

Driving Behavior Classification
for Heavy-Duty Vehicles Using LSTM Networks

by

Mehmet Emin Mumcuođlu

Submitted to
the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

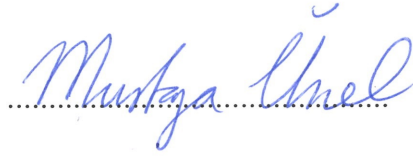
SABANCI UNIVERSITY

July 2019

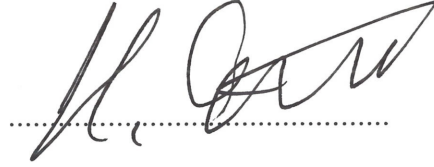
Driving Behavior Classification
for Heavy-Duty Vehicles Using LSTM Networks

APPROVED BY

Prof. Mustafa Ünel
(Thesis Advisor)


.....

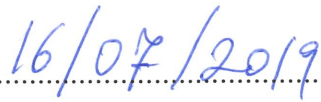
Asst. Prof. Hüseyin Özkan


.....

Asst. Prof. Ertuğrul Çetinsoy


.....

DATE OF APPROVAL:


.....

© Mehmet Emin Mumcuođlu 2019
All Rights Reserved

Driving Behavior Classification
for Heavy-Duty Vehicles Using LSTM Networks

Mehmet Emin Mumcuoğlu

ME, Master's Thesis, 2019

Thesis Advisor: Prof. Dr. Mustafa Ünel

Keywords: Driving Behavior, Driver Classification, Acceleration Behavior, Car Following, Road Design, Heavy-Duty Vehicles, LSTM Classifier

Abstract

Despite growing autonomous driving trend, human is still a major factor in the current vehicle technology. Drivers have a great impact on both fuel economy and accident prevention. Therefore, identification and evaluation of driving behaviors are crucial to improve the performance, safety and energy management of vehicle technologies, particularly for heavy-duty vehicles. In this thesis, several driving behaviors with different acceleration and car following characteristics are generated on a realistic truck model in IPG's TruckMaker simulation environment. A Long Short Term Memory (LSTM) classifier is then utilized to recognize driving behaviors. First, six drivers are defined based on their longitudinal and lateral acceleration limits. The classifier is trained using driving signals acquired from the simulated truck which follows an artificial training road with different trailer loads. The training road is designed to cover possible road curves that can be seen in highways. The model is tested with driving signals that are collected from a realistic road using the same method. Then, three drivers (calm, normal and aggressive) are defined based on their longitudinal acceleration profiles in car following and the classifier is trained and tested using driving signals of these drivers in different traffic scenarios. Results show that the proposed LSTM classifier is capable of successfully capturing the dynamic relations encoded in driving signals and recognizing different driving behaviors in small time samples.

LSTM Ağları Kullanılarak Ağır Vasıtalar için Sürücü Davranışlarının Sınıflandırması

Mehmet Emin Mumcuođlu

ME, Master Tezi, 2019

Tez Danışmanı: Prof. Dr. Mustafa Ünel

Anahtar kelimeler: Sürücü Davranışları, Sürücü Sınıflandırması, İvmelenme Davranışları, Araç Takibi, Yol Tasarımı, Ağır Vasıtalar, LSTM Sınıflandırıcısı

Özet

Araçlarda hızla büyüyen otonom sürüş eğilimine rağmen, insan mevcut araç teknolojileri için hala oldukça önemli bir faktör. Sürücüler, araçların hem yakıt ekonomisi hem de sürüş güvenliği üzerinde oldukça büyük bir etkiye sahip. Bu nedenle, sürücü davranışlarının belirlenmesi ve değerlendirilmesi özellikle ağır vasıta araçlar için mevcut araç teknolojilerinin performans, güvenlik ve verimliliklerinin geliştirilmesi açısından büyük önem taşıyor. Bu tez kapsamında, IPG'nin Truckmaker simülasyon ortamında gerçekçi bir araç modeli kullanılarak, ivmelenme ve araç takip karakteristikleri üzerinden çeşitli sürücü davranışları tasarlandı. Bu sürücü davranışlarının tanımlanması amacıyla bir Uzun Kısa Vadeli Hafıza Ağları (LSTM) sınıflandırıcısından yararlanıldı. İlk olarak, doğrusal, yanal ivmelenme limitleri üzerinden altı farklı sürücü tanımlandı. Sınıflandırıcı, sürücülerin araç modelinin 5 farklı yük ile yapay bir yolu takibinden toplanan sürüş sinyalleri ile eğitildi. Bahsedilen yol otoyollarda karşılaşılabilecek geometrileri kapsayacak şekilde tasarlandı. Model, gerçekçi bir yol üzerinden aynı yöntemle toplanan veriler ile test edildi. Sonrasında, araç takibindeki doğrusal ivmelenmeleri üzerinden üç sürücü (sakin, normal, agresif) tanımlandı ve bu sürücülerin farklı trafik senaryolarındaki sürüş sinyalleri ile sınıflandırıcının eğitimi ve testi gerçekleştirildi. Sınıflandırma sonuçları kullanılan LSTM tabanlı yapının, farklı sürüş davranışlarını içeren sürüş sinyallerinin dinamik ilişkilerini başarı ile yakalayabildiğini göstermiştir.

Acknowledgements

I would like to express my sincere gratitude and deep appreciation to my thesis advisor Prof. Dr. Mustafa Unel, for his invaluable academic and personal guidance, endless support and motivation throughout my Master study at Sabancı University. His unique vision, sincerity, and passion for research have greatly inspired me.

I would like to thank Asst. Prof. Huseyin Ozkan and Asst. Prof. Ertugrul Cetinsoy for their interest in my work and giving their valuable time to serve as my jurors.

I am thankful for financial and scientific support provided by Ford OTOSAN with the great collaboration of Product Development Team through the “Driver Evaluation” Project. Many thanks to Kerem Koprubasi, Metin Yilmaz, Mehmet Mutluergil and Onur Cicek.

I would especially like to thank Ph.D. candidate Gokhan Alcan for sharing his experience and knowledge as a colleague in “Driver Evaluation” Project and as a friend. I would also like to thank every member of Control, Vision, and Robotics Laboratory. Many thanks to Diyar Khalis Bilal, Hammad Zaki, Naida Fetic, Emre Yilmaz, Fatih Emre Tosun, Ugur Mengilli, Ayhan Aktaş, Umut Çalışkan and Zeynep Özge Orhan for their excellent help, sharing, and friendship during my graduate study.

Finally, I would like to thank my parents, İsmail and Seçil, dear brothers, Ali Haydar and Hasan Huseyin and my lovely sister Fatıma Zehra for their endless love and support.

Contents

Abstract	iii
Özet	iv
Acknowledgements	v
Contents	vi
List of Figures	ix
List of Tables	xii
1 Introduction	1
1.1 Motivation	2
1.2 Contributions of the Thesis	3
1.3 Outline of the Thesis	4
1.4 Publications	4
2 Literature Survey and Background	5
2.1 Driving Behavior Classification Methods	5

2.1.1	Driving Signals	5
2.1.2	Direct Methods	6
2.1.3	Indirect (Model-Based) Methods	7
2.2	Recurrent Neural Networks for Sequence Classification	7
2.2.1	Neural Networks	8
2.2.2	Recurrent Neural Networks	9
2.2.3	Long Short Term Memory Networks	11
3	Driving Behavior Analysis of a Truck Driver	13
3.1	Visual Investigation of the Vehicle Signals	13
3.2	Histogram Map Visualization of the Vehicle Signals	16
4	Data Generation	24
4.1	TruckMaker	25
4.2	Generation of a Simulation Truck Model	25
4.2.1	Physical Parameters	25
4.2.2	Powertrain	27
5	Acceleration Model	29
5.1	Experiment Design	29
5.1.1	Driver Design	29
5.1.2	Training and Test Road Design	32
5.2	Driver Classification	36
5.2.1	Data Generation for Acceleration Model	36
5.2.2	Long Short Term Memory Network	37

5.3	Results	39
5.3.1	Driver Comparison	39
5.3.2	Classification Results	43
6	Car Following Model	45
6.1	Experiment Design	46
6.1.1	Driver Design	46
6.2	Driver Classification	49
6.2.1	Data Generation for Car Following Model	49
6.3	Results	51
6.3.1	Driver Comparison	51
6.3.2	Classification Results	53
7	Conclusion and Future Works	54
	Bibliography	56

List of Figures

1.1	Fuel economy percentage for a set of bus drivers based on a reference consumption value [1]	2
2.1	Signal sources for driving style recognition algorithms.	6
2.2	An artificial neuron (left) and feedforward neural network (right).	8
2.3	Recurrent structure of the RNNs.	9
2.4	Input output mappings of RNNs	10
2.5	Vanishing gradient problem of Recurrent Neural Networks.	11
3.1	GPS (a), altitude (b) graphs of the example driving sequence.	14
3.2	Driving signals of the example driving sequence.	14
3.3	Driving signals of the example driving sequence.	15
3.4	GPS (a), altitude (b) graphs of Parts A, B, C of the example driving sequence.	16
3.5	Accelerator pedal position (%) vs engine speed (rpm) maps.	17
3.6	Engine actual percent torque (%) vs engine speed (rpm) maps.	18
3.7	Accelerator pedal position (%) vs vehicle speed (km/h) maps.	19
3.8	Accelerator pedal position (%) vs vehicle pitch angle (%) maps.	20
3.9	Acceleration (G) vs vehicle speed (km/h) maps.	21
3.10	Acceleration (G) maps for brake events.	22

3.11	Acceleration (G) maps for brake + retarder events.	23
4.1	IPG's TruckMaker, CarMaker, BikeMaker.	24
4.2	Vehicle, road, driver, traffic models of TruckMaker.	25
4.3	Physical parameters of the truck (Top) and trailer (Bottom).	26
4.4	Powertrain power flow.	27
4.5	Engine net torque (Top) and fuel consumption (Bottom) maps.	28
5.1	TruckMaker's driver model.	30
5.2	Acceleration limit parameters of drivers 1-3 (Top), 4-6 (Bottom).	31
5.3	Acceleration limit parameters of drivers 1-6.	32
5.4	X-Y-Z profile of an example road block.	33
5.5	(Top) Z-Y and (Bottom) Y-X profiles of an example training road segment.	34
5.6	(Top) Z-Y profile and (Bottom) Y-X profile of the training road.	35
5.7	Grade - central angle coverage map for training and test roads.	36
5.8	Classification model inputs.	36
5.9	LSTM structure.	37
5.10	Driver comparison based on vehicle speed, longitudinal and lateral acceleration signals acquired from the training road for drivers in the same set (10 ton trailer load).	40
5.11	Driver comparison based on vehicle speed, longitudinal and lateral acceleration signals acquired from the training road for drivers in a differing class (10 ton trailer load).	41
5.12	Vehicle speed, Longitudinal and lateral acceleration signals of all six drivers from the test road (10 ton trailer load).	42
5.13	Normalized confusion matrix of the training (Top) and the test set (Bottom).	43

5.14	Training and test accuracy graph for correct and differing classes. . .	44
6.1	Car following model outline.	46
6.2	Hysteresis thresholding for the target speed generator.	47
6.3	Mode 1 speed (Top) and acceleration (Bottom) profiles of the drivers 1-3.	49
6.4	Leading vehicle speed and acceleration profiles for the training and test simulations.	50
6.5	Driver comparison based on vehicle speed, throttle position, follow- ing distance and control mode selection signals acquired from the test road.	52
6.6	Normalized confusion matrix of the training (Top) and the test set (Bottom).	53

List of Tables

3.1	Horizontal and vertical road profile of Parts A, B, C of the example driving sequence.	16
5.1	Acceleration behavior model driver parameters	30
6.1	Car following model driver parameters.	48
6.2	Driver comparison based on average speed and fuel consumption . .	51

Chapter 1

Introduction

Vehicles have affected the world in a significant manner. The automotive industry has been the motive power of the economies of numerous countries. However, as automobiles come into human life, many concerns have occurred namely safety problems, excessive fuel consumption, noise and exhaust pollution and dependency on vehicles [2]. With the advance in availability of the technology, safety and efficiency related subjects have gained more attention.

Throughout the years, many systems are developed using proprioceptive and exteroceptive sensors to enhance vehicles for safer, better, and more efficient driving. Not only proprioceptive sensor-based systems such as Anti-lock Braking System (ABS) and Electronic Stability Program (ESP), but also many exteroceptive sensor-based systems such as Adaptive Cruise Control (ACC), Automatic Emergency Braking (AEB), lane keeping assist (LKAS) became standard for most of today's vehicles [3]. These exteroceptive sensor based systems are named Advanced Driver Assistance Systems (ADAS). Researchers aim to improve current ADAS technologies with a better understanding of driving behaviors.

1.1 Motivation

Despite the growing intelligence and autonomy in vehicles, the driver is still a major factor in vehicle technologies. Through years, drivers and driving behaviors have been investigated for various vehicle applications. Reducing fuel consumption and increasing safety are the main motivations in the majority of research.

Drivers have a dramatic effect on fuel efficiency. Especially for heavy-duty vehicles, incorrect driving behaviors create more significant differences due to high weights of these systems [4]. Based on the Liimatainen's [1] proposed fuel economy quantification method, fuel economy variance in a set of bus drivers is almost 30%. Figure 1.1 shows fuel economy percentages of that driver set based on a reference consumption model, which is calculated considering road geometry, vehicle type, and time of the day. Guo et al. [5] claimed that identifying and correcting undesired driving behaviors using driving tips reduced fuel consumption by 25% of a commercial truck with 25 tons of gross combined weight (GCW). These two studies show the importance of understanding driver factors to improve fuel efficiency.

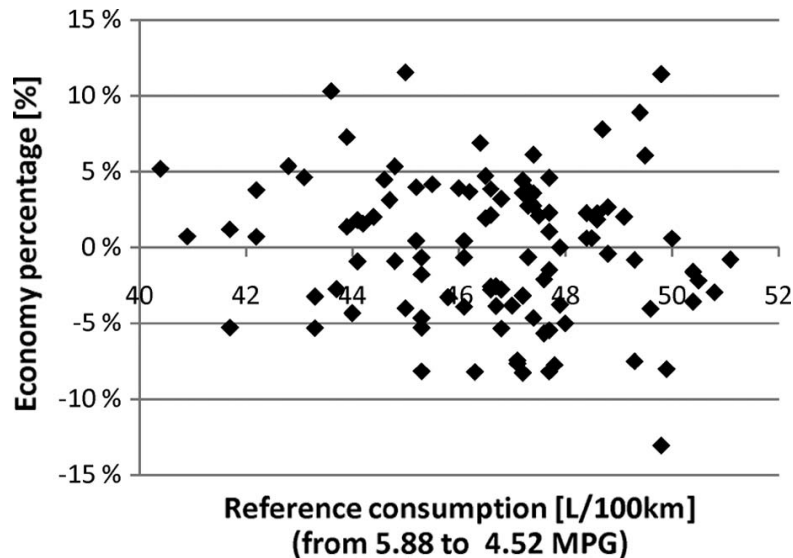


FIGURE 1.1: Fuel economy percentage for a set of bus drivers based on a reference consumption value [1]

Investigation of driving behaviors and a driver's physical state are also essential for road safety, considering 90% of the traffic accidents occur due to human errors [6]. Researchers proposed several methods that analyze vehicle signals and visual features of drivers to identify dangerous driving states such as fatigue, drunk, drowsy [7–11]. In the last three decades, ADAS has become a prevalent research field that aims to enhance road safety by assisting drivers during risky conditions. Developing such systems based on various driving behaviors rather than a generic driver type can increase the success of these systems on preventing accidents [12, 13].

1.2 Contributions of the Thesis

In this thesis, a dynamic classification algorithm based on Long Short Term Memory (LSTM) networks is designed for the identification of driving behaviors of a truck driver. Acceleration and car following models are proposed using this LSTM classifier with the purpose of driver evaluation. Inertial Measurement Unit (IMU) and onboard sensor measurements are used in a time window to train and test the models. All the vehicle signals are obtained from IPG's TruckMaker simulation software using a realistic model of a commercial truck. In the process of generating driver models, the investigation of the driving signals of a real truck driver is utilized. This investigation is performed using direct signal analysis and histogram analysis methods.

For the acceleration model, six drivers are defined with setting longitudinal and lateral acceleration limits and the maximum speed parameters. These drivers are simulated on a unique training road and a realistic test road controlling the truck model with varying trailer loads. With the proposed training road design, it is aimed to cover possible horizontal and vertical curves that can be observed in a highway. The classifier is trained with driving signals, specifically longitudinal and lateral accelerations, engine and vehicle speeds, and the road geometry. For car following model, three drivers are designed with particular aggression levels,

based on acceleration and deceleration characteristics in traffic. The classifier is trained with the inputs of vehicle signals, road geometry, and sensor signals, that are simulated for different traffic profiles.

1.3 Outline of the Thesis

Chapter 2 presents a literature review in driving behavior classification methodologies and an overview of Artificial Neural Networks (ANNs), Recurrent Neural Networks (RNNs), and Long Short Term Memory (LSTM) are introduced. Chapter 3 is on the analysis of an sample driving sequence of a real truck driver. Additionally, the histogram map based representation of driving signals are presented. Chapter 4 introduces the generation of driving data using IPG's Truckmaker and the modeling of a commercial truck for simulations. In Chapter 5, generation of the acceleration model and the classification methodology are explained. Simulation outputs and the classification results are provided for this model. In Chapter 6 car following model is explained with simulation outputs and classification results. Finally, conclusion of this thesis and possible future works are discussed in Chapter 7.

1.4 Publications

- M.E. Mumcuoglu, G. Alcan, M. Unel, O. Cicek, M. Mutluergil, M. Yilmaz, K. Koprubasi, "Driving Behavior Classification Using Long Short Term Memory Networks," 2019 International Conference of Electrical and Electronic Technologies for Automotive, Torino, IEEE, 2019

Chapter 2

Literature Survey and Background

Chapter 2 reviews the proposed methodologies for driving behavior classification in literature. Furthermore, a background on sequence classification using Recurrent Neural Networks is presented.

2.1 Driving Behavior Classification Methods

This section highlights the various valuable inputs for the driving behavior classification and presents the direct and indirect methods in the literature.

2.1.1 Driving Signals

Different vehicle signals, captured from various sources, are used for driving style recognition algorithms [14]. Longitudinal and angular velocities, accelerator and brake pedal positions, accelerations, steering wheel angle and fuel consumption are some of the well-known inputs for these algorithms [15–21]. Vehicle’s onboard IMU [15], GPS [19], radar or LiDAR [21] sensors are the main source for most

of these signals. Using smartphone sensors is also an alternative and low-cost solution for capturing these data [16–18].

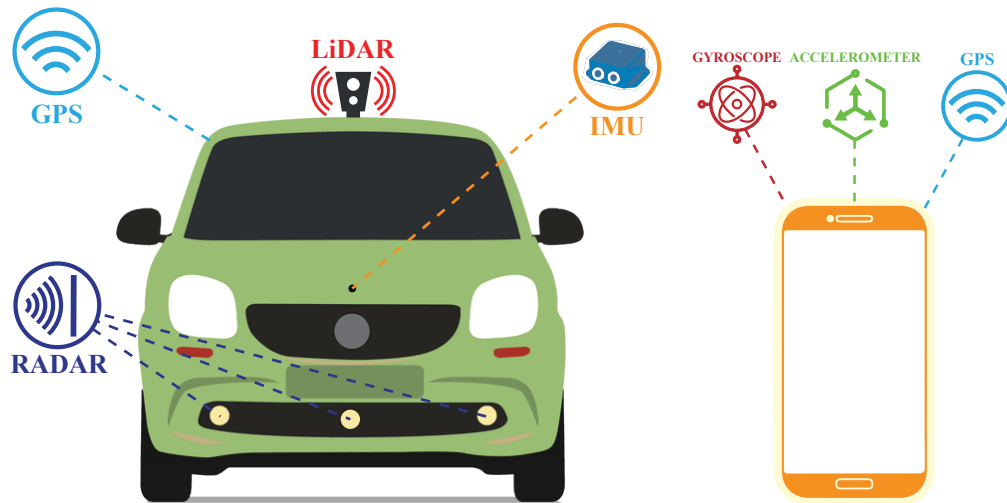


FIGURE 2.1: Signal sources for driving style recognition algorithms.

Different signals or signal combinations are more capable of distinguishing different driving behaviors. One of the approaches is identifying driving behaviors based on the correlation between acceleration and the aggression [18] or speed profiles and fuel consumption [19]. Another popular method is using the combination of acceleration with vehicle speed [20]. Additionally, using longitudinal and lateral accelerations together is also emphasized with their strength of capturing the majority of driving behaviors [21].

2.1.2 Direct Methods

Driving characteristics recognition algorithms can be grouped into two approaches, that are direct and indirect (model based) methods [22]. Direct methods directly investigate driving signals, which are mentioned in the previous part, mostly using pattern recognition or data analysis methods, while indirect methods require relevant driver models to extract and identify driving behaviors.

Zhang et al. [23] suggested a technique for driving skill characterization using three pattern recognition methods, which are decision tree, multilayer perceptron-artificial neural networks (MLP-ANN), and support vector machines (SVMs). Higgs and Abbas [24] examined relationships between driving behaviors and driver's state based on clustering with eight state-action variables, such as longitudinal and lateral accelerations, vehicle speed, and yaw angle. They segmented drivers into different patterns for each cluster.

2.1.3 Indirect (Model-Based) Methods

A relevant driver model needs to be established for indirect (model-based) techniques. This model should represent some of the fundamental driving behaviors of real drivers, such as acceleration, car following, lane change, and braking. Several stochastic process theories are utilized for driving behavior recognition problems. Hidden Markov Model (HMM), which is the simplest form of dynamic Bayesian networks, is commonly used to recognize driving behaviors [26, 27], and driver states [25]. Gadepally et al. [26] considered vehicles as hybrid-state systems, and used HMM to predict driver decisions in road intersections. ARX models are also suitable for capturing dynamic behaviors. Sundbom et al. [28] proposed a method to predict driving behaviors and classify them as aggressive or normal driving using a probabilistic ARX model. Akita et al. [29] and Sekizawa et al. [30] suggested stochastic switched-ARX (SS-ARX) to model the uncertainties of driving behaviors.

2.2 Recurrent Neural Networks for Sequence Classification

This section presents a background for the Recurrent Neural Networks with their strengths and challenges and a brief introduction to Long Short Term Memory networks for sequence classification.

2.2.1 Neural Networks

Larger supply of data and advances in technology in the recent years have brought a remarkable interest in data science and machine learning. Data science techniques are rapidly implemented in industry and its various applications have allowed it to reach an enormous practical success. Nowadays, most learning models are developed based on artificial neural networks (ANNs). With the ability to learn without handmade rules, ANNs are adequate solutions for both supervised and unsupervised machine learning problems [31].

A group of artificial neurons, named nodes, and links between these neurons like synapses form an artificial neural network. In each neuron, an output value is calculated with implementing an activation function to the weighted sum of its input values. Sigmoid, tanh and rectified linear unit (ReLU) functions are the most common activation functions. The activation function of output nodes is determined based on the task. For a classification problem with K number of classes, a softmax function is utilized in the output layer of K nodes. A sample feedforward NN structure with a sigmoid activation function is presented in Figure 2.2.

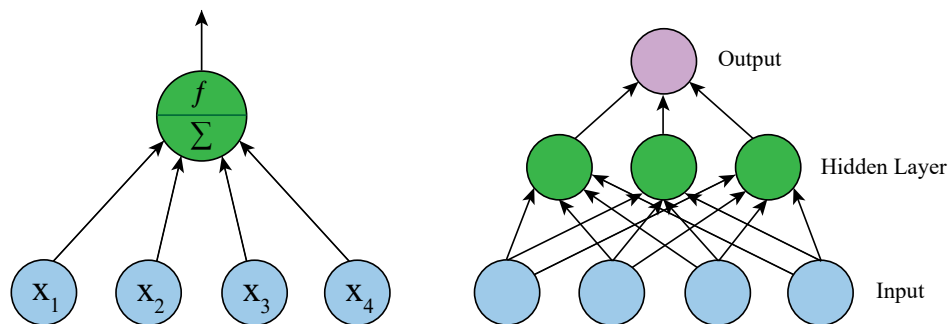


FIGURE 2.2: An artificial neuron (left) and feedforward neural network (right).

The majority of Neural Networks transfer data only from input layer to output layer, called feedforward NNs. This structure brings some limitations due to the assumption of independence between input and output layers. The feedforward NNs cannot meet the expectations when relations between samples exist as in

sequential data. Developing these models to learn sequences is very important, especially for the tasks where ANNs provide proper results.

2.2.2 Recurrent Neural Networks

Unlike ANNs, Recurrent Neural Networks are developed for sequence learning by using the feedback. RNN cells have multidimensional hidden states with non-linear dynamics [32]. States of each RNN cell are transferred to the next cell after the sample is included in the calculations. These hidden states operate as memory for the network and provide the ability to process the previous signals [33]. An RNN can map an input and output sequence at a current timestep, to predict the sequence in the next timestep.

An RNN is a supervised learning model, that consists of artificial neurons with one or more feedback loops [34]. These loops form recurrent cycles over sequence or time [35], as shown in Figure 2.3. To train an RNN in a supervised way, a dataset is needed with inputs and targets. The loss is calculated between outputs and targets. The aim is to minimize the loss with the optimization of the weights of the RNN.

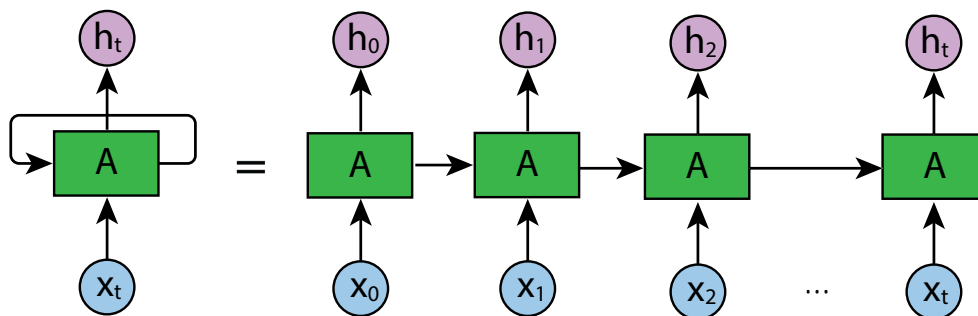


FIGURE 2.3: Recurrent structure of the RNNs.

The basic RNN structure has three layers, that are an input layer with N units, a recurrent hidden layer with M units and an output layer with P units. The inputs to input layer are a set of vectors through time t where $x_t = (x_1, x_2, \dots, x_N)$. The connection between the input and the hidden units are calculated with a weight

matrix of W_{IH} . The hidden units are connected to each other through time where $h_t = (h_1, h_2, \dots, h_M)$. The hidden state, which is the memory of the network, is defined as

$$h_t = f_H(w_{IH}x_t + w_{HH}h_{t-1} + b_h) \quad (2.1)$$

where $f_H()$ is the activation function, and b_H is the bias vector of the hidden layer. The connection between hidden and output units are computed with a weight matrix of W_{HO} . Finally the output layer $y_t = (y_1, y_2, \dots, y_P)$ is calculated as follows

$$y_t = f_o(w_{HO}h_t + b_o) \quad (2.2)$$

where $f_o()$ is the activation function, and b_o is the bias vector of the output units. The calculations above are repeated over time for the sequential input-target pairs. Some of the most popular activation functions are the sigmoid, tanh, and ReLU for hidden layer activation function. For classification of the sequential data, the softmax function is used as an activation function for the output layer.

As mentioned earlier, input (N) , hidden (M) and output (P) layer size of RNNs can be different. Many applications have been utilized with the different mappings between input and output layers such as one to one, many to one, one to many and many to many which are shown in Figure 2.4.

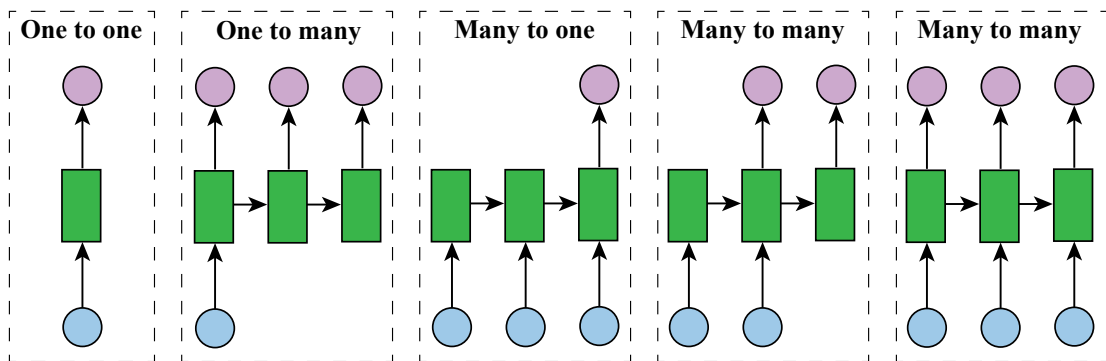


FIGURE 2.4: Input output mappings of RNNs

One to one mapping is the vanilla mode of processing from fixed-size input to fixed-size output as in image classification task. One to many mapping is the processing when a single or fixed-size input gives a sequence of outputs such as

image captioning takes a picture and outputs a sentence of words. Many to one is the mapping when a single output is generated for a sequence of input data such as in proposed driving behavior classification problem. Finally, many to many mapping is applied for sequential inputs and outputs are needed.

2.2.3 Long Short Term Memory Networks

The main advantage of RNNs over feedforward NNs is the ability to learn sequential data due to their memorization capability. However, the amount of memory that can be used is limited for a standard RNN structure. The effect of an input sample on the hidden layer decreases exponentially as new samples are processed, which is called vanishing gradient problem [36–38]). Because of this problem, RNNs cannot effectively learn a task when there is a distance of more than ten timesteps between a relevant input and the target [37]. The vanishing gradient problem is illustrated in Figure 2.5.

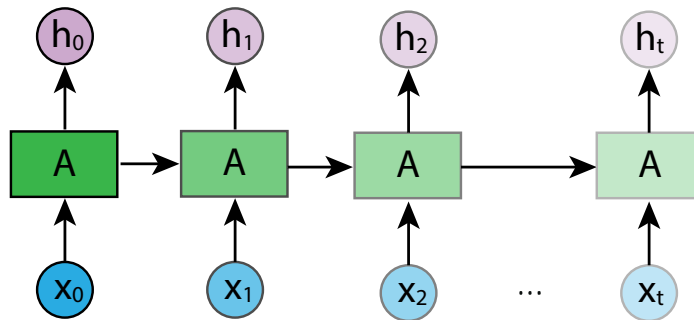


FIGURE 2.5: Vanishing gradient problem of Recurrent Neural Networks.

Many efforts were made to tackle the vanishing gradients of RNNs. These included using alternative algorithms to gradient descent e.g., simulated annealing and discrete error propagation ([38]), hidden states with different time delays ([39–41]) or time constants ([42]), hierarchical history compression ([43]). LSTM is the best-known method, particularly designed to handle the vanishing gradient problem.

With LSTM networks, the structure of hidden units are modified as memory cells with controlling the inputs and outputs of these units using gates. Gates of an LSTM cell control the flow of the data to maintain the relevant memory obtained from previous timesteps [44, 45]. However, the states of LSTM networks can grow independently for continual sequences, and the network can not detect when a memory becomes irrelevant. To solve this problem, the forget gate is proposed by Gers et al. ([46]) that allows LSTM cells to learn to reset the memory at proper times. An LSTM network using this structure can remember both long term and short term memory at the same time.

With the aim of classification of different driving behaviors from driving signals, an LSTM network with a many to one mapping between input and output sequences is utilized. The used structure proposed by Hochreiter et al. [45] will be presented in Chapter 5.

Chapter 3

Driving Behavior Analysis of a Truck Driver

In this chapter, the investigation of driving signals of a real driver controlling heavy-duty vehicle is presented using direct signal analysis and histogram analysis methods. This investigation aims to build a basis for designing driver and vehicle models that can represent the driving behaviors of real truck drivers.

3.1 Visual Investigation of the Vehicle Signals

The provided data is 11 minutes of a driving sequence that is collected from the truck, with a total mass of 22 tons, that is modeled in Chapter 4. The initial purpose of this data is to test the truck on a road that is more challenging than a typical highway or freeway. However, with the changing geometry of the route, dataset allows us to observe both some extremes and variance in driving signals for our driving behavior analysis.

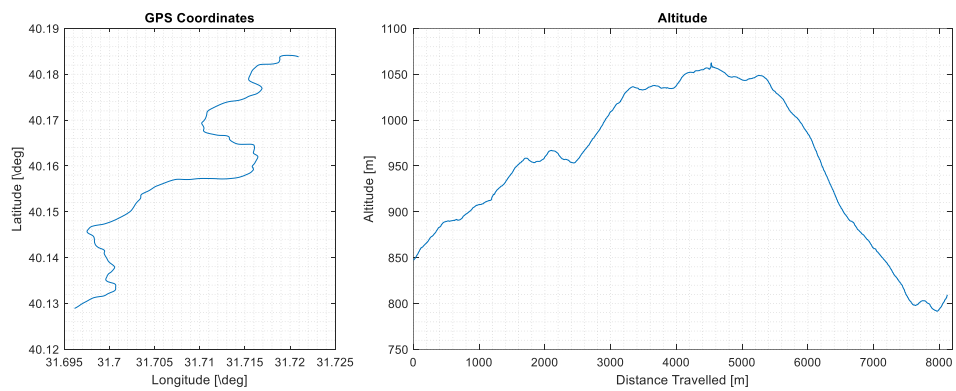


FIGURE 3.1: GPS (a), altitude (b) graphs of the example driving sequence.

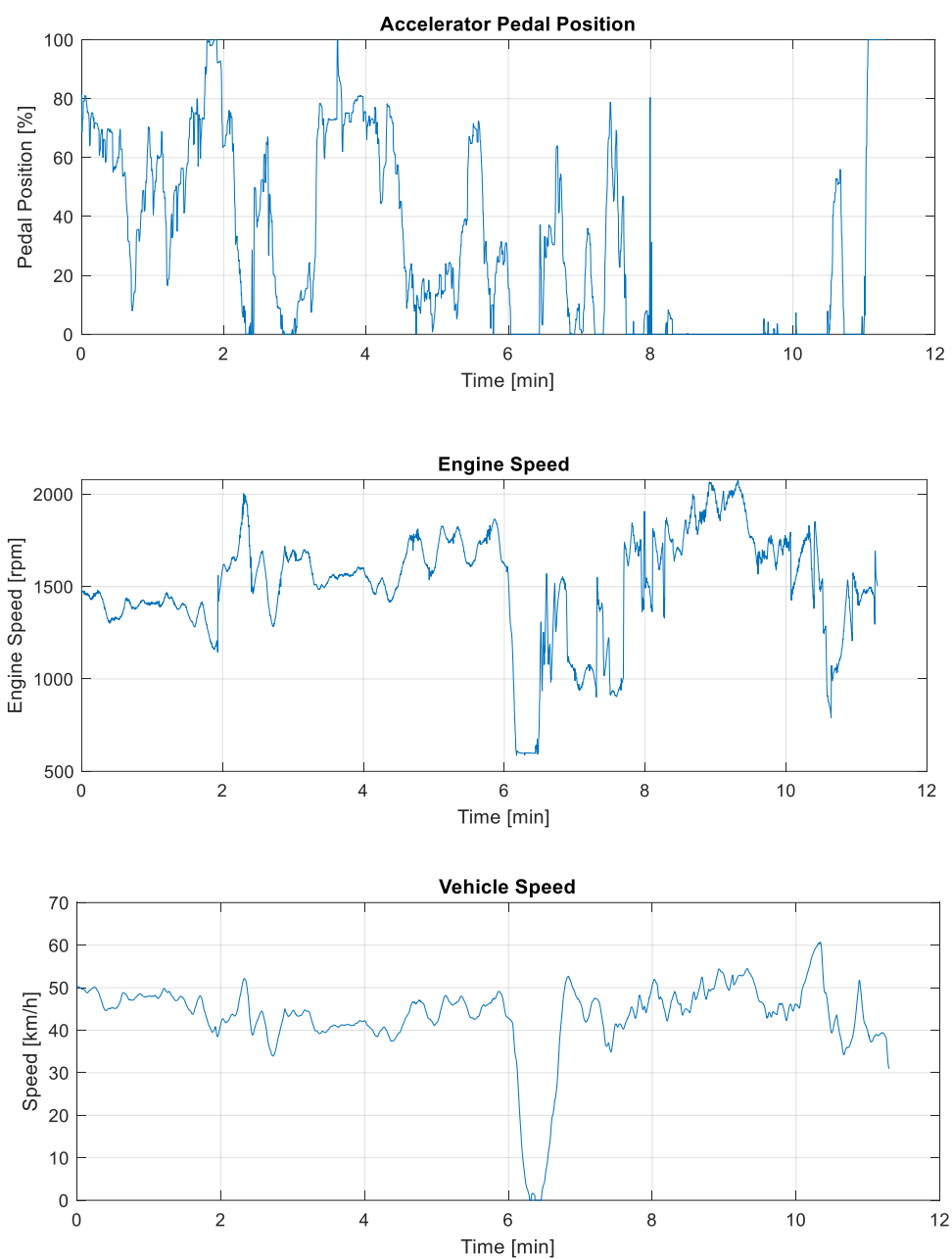


FIGURE 3.2: Driving signals of the example driving sequence.

Figure 3.2) shows the complexity of the horizontal and vertical geometry of the road with GPS and altitude signals. As a result of the vertical profile of the road, the vehicle is operated either with high throttle positions or with no throttle inputs in a significant portion of the data. However, even for these road conditions, the driver rarely used full throttle to accelerate the vehicle. Additionally, the driver affords to keep the vehicle end engine speeds steady especially during uphill sections.

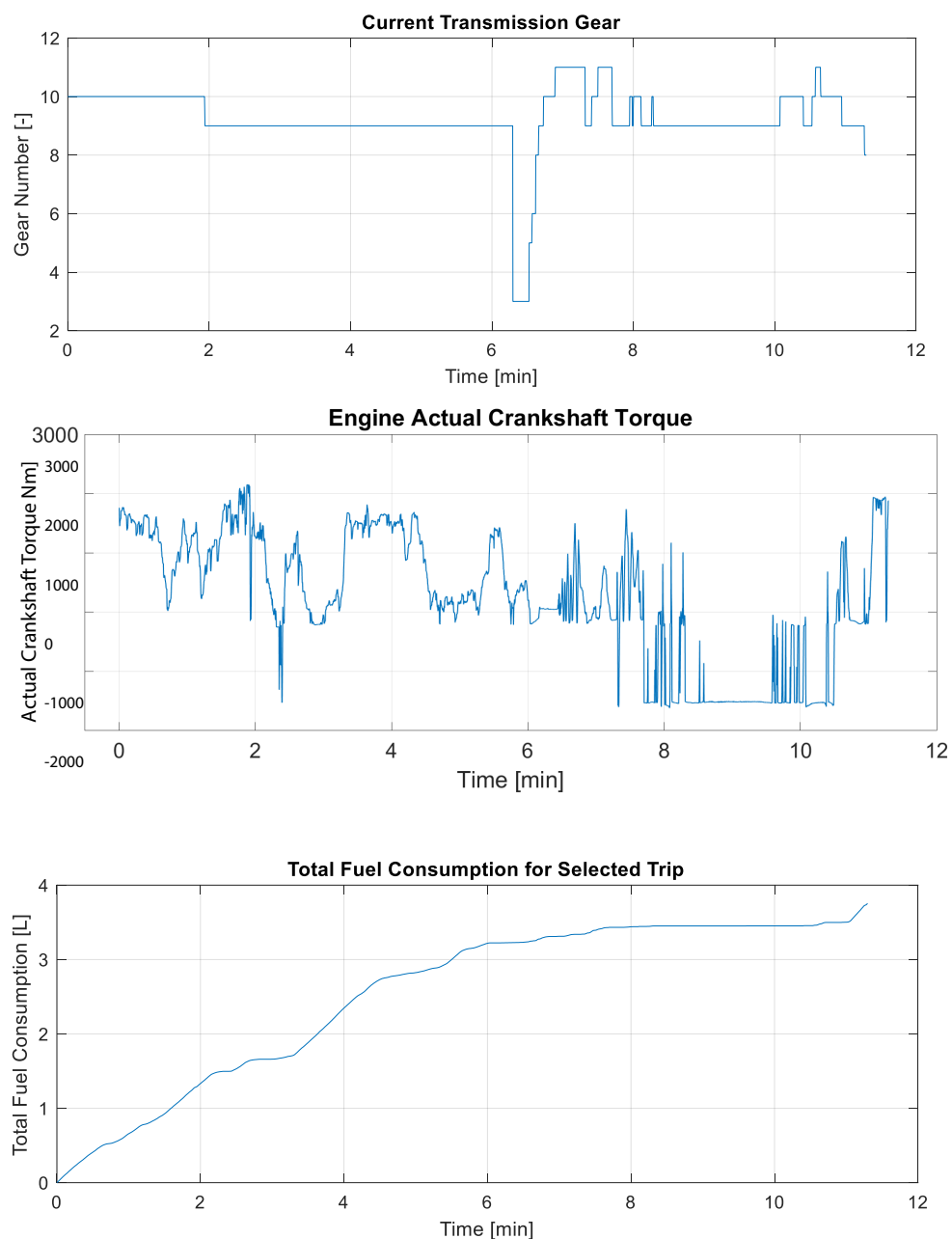


FIGURE 3.3: Driving signals of the example driving sequence.

3.2 Histogram Map Visualization of the Vehicle Signals

Histogram map of data highlights the density of the underlying patterns of signals. For our case, it is used to present different driving behaviors of the truck driver from the driving data, which are time-series signals of on-board vehicle sensors. [5, 47] showed that throttle position - engine speed and engine percent torque - engine speed maps provide a great representation of driver's performance. Seven histogram maps, including these two, are selected based on the relations between the signals and driving behaviors.

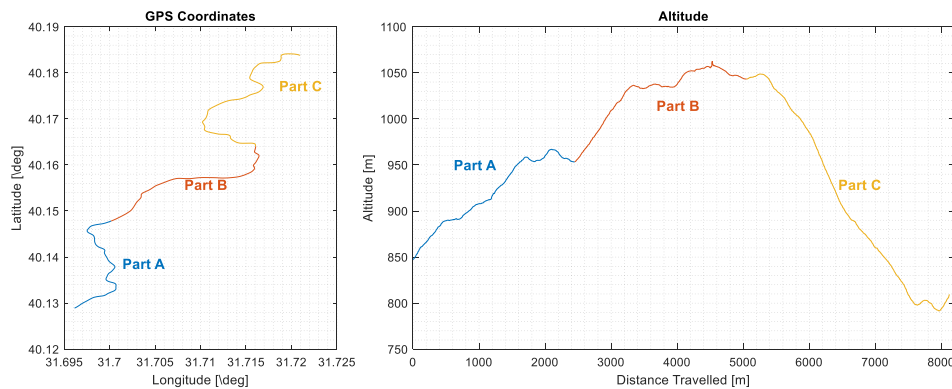


FIGURE 3.4: GPS (a), altitude (b) graphs of Parts A, B, C of the example driving sequence.

Existing driving data is divided into three 3.4 to analyze the changes in driving behaviors based on longitudinal and lateral geometry of the road. Selected maps for defined parts are presented in Figures 3.5 - 3.11) in which the driving behaviors, observed for longer time or more frequently, are represented with brighter colors.

	Part A	Part B	Part C
Slope Profile	Uphill	Uphill	Downhill
Avr. Grade (%)	5	3.8	-8.9
Curvature Profile	Curvy	Less Curvy	Curvy
Avr. Central Angle of Left or Right Turns ($^{\circ}$)	36.5	22.1	45.16

TABLE 3.1: Horizontal and vertical road profile of Parts A, B, C of the example driving sequence.

Some patterns are occurred in the upper portions of throttle - engine speed (TES) maps of Part A and B due to the uphill profiles, while the bins are placed densely at the bottom of the map of Part C due to the downhill road profile (Figure 3.5).

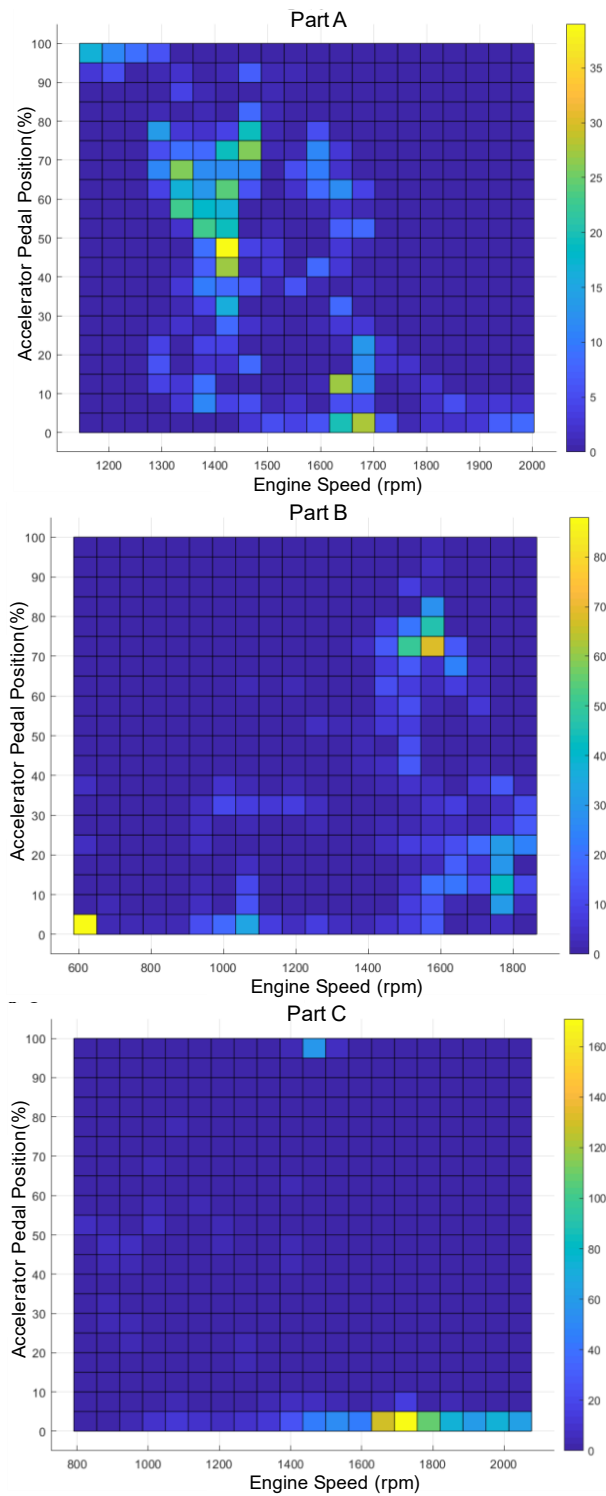


FIGURE 3.5: Accelerator pedal position (%) vs engine speed (rpm) maps.

Additionally, a more dispersed pattern in TES map of Part A is observed due to the higher curvature profile of the Part A. Similar observations can be made in engine torque - engine speed maps ((Figure 3.6))

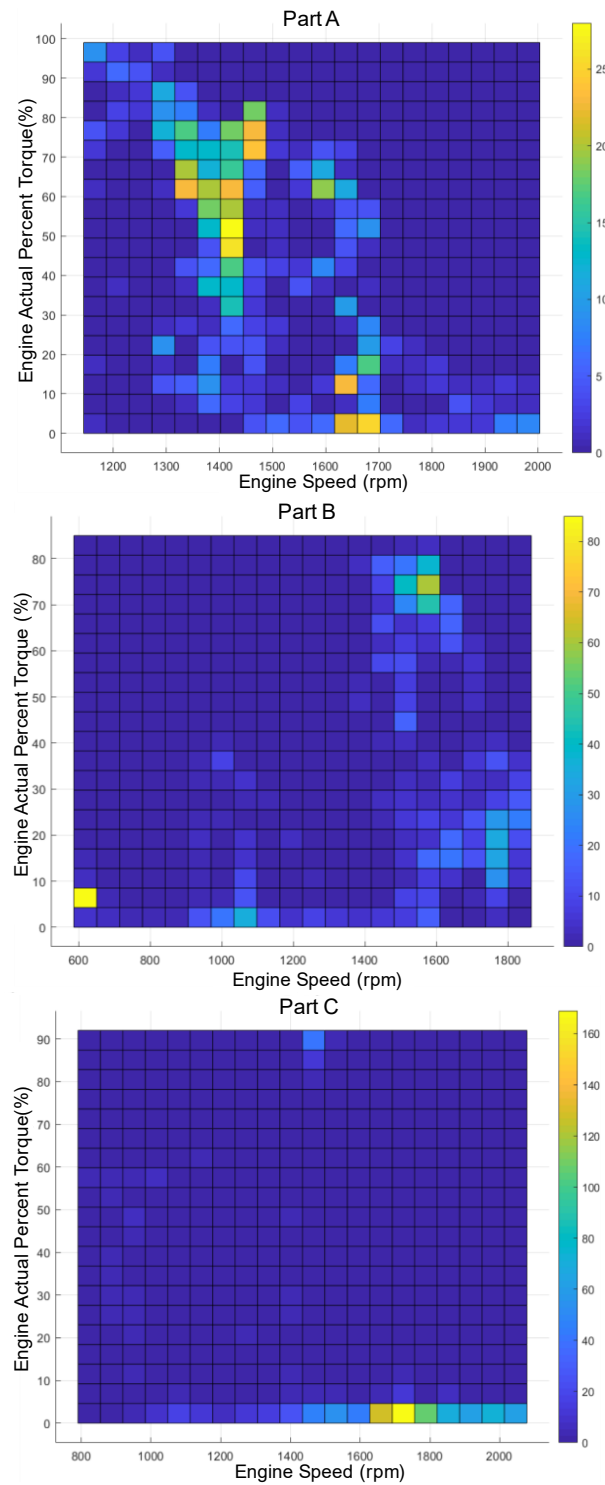


FIGURE 3.6: Engine actual percent torque (%) vs engine speed (rpm) maps.

In Part B driver was able to maintain narrower vehicle speed and engine speed band due to the lower curvature, which can be labeled as a better driving.

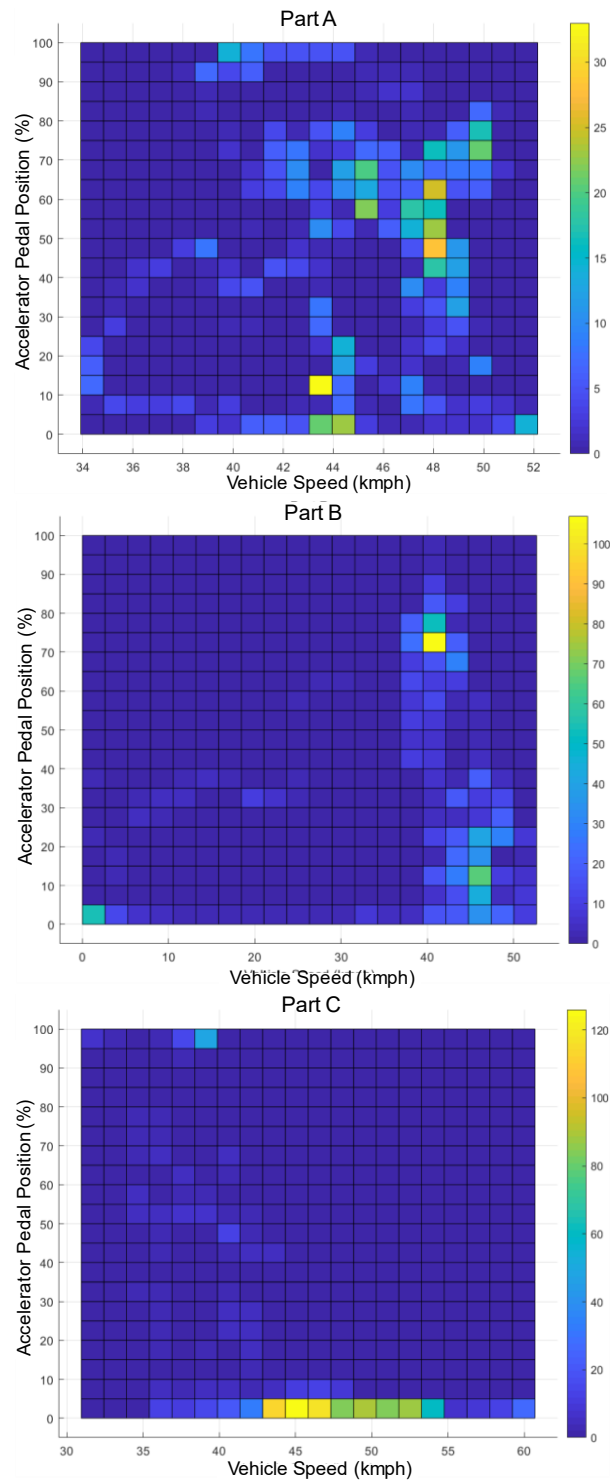


FIGURE 3.7: Accelerator pedal position (%) vs vehicle speed (km/h) maps.

The linear pattern in throttle - pitch angle maps can be observed easily from Figure 3.8 due to the direct relation between slope and torque/throttle requirements. Throttle patterns in negative slopes of Part B are can be undesirable but possible in some of the road conditions.

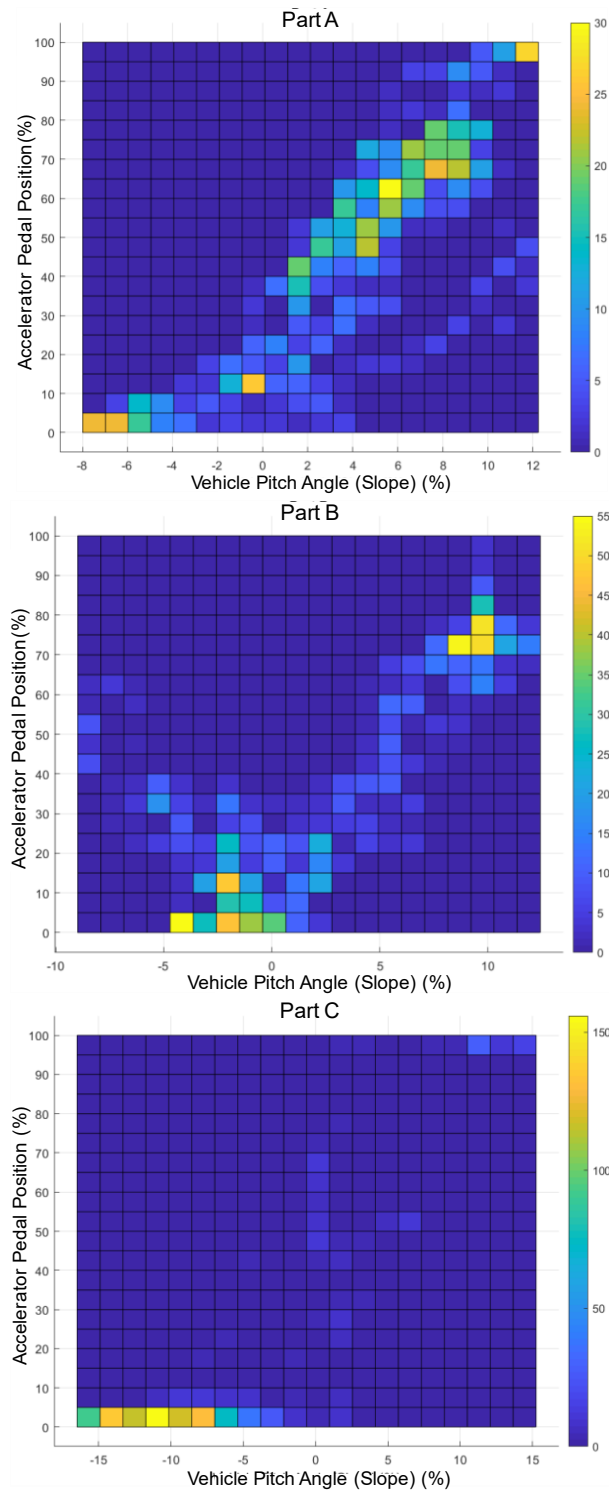


FIGURE 3.8: Accelerator pedal position (%) vs vehicle pitch angle (%) maps.

Acceleration signal of the driver, created with the derivative of the vehicle speed signal, provides essential information about driving characteristics. Figure 3.9 shows that, our expert driver prefers to control the truck with smaller accelerations in $+0.05g$, $-0.15g$ longitudinal acceleration range for this difficult test road.

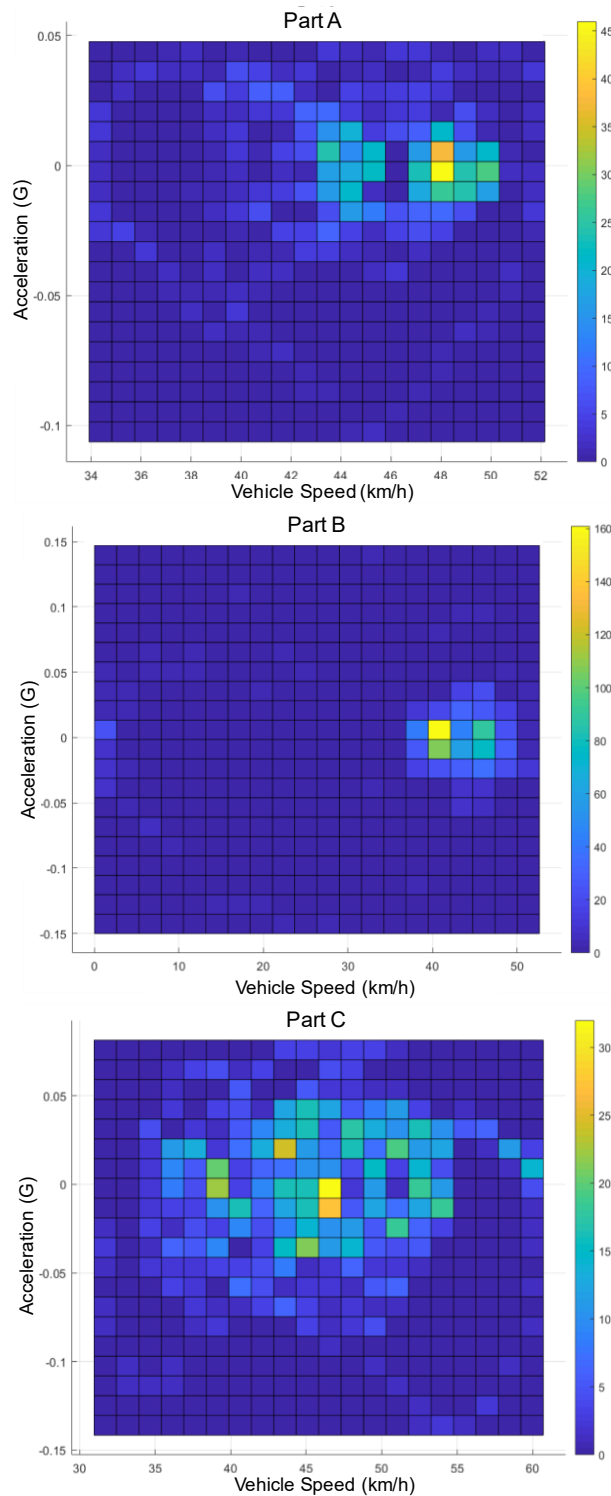


FIGURE 3.9: Acceleration (G) vs vehicle speed (km/h) maps.

Finally, braking behaviors of the driver are analyzed in Figure 3.10, 3.11 with acceleration histograms during regular braking and regular + retarder braking events for the entire data. Retarder is a specific braking system for heavy-duty vehicles to prevent damages in regular brake disks.

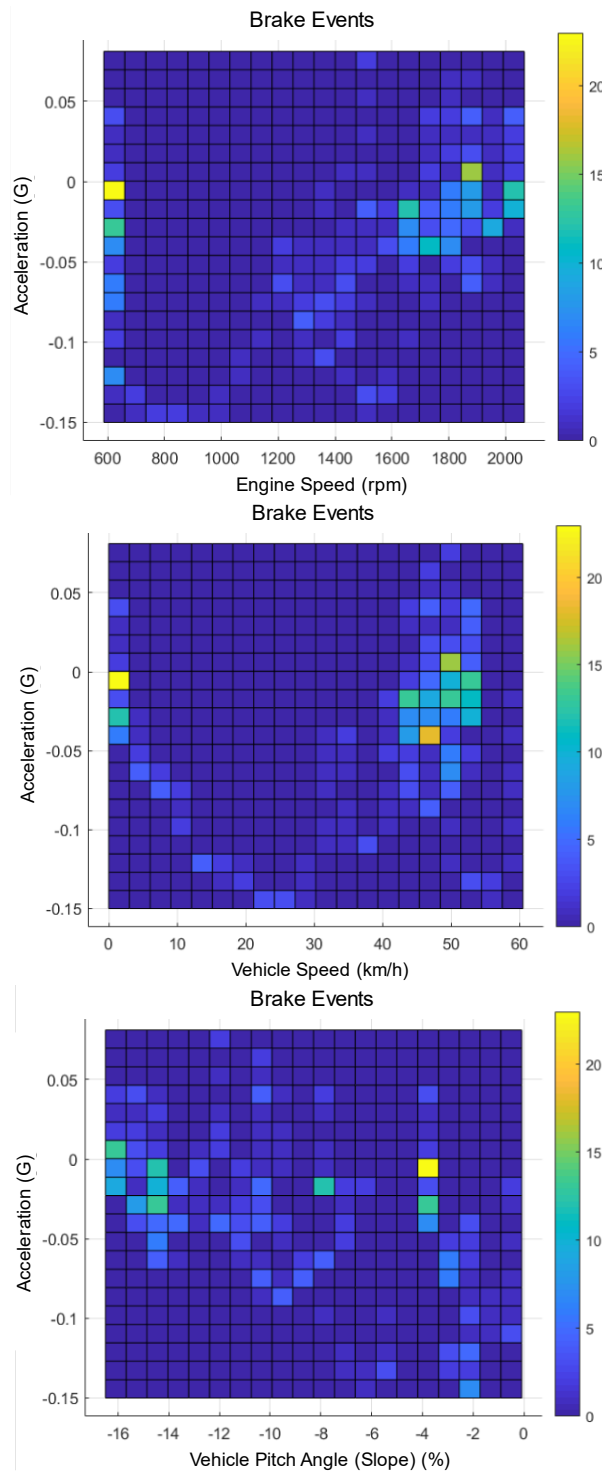


FIGURE 3.10: Acceleration (G) maps for brake events.

Acceleration maps of breaking events show that the driver was able to decelerate without using brakes during the whole driving sequence and to use retarder braking more frequently as expected from an expert driver.

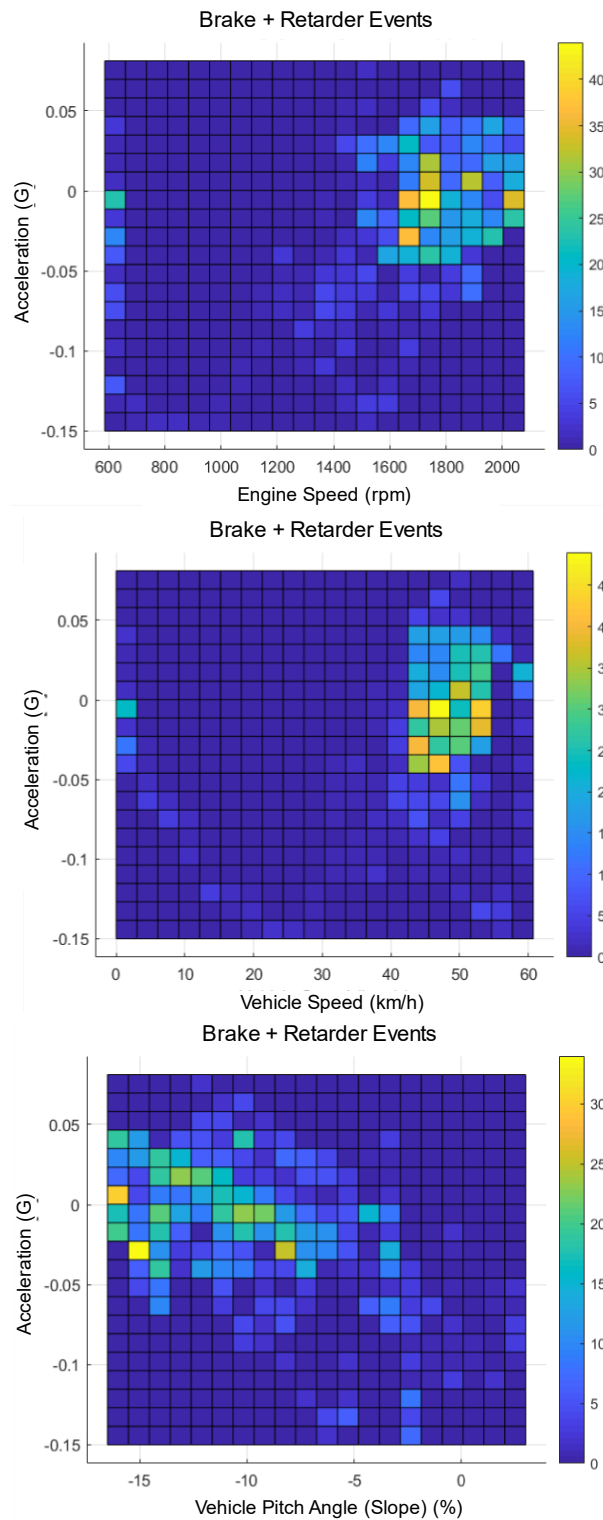


FIGURE 3.11: Acceleration (G) maps for brake + retarder events.

Chapter 4

Data Generation

Driver classification problems of the acceleration and car following models detailed in Chapters 5 - 6, driving data are needed to develop a learning algorithm. IPG's TruckMaker software [48] is utilized to generate the required driving data.



FIGURE 4.1: IPG's TruckMaker, CarMaker, BikeMaker.

This software is also used by vehicle companies to model their prototype products. In this section, the TruckMaker is represented. A reasonable model of a heavy-duty vehicle is generated for further simulations.

4.1 TruckMaker

TruckMaker is virtual testing software for heavy-duty vehicles made by IPG Automotive. It allows users to generate realistic vehicle, driver, road, and traffic models based on its built-in model database or design these models using an external computing environment such as MATLAB Simulink or National Instruments LabView.



FIGURE 4.2: Vehicle, road, driver, traffic models of TruckMaker.

4.2 Generation of a Simulation Truck Model

To train and test driving behavior classification models with realistic driving signals of driver models, the model of a commercial truck is generated in Truckmaker. Various physical and powertrain parameters of the truck and the trailer are used to create the vehicle model. Creating training data using this driver model, will allow us to test the algorithms directly using driving signals collected from the original vehicle controlled by real drivers.

4.2.1 Physical Parameters

To be able to obtain a reliable model, some of the key properties are determined that can represent the physical dynamics of the truck and the trailer. A standard

European cab-over and a trailer model from the Truckmaker database are modified using those key properties.

For the cab over model, the dimensional parameters such as the center of gravity (COG), axle positions and main outer dimensions, the mass parameters such as the kerb weights on the first and the second axles, and inertial parameters that are I_{xx} , I_{yy} , I_{zz} , I_{xy} , I_{xz} , I_{yz} are used. The key parameters are selected similarly for the trailer model with respect to the hitch position. Additionally, aerodynamic properties, such as frontal area and drag coefficient, are also considered for generated truck and trailer models. Selected parameters for both truck and trailer models are shown in Figure 4.3.

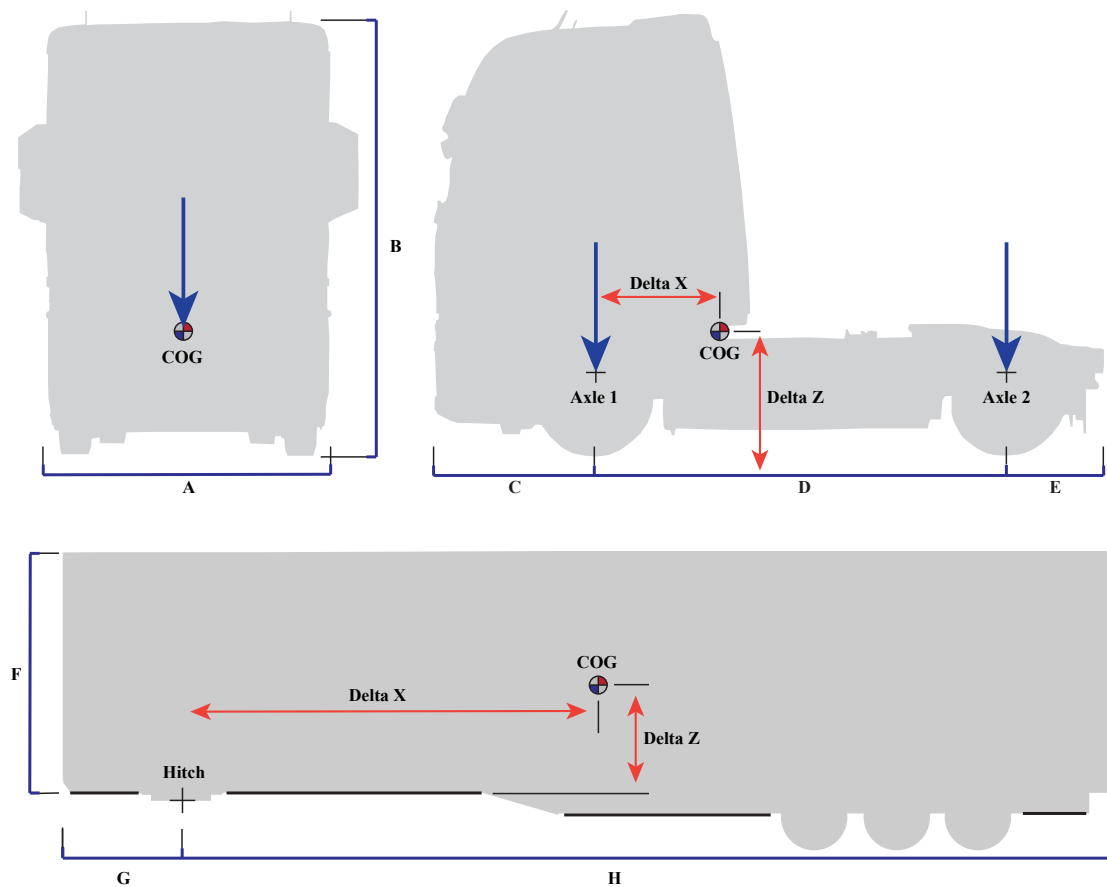


FIGURE 4.3: Physical parameters of the truck (Top) and trailer (Bottom).

4.2.2 Powertrain

Acquiring a vehicle model with relevant powertrain parameters is essential for proposed driving behavior classification models. The powertrain dynamics of the truck is modeled considering three main components of the power flow, namely engine, transmission, and differential, that can be seen in Figure 4.4.

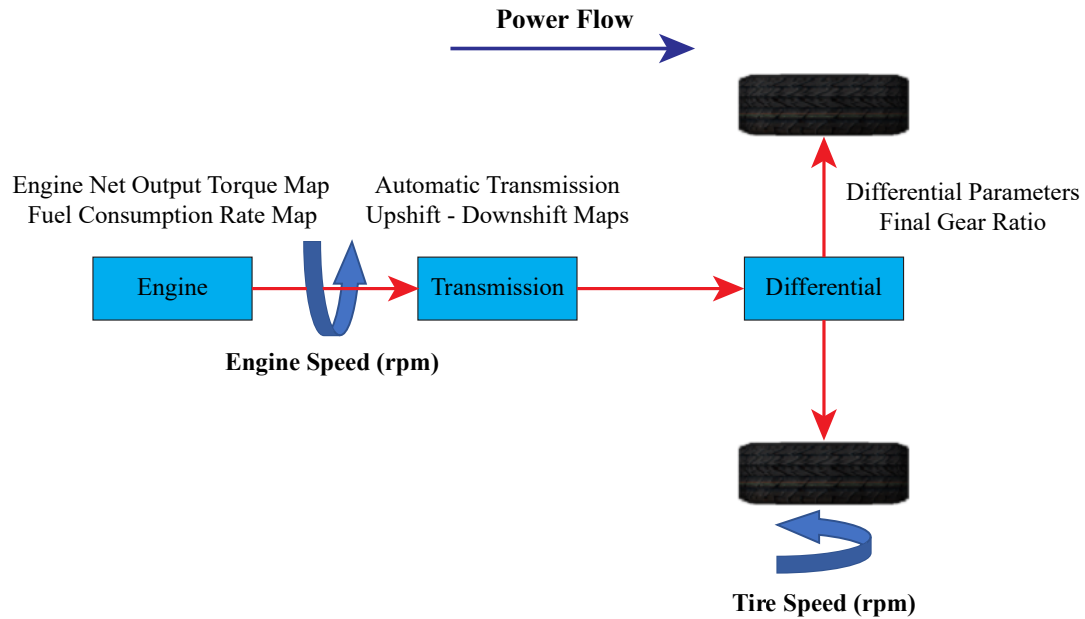


FIGURE 4.4: Powertrain power flow.

To model the engine engine net torque and fuel consumption maps are used. These maps provide engine net torque (Nm) and instantaneous fuel consumption (kg) values for engine speed - throttle position points. The maps are presented in Figure 4.5 with the parameters are scaled between 0 to 100 due to the presence of classified information.

A 12 forward speed automatic transmission model is created for the truck model. For each gear, a gear ratio, inertia, efficiency, an upshift and a downshift map of the real vehicle are set in the simulation model. The transmission model automatically switches between gears based on engine speed, throttle position using the upshift and downshift maps of the active gear. Finally, the differential is modeled using differential gear ratio (final drive ratio) and efficiency parameters.

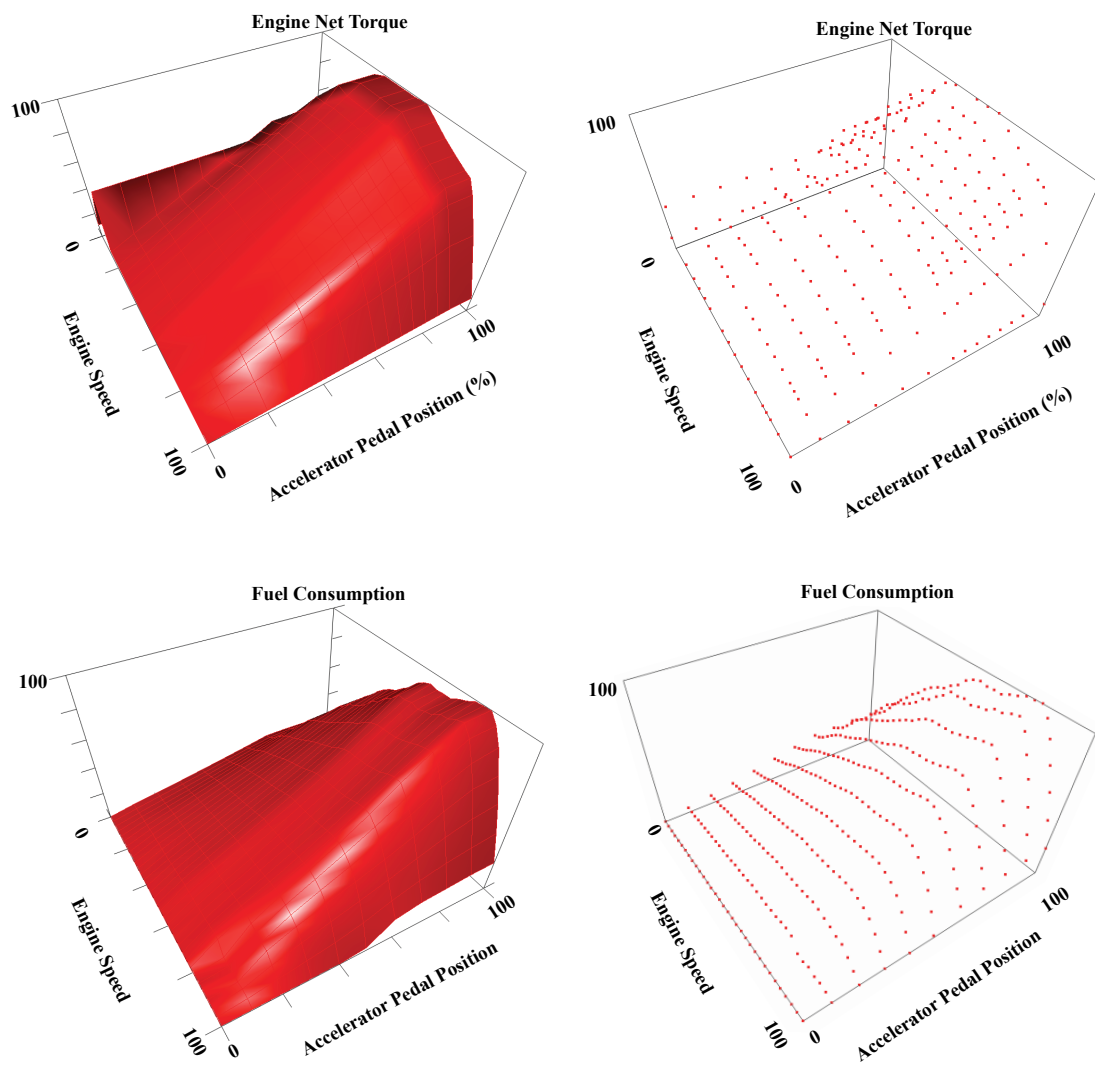


FIGURE 4.5: Engine net torque (Top) and fuel consumption (Bottom) maps.

Chapter 5

Acceleration Model

This chapter introduces the methodology and testing results for the proposed acceleration model. That includes the design of the driver, training/test road models, generation and preparation of driving signals of those drivers, introducing the LSTM based classification algorithm, and driver classification results.

5.1 Experiment Design

In this section, the methodology of designing drivers, with different acceleration characteristics, is presented. To this end, an artificial training road and a test road are generated for the acceleration model simulations.

5.1.1 Driver Design

In literature, it is known that acceleration signals can describe distinctive features of different driving behaviors. T. Krotak and M. Simlova [49] showed that, different driving behaviors of a beginner and an expert driver reflect on longitudinal - lateral acceleration signals significantly. The analysis in **Chapter 3** proved that the acceleration behaviors of a single driver vary with changing road geometries. In the light of this information, we developed the acceleration model using the

driver model of IPG’s commercially available software TruckMaker [48] that is parameterizable based on acceleration behaviors.

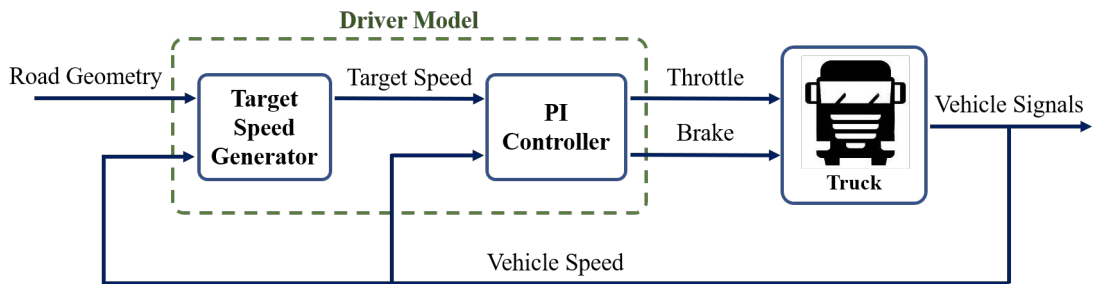


FIGURE 5.1: TruckMaker’s driver model.

The driver model of TruckMaker continuously generates target speeds with previewing the road curves to reach a cruising speed without exceeding certain acceleration limits. The model operates the vehicle at these target speeds using a PI controller that generates throttle and brake outputs, like a real driver (Figure 5.1). To mimic real driving behaviors, we set the values of five driver model parameters, namely maximum longitudinal acceleration and deceleration (braking), maximum lateral accelerations for left and right turns and the cruising speed.

Driver Nm.	Speed Limit	Longitudinal Acc. Limit Parameters		Lateral Acc. Limit Parameters	
	Tx [km/h]	ax_t [m/s ²]	ax_b [m/s ²]	ay_l [m/s ²]	ay_r [m/s ²]
1	90	0.1	2.5	1.4	1.4
2	90	0.1	1.5	0.6	0.6
3	90	0.1	1.0	0.2	0.2
4	90	0.5	2.5	1.4	1.4
5	90	0.5	1.5	0.6	0.6
6	90	0.5	1.0	0.2	0.2

TABLE 5.1: Acceleration behavior model driver parameters

Six driver classes are designed for the simulations using the parameters given in Table 5.1. First three drivers (Driver 1-3) have lower longitudinal acceleration

limits than the last three drivers (Driver 4-6). Lateral acceleration and (braking) deceleration limits are decreasing gradually from driver 1 to 3, or driver 4 to 6, in each set (Figure 5.2).

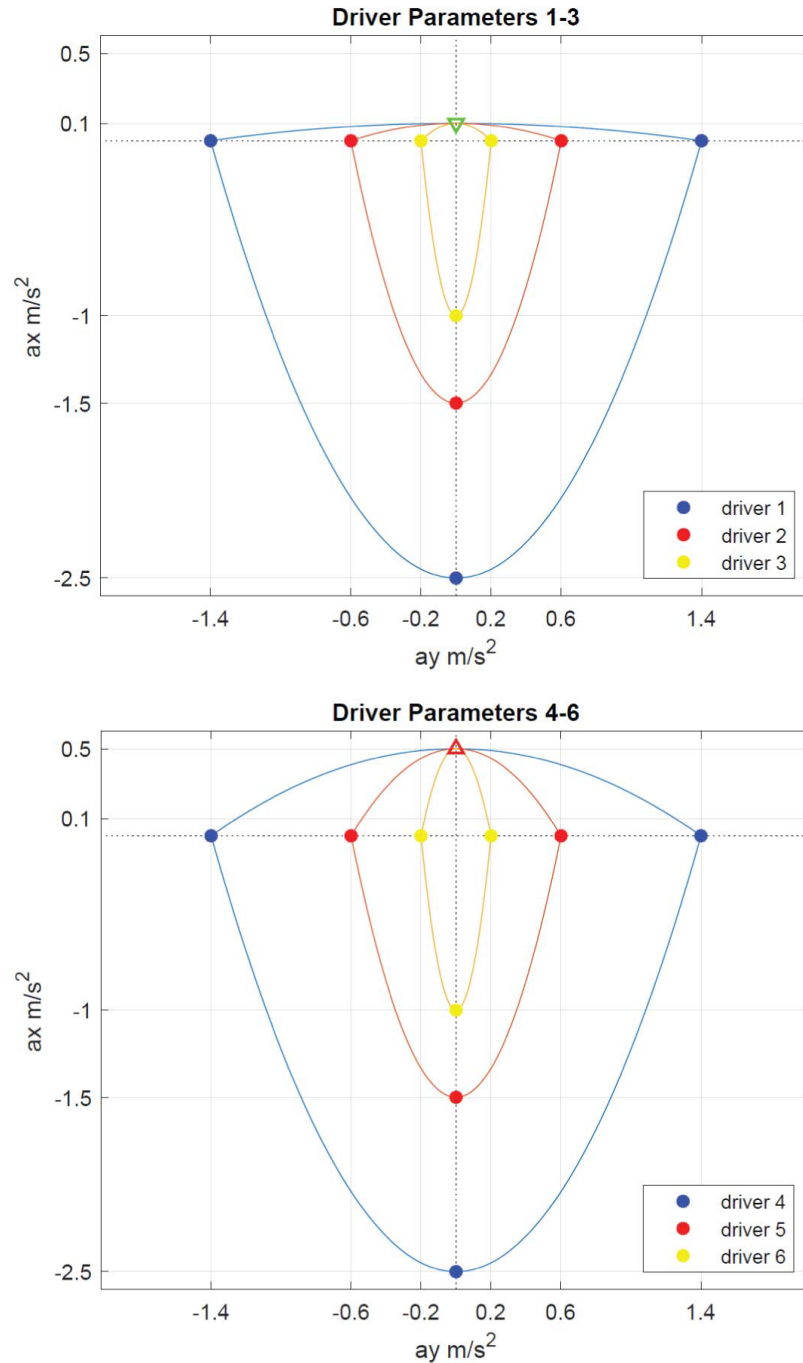


FIGURE 5.2: Acceleration limit parameters of drivers 1-3 (Top), 4-6 (Bottom).

Drivers with the same lateral and different longitudinal acceleration limits are named as differing classes (Driver 1-4, 2-5, 3-6) due to their similar behaviors in

some conditions . Acceleration limits of all drivers are visualized in Figure 5.3.

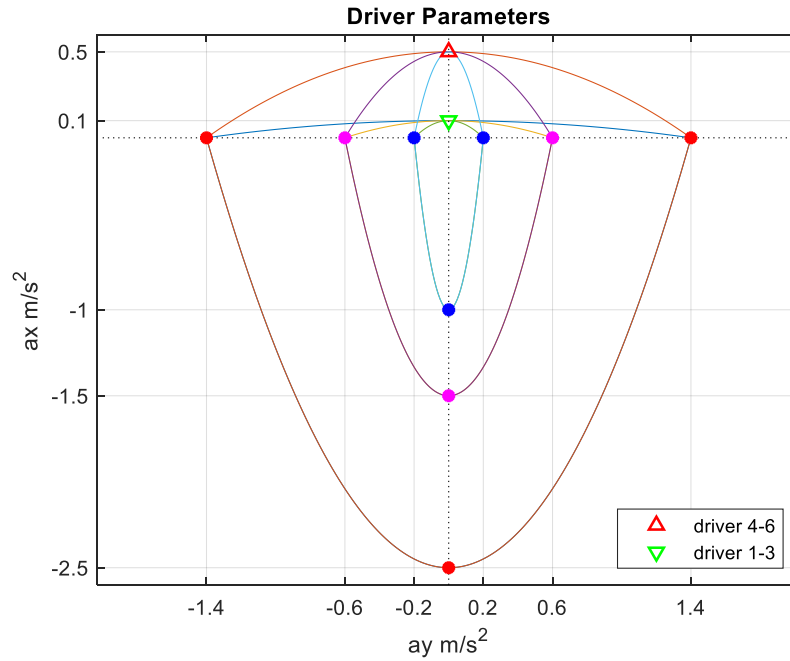


FIGURE 5.3: Acceleration limit parameters of drivers 1-6.

5.1.2 Training and Test Road Design

An artificial training road is designed to extract different longitudinal and lateral acceleration behaviors of drivers using different road profiles. The aim is to train the proposed classification algorithm with driving signals covering most of the possible road geometries of a freeway or a rural highway.

To generate simulation routes systematically, the road block concept is generated. A road block is designed to be 200 m length of an arc that is defined by 6 points in 3D space. It represents a small portion of a road with constant grade and horizontal curve profile (without superelevation). A road block has two parameters which are grade of the road in percentage and central angle of the horizontal curve in degrees. Several road blocks are sequentially added to form a complete training/test road for simulations.

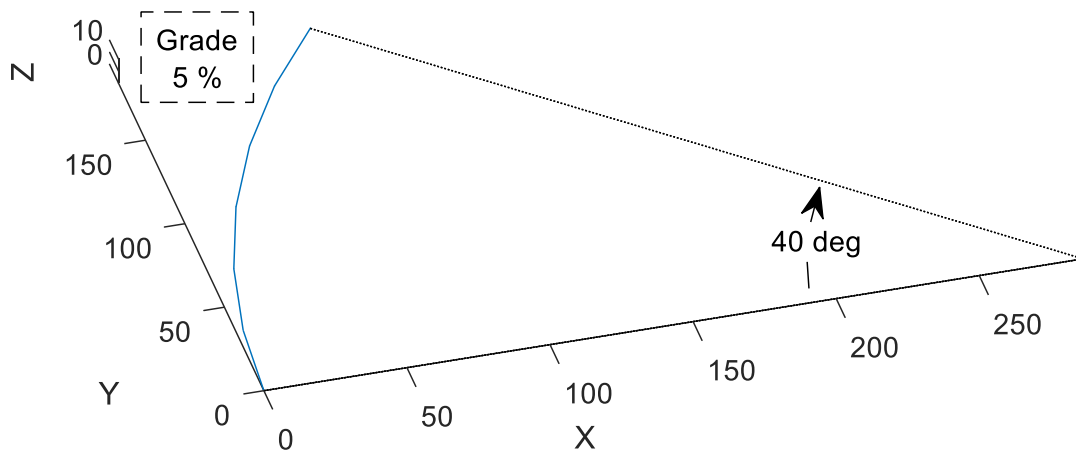


FIGURE 5.4: X-Y-Z profile of an example road block.

The grade parameter represents uphill when it is positive and downhill when it is negative. The central angle parameter represents a right-turn when it is positive, a left-turn when it is negative, and a straight road when it is zero. An example of a road block is presented in Figure 5.4.

In order to increase the modeling accuracy for different road types, the training road is targeted to cover different combinations of grade – central angle parameters in desired ranges. These ranges are determined based on the radius of the curvature and the length of the grade [50].

Altitude and curvature profile of a road segment consists of two parts. In the first part, road blocks with positive and negative grades are positioned next to each other from lower slopes to higher slopes. This part composes different vertical crest and sag curves. In the second part, there is one road block with 0 % grade in after each uphill and downhill block (Figure 5.5, Z-Y profile). The same pattern is used for horizontal curves with one central angle value in a segment (Figure 5.5, Y-X profile). The plain road blocks between curves will let driver models accelerate starting from different speeds to reach their target speed.

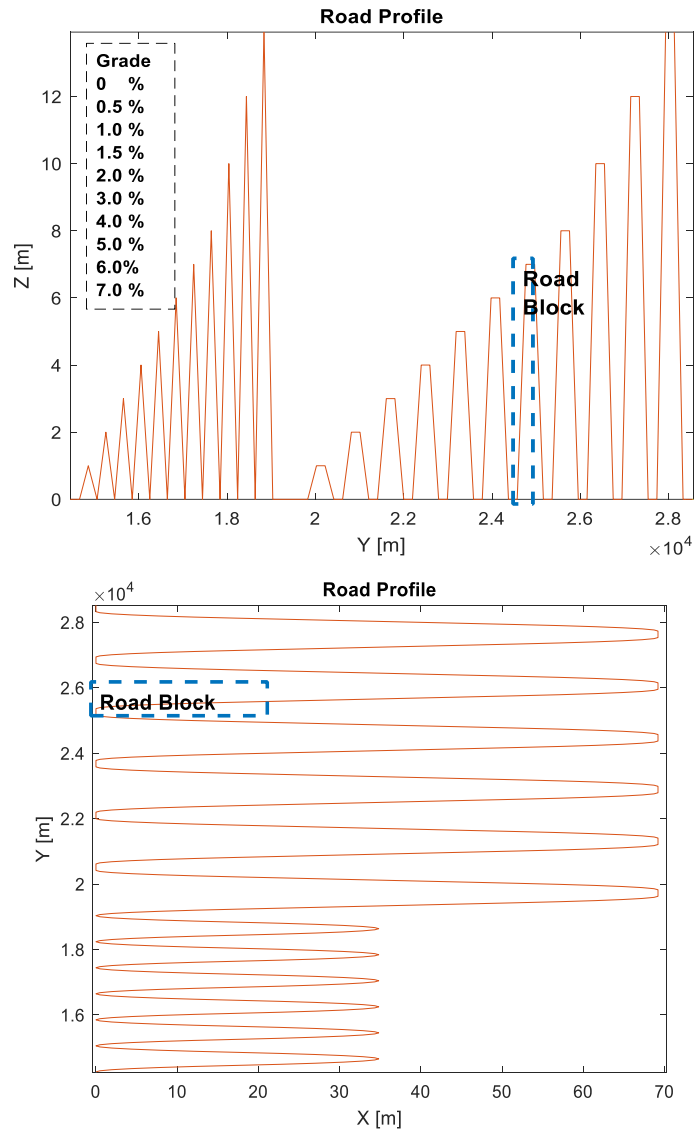


FIGURE 5.5: (Top) Z-Y and (Bottom) Y-X profiles of an example training road segment.

To cover determined grade and central angle ranges, 8 road segments are sequentially added to form the complete training road. Each segment follows the same grade and horizontal curve pattern with a different central angle value. The segments are depicted in Figure 5.6 with different colors.

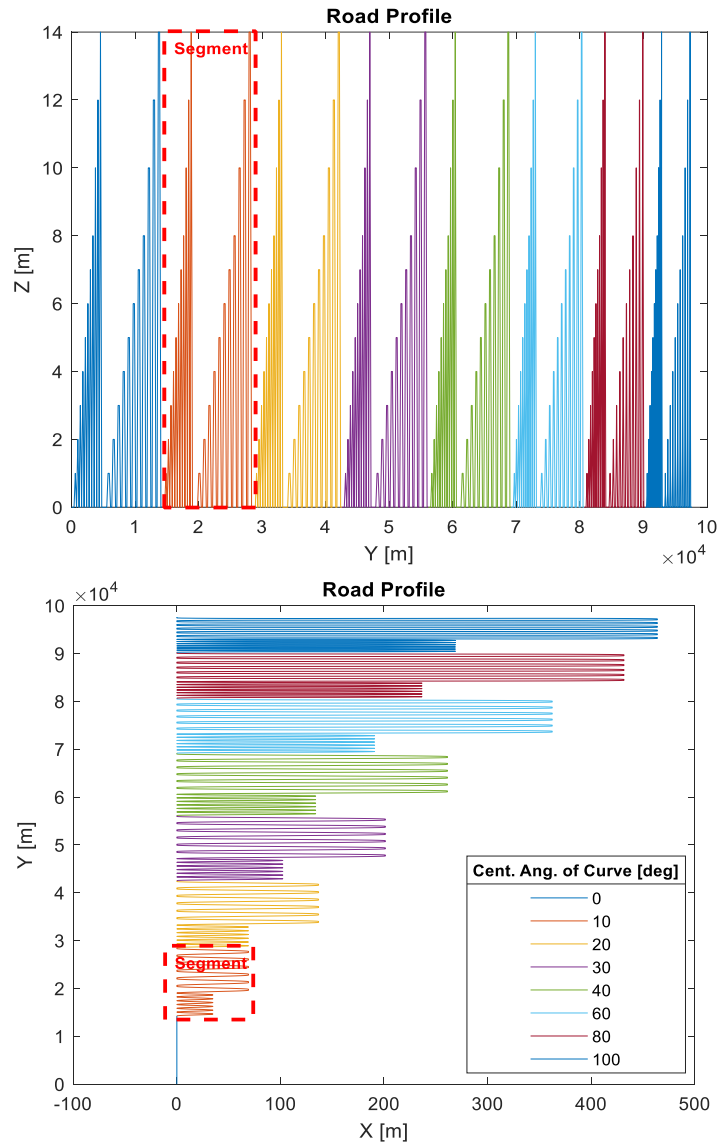


FIGURE 5.6: (Top) Z-Y profile and (Bottom) Y-X profile of the training road.

The designed training road is 114.7 km that involves 8 road segments with 72 road blocks in each. In this road, lower grade - central curve intersections are covered more densely, since the existence of simultaneous sharp curvatures and high grades in a highway is uncommon. Finally, an 8 km, smoother road is designed using road blocks with arbitrary parameters to test the developed algorithm. Coverages of the training and test roads are shown in Figure 5.7.

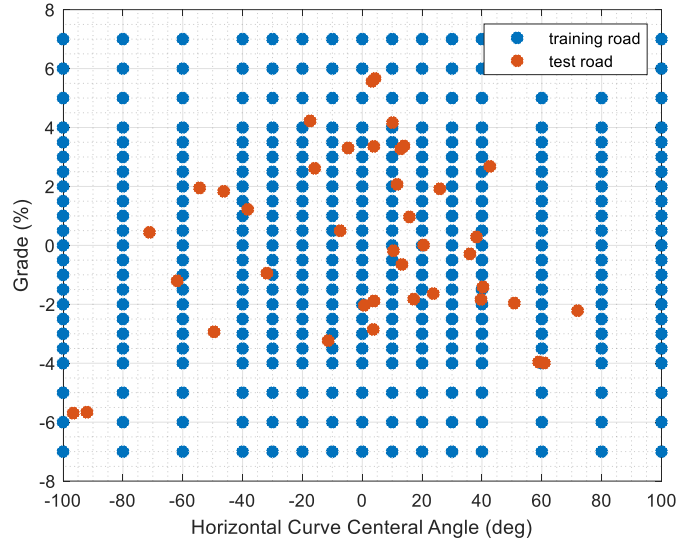


FIGURE 5.7: Grade - central angle coverage map for training and test roads.

5.2 Driver Classification

In this section, the data generation method and the LSTM based classification algorithm are explained.

5.2.1 Data Generation for Acceleration Model

Driving data of the proposed drivers are produced using TruckMaker. Each driver is simulated on both training and test roads using the vehicle model that is explained in Chapter 4 with five different trailer loads, i.e., 0, 5, 10, 15, 20 tones. In total, 60 different driving data of six driver classes is recorded at 5 Hz.

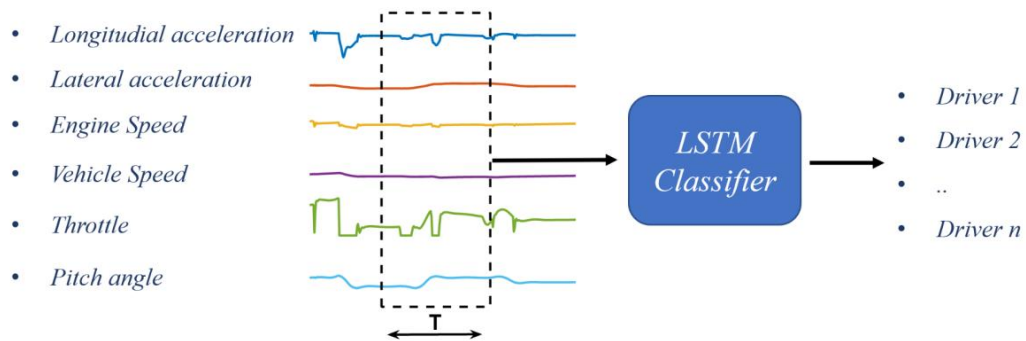


FIGURE 5.8: Classification model inputs.

Longitudinal and lateral accelerations, engine and vehicle speeds, and pitch angle of the vehicle have been selected as inputs to the proposed algorithm. 30 second time-window (T) is shifted through all the driving data with the period of 15 seconds. The signals within the window are labeled with its driver number and used as a sample for classification model (Figure 5.8).

5.2.2 Long Short Term Memory Network

In order to classify the driving behavior, the temporal relations of the selected inputs are targeted to be explored. For this purpose, a type of recurrent neural network called Long Short Term Memory (LSTM) [45] network is selected.

A typical LSTM neural network includes an input layer, a recurrent hidden layer and an output layer. Different from classical neural networks, LSTM networks have memory capabilities with three gates called: forget gate, update gate and output gate (Figure 5.9).

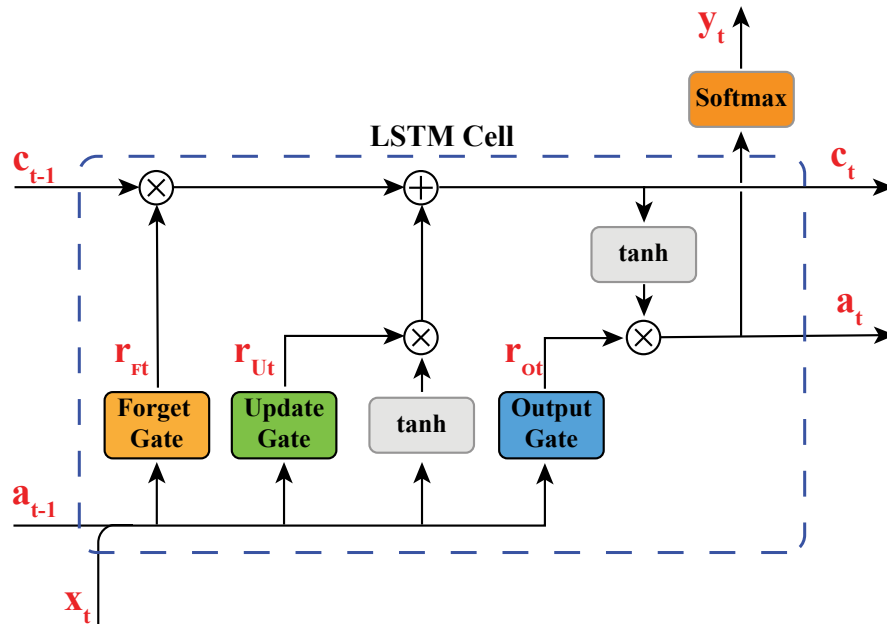


FIGURE 5.9: LSTM structure.

The gates in an LSTM structure are independent neural networks with the same dimension and sigmoid activation functions. Input to these gates is the concatenation of measurements ($x(t)$) and the output state of previous LSTM cell ($a(t-1)$). In this study, measurements include the signals shown in Figure 5.8 for a time period of T.

Forget gate adjust the information to discard from the cell, update gate and tanh gate decide the values from the input to update the memory state ($c(t)$), and output gate determines what to output based on input and memory of the cell. A typical LSTM neural network can be implemented by the following equations

$$\Gamma_f = \sigma(W_f [a_{t-1}, x_t] + b_f) \quad (5.1)$$

$$\Gamma_u = \sigma(W_u [a_{t-1}, x_t] + b_u) \quad (5.2)$$

$$\tilde{c}_t = \tanh(W_c [a_{t-1}, x_t] + b_c) \quad (5.3)$$

$$\Gamma_o = \sigma(W_o [a_{t-1}, x_t] + b_o) \quad (5.4)$$

$$c_t = \Gamma_u * \tilde{c}_t + \Gamma_f * c_{t-1} \quad (5.5)$$

$$a_t = \Gamma_o * c_t \quad (5.6)$$

where W_f, W_u, W_c, W_o, W_r are the weight matrices and b_f, b_u, b_c, b_o, b_r are the bias vectors of corresponding operations. At the end of the recurrent calculations of 5.1 - 5.6 for a time period (T), the last output state (a_{last}) is fed to the output layer and employed in softmax function to find the probabilities for each class as follows:

$$z = W a_{last} + b \quad (5.7)$$

$$softmax(X_i) = \frac{\exp(X_i)}{\sum_{j=1}^k \exp(X_j)}, \quad i = 1, 2, \dots, k \quad (5.8)$$

where W and b are the weight and the bias vectors in the output layer, and k is the number of classes.

5.3 Results

This section presents a comparison between the designed drivers based on their simulation outputs and a discussion of the classification results in the light of this driver comparison.

5.3.1 Driver Comparison

The effects of selected driver parameters can be easily observed from the signals presented for different segments of the training road (Figure 5.10 - 5.11). For instance, the lateral acceleration plot in Figure 5.10 shows that the Driver 1 remains in the $|ay| < 1.4m/s^2$ band while Driver 2 in $|ay| < 0.6m/s^2$ and Driver 3 in $|ay| < 0.1m/s^2$ during the whole trip. To obtain such conditions, these three models operate with various speed profiles in different curves. Longitudinal acceleration behaviors slightly vary between these models as expected. A similar comparison could be made between the last three drivers.

It is also significant to compare the drivers with the same lateral acceleration limits. In that sense, comparison of driving speed, longitudinal and lateral acceleration signals for Driver 1- 4 is given in Figure 5.11. Driver 4 has operated with a higher longitudinal acceleration profile than Driver 1, while the lateral acceleration profile is almost the same. However, all driving signals of such driver couples overlap in some cases that can be observed easier from the test road. We have named these driver couples as “differing class” for further discussions, i.e., Driver 1-4, 2-5, 3-6.

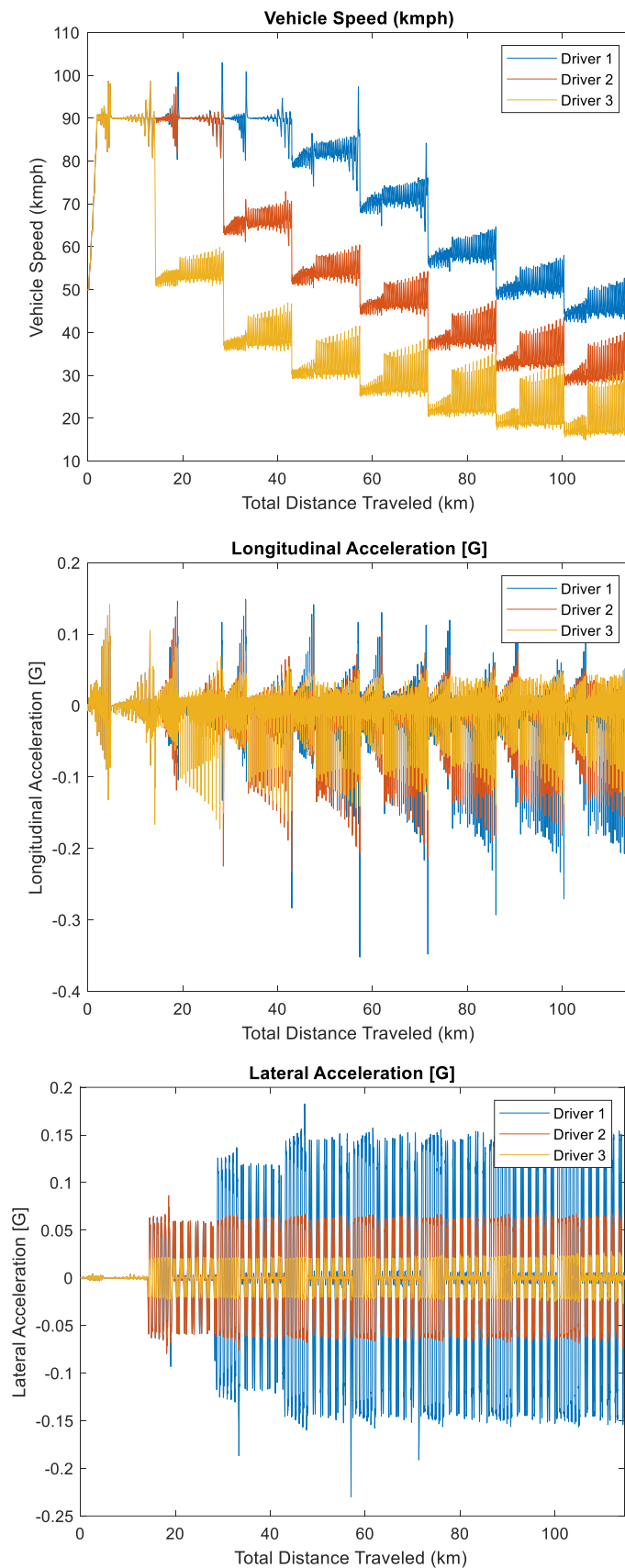


FIGURE 5.10: Driver comparison based on vehicle speed, longitudinal and lateral acceleration signals acquired from the training road for drivers in the same set (10 ton trailer load).

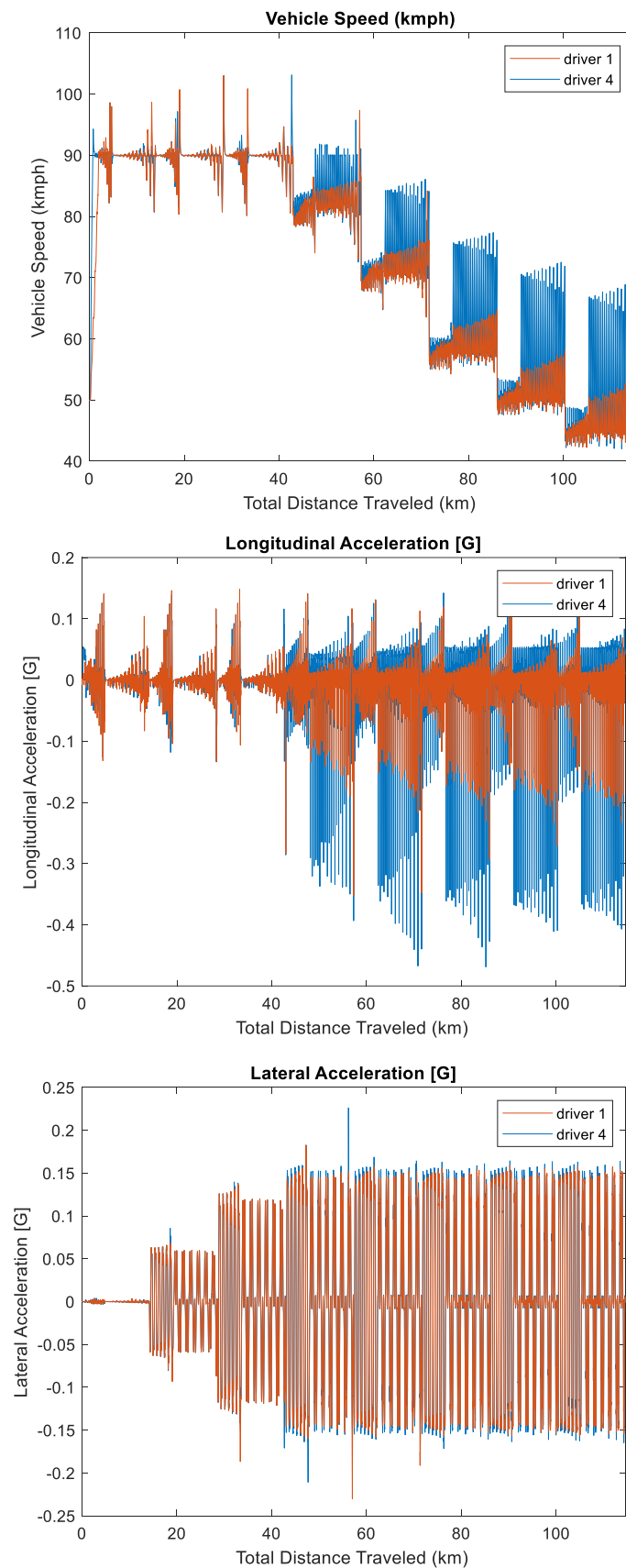


FIGURE 5.11: Driver comparison based on vehicle speed, longitudinal and lateral acceleration signals acquired from the training road for drivers in a differing class (10 ton trailer load).

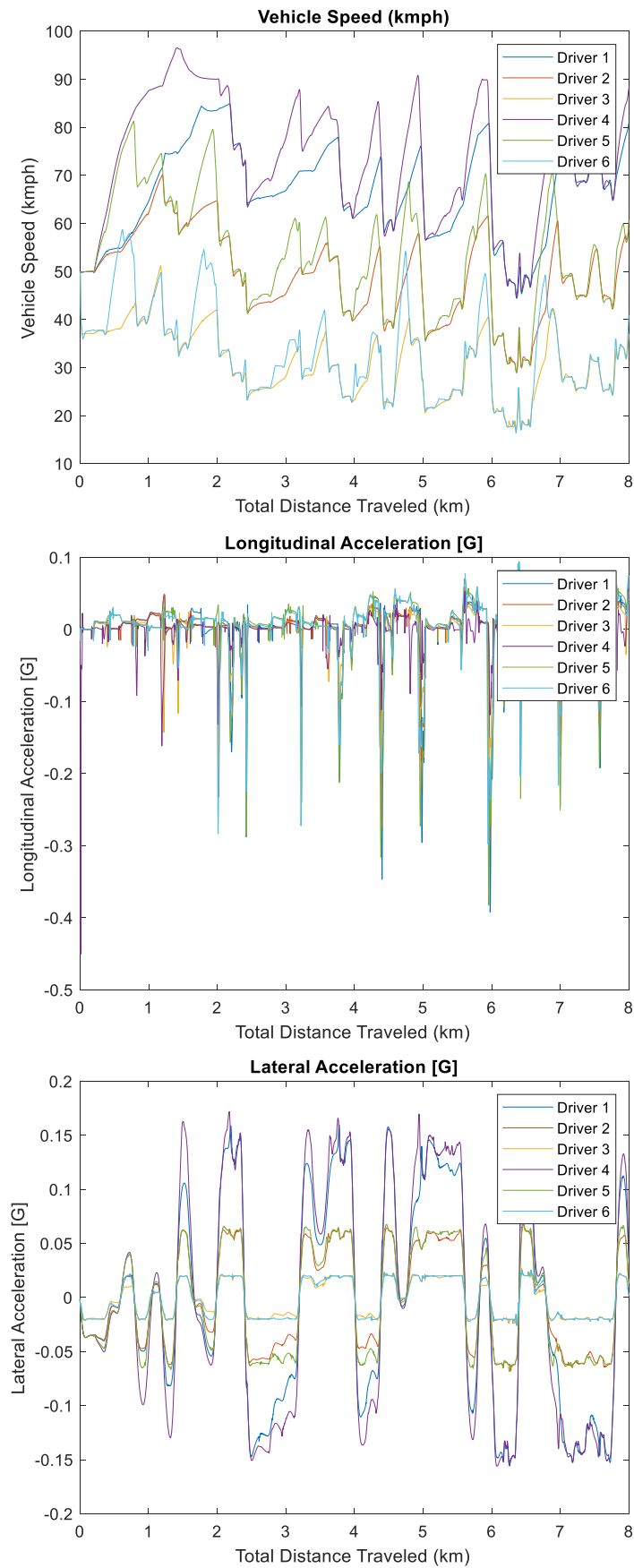


FIGURE 5.12: Vehicle speed, Longitudinal and lateral acceleration signals of all six drivers from the test road (10 ton trailer load).

5.3.2 Classification Results

The proposed LSTM network structure is trained and tested with the driving signals of drivers in a small time-window. Driving signals are collected using a realistic truck model with five different trailer loads. The dynamics of the vehicle change significantly based on the mass of the vehicle.

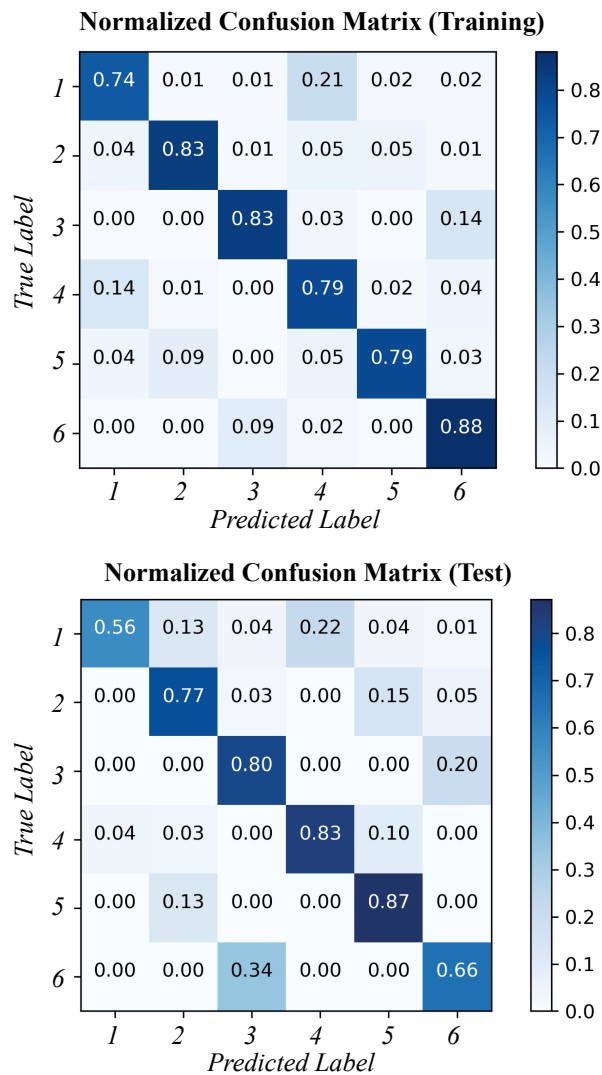


FIGURE 5.13: Normalized confusion matrix of the training (Top) and the test set (Bottom).

Experiments have shown that size of the window (T) considerably affect the classification results. When the T is too small, sample data become insufficient to distinguish drivers. On the other hand, a large T lead to unrealistic signal forms

due to the repetitive nature of the training road. As a result, a 30 second window is selected that provide decent amount of information about driving behaviors in each sample. Additionally, shifting the window with the period of 15 seconds have enhanced the training performance.

Classification accuracies of the training and the test sets are respectively 82.24 % and 74.70 %. Confusion matrix of training and test roads (Figure 5.13) show that, majority of the misclassifications made between differing classes. When differing classes are assumed as correct outputs, the accuracy of training and test sets are respectively, 93.1 % and 92.8 % (Figure 5.14). Although, a differing classification cannot be labeled directly as a true output, there is a high chance that differing classes behave the same in the majority of misclassified samples.

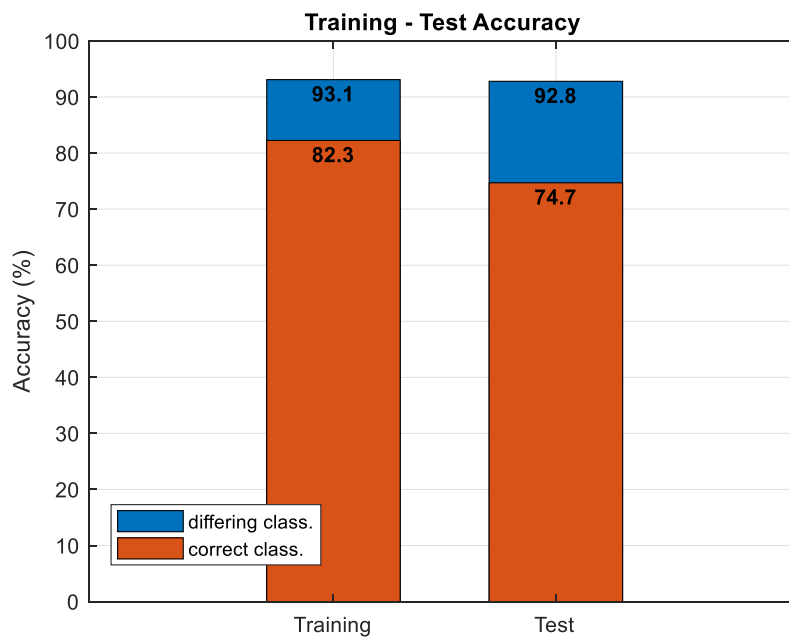


FIGURE 5.14: Training and test accuracy graph for correct and differing classes.

Nevertheless, the classification results have proven that the proposed LSTM structure is effective at extracting dynamic driving behaviors of drivers in a time-window.

Chapter 6

Car Following Model

The car following, which represents how a driver behaves while driving behind another vehicle, forms the majority of urban driving situations. In general, this characteristic of a driver is affected by two main categories that are personal differences of drivers, such as age, gender, aggression, driving skill and vehicle properties, and situational factors, such as driving time/day, weather and road conditions, fatigue, stress, alcohol and drugs [51].

The differences between drivers create the variation in headway preference, which is a measure of the distance between vehicles in time [52]. For instance, male drivers choose lower headways than females [53] and aged drivers (59+) prefer almost 25% higher headways than the normal age driver (23-37) [54].

In this chapter, the methodology and testing results for the proposed car following model is presented. That includes the design of three drivers with different aggression levels, generation and preparation of driving signals of those drivers, and classification results.

6.1 Experiment Design

This section presents the methodology of designing drivers, with different aggression levels based on their car following behaviors.

6.1.1 Driver Design

As a driver model, a target speed generator (TSG) is designed instead of Truck-maker's TSG based on the following distance for this car following model. Proposed TSG algorithm generates target speeds for the truck depending on the distance and speed difference measurements and the vehicle speed signal. Distance measurements are acquired from the radar sensor which is placed in front of the truck model in the simulation environment. The PI controller of the TruckMaker controls the truck with throttle and brake outputs as in previous acceleration model 6.1.

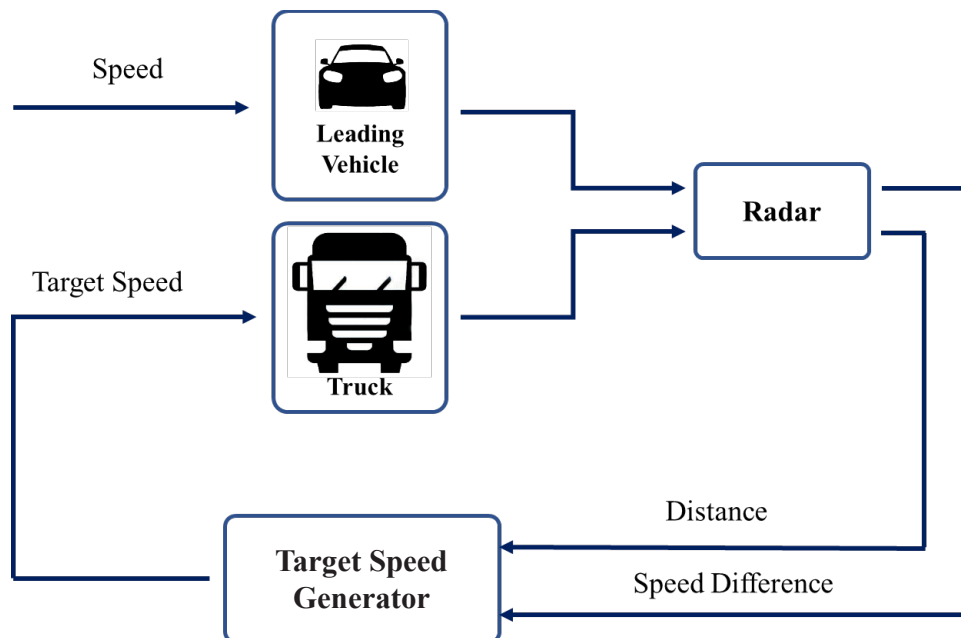


FIGURE 6.1: Car following model outline.

Four different driving modes are defined considering the aim of mimicing the driving behaviors of a real driver and simulating the truck model smoothly in data

generation. First, a hysteresis thresholding perspective is applied to the selection between driving modes. Two distance thresholds (λ_1 , λ_2) determine the characteristics of the designed drivers 6.2.

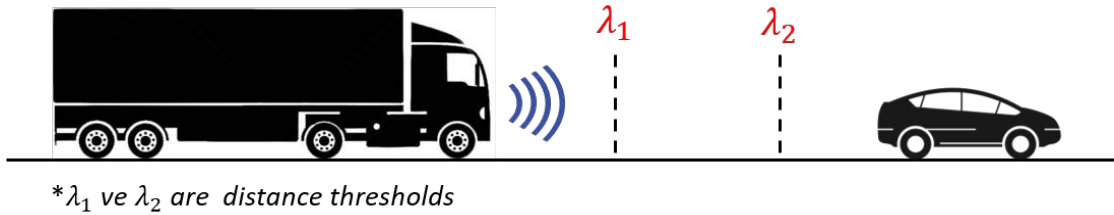


FIGURE 6.2: Hysteresis thresholding for the target speed generator.

The designed driving modes for TSG are named as follows;

- **Mode 1:** Accelerate
- **Mode 2:** Maintain current speed
- **Mode 3:** Decelerate
- **Mode 4:** Stop

The first mode is designed for driver to accelerate with an independent speed profile from the leading vehicle. The second mode is a transition mode for the driver to prepare for leading vehicle's next action. The third mode is the deceleration (braking) process. During stops, sudden changes between mode 3 and 1 lead oscillation due to the dynamics of heavy duty vehicle. The fourth mode aims to prevent these oscillations from happening. The rules for mode selection based on sensor measurements are given below.

- **Mode 1:** ($\lambda_2 < Distance$) | ($0 < speed\ difference$)
- **Mode 2:** ($\lambda_1 < Distance < \lambda_2$) & ($speed\ difference < 0$)
- **Mode 3:** ($Distance < \lambda_1$) & ($speed\ difference < 0$)
- **Mode 4:** ($Distance < 10\ m$) & ($V_{truck} < 2\ km/h$)

The proposed TSC generates different speed profiles based on equations 6.1 - 6.4 respectively for Mode 1-4.

$$V_{target} = b * t^a \quad (6.1)$$

$$V_{target} = V_{truck}^- \quad (6.2)$$

$$\Delta = speed\ difference * \frac{d}{distance} \quad V_{target} = V_{truck}^- + \Delta \quad (6.3)$$

$$V_{target} = 0 \quad (6.4)$$

Three driver models are designed using the proposed target speed generator with parameters given in Table 6.1. From driver 1-3, drivers chose higher accelerations and speeds, lower distance thresholds which means late decelerating, and more aggressive braking. Unlike the drivers defined in the chapter 5, defined drivers can be labeled as calm, normal and aggressive respectively for drivers 1-3.

Driver Num.	Mode Selection Parameters		Mode 1 Param ($b * t^a$)		Mode 3 Parameters
	Threshold 1	Threshold 2	a	b	d
1	125 m	200 m	0.8	2.05	1.5
2	100 m	180 m	0.7	3.86	1.3
3	75 m	150 m	0.6	7.07	1.0

TABLE 6.1: Car following model driver parameters.

A speed profile using equation 6.1 is set for each driver for the acceleration behaviors of these drivers. These speed profiles allows drivers to use higher accelerations in lower speeds which is necessary due to vehicle dynamics. Defined speed profiles and resulting accelerations for drivers 1-3 are shown in Figure 6.3.

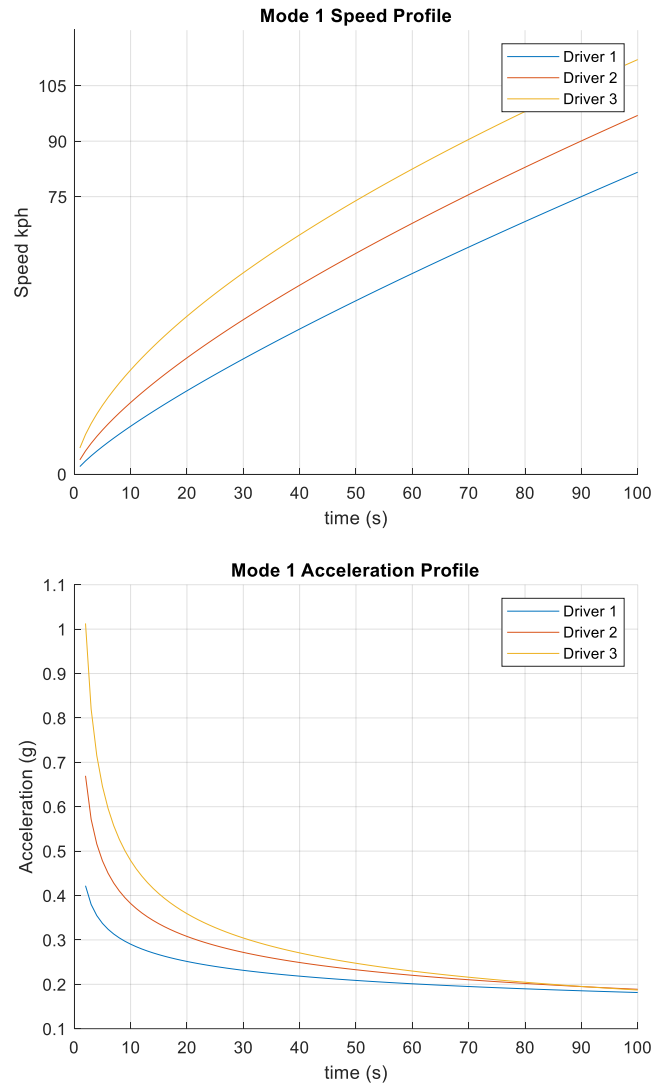


FIGURE 6.3: Mode 1 speed (Top) and acceleration (Bottom) profiles of the drivers 1-3.

6.2 Driver Classification

In this section, the data generation method for the car following model is presented.

6.2.1 Data Generation for Car Following Model

Driving signals of designed drivers are simulated using the same vehicle model that is explained in **Chapter 4** with 10 ton trailer load on a realistic road model.

For both training and test simulations a leading vehicle speed profile is required (Figure 6.1). Driver models control the vehicle vehicles based on their defined car following characteristics.

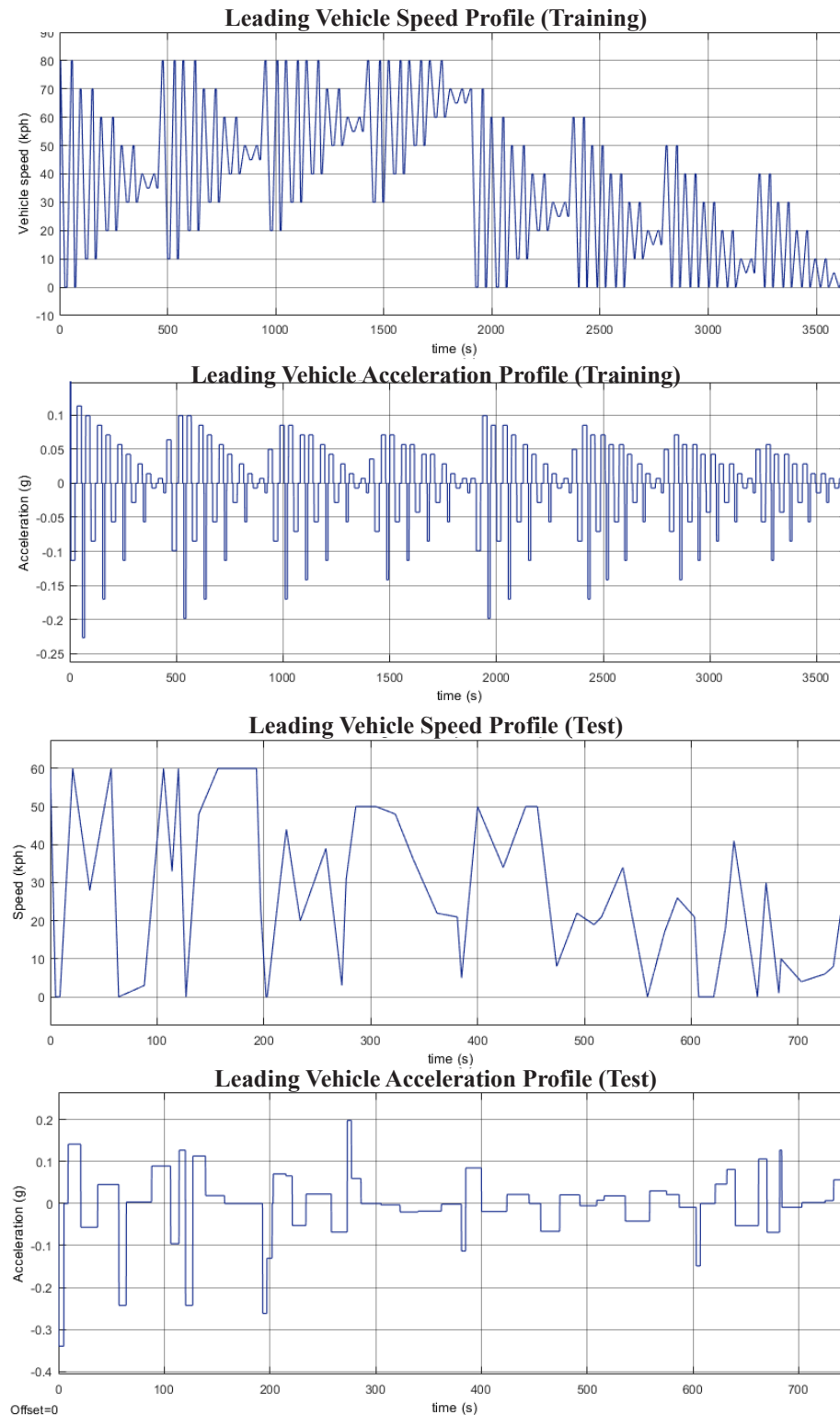


FIGURE 6.4: Leading vehicle speed and acceleration profiles for the training and test simulations.

Two speed profiles for leading vehicles are designed to generate training and test simulations. Speed profile for the training consists of braking events with varying decelerations while the speed profile for test consists of arbitrary braking events Figure 6.4.

Longitudinal acceleration, vehicle speed, throttle position, pitch angle and distance to forward vehicle signals of the truck model is selected for the car following classification model. Classification samples are generated similarly to the previous classification model (Figure 5.8) by shifting 12 second time-window (T) through all driving data with the period of 6 seconds.

6.3 Results

This section presents a comparison between the designed drivers based on their simulation outputs and a discussion of the classification results.

6.3.1 Driver Comparison

The average speed - total fuel consumption values acquired training and test simulations, presented in Table 6.2 provides a comprehensive comparison between the drivers. Because car following is dependent on leading vehicle speed, driver models can only operate with similar average speeds. However, the aggression level of drivers dramatically increases total fuel consumption.

Driver	Training Simulations		Test Simulations	
Num.	Average Speed km/h	Fuel Consumption L/100km	Average Speed km/h	Fuel Consumption L/100km
1	36.028	38.606	28.144	44.2962
2	36.107	41.500	28.284	44.6562
3	36.123	47.643	27.963	48.7414

TABLE 6.2: Driver comparison based on average speed and fuel consumption

Figure 6.5 shows that from driver 1 to 3 higher throttle positions and lower following distances are observed, as expected. Even though these aggressive behaviors create instant higher speeds, they eventually lead lower average speeds due to more frequent stops.

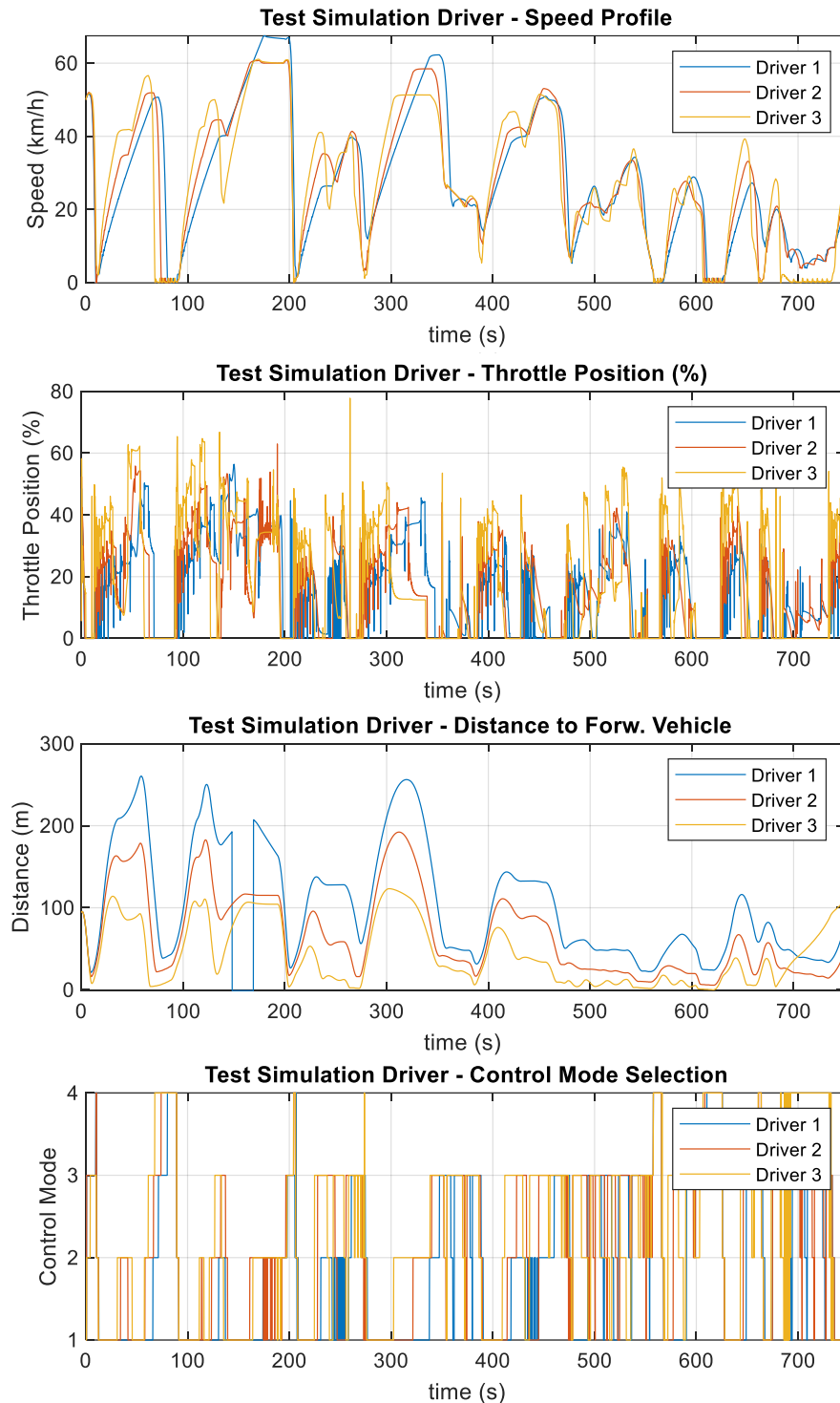


FIGURE 6.5: Driver comparison based on vehicle speed, throttle position, following distance and control mode selection signals acquired from the test road.

6.3.2 Classification Results

The accuracy of 99.22% for the training, 76.96% for the test sets are achieved. The common error is predicting an aggressive driver model with a less aggressive one, that can be seen from the confusion matrix of test results (Figure 6.6).

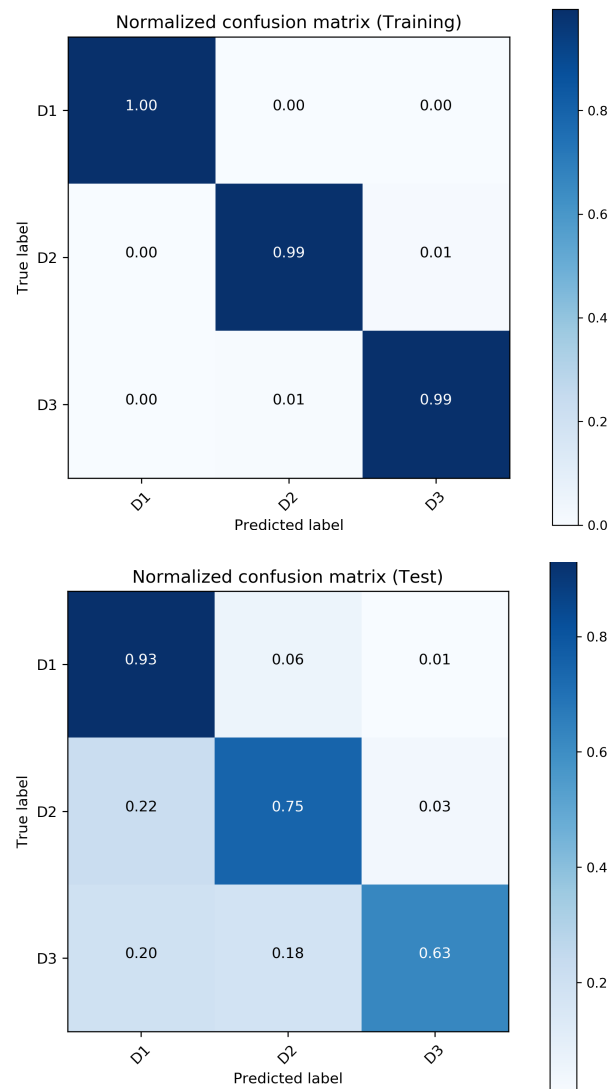


FIGURE 6.6: Normalized confusion matrix of the training (Top) and the test set (Bottom).

Chapter 7

Conclusion and Future Works

We have proposed a new method using a Long Short Term Memory (LSTM) network with the aim of recognizing different driving behaviors in a small time window. Different driving behaviors with different acceleration and car following characteristics are designed. First, six drivers are created based on longitudinal and lateral accelerations using the driver model of IPG's TruckMaker. Driving signals of driver models are collected from a realistic truck model with varying carry loads in Truckmaker. An artificial training road is designed to simulate possible road geometries and to extract driving behaviors of designed driver models. The accuracy of the LSTM classifier achieved up to 93.1% and 82.3% in training and test cases, respectively. For car following model, three drivers with different aggression levels are created using an algorithm that mimics car following characteristics of a driver with four driving modes. Driving signals of drivers are obtained as in the acceleration model and labeled for a shorter time window. The classifier predicted the car following behaviors with 99.2% training and 77.0% test accuracy.

Results show that the proposed LSTM classifier has the capability of capturing dynamic relations of driving signals in a small time window. With the ability to recognize driving behaviors, more customized Advanced Driver Assistance Systems can be developed to enhance efficiency and the safety of vehicles.

As future work, prediction errors of classification algorithms will be investigated to improve driver designs. Developed models will be used to classify the driving signals of actual drivers controlling the real truck.

Bibliography

- [1] H. Liimatainen, “Utilization of fuel consumption data in an ecodriving incentive system for heavy-duty vehicle drivers,” *IEEE Transactions on intelligent transportation systems*, vol. 12, no. 4, pp. 1087–1095, 2011.
- [2] L. Vlacic, M. Parent, and F. Harashima, *Intelligent vehicle technologies*. Elsevier, 2001.
- [3] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, “Three decades of driver assistance systems: Review and future perspectives,” *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 4, pp. 6–22, 2014.
- [4] H. J. Walnum and M. Simonsen, “Does driving behavior matter? an analysis of fuel consumption data from heavy-duty trucks,” *Transportation research part D: transport and environment*, vol. 36, pp. 107–120, 2015.
- [5] P. Guo, Z. Li, Z. Zhang, J. Chi, S. Lu, Y. Lin, Z. Shi, and J. Shi, “Improve fuel economy of commercial vehicles through the correct driving,” in *Proceedings of the FISITA 2012 World Automotive Congress*, pp. 87–96, Springer, 2013.
- [6] J. R. Treat, N. Tumbas, S. McDonald, D. Shinar, R. D. Hume, R. Mayer, R. Stansifer, and N. Castellan, “Tri-level study of the causes of traffic accidents: final report. executive summary.” 1979.
- [7] B.-C. Yin, X. Fan, and Y.-F. Sun, “Multiscale dynamic features based driver fatigue detection,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 03, pp. 575–589, 2009.

-
- [8] C. C. Liu, S. G. Hosking, and M. G. Lenné, “Predicting driver drowsiness using vehicle measures: Recent insights and future challenges,” *Journal of safety research*, vol. 40, no. 4, pp. 239–245, 2009.
- [9] P. M. Forsman, B. J. Vila, R. A. Short, C. G. Mott, and H. P. Van Dongen, “Efficient driver drowsiness detection at moderate levels of drowsiness,” *Accident Analysis & Prevention*, vol. 50, pp. 341–350, 2013.
- [10] X. Fan, B.-c. Yin, and Y. Sun, “Yawning detection based on gabor wavelets and lda,” *Journal of Beijing university of technology*, vol. 35, no. 3, pp. 409–413, 2009.
- [11] Z. Zhang and J. Zhang, “A new real-time eye tracking based on nonlinear unscented kalman filter for monitoring driver fatigue,” *Journal of Control Theory and Applications*, vol. 8, no. 2, pp. 181–188, 2010.
- [12] J. Wang, L. Zhang, D. Zhang, and K. Li, “An adaptive longitudinal driving assistance system based on driver characteristics,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 1–12, 2012.
- [13] A. Bolovinou, A. Amditis, F. Bellotti, and M. Tarkiainen, “Driving style recognition for co-operative driving: A survey,” in *The Sixth International Conference on Adaptive and Self-Adaptive Systems and Applications*, pp. 73–78, Citeseer, 2014.
- [14] C. M. Martinez, M. Heucke, F.-Y. Wang, B. Gao, and D. Cao, “Driving style recognition for intelligent vehicle control and advanced driver assistance: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 666–676, 2017.
- [15] M. Van Ly, S. Martin, and M. M. Trivedi, “Driver classification and driving style recognition using inertial sensors,” in *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1040–1045, IEEE, 2013.

-
- [16] D. A. Johnson and M. M. Trivedi, "Driving style recognition using a smart-phone as a sensor platform," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1609–1615, IEEE, 2011.
- [17] M. Fazeen, B. Gozick, R. Dantu, M. Bhukhiya, and M. C. González, "Safe driving using mobile phones," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1462–1468, 2012.
- [18] R. Stoichkov, "Android smartphone application for driving style recognition," *Department of Electrical Engineering and Information Technology Institute for Media Technology, July*, vol. 20, 2013.
- [19] V. Manzoni, A. Corti, P. De Luca, and S. M. Savaresi, "Driving style estimation via inertial measurements," in *13th International IEEE Conference on Intelligent Transportation Systems*, pp. 777–782, IEEE, 2010.
- [20] L. Xu, J. Hu, H. Jiang, and W. Meng, "Establishing style-oriented driver models by imitating human driving behaviors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2522–2530, 2015.
- [21] A. Doshi and M. M. Trivedi, "Examining the impact of driving style on the predictability and responsiveness of the driver: Real-world and simulator analysis," in *2010 IEEE Intelligent Vehicles Symposium*, pp. 232–237, IEEE, 2010.
- [22] W. Wang and J. Xi, "A rapid pattern-recognition method for driving styles using clustering-based support vector machines," in *2016 American Control Conference (ACC)*, pp. 5270–5275, IEEE, 2016.
- [23] Y. Zhang, W. C. Lin, and Y.-K. S. Chin, "A pattern-recognition approach for driving skill characterization," *IEEE transactions on intelligent transportation systems*, vol. 11, no. 4, pp. 905–916, 2010.
- [24] B. Higgs and M. Abbas, "Segmentation and clustering of car-following behavior: Recognition of driving patterns," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 81–90, 2014.

- [25] E. Tadesse, W. Sheng, and M. Liu, “Driver drowsiness detection through hmm based dynamic modeling,” in *2014 IEEE International conference on robotics and automation (ICRA)*, pp. 4003–4008, IEEE, 2014.
- [26] V. Gadepally, A. Krishnamurthy, and U. Ozguner, “A framework for estimating driver decisions near intersections,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 637–646, 2013.
- [27] D. Mitrovic, “Reliable method for driving events recognition,” *IEEE transactions on intelligent transportation systems*, vol. 6, no. 2, pp. 198–205, 2005.
- [28] M. Sundbom, P. Falcone, and J. Sjöberg, “Online driver behavior classification using probabilistic arx models,” in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 1107–1112, IEEE, 2013.
- [29] T. Akita, S. Inagaki, T. Suzuki, S. Hayakawa, and N. Tsuchida, “Hybrid system modeling of human driver in the vehicle following task,” in *SICE Annual Conference 2007*, pp. 1122–1127, IEEE, 2007.
- [30] S. Sekizawa, S. Inagaki, T. Suzuki, S. Hayakawa, N. Tsuchida, T. Tsuda, and H. Fujinami, “Modeling and recognition of driving behavior based on stochastic switched arx model,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 4, pp. 593–606, 2007.
- [31] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning hierarchical features for scene labeling,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1915–1929, 2012.
- [32] I. Sutskever, J. Martens, and G. E. Hinton, “Generating text with recurrent neural networks,” in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 1017–1024, 2011.
- [33] T. Mikolov, A. Joulin, S. Chopra, M. Mathieu, and M. Ranzato, “Learning longer memory in recurrent neural networks,” *arXiv preprint arXiv:1412.7753*, 2014.

-
- [34] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.
- [35] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, “Recurrent neural network based language model,” in *Eleventh annual conference of the international speech communication association*, 2010.
- [36] S. Hochreiter, “Untersuchungen zu dynamischen neuronalen netzen,” *Diploma, Technische Universität München*, vol. 91, no. 1, 1991.
- [37] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, “A field guide to dynamical recurrent neural networks,” in *chapter Gradient Flow in Recurrent Nets: The Difficulty of Learning Long-Term Dependencies*, pp. 237–243, Wiley-IEEE Press, 2001.
- [38] Y. Bengio, P. Simard, P. Frasconi, *et al.*, “Learning long-term dependencies with gradient descent is difficult,” *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.
- [39] K. J. Lang, A. H. Waibel, and G. E. Hinton, “A time-delay neural network architecture for isolated word recognition,” *Neural networks*, vol. 3, no. 1, pp. 23–43, 1990.
- [40] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, “Learning long-term dependencies in narx recurrent neural networks,” *IEEE Transactions on Neural Networks*, vol. 7, no. 6, pp. 1329–1338, 1996.
- [41] T. A. Plate, “Holographic recurrent networks,” in *Advances in neural information processing systems*, pp. 34–41, 1993.
- [42] M. C. Mozer, “Induction of multiscale temporal structure,” in *Advances in neural information processing systems*, pp. 275–282, 1992.
- [43] J. Schmidhuber, “Learning complex, extended sequences using the principle of history compression,” *Neural Computation*, vol. 4, no. 2, pp. 234–242, 1992.
- [44] Q. V. Le, N. Jaitly, and G. E. Hinton, “A simple way to initialize recurrent networks of rectified linear units,” *arXiv preprint arXiv:1504.00941*, 2015.

-
- [45] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [46] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with lstm,” 1999.
- [47] I. Carpatorea, *Methods to quantify and qualify truck driver performance*. PhD thesis, Halmstad University Press, 2017.
- [48] “Company, i., ipg truckmaker.” <https://ipg-automotive.com/products-services/simulation-software/truckmaker/>. Revision: 2019, Accessed: 2019-06-26.
- [49] T. Krotak and M. Simlova, “The analysis of the acceleration of the vehicle for assessing the condition of the driver,” in *2012 IEEE Intelligent Vehicles Symposium*, pp. 571–576, IEEE, 2012.
- [50] M. W. Hancock and B. Wright, “A policy on geometric design of highways and streets,” 2013.
- [51] T. A. Ranney, “Psychological factors that influence car-following and car-following model development,” *Transportation research part F: traffic psychology and behaviour*, vol. 2, no. 4, pp. 213–219, 1999.
- [52] S. Panwai and H. Dia, “Comparative evaluation of microscopic car-following behavior,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 3, pp. 314–325, 2005.
- [53] L. Evans and P. Wasielewski, “Risky driving related to driver and vehicle characteristics,” *Accident Analysis & Prevention*, vol. 15, no. 2, pp. 121–136, 1983.
- [54] J. Wu, M. Brackstone, and M. McDonald, “The validation of a microscopic simulation model: a methodological case study,” *Transportation Research Part C: Emerging Technologies*, vol. 11, no. 6, pp. 463–479, 2003.