CHALMERS
UNIVERSITY OF TECHNOLOGY

Design and verification of a tool to automate test procedures for autonomous emergency braking systems

Master’s thesis in Automotive Engineering

He Song

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2019

MASTER’S THESIS IN AUTOMOTIVE ENGINEERING

Design and verification of a tool to automate test procedures for autonomous emergency braking systems

He Song

Department of Mechanics and Maritime Sciences
Division of Vehicle Safety
Unit of Crash Analysis and Prevention
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2019

Design and verification of a tool to automate test procedures for autonomous emergency braking systems

He Song

Master's Thesis 2019:50
Department of Mechanics and Maritime Sciences
Division of Vehicle Safety
Unit of Crash Analysis and Prevention
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Cover:
Volvo truck at testing track.

Department of Mechanics and Maritime Sciences
Göteborg, Sweden 2019-06-11

Design and verification of a tool to automate test procedures for autonomous emergency braking systems

Master's thesis in Automotive Engineering
He Song
Department of Mechanics and Maritime Sciences
Division of Vehicle Safety
Unit of Crash Analysis and Prevention
Chalmers University of Technology

Abstract

A large number of road crashes are caused by late braking due to distraction, inattention, poor visibility, or other unexpected situations. It is difficult for most drivers to deal with such critical situations and react in time to avoid a crash. AEB systems are now widely used to help the driver avoid such accidents or at least reduce their severity. However, before AEB systems' introduction to the market, their performances need to be verified in test-track for certification purposes.

Within this thesis, a tool to automate test procedures for AEB systems was designed using a MATLAB GUI. The aim of this tool is to improve test efficiency by providing timely feedback to the test engineers about the outcome of a test. The waterfall model was used to design and verify the tool. The requirements for the tool were collected from the future users of the tool. The algorithm and interface design were then performed for the tool based on the collected requirements. After implementation of the code, the tool was verified at different levels to validate its effectiveness.

The results from the tool technical and acceptance tests show that the tool successfully fulfilled all the user requirements. The tool interface is user-friendly and the tool is also robust enough to evaluate testing results for various test cases.

Key words: Active safety, MATLAB GUI, Legal requirements, Track testing, Waterfall model, Software development, Software testing

Contents

Preface

This thesis work has been conducted as a partial requirement for the Master of Science degree in Automotive Engineering at Chalmers University of Technology, Gothenburg, Sweden. The project was carried out at Vehicle Dynamics Group, Volvo Group Trucks Technology in Gothenburg from February to June 2019.

This thesis was carried out under the guidance of Marco Dozza as an examiner. I would like to thank him for his technical support and valuable advices. I would also like to acknowledge my supervisor at Volvo Group Trucks Technology namely Henrik Lindh for his continuous guidance and support. All test data have been collected in AstaZero with the help of Jeferson Sestrem. I also discussed a lot with him about the requirements for the tool. He took part in user acceptance testing for the tool and gave me helpful suggestions. I really appreciate Jerferson Sestrem for his contribution to this thesis work. I am also grateful to all the colleagues in Vehicle Dynamics Group for their help.

Finally, I want to thank my family and friends for the support and help during my thesis project.

Göteborg June 2019-06-11

He Song

Notations

ABS	Anti-lock Braking System
ADAS	Advanced Driver Assistance System
AEB	Autonomous Emergency Braking
CAN	Controller Area Network
ESP	Electronic Stability Program
FCW	Front Collision Warning
GPS	Global Positioning System
GUI	Graphic User Interface
HGV	Heavy Goods Vehicle
HIL	Hardware-In-the-Loop
HMI	Human Machine Interface
LDW	Lane Departure Warning
LIDAR	Light Detection and Ranging
NCAP	New Car Assessment Programme
RADAR	Radio Detection and Ranging
RCP	Rapid Control Prototyping
SDLC	Software Development Life Cycle
SIL	Software-In-the-Loop
TTC	Time-To-Collision

1 Introduction

1.1 Background

Every year in Europe alone, more than 40000 casualties and 1.4 million injuries are caused by vehicle-related accidents (Lesemann et al., 2010 & Gietelink et al., 2006). Although advances in passive safety have made passenger cars ever safer, the safety potential of further improvements in passive safety features is limited. However, active safety systems like Anti-lock Braking System (ABS) and Electronic Stability Program (ESP) offer possibilities for improving driving comfort and traffic safety by assisting the driver in his driving task. Moreover, advanced driver assistance systems (ADASs) have the potential to significantly reduce the number of road accidents by assisting the driver recognizing and reacting to potentially dangerous traffic situations or even autonomously intervene. According to several surveys, ADASs can prevent up to 40% of traffic accidents, depending on the type of ADAS and the type of accident scenario (Gietelink et al., 2006).

The introduction of active safety systems has a significant impact on testing methods for vehicles: the testing procedures do not only require to bring the vehicle itself into a predefined driving state, but they also need to place the vehicle into a specific location on the road, or even other traffic members into a given relation to the vehicle under test (Schöner et al., 2009). Active safety systems are also expected to meet high requirements in terms of performance, reliability and safety. Therefore, they must be tested for the wide variety of complex traffic situations and conditions that the system should be able to recognize and handle (Gietelink et al., 2006).

In the automotive industry, 'V' diagram is widely used with a 'top-down' approach to design and a 'bottom-up' approach to validation, although in practice the process does not strictly follow all phases in this sequence and goes through several iteration loops. For active safety systems, various 'in-the-loop' simulation tools are increasingly used for verification (Gietelink et al., 2006). For example, TNO has developed a simulation and a testing environment that can be used for standardized test programs including PreScan and VeHIL. PreScan is a Software-in-the-Loop (SIL) environment whereas VeHIL is its Hardware-in-the-Loop (HIL) counterpart (Hendricks et al., 2010).

Although more and more virtual development methods are used for testing and verification of active safety systems, there is still a need to verify the sensor and the overall system performance finally in a real environment (Schöner et al., 2009). Since verification only confirms compliance with the specification, errors in the specification may result in a faulty product. It is therefore important to perform validation of the integrated system against its requirements, especially for certification purposes (Gietelink et al., 2006).

1.2 Aim

Active safety systems require a growing number of tests to validate their safety and performance. These tests are necessary for certification purposes and can be time-consuming, complex, and expensive. AEB is one of the most promising active safety systems to increase safety and its commercialization requires several tests in test-track to verify that the system complies with the legal regulations. The objective of this thesis work is to develop a tool that could assist tests of safety performances for AEB systems by automating the test procedures, thus improving the efficiency of test track testing and a prompt assessment of whether a test is passed or failed. In this thesis, the AEB system includes forward collision warning (FCW) system and will automatically brake if the driver doesn't respond to the FCW.

1.3 Limitations

This thesis focus on AEB system for heavy vehicles and the tool is designed based on legal requirements for moving target tests and stationary target tests. However, it can be easily extended with other legal requirements or internal requirements. Besides, other active safety systems like lane departure warning (LDW) system can also be implemented in the same way if there will be legal requirements for LDW system in the future.

1.4 Thesis outline

The introduction is followed by Chapter 2, literature review, where previous researches about AEB system including its development, benefits, regulations, requirement specifications, and test procedures are introduced. The waterfall model for developing and verifying the tool, requirements and design for the tool, the test equipment, and data collection are presented in Chapter 3. The designed tool and its verification are presented in Chapter 4. Finally, the discussion and conclusion are presented in Chapter 5 and 6 respectively.

2 Literature review

In this Chapter, research about AEB systems is presented. The Chapter starts with the definition of AEB system. Then the development of AEB system is introduced, including sensor fusion and control algorithm. Moreover, the study about benefits of AEB system is provided to verify its effectiveness. The regulations about AEB system show how they contribute to the improvement of AEB system step by step. Lastly, today's legal requirement specifications and test procedures for AEB system for high duty vehicles are presented, which will guide the design of the tool.

2.1 AEB system

Using technologies such as RADAR, camera and/or LIDAR to identify other vehicles and in many cases pedestrians or cyclists ahead of the vehicle, AEB system supports the driver e.g. with an audio, visual and/or haptic warning or automated braking if the driver does not respond in time to avoid or mitigate imminent crashes, saving countless lives, injuries and inconvenience (Den Camp et al., 2017 & Van Ratingen et al., 2016).

AEB systems were first introduced to avoid and mitigate rear-end car-to-car crashes. Currently they are also developed for vulnerable road users, such as pedestrians, cyclists and motorcyclists. Further developments in AEB may address more complex crash scenarios, such as cross-junction, head-on and reversing accidents (Euro NCAP, 2017).

AEB systems vary depending on different manufacturers, but usually they have three phases. In the first phase, the driver gets an audio and visual alert on their dashboard that a collision is likely to happen. If the drive doesn't start to brake, Phase 2 applies pressure to the anti-lock braking system. That way, the driver quickly gets some assistance when they do hit the brake. If the driver still hasn't taken action, the system goes into Phase 3 by applying brakes in an attempt to avoid a potential collision (Nowak, 2018).

2.2 Development of AEB system

2.2.1 Sensor fusion

Most of today's vehicles use radar and vision sensors for AEB system to identify other objects ahead of the subject vehicle. No sensor type can work well for all the tasks in all conditions. Therefore, sensor fusion will be essential to provide redundancy for development of AEB system, thus improving the ability of the system to accurately detect an object.

Lee, Yi et al. (2011) presented their AEB system control algorithm which consists of two parts: obstacle detection part and main controller part. In the obstacle detection part, front obstacle information was measured and collected by using the vision sensor and radar sensor. The main controller is composed of two control stages: upper and lower level controller. The upper level controller decides the control mode while the lower level controller determines warning level and braking level to maintain the longitudinal safety.

Lee, Kim et al. (2012) presented an integrated driving path estimation algorithm for AEB system using multi-sensor fusion. The path prediction is first based on vehicle states and vision data. For application to dynamic maneuvering situation, they introduced the driving mode index which allows a detection of the driver maneuver intention. In accordance with the driving mode, the driving paths from vehicle dynamics and vision sensor are fused into ultimate driving path. The proposed algorithm can detect the driver intention and provide reliable path prediction in dynamic maneuver situation such as lane change and entering a curve, which can enhance the capabilities of AEB system.

Lee, Shin, Kwon (2017) used three different types of sensors including radar, camera, and LIDAR in their research. They present an optimized sensor fusion strategy and decision-making algorithm for AEB pedestrian. The designed AEB-pedestrian system is tested using a vehicle equipped with an AEB system and dummy moving system. The test results show that the proposed multi-sensor fusion system provides robust and reliable track management. The performance of the AEB system is enhanced by using a braking model to predict the collision avoidance time and by designing the system activation zone according to the relative speed and possible distance required to stop for pedestrians.

2.2.2 Control strategy

In general, the AEB system employ a safety indicator, for example Time-To-Collision (TTC), to measure the potential danger of impact into obstacles. Once the TTC is smaller than desired threshold, the AEB system would activate the braking system. Many researches about improving the performance of AEB system have been performed since it was introduced. They developed advanced control strategies by taking road slope, friction and driver behavior into account, designing more developed control algorithm and so on.

Han et al. (2014) developed an AEB strategy based on road friction considering that the conventional strategy is designed with a constant road friction. They use a combined-slip tire model to estimate peak road friction, which is then used to obtain braking threshold of TTC. The adaptive TTC threshold is another brake criterion added into the AEB control strategy, which makes AEB adapt to different road surfaces and active more adequately. The simulation results show that the proposed control strategy has better performance, especially for the medium road friction situation where collision could be avoided. Kim, Lee and Yi (2015) present another friction coefficient estimation algorithm which consists of two parts: a dynamics based maximum friction coefficient estimation algorithm, and an updating process based on the effects of states (slip ratio, vehicle speed and load of each tire). It has been shown that the proposed friction coefficient estimation algorithm could enhance the performance of AEB system algorithm.

Kim, Shin et al. (2018) proposed an AEB control algorithm to compensate for the effects of the slope and the friction of road. The configuration of the proposed AEB system is illustrated in Figure 2.1 where the AEB is trigged in an adaptive manner based on road conditions. In particular, the minimum stopping distance is adjusted considering the road friction coefficient and slope angle. The simulation results demonstrated that the proposed AEB is very effective in sloped and low friction road. The experiment tests conducted with a passenger car where the proposed algorithm is embedded in the RCP (Rapid Control Prototyping) unit show that the proposed AEB system has robust collision avoidance performance for various speeds on the slope road.

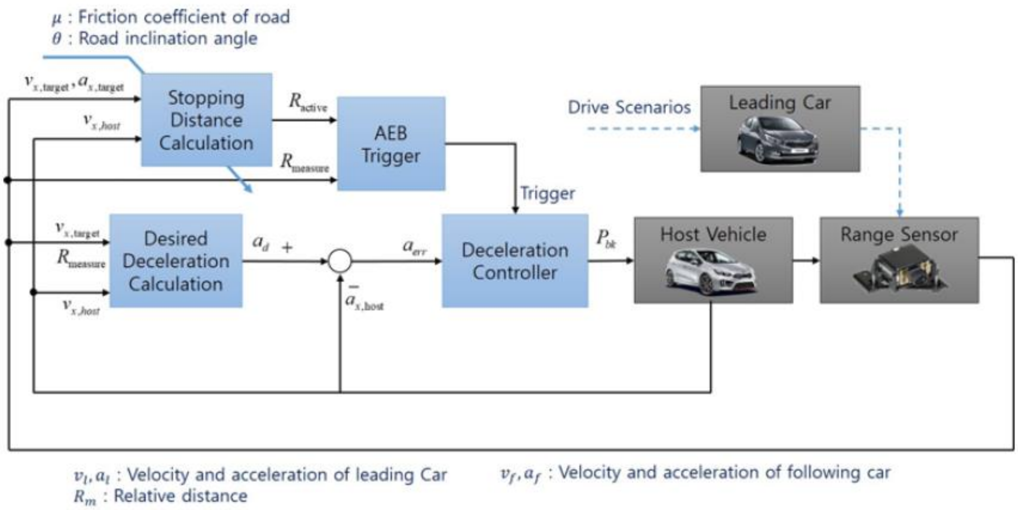

Figure 2.1 Configuration of the proposed AEB system (Kim, Shin et al., 2018).

Guo et al. (2014) presented two emergency braking controllers in the paper. The braking controller for deceleration was designed base on sliding mode while the other one for emergency was designed based on single neuron PID method. The effectiveness of the proposed controller was verified by co-simulations between CarSim and Simulink.

Zhang et al. (2017) proposed a novel AEB system based on nonlinear model predictive algorithm to optimize the active safety and vehicle handling comfort. A hierarchical control system is first designed to decouple and coordinate the driver-vehicle-surroundings interaction system as a decision-making system of AEB system. Based on a coordinated cost function of tracking safety, comfort, and fuel economy, a multi-objective optimization controller is then designed using the theory of non-linear model predictive control.

Koglbauer et al. (2018) investigated drivers’ evaluation of a conventional AEB system and an adaptive one based on road friction. Ninety-six drivers assigned to 5 age groups drove with AEB in the simulator. They evaluated the AEB’s braking actions in response to an imminent rear-end collision at an intersection for both high-friction and low-friction roads. The results show that the reported potential benefits of AEB can be further improved by including road friction in an adaptive AEB braking strategy. Besides, drivers’ subjective safety and trust were significantly improved when driving with the adaptive AEB compared to the conventional AEB.

Duan et al. (2017) extracted the top three scenarios of vehicle-bicycle conflicts in China from naturalistic driving datasets. These three scenarios were reconstructed in a driving simulator to investigate Chinese drivers’ braking behavior. Based on the results, an adaptive Bicyclist-AEB system was proposed, which has the potential to advance the AEB intervention timing adaptively without annoying drivers.

Brännström et al. (2008) presented a situation assessment algorithm that estimates driver distraction by continuously assessing the steering actions of the driver. A collision avoidance system was then proposed, which combined the situation assessment with a threat assessment algorithm that estimates the effort needed to avoid a collision. The test results showed that the situation assessment algorithm enables earlier interventions when the driver is assessed as being distracted without causing a significant increase of false interventions.

2.3 Benefits of AEB system

Euro NCAP studies show that 90% of traffic accidents are caused by driver's inattention or distraction and AEB system can avoid about 27% of traffic accidents (Zhao et al., 2017). A study by EURO NCAP and Australasian NCAP concluded that AEB system lead to a 38% reduction in real-world rear-end crashes at low speeds. According to estimates by the European Commission, AEB system could save more than 1,000 lives every year within the EU (UNECE, 2019).

Fildes et al., (2015) evaluated the effectiveness of low speed AEB based on real-world crash experience from six different countries. Their findings showed 38 percent overall reduction in rear-end crashes for vehicles fitted with low-speed AEB compared to a comparison sample of equivalent vehicles. There was no statistical evidence of any difference in effect between urban (≤60 km/h) and rural (>60 km/h) speed zones.

Rosén et al. (2010) studied the potential effectiveness of a pedestrian injury mitigation system by autonomous braking. The database from the German In-depth Accident Study (GIDAS) was queried for pedestrians hit by the front of cars from 1999 to 2007. It was found that the effectiveness at reducing fatally and severely injured pedestrians in frontal collisions with cars reached 40% and 27% respectively at a field of view of 40°.

Cicchino (2016) evaluated the effectiveness of FCW alone, low-speed AEB, and FCW with AEB in reducing front-to-rear crashes and injuries. He found that rear-end striking crash involvement rates reduced by 27%, 43%, 50% respectively by FCW alone, low-speed AEB, and FCW with AEB. Rates of involved injuries were reduced by 20%, 45% and 56% respectively.

Strandroth et al. (2012) studied potential benefits of AEB on front crashes between heavy goods vehicles (HGV) and passenger cars in reducing injury risk. Results showed that AEB activated on HGV and passenger cars in frontal collisions could possibly reduce the closing velocity by approximately 30 km/h on average, which would result in a 73% reduction of moderate and severe injuries (MAIS2+) on the passenger car occupants.

2.4 Regulations about AEB system

The availability and quality of vehicle safety is determined by a combination of internal and national regulation, consumer information as well as specific initiatives by the car manufacturing industry. Legislation aims for a minimum but high level of protection across the product line; consumer information aims to encourage the highest possible levels of safety performance based on state-of-the-art testing and protocols; and car industry policies increasingly promote safety as a marketable commodity (European Road Safety Observatory, 2016).

2.4.1 Legislation

The World Forum for Harmonization of Vehicle Regulations (WP.29) is a permanent working party in the institutional framework of the United Nations with a specific mandate and rules of procedure, which works as a global forum offering a unique framework for globally harmonized regulations on vehicles. The Working Party on Braking and Running Gear was the subsidiary body of WP.29 that prepares regulatory proposals on vehicle automation, ADAS, active safety, braking and running matters to WP.29 (UNECE, 2019).

The regulation (EC) No 661/2009 of 13 July 2009 foresees mandatory fitting of AEB systems and LDW Systems on heavy-duty vehicles. For all EU heavy commercial vehicles, AEB system have to be equipped from 1 Nov 2013 for new types of vehicle and 1 Nov 2015 for all new vehicles (European Road Safety Observatory, 2016).

A new draft United Nations Regulation for AEB system for cars and light commercial vehicles has been agreed by 40 countries. The new UN Regulation will impose strict and internationally harmonized requirements for the use of AEB system at low speeds, even in complex and unpredictable situation such as traffic in urban areas. The Regulation sets out test requirements for the deployment of AEB system at a range of different speeds. It was expected that the new regulation would enter into force in early 2020 (UNECE, 2019).

2.4.2 Consumer information

Consumer information provides prospective car buyers with factual information about the safety performance of cars in accidents and encourages manufacturers to introduce evidence-based safety designs beyond those required by legislative norms. There are currently nine similar but not identical New Car Assessment Programmes (NCAP) in different areas to provide reliable, comprehensive and timely consumer information on the safety of new cars (European Road Safety Observatory, 2016).

The European New Car Assessment Programme (Euro NCAP) is a European car safety performance assessment programme founded in 1997. Twenty years on, 9 out of 10 cars sold on the European market hold a Euro NCAP rating and over 630 safety rating were published, some 1,800 cars crash-tested and over 160 million Euro was collectively spent to make cars safer. It was estimated that more than 78,000 lives have been saved since the first results were presented in February 1997 (Euro NCAP, 2018).

In 2014, a big step forward is taken by Euro NCAP when adding crash avoidance systems such as Autonomous Emergency Braking and Lane Keep Assist/Lane Departure Warning tests to the overall star rating. Euro NCAP expand its safety rating by including AEB technology for pedestrians from 2016 and broaden their tests on AEB technology to include bicyclist crash scenarios in 2018, considering that vulnerable road users, such as pedestrians, cyclists and motorcyclists, account for almost half of Europe’s total road fatalities and bicyclists’ deaths are on the rise in many countries (Euro NCAP, 2018).

According to Euro NCAP 2025 Roadmap, they expect AEB technology to continue to evolve in the years ahead and more and more manufacturers are adding additional sensors and combining multiple sensor types together in “fusion” to offer the potential to address new and more complex crash scenarios. Euro NCAP has identified three priority areas where the rating scheme will be updated to reflect the progress in industry: back-over or reversing crashes (2020), crossing and tuning maneuvers (2020), and head-on scenarios (2022) (Euro NCAP, 2017).

2.4.3 Industry policies

In recent years, safety has been increasingly marketed by car manufacturers and they have introduced different vehicle safety measures without legislation, in advance of legislation or in response to consumer information programmes. For example, the Volvo Group has set a highly ambitious goal and states that ‘Our ultimate goal is zero accidents with Volvo Group products (European Road Safety Observatory, 2016).
The European industry associations include the European Car Manufacturers Association ACEA, ACEM (motorcycle industry) and the IRU (truck and bus industry). Different companies come together to coordinate proposals and make road safety pledges (European Road Safety Observatory, 2016).

2.5 Legal requirements for AEB system

The legal requirement specifications and test procedures introduced in this and next section are cited from regulation No. 131-01 (UNECE, 2014) of Economic Commission for Europe of the United Nations.

2.5.1 Performance requirements

The performance requirements stipulate how the AEB system should perform in different situations.

1. The AEB system shall provide the driver with appropriate warnings (including collision warning, failure warning and deactivation warning) according to different scenarios:

    1) A collision warning when the system has detected the possibility of a collision with a proceeding vehicle in the same lane which is travelling at a slow speed, has slowed to a halt or is stationary. The warning should be as specified in the 1st requirement of Section 2.5.4.

    2) A failure warning when there is a failure in the AEB system that prevents the requirements of this Regulation being met. The warning shall be as specified in the 4th requirement of Section 2.5.4 below. There shall not be an appreciable time interval between each AEB system self-check. Subsequently there shall not be an appreciable delay in illuminating the warning signal when there is an electrically detectable failure.

    3) A deactivation warning when the vehicle is equipped with a means to manually deactivate the AEB system and the system is deactivated. This warning shall be as specified in the 2nd requirement of Section 2.5.3 below.

2. The AEB system shall have an emergency braking phase with the purpose of significantly decreasing the speed of the subject vehicle. This shall be tested in accordance with Section 2.6.2 and 2.6.3.

3. The AEB system shall be active at least within the speed range of 15 km/h up to the maximum design speed of the vehicle, and at all vehicle load conditions, unless it is manually deactivated as Section 2.5.3 below.

4. The AEB system shall be designed to minimize the generation of collision warning signals and to avoid autonomous braking in situations where the driver would not recognize an impending forward collision. This shall be tested in accordance with Section 2.6.6 (UNECE, 2014).

2.5.2 Interruption by the driver

The AEB system should provide the driver means to interrupt all the different phases:

1. The AEB system shall provide the driver the means to interrupt the collision warning phase. However, when a vehicle braking system is used to provide a haptic warning, the system shall provide a means for the driver to interrupt the warning braking.

2. The AEB system shall provide the means for the driver to interrupt the emergency braking phase.

3. In both cases above, the interruption may be initiated by any positive action (e.g. kick-down, operating the direction indicator control) that indicates that the driver is aware of the emergency situation. A list of these positive actions should be provided to the Technical Service at the time of type approval and annexed to the test report (UNECE, 2014).

2.5.3 Function deactivation

When the driver is provided with a means to deactivate the AEB system, the following conditions shall apply:

1. The AEB system function shall be automatically reinstated at the initiation of each new ignition cycle.

2. A constant optical warning signal shall inform the driver that the AEB system function has been deactivated. The yellow warning signal specified in 4th requirement of Section 2.5.4 below may be used for this purpose (UNECE, 2014).

2.5.4 Warning indication

The requirements of warning indication are listed, including the types, sequence, timing and so on:

1. The collision warnings mentioned in the first requirement of Section 2.5.1 shall include at least two different modes selected from acoustic, haptic or optical.

The timing of the collision warning signals shall be such that they provide the possibility for the driver to react to the risk of collision and take control of the situation, and shall also avoid nuisance by too early or too frequent warnings. This shall be demonstrated in accordance with the provisions of 2nd requirements of Section 2.6.2 and 2.6.3.

2. A description of the warning indication and the sequence in which the collision warning signals are presented to the driver shall be provided by the vehicle manufacturer at the time of type approval and recorded in the test report.

3. Where an optical warning is used as part of the collision warning, the optical signal may be the flashing of the failure warning signal specified in the 4th requirement of Section 2.5.4 below.

4. The failure warning referred to in 1st requirement of Section 2.5.1 above shall be a constant yellow optical warning signal.

5. Each optical warning signal shall be activated either when the ignition (start) switch is turned to the "on" (run) position or when the ignition (start) switch is in a position between the "on" (run) and "start" that is designated by the manufacturer as a check position (initial system (power-on)). This requirement does not apply to warning signals shown in a common space.

6. The optical warning signals shall be visible even by daylight, the satisfactory condition of the signals must be easily verifiable by the driver from the driver's seat.

7. When the driver is provided with an optical warning signal to indicate that the AEB system is temporarily not available, the signal shall be constant and yellow in color. The failure warning signal specified in 4th requirement of Section 2.5.4 above may be used for this purpose (UNECE, 2014).

2.5.5 Provisions for the periodic technical inspection

For a periodic technical inspection, the AEB system should apply to the following conditions:

It shall be possible to confirm the correct operational status of the AEB system by a visible observation of the failure warning signal status, following a "power-ON" and any bulb check. If the failure warning signal is in a common space, the common space must be observed to be functional prior to the failure warning signal status check.

At the time of type approval, the means to protect against simple unauthorized modification of the operation of the failure warning signal chosen by the manufacturer shall be confidentially outlined. Alternatively, this protection requirement is fulfilled when a secondary means of checking the correct operational status of the AEB system is available (UNECE, 2014).

2.6 Test procedures for AEB system

2.6.1 Test conditions

The test shall be performed at following conditions:

- The test shall be performed on a flat, dry concrete or asphalt surface affording good adhesion.
- The ambient temperature shall be between 0 and 45 ℃.
- The horizontal visibility range shall allow the target vehicle to be observed throughout the test.
- The test shall be performed when there is no wind liable to affect the results.
- The vehicle shall be tested in a condition of load to be agreed between the manufacturer and the Technical Service. No alteration shall be made once the test procedure has begun (UNECE, 2014).

2.6.2 Warning and activation test with a stationary target

The vehicle shall be tested with a stationary target. The test and performance requirements are as follows:

1. The subject vehicle shall approach the stationary target in a straight line for at least two seconds before the functional part of the test with a subject vehicle to target centerline offset of not more than 0.5 m.

   The functional part of the test shall start when the subject vehicle is travelling at a speed of 80 ± 2 km/h and is at a distance of at least 120 m from the target.

   From the start of the functional part until the end of test there shall be no adjustment to any control of the subject vehicle by the driver other than slight adjustments to the steering control to counteract any drifting.

2. The timing for the collision warning modes referred to in 1st requirement of Section 2.5.4 above shall comply with the following:

   1) At least one warning mode shall be provided no later than the time specified in Table 2.1, Column B.

      In the case of the vehicles referred to first category in Table 2.1, the warning shall be haptic or acoustic.

      In the case of the vehicles referred to second category in Table 2.1, the warning shall be haptic, acoustic or optical.

   2) At least two warning modes shall be provided no later than the time specified in Table 2.1, Column C.

3) Any speed reduction during the warning phase, shall not exceed either 15 km/h or 30% of the total subject vehicle speed reduction, whichever is higher.

Table 2.1 Warning and activation test requirements – pass/fail values (UNECE, 2014).

A	B	C	D	E	F	G	H
Vehicle category	Stationary target			Moving target			
	Timing of warning modes		Speed reduction	Timing of warning modes		Speed reduction	Target speed
	At least 1	At least 2		At leas 1	At least 2		
M_3, $N_2 > 8\ t$ and N_3	Not later than 1.4 s before the start of emergency braking phase	Not later than 0.8 s before the start of emergency braking phase	Not less than 20 Km/h	Not later than 1.4 s before the start of emergency braking phase	Not later than 0.8 s before the start of emergency braking phase	No impact	12±2 km/h
$N_2 \leq 8\ t$ and M_2	Not later than 0.8 s before the start of emergency braking phase	Before the start of the emergency braking phase	Not less than 10 Km/h	Not later than 0.8 s before the start of emergency braking phase	Before the start of the emergency braking phase	No impact	67±2 km/h

3. The collision warning phase shall be followed by the emergency braking phase.

4. The total speed reduction of the subject vehicle at the time of the impact with the stationary target shall be not less than the value specified in Table 2.1, Column D.

5. The emergency braking phase shall not start before a TTC equal to or less than 3.0 s. Compliance shall be verified by either actual measurement during the test or using documentation provided by the vehicle manufacturer, as agreed between the Technical Service and the manufacturer (UNECE, 2014).

2.6.3 Warning and activation test with a moving target

The vehicle shall be tested with a moving target. The test and performance requirements are as follows:

1. The subject vehicle and the moving target shall travel in a straight line in the same direction, for at least two seconds prior to the functional part of the test, with a subject vehicle to target centerline offset of not more than 0.5 m.

   The functional part of the test shall start with the subject vehicle travelling at a speed of 80 ± 2 km/h, the moving target at speed of the value specified in Table 2.1, Column H, and a separation distance of at least 120 m between them.

From the start of the functional part of the test until the subject vehicle comes to a speed equal to that of the target there shall be no adjustment to any subject vehicle control by the driver other than slight steering adjustments to counteract any drifting.

2. The timing for the collision warning modes referred to in 1st requirement of Section 2.5.4 above shall comply with as follows:

    1) At least one haptic or acoustic warning mode shall be provided no later than the time specified in Table 2.1, Column E.

    2) At least two warning modes shall be provided no later than the time specified in Table 2.1, Column F.

    3) Any speed reduction during the warning phase shall not exceed either 15 km/h or 30% of the total subject vehicle speed reduction, whichever is higher.

3. The emergency braking phase shall result in the subject vehicle not impacting the moving target.

4. The emergency braking phase shall not start before a TTC equal to or less than 3.0 s. Compliance shall be verified by either actual measurement during the test or using documentation provided by the vehicle manufacturer, as agreed between the Technical Service and the manufacturer (UNECE, 2014).

2.6.4 Failure detection test

Simulate an electrical failure of the AEB system, for example by disconnecting the power source to any AEB system component or disconnecting any electrical connection between AEB system components. When simulating such a failure, neither the electrical connections for the driver warning signal of 4th requirement of Section 2.5.4 above nor the optional manual AEB system deactivation control of 4th requirement of Section 2.5.4 shall be disconnected.

The failure warning mentioned in the 4th requirement of Section 2.5.4 above shall be activated and remain activated not later than 10 s after the vehicle has been driven at a speed greater than 15 km/h. The warning signal shall also be reactivated immediately after a subsequent ignition "off" ignition "on" cycle with the vehicle stationary as long as the simulated failure exists (UNECE, 2014).

2.6.5 Deactivation test

Turn the ignition (start) switch of the vehicle to the "on" (run) position and deactivate the AEB system. The warning signal mentioned in the 2nd requirement in Section 2.5.4 above shall be activated.

Turn the ignition (start) switch to the "off" position. Again, turn the ignition (start) switch to the "on" (run) position and verify that the previously activated warning signal is not reactivated, thereby indicating that the AEB system has been reinstated as specified in the 1st requirement of Section 2.5.4 above. If the ignition system is activated by means of a "key", the above requirement shall be fulfilled without removing the key (UNECE, 2014).

2.6.6 False reaction test

The false reaction test requires another two stationary vehicles. The requirements for them and the subject vehicle are as follows:

1. Two stationary vehicles, of Category M1 AA saloon, shall be positioned:

    1) To face in the same travel direction as the subject vehicle

    2) With a distance of 4.5 m between them

    3) With the rear of each vehicle aligned with the other

2. The subject vehicle shall travel for a distance of at least 60 m, and at a constant speed of 50 ± 2 km/h to pass centrally between the two stationary vehicles.

    During the test there shall be no adjustment of any subject vehicle control other than slight steering adjustments to counteract any drifting.

3. The AEB system shall not provide a collision warning signal and shall not initiate the emergency braking phase (UNECE, 2014).

3 Methodology

To develop and verify the tool which can evaluate the test results for AEB system, the waterfall software development life cycle (SDLC) was used. The Waterfall SDLC model is a sequential software development process, which comprises five phases: requirements, design, implementation, testing, and maintenance. This thesis will follow the phases step by step to design and verify the tool.

In this chapter, the waterfall SDLC model is first introduced. Its first two phases for developing the tool are then presented. Before entering the code implementation stage, test data for AEB system is required. Therefore, the test equipment is introduced here, followed by the data collection.

3.1 The waterfall model

The waterfall SDLC model was first proposed by Benington in 1956. It was modified with feedback loops by Royce in 1970 so that each preceding stage could be revisited (Ruparelia N. B., 2010). The waterfall model defines several consecutive phases as shown in Figure 3.1.

The requirements phase is a complete and comprehensive description of the behavior of the software to be developed. Both functional and non-functional requirements need to be collected from the users. The functional requirements include such requirements as purpose, scope, perspective, software attributes, user characteristics, functionalities specifications, interface requirements and so on. In contrast, the non-functional requirements include constraints, limitations, and requirements imposed on the design and operation of the software including reliability, scalability, testability, availability, maintainability and so on (Bassil Y., 2012).

The design phase is the process of planning and problem solving for a software solution. It needs to bring down whole knowledge of requirements for the software design. It implicates software developers and designers to define the plan for a solution which includes algorithm design, software architecture design, logical diagram design, concept design, graphical user interface design, data structure definition and so on (Bassil Y., 2012 & Sharma M. K., 2017).

The implementation phase is also known as programming or coding phase. It refers to the realization of requirements and design specifications into a concrete executable program, database, website, or software component through programming and deployment. This phase is where the real code is written in the suitable programming language. In other words, it is the process of converting the whole requirements and blueprints into a production environment (Bassil Y., 2012 & Sharma M. K., 2017).

The testing phase is also known as verification and validation which is a process for checking that a software solution meets the original requirements and specifications and that it accomplishes its intended purpose. In fact, verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase, while validation is the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. Moreover, the testing phase is to remove errors or mistakes, bugs to make it error free good quality software product (Bassil Y., 2012 & Sharma M. K., 2017).

The maintenance phase is the process of modifying a software solution after delivery and deployment to refine output, correct errors, and improve performance and quality. Additional maintenance activities can be performed in this phase including adapting software to its environment, adding new feature to existing software, and increasing software reliability (Bassil Y., 2012).

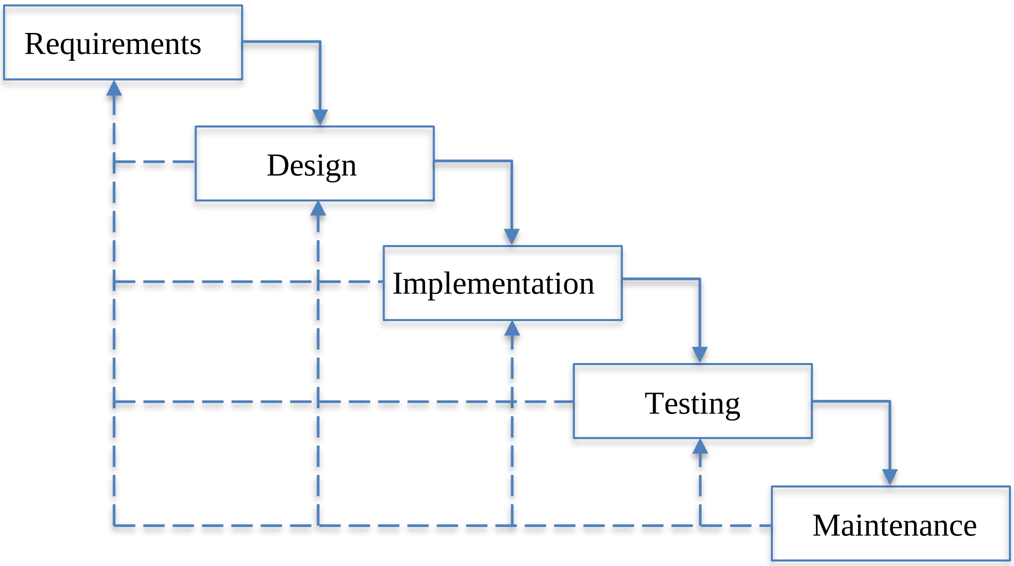

Figure 3.1 The waterfall model with feedback loops (Bassil Y., 2012).

3.2 Requirements

3.2.1 Functional requirements

The functional requirements are defined by means of use cases which describes users' interactions with the tool. The use case of the tool to be developed include the following contents.

3.2.1.1 Brief description

The tool is designed to evaluate the testing results of AEB system, i.e. whether the test is passed or failed based on the legal requirements. It can be used in two test scenarios: stationary target test and moving target test. Besides, the tool can be used to plot all the signals and present videos or sound signal.

3.2.1.2 Actors

An actor is a person or system involved with the use case. Here in this use case, the actor is AEB system test engineer.

3.2.1.3 Preconditions

The test data should be acquired before the evaluation. The data format should be .mat, which can be accessed with MATLAB. Besides, the data should also be placed in the right folder to get right results. The required signals to develop the tool include:

- Status_sv

The DGPS status of the subject vehicle

- Status_tg1
The DGPS status of the target vehicle

- Head_Up_display
Visual warning signal

- Left_Loud_Speaker
Audio warning signal from left loud speaker

- Right_Loud_Speaker
Audio warning signal from right loud speaker

- CM_status
Brake status signal from driver assistance control unit

- FrontalCollisionDriverAlert
Warning status signal from driver assistance control unit

- Speed_kmh
Subject vehicle course over ground speed

- Spd_tg1_kmh
Target vehicle course over ground speed

- LngRsv_tg1
Longitudinal distance between the subject and target vehicle measured in the direction of the subject vehicle heading

- LatRsv_tg1
Lateral distance between the subject and target vehicle measured at right angles to the subject vehicle heading

- T2Csv_tg1
Time to collision with target vehicle derived from LngRsv_tg1 and LngSsv_tg1 (where LngSsv_tg1 is the speed between the subject and target vehicle in the longitudinal direction, with respect to subject vehicle heading)

- AmbientAirTemperature
The ambient air temperature when performing tests

- CollSituationHazard
The status for request of hazard light

3.2.1.4 Basic flow

The basic flow is the normal course of events or what happens most of the time from the start to the end of use case.

This use case begins when an actor wants to evaluate the test results for stationary and moving target tests of AEB system. The basic flow is listed below:

- The test engineer loads the data first, all the information of the test is presented and he or she may check the information is correct or not

- The actor can evaluate the test results if all the information is correct, the tool will present immediately that if the test is passed or failed and which requirements are not fulfilled

- The actor can check all the signals one by one for more information in detail, the corresponding plot will be presented

- The actor can also load other types of data, like videos or audio signals. The tool shall display them

3.2.1.5 Alternative flow

An alternative flow is a variation from the basic flow, it can be an error or an unexpected condition. When an exception is encountered, it prevents the basic flow from directly presenting the results until it is addressed. The alternative flow in this use case includes:

- In the first step of basic flow above, if the actor finds out that the wrong file is loaded, he or she may load another file again

- In the second step of basic flow above, if part of the data is missing or other mistakes lead to failure of estimation, the tool may notify the actor and the use case end

- In the last step of basic flow above, if the wrong file is loaded, the actor may load another one again

3.2.1.6 Schematic diagram

The schematic diagram in Figure 3.2 shows both the basic flow and alternative flow. The tool should include mainly three functions: load data, run evaluation and present signals.

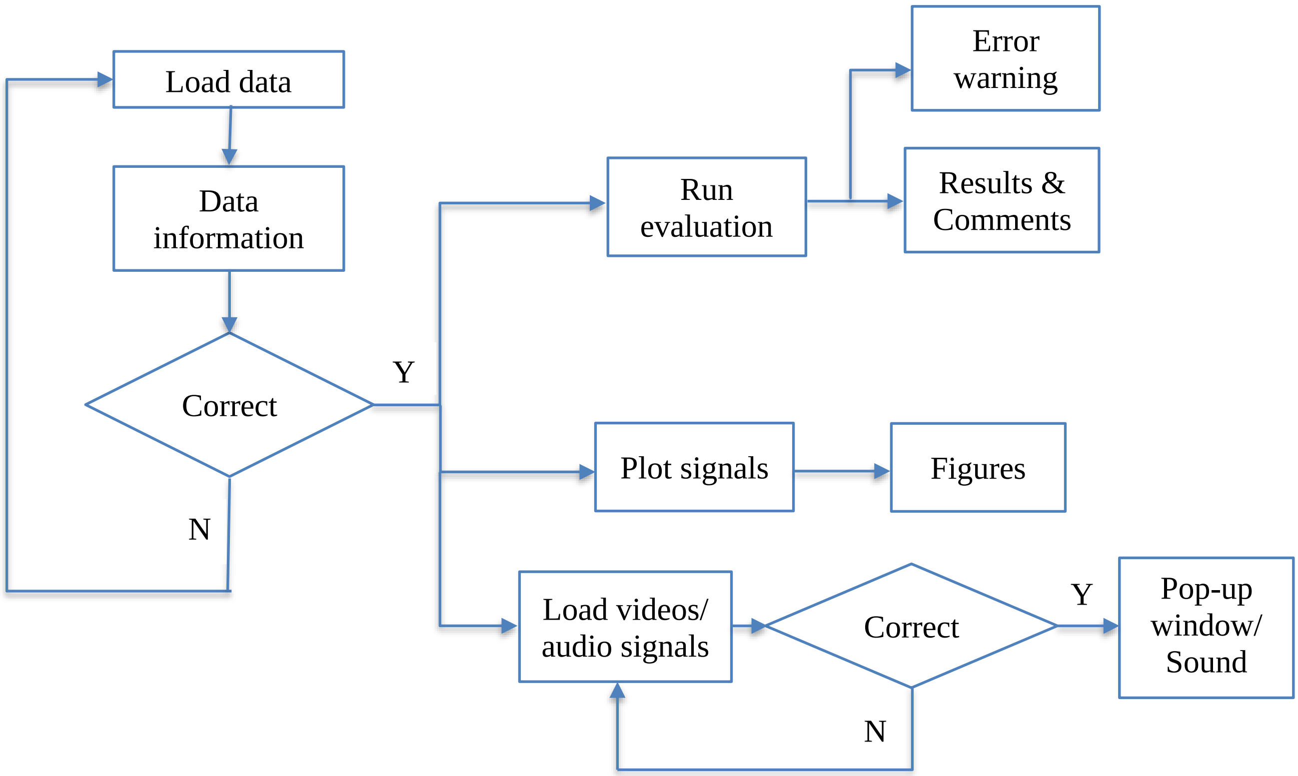

Figure 3.2 The schematic diagram of basic flow and alternative flow.

3.2.1.7 HMI requirements

The tool shall be designed to be user-friendly, for example including instructions about how to use it. Besides, the presentation about data information and test results should be precise and clear for the users.

- Data information
  When presenting the information of the loaded file, the tool shall present the path and the name of the file and especially indicate its test scenario. They are used for the actor to check if it is the intended one.

- Test results
  After running the evaluation, the tool should present a green sign for pass and a red sign for failure. At the same time, the tool shall present comments showing why the test is passed or failed. The unfulfilled requirements should be presented different with the fulfilled requirements.

3.2.1.8 Software

The tool is designed with MATLAB GUI (Graphical User Interface), which is a very friendly development platform for design of graphical user interface.

3.2.2 Non-functional requirements

3.2.2.1 Reliability

The tool should be designed to be robust, which means effectiveness of the tool should be valid not only for these passed tests but also for the failed tests. The tool should not present “pass” signals when the test is actually failed and vice versa. If there are some unexpected errors about the test data, the tool should present the users warnings.

3.2.2.2 Scalability

The tool may further need to be extended by including more signals and more test scenarios or adding new features such as evaluation for LDW system, which should also be taken into consideration when developing the tool.

3.3 Design

3.3.1 Algorithm design

The algorithm design is based on the requirement specifications and test procedures of AEB system for heavy-duty vehicles, which are presented in Section 2.5 and 2.6. Since the tool is designed for moving target test and stationary target test, the test requirements and performance requirements of AEB system for these two tests are analyzed carefully. The algorithm design of the tool for moving target test and stationary target test are then concluded as follows.

3.3.1.1 Moving target test

- To evaluate the test results, the start and end point of the test should be determined first. The start point of the test is when the distance between subject vehicle and target vehicle is 120 m (LngRsv_tg1 = 120). The end point is when the request for hazard light changes from activated to deactivated (CollSituationHazard = 1 at the last moment). If there is no start point or end point, it is impossible to evaluate the test result and a warning message should be displayed to the user.

- The tool should present a “Pass” sign if the following conditions are fulfilled, otherwise a “Fail” sign should be given

    a) The temperature should be between 0 and 45 degrees Celsius from the start point to the end point (0 < AmbientAirTemperature < 45)

    b) The VBOX should have 2 cm precision from the start point to the end point which means it has fixed solution status (Status_sv = 4 & Status_tg1 = 4)

    c) The subject vehicle should keep a constant speed from the start point to the brake point (Speed_kmh: 80 ± 2 km/h)

    d) The subject vehicle should keep aligned with the target vehicle from start point to the end point (-0.5 < LatRref_tg1 < 0.5)

    e) The target vehicle should keep a constant speed from the start point to the end point (Spd_tg1_kmh: 12 ±2 km/h)

    f) Collision between subject vehicle and target vehicle is avoided (-0.05 < LngRscv_tg1 < 120)

g) Warning shall not be later than 1.4 s before full brake. For visual warning, the time when the head up display is activated (Head_Up_Display > 1.6 V) should be 1.4 s earlier than the time when the subject vehicle starts to full brake (CM_Status = 3). For audio warning, the time when the loud speakers are activated (Left_Lound_Speaker, Left_Lound_Speaker < -0.1 V) should be 1.4 s earlier than the time when the subject vehicle starts to full brake

h) Pre brake shall not brake more than 30% of the total deceleration or more than 15 km/h. The speed difference shall not be greater than 15 (Speed_kmh change < 15 km/h) between the subject vehicle starts to pre brake (CM_Status = 2) and starts to full brake (CM_Status = 3)

i) Full brake shall not start earlier than 3 s TTC (time to collision). (T2Csv_tg1 < 3 when the subject vehicle starts to full brake)

3.3.1.2 Stationary target test

- The tool should decide the start and end point for stationary target test. The start point of the test is when the distance between subject vehicle and target vehicle is 120 m (LngRsv_tg1 = 120). The end point is when the request for hazard light changes from activated to deactivated (CollSituationHazard = 1 at the last moment). Just the same for moving target test, the tool should warn the user if there is no start point or end point for stationary target test.

- The tool should present a “Pass” sign if the following conditions are fulfilled, otherwise a “Fail” sign should be displayed

a) The temperature should be between 0 and 45 degrees Celsius from the start point to the end point (0 < AmbientAirTemperature< 45)

b) The VBOX should have 2 cm precision from the start point to the end point which means DGPS has fixed solution status. Different from moving target test, the target vehicle here is stationary, there is only one DGPS for subject vehicle (Status_sv = 4)

c) The subject vehicle should keep a constant speed from the start point to the brake point (Speed_kmh: 80 ± 2 km/h)

d) The subject vehicle should keep aligned with the target vehicle from start point to the end point (-0.5 < LatRref_tg1< 0.5)

e) The target vehicle should keep stationary from the start point to the end point (Spd_tg1_kmh: 0 km/h)

f) Subject vehicle speed should decrease at least 20 km/h at impact (Speed_kph < 60 km/h at impact) or the collision is avoided (-0.05< LngRscv_tg1< 120)

g) Warning shall not be later than 1.4 s before full brake. For visual warning, the time when the head up display is activated

(Head_Up_Display > 1.6 V) should be 1.4 s earlier than the time when the subject vehicle starts to full brake (CM_Status = 3). For audio warning, the time when the loud speakers are activated (Left_Lound_Speaker, Left_Lound_Speaker < –0.1 V) should be 1.4 s earlier than the time when the subject vehicle starts to full brake

h) Pre brake shall not brake more than 30% of the total deceleration or more than 15 km/h. The speed difference shall not be greater than 15 (Speed_kmh change < 15 km/h) between the subject vehicle starts to pre brake (CM_Status = 2) and starts to full brake (CM_Status = 3)

i) Full brake shall not start earlier than 3 s TTC (time to collision). (TTC < 3 s when the subject vehicle starts to full brake)

3.3.2 HMI design

As is mentioned above about the requirement for software, the HMI design for the tool is based on MATLAB GUI. In the MATLAB GUI development environment called GUIDE, the graphical user interface can be designed typically by first creating a figure and populating it with components from a graphic layout editor. These components are called uicontrol objects. An associated code file containing callback functions for these components will also be created. One can program each object via its callback function to perform the action you intend it to do when a user activates the component (Espinosa H. G. et al., 2013).

The layout editor is shown in Figure 3.3. The uicontrol objects are located at its left side, including buttons, sliders, texts, axes and so on. One can select uicontrol objects and arrange them in the right area. When laying out the uicontrol objects are finished, press the “run” button and GUIDE automatically generates two files: a .fig file and an .m file. The .fig file can be modified by editing the GUI in the layout editor again. The .m file contains functions that control the callbacks, one can implement the designed algorithms to realize the operation you want. Besides, it also contains another two functions called opening function and output function. The opening function creates the interface before the user has any action to the components. The output function returns output to the command line.

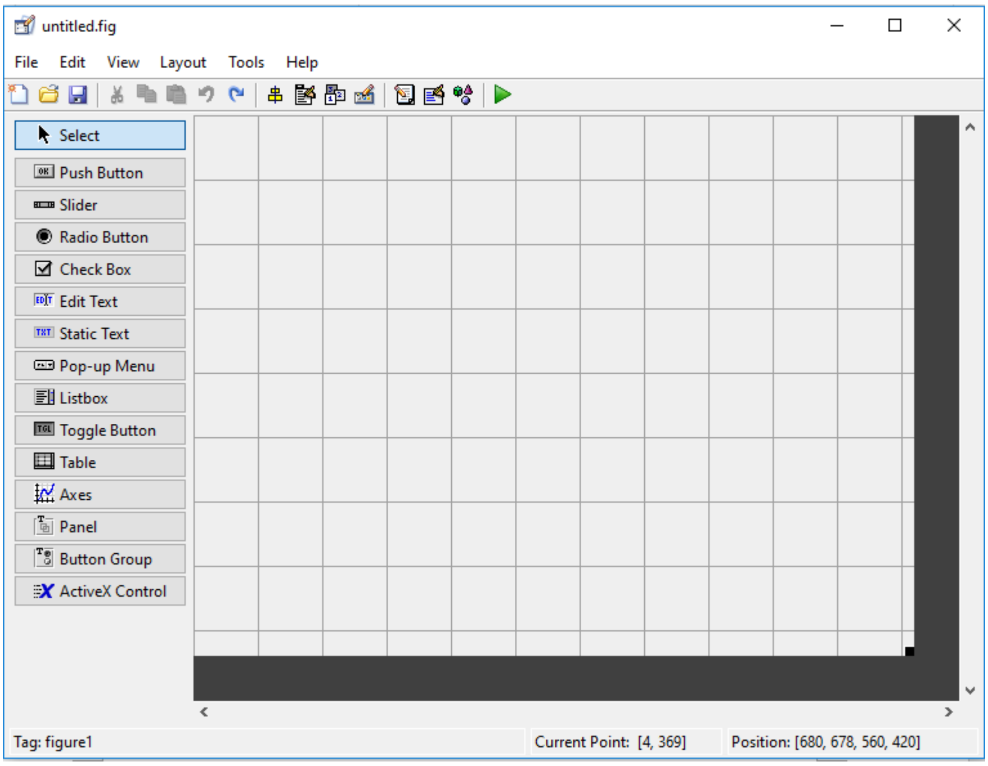

Figure 3.3 The layout editor.

3.4 Test equipment

The set-up diagram of the test equipment is shown below, where the M-Log is the main unit. A log file is recorded by pressing the logger button. To the M-Log a V-Box is connected providing the GPS position. Two USB cameras and a microphone are also connected to the M-Log. The M-sense connected to the M-Log was used as an A/D converter to get the analog input from both speakers and the head-up display. Besides, the traffic data from the CAN-buses on the truck was logged.

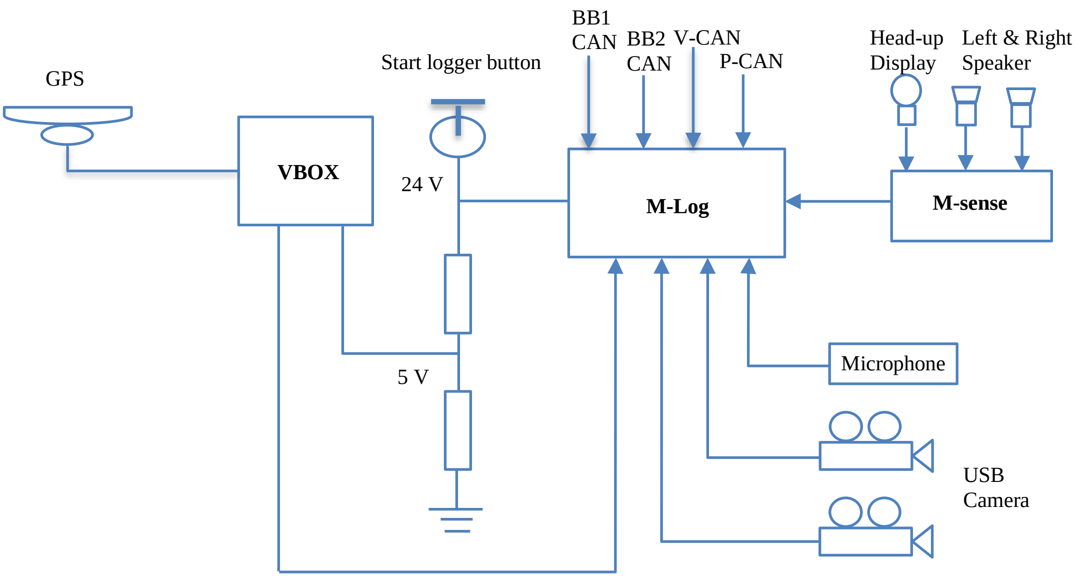

Figure 3.4 The set-up diagram of the test equipment.

3.4.1 M-Log

The M-Log is from IPEtronic, a manufacturer of mobile measurement technology in the automotive industry. The M-Log is a high-performance device for data acquisition, the data logged with the M-Logger includes:

- GPS position provided by VBOX
- CAN-channels from the truck
- Videos from USB cameras, one monitoring the road ahead and the other filming the instrument cluster
- The sound of warning signals recorded by a microphone
- The analog input of warning signals from head-up display and speakers via a M-sense

Figure 3.5 The M-Log (IPETRONIK, 2019).

3.4.2 VBOX

The VBOX from RACELOGIC is shown in Figure 3.6, which is designed for tests where positional accuracy is of key importance. To get a secure truth of the position on the target vehicle for AEB system test, a VBOX unit was used together with a D-GPS to get an accuracy of 2 cm. The V-Box needs to be calibrated for a certificate, which is only valid for one year.

Figure 3.6 The VBOX (RACELOGIC, 2019).

3.4.3 Balloon car

The target vehicle for moving target test is a balloon car on top of a moving rig. The rig is a metal frame with a mounting for GPS antenna in front. The balloon rig is towed in a rope which is approximately 12 m long. The target vehicle for the stationary case is balloon car almost the same as the moving but with small differences: no rig underneath and standing on plastic blocks. The balloon cars used for moving and stationary target test are shown below.

Figure 3.7 The balloon car for moving target test.

Figure 3.8 The balloon car for stationary test.

3.5 Data collection

After all the data is logged with M-Log, connect an USB stick to the M-Log and one can get the data. When the data is transferred from M-Log to the USB stick, there will be a red and blue blinking light on the M-Log. The light turns to be red after all the data is transferred.

The data is stored in a MEA-zip file, extract the file and the data of interest is just in that folder. It includes 1 audio signal (.wav), 2 video signal (.avi) and 1 data file (.bin) for each test. The .bin file needs to be converted to .blf format with IPE Converter, then the data can be open with CANalyzer.

Set up for CAN channels in CANalyzer, the data can then be used for analysis. The interface for analysis in CANalyzer is shown in Figure 3.9. At the left side, you can choose the signals you want from all the logged data. The figure below contains all the required signals for the tool design, which are listed in Section 3.2.1.3. The corresponding plots for these signals are presented at the right side.

All the required signals are then converted to .mat format in CANalyzer, which can be further accessed by the tool to evaluate the test result. With the data of .mat format, the requirements and design for the tool introduced above, the codes of the tool were then implemented, i.e. the third phase of the waterfall model.

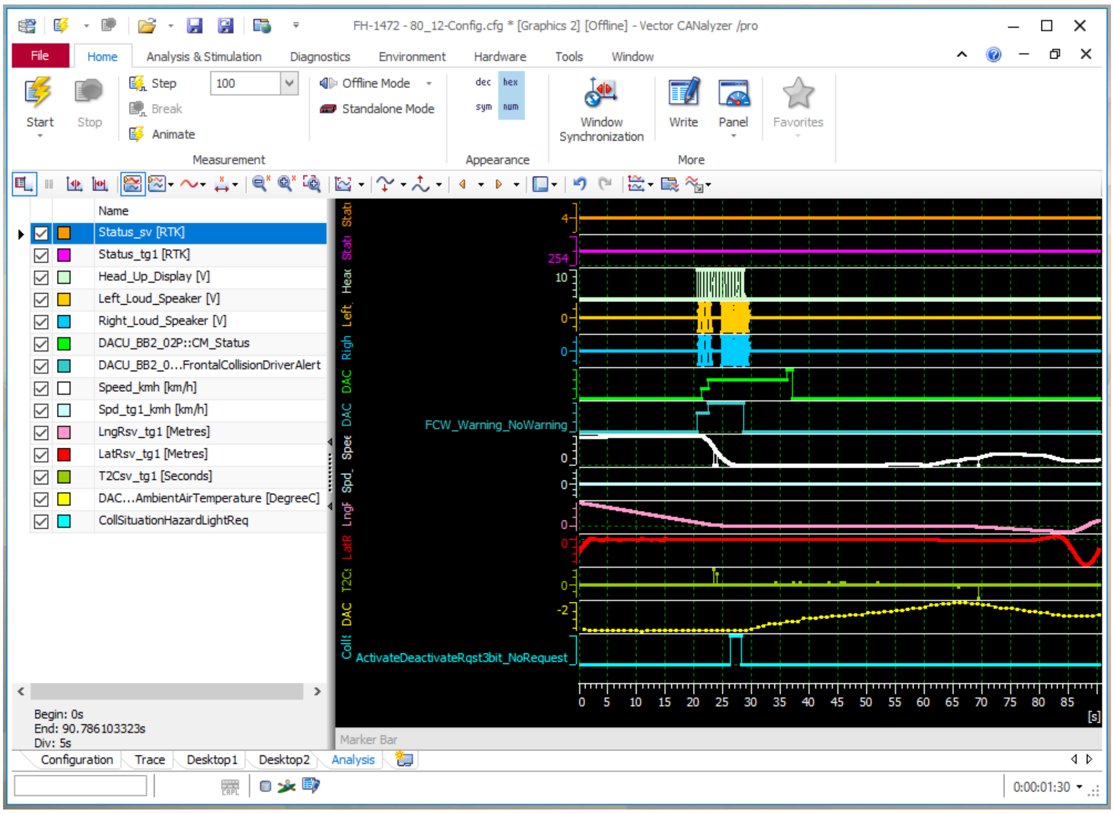

Figure 3.9 The analysis interface in CANalyzer.

4 Results and verification

This chapter starts with the graphic user interface designed using MATLAB GUI. A brief introduction about how to use the tool is presented. The testing of the tool was then performed, including unit testing, system testing and user acceptance testing.

4.1 Graphic user interface

The graphic user interface of the designed tool is presented in Figure 4.1. It can be divided into three parts.

The first part at the top of the interface has the function of loading the data file and displaying corresponding information about the data file. When pushing the button "Load", one can choose the data file needed to be tested. The information about the data file will be displayed immediately including full path name, test scenario, and ID. Only when the data file is located in the right folder (stationary target test or moving target test), the test result can then be evaluated.

The second part at the middle of the interface has the function of evaluating the test results and displaying passed and failed requirements. After loading the data file correctly, one can push the "Run" button to evaluate the test results. The test is failed if the test result displays a red "Fail" sign while it is passed with a green "Pass" sign. The test results in detail presents which requirements are fulfilled and which are not. The unfulfilled requirements are displayed with a "*" sign ahead of the text.

The third part at the bottom of the interface has the function of plotting signals and displaying videos or audio signals. All the signals will be listed at the bottom left corner when the data file is loaded. Click on the signal, the corresponding plot will be displayed. When clicking on videos or audio signal, one need to load the correct file first. For videos, there will be a pop-up widow to display the video. For audio signal, it will be plotted at the bottom right corner. There are three push buttons called "Start", "Pause", and "Resume", which can be used to control the audio signal.

At the top right corner, there are a logo "AEBS Testing Tool" and two push buttons "Help" and "About". The "Help" button will pop up a window showing how to use this tool when being pushed. The "About" button will pop up a window showing the ownership of this tool.

There are also some small tools at the top left toolbar. They include "zoom in", "zoom out", "cursor", and "pan", which can be useful for the analysis of the signal plots.

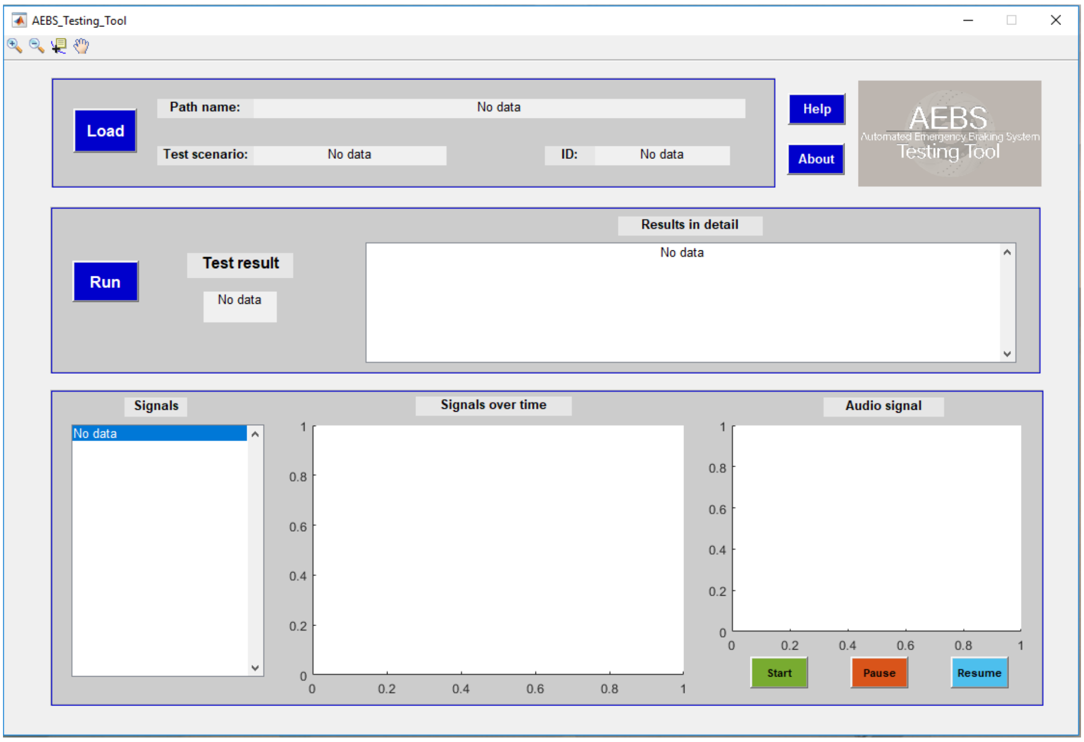

Figure 4.1 The graphic user interface of the tool.

4.2 Unit testing

The unit testing of the tool means that each function of the tool will be tested separately. Three main functions including load data, evaluate results, display signals are tested one by one to check whether they work properly.

4.2.1 Load data

One data file from moving target test and one from stationary target test are chosen to test the function of loading data. The graphic user interfaces after loading these two files are shown in Figure 4.2 and 4.3 respectively. They both can show correct information about the loaded data file, which indicates that the function of loading data works well.

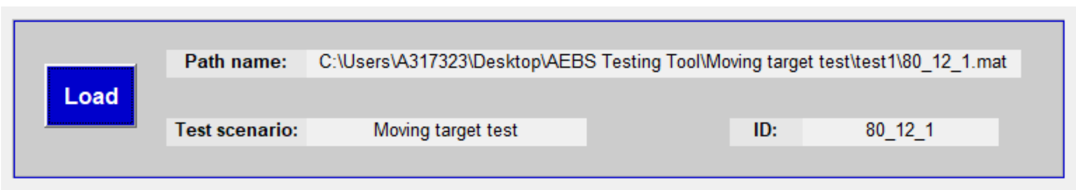

Figure 4.2 The graphic user interface for loading moving target test data.

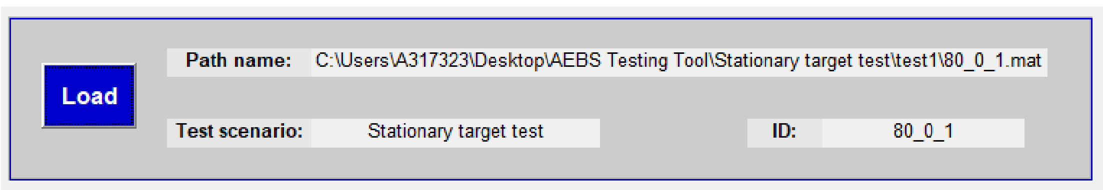

Figure 4.3 The graphic user interface for loading stationary target test data.

4.2.2 Evaluate results

After running the evaluation, the tool should present a green sign for pass and a red sign for failure. At the same time, the tool shall present comments showing why the test is passed or failed.

Test results from a moving target test are shown in Figure 4.4. The test is failed since the tool display a red “Fail” sign. The reason for the failure can be found in “Results in detail”. It is that “The subject vehicle does not keep aligned with the target vehicle” since the text is displayed with a “*” sign. A testing for evaluating stationary target test was also performed. The test is passed with the green “Pass” sign. One can see that there is no text with a “*” sign, which means all the requirements for the test are fulfilled. Notably, only part of the detailed results is shown for both figures. The slider on the right can help see the entire text.

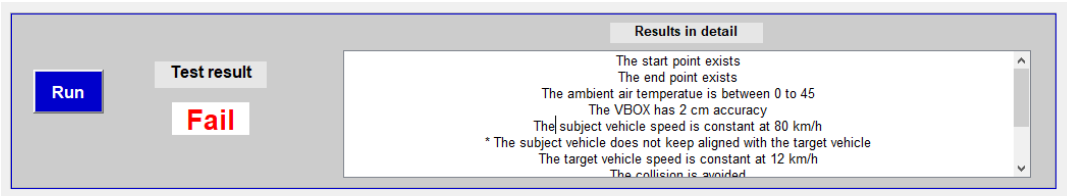

Figure 4.4 The graphic user interface for evaluating moving target test.

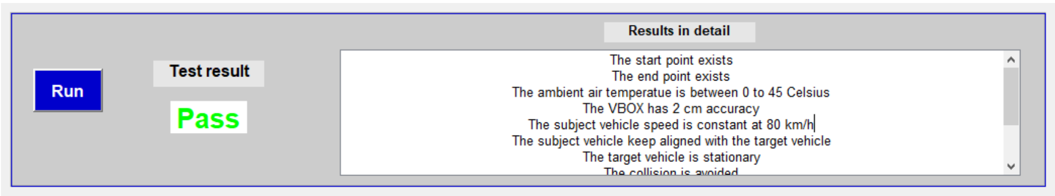

Figure 4.5 The graphic user interface for evaluating stationary target test.

4.2.3 Display signals

The tool also has the function of displaying different signals. For normal signals listed above, its plot will be displayed when being chosen. Figure 4.6 shows the plot for lateral range between subject and target vehicle. It also includes the plot for the audio signal, but one can hear the sound only when pressing the “Start” button. The sound can also be stopped or continued by “Pause” and “Resume” button respectively. The pop-up window for playing videos is presented in Figure 4.7. It is actually a movie player, which is quite user-friendly to operate on videos.

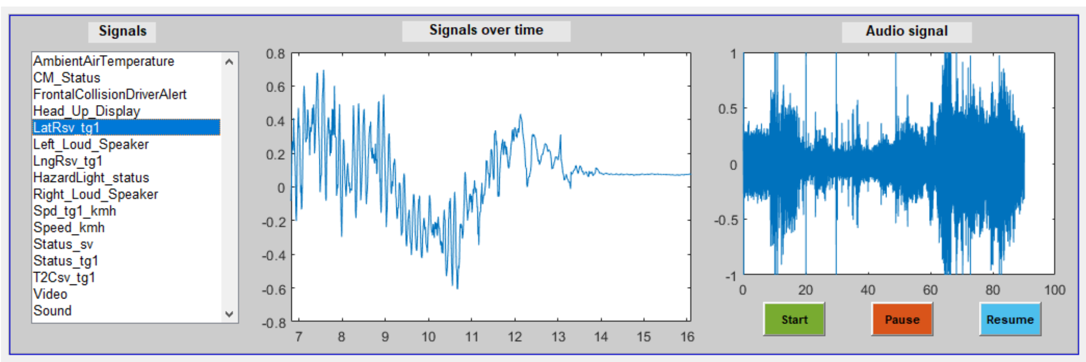

Figure 4.6 The graphic user interface for plotting signals.

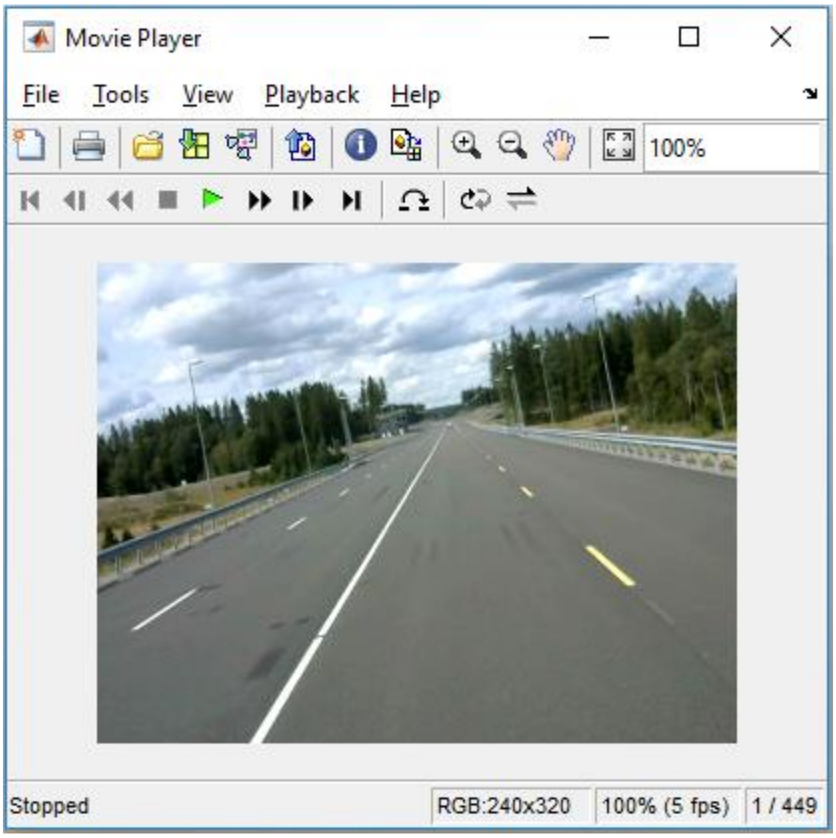

Figure 4.7 The pop-up window for displaying videos.

4.3 System testing

System testing is done to verify the tool meets its requirements and ensure that the tool works well in all different cases. Five test cases chosen from moving and stationary target test respectively are used to test the effectiveness of the tool.

4.3.1 Moving target test cases

The first moving target test case has only 7 signals due to data loss. The test result for this case was evaluated with the tool developed in this thesis. After running the evaluation, the test result displays a red “Fail” sign which means the test is failed. There is also a pop-up window showing the user that the data is lost. The graphic user interface for this case is shown in Figure 4.8.

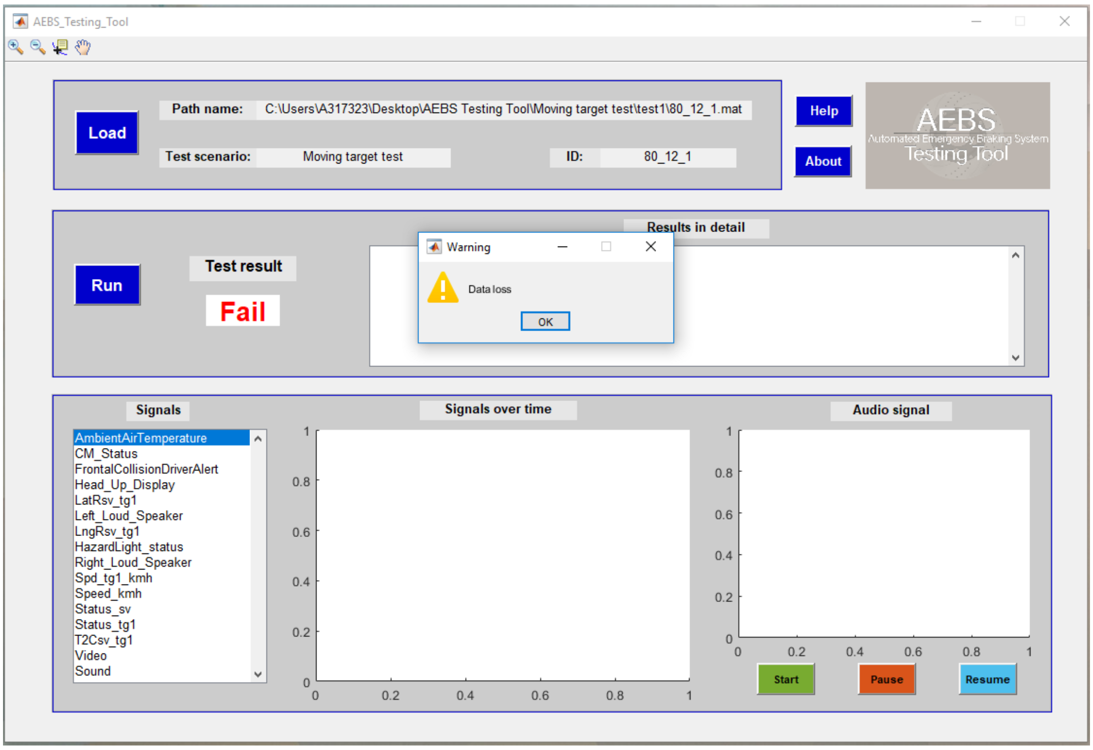

Figure 4.8 The graphic user interface for moving target test case 1.

The test result for the second moving target test case is also failed according to its graphic user interface shown in Figure 4.9. As the pop-up window shows, the reason is that there is no start point or end point, thus failing to continue evaluation. The start point actually exists for this case. However, the end point cannot be found since the request for hazard light is not activated. The status of hazard light request is shown in Figure 4.10, where it is zero for the whole test process. This test case shows that the tool can successfully detect that the end point is missing.

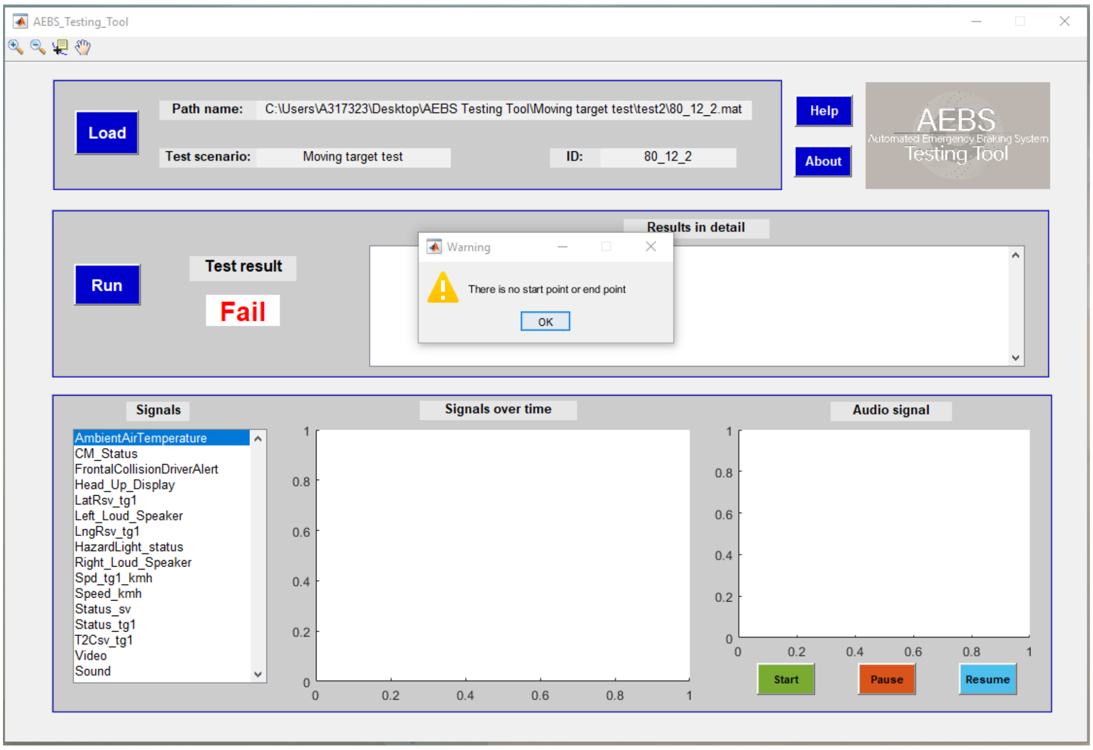

Figure 4.9 The graphic user interface for moving target test case 2

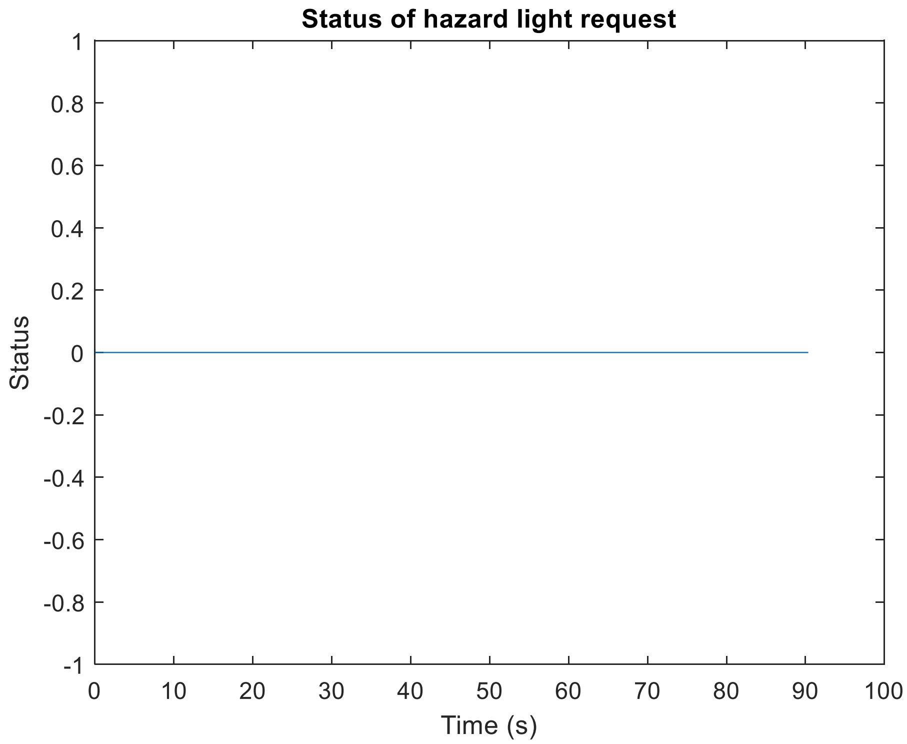

Figure 4.10   Status of hazard light request.

The graphic user interface for the third moving target test is shown in Figure 4.11. The test is failed because the subject vehicle does not keep aligned with the target vehicle. The lateral distance between subject and target vehicle is also plotted in Figure 4.11, which is not in the required range from -0.5 to 0.5 m. Therefore, the test result evaluated with the tool is correct and the unfulfilled requirement is also presented precisely.

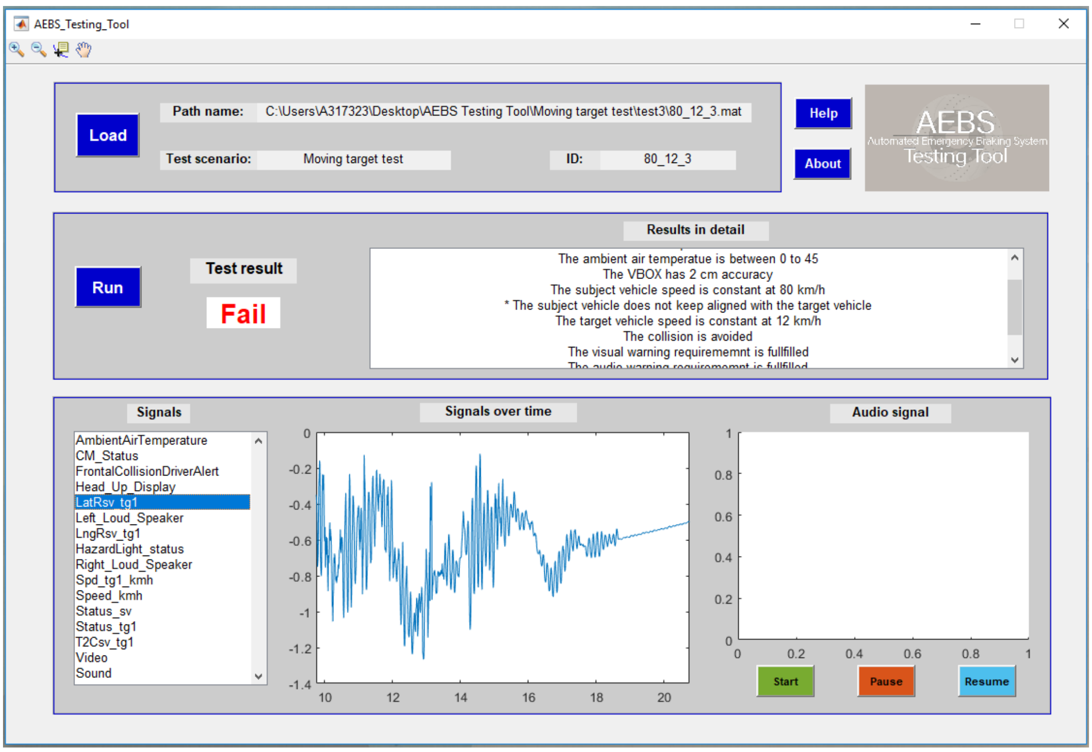

Figure 4.11   The graphic user interface for moving target test case 3.

The test result for the fourth moving target test case is failed as shown in the graphic user interface in Figure 4.12. There are several unfulfilled requirements for this test case:

- The VBOX does not have the required 2 cm accuracy
- The lateral distance is beyond the required range, just the same as the last case
- The target vehicle does not keep at 12 km/h during the test process
- The collision is not avoided, which means the longitudinal distance is smaller than 0

The longitudinal distance is plotted in Figure 4.12, the point (14.21, -4.188) is also marked in the plot with the cursor tool. Since there is at least one point out of range, the requirement for avoiding the collision is obviously unfulfilled in this case. If one double check other unfulfilled requirements, they are not fulfilled just as the tool shows.

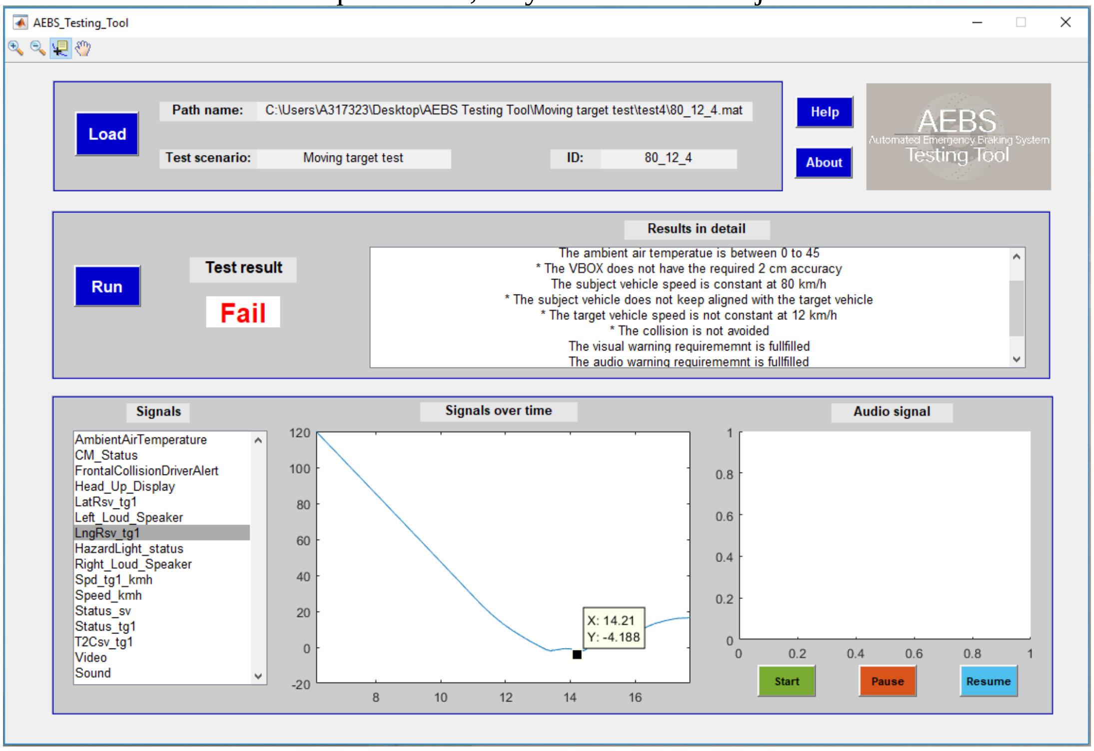

Figure 4.12 The graphic user interface for moving target test case 4.

The graphic user interface for the last moving target test is presented in Figure 4.13. The test is failed due to the following reasons:

- The subject vehicle speed is not constant at 80 km/h
- The subject vehicle does not keep aligned with the target vehicle
- The visual warning requirement is not fulfilled
- The audio warning requirement is not fulfilled

- The speed declines more than 15 km/h in prebrake phase
- Full brake starts earlier than 3 s TTC

Actually, the first reason is the main cause of other unfulfilled requirements in this case. The speed of the subject vehicle is presented in Figure 4.13. The subject vehicle was driven towards the target vehicle at the speed of around 50 km/h, which is far away from the required 80 km/h. Therefore, the performance requirements for AEB system are all unfulfilled, including requirements for visual and audio warning, speed decline and full brake timing.

Figure 4.13 The graphic user interface for moving target test case 5.

4.3.2 Stationary target test cases

Just the same as the first moving target test case, the first stationary target test also has only 7 signals due to data loss. The graphic user interface after running the evaluation for this case is shown in Figure 4.14. The red “Fail” sign means that the test is failed. The pop-up window presents the user that the data is lost in this test case.

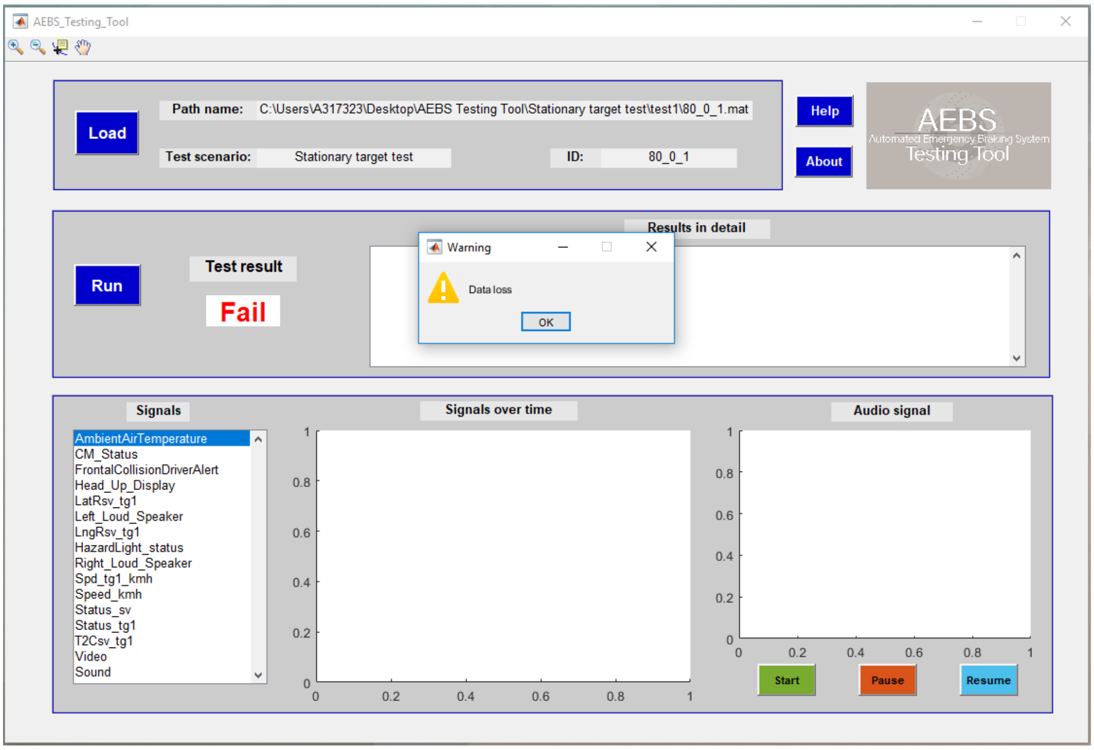

Figure 4.14 The graphic user interface for stationary target test case 1.

The second case for stationary target test is failed due to no start point or end point. When one check the signals, the hazard light request is not activated. Therefore, the end point is missing for this case. If the start point or end point does not exist for the test case, the evaluation cannot be continued. Instead the tool will pop up a warning window showing that there is no start point or end point, which is also presented in Figure 4.15.

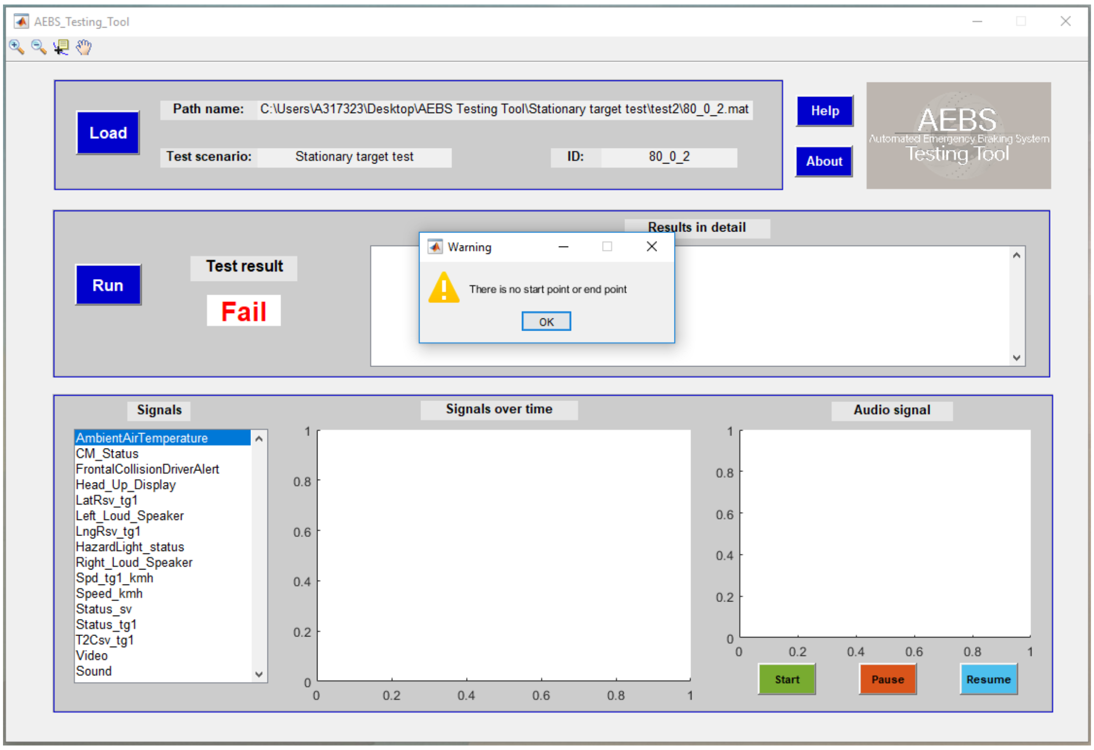

Figure 4.15 The graphic user interface for stationary target test case 2.

Figure 4.16 presents the graphic user interface after running the third stationary target test. The test is failed because

- The ambient air temperature is not between 0 to 45 ℃
- The subject vehicle speed is not constant at 80 km/h
- The subject vehicle does not keep aligned with the target vehicle

The signal of ambient air temperature is plotted in Figure 4.16. The temperature is below 0 ℃, which is beyond the required range. The other requirements are also failed just as the tool displays.

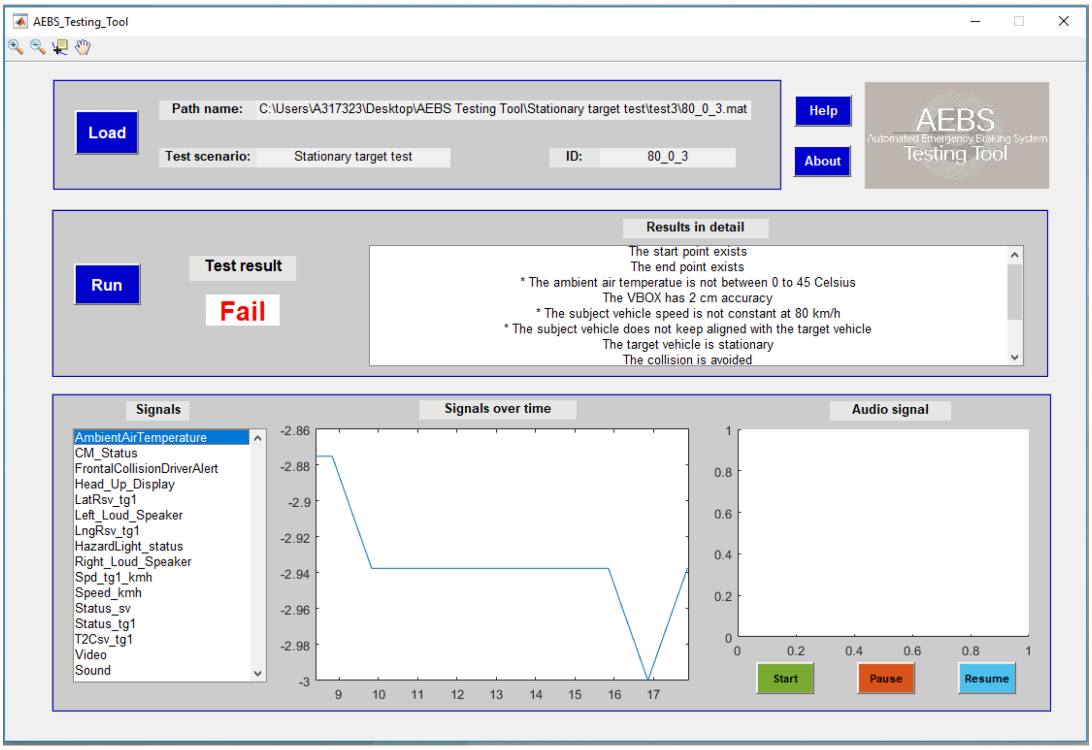

Figure 4.16 The graphic user interface for stationary target test case 3.

The fourth stationary target test is failed. Its graphic user interface after running evaluation is displayed in Figure 4.17, where also presents the unfulfilled requirements:

- The VBOX does not have the required 2 cm accuracy
- The subject vehicle speed is not constant at 80 km/h
- The subject vehicle does not keep aligned with the target vehicle

The signal of DGPS status for subject vehicle is plotted in Figure 4.17. It does not keep at 4 in the process, which means the accuracy of VBOX does not fulfill the requirement. In fact, the other requirements are also unfulfilled. The tool displays all the unfulfilled requirements successfully.

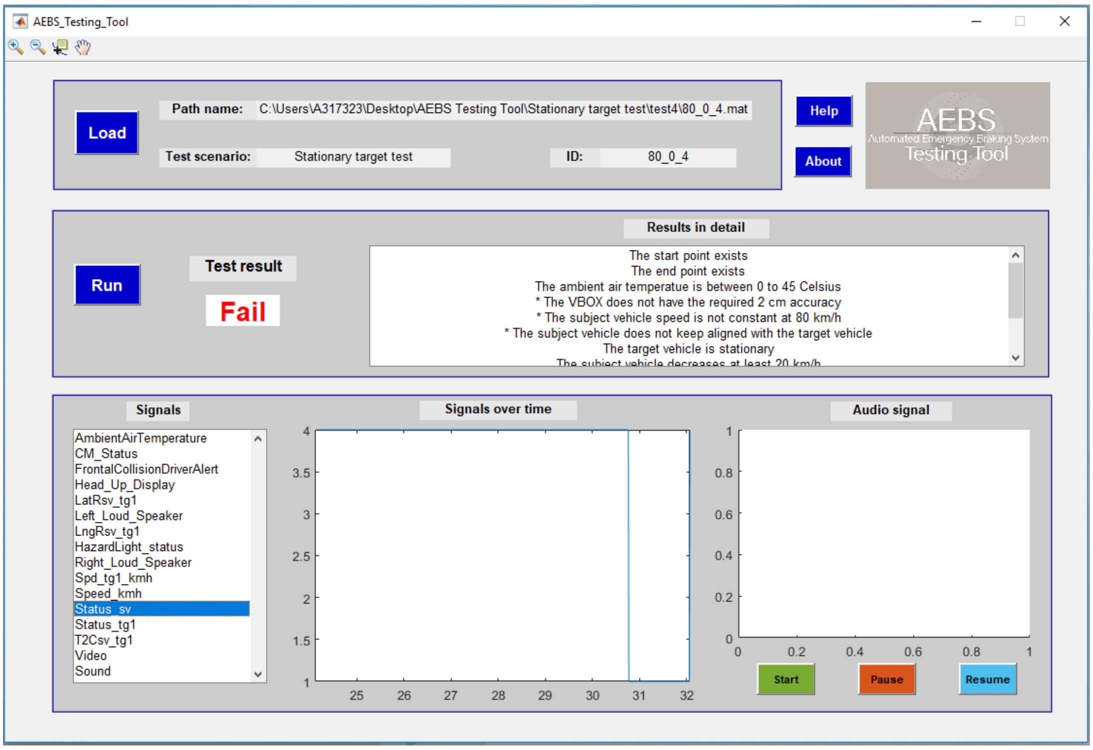

Figure 4.17 The graphic user interface for stationary target test case 4.

The fifth stationary target test is passed as Figure 4.18 displays. When one check all the requirements, it can be found that they are fulfilled for this test case. Therefore, the tool displays the correct test result.

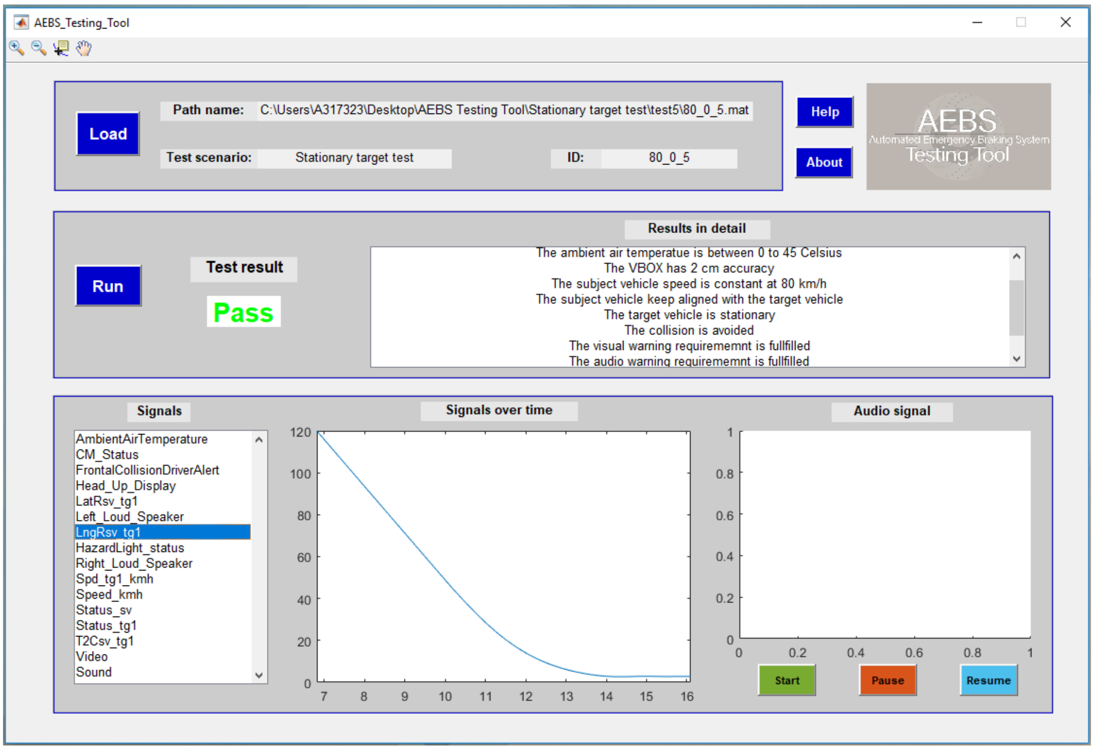

Figure 4.18 The graphic user interface for stationary target test case 5.

4.4 User acceptance testing

User acceptance testing is process of verifying the tool was designed as the users expected. It is the last phase to ensure that the tool is acceptable by the end users according to their requirements. Since the users are involved in this phase, they test the tool in a direct way and may find out problems which are missed by the developers during the previous phases.

The tool was handed over to the users after the system testing was performed. What the users did was black box testing. They do not know the internal codes but the requirements the tool should meet. They were taught to how to use the tool first. The test cases were randomly chosen by the users.

Based on the feedbacks of end users, the tool was user-friendly and well designed in general. The tool can also fulfil the requirements defined at the very beginning. However, there are also some advices for further improving the tool:

- The text of unfulfilled requirements is presented with a “*” sign, which is not so friendly for users. It is preferable to be written in red or have a red background

- The control of the audio signal would be better if there is a slider controlling the video to skip forward and backward

5 Discussion and future work

5.1 Discussion

The section presents the discussion of methods and results. The designed tool, the software development life cycle, the development environment MATLAB/GUI and the software verification testing are discussed.

5.1.1 The designed tool

The real-world testing of AEB systems is indispensable, not only because it can test the sensors and the whole system in a real environment but also because it is prescribed by the legislation. The tool was designed to help the test engineers evaluate the track testing results of AEB systems, which was verified to be effective for various test cases.

Actually, it is also very helpful for test engineers to improve the testing efficiency. The legal requirements and test procedures require the AEB system to be tested at some specified situations. For example, the subject vehicle is required to travel at specified speed and have a specified relative position with the target. There are also many performance requirements for the AEB system, such as the timing for warning and braking, the speed reduction at different phases. The test engineers used to extract useful information from different signals manually and then check if each requirement is fulfilled. The process could be very difficult and time-consuming for test engineers to check all the requirements for the AEB system to pass the test. The designed tool helps the test engineers evaluate the test result by automatically checking the requirement one by one, thus saving the time for analysis and data processing dramatically.

Besides, the requirements for the testing of AEB systems may have changes in the future. The changes may come from the legal department, the market or the industry. There could be higher performance requirements or more complex test scenarios for the testing of AEB systems. All the changes will promote the development of AEB systems. AEB systems may be expected to recognize and handle different situations more accurately. The designed tool has the potential of being extended to adapt to the new requirements in a similar way. The new requirements need to be collected to guide the algorithm design while there will be no big changes for the HMI design.

5.1.2 Software development life cycle

There are many SDLC models for software development. SDLC models will guide the developers within the whole development process, which is very essential for developing the software in an effective way. As a widely used SDLC model, the waterfall model was selected to design and verify the developed tool in this thesis.

With the waterfall model, the development moves to next phase until the previous phase is completed. Besides, the previous phases may need to modify according to feedbacks from lower phases. The development lifecycle could be problematic if the requirements are dynamic in the development process. Consequently, the development may lead to delayed deliveries and overspending budget.

However, the waterfall model is suitable for the designed tool in this thesis. The user requirements for the tool are well understood at first and there is almost no change for the requirements. Therefore, the main focus is to have detailed discussion with users about requirement specifications and get all the requirements as accurate as possible. Moreover, it will be easier to set a proper schedule for the tool development after collecting all the requirements.

5.1.3 MATLAB/GUI

The MATLAB/GUI based tool was designed to help test engineers to evaluate the test results effectively. This tool provides a user-friendly interface and robust performance for different test cases. After loading the correct data file, the user can evaluate the test result immediately. The tool will present the user the test is passed or failed and list all the fulfilled and unfulfilled requirements. The unfulfilled requirements are displayed different with fulfilled ones. The tool also allows the user to further analyse the test data by plotting signals, playing audio signals and displaying videos.

MATLAB/GUI is a powerful tool for graphic user interface design. It is easy-to-use for developers since it has implemented build-in functions for uicontrol objects. The developers can just focus on programming for the callbacks. The designed interface is also friendly for the users. The uicontrol objects in the interface are intuitive and behave in a predictable way so that the user knows what to expect after taking action on different uicontrol objects.

5.1.4 Software testing

Software testing is an essential process to verify whether the developed software meets all the specified requirements, thus improving the quality of the software. The tool was verified through different levels of testing in this thesis.

Unit testing was first performed to check whether the individual functions work well separately. The results of unit testing show that all three functions work properly. However, there may be mistakes when these three functions are integrated. The reason can be that there are data transmission between different functions. Therefore, system testing should be performed.

System testing tested the system as a whole to verify that the tool meets all the requirement specifications. Different moving and stationary target test cases were selected to validate the effectiveness of the tool at a system level. The results show that the designed tool worked well for various test cases. The tool presented accurate results about those test cases that can be evaluated. There were also warnings to the users for lost data or unexpected errors.

The user acceptance testing was then conducted by end users to assess whether it is acceptable. The users did black box testing about the tool and the results show that the tool was well developed. The tool meets the functional and non-functional requirements defined in the requirements phase, though there are also some improvement suggestions about the HMI design.

Different levels of testing for the tool demonstrate that the tool fulfills all the requirements. The tool is well designed, user-friendly and also has robust performance for various test cases. However, there may be errors in the future since the tool were only verified using finite test cases. Besides, the requirements may be changed for the tool. Further work needs to be performed to improve the quality of the tool. That is the maintenance phase introduced in the waterfall model.

5.2 Future work

The tool designed for AEB system works well at the moment. According to the advices from the end users, the tool can be further designed by increasing the user interface design and better control of the audio signal. Besides, the tool may need to be tested with more test cases in case there will be unexpected errors for special test cases.

Except for moving and stationary target test, there are also some internal requirements for AEB system working at other test scenarios like intersection or curve road. The future work about this tool can involve such test scenarios so that the tool can be used to evaluate test results based on internal requirements.

Besides, the tool can be also extended with other active safety systems. Another challenge can be expansion of the tool for LDW system. The requirement specifications and test procedures for LDW system based on legal requirements should be investigated first. Since different required signals may be used for LDW system, the user interface should be further designed.

6 Conclusion

Because of stricter legislation, higher customers' expectation, and the increasing number of vehicle manufacturers' new requirements about AEB system, future test track evaluation of AEB systems will include an increasing number of tests.

The main contribution of this thesis is the designed tool which is a first step in helping test engineers to evaluate the test results automatically and greatly improve time efficiency. The tool is verified to be well developed with user-friendly interface and robust performance.

The tool was developed using MATLAB/GUI. The development environment MATLAB/GUI has proven to be powerful for graphic user interface design and user-friendly for both developers and users.

The method used to develop and verify this tool is the SDLC waterfall model. The waterfall model is suitable for software development when the user requirements are completely understood at the beginning and there is no big change for the requirements during the development process.

7 References

Lesemann M., Lutzow J., Zlocki A. (2010): eVALUE — A Test Program for Active Safety Systems. *17th ITS World Congress*, Busan, October 2010.

Gietelink O., Ploeg J., De Schutter B., Verhaegen M. (2000): Development of Advanced Driver Assistance Systems with Vehicle Hardware-in-the-Loop Simulations. *Vehicle System Dynamics*, Vol. 44, No. 7, 2006, pp. 569-590.

Schoner H-P., Neads S., Schretter N. (2009): Testing and Verification of Active Safety Systems with Coordinated Automated Driving. *Proceedings of the 21st Enhanced Safety of Vehicles Conference*, Stuttgart, June 2009.

Hendriks F., Pelders R., Tideman M. (2010): Future Testing of Active Safety Systems. *SAE International Journal of Passenger Cars-Electronic and Electrical Systems*, Vol. 3, No. 2, December 2010, pp. 170-175.

Den Camp O. O., Montfort S. van, Uittenbogaard J., Welten J. (2017): Cyclist Target and Test Setup for Evaluation of Cyclist-Autonomous Emergency Braking. *International Journal of Automotive Technology*, Vol. 18, No. 6, 2017, pp. 1085-1097.

Van Ratingen M., Williams A., Lie A., et al. (2016): The European New Car Assessment Programme: A historical review. *Chinese Journal of Traumatology*, Vol. 19, No. 2, April 2016, pp. 63-69.

Nowak P. (2018): How Automatic Emergency Braking Works. *The Golbe and Mail*, URL: https://www.theglobeandmail.com/drive/article-how-automatic-emergency-braking-works/.

Lee T., Yi K., Kim J., Lee J. (2011): Development and Evaluations of Advanced Emergency Braking System Algorithm for the Commercial Vehicle. *Enhanced Safety of Vehicles Conference*, Washington DC, 2011.

Lee D., Kim B., Yi K., Lee J. (2012): Development of an Integrated Driving Path Estimation Algorithm for ACC and AEB system Using Multi-sensor Fusion. *2012 IEEE 75th Vehicular Technology Conference*, May 2012, pp. 1-5.

Lee H. K., Shin S. G., & Kwon D. S. (2017): Design of Emergency Braking Algorithm for Pedestrian Protection Based on Multi-sensor Fusion. *International Journal of Automotive Technology*, Vol. 18, No. 6, December 2017, pp. 1067-1076.

Han I. C., Luan B. C., Hsieh F. C. (2014): Development of Autonomous Emergency Braking Control System Based on Road Friction. *2014 IEEE International Conference on Automation Science and Engineering*, 2014.

Kim T., Lee J., Yi K. (2015): Enhanced Maximum Tire-road Friction Coefficient Estimation Based Advanced Emergency Braking Algorithm. *2015 IEEE Intelligent Vehicles Symposium*, 2015, pp. 883-888.

Kim H., Shin K., Chang I., Huh K. (2018): Autonomous Emergency Braking Considering Road Slope and Friction Coefficient. *International Journal of Automotive Technology*, Vol. 19, No. 6, 2018, pp. 1013-1022.

Guo L., Ren Z., Ge P., Chang J. (2014): Advanced Emergency Braking Controller Design for Pedestrian Protection Oriented Automotive Collision Avoidance System. *The Scientific World Journal*, 2014.

Zhang R., Li K., He Z., Wang H., You F. (2017): Advanced Emergency Braking Control Based on a Nonlinear Model Predictive Algorithm for Intelligent Vehicles. *Applied Sciences*, Vol. 7, No. 5, 2017.

Koglbauer I., Holzinger J., Eichberger A., Lex C. (2018): Autonomous Emergency Braking Systems Adapted to Snowy Road Conditions Improve Drivers' Perceived Safety and Trust. *Traffic injury prevention*, Vol. 19, No. 3, 2018, pp. 332-337.

Duan J., Li R., Hou L., Wang W., et al. (2017): Driver Braking Behavior Analysis to Improve Autonomous Emergency Braking Systems in Typical Chinese Vehicle-bicycle Conflicts. *Accident Analysis & Prevention*, 2017, pp. 74-82.

Brännström M., Sjöberg J., Coelingh E. (2008): A Situation and Threat Assessment Algorithm for a Rear-end Collision Avoidance System. *2008 IEEE Intelligent Vehicles Symposium*, June 2008, pp. 102-107.

UNECE (2019): UN Regulation on Advanced Emergency Braking Systems for Cars to Significantly Reduce Crashes, URL: https://www.unece.org/info/media/presscurrent-press-h/transport/2019/un-regulation-on-advanced-emergency-braking-systems-for-cars-to-significantly-reduce-crashes/doc.html.

Zhao M., Wang H., Chen J., Xu X., He Y. (2017): Method to Optimize Key Parameters and Effectiveness Evaluation of the AEB system Based on Rear-end Collision Accidents. *SAE International Journal of Passenger Cars-Electronic and Electrical Systems*, 2017, pp. 310-317.

Fildes B., Keall M., Bos N., Lie A., Page Y., Pastor C., et al. (2015): Effectiveness of Low Speed Autonomous Emergency Braking in Real-world Rear-end Crashes. *Accident Analysis & Prevention*, August 2015, pp. 24-29.

Rosén E., Källhammer J. E., Eriksson D., Nentwich M., Fredriksson R., & Smith K. (2010): Pedestrian Injury Mitigation by Autonomous Braking. *Accident Analysis & Prevention*, Vol. 42, No. 6, 2010, pp. 1949-1957.

Cicchino J. B. (2017): Effectiveness of Forward Collision Warning and Autonomous Emergency Braking Systems in Reducing Front-to-rear Crash Rates. *Accident Analysis & Prevention*, 2017, pp. 142-152.

Strandroth J., Rizzi M., Kullgren A., Tingvall C. (2012): Head-on Collisions Between Passenger Cars and Heavy Goods Vehicles: Injury Risk Functions and Benefits of Autonomous Emergency Braking. *IRCOBI Conference*, 2012, pp. 342-351.

European Road Safety Observatory (2016): Vehicle Safety 2016, URL: https://ec.europa.eu/transport/road_safety/sites/roadsafety/files/ersosynthesis2016-vehiclesafety15_en.pdf.

The United Nations Economic Commission for Europe (2019), URL: https://www.unece.org/trans/main/wp29/meeting_docs_wp29.html.

The European New Car Assessment Programme (Euro NCAP) (2018), URL: https://www.euroncap.com/en/about-euro-ncap/timeline/.

The European New Car Assessment Programme (2017): Euro NCAP 2025 Roadmap, URL: https://www.euroncap.com/en/for-engineers/technical-papers/.

The United Nations Economic Commission for Europe (UNECE) (2014), URL: https://eur-lex.europa.eu/legal-content/GA/TXT/?uri=CELEX:42014X0719(01).

Ruparelia N. B. (2010): Software Development Lifecycle Models. *ACM SIGSOFT Software Engineering Notes*, Vol. 35, No. 3, 2010, pp. 8-13.

Bassil Y. (2012): A Simulation Model for the Waterfall Software Development Life Cycle. *International Journal of Engineering & Technology*, Vol. 2, No. 5, May 2012, pp. 742-749.

Sharma M. K. (2017): A Study of SDLC to Develop Well Engineered Software. *International Journal of Advanced Research in Computer Science,* Vol. 8, No. 3, March 2017, pp. 520-523.

Espinosa H. G., James D. A., Kelly S., Wixted A. (2013): Sports Monitoring Data and Video Interface Using a GUI Auto Generation Matlab Tool. *Procedia Engineering*, 2013, pp. 243-248.

IPETRONIK (2019), URL: https://www.ipetronik.com/data-loggers-detail/mm_datalogger_alias/m-log-v3.html.

RACELOGIC (2019), URL: http://www.racelogic.co.uk/index.php/en/applications.