand how a driver's actions affect their operations, either positively or negatively. It also encourages drivers to become more involved and engaged while driving. When the scoring system is fair and easy to follow, it not only helps identify problems but also sets clear goals and rewards for drivers, encouraging them to improve their performance. We have implemented this safety scoring system using unsupervised learning and object detection algorithms to improve the existing methodology for implementing the safety score at Volvo.

1.2 Objective

The study aims to create a safety scoring system for truck drivers. This is achieved by analyzing the longitudinal acceleration signals of trucks during Pre-brake and Full-brake events in the CW-EB system, as described in Section 1.4.1. Additionally, an object detection technique is employed to further refine the analysis. Based on these factors, a safety score is formulated to evaluate and validate driver performance.

1.2.1 Scope

- Study the patterns of rapid deceleration during near crash events.
- Group similar patterns of driving style and driving behaviour of various drivers using Linde-Buzo-Gray algorithm [3], K-Means [4], and TTC principles [5].
- Apply You Only Look Once [6] object detection on the video during the event to determine the behaviour of the vehicle in-front/ego vehicle.
- Formulate a comprehensive scoring system that understands the intention of the drivers and context of the surrounding in which the unsafe event has occurred.
- AI chatbot (chatgpt and claude) were used for language improvement in the report writing, especially used in sentence correction and spelling checks.

1.2.2 Limitations

This work is subjected to the following limitations:

- The analysis is subjective to the data obtained from the CW-EB databases and similar results are guaranteed but there can be some deviation in the results when a new extract of data is used. Pre-processing is a must in the ways mentioned.
- The accuracy of distance travelled, acceleration, and velocity of the leading vehicle is not guaranteed.
- The object detection model was trained on a limited number of images due to time constraints and might provide wrong or no results on new image/video.
- Real time deployment of the object detection model is not guaranteed as the work is in initial stages and no benchmarking was conducted.

1.3 Related Work

The related works highlight various approaches to driver risk assessment and behavior analysis using different methodologies and datasets. Zheng et al. [7] conducted a naturalistic driving experiment in China, collecting extensive data on near-crash incidents to classify driving risk levels through K-means cluster analysis. Naito et al. [8] focused on classifying rapid deceleration events into distinct braking patterns to evaluate driver risk levels. Mumcuoglu et al. [9] developed a real-time evaluation system for heavy-duty vehicles, using an Long Short-Term Memory (LSTM) based model to classify risky acceleration and braking behaviors [10]. Abbas et al. [11] used discriminant analysis on data from the Naturalistic Truck Driving Study to predict safety-critical events based on deviations in trucking behavior. Pirhonen et al. [12] introduced a novel brake light detection algorithm using YOLOv3 and Lab colorspace thresholding to enhance predictive braking systems.

1.4 Background Information

An overview of the ADAS sub-systems is provided to explain how each sub-system functions and contributes to the development of the safety scoring system that we aim to implement.

1.4.1 Collision Warning with Emergency Brake

The Collision warning with emergency brake (CW-EB) is a system that alerts the drivers to avoid the risk of rear end collision against the vehicle in front. The system measures and detects the locations and dynamics of vehicles within the sensor's area of vision by combining data from RADAR and camera. When a vehicle is detected ahead, the system assesses whether it poses a risk and whether the driver has enough time to avoid it. The device alerts the driver first if there is very little time left for them to respond by braking or steering. The system applies the brakes to the truck if it determines that a collision is still likely to happen. This is done to prevent a collision or lessen its severity. CW-EB is activated automatically when the truck is started (at key ON), and is available at speeds above 10 km/h. The pre-warning is available above 30 km/h [13].

There are five major phases at which the CW-EB system works:

- **No warning**, this is a phase where there is no close proximity between the ego and the leading vehicle.
- **Cautionary warning**, this is the period where the ego vehicle is entering and getting closer to the leading vehicle and a visual indicator is displayed on the head-up display to make the driver cautious.
- **Imminent warning**, this is the period in which the ego vehicle is almost close and will eventually lead to a collision, visual and auditory signs are given to the driver to be aware and behave according to situation.
- **Mitigating**, this is the period where the ego vehicle's speed is drastically lowered and is almost at the verge of stopping to avoid crashing with the leading vehicle.
- **Overriding**, this is the period in which the ego vehicle is in the closest possible proximity, such as situations like drowsiness or un-consiousness, and this will lead to a collision with the leading vehicle when the driver is not cautious

enough, so the CW-EB system will perform a full brake and make the ego vehicle stop completely.

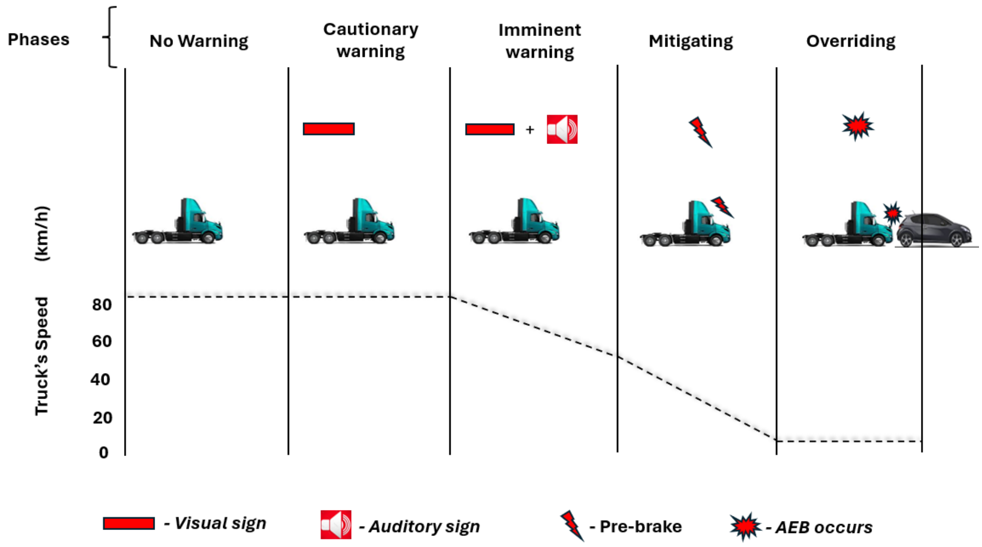


Figure 1.1: Stages at which the CW-EB Activation Sequence occurs

1.4.2 Lane Keeping System

Lane Keeping Systems (LKS) with corrective steering is a driver assistance system that provides steering control and warnings when the truck approaches lane markings [13]. If the truck inadvertently deviates from its lane, the driver may receive a warning through a vibrating steering wheel, a steering intervention, or both. These functions are controlled by a switch on the instrument panel, and the corresponding LED lights up when a function is activated. The system operates under specific conditions:

- The steering and/or warning function is turned on.
- The truck is traveling at a speed greater than approximately 55 km/h.
- The lane is clearly marked and at least 2.8 meters wide.
- There are no intentional maneuvers or the vehicle is returning to its lane.
- The driver has not used the turn signals or recently applied the brakes.

1.4.3 Driver Alert Support

Driver Alert Support (DAS) is a driver support system that informs about the driver's level of attention on driving. The system attracts the driver's attention if driving ability is impaired, for example, if the driver is falling asleep. DAS is switched on automatically when the truck is started and the function is activated at a speed of 65 km/h [13].

The system is controlled by a switch on the instrument panel. When the LED in the switch illuminates, the system is turned on. DAS uses the truck's lane keeping to determine the driver's level of attention. When the truck is not following the road markings in a smooth manner, the driver is warned by means of a message in the display and an acoustic signal.

For DAS to work, the following conditions must be met:
- Driver Alert Support (DAS) is turned on.
- The road has legible lane markings.
- DAS initiates itself when the truck's speed is greater than 65 km/h to actively monitor the drivers ability to continuously drive the truck.

2
Theory

This chapter delves into the concepts and methods involved in developing the safety scoring system for Volvo. It provides a detailed explanation of the techniques used to offer a clearer understanding of how the safety scoring system is formulated.

2.1 Outlier

Outliers are observations or measures that are suspicious because they are much smaller or much larger than the vast majority of the observations as in figure 2.1 [14]. They are data points that stand out due to their significant deviation from the typical patterns observed within a dataset. Outliers can appear for many reasons, such as natural deviations in population behavior, fraudulent activities, and human or system errors. Outlier detection is a statistical procedure that aims to find such events or items in a given dataset. Use of outlier detection for the safety scoring system plays a major role in assessing and enhancing driver safety. By using AI models in outlier detection, the aim is to identify abnormal driving behaviour among a large dataset of fleet records. Traditional outlier detection methods, such as Z-score [15]

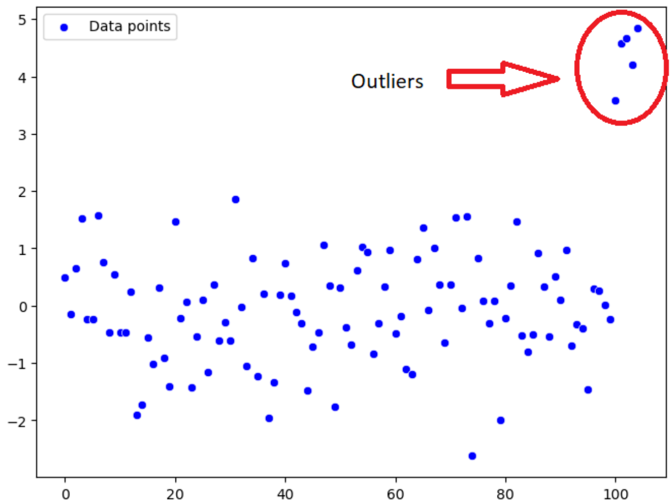


Figure 2.1: Outlier data points from a scatter plot

and Interquartile Range [16], have been used for identifying data points that deviate significantly from the norm. Although they are effective in many scenarios, these methods come with limitations. They often assume specific data distributions and struggle with non-linear relationships and high-dimensional datasets. On the other hand, AI-based outlier detection methods adapts to changing patterns, handle high-dimensional data, and automatically capture complex relationships. Their ability to learn from data make them valuable in scenarios with evolving driving behaviors.

2.2 Vector Quantization

Vector Quantization (VQ) is a technique that quantizes continuous or discrete data into a finite number of representative vectors, also referred to as codevectors or simply referred to as centroids, and it is widely used in applications like speech and image compression, data clustering, and data pre-processing tasks. Vector quantization seeks to achieve a compact representation of the data while retaining as much as information possible by minimizing the distortion between the input data and the codevectors.
The benefits of using vector quantization techniques [17]:

- **Data compression**: VQ is suited for applications like picture and audio compression since it can accomplish large data compression with very little information loss.
- **Noise reduction**: By substituting representative code vectors for individual data points, VQ can assist noise filtering in data by producing smoother and more reliable representations.
- **Pattern recognition**: For tasks like feature extraction, clustering and classification, VQ can be used for finding patterns or structures in data.

2.2.1 Linde-Buzo-Gray Alogrithm

The Linde–Buzo–Gray algorithm is an important method used for vector quantization. It was developed by Yoseph Linde, Andres Buzo, and Robert M. Gray, and primarily aimed at designing the optimal codebook for representing a set of data vectors [18].
The LBG algorithm is effective for designing codevectors from a large data, obtaining the hidden patterns as representative vectors and forming a converged codebook. The optimization objective is usually to minimize the distortion or mean squared error between the data vector and its closest code vector in the codebook.

The LBG algorithm, illustrated in Figure 2.2, works in the following way:

1. **Initialize Codebook**: As a first step, we initialize a codebook at the mean of the entire dataset.
2. **Expanding Codebook**: The codebook is expanded by adding and subtracting a small value ϵ which equals to 0.01, it creates a slight variation in the initial centroid to start the clustering process effectively.
3. **Assign data points to the Nearest Centroid**: By calculating the nearest Euclidean distance to data points from each centroid, a cluster id is assigned to cluster data points of similar ids as cluster groups.
4. **Update Centroid**: Now the mean of the data points that have similar cluster ids are taken and updated as the new updated codebook.

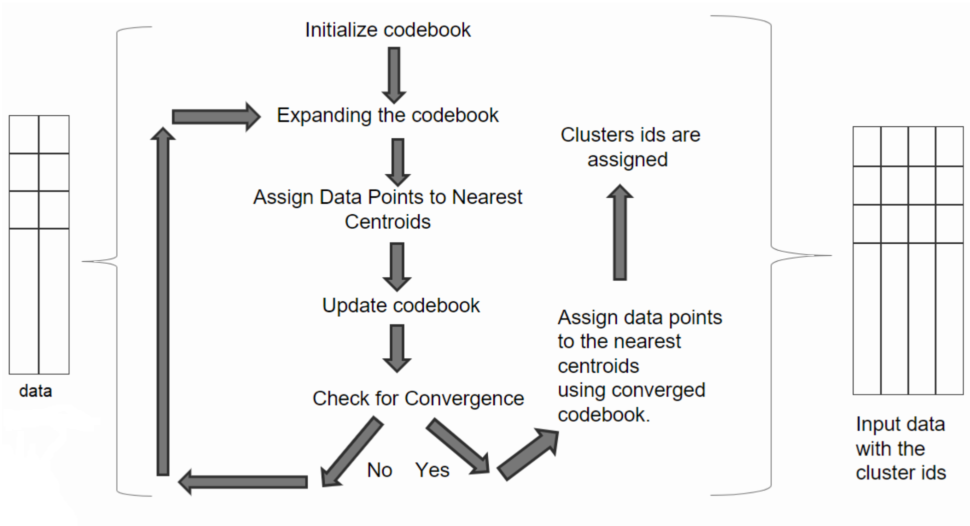


Figure 2.2: Workflow of LBG clustering.

5. **Convergence Check**: Here, we check if the distance between the old and the new code vector is less than ϵ or not. If this criterion is met then we have the converged code vector which is the representative vector that we need, else we go to step 2 to start expanding the updated code vector and this goes on until we obtain the required converged codebook.

2.3 Clustering Techniques

Clustering is a type of unsupervised learning where the references need to be drawn from unlabelled datasets. The goal of clustering is to group data so that the data points in the same group are more similar to each other than to those in other groups. In short, it is a collection of objects based on their similarities and dissimilarities.

This technique can be used in solving the following problems [19]:

- **Anomaly, novelty or discord detection**: Anomaly detection are methods to discover unusual and unexpected patterns.
- **Pattern discovery**: To discover interesting patterns in databases.

This can also be referred to as a **Semi-Supervised Machine Learning** approach, as the data set provided was labeled. The unsupervised model was trained to capture the hidden patterns in the labeled dataset [20].

2.3.1 K-means Clustering

K-means is a centroid-based clustering algorithm, where we calculate the distance between each data point and a centroid to assign it to a cluster. The goal is to identify the K number of groups in the dataset [4].

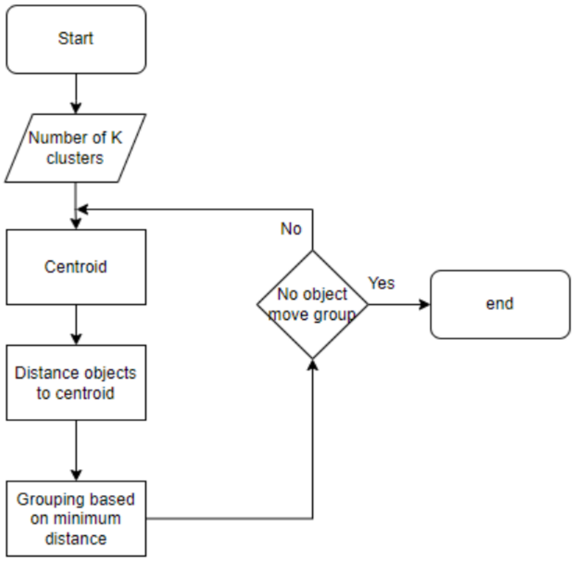


Figure 2.3: Workflow of K-means clustering [21]

Assigning each data point to a group is an iterative process that gradually results in data points being clustered according to comparable traits, as seen in Figure 2.3. The goal is to determine which correct group each data point belongs to minimizing the sum of the distance between the data points and the cluster centroid.

The **Elbow method**, see Figure 2.4, is a graphical method that finds the ideal K value based on the premise that the sum of the squared distances between points and their cluster centers (WCSS) will decrease with increasing number of clusters. Thus the data is effectively divided into progressively smaller groups, and the point at which the fit of the clusters is not considerably improved by adding more clusters is sought after.

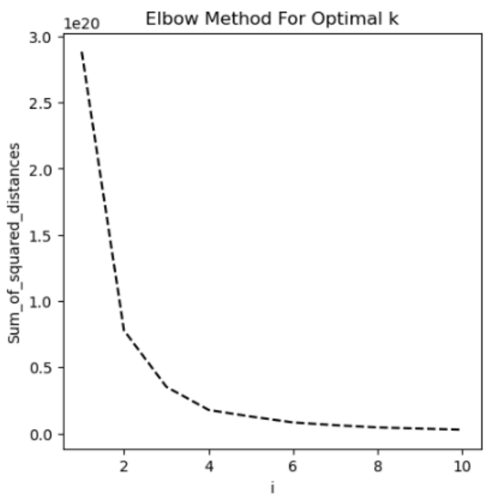


Figure 2.4: Elbow Method for finding the optimal K value for given data.

2.3.1.1 Time Series K-Means Clustering

For a given dataset of n time-series data $D = \{F_1, F_2, \ldots, F_n\}$, where F_i are the features chosen from the dataset, time series K-means clustering is the process of forming the unsupervised partitioning of D into $C = \{C_1, C_2, \ldots, C_k\}$, in such a way that homogeneous time-series are grouped together based on a certain similarity measure [19]. Each C_i is called a cluster, where $D = \bigcup_{i=1}^{k} C_i$ and $C_i \cap C_j = \emptyset$ for $i \neq j$ [19].

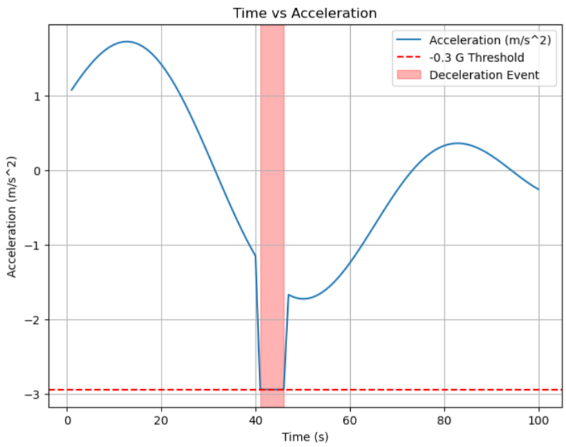


Figure 2.5: Example of sliding window for *tp* seconds of deceleration data was then segmented out for clustering to analyse the braking behaviour of drivers.

In this thesis the **Subsequence time-series clustering**, a set of subsequences of a time-series data was extracted via a sliding window i.e. a segment from a single long time-series data [19] is extracted to analyze the rapid deceleration patterns of drivers. In Figure 2.5, the deceleration event is the segment, data points in that segment were only used for the classification problem.

The segmented raw-time series has acceleration data with timestamps for *tp* time-period, *tp* is the deceleration event from which the segment is extracted . Negative accelerations were only considered for each driver data as the input to ML model and it followed the workflow as in Figure 2.6 and it was pre-processed as described in Section 3.2.

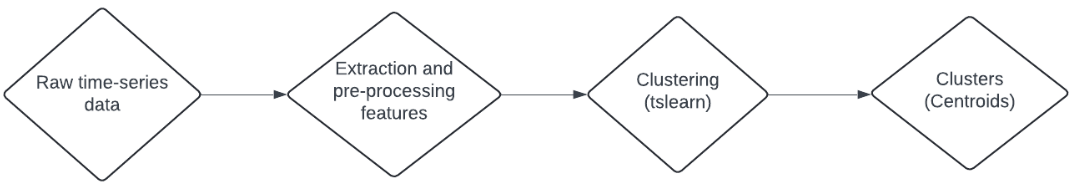


Figure 2.6: Overview of the Clustering

2.3.1.2 Classifying the Driving risk of Drivers using K-Means Clustering

Driving risk is defined as a potential threat that may cause vehicle crashes or other accidents [22]. Emergency brakes are typically the primary indicator of the driver's exposure to driving risk. As a result, the idea to construct a typical deceleration curve of a braking process is that the driving danger level can be represented by the characteristics of a braking process that is highlighted below,

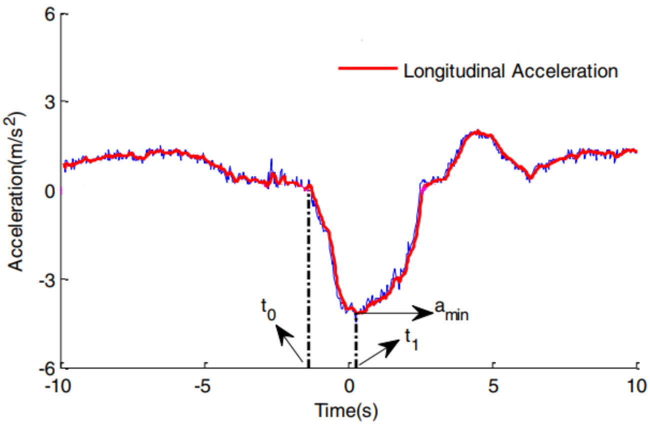


Figure 2.7: Key features of driving risks are a_{min}, t_0, t_1 [22].

The following three features are adopted to represent the driving risk level involved in a typical near-crash case during naturalistic driving [22], see Figure 2.7:

- Maximum deceleration during the braking process a_{min}.
- Average deceleration a_{average} from the braking triggering point t_0 to the point of maximum deceleration t_1.
- Percentage reduction in the vehicle kinetic energy η_{E} from the braking triggering point t_0 to the point of maximum deceleration t_1.

By taking into account these key features of driving risk the K-mean clustering is implemented to find the Aggressive, Smooth, and Efficient drivers based on the acceleration profile of each driver.

$$X = \left[a_{\text{min}}, a_{\text{average}}, \eta_E\right]^T$$

The feature engineering is done from the above and the feature $\mathbf{X}$ is used in the K-Means Clustering to find the category of each driver. It is a very popular and effective clustering algorithm whose inputs are the number of clusters K and the data set. The algorithm starts with initial estimates for the K-centroids, which can either be randomly generated or randomly selected from the data set. The algorithm then iterates between data assignment and centroid update steps.

2.4 Activation Sequence from CW-EB

The proximity described in Section 1.4.1, where types of CW-EB warnings are defined, could be with respect to time or with respect to distance as discussed in [23] such as:

1. The Time remaining between the ego and leading vehicles to crash if both the vehicles continue to travel at the current speed and trajectory.
2. The total distance traveled from the moment a driver finds a need to stop until the vehicle reaches a complete halt after depressing the brake pedal.

These play a crucial role in setting a threshold to classify the drivers into the phases as given in Figure 1.1 which simulates the stages that could have occurred. Figure 2.8 is an example of a truck following a car and its phases are given by numbers indicating the places where warnings are detected and places where the sensors get signals to analyze the situation.

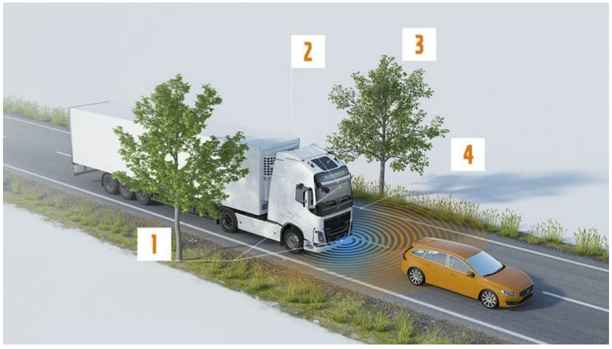


Figure 2.8: Emergency Braking Stages

2.4.1 Time-to-Collision

The reaction time is the action response time when the driver encounters an unsafe situation. It is the time taken by the driver to react to avoid near crash/collision with the leading vehicle and it also reflects how the driver perceives the situation in that moment [24].

The proximity is measured through **Time** by **TTC** principle. It is defined as the following expression [23]:

$$\mathrm{TTC} = \frac{D(t)}{V_d}, \tag{2.1}$$

where $D(t)$ is the distance between the vehicles and V_d is the velocity difference between the ego vehicle and the leading vehicle:

$$V_d = V_E - V_L \tag{2.2}$$

where V_E is the velocity of the ego vehicle and V_L is the velocity of the leading vehicle.

When $V_d \leq 0$, the ego vehicle operates in safe conditions. On the other hand, if $V_d \geq 0$, this indicates a near-collision event [25].

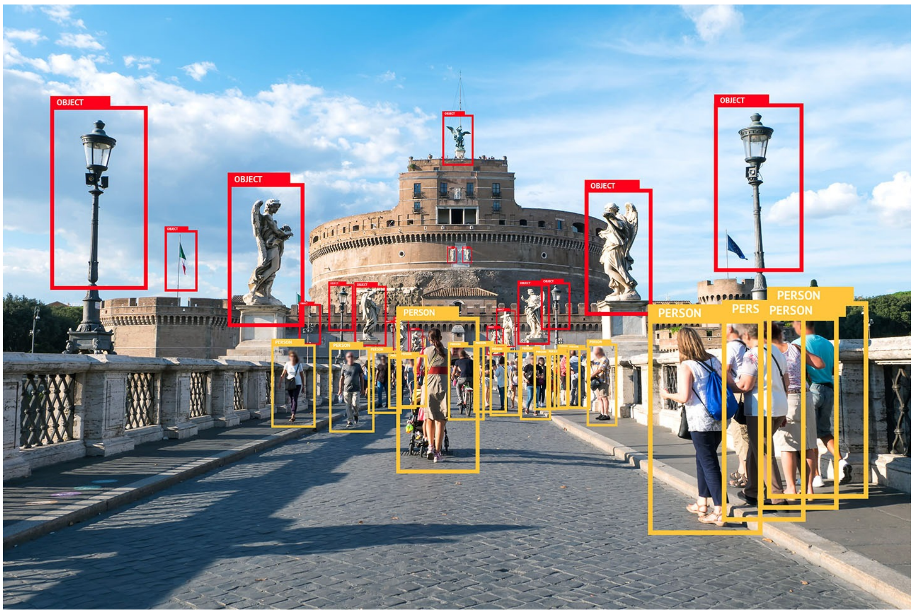


Figure 2.9: Bounding boxes to classify objects within an image using object detection [26].

2.5 Object Detection

Object detection is a computer vision technique used to identify and classify objects within an image or a video [27]. Unlike image classification, which only classifies images into different classes, object detection shows the location of the object in an image or a video and draws a bounding box or a mask around it. YOLO employs Convolutional Neural Networks [28] to classify and draw the bounding box coordinates for every object that is observed. Object detection has enormous applications across various fields including autonomous vehicles, security and surveillance, robotics, and health care.

2.5.1 Object detection algorithms

Single-shot detectors and two-shot (or multi-shot) detectors are the two general categories into which object detection algorithms can be divided. The object detection task is approached differently by these two types of algorithms.

2.5.1.1 Single-Shot Algorithms

A single-shot algorithm detect objects in images in one go. They treat object detection as a single task, predicting both the location and the category of objects in a single pass through the network. This approach is fast and efficient, making it suitable for real-time applications such as autonomous driving and surveillance.

- **How They Work:**
    1. The algorithm divides the image into a grid and checks each grid cell to predict whether it contains an object.

2. For each cell, it predicts bounding boxes (the location of objects) and the probabilities of the object's classes. YOLO is an example algorithm.

2.5.1.2 Multi-shot algorithms

Multi-shot algorithms break down the object detection process into multiple stages. They first generate potential regions where objects might be located and then refine these regions to classify and accurately locate the objects. This step-by-step approach is usually more accurate but slower compared to single-shot methods.

- **How They Work:**
  1. First, they propose several regions in the image that might contain objects (region proposal stage).
  2. Then, each proposed region is examined in detail to classify the object and refine its bounding box (classification and localization stage).
- **Example Algorithms:**
  1. R-CNN [10]
  2. Faster R-CNN [29]

2.6 You Only Look Once

YOLO is a state-of-the-art, real-time object detection algorithm introduced in 2015 by Joseph Redmon et al [30]. It frames object detection as a regression problem to spatially separated bounding boxes and associated class probabilities [27]. It is designed to pinpoint objects accurately while being efficient with computing power. This efficiency comes from its ability to predict where and what objects are in a single sweep through the data. This makes it a great option for tasks such as real-time surveillance or video feed analysis where speed is a priority.

2.6.1 How YOLO works

YOLO is an object detection algorithm that operates by dividing the input image into an $S \times S$ grid and predicting bounding boxes and class probabilities for each grid cell. It uses a single convolutional network to predict bounding boxes and class probabilities simultaneously.

2.6.2 Image to Grid Mapping

- Let the input image size be $W \times H \times 3$, where W and H represent the width and height of the image in pixels, and 3 denotes the three color channels—Red, Green, and Blue (RGB)—capturing the color information for each pixel in the image.
- Divide the image into an $S \times S$ grid. Each grid cell is responsible for detecting objects whose centers fall inside the cell.

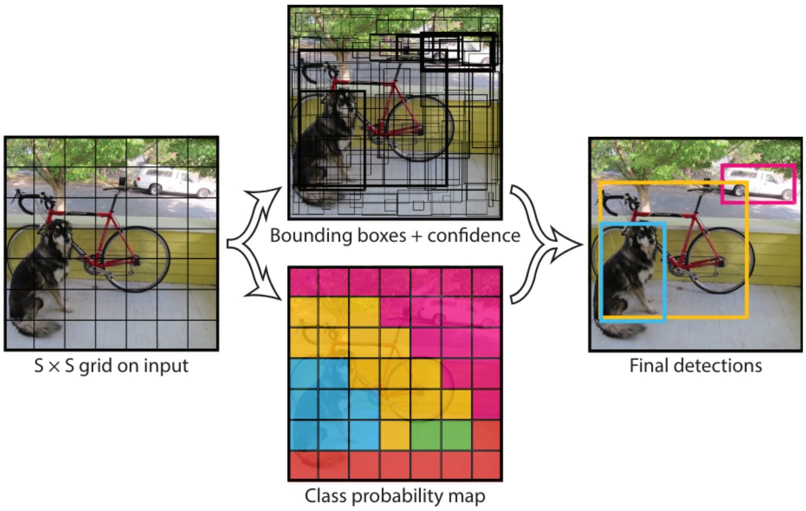


Figure 2.10: YOLO model detection as a regression problem [30].

2.6.3 Bounding Box Prediction

- Each grid cell predicts B bounding boxes, with each bounding box described by five parameters: $x, y, w, h, \mathit{confidence}$.
- x and y are offsets of the center of the bounding box relative to the bounds of the grid cell, typically scaled to be between 0 and 1.
- w and h are the width and height of the bounding box relative to the whole image, also scaled between 0 and 1.
- The *confidence* score represents two things: $P(\text{object}) \times \text{IOU}$, where $P(\text{object})$ is the probability that an object is present in the box, and IOU (Intersection Over Union) is the predicted box's overlap with the ground truth box.

2.6.4 Class Prediction

- Each grid cell also predicts a conditional class probability $P(\text{class}_i \mid \text{object})$ for each of the C classes.
- Thus, for each grid cell, we have a total of $B \times (5) + C$ predictions.

2.6.5 Loss Function

YOLO uses a multi-part loss function to optimize the network. The loss function includes:

- **Localization Loss**: This part measures the error in predicting the bounding box coordinates of detected objects. It typically uses the Mean Squared Error between the predicted and ground truth bounding box coordinates. YOLO uses a modified version that focuses on the center coordinates (x, y) and the width and height (w, h) of the boxes.
- **Objectness Loss**: This component assesses how well the model predicts whether an object is present in a given bounding box. YOLO predicts an

objectness score for each bounding box, indicating the likelihood of it containing an object. The loss here is often calculated using binary cross-entropy for the objectness score.

- **Classification Loss**: For each detected object, YOLO also predicts a class label. The classification loss quantifies the difference between the predicted class probabilities and the actual class labels. This is typically calculated using categorical cross-entropy.

2.7 Safety Scoring System for the truck drivers

The safety scoring system aims to evaluate the drivers and how they operate the truck. This is done by considering their driving approach during an unsafe event. Especially in near-collision situations, by investigating the braking and accelerating profiles a score is formulated.

2.7.1 Evaluation of Rapid Deceleration Patterns

The safest rapid deceleration pattern and segment of each driver is compared in the following aspects to intuitively get a score for how a driver brakes during an unsafe event:

1. **Magnitude of Deviation**: By finding the area between the safe pattern and segment of individual drivers, overall deviation from the reference is calculated.
2. **Consistency**: The frequency with which the segment crosses over the reference pattern indicates how often the driver harsh brakes reflecting the inconsistency in their braking behavior.
3. **Euclidean distance between the deepest points**: The distance between the segment's deepest point and the reference pattern's deepest point is calculated to quantify the intensity of the braking and assess the deviation between the two points.
4. **Braking Intensity**: The position of the actual deepest point, whether above or below the reference pattern, is analyzed to determine if the driver exhibits an aggressive braking pattern.

These measures are well explained in the methodology chapter in section 3.7.1.

2.7.2 Evaluation of Driving Risks

The features a_{min}, a_{average}, and η_{E}, shown in Figure 2.7, constitute the feature vector $\mathbf{X}$, which is detailed in Section 3.3. Each segment is categorized into **Low**, **Moderate**, and **High** risk levels, with individual weights assigned in the range of 0 to 100, representing the quantified risk for each driver during unsafe events.

2.7.3 Evaluation of Activation Sequence Warnings

The **5 stages of CW-EB systems** given in Figure 1.1 are assigned an individual weight. Each signifies the impact it creates when it has occurred during the unsafe event. Based on the sequence encountered by the driver the score is given from 0 to 100.

2.7.4 Evaluation of False Positives

Classifying whether the leading vehicle **is Braking or is in Normal** i.e. idling/accelerating scenarios. The ego vehicle's driver is given a fault status of 0 or 1 to know whether the driver is actually at fault .

2.7.5 Grading the driver

Figure 2.11 shows the final safety score calculation as a weighted average of the braking pattern, driving risk, and activation warnings, combined with the fault status of the YOLO object detection. Together, these factors provide insight into the **intention of the driver and context in which the unsafe event has occurred**, formulating a score that is more accurate and reliable than the existing scoring method in Volvo Service Reports.

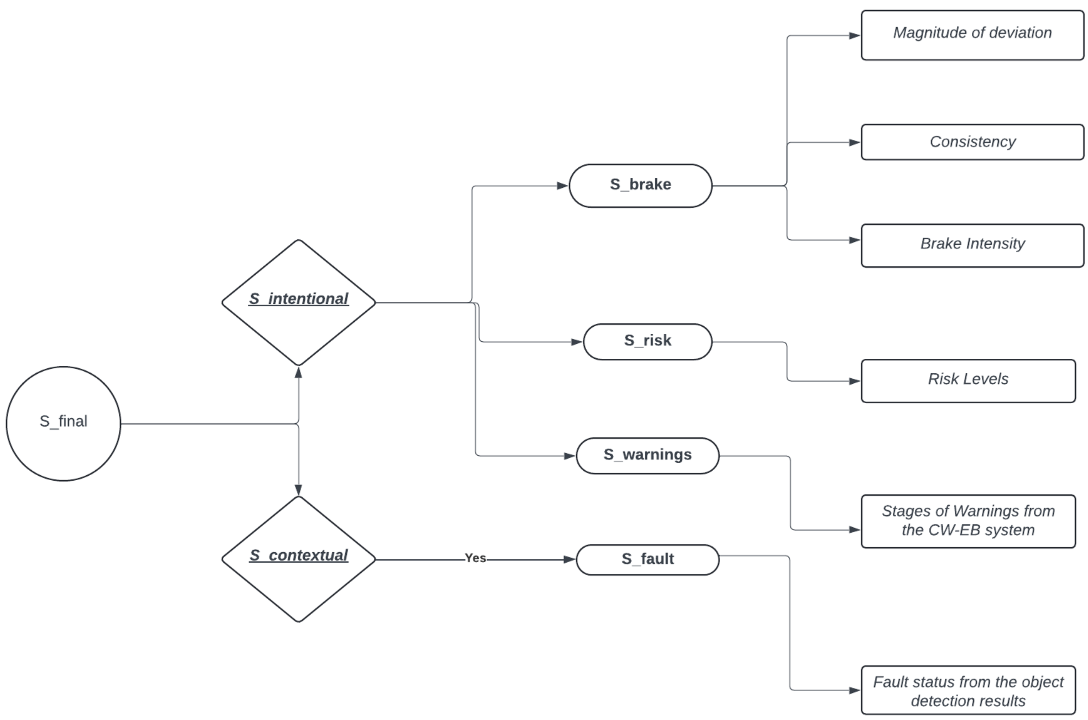


Figure 2.11: Overview of the safety scoring system

3

Methods

This chapter focuses on the methodology illustrated by Figure 3.1 that was implemented based on the findings of the characteristics of longitudinal acceleration to formulate a comprehensive safety scoring system. We focused on the following features: vehicle-id, reading-id, timestamps, longitudinal acceleration, and longitudinal velocity of the truck.

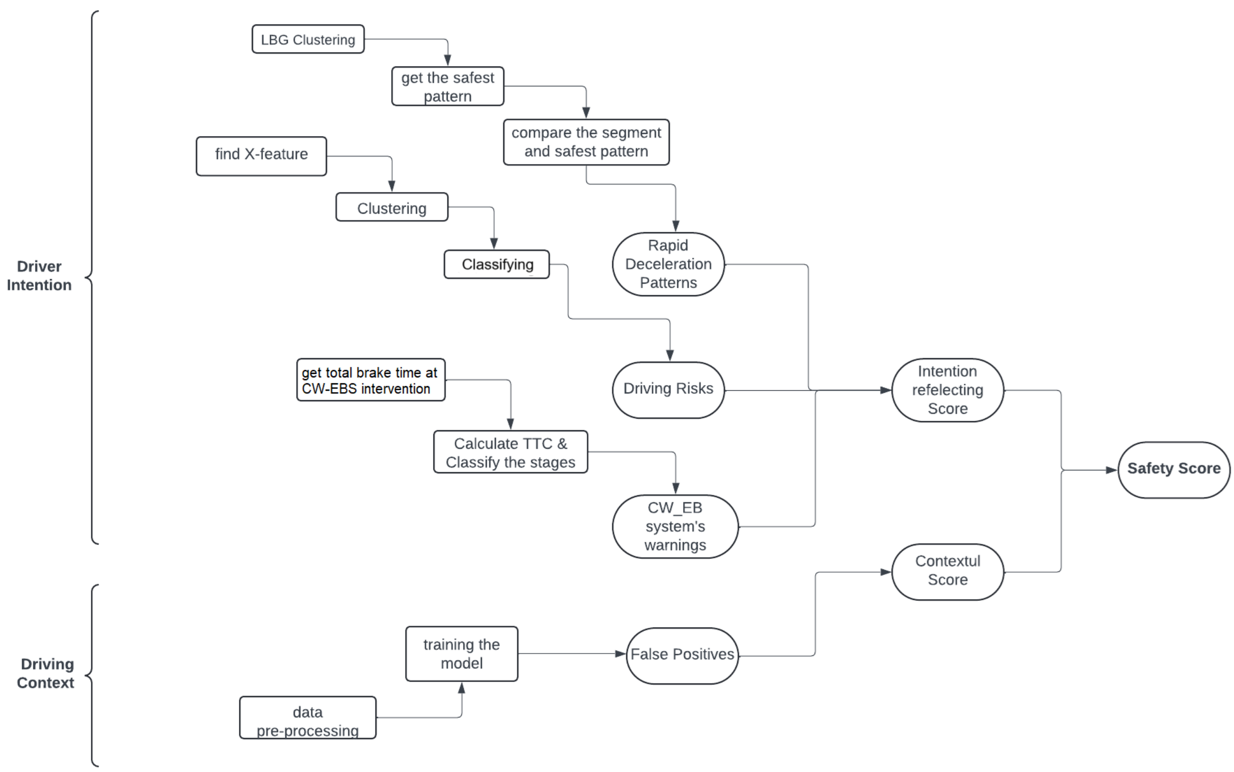


Figure 3.1: Working methodology of the Safety Scoring System

As in the above figure, the intention reflecting score is calculated based on the results from the Rapid Deceleration Pattern, Driving Risk and CW_EB warnings. The Contextual Score is calculated by finding the fault status. The entire safety score can be divided into $S_{\text{intentional}}$ and $S_{\text{contextual}}$.

3.1 Understanding the acceleration curve

The longitudinal acceleration measures the rate of change of a vehicle's forward speed, which gives a detailed picture of how the driver is operating the vehicle. The dynamic changes in the vehicle's longitudinal acceleration reflects the changes in the braking of the truck and by studying these deceleration during unsafe events (Pre-Brake and/or Full Brake events) will give categorical data to quantify the behaviour and reaction of the drivers. Referring to Figure 3.2, three major phases occur, the depressing phase, the brake occurring point, and the releasing phase. These phases refer to the brake pedal's position and how the position evolves in time to reduce the speed of the truck during an unsafe event.

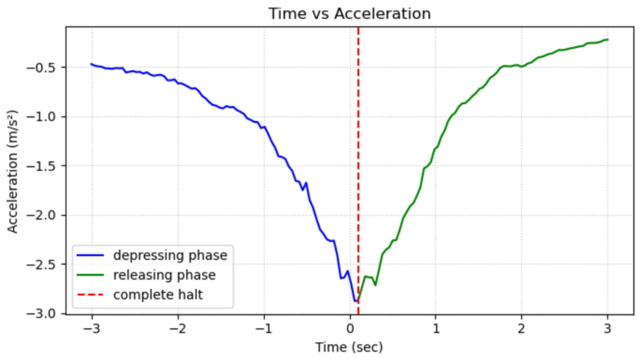


Figure 3.2: How braking and accelerating looks in vehicles

Based on the figure 3.2, three main phases were identified from the acceleration profiles of each RID, and a safety score was calculated to evaluate rapid deceleration patterns.

3.2 Data collection and Pre-Processing

Here, primary focus was provided to the dataset given by the ADAS team, which gave clear sensor signals of the vehicle's speed, acceleration, velocity, GPS etc., form which the required parameters such as listed in 3 were separately extracted for each VID and its respective RID at each instance in time using SQL queries.

For the analysis, the longitudinal acceleration and longitudinal velocity and its respective timestamps were used in most of the cases and these signals were smoothed using the Savitzky-Golay filter to retain the originality of the extracted signals to acquire the behaviour of the driver at that instance in time.

The Savitzky-Golay filter is a widely-used technique in signal processing aimed at reducing noise and enhancing the smoothness of signal trends [31]. It is a low pass filter and it is particularly used in noise smoothing and retaining of the original features of the signal in an odd-sized window intervals.

The smoothed signals as in figure 3.3 are used as such in the driving risk analysis and the negative accelerations of the smoothed signals were alone selected for understanding the braking behaviour of drivers during rapid deceleration pattern analysis.

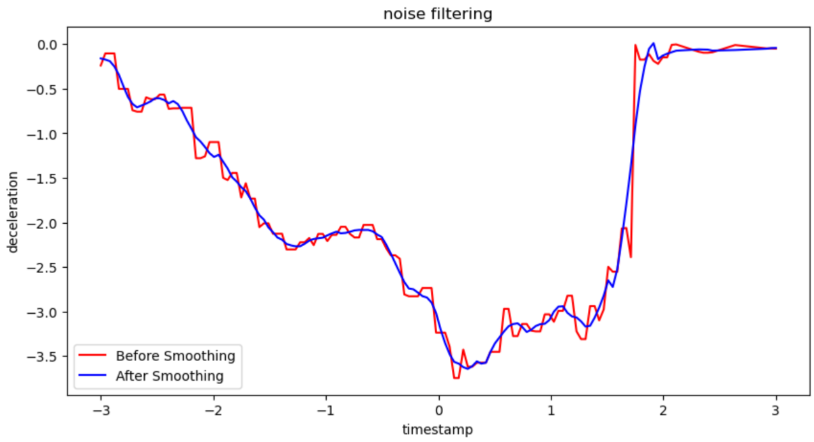


Figure 3.3: Savitzky-Golay Filtering with window size = 11

3.3 Classification of Driving Risk

The study of the acceleration patterns of the truck was conducted by extracting three features as in the figure 3.4 from the unsafe event. These are minimum deceleration a_{min}, average deceleration $a_{average}$ and percent reduction in kinetic energy η_E. Figure 3.4 shows the plot that visually illustrates the three features [22].

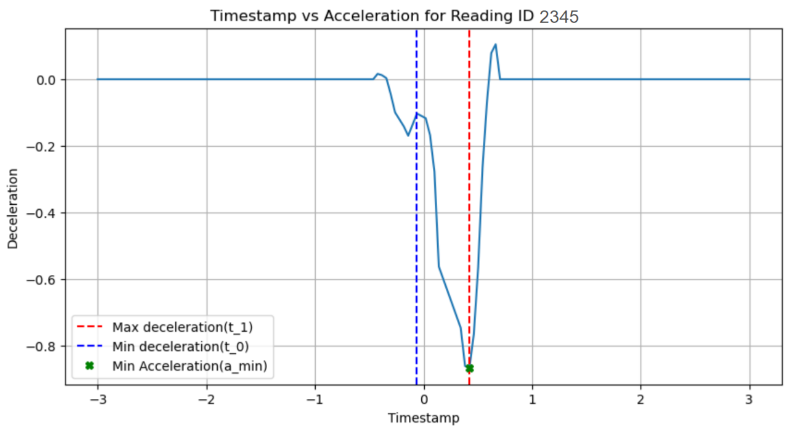


Figure 3.4: a_{min}, t_0 and t_1

The minimum deceleration captures the peak intensity of the braking, the average deceleration represents the overall rate of speed change and the percentage kinetic energy reduction indicates the total effect on the vehicle's motion. Hence these 3 parameters are estimated from the acceleration profiles of each RID as shown the figure 3.4 and used as the feature X in finding the risk levels.

3.3.1 Minimum Deceleration

The minimum deceleration is the lowest acceleration value during the braking process. A higher maximum deceleration indicates a more severe braking event, which typically corresponds to a higher risky scenario.

The minimum deceleration a_{min} is the smallest acceleration $a(t)$ value as in (3.1), that is captured when the unsafe event has occurred for a time instance of $-m$ to m,

$$a_{min} = \min_{t \in [-m,m]} a(t). \tag{3.1}$$

3.3.2 Average Deceleration

The average deceleration is calculated from the point the braking is triggered to the point of maximum deceleration as shown in the figure 3.4. It provides a measure of how gradually or abruptly the vehicle decelerates during an emergency braking event. The average deceleration $a_{average}$ can be calculated by the following formula,

$$a_{\text{average}} = \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} a(t), dt = \frac{1}{t_1 - t_0} [v(t_1) - v(t_0)] \tag{3.2}$$

where $v(t)$ and $a(t)$ denote the vehicle velocity and acceleration, respectively.

3.3.3 Percentage reduction in the kinetic energy

The kinetic energy reduction η_E is the percentage reduction in the vehicle's kinetic energy from the braking triggering point to the point of maximum deceleration. This metric reflects how much kinetic energy is dissipated during the braking event, which correlates to the severity and potential risk involved in a near-crash situation. The percentage reduction in the vehicle kinetic energy η_E can be calculated as following,

$$\eta_e = \frac{1}{2} M v^2(t_0) - \frac{1}{2} M v^2(t_1) = 1 - \left[\frac{v(t_1)}{v(t_0)} \right]^2 \tag{3.3}$$

where M is the vehicle's mass.

With these key features based on the acceleration and velocity profiles of each driver, the driving risk is clustered into the following table,

Table 3.1: Driving Risk Patterns

Driving Behavior	a_{min}	$a_{average}$	η_E
Aggressive	High	High	High
Smooth	Low	Moderate	Moderate
Efficient	Moderate	Moderate	Moderate

3.3.4 Normalizing the Ego Vehicle data

Calculating a_{min}, t_0, and t_1 from the existing data posed a challenging task. The information about the leading vehicle is only available at the time of intervention of the EB-CW system in the ego vehicle. To interpolate the acceleration and velocity data of the leading vehicle, only the data points within the time range of the brake triggering point and the brake occurrence point were considered. The mean approximations of these data points were used to calculate a_{average} and η_E.

Determining a_{min} was straightforward; however, identifying the exact point where the minimum deceleration occurred, i.e., the **trigger point of the unsafe event**, was more challenging. This was addressed by calculating the **gradient between consecutive deceleration points** from t_0 to t_1. If the gradient exceeded $0.3\,\text{m/s}^2$, that point was identified as the trigger point during the deceleration phase. Based on this criterion, the trigger point of the unsafe event was determined.

3.4 Classification of Rapid Deceleration Patterns

Two techniques were implemented in finding the rapid deceleration patterns to cluster the braking maneuver of each driver into **Emergent**, **Situation-aware**, **Intensive and long**, and **Moderate**, as in [32].

3.4.1 Vector Quantization vs. Time-Series K-means

The LBG clustering 3.4.2 method gave very good and valid results compared to that of the clustering results obtained from the Time-Series K-means clustering approach.

The clusters formed by the LBG method were quite reasonable and aligned well with the types mentioned here [32]. In contrast, the Time Series K-Means (TSKM) approach struggled to identify the same patterns, particularly for the **Emergent braking** pattern, which is the rarest and the most hazardous pattern. It accounted for 15 RID from a total of 189 RID, which is a relatively small amount of data for an ML model to detect such a pattern effectively.

LBG clustering was successful because it is a signal processing technique that creates codebook. It handles each data point for formulating a codevector that best represents its nature in the input data.

3.4.2 LBG Clustering Observations

Based on the depressing and releasing of the brake pedal at the time of the unsafe event, the clustered rapid deceleration data is classified as follows [32]:

1. **Intensive and long braking**: The driver used the brakes firmly and slowly while braking from a high speed. It belongs to cluster-id 0.

2. **Situation-aware braking**: The driver removed their foot from the brake and depressed it gradually before releasing it quickly because they were aware of the traffic condition ahead of time.It belongs to cluster-id 1.
3. **Moderate braking**: The driver decelerated gradually, slowed down and either halted or depressed the brake steadily before releasing it.It belongs to cluster-id 2.
4. **Emergent braking**: The driver quickly applied and released the brakes when the leading vehicle abruptly slowed down. It belongs to cluster-id 3.

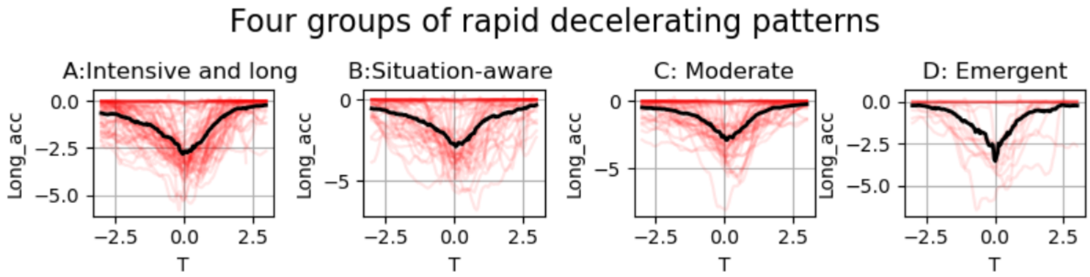


Figure 3.5: Observed Rapid Deceleration Patterns

From figure 3.5, the red signals represent acceleration curves that has been classified using LBG algorithm, and the black curve represent the mean approximation of the red sign. The mean approximation illustrate how smoothly the driver has transitioned from the depressing phase to releasing phase. It also indicates the sharpness of the braking in each type, the narrower the brake, the more dangerous the braking process becomes. Hence by looking into the narrowness of all the patterns, it can be seen that the safest deceleration pattern is **Moderate** as it is the least narrow in nature compared to all other patterns, and the most unsafe deceleration pattern is **Emergent** as it has the sharpest curve among the four patterns. The other two patterns, **Intensive and long** and **Situation-aware**, also come under the hood of unsafe decelerating patterns but being situation aware and making intense and long brakes are comparatively less unsafe than the **Emergent** type.

3.5 Time To Collision Warning

The Time-To-Collision is a critical factor in assessing and classifying potential forward collision scenarios involving the vehicle. As outlined in Section 2.4.1, TTC is computed to quantify the risk of an imminent collision. This metric plays a pivotal role in developing an index that aids in assigning safety scores to different driving events. By leveraging the TTC value, a systematic approach is employed to evaluate the proximity and likelihood of collisions, contributing to a robust framework for safety scoring [5].

```
# Calculating TTC
TTC = D(t)/V_d
# Appended to the dataframe as df['TTC_Risk_type']
```

```
if TTC < 0.5:
    disp('Overriding')
elif 0.5 <= TTC < 1.5:
    disp('Mitigating')
elif 1.5 <= TTC < 2.5:
    disp('Imminent warning')
elif 2.5 <= TTC < 3.3:
    disp('Cautionary warning')
else:
    disp('No Warning')
end
```

Based on the above criteria, the activation sequence of the CW-EB system is simulated and these are later evaluated based on the counts of the risk type on each vehicle-id and reading-id and given as a percentage to form the safety score.

3.6 YOLOv8 Object Detection

The events discussed previously do not have to be necessarily caused due to the driver's fault. There are various factors such as traffic density, sudden changes in direction or sudden braking by the leading vehicle, climatic conditions, etc., that can influence the truck and the driver's behavior. To mitigate this issue, we propose an idea assuming that the truck captures a video at the time of the events getting triggered. A YOLOv8 object detection model (Section 2.6) is trained and applied on the captured video to check if the ego vehicle has any sudden changes in braking. Understanding what caused the event can help build a less error prone scoring system.

The model identifies vehicles in front, and determines their braking status based on the illumination of brake tail lamps. If the tail lamp is ON, the model detects it as "Braking" and if the tail lamps are OFF, the model detects it as "Normal". When a warning or event from the CW-EB system occurs, a video is captured along with other ADAS parameters. The model analyses the recorded video and performs object detection to check if there is a leading vehicle and identifies its braking status. If the model detects any harsh braking activity from the leading vehicle, we can say that the warning was not caused due to the ego vehicle driver's fault but rather because of sudden braking activity by the leading vehicle.

3.6.1 Dataset

The data set used to train the model contains cropped images of vehicle tail lights in various stages and combinations of braking, non-braking, and turning, from a research that used the CNN-LTSM model to categorize car tail lights [33]. The dataset was then divided into two categories, 'Braking' and 'Normal', with 200 images in each category. The 'Braking' category includes all images with braking lights on (Braking, Braking and turning right, Braking and turning left). The

'Normal' category includes all images with braking lights off (All lights off, Turning left, Turning right).

3.6.2 Dataset annotation and splitting

In machine learning, annotation is the process of labelling images by drawing bounding boxes to provide ground truth for the objects in an image. This annotated image helps the machine learning model to learn patterns and make predictions accurately. The 211 images from each category of the dataset were manually annotated using Roboflow, an open source online annotation tool. Figure 3.8 shows a single image from each category labeled as Braking and Normal. After labelling all the images with their respective labels, the dataset was divided into 75% 'Training set', 15% 'Validation set' and 10% 'Test set'.

Figure 3.6: Braking label

Figure 3.7: Normal label

Figure 3.8: Braking and Non-braking annotation made in Roboflow.

The labelled dataset is then typically divided into three subsets: the training set, the validation set, and the test set. These subsets serve distinct purposes in the model development and evaluation process.

Training set

- **Purpose**: The training set is used to train the machine learning model. This means that the model learns from this data by adjusting its parameters (weights in neural networks) based on the input-output pairs in this set.
- **Content**: It contains the majority of the data, usually 70-80% of the total dataset.
- **Usage**: The model uses this data to learn patterns, make predictions, and minimize error.

Validation set

- **Purpose**: The validation set is used to tune the model's hyperparameters and to evaluate the model during training. It helps in selecting the best model and avoiding overfitting.
- **Content**: It generally contains 10-15% of the total dataset.
- **Usage**: During training, the model's performance on the validation set is monitored to adjust hyperparameters (like learning rate, depth of trees, number of layers in neural networks, etc.). It helps in making decisions such as early stopping when performance on the validation set starts to degrade.

Test set

- **Purpose**: The test set is used to evaluate the final model's performance after the training and validation processes are complete. It provides an unbiased assessment of the model's ability to generalize to new, unseen data.
- **Content**: It usually makes up 10-15% of the total dataset.
- **Usage**: The model does not see the test set during training or validation. The test set is only used once to assess the final model's accuracy, precision, recall, F1-score, and other performance metrics.

3.6.3 Training

Training is the process of teaching the machine learning model to identify and locate objects within images. This process includes several steps and uses a labeled dataset, 'Braking' and 'Normal' in our case, to help the model learn the features and patterns associated with various objects. Training a model usually requires GPUs (Graphics Processing Units) since the process is computationally intensive and involves large scale data processing, complex calculations and parallel processing. Considering the computational complexity, a cloud based Jupyter notebook environment called Google Colaboratory, or Colab [34] in short, was used. Colab provides free access to powerful hardware accelerators, including GPUs and TPUs (Tensor Processing Units).

The training process uses YOLOv8 pre-trained weights from the official YOLO repository [6]. The model was trained for 100 epochs. An epoch refers to one complete pass through the entire training dataset. During training, the model processes all the training data once to adjust its internal parameters (weights) to learn from the data.

3.7 Safety Scoring System

The final Safety Score is formulated as :

$$S_{\text{final}} = S_{\text{contextual}} + S_{\text{intentional}} \tag{3.4}$$

where,

$$S_{\text{intentional}} = w_1 \times S_{\text{brake}} + w_2 \times S_{\text{risk}} + w_3 \times S_{\text{warning}} \tag{3.5}$$

$$S_{\text{intentional}} = w_4 \times S_{\text{object}} \tag{3.6}$$

where w_1, w_2, w_3, and w_4 are weights that are given based on the **accuracy of the results obtained** from various evaluation techniques and these weights also help in obtaining the final score from each technique:

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0.2 \\ 1 \end{bmatrix}, \tag{3.7}$$

where,

- S_{brake} is the rapid deceleration pattern analysis score and it follows,

$$S_{\text{brake}} = \text{Magnitude of Deviation} + \text{Consistency} + \text{Brake Intensity} \tag{3.8}$$

- S_{risk} is the driving risk analysis score and it follows,

$$S_{\text{risk}} = \text{Low}\|\text{Moderate}\|\text{High} \tag{3.9}$$

- S_{object} is the classification of Braking or Normal scenario and it follows,

$$S_{\text{object}} = \text{Braking}\|\text{Normal} \tag{3.10}$$

- $S_{warnings}$ is the Activation Sequence score and it follows,

$$S_{\text{warnings}} = \text{no warning}\|\text{caution}\|\text{imminent}\|\text{mitigate}\|\text{override} \tag{3.11}$$

Let us look into VID: DFG and its respective RID: 236 and 813, to see how the S_{brake}, S_{risk}, S_{warning} and S_{object} are formulated to get the S_{final}.

3.7.1 Calculating the scoring metrics from Rapid deceleration braking

To get S_{brake}, we follow a metric to come up with a value to quantify the safest way to brake during an unsafe event. After finding the four types of rapid deceleration patterns, the Moderate pattern is considered to be the safest pattern. Each of the segments is synchronized [35] with the safest pattern, to form the score-determining factors that calculate and quantify how safely the driver brakes during an AEBS warning.

3.7.1.1 Magnitude of the deviation

The segment and safest rapid deceleration pattern are fundamentally two curves whose area can tell how much the segment deviates from the safest pattern. Hence the deviation is estimated by finding the integral difference between the two curves to get the area:

$$A = \int_{t_0}^{t_1} \left| f_{\text{safe}}(t) - f_{\text{seg}}(t) \right| \, dt, \tag{3.12}$$

where

- $f_{\text{safe}}(t)$ is the acceleration curve of the safest pattern;
- $f_{\text{seg}}(t)$ is the acceleration curve of the segment of individual RIDs.

Hence, the area indicates the amount of deviation from the safe pattern.

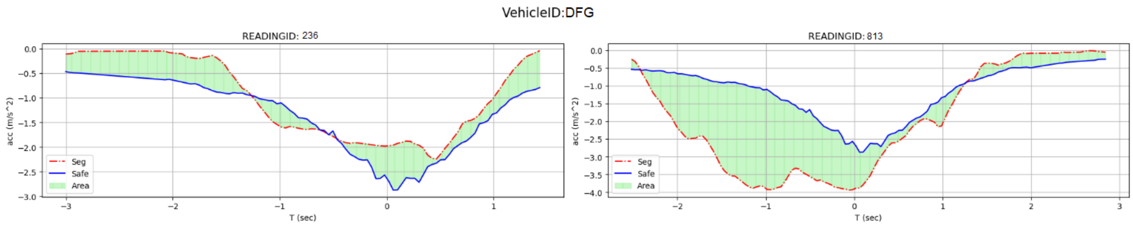


Figure 3.9: Area between the segment and the safest pattern.

3.7.1.2 Consistency of the Braking

By taking the number of times the segment and the safest pattern intersect, a conclusion can be made on how consistent is the driver in braking safely. The count of the intersection suggest the following:

- Multiple intersections infer that the braking behaviour of the driver is inconsistent and the driver is not driving safely.
- Lower number of intersections say that the driver is consistent in their braking behaviour.

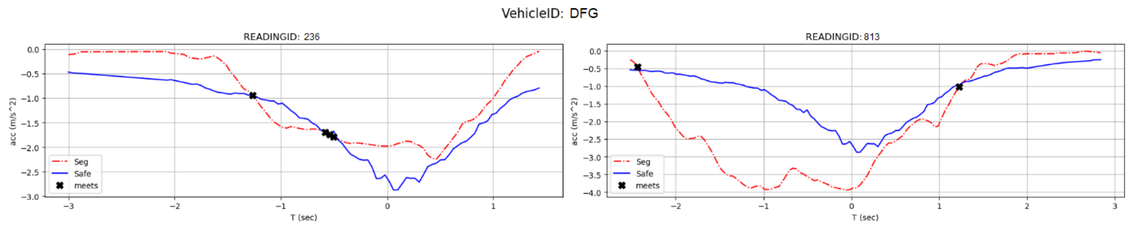


Figure 3.10: Crossings between the segment and the safest pattern

3.7.1.3 Braking Intensity

The braking intensity is calculated using the Euclidean distance between the deepest dips of the curves and a flag value that says whether the segment is below or above the safest pattern. The intensity is based on the distance between these two deepest points, while the flag is obtained by comparing the mean of the segment to the mean of the safest pattern. If the mean of the segment is not above the mean of the safe pattern, it is not safe. The Braking Intensity suggests that,

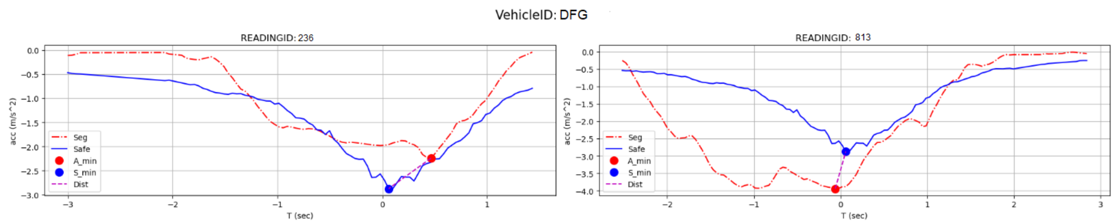


Figure 3.11: Intensity between the segment and the safest pattern

- If the segment's deepest point is below the safe pattern's deepest point, then the driver is more on the aggressive side of driving style.

- If the segment's deepest point is above the safe pattern's deepest point, then the driver is on the side of safer side of driving style.

The Euclidean distance indicates the level of aggressiveness the driver possesses.

From figures 3.9, 3.10 and 3.11 the scoring forming factors are represented for the VID: DFG and after finding these S_{brake} is calculated.

3.7.2 Calculating the score from Rapid deceleration braking

Now, the score-determining factors are formulated but these factors are eventually Euclidean distance, counts and area that needs to be converted into a score. Hence we penalize each factor with rules called penalty to calculate the S_{brake}. Penalty is

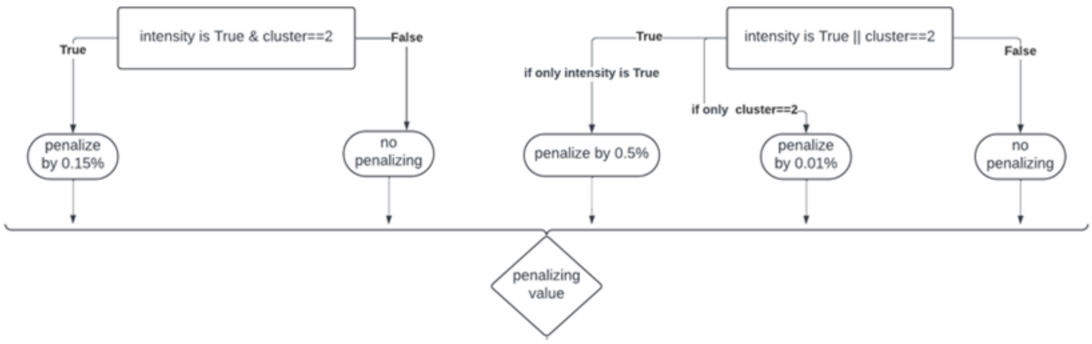


Figure 3.12: Percentage used for Penalty

an approach that reducing a portion as given in figure 3.12 of the formulated score S_{brake}. This is mainly done based on the intensity flag being True and cluster id being 2.

1. **Intensity Flag == True**
   - This value being true directly says that the driver is not braking in a safe way. As described in Section 3.7.1.3 , if the mean of the segment is not always high it is not safe.

2. **Segments belonging to Safe Pattern**
   - In this case, segments belonging to cluster-id 2 represent the safest way to braking. However, simply being part of cluster-id 2 does not automatically guarantee their safeness. The segments belonging to cluster-id 2 are further validated by score-determining factors to confirm whether they are truly safe. To ensure accuracy, they are penalized highly as shown in Figure 3.12.

Other than the above two conditions, all the segments, irrespective of which cluster id they belong to, where penalized based on the score-determining factor but the earlier discussed type is penalized highly. The score-determining factors are penalized in the following ways:

1. **Penalty on Area**:

- The goal is for segments within the same cluster to exhibit minimal deviation, creating a small overlap area.
- If a segment has a large area and a high-intensity flag but still belongs to cluster-id 2, it is not automatically considered safe. In such cases, a penalty of 0.15% is deducted from the final score.

2. **Penalty of Crossings**:
   - Segments from cluster-id 2 are expected to have more intersections with the same pattern.
   - Based on the number of intersections, a penalty is applied. If a segment has 4 to 8 intersections and it belongs to cluster-id 2, the final score is penalized by 0.05% to 0.1%.
3. **Penalty of Distance**:
   - For cluster-id 2 segments, the distance from the safe pattern should be smaller than the median of the calculated Euclidean distance. However, if the distance is larger than the median of the calculated Euclidean distance and the intensity flag is true, a penalty is applied.
   - If the distance is larger than the median, the segment belongs to cluster-id 2, and intensity flag is True, a penalty of 0.15 is deducted from the final score.

Now the penalizing values are determined and the score S_{brake} has to be calculated, the score for braking is a simple percentage calculation based on the ratio of the current score-determining factor to its largest value. After finding the score-determining factor, the median and max values are found and used in the final score calculation as shown in Figure 3.13. The score S_{brake} is obtained as a weighted average of these factors as discussed in Section 3.7 and given as a Score in Table 3.2.

VID	RID	Cluster	Area	Distance	Crossings	Intensity	S_brake
DFG	236	2	1.877	0.748	4	False	74.68
	813	1	5.473	1.074	2	True	74.60

Table 3.2: Rapid Deceleration Pattern Results for Vehicle ID: DFG

3.7.3 Calculating the score from driving risk

The driving risk is determined by calculating **X**, which is derived using (3.1), (3.2), and (3.3). The parameters for **X** are obtained from Figure 2.7, with a detailed explanation provided in Section 3.3 and the extraction process described in Section 3.3.4 .

The driving risk is obtained by grouping the a_{min}, $a_{average}$, and η_E values through K-means clustering, the mean of the clustered groups are taken into consideration to label the X into Low, Moderate, and High risk, which infers to the Efficient, Smooth, and Aggressive driving behaviours of Table 3.3.3.

The mean values in Figure 3.14 help in deciding the risk levels on the acceleration values and categorizing the data points into the above discussed driving behaviours.

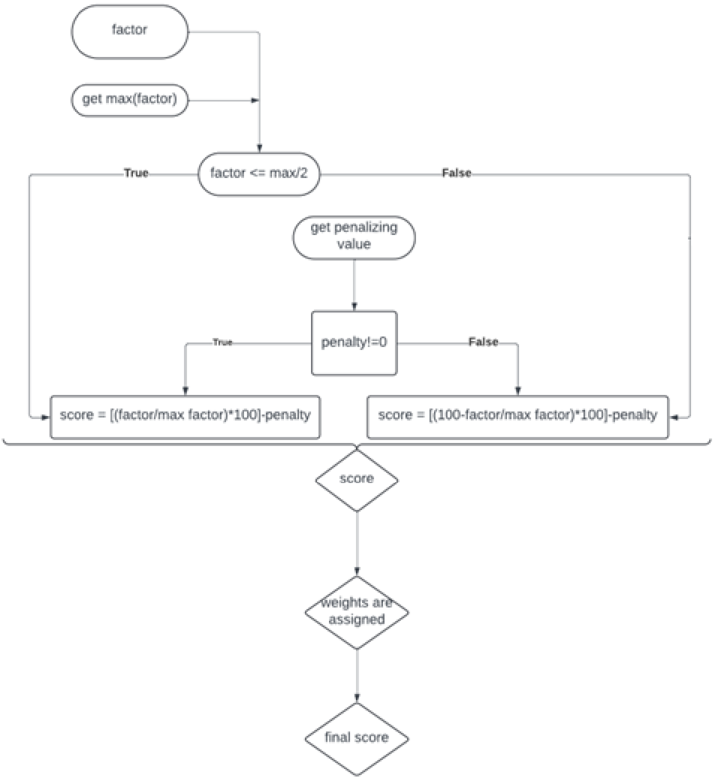


Figure 3.13: Scoring using the chosen penalty

```
Mean feature values for each risk level:
Low Risk:
a_1        -4.626413
avg_acc    -4.433064
eta         0.418484
dtype: float64

Moderate Risk:
a_1        -3.131205
avg_acc    -2.208475
eta         0.457599
dtype: float64

High Risk:
a_1         -1.112860
avg_acc    -15.837952
eta          0.934769
dtype: float64
```

Figure 3.14: Mean of the risk clusters

The score is assigned as weights to the final safety score, based on the type of driving behaviour that the driver exhibits and varies between 0 and 100.

RID	a_min	avg_acc	eta	risk_level	S_risk
236	-2.244806	-2.437463	0.600882	Moderate	50.0
813	-3.941707	-3.867659	0.370885	Low	100.0

Table 3.3: Risk levels of the VID:DFG

3.7.4 Calculating the score from CW-EB system warnings

The five stages of the activation sequence, Figure 1.1, are found with the total brake time during the intervention of the CW-EB system and with the help of the TTC principle, a score between 0 and 100 in Table 3.4 is based on the conditions in Section 3.5.

VID	RID	TTC	Warnings	S_warnings
A	236	0.80	Mitigate	40.0
A	813	0.28	Override	20.0

Table 3.4: Warning Level of the VID: DFG

3.7.5 Calculating the score from object detection

The contextual score from the object detection is essential to the final safety score as it says if the driver is truly at fault or not. It explains that the driver perceives an unsafe event because of the leading vehicle's unplanned and careless driving style, and to tackle this situation the driver performs a pre-brake or full-brake which is indeed a good and thoughtful reaction to the currently perceived situation, safely handling the danger of near crashing.

Section 3.6 explains on how the braking status with tail lamp of leading vehicle is used for creating the fault status. The results from the object detection is gathered into a boolean status as 0 or 1, which tells whether the unsafe event has occurred due to the driver's inattentive behaviour or not. From the object detection results the score is simply obtained by multiplying with 100. The S_{obj} for VID:DFG is shown in Table 3.5.

3.7.6 The final Safety Score

Finally, the safety score is calculated based on all the parameters explained above and it is the weighted average of all the parameters, for the VID: DFG the final score looks like the following,

VID	RID	S_{brake}	Warnings	S_{warnings}	Risk	S_{risk}	S_{obj}	S_{final}
DFG	236	74.68	Mitigate	40	Moderate	50	100	66.17
	813	74.60	Override	20	Low	100	100	73.65

Table 3.5: Final Safety Score for Vehicle ID: DFG

Note: The weights w_1, w_2, w_3, and w_4 of (3.7) could have been equally assigned, but here the values were assigned based on the accuracy of the results obtained from each evaluation technique. The most inaccurate results were obtained from the K-means evaluation of driving risks and it is given the least weight. The other evaluation methods are equally assigned weights .

4
Results and Discussions

This chapter focuses on the major findings of the thesis and discusses about the results obtained from each of the evaluation technique that was developed from formulating the safety scoring system.

4.1 Classification of Rapid Deceleration Pattern

As in Section 3.4 two techniques were used for finding the rapid deceleration pattern that occurred during an unsafe event, and it was observed that the LBG clustering performed best. In this section, both of the results are attached for a clearer picture of how the clustered rapid deceleration curves looked.

4.1.1 Clustering Outcomes - LBG algorithm

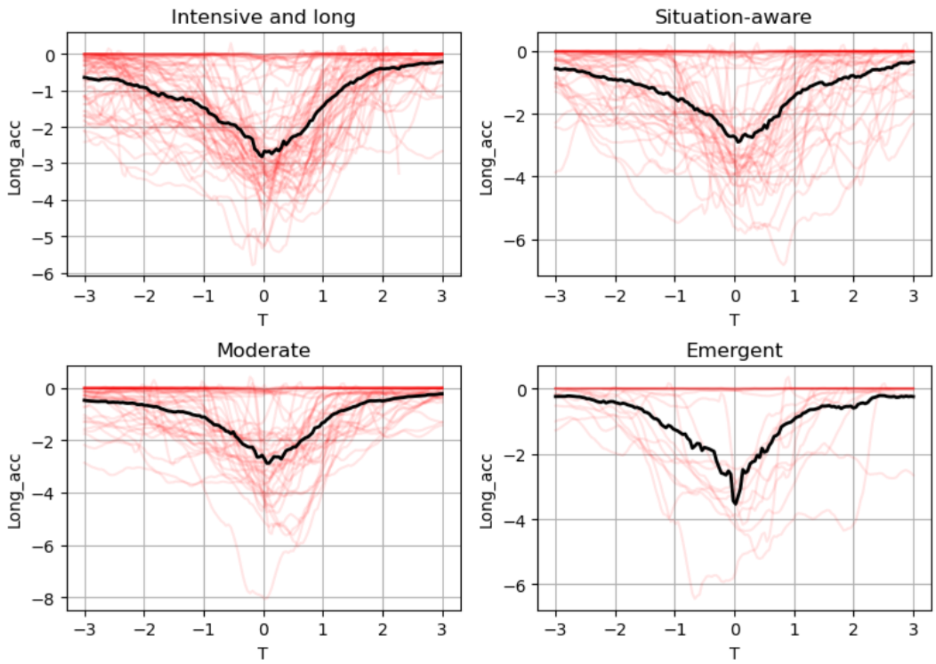


Figure 4.1: Rapid Deceleration Patterns observed from the Acceleration data using the LBG algorithm

By taking the mean approximation of the safest rapid deceleration pattern as the base for the calculation, the magnitude of deviation, consistency, and braking intensity were evaluated by synchronizing and overlapping the signals.

The histogram of the $final\_score$ shown in Figure 4.2 infers that a total of 133 RID have a safety score of above 60 and most of the drivers come under the hood of safe driving style. While the other histograms refer to the score-determining factors which were used in calculating S_{brake} as discussed in Section 3.7.1.

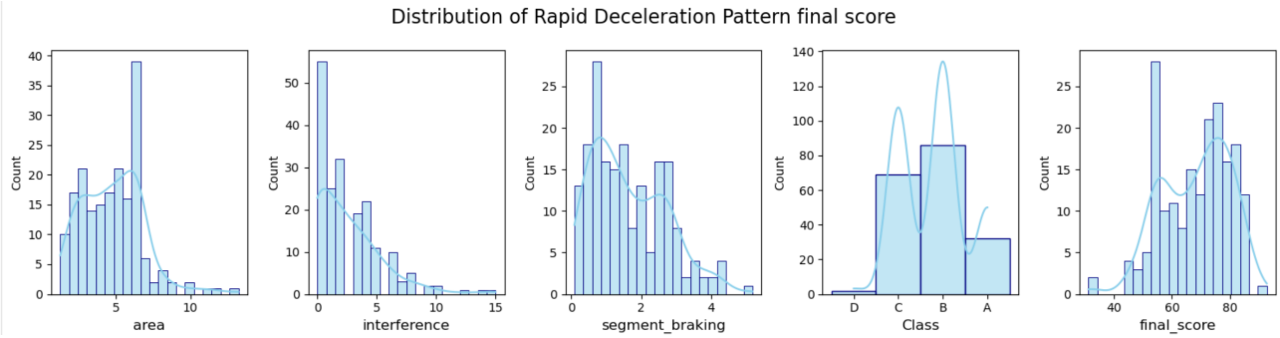


Figure 4.2: Distribution of the score formulating parameters from the S_{brake}

4.1.2 Clustering Outcomes - TSKM

The results obtained from the TSKM did not provide any significant deceleration pattern, it failed to distinguish the hidden difference in the acceleration curves. From figure 4.3, it can be seen that the mean approximation of the acceleration curves doesn't have the Emergent type or Moderate type of rapid deceleration pattern as in [32] which is the base for forming the S_{brake} as discussed in 3.7.1.

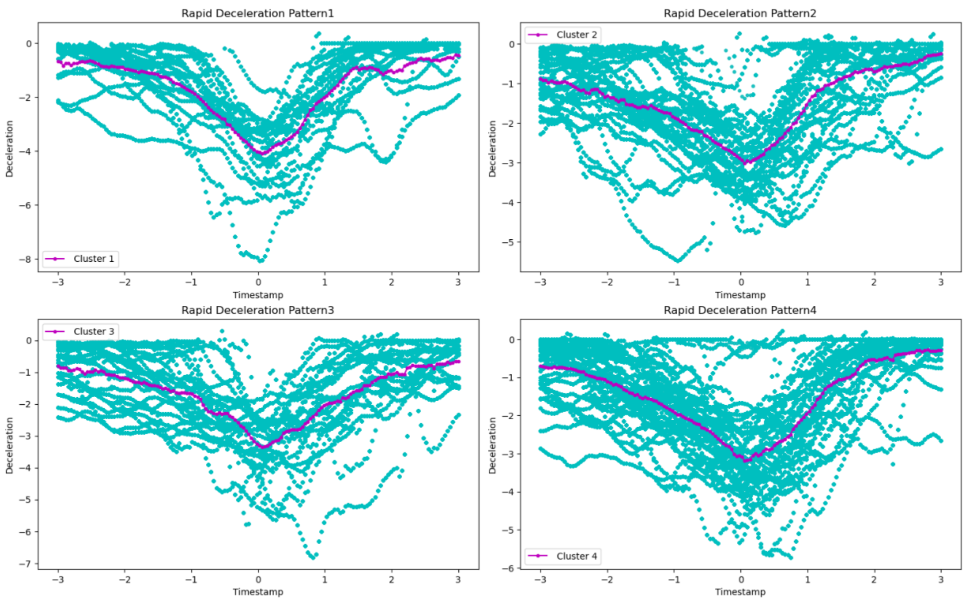


Figure 4.3: Rapid Deceleration Patterns observed from the Acceleration data using Time-Series K-Means

The reasons why LBG algorithm worked the best are as follows.

1. The ϵ value helps in formulating the converged codevector as discussed in Section 2.2.1. This tuning-parameter adds and subtracts the data-point in an iterate way as explained in Figure 2.2 to achieve the best codebook that represents the entire dataset.
2. Smoothing of the acceleration curves with a sliding window size of 11 using Savitzky-Golay filter became advantageous. The data pre-processing steps helped in obtaining the required results, refer to Section 3.2 for more explanation.

4.2 Classification of Driving Risks

We focused on risk factors X that reveals the driver's behavior in near-crash scenarios. These scenarios are defined as situations where drivers must perform emergency braking to avoid a collision with the leading vehicle [22].

Figure 4.4 illustrates the clustering of different RID into Low, Moderate, and High driving risks and it can be seen that,

- The majority of drivers fall into the Moderate and Low driving risk category.
- Only 1 out of 183 RID is classified as High driving risk, as shown in the distribution in Figure 4.5. It suggests only 1 RID is behaving Risky will the previous evaluation technique insist that there are drivers who have S_{brake} below 60 out of 100(almost 50 drivers).

These findings suggest that Volvo truck drivers are generally attentive and careful in their driving style. According to the risk categories outlined in Table 3.3.3, the vast majority of these drivers can be classified as Smooth and Efficient drivers.

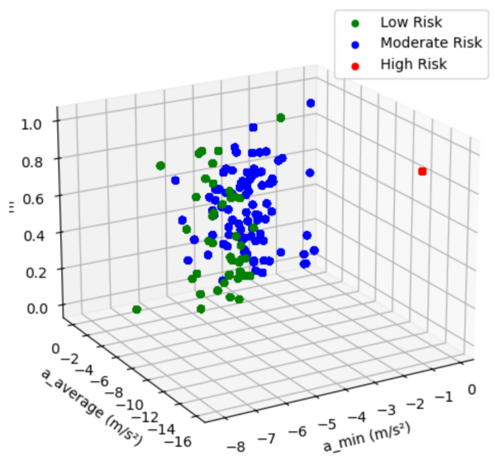


Figure 4.4: Driving risk analysis based on the X of drivers by K-Means

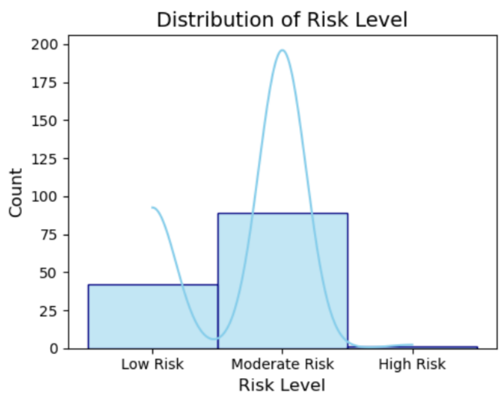


Figure 4.5: Distribution of the driving risk over the data

Note: Unsupervised learning is an ML model that requires huge categorical data. In this case, full-brake and pre-brake events are used for the analysis and they are the categorical data for the CW-EB system. Full-brake event is the **rarest of all**

the events that can occur in a near crash event in the CW-EB system. Since this event occurs very rarely, the model leverages this small class of events leading to class imbalance in the dataset. Due to this reason, the driver risk analysis of Section 3.3 is given the least importance in the final safety score calculation.

4.3 Warning stages of the CW-EB System

Heavy-duty vehicles like trucks in general need the TTC incorporated in their active safety systems due to the larger size, heavier weight, and longer stopping distance. It helps drivers and active safety systems anticipate potential collisions and take appropriate actions to avoid them.

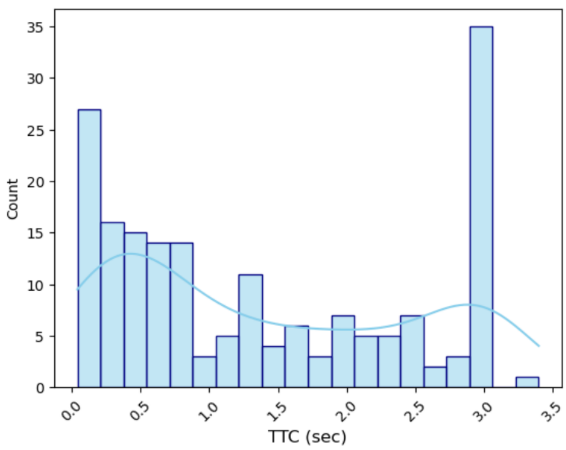


Figure 4.6: Distribution of TTC value for 183 reading-ids.

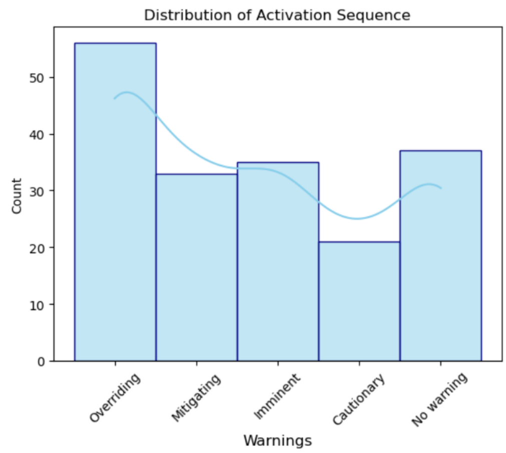


Figure 4.7: The five stages of the Activation Sequence and its frequency of occurrence in the extracted data.

From Figure 4.6, it can be seen that the drivers in the overriding and mitigating phases are in trouble as they have very less time to avoid the near collision. Hence these drivers can lead to a near crash event while the remaining 133 drivers reflect no potential threat.

Note: The TTC calculation and the warning stages are simulated with the data available from the ego vehicle only at the time of intervention of the AEBS. Section 3.3.4 describes how the ego vehicle data is normalized and Section 3.5 shows how the relative distance and relative velocity for the leading and following vehicle were calculated.

4.4 Object Detection

4.4.1 Model Performance Overview

The model successfully categorized vehicles into two states: "braking" (brake lights illuminated) depicted in Figure 4.8, and "normal" (no brake lights) depicted in Figure 4.9. It was trained on a limited dataset containing images of vehicles with and without illuminated brake lights. The dataset comprised 211 images, split into training and validation sets. Despite the constrained size of the dataset, the model demonstrated a basic capability to distinguish between the two states.

Figure 4.8: The YOLOv8 model detects braking vehicles.

Figure 4.9: The YOLOv8 model detects non-braking vehicles.

4.4.2 Quantitative Results

Table 4.1 shows the quantitative results of the object detection model.The results shows varying performance between the "Braking" and "Normal" classes. While the precision is higher for "Normal" instances (88.9%), recall is higher for "Braking" instances (63.3%), indicating a trade-off between detecting braking instances accurately versus minimizing false positives in normal instances.

Class	Images	Instances	Precision	Recall	mAP@50
all	20	56	0.761	0.509	0.719
Braking	11	30	0.633	0.633	0.699
Normal	9	26	0.889	0.385	0.721

Table 4.1: Evaluation metrics for object detection model.

- Precision: Measures the accuracy of positive predictions. For the “all” class (combining braking and normal), it achieves 76.1%, meaning that 76.1% of detected instances were correctly identified as braking or normal.
- Recall: Indicates the percentage of actual positives that were correctly identified by the model. It achieved 50.9%, meaning that 50.9% of actual braking or normal instances were correctly detected.
- mAP@50: Mean Average Precision at an Intersection over Union (IoU) [36] threshold of 0.5. This metric is crucial for object detection tasks, indicating the model’s ability to accurately localize objects. A value of 0.719 for the “all” class suggests good performance in object localization.

4.4.3 Challenges

While the model showed initial promise, several challenges were identified that impacted its overall performance:

- **Limited Dataset Size**: The unavailability of dataset and the small size of the training dataset restricted the model’s ability to generalize effectively, resulting in lower accuracy and higher variability in predictions.
- **Unilateral detection**: The brake lights in some images are overexposed due to sunlight, making them appear inactive even when the vehicle is braking. This confuses the model, causing it to detect only a single brake light leading to unilateral detection.
- **High-Resolution Image Performance**: The model struggled with high-resolution images, where detection accuracy significantly dropped due to inadequate feature learning from such images.
- **False Positives and False Negatives**: The model occasionally misclassified the braking status, resulting in both false positives and false negatives.
- **add stuff around unilateral detection**

4.5 Formulating the Safety Score

The final score is the weighted average of the rapid deceleration pattern, the driver’s driving risk levels, and the warning stages from the activation sequence of the CW-EB systems as given in Section 3.7. Additionally, the results from object detection are included to check if the unsafe event occurred primarily due to the driver not being attentive enough or not.

The scores for 183 RIDs are displayed in Figure 4.10, and it is evident that most of the RID have a safety score ranging from 60 to 95, i.e, a total of 50 drivers are

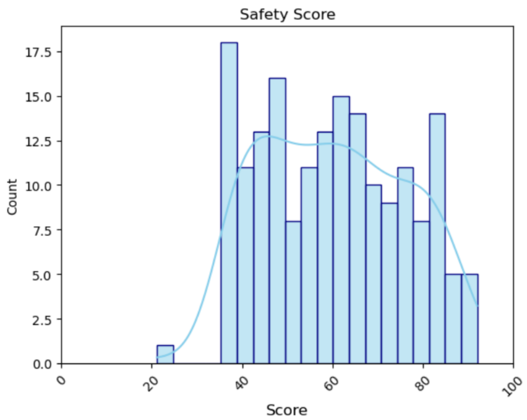


Figure 4.10: Final Safety Score

below 60 out of 100. RIDs with scores below 40 are identified as drivers requiring additional coaching on safe driving practices. Those with scores below 40 might be considered outliers due to their emergent rapid deceleration style, high driving risk, and tendency to override and mitigate warnings, which potentially makes them liable for unsafe braking events. All these parameters are checked along with the fault status obtained from the object detection and then the final score is evaluated.

The below table shows the top 20 RID that have very good safety scores out of the 183 RIDs,

RID	S_{brake}		S_{risk}		$S_{warnings}$		S_{object}	S_{final}
	class	score	level	score	phases	score	fault	Score
416	B	78.08620383676126	No warning	100.0	Low Risk	100.0	100	94.52155095919031
510	B	72.87150359274827	No warning	100.0	Low Risk	100.0	100	93.21787589818706
100	A	81.44165707761766	Cautionary warning	80.0	Low Risk	100.0	100	90.36041426940442
847	C	58.60798323745577	No warning	100.0	Low Risk	100.0	100	89.65199580936394
092	B	70.18297450511324	Cautionary warning	80.0	Low Risk	100.0	100	87.54574362627831
246	A	82.17407869190723	Imminent warning	60.0	Low Risk	100.0	100	85.5435196729768
433	B	78.34812457083287	Imminent warning	60.0	Low Risk	100.0	100	84.58703114270821
049	C	61.93420471461793	No warning	100.0	Low Risk	75.0	100	84.23355117865448
618	C	54.65145165442023	Cautionary warning	80.0	Low Risk	100.0	100	83.66286291360505
710	C	57.51157370738023	No warning	100.0	Low Risk	75.0	100	83.12789342684505
835	B	72.21729644735589	Imminent warning	60.0	Low Risk	100.0	100	83.05432411183898
541	A	81.11073463587148	No warning	100.0	Moderate Risk	50.0	100	82.77768365896787
356	C	55.11724817985999	No warning	100.0	Low Risk	75.0	100	82.529312044965
203	C	53.77751243966272	No warning	100.0	Low Risk	75.0	100	82.19437810991568
477	B	78.66565241554412	No warning	100.0	Moderate Risk	50.0	100	82.16641310388603
633	B	76.767918344934	No warning	100.0	Moderate Risk	50.0	100	81.6919795862335
410	B	76.00177962335832	No warning	100.0	Moderate Risk	50.0	100	81.50044490583957
784	C	50.82549313411392	No warning	100.0	Low Risk	75.0	100	81.45637328352848
427	C	59.34012241402464	Cautionary warning	80.0	Low Risk	75.0	100	78.58503060350617
524	A	84.26686423503327	Cautionary warning	80.0	Moderate Risk	50.0	100	78.56671605875832

Table 4.2: Safety Score for the top 20 Drivers out of 183.

4.6 Comparing and Validating

The existing methodology at Volvo use **histograms** to evaluate unsafe events for the distance travelled by the driver in the truck is a superficial approach that does not consider the underlying cause of the occurrence; is it really the fault of the driver for harsh braking to occur or is it because of the leading vehicle?

All these issues are addressed through the methodology presented in this thesis, as illustrated in Figure 3.1. The methodology evaluates drivers using **unsupervised learning and object detection**. It categorizes drivers based on **Intentional and Contextual Scores**, as depicted in Figure 3.1. This process ultimately forms the Safety Scoring System for Volvo truck drivers.

- **Intentional Scoring System**:
  A time series data reflects the traffic situation in which the driver is travelling and this reflects the intentions the driver has before making an unsafe event.
  1. A score that reflects how safely the driver brakes during an unsafe event.
  2. A score that reflects the potential of the driver to cause a near crash event.
  3. A score that reflects the time range in which the crash could have occurred.

- **Contextual Scoring System**:
  A video from the dash cam of the truck is processed to come to a conclusion on who is actually at fault during an unsafe event.
  1. A score is assigned based on the braking status that is obtained by processing the detection of tail lamps of the leading vehicle.

The formulated methodology does not consider the counts of risky events, rather it evaluates each driver based on their braking behaviour, their potential threat to cause a near crashing, and identifying the false positive scenarios that occur during an emergency braking event (Braking or Normal detection from YOLO). By using these evaluations we validate the drivers precisely and accurately range them in a scale of bad to good drivers.

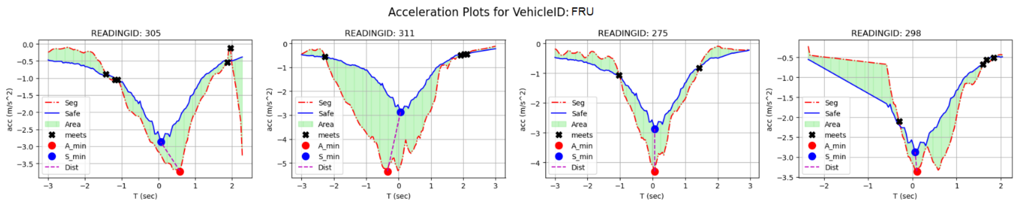
5

Illustrating the Safety Score

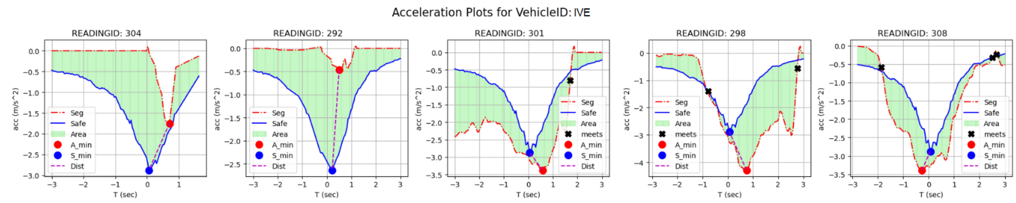
Randomly, 3 VID were considered and their respective RID were also taken, results are illustrated to show how the final safety score was calculated.

5.1 Intentional Score

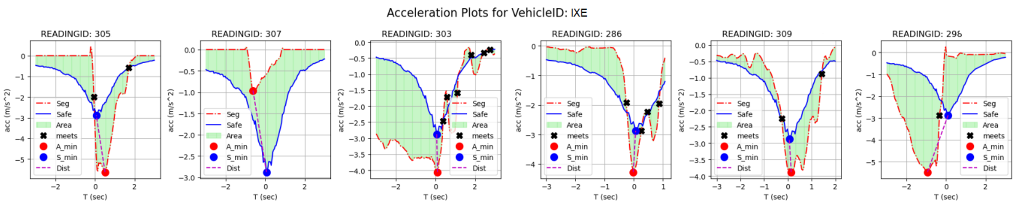
Results from S_{brake} for the randomly chosen VID suggest comparing the mean of the safest deceleration pattern with the segments, and the score quantifying techniques are found below,



Vehicle ID: FRU



Vehicle ID: IVE



Vehicle ID: IXE

Figure 5.1: The area, intersections, and the straight line distance for VIDs FRU, IVE, and IXE

The above figure shows,

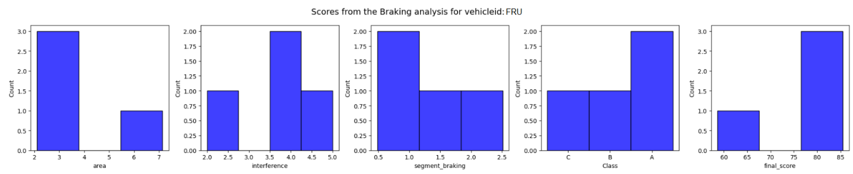
1. In VID: FRU
   RID: 298 has a good safety score among other RIDs.

   (a) RID: 298 has the smallest area.
   (b) RID: 275 has the least number of intersections with the safest pattern.
   (c) RID: 298 has the smallest Euclidean distance.

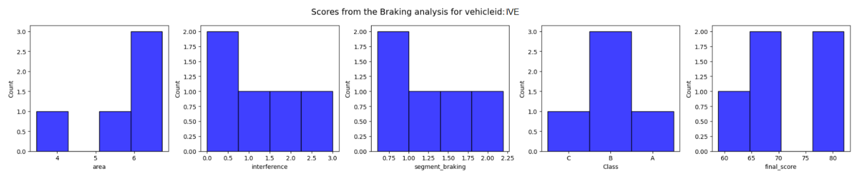
2. In VID: IVE
   RID: 308 has a good safety score among other RIDs.
   (a) RID: 308 has the smallest area.
   (b) All the RIDs had the least number of intersections with the safety pattern, especially the RID: 301 and RID: 292 have no intersections with the safest pattern.
   (c) RID: 308 has the smallest Euclidean distance.

3. In VID: IXE RID: 297 has a good safety score among other RIDs.
   (a) RID: 297 has the smallest area.
   (b) RID: 303 has no intersections with the safest pattern.
   (c) RID: 297 has the smallest Euclidean distance.

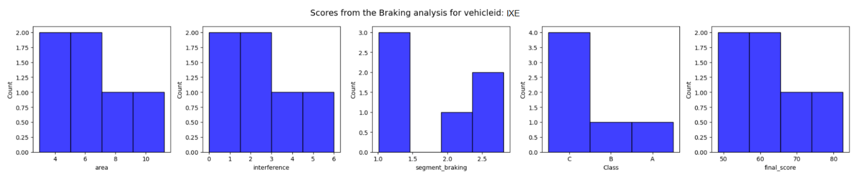
The VID: FRU has the best score for S_{brake} among the three VIDs. The histograms below helped in validating and understanding the impact that S_{brake} has in the final safety score.



Vehicle ID: FRU



Vehicle ID: IVE

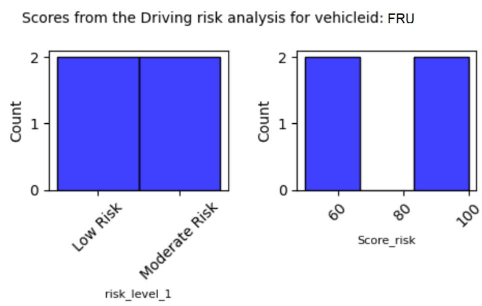


Vehicle ID: IXE

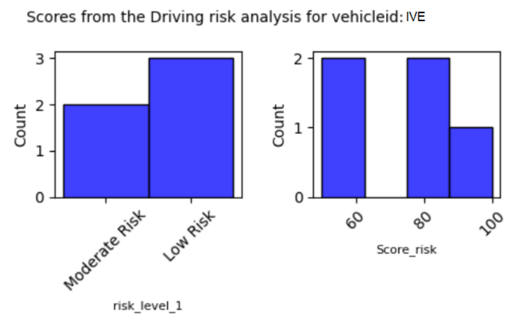
Figure 5.2: Magnitude of the deviation, Consistency of Braking, and Braking Intensity of VIDs FRU, IVE, and IXE with the score S_{brake} and its class.

Results from S_{risk} suggest to classify the drivers based on their potential to cause a near crash. As explained in section 3.7.3, the drivers are classified into a range of low to high risk levels and each of the chosen VIDs say,

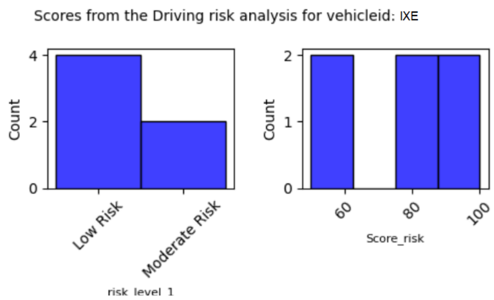
1. None of the VIDs have high-risk levels.
2. Most of the VIDs have low-risk levels.
3. Among 15 RIDs, 9 RIDs have low-risk level and 6 RIDs have moderate-risk level.



Vehicle ID: FRU



Vehicle ID: IVE



Vehicle ID: IXE

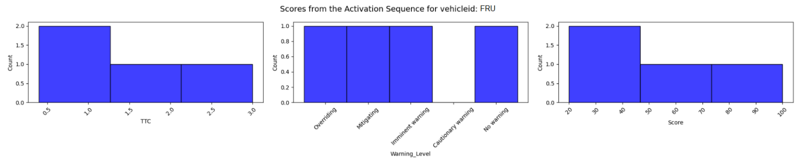
Figure 5.3: Driving Risk Levels and its Score for the sequence of VIDs FRU, IVE and IXE with the final score and class.

Results from $S_{warnings}$ classifies drivers based on their time left for the ego and the leading to collision. The warnings refer to the stages of the potential near crash situation that the drivers could cause and the chosen VIDs say,

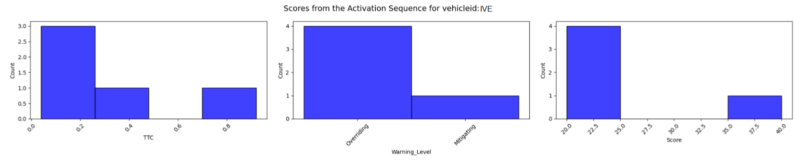
1. In VID: FRU RID: 305 has the best score for $S_{warnings}$.
   (a) RID: 311 had no warning.
   (b) RID: 305 had Overriding warning.

2. In VID: IVE RID: 301 has the best score for $S_{warnings}$.
   (a) Most of the RIDs have Overriding warnings.

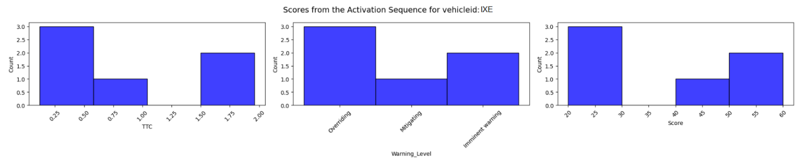
3. In VID: IXE RID: 303 and RID: 286 have the best score for$S_{warnings}$.
   (a) 3 out of 6 RIDs have Overriding warnings while 2 out 6 RIDs have Imminent warnings.



Vehicle ID: FRU



Vehicle ID: IVE



Vehicle ID: IXE

Figure 5.4: TTC, Warning Levels, and Score for the sequence for the Vehicle-Ids FRU, IVE and IXE

5.2 Contextual Score

There were some mis-classifications and unilateral detections (only one of the two brake lights were detected) as in 4.4.3 done by the model. This occurs on images where the brake lights of the leading vehicle is exposed more to sunlight, as shown in right corner of the Figure 5.5. This exposure confuses the model to detect it as a normal situation even if the vehicle is braking. Another scenario is when the target vehicle has a fault in the brake lights. If the brake lights are not working in the leading vehicle, the model mistakes it as a normal situation as shown in the middle image from Figure 5.5. Using a larger dataset for training the model with a variety of pre-processed vehicle images can solve the issue.

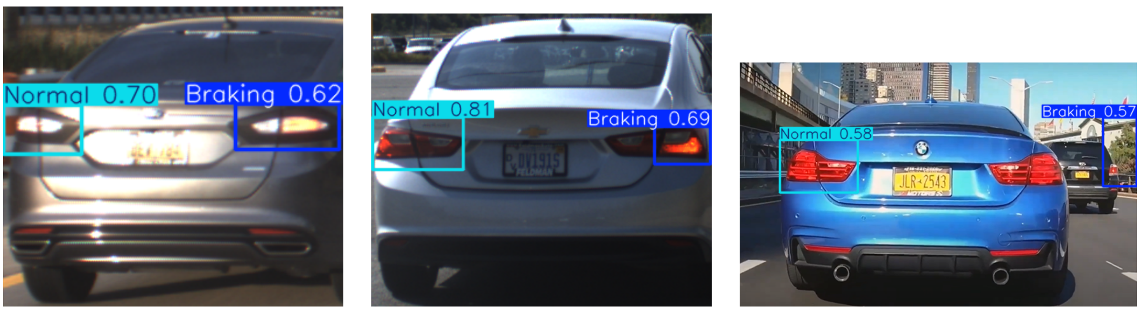


Figure 5.5: False positive detection of leading vehicles.

5.3 Final Safety Score

Finally, the safety score is calculated as a weighted average of the intentional and contextual score as in Section 3.7.

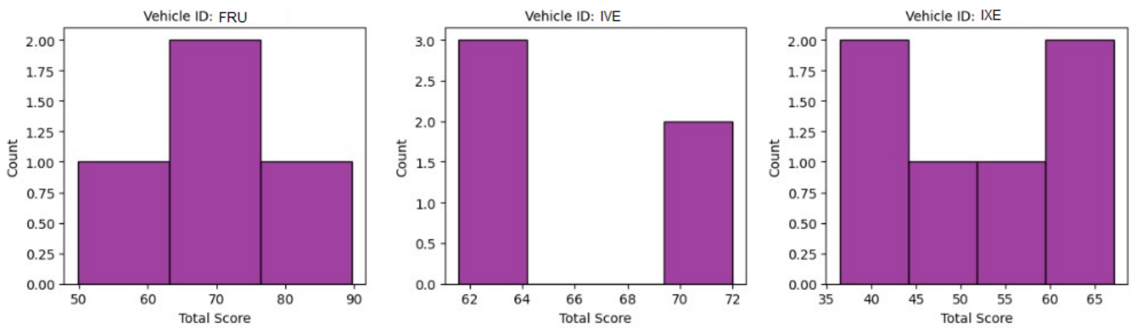


Figure 5.6: Final Score for the vehicle-ids FRU, IVE and IXE.

From the above histogram, the RID: 311 has the highest score and the RID: 304 has the lowest score from the final safety score. The below table is an elaborated representation of the final safety score obtained from the formulated safety scoring method for the chosen VIDs. To understand how the score is exactly calculated, going through the methodology section will be a great start.

RID	S_{brake}					S_{risk}		$S_{warnings}$		S_{object}	S_{final}
	Area	Consistency	Distance	Intensity	Score	Level	Score	Phases	Score	Fault	Score
305	2.739562	5	1.004570	True	79.252020	Overriding	20.0	Low Risk	100.0	0	49.813005
311	7.135753	4	2.509570	True	58.607983	No warning	100.0	Low Risk	100.0	100	89.651996
275	2.605376	2	1.421122	True	81.248372	Mitigating	40.0	Moderate Risk	50.0	0	42.812093
298	2.104797	4	0.482920	True	85.518612	Imminent warning	60.0	Moderate Risk	50.0	0	42.812093
304	5.995191	1	0.722720	True	76.384863	Overriding	20.0	Moderate Risk	50.0	0	36.596216
292	6.115681	0	2.193195	False	58.850949	Overriding	20.0	Low Risk	75.0	100	63.462737
301	5.147612	0	1.296710	False	66.958202	Mitigating	40.0	Low Risk	75.0	100	70.489551
296	6.724000	2	1.565617	True	67.976608	Overriding	20.0	Low Risk	100.0	0	46.994152
308	3.464700	3	0.604276	True	82.043949	Overriding	20.0	Moderate Risk	50.0	0	38.010987
307	6.744621	2	2.787930	False	50.845979	Overriding	20.0	Low Risk	75.0	100	61.461495
303	5.678654	0	2.032254	False	61.101238	Imminent warning	60.0	Low Risk	75.0	100	74.025309
286	8.244584	6	1.180152	True	60.280928	Imminent warning	60.0	Low Risk	100.0	0	55.070232
309	3.065808	4	1.400147	False	67.312725	Mitigating	40.0	Moderate Risk	50.0	0	39.328181
297	2.959398	2	1.010411	True	82.554451	Overriding	20.0	Moderate Risk	50.0	0	38.138613
675	11.225968	1	2.810179	True	48.519227	Overriding	20.0	Low Risk	100.0	0	42.129807

Table 5.1: Evaluation Results from the formulated Safety Scoring Method

6

Conclusions

Drivers perceive their actions based on the road traffic situations, hence evaluating their driving style using a single criterion or method is difficult. In order to overcome this difficulty, four evaluation methods were incorporated to intuitively get an Intentional and a Contextual Score combined together forming the Safety Score. By using time-series data of longitudinal accelerations, we have tried to indicate and reflect the dynamic changes in the road traffic situation and how the driver operated the truck during those situations.

1. **Rapid Deceleration Pattern**:
   - From this analysis, we find the best way to decelerate during an event. This is captured by LBG clustering where Moderate Pattern reflects the safest way to brake.
   - Using the mean of the safest pattern, Magnitude of Deviation, Consistency and Braking Intensity were formulated to obtain a score for each segment.
2. **Driving Risks**:
   - By clustering the feature X using K-Means, driving risks were classified to understand the potential threat of near crashing.
3. **TTC warnings**:
   - By estimating the time remaining for the truck to collide with the leading vehicle, a classification of different warning stages as in Figure 1.1 were found to understand the possibility of creating a near crash situation.
4. **Object Detection**:
   - By detecting the brake lights of the leading vehicle being on, a classification of Braking to Normal is done to find if the pre-brake or full brake event occurred because of the ego vehicle driver being inattentive during a near crash event or not.

These individual methods were quantified using different metrics and were aggregated to a weighted average to formulate the final score. The weighted averages reflected that most of the RIDs have good safety scores, putting them in a smooth and efficient driving style.

Drivers with low scorers are recommended to undergo training programs focused on improving their driving habits, along with coaching designed to help them achieve smoother and more efficient driving.

7
Future Work

The safety scoring system based on driver behavior and braking patterns shows great promise. However, there are many opportunities for further enhancements and refinement. Currently, our focus is solely on data from the CW-EB system. Future improvements could include **additional ADAS data**, such as Lane Keeping System, which provide insights into lane discipline and potential distractions, and Driver Assistance System, which monitor driver drowsiness and inattention to assess overall alertness. Along with **data being collected from testing units** in an experimental setup can also improve the accuracy and reliability of the data.

In this study, we used the LBG algorithm and K-means clustering for pattern recognition. However, as we incorporate new data sources, it may be beneficial to implement more advanced machine learning techniques. For example, **reinforcement learning** could be integrated to allow for dynamic adaptation, enabling the system to continuously learn from incoming data and adjust accordingly to improve accuracy over time.

By creating different ways to analyze the rapid deceleration pattern of drivers could involve examining the complete acceleration profile. This would include assessing the deceleration and acceleration curves separately for all drivers based on the selected pattern and calculating a weighted average score.

Additionally, the accuracy and robustness of the object detection model can be significantly enhanced by incorporating a **larger dataset**, preferably collected from extensive field testing with trucks. This approach will ensure a diverse range of data, which will positively impact the model's training process and improve its reliability. Expanding the dataset will also allow for the inclusion of more detailed behavioral categories, such as gradual deceleration, sudden stops, lane changes, and the presence of obstacles.

Beyond detecting vehicle behavior, there is scope to enhance the object detection model to analyze **weather conditions**, and **other environmental factors**. For instance, recognizing wet or icy roads, and poor visibility conditions could provide a more comprehensive context for the safety scoring system.

By pursuing these avenues, we can develop a more comprehensive and sophisticated safety scoring system that better evaluates and pushes drivers to a safe driving style.

Bibliography

[1] N. S. Council. “National safety council injury facts.” (2024), [Online]. Available: https://injuryfacts.nsc.org/motor-vehicle/road-users/large-trucks/.

[2] Nauto. “What is driver behaviour.” (2023), [Online]. Available: https://www.nauto.com/glossary/what-is-driver-behavior.

[3] Y. Nagaraj. “Lbg-clustering.” (2023), [Online]. Available: https://www.linkedin.com/pulse/lindebuzogray-yeshwanth-n/ (visited on 2023-09-12).

[4] N. Sharma. “Kmeans clustering.” (2024), [Online]. Available: https://neptune.ai/blog/k-means-clustering.

[5] Y.-L. Chen, “An explicit and novel forward collision probability index,” in *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, IEEE, 2016, pp. 1778–1782.

[6] Ultralytics. “Ultralytics yolo documentation.” (2024), [Online]. Available: https://docs.ultralytics.com/.

[7] Y. Zheng, J. Wang, X. Li, C. Yu, K. Kodaka, and K. Li, “Driving risk assessment using cluster analysis based on naturalistic driving data,” in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2014, pp. 2584–2589.

[8] A. Naito, C. Miyajima, T. Nishino, N. Kitaoka, and K. Takeda, “Driver evaluation based on classification of rapid decelerating patterns,” in *2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, IEEE, 2009, pp. 108–112.

[9] M. E. Mumcuoglu, G. Alcan, M. Unel, *et al.*, “Driver evaluation in heavy duty vehicles based on acceleration and braking behaviors,” in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, IEEE, 2020, pp. 447–452.

[10] S. Wang, X. Zhao, Q. Yu, and T. Yuan, “Identification of driver braking intention based on long short-term memory (lstm) network,” *IEEE Access*, vol. 8, pp. 180 422–180 432, 2020.

[11] M. Abbas, B. Higgs, A. Medina, and C. D. Yang, “Identification of warning signs in truck driving behavior before safety-critical events,” in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2011, pp. 558–563.

[12] J. Pirhonen, R. Ojala, K. Kivekäs, J. Vepsäläinen, and K. Tammi, “Brake light detection algorithm for predictive braking,” *Applied Sciences*, vol. 12, no. 6, p. 2804, 2022.

[13] “Drivers guide,” Sweden, Tech. Rep., 2023.

[14] D. Cousineau and S. Chartier, “Outliers detection and treatment: A review.,” *International journal of psychological research*, vol. 3, no. 1, pp. 58–67, 2010.

[15] D. Kindness. “Z-score.” (2024), [Online]. Available: `https://www.investopedia.com/terms/z/zscore.asp`.

[16] J. Frost. “Iqr method.” (2024), [Online]. Available: `https://statisticsbyjim.com/basics/interquartile-range/`.

[17] SaturnCloud. “Vector quantization technique.” (2024), [Online]. Available: `https://saturncloud.io/glossary/vector-quantization/`.

[18] Y. Linde, A. Buzo, and R. Gray, “An algorithm for vector quantizer design,” *IEEE Transactions on communications*, vol. 28, no. 1, pp. 84–95, 1980.

[19] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, “Time-series clustering–a decade review,” *Information systems*, vol. 53, pp. 16–38, 2015.

[20] “Scikit-documentation.” (2024), [Online]. Available: `https://scikit-learn.org/stable/modules/semi_supervised.html`.

[21] A. R. Anil and J. Anudev, “Driver behavior analysis using k-means algorithm,” in *2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT)*, IEEE, 2022, pp. 1555–1559.

[22] Y. Zheng, J. Wang, X. Li, C. Yu, K. Kodaka, and K. Li, “Driving risk assessment using cluster analysis based on naturalistic driving data,” in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2014, pp. 2584–2589.

[23] Y.-L. Chen, “An explicit and novel forward collision probability index,” in *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, IEEE, 2015, pp. 1778–1782.

[24] H. M. Group. “Volvo-the-brake-report.” (2024), [Online]. Available: `https://thebrakereport.com/volvo-trucks-collision-warning-emergency-brake/`.

[25] Y. Zhang, E. K. Antonsson, and K. Grote, “A new threat assessment measure for collision avoidance systems,” in *2006 IEEE Intelligent Transportation Systems Conference*, IEEE, 2006, pp. 968–975.

[26] A. A.I. “How to implement object detection using deep learning: A step-by-step guide.” (2024), [Online]. Available: `https://www.augmentedstartups.com/blog/how-to-implement-object-detection-using-deep-learning-a-step-by-step-guide`.

[27] Encord. “Yolo object detection explained: Evolution, algorithm, and applications.” (2024), [Online]. Available: `https://encord.com/blog/yolo-object-detection-guide/`.

[28] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 7464–7475.

[29] M. R. Gunawan and E. C. Djamal, "Spatio-temporal approach using cnn-rnn in hand gesture recognition," in *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)*, IEEE, 2021, pp. 385–389.

[30] J. Redmon, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[31] DataTechNotes. "Filtering approach." (2024), [Online]. Available: `https://www.datatechnotes.com/2022/05/smoothing-example-with-savitzky-golay.html`.

[32] A. Naito, C. Miyajima, T. Nishino, N. Kitaoka, and K. Takeda, "Driver evaluation based on classification of rapid decelerating patterns," in *2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, IEEE, 2009, pp. 108–112.

[33] H.-K. Hsu, Y.-H. Tsai, X. Mei, *et al.*, "Learning to tell brake and turn signals in videos using CNN-LSTM structure," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2017, pp. 1–6.

[34] "Google colaboratory." (2024), [Online]. Available: `https://colab.research.google.com/`.

[35] andrewm4894. "Synchronizing." (2023), [Online]. Available: `https://andrewm4894.com/2020/09/03/time-series-clustering-with-tslearn/`.

[36] D. Shah. "Mean average precision (mAP) explained: Everything you need to know." (2022), [Online]. Available: `https://www.v7labs.com/blog/mean-average-precision`.

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

AVANCEZ
1829
CHALMERS
UNIVERSITY OF TECHNOLOGY