Graduate School Form 30
(Revised 08/14)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Bo Tang

Entitled
Pedestrian Protection Using the Integration of V2V Communication and Pedestrian Automatic Emergency Braking System

For the degree of Master of Science in Electrical and Computer Engineering

Is approved by the final examining committee:

Stanley Yung-Ping Chien

Yaobin Chen

Lingxi Li

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): Stanley Yung-Ping Chien

Approved by: Brian King 11/24/2015

Head of the Department Graduate Program Date

PEDESTRIAN PROTECTION USING THE INTEGRATION OF V2V COMMUNICATION AND PEDESTRIAN AUTOMATIC EMERGENCY BRAKING SYSTEM

A Thesis

Submitted to the Faculty

of

Purdue University

by

Bo Tang

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

December 2015

Purdue University

Indianapolis, Indiana

ACKNOWLEDGMENTS

I would like to gratefully thank my major professor Dr. Stanley Yung-Ping Chien for his many instructions and assistance in my research. I also would like to thank Dr. Yaobin Chen and Dr. Lingxi Li who are my committee members in my final defense for their great recommendations and suggestions during the course of my preparation of my thesis. This study sponsored by the Crash Imminent Safety University Transportation Center. We also would like to thank TASS International for providing us the PreScan software to support this research.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

ABSTRACT

Tang, Bo M.S.E.C.E., Purdue University, December 2015. Pedestrian Protection Using the Integration of V2V Communication and Pedestrian Automatic Emergency Braking System. Major Professor: Stanley Yung-Ping Chien.

The Pedestrian Automatic Emergency Braking System (PAEB) can utilize on-board sensors to detect pedestrians and take safety related actions. However, PAEB system only benefits the individual vehicle and the pedestrians detected by its PAEB. Additionally, due to the range limitations of PAEB sensors and speed limitations of sensory data processing, PAEB system often cannot detect or do not have sufficient time to respond to a potential crash with pedestrians. For further improving pedestrian safety, we proposed the idea for integrating the complimentary capabilities of V2V and PAEB (V2V-PAEB), which allows the vehicles to share the information of pedestrians detected by PAEB system in the V2V network. So a V2V-PAEB enabled vehicle uses not only its on-board sensors of the PAEB system, but also the received V2V messages from other vehicles to detect potential collisions with pedestrians and make better safety related decisions. In this thesis, we discussed the architecture and the information processing stages of the V2V-PAEB system. In addition, a comprehensive Matlab/Simulink based simulation model of the V2V-PAEB system is also developed in PreScan simulation environment. The simulation result shows that this simulation model works properly and the V2V-PAEB system can improve pedestrian safety significantly.

1. INTRODUCTION

1.1 Background and Motivation

According to a recent study performed by World Health Organization (WHO), more than 1.24 million road traffic deaths occur each year and 22 percent of them are pedestrians. In recent years, pedestrians are among the most vulnerable road users and most accidents occurs when pedestrians trying to cross highways. Studies indicate that males make up a higher proportion of pedestrian deaths and injuries in traffic accidents than females. Additionally, in developed countries, older pedestrians are often involved in road accidents, while in underdeveloped or developing areas, children and young people are more often affected. About 1.24 million of road users lose their lives on the world's roads annually, making road traffic injuries the eighth leading cause of death around the world, and the leading cause of death for young people aged between 15 and 29 years [1].

There are many methods to improve the safety of pedestrians on the roads. For example, we can adopt and enforce new and existing traffic laws to reduce speeding, curb drinking and driving, decrease mobile phone use and other forms of distracted driving. We also can put in place infrastructure which separates pedestrians from other traffic (sidewalks, raised crosswalks, overpasses, underpasses, refuge islands and raised medians), lowers vehicle speeds (speed bumps, rumble strips and chicanes) and improves roadway lighting. Additionally, we can develop and enforce vehicle design standards for both active and passive systems. passive safety usually refers to features that help reduce the effects of an accident, such as seat belts, airbags and strong body structures. However, active safety is increasingly being used to describe systems (such as the Advanced Driver assistance System) that use an understanding of the state of the vehicle to avoid or mitigate the effects of a crash.

Pedestrian Automatic Emergency Braking (PAEB) system is a type of active system which uses various types of on-board sensors (such as radar, mono/stereo camera, infrared etc.) to detect the potential crash to pedestrians. The PAEB system alerts the driver if there is an imminent collision and supports collision avoidance by applying the brake automatically if the driver does not take braking action [2]. However, due to the range limitations of PAEB sensors and speed limitations of sensory data processing, the PAEB system often cannot detect or do not have sufficient time to respond to a potential crash. For example, the collision with pedestrian will be unavoidable if the vehicle is travelling too fast or the pedestrian enters the path of the vehicle too quickly. Another example is that the PAEB system cannot detect the pedestrians behind obstacles. If the pedestrian suddenly comes out from the behind of an obstacle, the PAEB system might have not enough time to react. Additionally, the harsh weather or ambient conditions may also affect the performance of the sensors used by the PAEB system.

On the other hand, due to the fast advancement of wireless communication technology, V2V communication becomes practical. The exchanged information through V2V enables the vehicles to make better decisions in driving control and safety [3]. In the V2V communication network, each vehicle is one communicating node, providing each other with safety related information, such as safety warning and traffic information. With the help of shared information, one vehicle can acquire a full picture of its driving environment and obtain more information to make better decisions. In the vehicular communication systems, the vehicles can cooperate with each other which making them more effective in avoiding road accidents and traffic jams than if each vehicle tries to solve these problems individually.

To improve the performance of PAEB, we proposed the idea to integrate the complimentary capabilities of V2V and PAEB (V2V-PAEB) to allow the information of pedestrians sensed by the PAEB system of one vehicle to be shared in the V2V network and to be used by the the PAEB system of other vehicles. So if the PAEB system on one specific vehicle fails in detecting the potential collision with a pedestrian, it may still have a chance to aware about this pedestrian from the messages sent by other vehicles and then take proper safety actions. By this means, the V2V-PAEB will improve the road safety and have a better performance than PAEB system.

1.2 Related Work and Major Contributions

PAEB systems already benefit the road safety significantly in the real world, but the V2V communication systems are still under research. To the best knowledge of the authors, there is no architecture and principle of operations of the V2V-PAEB system have been defined. In this study, we defined the architecture and the information processing stages of the V2V-PAEB system. The V2V-PAEB system defined in this study consists of 10 blocks and each block is designed to solve some specific problems. The input and output of the V2V-PAEB system as well as its each block are also defined clearly. This study provides others with the quick start for studying the V2V-PAEB system.

One essential way to develop and evaluate a V2V-PAEB system is to develop a real combined V2V and PAEB system and conduct real vehicle tests. However, this approach is quite costly, dangerous and time consuming, so we developed a Matlab/Simulink based simulation model of V2V-PAEB system. Various complex and severe crash scenarios can be generated in the simulation environment, and the information processing and control algorithms can be easily developed and verified. This simulation model is organized according to the information processing stages and the problems need to be solved of a V2V-PAEB system. Currently, the architecture as well as some basic control algorithms have been implemented, and it works properly under many common conditions. However, since the V2V-PAEB system is quite complicated, this study did not solve all the problems, so the V2V-PAEB simulation model may not work properly under some specific conditions and it still need further study. With the predefined architecture and function blocks of this model, we can immediately start to upgrade the algorithms in the corresponding blocks of the model in the future. Since the input and output for each block is defined fixed, so when the algorithms in one block is upgraded, other blocks are not affected. This V2V-PAEB model provides us the convenience of focusing on solving the V2V-PAEB integration problems and minimizing the effort in writing supporting software.

Additionally, we also tested the performance of the V2V-PAEB simulation model using several typical scenarios which described in [4]. Paper [4] used an exhaustive analysis method to identify the scenarios that the V2V-PAEB system can improve the pedestrian safety theoretically. Totally 96 out of 168 pedestrian related scenarios can benefit from the V2V-PAEB system were identified. However, none of them have been proved that the V2V-PAEB system can really benefit the pedestrian safety in such scenarios. We can use the proposed V2V-PAEB simulation model to test these scenarios presented in [4].

1.3 Thesis Organization

The thesis has 7 chapters and it is organized as follows. The next chapter, Chapter 2 will introduce the basic concepts of PAEB systems. Additionally, the most frequent used sensors in PAEB systems will also be presented in this chapter. Chapter 3 will provide the description of V2V communication systems. Chapter 4 describes the PreScan simulation environment that used for developing the proposed V2V-PAEB simulation model. Chapter 5 provides the detail definition of the architecture and the information processing stages of V2V-PAEB system. current implementation of the V2V-PAEB simulation model. In Chapter 6 we will use PreScan software to test the proposed simulation model, and discuss the benefits of V2V-PAEB system based on the simulation results. We will conclude this study in Chapter 7 to make a summary of of the V2V-PAEB system and also discuss the future work based on current study.

2. PAEB SYSTEM

2.1 Description of PAEB System

The PAEB system is specially designed for protecting pedestrians and it is one of the key features of AEB systems. The PAEB system judges the probability of a collision based on the position and relative speed of the vehicle with respect to an pedestrian, and either help the driver to avoid the collision by triggering proper warnings or help to mitigate collision damage by activating devices such as automatic brake assist, automatic steering, and so on [5].

In February 2003, Toyota Motor Corporation developed the first commercial AEB system and brought it to market with its high-end vehicles. Currently, the AEB technology has advanced to be able to detect and protect pedestrians as well as frontal collisions at both the intersections and roads [6,7]. In contrast, most conventional automatic braking systems are designed to activate only when a collision is unavoidable, help to avoid the collision or mitigate the damage caused by the collision. However, new systems have since been developed that are capable of avoiding some types of collisions automatically.

AEB systems usually improve the road safety in two ways: firstly, they use on-board sensors to detect objects and help to avoid accidents by identifying critical situations early and warning the driver; and secondly they reduce the severity of collisions which cannot be avoided by lowering the speed of crash and, in some cases, by preparing the vehicle and restraint systems for impact [8].

Most AEB systems use a combination of various types of sensors such as radar, (stereo) camera and/or lidar-based technology to identify potential collision partners ahead of the vehicle. Then AEB systems combine this information with the vehicle's own state information such as its travel speed and trajectory to determine whether a critical situation is developing. If a potential collision is detected, AEB systems generally first try to avoid the impact by warning the driver to take proper actions. If the driver takes no action and a collision is unavoidable, the system will then apply the brakes automatically to reduce the damage of collision. Some systems apply full braking pressure during the braking process, while others can apply an elevated level according to the emergency level. Either way, the intention is to reduce the speed with which the collision takes place. Some systems deactivate as soon as the driver takes actions to avoid this potential collision [8].

In the year 2013, Toyota Motor Corporation has developed an AEB system that uses automatic steering in addition to automatic braking to help prevent collisions with pedestrians. It is the first commercial AEB system that has automatic steering capability. Toyota is committed to developing safety technologies that help eliminate traffic fatalities and injuries involving pedestrians and other vulnerable road users. The new AEB system with Pedestrian-avoidance Steer Assist can help prevent collisions in cases where automatic braking alone is not sufficient, such as when the vehicle is travelling too fast or a pedestrian suddenly steps into the vehicle's path. If the on-board sensors detect the pedestrians in front of the vehicle and the system determines that there is a potential collision, then the AEB system will issue a driver warning immediately to encourage the driver to take evasive actions. The automatic braking functions are activated if the potential collision is urgent. If the system determines the potential collision is unavoidable by braking alone and there is sufficient room for avoidance, it will activate the steer assist to steer the vehicle away from the pedestrian in front of vehicle.

As an emerging active safety system, the AEB technology is already showing great benefits in improving road safety in the real world. A recent report performed by IIHS shows that the AEB technology can reduce insurance injury claims by as much as 35%. The 10 manufacturers committing to across-the-board AEB represented 57% of U.S. light-duty vehicle sales in 2014 [9].

Vehicle-mounted sensors are useful in detecting pedestrians and other objects on the road. However, visibility from the vehicle is limited. It is often the case that it is difficult or impossible to observe the dangerous object from the vehicle itself. Due to the range limitations of PAEB sensors and speed limitations of sensory data processing, PAEB systems often cannot detect or do not have sufficient time to respond to a potential crash. For example, the potential collision with pedestrians cannot be avoided if the vehicle is travelling too fast or the pedestrian enters the path of the vehicle too quickly. Another example is that the PAEB cannot detect the pedestrians behind the obstacles. If the pedestrian suddenly comes out from the behind of an obstacle, PAEB might have not enough time to react. Additionally, the performance of the sensors in PAEB systems can be easily affected by the harsh weather or ambient environments.

2.2 Commonly Used Sensors in PAEB System

There are many types of sensors can be used for pedestrian detection in PAEB systems. The following paragraphs present the detail information of the sensors that commonly used in PAEB system. Additionally, we also discussed the advantages and disadvantages for each type of sensor.

1. Radar Sensor

   Radar is short for Radio Detection and Ranging or Radio Angle Detection and Ranging. It is a system works in the frequency domain and can be used to detect, range and track both moving and fixed objects such as vehicles and pedestrians. Radar sensor usually transmits strong electromagnetic waves, specifically radio waves, and uses a receiver to listen for any reflections from the obstacles. The radar systems use the reflected signals from the detected objects to identify their range, direction and speed. Sometimes the type of objected can also be identified. Since the amount of signal returned to the receiver is tiny, so the radar sensor usually amplify the reflected radio signals many times in order to detect them easily. Thus radar is suitable for long range detection. However, other detectors that based on sound or visible light will have a worse performance than radar because such type of reflections are too weak to detect at a large range [10, 11].

In the current PAEB systems, many types of radar systems can be used and among which the Pulse-Doppler radar is the most frequent used. The Pulse-Doppler is a type of radar system that uses the Doppler effect to detect and locate the obstacles or objects in front of the sensor. The Doppler effect is the change in frequency of a wave (or other periodic event) for an observer moving relative to its source. It was first proposed by Doppler in the year 1842. The Doppler radar system uses a transmitter to send out short pulses of waves and simultaneously listens for the reflecting signals from objects using the receiver. The range of the object is determined by examining the time delay between the pulse transmission and reflection. The speed of the object can be identified by observing the change in frequency of the waves between the transmitted signal and the reflected signal.

The advantage of the radar sensor is that it can be used for long range and short range detection. The difference is that a different frequency is used. In the long range detection, the 76-77 GHz frequency is used. While for the short range detection, a typical frequency of 24.125 GHz will be used [11].

2. Lidar

   Lidar is an acronym for Light Detection and Ranging; or Laser Imaging Detection and Ranging, which is also frequently used in PAEB systems to detect and track the objects. The difference with radar sensor is that the lidar sensor determines the distance to an object or surface using laser (Light Amplification by Stimulated Emission of Radiation) pulses instead of radio waves. Comparing with the radio waves, the laser pulses use a much shorter wavelengths of the electromagnetic spectrum. In general it is possible to image an object only about the same size as the wavelength, or larger. Thus the lidar sensor usually has a higher resolution than radar sensor. However, since a shorter wavelength is used, it usually has a shorter detection range than radar sensor. Additionally, in the lidar system, the range to an object is also determined by measuring the time delay between transmission of a pulse and detection of the reflected signal [11].

At radar (microwave or radio) frequencies a metallic object generates a significant reflection and can be easily detected by the radar sensor. However non-metallic objects, such as water and concrete usually generate weaker reflections or even no detectable reflection at all, meaning some objects or obstacles are hardly detected by radar sensors. Laser sensor (always a part of lidar sensor) provides a better performance in such conditions. A laser sensor is an optical source that emits laser light in a coherent beam. Laser light is typically near-monochromatic consisting of a single wavelength and emitted in a narrow beam, so it has a good directional feature. However, many common light sources, such as the incandescent light bulb usually emit incoherent lights in almost all directions and over a wide spectrum of wavelengths. Additionally, the wavelengths of laser usually range from about 10 micrometers to the UV (ca. 250 nm) which are much smaller than that can be achieved by radar sensor systems. So a lidar system can offer much higher resolution than radar and sometimes can obtain the images of the detected objects. Based on the image, sometimes the type of the objects can be identified by applying appropriate classifiers [11, 12].

3. Infrared Camera

   An infrared camera (also called thermal imaging camera) is a device that generates an image using infrared radiation, similar to a common camera that generates an image using visible light. Instead of the 400700 nanometer range of the visible light camera, infrared cameras operate in wavelengths as long as 14,000 nm (14 m) [13].

The infrared camera makes PAEB systems possible to have the ability to see in low light conditions, especially at nights, where the common vision camera usually have very poor performances. The drivers also have poor night vision compared to many animals. That's because the human eye lacks a tapetum lucidum. With the help of infrared camera, the road safety can be improved significantly in bad light conditions.

4. Vision Camera

   A vision camera is a device that captures the information of reality that constitutes an image. Vision cameras are used in electronic imaging devices of both analog and digital types. Currently, the most frequent used types of image sensors are semiconductor charge-coupled devices (CCD) or active pixel sensors in complementary metaloxidesemiconductor (CMOS). Both types of sensors are used to capture light and convert it into electrical signals [14].

   The CCD image sensor has thousands of cells and each cell is an micro analog device that representing one pixel of an image. When light hits the chip it is held as a small electrical charge in each cell. The tiny charges are then converted to voltage one pixel at a time as they are read from the chip. Then the CCD sensor uses some additional circuit to convert the voltage into digital signals. Usually different voltage levels represent different colors. A CMOS imaging chip is a type of active pixel sensor made using the CMOS semiconductor process. Extra circuit next to each image sensor converts the light energy to a voltage. Similar with the CCD image sensor, the CMOS also needs additional circuit to convert the voltage to digital information. Currently, CMOS image sensor is more popular than CCDs, and most digital still cameras use a CMOS sensor instead of a CCD. However, CCD is still in use for cheap or low-end cameras [14].

5. Ultrasonic

   Ultra-sonic sensors are frequently used in automotive industry for the short range obstacle detection, especially for back maneuver assist applications. The ultra-sonic sensors usually utilize a 40 kHz sound pressure wavelength to detect obstacles or features. The ultra-sonic sensor usually have an very short detection range of approximately 1 to 3 meters [11].

Additionally, the ultra-sonic sensors can provide a wide angle for detecting objects. In the horizontal direction, the maximum detection range can be 100 degrees, while for the vertical direction it would be 60 degrees. However, the ultra-sonic sensors are easily distored by the reflections of the road and they provide very poor positioning capabilities [11, 15].

The sensors mentioned above are frequently used for the pedestrian detection in today's PAEB systems. However, each type of sensor has its advantages and limitations. In order to enhance the advantages and overcome the limitations, the PAEB system usually uses a combination of multiple numbers and types of sensors that give complementary information. The following table presents the advantages and limitations for each type of sensor.

Table 2.1. The advantages and limitations of different sensors.

Sensor	**Advantages**	**Limitations**
Radar	1.Detect objects with reflections. 2.Accurate speed and distance detection. 3.Suitable for short/long range detection.	1.Beams are easily blocked. 2.Resolution is very low. 3.The sensor size is big.
Lidar	1.Can detect small obstacles. 2.Accurate speed and distance detection. 3.Higher resolution than Radar.	1.Smaller detection range than Radar. 2.Provides poor resolution. 3.The sensor size is big.
Infrared Camera	1.Usable under night conditions. 2.Provide high resolution image.	1.Can not identify traffic signs.
Vision Camera	1.Provide high resolution image. 2.Showing images of reality. 3.The sensor size is small.	1.Difficult for data processing. 2.Bad speed and distance detection. 3.Bad obstacle detection.
Ultrasonic	1.Suitable for short range detection. 2.Has a high angular detection range.	1.Easily distorted by reflections. 2.No angular position provided. 3.No echo cancellation.

3. V2V COMMUNICATION SYSTEM

V2V is an automobile wireless communication technology designed to allow vehicles to talk to each other share some useful information. In the V2V network, many types of information can be shared among vehicles, such as the state information of vehicles, safety warnings and traffic information. The V2V communication system is an cooperative approach which can be more effective in avoiding accidents on roads and improving traffic flows than if each vehicle tries to solve these problems individually. For example, by sharing the vehicle state information among vehicles, the V2V communication could help to warn the drivers about the vehicles in the blind spot or that unseen by the drivers. Additionally, by sharing the traffic congestion information the traffic flow can be redirected and the flow rate can be improved. So some special vehicles such as ambulance vehicle and police vehicle can plan their trajectory effectively to reduce the rescue time [16].

V2V communications systems use dedicated short-range radio communication (DSRC) to exchange messages containing different types of information, such as the vehicle information (e.g., vehicle's speed, heading, braking status). V2V devices use the shared information from other vehicles to detect dangerous situations and determine if a warning to the vehicles driver should be issued in order to avoid or reduce the severity of collisions.

In DSRC based V2V communication system, the V2V messages have a transmission range of approximately 1000 meters, which exceeds the capabilities of systems with different types of sensors and allowing more time to warn drivers and give them more time to react to a potential dangerous situation. In addition, these radio messages are not easily blocked by the obstacles. However, the on-board sensors such as radar sensors and cameras usually suffer these problems. For example, situations such as those where an oncoming vehicle emerges from behind a truck, or perhaps from a blind spot. In those situations, V2V communications can detect the potential collisions much earlier than radar or camera sensors. Additionally, V2V technology can also be combined with existing on-board sensor systems such as radar and camera sensors to provide even greater benefits than either approach alone. So the information provided by the on-board sensors can also be shared in the V2V network. In this case, a vehicle not only broadcast what they themselves are doing, but also what they have "seen". This combined approach could also augment system accuracy and produce more applications in automotive industry, and it will become a foundation for developing the auto-mated vehicles [16].

Based on DSRC technology, many safety applications that help drivers with different aspects of driving can be implemented, like warning about stopped vehicles in the road ahead, vehicles speeding unexpectedly through intersections, vehicles in blind spots, etc. NHTSA's analysis of two potential applications, "intersection movement assist" (IMA) and "left turn assist" (LTA), indicated there could be a 50 percent reduction, on average, in crashes, injuries, and fatalities for just these two applications. Applied to the full national vehicle fleet, this could potentially prevent 400,000 to 600,000 crashes, 190,000 to 270,000 injuries, and save 780 to 1,080 lives each year. Of course, the addition of other V2V and vehicle-to-infrastructure (V2I) safety applications would save even more lives [16].

DSRC is a bidirectional wireless communication technology permitting secure and fast messaging needed for safety applications. DSRC works in a 75 MHz band of the 5.9 GHz spectrum and has a max range of approximately 1000 meters depending on the surrounding environment. This band affords a relatively clean operating environment with very few preexisting users, allowing for a relatively unimpeded and interference-free communication zone. DSRC-based devices can be installed directly in vehicles when originally manufactured. In the DSRC V2V network, many types of V2V messages can be shared among vehicles and serving different safety related purposes. Among these V2V messages, the basic safety message (BSM) most frequently used. In the basic system defined by the SAE J2735 standard, each moving vehicle updates and sends its own BSM every 100ms over the WAVE Short Message (WSM) channel. The BSM is exchanged between vehicles and contains vehicle dynamics information such as heading, speed, and location. The BSM is updated and broadcast to surrounding vehicles every 100 ms. The information is received by the other vehicles equipped with V2V devices and processed to determine collision threats. Based on that information, many applications can be developed. For example, it can be used to detect dangerous situations around this vehicle and if required, a warning could be issued to drivers to take appropriate action to avoid an potential collision [16].

Although current V2V has great potential in improving road safety, there are still some limitations. Firstly, V2V can only benefit the safety for the V2V-enabled vehicles. There are millions of non-V2V-enabled vehicles that cannot benefit from the V2V technology. Secondly, current V2V-enabled vehicles only broadcast their own state information. They will not broadcast what they have seen. So the non-V2V-enabled objects such as pedestrians, animals and cyclist also cannot benefit from V2V technology. Thirdly, since the V2V network is an open network, it might be vulnerable to malicious cyberattacks and the privacy is also at risk.

The US Department of Transportation is committed to the use of DSRC technologies for active safety for both V2V and vehicle-to-infrastructure (V2I) applications. DSRC supports innovation and product differentiation through the use of proprietary applications. DSRC also maintains interoperability by providing standard message sets that can be universally generated and recognized by these proprietary applications. Society of Automotive Engineers (SAE) has created SAE J2735 message sets over DSRC. SAE J2735 defines a set of V2V, V2I, V2X messages. SAE also provided a DSRC implementation guide that provided details of standardized message formats (sets, frames, elements) to support interaction in DSRC applications. SAE J2735 messages are categorized in 15 types based on their typical use. The most relevant message type for V2V-PAEB is the Basic Safety Message (BSM) that describes the operation of the sending vehicle that can affect the safety of other vehicles. BSM is used in multiple safety related applications such as Blind Spot Warning, Cooperative

Adaptive Cruise Control (ACC), and Lane Change Warning (LCW). These applications are largely independent of each other. The BMS receiving vehicles make use of the incoming stream of BSMs from surrounding (nearby) vehicles to detect potential events and dangers [17].

Except for the Vehicle-to-Vehicle communications, the Vehicle-to-Everything communications (V2X) is currently drawing more attentions from researchers. In V2X communication systems, one communication node not only be able to communicate with other vehicles, but also the traffic lights, toll gates, pedestrians, and even the owner's home. In words, anything that has DSRC device equipped can talk to other nodes in the V2X communication networks and use the shared information to do whatever can help improve safety or services [18].

Until 2015, the V2X technology is still in its infant stage and only Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) Communications technologies have been comprised. However, many researches on V2X communication have already under their way and various V2X applications will be brough out and change the world. The expected V2V mandate by the NHTSA in the U.S. will release the huge potential of the market as the new requirements for the installation of DSRC modules in new vehicles will be the first step towards wider V2V adoption. The strong regulatory support, coupled with the introduction of OEM Car-to-X technologies will increase the penetration of V2V enabling first a higher V2I penetration in new vehicles and second the Vehicle-to-Pedestrians (V2P) and Vehicle-to-Home (V2H) communications sub-markets to materialise from 2016 onwards. Towards the end of the forecast, the integration of V2V, V2I, sensors and ADAS will make autonomous driving a reality and the road safety will be improved significantly [18].

Table 3.1 shows the messages have been defined by the SAE J2735 standard. The SAE J2735 standard only defines the messages for a vehicle sender to describe the operation and state of the vehicle itself, but does not define any message for a vehicle sender to describe the information of other objects around itself. In order to send the sensory information of a PAEB system of a vehicle to other vehicles through V2V (for notify other vehicles the potential collision), a new set of V2V messages for the description of PAEB sensed objects needs to be defined. In this thesis, we will develop a new type of V2V message for sharing the sensory information among vehicles. We will describe this new V2V message in detail in section 5.

Table 3.1.: SAE J2735 defined messages.

ID	Message	Description
1	MSG_A_la_Carte	A message which is composed entirely of message elements determined by the sender for each message.
2	MSG_BasicSafetyMessage(BSM)	This message (at time referred to as the "heartbeat message") is used in a variety of applications to exchange safety data regarding vehicle state.
3	MSG_CommonSafetyRequest	This message provides a means by which a vehicle participating in the exchange of the basic safety message can unicast requests to other vehicles for addition information which it requires for the safety applications it is actively running.
4	MSG_EmergencyVehilceAlert	This message is used to broadcast warning messages to surrounding vehicles that an emergency vehicle (typically an incident responder of some type) is operating in the vicinity and that additional caution is required.
5	MSG_IntersectionCollisionAvoidance	This message deals with providing data from the vehicle to build intersection collision avoidance systems with. It identifies the intersection being reported on and the recent path and accelerations of the vehicle.
6	MSG_Map	This message is used as wrapper object to relate all the types of maps defined in the standard.

continued on next page

Table 3.1.: *continued*

ID	Message	Description
7	MSG_NMEA_Corrections	This message is used to encapsulate NMEA 183 style differential corrections for GPS radio navigation signals as defined by the NMEA (National Marine Electronics Association) committee in its Protocol 0183 standard.
8	MSG_ProbeDataManagement	This message is taken at a defined snapshot event to define RSU coverage patterns such as the moment an OBU joins or becomes associated with an RSU and can send probe data.
9	MSG_ProbeVehilceData	The probe vehicle message is used to exchange status about a vehicle with other (typically RSU) DSRC readers to allow the collection of information about typically vehicle traveling behaviors along a segment of road.
10	MSG_RoadSideAlert	This message Road side alert is used to send alerts for nearby hazards to travelers.
11	MSG_RTCM_Corrections	This message is used to encapsulate RCTM differential corrections for GPS and other radio navigation signals as defined by the RTCM (Radio Technical Commission For Maritime Services) special committee number 104 in its various standards.
12	MSG_SignalPhaseAndTiming	This message is used to convey the current status of a signalized intersection.
13	MSG_SignalRequestMessage	The Signal Request Message is a message sent by a vehicle to the RSU in a signalized intersection.
14	MSG_SignalStatusMessage	The Signal Status Message is a message sent by a RSU in a signalized intersection.
15	MSG_TravelerInformation Message	Traveler Information is designed to enable broadcast advisory messages to the vehicle driver based upon location and situation relevant information.

4. PSCAN SOFTWARE

In this study, we will not develop the real V2V-PAEB system to study the integration of V2V communication system and PAEB system. Instead we will use PreScan to develop a simulation model of the V2V-PAEB system. So the algorithms can be easily created and verified. We also will use PreScan to develop various simulation scenarios to test the V2V-PAEB simulation model. Given a specific accident scenario, it is easy to discover the cause of the accident as well as which driver support system concept could have prevented it. By changing the weather and light conditions, or by adding disturbances such as sensor noise and sensor drift, the systems robustness can be checked. So PreScan will be used extensively in this study. In this section, the general description of PreScan is presented.

PreScan is a physics-based simulation platform that is used in the automotive industry for development of Advanced Driver Assistance Systems (ADAS) that are based on sensor technologies such as radar, laser/lidar, camera and GPS. It is also used for designing and evaluating V2V communication applications as well as autonomous driving applications. PreScan provides a dedicated pre-processor (GUI) that allows users to build and modify traffic scenarios within minutes using a database of road sections, infrastructure components (trees, buildings, traffic signs), actors (cars, trucks, bikes and pedestrians), weather conditions (such as rain, snow and fog) and light sources (such as the sun, headlights and lampposts). PreScan also provides a Matlab/Simulink interface that enables users to design and verify algorithms for data processing, sensor fusion, decision making and control. In addition, a 3D visualization viewer shows a visual representation of the created scene during simulation and animation. The scene can be viewed at from multiple viewpoints. A viewpoint is a camera view specified by a camera position, orientation, view angles and zoom level [19].

Paper [20] presents how to use PreScan software to conduct simulation experiments. The V2V-PAEB simulation model works using PreScan in four easy steps. First, the requested experiment scenarios can be built using PreScans graphic user interface (GUI); second, add the sensors to vehicles and configure them properly; third, develop simulation models in PreScans engineering workspace; fourth, run the experiment and obtain the simulation result. The following sections present the components and the use of PreScan.

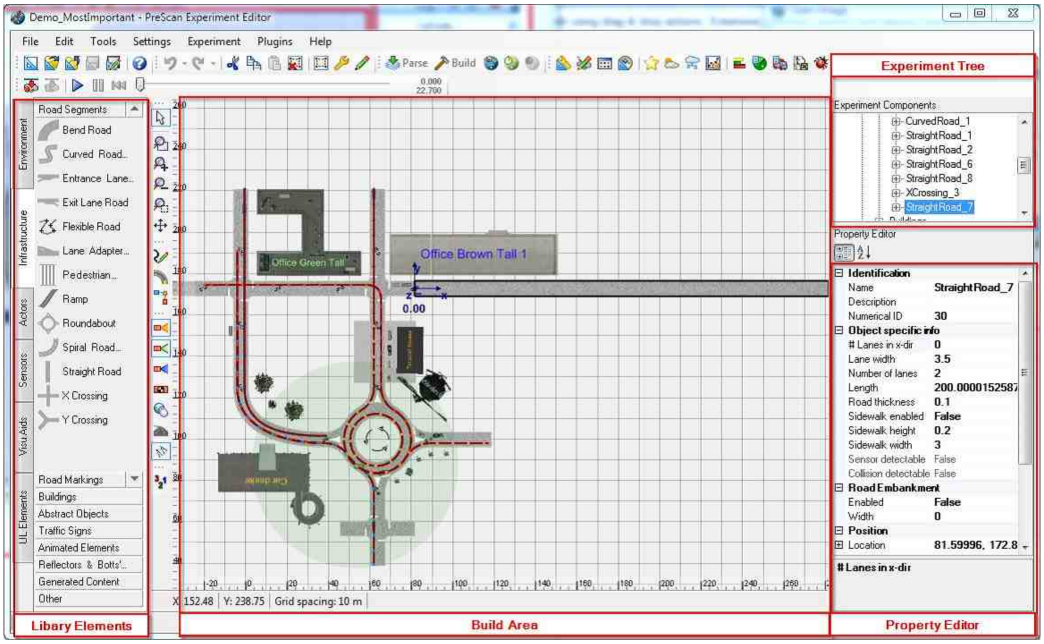
4.1 Graphical User Interface

PreScan provides a dedicated pre-processor (GUI) that allows users to build and modify traffic scenarios within minutes using a database of road sections, infrastructure components (trees, buildings, traffic signs), actors (cars, trucks, bikes and pedestrians), weather conditions (such as rain, snow and fog) and light sources (such as the sun, headlights and lampposts).

Key features to the GUI are its predefined and freely configurable library elements which enable the users to quickly build an experiment using drag and drop actions. Extensive reporting, preview and parsing mechanisms have been implemented helping the users to understand what type of experiment they have built. These features also help the users to identify discrepancies in their experiment that need to be solved before they can actually execute an experiment in the Engineering Workspace. Automatic conversion of older experiments is also taken care of by the GUI. The GUI has some distinct parts as can be seen in Figure 4.1.

Tabs on the left hand side represent various library elements available. Library elements include actors (cars, animated humans, trucks, etc.), infrastructural elements (buildings, roads, trees, etc.) and sensors (cameras, radars and lidars, etc.).

Tabs on the right of the window is a so-called experiment tree: this tree shows the relationships between elements within the experiment being built. Using this experiment tree we can quickly see what type of sensor was installed on a car and



⊟ **Identification**	
Name	**StraightRoad_7**
Description	
Numerical ID	**30**
⊟ **Object specific info**	
# Lanes in x-dir	**0**
Lane width	**3.5**
Number of lanes	**2**
Length	**200.0000152587**
Road thickness	**0.1**
Sidewalk enabled	**False**
Sidewalk height	**0.2**
Sidewalk width	**3**
Sensor detectable	False
Collision detectable	False
⊟ **Road Embankment**	
Enabled	**False**
Width	**0**
⊟ **Position**	
⊞ Location	**81.59996, 172.8**

# Lanes in x-dir

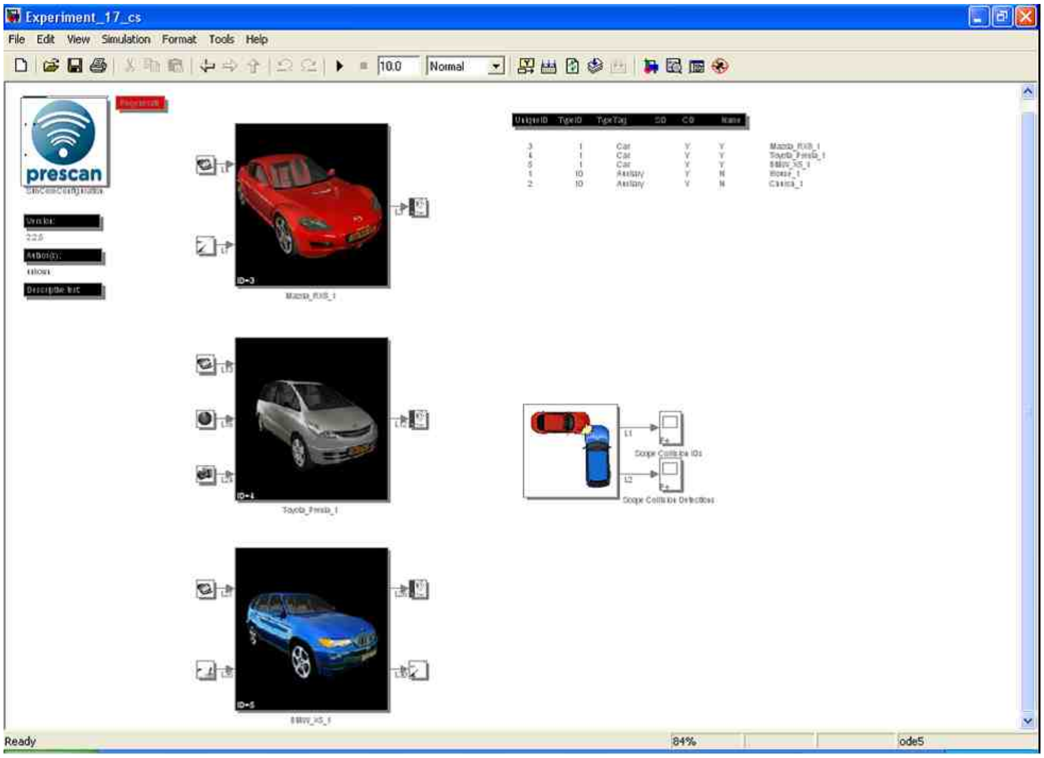
Fig. 4.1. The graphic user interface (GUI) of PreScan software.

which trajectories have been assigned to it. Information not directly needed or not directly visible in the property editor, which is just adjacent to and down below the experiment tree, can be accessed using the object configuration dialog box that is invoked by pressing the right mouse button when hovering over the object of choice.

In the middle there is the build area where the users have a top view of all elements placed in PreScans world being the prime user interface for setting up an experiment. We can directly drag the elements from the library element area and put them in the build area. This build area has an origin point of GPS coordinate and experiment axis coordinate system. Once an element is put in the build area, its GPS location as well as it coordinate in the experiment axis system is determined.

4.2 Engineering Workspace

PreScan provides a Matlab/Simulink interface that enables users to design and verify algorithms for data processing, sensor fusion, decision making and control. Simulink is developed by MathWorks. It is a graphical programming environment for modeling, simulating and analyzing multi-domain dynamic systems. In this thesis, a V2V-PAEB simulation model was created and tested using PreScan. This simulation model will be shown in detail in section 5. Figure 4.2 is a sample of the Engineering Workspace. All the actors that created in the GUI are effectively compiled into this dedicated MATLAB/Simulink Engineering Workspace.



UniqueID	TypeID	TypeTag	SD	CD	Name
3	1	Car	Y	Y	Mazda_RX8_1
4	1	Car	Y	Y	Toyota_Previa_1
5	1	Car	Y	Y	BMW_X5_1
1	10	Auxiliary	Y	N	House_1
2	10	Auxiliary	Y	N	Camera_1

Fig. 4.2. The engineering workspace of PreScan software.

The most striking elements in the Compilation Sheet are the simulation models of actors. All the simulation models of the actors that added in the GUI will be presented in the Engineering Workspace. Note that they have input and output ports. For example, the silver car in the middle has three systems on board, viz. GPS system (known as SELF port in PreScan), a sensor called AIR (idealized sensor) and a stereo camera system. On the top right there is a table telling what participants are present in the experiment and what specific top-level properties they have. Also note the block introduced in the middle: this block outputs collision detection information.

When double clicking an actor model we will find more blocks or models generated by the PreScan GUI. Figure 4.3 shows the internal blocks and models of the silver vehicle model in Figure 4.2. All blocks generated by PreScan are in gray whilst the ones inserted by the user are in default black. Next to the GPS system we also see the presence of an antenna receiver. The various messages that can be broadcast are defined in PreScan's GUI. Apart from these blocks there is also a GUI inserted trajectory and simple dynamics model. Since our V2V-PAEB simulation model a vehicle level model, it will also be added here.

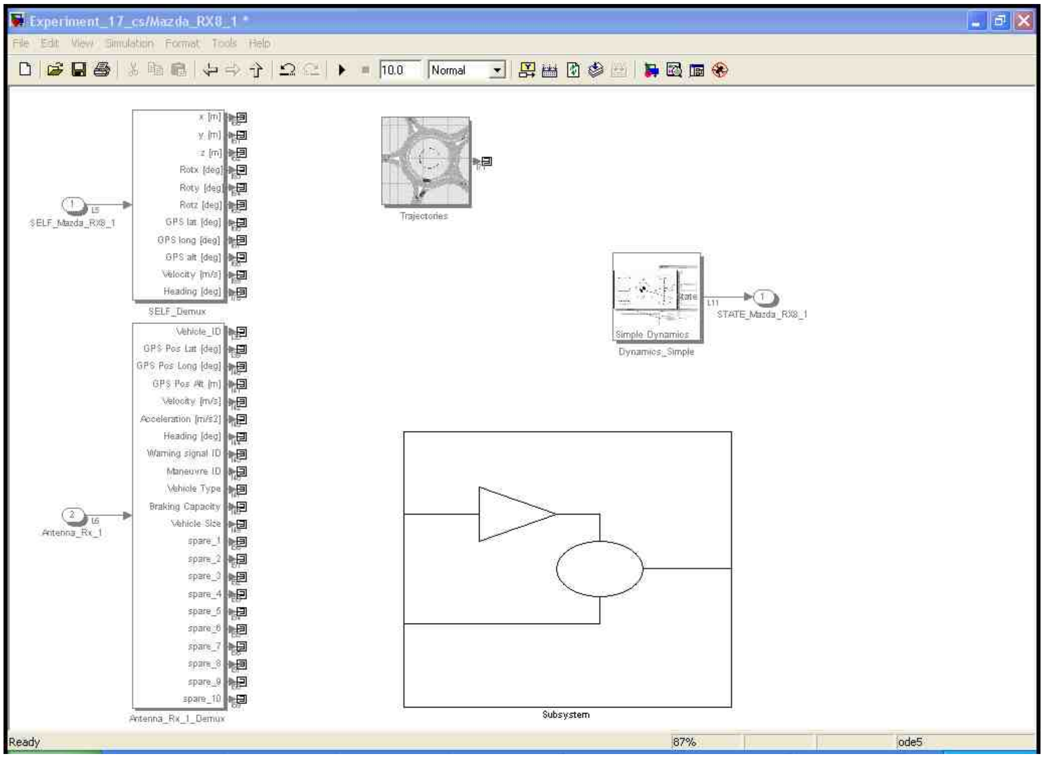


Fig. 4.3. The components of the vehicle simulation model.

4.3 The 3D Visualization Viewer

The 3D Visualization Viewer is used to visualize the experiment when the simulation is running. So the users can intuitively observe the simulation process for each experiment. Predefined viewpoints (like top view and a default scene view) are available but the users can define their own viewpoint whenever desired. The visualization Viewer comes with an intuitive navigation by using the mouse. Figure 4.4 shows an example of the simulation view captured by the 3D viewer. We can see that for the same experiment, we can have multiple viewpoints. So we do not miss the details about the running simulation. Additionally, the 3D Visualization Viewer can be used to generate movies or individual pictures of view-points selected. Individual picture formats supported include PNG and JPG. With the help of the 3D Visualization Viewer, we can record the simulation process or simulation results for further study.

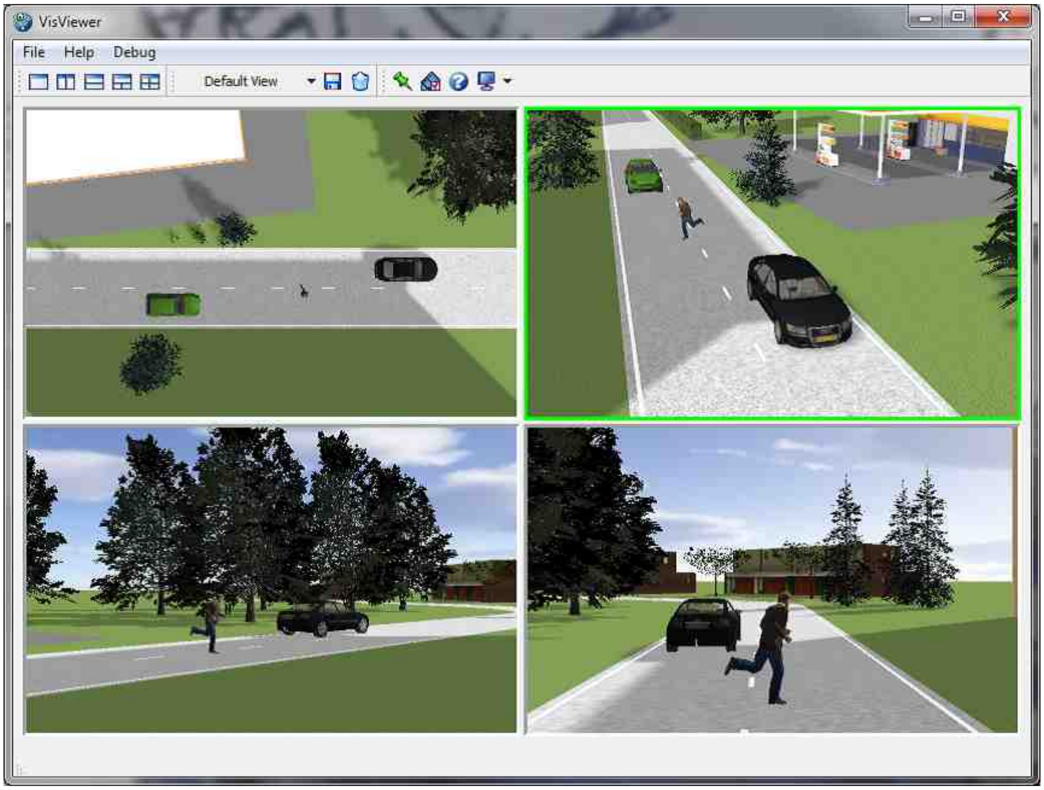


Fig. 4.4. The view captured by 3D viewer.

5. V2V-PAEB SYSTEM

As mentioned above, both the PAEB system and V2V communication systems have their limitations in improving road safety. The V2V-PAEB system is designed for further improving the road safety for pedestrians by integrating the complementary capabilities of V2V and PAEB systems together. The uniqueness of this system is that the information collected by a PAEB system on one vehicle can be shared with other vehicles in the V2V network. Comparing with the pure V2V system and PAEB system, the V2V-PAEB system has many advantages. It compensates the limitations and disadvantages of both the V2V system and PAEB system.

One limitation of the PAEB system is its short detection range. Usually the PAEB system has the max detection range of approximately 80 to 100 meters, and it will difficult for the PAEB system to detect the objects beyond its max detection range. Sometimes this detection range is not enough especially when the vehicle or the objects are moving too fast. In this case, the PAEB system will have no enough time to react to the potential collisions. By sharing the pedestrian information among the V2V network, the detection range of PAEB system can be extended significantly because the V2V system a max transmission range of approximately 1000 meters.

The PAEB system might also have sensor failures under some specific conditions. It may fail to detect the pedestrians behind obstacles. While for the V2V-PAEB system, other vehicles can tell this vehicle about this pedestrian if they have detected this pedestrian. With the help of other vehicles, this vehicle can ”see” the pedestrians behind obstacles and it will give the vehicle more time to react.

Additionally, the PAEB system might fail to detect pedestrians if weather condition or lighting condition is bad (such as thick fog and dark night). Although the sensors of V2V-PAEB system will have the same problems, the performance of PAEB system still can be improved by sharing the PAEB information. If one vehicle failed to detect a specific pedestrian, but other vehicles may detect it successfully. So the vehicle that failed to detect this pedestrian from PAEB system can "see" this pedestrian from the received V2V-PAEB Messages.

The V2V-PAEB system can benefit the road safety for non-V2P-enabled pedestrians. Just similar with V2V communication, vehicle to pedestrian (V2P) communication is also under its way. However, V2P communication always requires the pedestrians being equipped with some specific devices for sharing their information. This will be quite inflexible and costly, and the V2P technology cannot benefit the pedestrians that have no V2P devices. However, the V2V-PAEB system also can share the pedestrian information among the V2V network without any additional devices for the pedestrians. So comparing with V2P technology, the V2V-PAEB system can benefit the safety for both the V2P-enabled and non-V2P-enabled pedestrians economically.

V2V-PAEB enabled vehicle utilizes the on-board sensor systems to detected pedestrians and send the information of the detected pedestrians to the nearby vehicles through V2V communication systems. Meanwhile this vehicle can also receive such type of messages from other vehicles. So this vehicle should predict the probability of collision with the pedestrians detected from both the PAEB system and received V2V messages, and make safety related decisions.

Figure 5.1 is an simple example showing how the V2V-PAEB system works. Two vehicles are running fast on the road, but one pedestrian suddenly walks into the path of them. The V2V-PAEB system on the yellow vehicle detects this pedestrian, and then sends out a V2V-PAEB Message to report this pedestrian to the blue vehicle. At the same time, if there is a potential collision between the yellow vehicle and this pedestrian, a driver warning will be triggered. If the collision is inevitable, the automatic braking will be started. For the blue vehicle, the on board V2V-PAEB system fails in detecting this pedestrian because its view is blocked by the yellow vehicle. However, the blue vehicle receives the V2V-PAEB Message from the yellow vehicle, so the on board V2V-PAEB system still can "see" and protect this pedestrian.

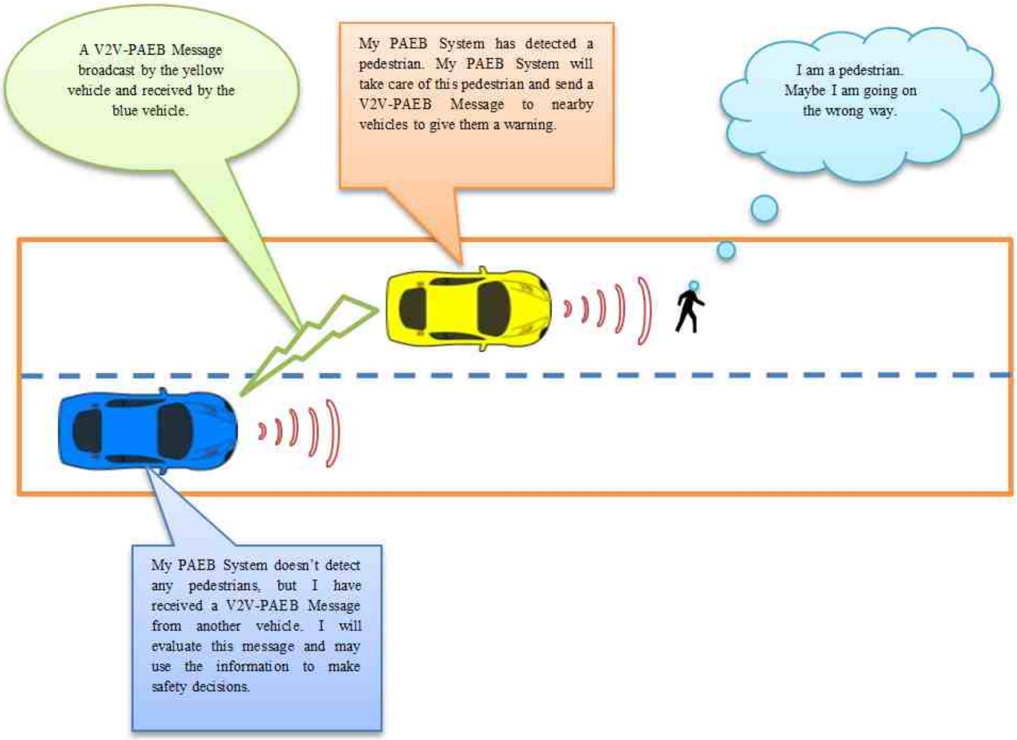

A V2V-PAEB Message broadcast by the yellow vehicle and received by the blue vehicle.
My PAEB System has detected a pedestrian. My PAEB System will take care of this pedestrian and send a V2V-PAEB Message to nearby vehicles to give them a warning.
I am a pedestrian. Maybe I am going on the wrong way.
My PAEB System doesn't detect any pedestrians, but I have received a V2V-PAEB Message from another vehicle. I will evaluate this message and may use the information to make safety decisions.

Fig. 5.1. A simple example of V2V-PAEB working process.

The V2V-PAEB model has many input parameters and output parameters. Figure 5.2 shows an example diagram of V2V-PAEB simulation model connected with its required supporting models in a vehicle model. The block in the center is the V2V-PAEB simulation model and these on both sides are the peripheral supporting models. The V2V-PAEB simulation model absorbs useful information from the models on the left side and uses this information to detect potential collisions with pedestrians and make proper safety decisions. Then the models on the right side are informed about these decisions and take proper actions to avoid or mitigate the potential collision accordingly.

V2V-PAEB system usually use various types and different numbers of sensors to detect pedestrians. Fundamentally, the sensors provide the position and motion direction of pedestrians. Currently, the V2V-PAEB simulation model supports two

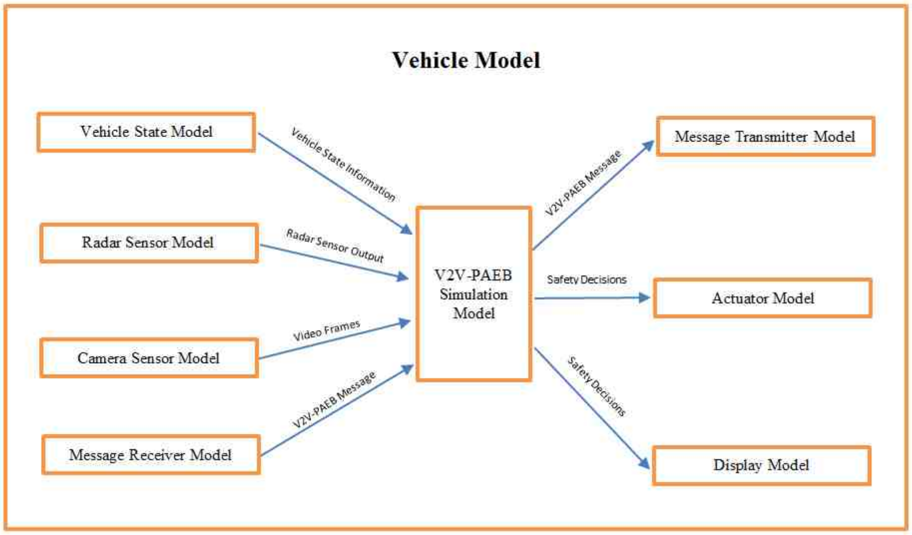


Fig. 5.2. An example diagram of V2V-PAEB simulation model.

basic sensors (one radar sensor and one camera sensor). The information processing stages of V2V-PAEB system do not change no matter what types and how many sensors are used, so sensors can be easily added to or removed from this V2V-PAEB simulation model only with minor modifications. In the real world, the sensors of the same type usually have different performance and specifications. Then the V2V-PAEB system will also have different performance on different vehicles due to the variation of sensor accuracy. So in order to study how the variation of sensor accuracy will affect the performance of V2V-PAEB system, the sensor models should be able to be configured with different performance.

The Vehicle State Model should be able to provide the host vehicle's real-time state information. The vehicle state information usually include the vehicle's speed, heading direction and GPS location and so on. The vehicle state information is used for predict potential collision with pedestrians and make proper safety decisions, so the accuracy of the state information are critical to the performance of V2V-PAEB system. For example, the pedestrian's location contained in the V2V-PAEB Message is calculated from the GPS location of vehicle. If the vehicle's GPS location is inaccurate, the pedestrian's location will also be inaccurate, and it will cause a poor performance of V2V-PAEB system. In reality, the information generated by GPS devices or other meter devices are usually have some errors. The Vehicle State Model should be able to be configured to provide information with different accuracy for us to study the influence of the inaccuracy of vehicle state information.

The Message Receiver Model is responsible for receiving V2V-PAEB Message that sent by other vehicles. V2V-PAEB Message is a type of V2V communication message that used for vehicles to share the information of pedestrians detected by their V2V-PAEB systems. This model should be able to queue the received V2V messages if multiple messages arrive at the same time. Additionally, since there is always a transmission delay and packet loss of the messages in the real world, the Message Receiver Model should provide the means for simulating such cases.

There are two types of output data generated by this V2V-PAEB simulation model: the V2V-PAEB Message and safety decisions (See Figure 5.3). The V2V-PAEB Message usually goes to the Message Transmitter Model and then being sent out to the nearby vehicles. The safety decisions usually go to both the Actuator Models to take proper actions and the Display Model for displaying the simulation process and results. For the Message Transmitter Model, the message transmission frequency should be configurable, so we can study how the message transmission interval will affect the performance of V2V-PAEB system. Although the recommended DSRC message transmission interval is 100ms, it might not be suitable for V2V-PAEB Message. That is because other DSRC messages are being sent only an event of vehicle is going to happen (such as lane changing, hard braking), it does not happen frequently, so there are not so many messages being transmitted among vehicles. However, the V2V-PAEB system will send out V2V-PAEB Message whenever the vehicle detects pedestrians. If there are too many vehicles and pedestrians in a small area, there will be a message explosion. The suitable transmission interval should be determined after doing specific simulations. While for the actuator models, they should be able to provide some basic actions such as braking, steering and accelerating, and these actions should be controllable from the V2V-PAEB simulation model.

Figure 5.3 shows the architecture and information processing stages of the V2V-PAEB simulation model. The processing is organized as a waterfall, the input information is distilled in successive stages, until finally the vehicle makes safety decisions. The Sensory Data Preprocessing stage processes raw sensory data using simple cues and fast algorithms to identify potential pedestrian candidates. This stage needs to have high detection rate even at the expense of allowing false alarms. The Pedestrian Detection stage then applies more complex algorithms to the candidates from the Sensory Data Preprocessing stage in order to separate genuine pedestrians from false alarms. In stage Track (1), the detected pedestrians are tracked overtime to get their trajectories. Once any pedestrians are detected by the on-board sensor systems of vehicle, then the Send V2V-PAEB Message stage constructs a V2V-PAEB Message and sends it to the nearby vehicles immediately. On the other hand, this vehicle may also receive multiple V2V-PAEB Messages from other vehicles. The V2V-PAEB Message Preprocessing stage periodically processes the received messages with a proper cycle to obtain the motion and state information of pedestrians contained in these messages. Then the V2V-PAEB Message Merge stage merges all the pedestrians contains in different messages together to obtain a whole set of pedestrians that detected by other vehicles. The Pedestrian Information Merge stage merges the two set of pedestrians (one from the Pedestrian Detection stage and the other one from the V2V-PAEB Message Merge stage) together to obtain a complete set of detected pedestrians surrounding the host vehicle. In stage Track (2), the pedestrians obtained from messages are also tracked overtime to get their trajectories. These trajectories from both Track (1) and Track (2) can then be sent to Collision Prediction stage for predicting the probability of collision between the host vehicle and pedestrians. In the case of high probability of collision, the driver is given an appropriate warning that enables corrective actions. If the collision is imminent, the automatic braking could also be triggered to decelerate the vehicle and reduce the severity of collision [21].

The V2V-PAEB simulation model is not able to run by itself because it is one component of the vehicle. When trying to run this V2V-PAEB simulation model,

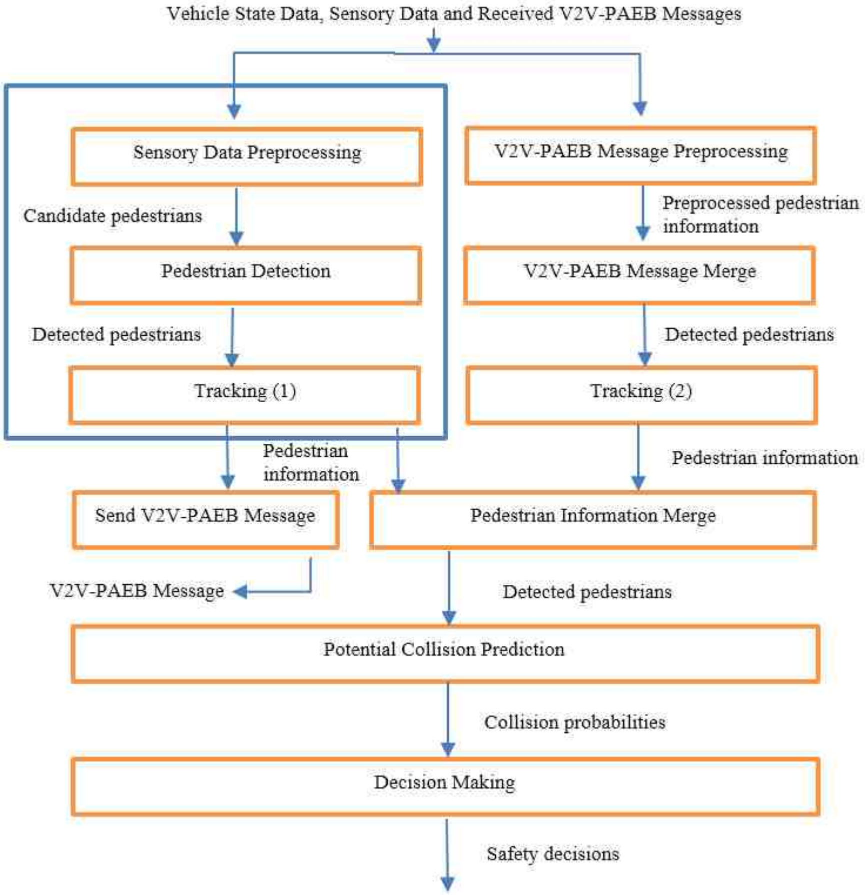


Fig. 5.3. The information processing flow of V2V-PAEB Model.

we should place it in a vehicle model and connect it with its peripheral supporting models. Usually a simulation environment where the simulation experiment can take place is also required. Some third party software (Such as PreScan, LabView and CarSim) can provide such simulation environment and models, so we do not have to develop these peripheral models as well as the simulation environment by ourselves. When using this V2V-PAEB simulation model, we should firstly use the software to generate the simulation experiment, the vehicle models as well as the sensor models. Then put the V2V-PAEB simulation model in the vehicle model and connect it to the models such as radar model and actuator models. After adding algorithms to the V2V-PAEB model, then this experiment is ready to run. Chapter 6 presents how to test the V2V-PAEB in the PreScan simulation software in detail.

5.1 Inputs of V2V-PAEB Model

Similar with PAEB system, the V2V-PAEB system may also have various type of input parameters. In this thesis, we are trying to develop a simulation model of V2V-PAEB system rather than a real V2V-PAEB system. There will be some inputs used only for simulation purpose, and a real V2V-PAEB system will not have such type of inputs. In this section, we will generally discuss the possible input parameters of the V2V-PAEB simulation model. Additionally, we will also describe the example input parameters we have implemented in detail.

5.1.1 Possible Input Parameters of V2V-PAEB Model

Input Parameters from Sensors

Section 2 has presented the commonly used sensors in PAEB system. Each type of sensor has its advantages and limitations, so the PAEB system may use a combination of different types and number of sensors to detect pedestrians robustly. For the V2V-PAEB system, it would be the same story. Usually, different types of sensors have different capabilities. Even for the sensors of the same type, their capabilities and performances can be implemented differently by different manufacturers. For example, there are many types of vision cameras can be used for pedestrian detection, they may generate videos/images with different resolutions. In addition, most of the commonly used cameras can only generate raw video/images. In this case, the V2V-PAEB simulation model has to use image processing to detect pedestrians from the camera data. However, some high-end camera systems can generate both the raw data and the processed data. For example, Mobileye is a technology company that develops vision-based advanced driver assistance systems (ADAS) providing warnings for collision prevention and mitigation. The firm's pedestrian detection technology is based on the use of mono cameras only, using pattern recognition and classifiers with image processing and optic flow analysis. Both static and moving pedestrians can be detected to a range of approximately 30m using VGA resolution cameras [22]. The camera system provided by Mobileye can identify the type of detected objects, such a vehicles and pedestrians. Whats more, the location, speed and trajectory of the detected objects can also be determined. If the Mobileye styled camera is used in the V2V-PAEB system, the image processing is not essential.

PreScan software provides the simulation models for the commonly used sensors used in PAEB system. The capability and performance of these simulation models of sensors can be configured, and they can provide the requested data to the V2V-PAEB simulation model, so we can focus on developing the internal algorithms of V2V-PAEB simulation model, and do not need to waste time to developing the supporting sensor models.

Input Parameters from DSRC Receiver

A DSRC Receiver is used for receiving V2V messages from other vehicles. In this thesis, we use V2V-PAEB Message to share the information of detected pedestrians among the vehicles. Ideally, there is no latency and packet loss when transmitting the V2V messages in the DSRC network. However, there is always an uncertain time delay and packet loss rate in the real world. Especially when there are too many vehicles in the network or the transmission channel is interfered. Generally, a higher transmission latency or packet loss rate can reduce the performance of V2V-PAEB system. In the future, we will study how the transmission latency and packet loss will affect the performance of V2V-PAEB system.

PreScan provides us with a DSRC receiver model and we can use it to receive V2V-PAEB Messages in our V2V-PAEB system. It also provides us with the convenience for simulating the transmission delay and packet loss.

Vehicle State Information

The V2V-PAEB system also require the vehicle own state information when generating the V2V-PAEB Message, detecting potential collisions with pedestrians and making safety decisions. The vehicle state information usually includes its location, speed, heading direction, acceleration, throttle state and brake state and so on. Currently, most of the vehicle state information can be easily obtained with desired accuracy except for the location information of vehicle. In the V2V-PAEB system, the location of vehicles and the detected pedestrians are described using GPS coordinates. The Global Positioning System (GPS) can provide the location, altitude, and speed with near-pinpoint accuracy, but the system has intrinsic error sources that have to be taken into account when a receiver reads the GPS signals from the constellation of satellites in orbit.

U.S. government is committed to providing GPS to the civilian community at the performance levels specified in the GPS Standard Positioning Service (SPS) Performance Standard. For example, the GPS signal in space will provide a ”worst case” pseudo-range accuracy of 7.8 meters at a 95% confidence level. The actual accuracy users attain depends on factors outside the government’s control, including atmospheric effects, sky blockage, and receiver quality. Real-world data from the FAA show that their high-quality GPS SPS receivers provide better than 3.5 meter horizontal accuracy. Higher accuracy is attainable by using GPS in combination with augmentation systems. These enable real-time positioning to within a few centimeters, and post-mission measurements at the millimeter level [23].

In the V2V-PAEB system, the locations of the shared pedestrians are calculated based on the GPS location of host vehicle. On the receiver side, these GPS location of pedestrians will be converted into vehicles local coordinate system for predicting potential collisions and making safety decisions. If the GPS location of vehicle is not accurate, the location of pedestrians will be also inaccurate, so the accuracy of GPS sensor is critical to the performance of V2V-PAEB system. In this thesis, we will study the influence of the inaccurate of PGS location to the performance of V2V-PAEB system. Additionally, we will also study how to calibrate the inaccurate locations of the pedestrians obtained from the received V2V-PAEB Messages. PreScan software provides the model of GPS sensor, and its accuracy of output data also can be configured.

Global Variables and Simulation Configurations

In order to study the performance of V2V-PAEB system under different conditions, the V2V-PAEB simulation should provide the interface to be configured with different capabilities. For example, currently only some high-end vehicles have PAEB systems and V2V communication systems are still under developing and testing. So in the real world, most of the vehicles on the road have neither PAEB systems nor V2V communication systems. In this study, we should study how the coexisting of vehicles with different capabilities will affect the performance of V2V-PAEB system. However, we do not have to develop different simulation models. We can configure the V2V-PAEB simulation model with different capabilities (V2V only, PAEB only, V2V-PAEB, non-V2V and non-PAEB) instead. We can use global variables to configure the V2V-PAEB model with different capabilities.

5.1.2 Example Input Parameters of V2V-PAEB Model

Since V2V-PAEB simulation model may have various possible inputs, it is impossible to present all of them in this thesis. Additionally, although the V2V-PAEB simulation model is designed to be able to accept any type of inputs, current implementation of this model only uses some typical inputs. Similar with previous section, currently the V2V-PAEB simulation model accepts four types of information from the following models:

1. Vehicle State Model.

   The Vehicle State Model provided by PreScan can generate the real-time state information for the host vehicle. As can be seen in Table 5.1, this model uses both global experiment axis system and GPS coordinate to describe the location of host vehicle.

   PreScan provides a global experiment axis system for positioning the vehicles and pedestrians. The origin of this experiment axis system is determined when the user creates this experiment in PreSans GUI. Each vehicle and pedestrian in the simulation experiment is assigned coordinate (X, Y, Z) to describe its position. Comparing with GPS, the global axis coordinate system is accurate. The definition of the global coordinate axis system is defined in Figure 5.4.

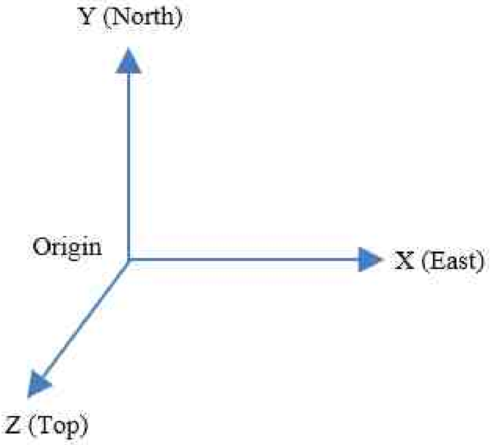


Fig. 5.4. The global coordinate system defined in PreScan.

   In the real world, we do not have the global axis coordinate system, so we use GPS instead. We should specify the GPS location of this experiment when we creating this simulation experiment using PreScans GUI. Then each vehicle and pedestrian will be assigned a GPS location automatically.

Table 5.1. The input parameters from Vehicle State Model.

Item	**Descriptions**
X[m]	The X coordinate of the vehicle in global experiment axis system.
Y [m]	The Y coordinate of the vehicle in global experiment axis system.
Z [m]	The Z coordinate of the vehicle in global experiment axis system.
Latitude [deg/min/sec]	The GPS Latitude position of the host vehicle. It is represented in degrees, minutes and seconds.
Longitude [deg/min/sec]	The GPS Longitude position of the host vehicle. It is represented in degrees, minutes and seconds.
Altitude [m]	The GPS Altitude position of the host vehicle.
Rot X [deg]	The x-rotation of the vehicle with respect to experiment axis system.
Rot Y [deg]	The y-rotation of the vehicle with respect to experiment axis system.
Rot Z [deg]	The z-rotation of the vehicle with respect to experiment axis system.
Yaw Rate [deg/s]	The Yaw (Turning) rate of the vehicle.
Velocity [m/s]	The moving velocity of the vehicle.
Heading Direction [deg]	The moving direction of the vehicle. North is 0 degree. This value is range from 0 to 360 in clockwise direction.
Acceleration [m/s^2]	The acceleration of vehicle. Positive for acceleration and minus for deceleration.
Throttle State [%]	The throttle state of vehicle. This value is range from 0 to 100. 0 means no throttle is applied and 100 percent means full gas throttle is applied.
Brake State [%]	The brake state of vehicle. This value is range from 0 to 100. 0 means no brake is applied and 100 percent means full brake is applied.
Steering Angle [deg]	The steering Angle of vehicle. Positive for clockwise and minus for anticlockwise.

2. Radar Sensor Model.

   Table 5.2 shows the input parameters from Radar Sensor Model. The Radar Sensor Model can generate the processed data of detected objects. In PreScan, the output signal of Radar sensor contains at most 32 signals by default, meaning that up to 32 objects are reported. Usually, the processed data includes the location and speed information of the detected objects. And they are described using the sensor coordinate system. Figure 5.5 shows the definition of the radar sensor coordinate system.

Table 5.2. The input parameters from Radar Sensor Model.

Item	**Descriptions**
Beam ID [-]	The radar sensor can be configured with different number of beams with different angle coverages. This Beam ID is used for indicating which beam is in active in the current simulation time step.
Range (R) [m]	The distance between the radar sensor and the detected objects. The Range is defined in Figure 5.5.
Doppler Velocity [m/s]	Velocity of target point, relative to the sensor, along the line of-sight between sensor and target point.
Theta (θ) [deg]	Azimuth angle in the sensor coordinate system at which the target is detected. The Azimuth angle is defined in Figure 5.5.
Phi (ϕ) [deg]	Elevation angle in the sensor coordinate system at which the target is detected. The Elevation angle is defined in Figure 5.5.
Target ID [-]	Numerical ID of the detected target.
Energy Loss [dB]	Ratio received power / transmitted power.
Alpha (α) [deg]	Azimuthal incidence angle of the sensor on the target object.
Beta (β) [deg]	Elevation incidence angle of the sensor on the target object.
TIS Data [-]	An array signal contains all of the sensors output.
Doppler Velocity X/Y/Z [m/s]	Velocity of target point, relative to the sensor, along the line of-sight between sensor and target point, decomposed into X, Y, Z of the sensors coordinate system.
Target Type ID [-]	This is used for specifying the type of the detected object if the radar can identify the type of detected objects. The types are defined in Table 5.5.

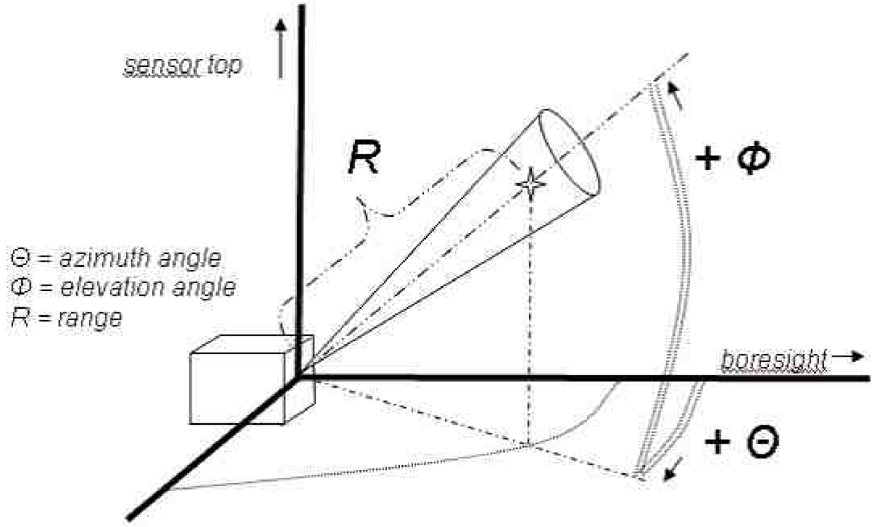


Fig. 5.5. Definitions for range, azimuth and elevation.

3. Camera Sensor Model.

   Table 5.3 shows the input parameters from Camera Sensor Model. The Camera Sensor Model can generate both the raw video/images and the processed data to the V2V-PAEB model. The V2V-PAEB model can use the raw video/images to image processing for pedestrian detection. Since the camera sensor can also identify the type of detected objects and their information such as location and speed, the V2V-PAEB model can use the processed data directly. Just the same as radar sensor, the objects detected by the camera sensor are also described in the sensor coordinate system.

   The output signal of the camera sensor model always contains 32 signals, meaning that up to 32 objects are reported. If fewer objects are detected, the unused signals are reported as 0. If more objects are present on the sensor's view, those objects will not be part of the sensor output signal. Readings being output are not sorted. If N objects are detected, they are reported as the first N out of 32 signals, but the data set output itself is not in a specific order.

Table 5.3. The input parameters from Camera Sensor Model.

Item	**Descriptions**
Object ID [-]	Numerical ID of the detected object. It is assigned by the camera sensor.
ObjectTypeID [-]	The Type ID of the detected object. If this object is identified successfully, it will be assigned with an Type ID. The types are defined in Table 5.5.
Width [pixel]	Width of the object's strict bounding rectangle, as a fraction of the screen width.
Height [pixel]	Height of the object's strict bounding rectangle, as a fraction of the screen height.
Range [m]	Range at which the target object has been detected. The distance to the nearest point is returned. The Range is defined in Figure 5.5.
RangeX [m]	The X component of the Range, in sensor coordinates.
RangeY [m]	The Y component of the Range, in sensor coordinates.
RangeZ [m]	The Z component of the Range, in sensor coordinates.
DopplerVelocity [m/s]	The velocity of target point, relative to the sensor, along the line-of-sight between sensor and target point.
Doppler Velocity X/Y/Z	The velocity of target point, relative to the sensor, along the line-of-sight between sensor and target point, decomposed into X, Y, Z of the sensor's coordinate system.
Theta (θ)[deg]	Azimuth angle in the sensor's coordinate system at which the target is detected. The Azimuth angle is defined in Figure 5.5.
Phi (ϕ) [deg]	Elevation angle in the sensor's coordinate system at which the target is detected. The Elevation angle is defined in Figure 5.5.
Video Frame	The Camera Sensor Model should be able to generate video frames at a proper frame rate. There is no requirement for the size and color depth of the frames.

4. DSRC Receiver Model.

   The DSRC Receiver Model can receive DSRC messages sent from other vehicles, and then put the received message into a buffer and feed to the V2V-PAEB model. Table 5.4 presents the format of the V2V-PAEB Message. This message consists of two parts, the vehicle information and pedestrian information. The vehicle information contains the vehicle information such as its position, speed, heading direction and so on. The vehicle information contains the information of detected pedestrians such as their position, speed and size and so on. In the future, the information of vehicle except for the sender vehicle id should be eliminated because it is duplicated with BSM messages. However, since the BSM messages are currently not implemented in the V2V-PAEB simulation model, so we integrate the BSM message to the V2V-PAEB Message.

   Table 5.5 presents the type of objects defined in PreScan simulation environment. Currently, there are totally 17 types of objects defined. These object type IDs are usually used by the PAEB system and V2V system to identify a detected object. In this study, the most commonly used actors are the Car, Motor, and Truck/Bus and Human.

   The variables for simulation environment set up is another type of input parameters for the V2V-PAEB simulation model. Table 5.6 shows a list of the simulation environment set up variables and configurations. These variables should be configured properly before each simulation run. These variables are saved into the model once being initialized.

5.2 Outputs of V2V-PAEB Model

Table 5.7 shows the output parameters of the V2V-PAEB simulation model. These output parameters will go to the actuator components of vehicle to take proper actions. Since the V2V-PAEB simulation model is developed using PreScan software,

Table 5.4. The Formation of V2V-PAEB Message.

Item	**Descriptions**
Vehicle Information	
Temporary ID [-]	The temporary ID used to identify this vehicle in the V2V network.
Event Time [UTC]	The event time of this message.
Vehicle Size [-]	The size of this vehicle.
Latitude [deg/min/sec]	The GPS latitude coordinate of vehicle.
Longitude [deg/min/sec]	The GPS longitude coordinate of vehicle.
Altitude [m]	The GPS altitude coordinate of vehicle.
Heading Direction [deg]	The vehicles heading direction. North is 0 degree and go clockwise to 360.
Speed [m/s]	The moving speed of this vehicle.
Acceleration [m/s^2]	The acceleration of this vehicle.
Brake System Status	The status of the brake system of vehicle.
Pedestrian Information	
Number of pedestrians [-]	The number of pedestrians contained in this message.
ID [-]	The sender assigned ID to the pedestrians.
Confidence [%]	This value is used to described how confident this pedestrian is identified.
Size [-]	The size of the pedestrian.
Color [-]	The color of the pedestrian.
Latitude [deg/min/sec]	The GPS latitude coordinate of the pedestrian.
Longitude [deg/min/sec]	The GPS longitude coordinate of the pedestrian.
Altitude [m]	The GPS altitude coordinate of the pedestrian.
Speed [m/s]	The moving velocity of the pedestrian.
Heading Direction [deg]	The moving direction of the pedestrian. North is 0 degree. This value is range from 0 to 360 in clockwise direction.
...	... The information for the second, third, and rest of detected pedestrians.

and PreScan provides actuator models. We use the PreScan provided actuator model, so the V2V-PAEB model does not have its own actuator components. The typical actions are including driver warning, automatic braking, and automatic steering. For example, if the Driver Warning Flag is set, the vehicle probably will trigger a warning

Table 5.5. The Definitions of Object Type IDs in PreScan.

Object Type ID	**Description**
1	Car
2	Motor
3	Truck/Bus
4	Human
5	Calibration element
6	Trailer
7	Actors other
8	Road (segment)
9	Building
10	Nature elements
11	Traffic sign
12	Animated traffic sign
13	Abstract object
14	Underlay
15	Infra other
16	Static other
17	Moving other

to warn the driver by sound, light or vibration. If the Automatic Braking Flag is set, the vehicle will apply Brake Pressure to the brake systems and then the vehicle starts to decelerate. Some approach supports automatic steering when the braking is going on. In this case, Automatic Steering Angle is used to control the steering state of vehicle. In addition, if any pedestrians have been identified, the vehicle then constructs a V2V-PAEB Message and then the Message Transmitter Model sends this message out.

5.3 Detail Description of the V2V-PAEB System

This section describes the architecture and the information processing process of V2V-PAEB simulation model in details. The implementation of the proposed V2V-PAEB simulation model is divided into 10 stages. Each stage is designed to solve

Table 5.6. The Simulation Configurations of V2V-PAEB Simulation Model.

Item	**Description**
Vehicle cut-off speed [km/h]	The highest speed limit of the vehicle.
Max Braking Pressure [bar]	The max pressure can be applied when braking.
Braking System Delay [ms]	The delay of the braking system.
Lane Width [m]	The width of the lane.
Radar Max Range [m]	The max range that the radar sensor can detect.
Radar Max Theta [deg]	The max theta angle of the radar sensor.
Radar Frequency [Hz]	The sample frequency of the radar sensor.
Max Num. of Radar Detected Objects	The max number of objects that the radar sensor can detect.
Max Num. of Camera Detected Objects	The max number of objects that the camera sensor can detect.
Camera Sampling Rate	The frame rate of the camera sensor.
Compilation Sheet Frequency [Hz]	The simulation frequency of this experiment.
EGO Parameters Visualization	This is used for specify the update frequency of visualization components.
Max V2V Message Length	The max length of the V2V-PAEB Message.
Max Num. Vehicles Supported	The max number of vehicles that can be equipped with the V2V-PAEB simulation model.
V2V-PAEB Message Frequency	The broadcasting frequency of the V2V-PAEB Message.
Vehicle ID	The unique ID that used for identify a specific vehicle.
Vehicle Type	This is used for specifying the type of current host vehicle. (Etc. Sedan, SUV, or Truck).
Vehicle Color	This is used for specifying the color of current host vehicle.
Vehicle GPS Accuracy	This is used for specifying the GPS accuracy of the host vehicle.
Vehicle Capability	This is used for defining the capability of this vehicle. 1=non-V2V and non-PAEB; 2= V2V only; 3=PAEB only; 4=V2V-PAEB.

some specific problems. Each stage is represented by a block in Figure 5.3. So when describing these stages, the author will firstly presents the ultimate goal of this stage, and what problems need to be solved at this stage. Then, the author describes how

Table 5.7. The output of V2V-PAEB simulation model.

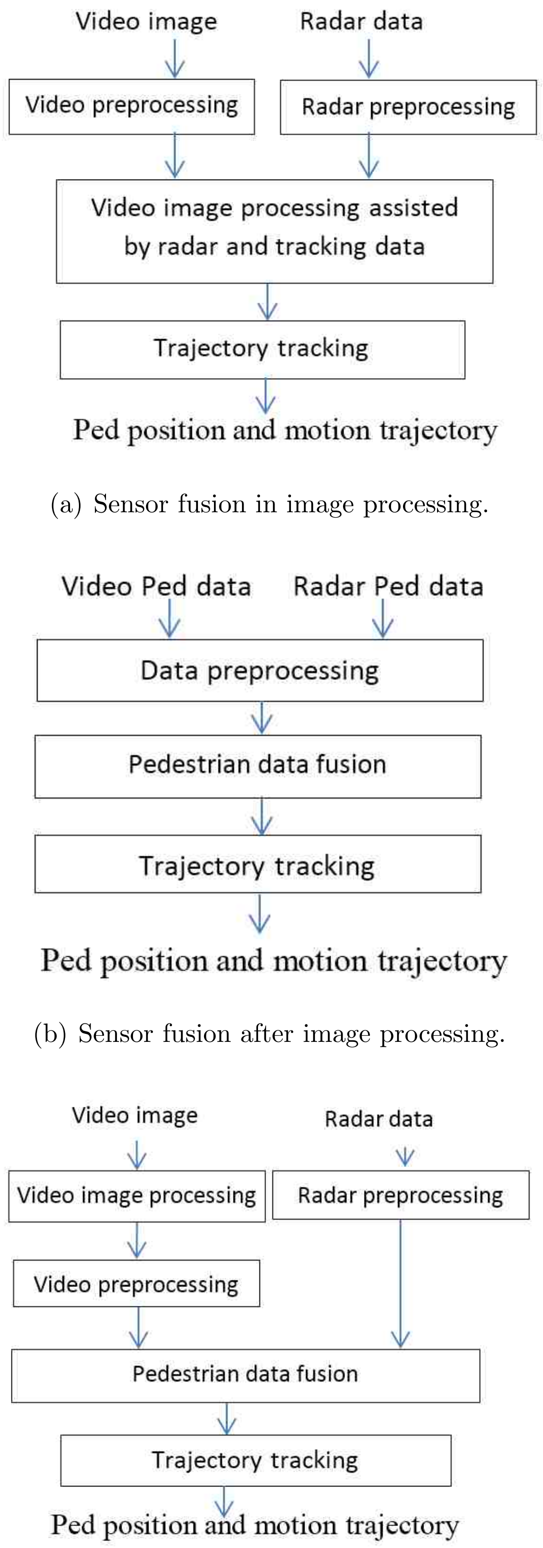
Object Type ID	**Description**
Pedestrian Detection Flag [Y/N]	This flag is used to indicate whether any pedestrians have been detected. If so, this parameter will be set to Y. Otherwise, it will be set to N.
Driver Warning Flag [Y/N]	This flag is used to indicate if the driver warning should be triggered. If this parameter is set to Y, then a driver warning will be triggered immediately. Otherwise, if this parameter is set to N, then the vehicle will do nothing.
Automatic Braking Flag [Y/N]	This flag is used to indicate if the automatic braking should be started. If it is set to Y, then the automatic braking will be started immediately. Otherwise, if this parameter is set to N, then the vehicle will do nothing.
Brake Pressure [bar]	This value is used to control the deceleration of the vehicle when the automatic braking is started. Once the Automatic Braking Flag is set to Y, then the Brake Pressure should be assigned a value between zero and the Max Braking Pressure. Otherwise, if the Automatic Braking Flag is set to N, then this parameter should be zero.
Automatic Steering Flag [Y/N]	This flag is used to indicate if the automatic steering control should be started. If it is set to Y, then the automatic steering will be started immediately. Otherwise, if this parameter is set to N, then the vehicle will do nothing.
Automatic Steering Angle [deg]	This value is used to specify the steering wheel status. A positive value means turning right, and minus value means turning left. Once the Automatic Steering Flag is set to Y, then this parameter should be assigned with a meaningful value. Otherwise, it should be zero.
Time To Collision [s]	Time to Collision represents for how much time left before the collision occurs. If many pedestrians are detected at the same time, and each one has its TTC. Then this parameter will be set to the smallest of them.
V2V-PAEB Message [array]	The V2V-PAEB Message that contains the pedestrians that detected by PAEB system on vehicle. Vehicles use this message to share the pedestrian information through V2V communication.

this stage has been actually implemented. The purpose of this thesis is proposing a simulation tool for the development of V2V-PAEB system rather than providing a complete simulation model of a V2V-PAEB system. Users using this simulation tool should develop their own model utilizing the proposed simulation tool. Since there are many open questions on the implementation of each stage, the models of some stages provided by this tool are primitive. The best solutions for some stages will require long term research or provided by the V2V-PAEB developer.

The blue block in Figure 5.3 is the pedestrian detection part of PAEB. The output of this block is the detected pedestrians and their position and motion trajectories (position and direction of each pedestrian in global coordinates at a specific time and motion trajectory function).

There are many possible approaches in pedestrian detection, which depends on the available input. The variation of the input from the radar is assumed to be a list of objects detected by the radar and their radar cross section values (or object types) for all radar included to a PAEB. The variation of the input of a camera can be sequence of image frames, or a list of objects identified from image processing. Figure 5.6 shows main three possible PAEB simulations structures corresponding to input information. More variations can be developed with variations based on the number of input sensors and additional type of input sensors. However, the data format of the output of PAEB block should be the same for all input variations.

The pedestrian tracking data from previously identified pedestrian trajectories can be used for sensor fusion. The preprocessing blocks are interfacing block to translate the data to the format needed by next block. The trajectory tracking blocks of all three different PAEB simulation approaches have the same input and output.



(a) Sensor fusion in image processing.

(b) Sensor fusion after image processing.

(c) Sensor fusion with processed image data input.

Fig. 5.6. Three main possible ways of PAEB simulation approaches.

5.3.1 Sensory Data Preprocessing

Goal and Problems Description

The goal of this Sensory Data Preprocessing stage is processing raw sensory data using simple cues and fast algorithms to identify potential pedestrian candidates, or preparing the input sensory data in well-organized format. As we have mentioned above, there are many possible types of sensors can be used by PAEB system. So the Sensory Data Preprocessing may have different structures according to different types of input information. Additionally, different types of sensors usually provide different type of information. Even the sensors of the same type can be implemented differently and provide different capabilities and output information. The information may not in the desired format, or some of the information is not needed, so this block should be designed as an interface block that can handle whatever types of sensory input information and organize the input data in desired mode and prepare it for future use.

Possible PAEB Simulations Structures

As mentioned in Section 2.2, various types of sensors can be employed for the pedestrian detection systems in vehicle, so the input parameters of this stage can vary in different approaches. Commonly used sensors for detecting pedestrians are camera sensors in various configurations using visible light and infrared (IR) radiation, as well as Radar and Laser sensors. Every sensor has its advantages and limitations. Even for the same type of sensor, the performance and capabilities can be implemented differently. In order to enhance the advantages and overcome the limitations, one can use a combination of multiple sensors that give complementary information.

Figure 5.6 shows main three possible PAEB simulations structures corresponding to different types of input information. These structures have considered the three stages described in Figure 5.3 together, so if a different structure is applied to these three stages, the goals and problems to be solved for each stage will be different. The following paragraphs describe these structures in detail respectively. Since there are too many different combination of sensors can be applied in PAEB system, so for simplicity, we will take only two sensors (one radar sensor and one camera sensor) for example when we discuss the following structures.

1. Sensor Fusion in Image Processing

   In this structure, the radar data will be used for assisting the image processing to detect pedestrians. This Sensory Data Preprocessing will firstly preprocess the sensory data separately for each type of sensor. Then use sensor fusion algorithms to obtain a list of candidate pedestrians, and provide these candidate pedestrians to the next stage Pedestrian Detection to final identification. If this structure is used, then two types of input parameters are required: (1) the input data from radar sensor; (2) the video image generated by camera sensor.

   The Radar Preprocessing block will do the preprocessing for radar input. Radar sensor can detect many objects at a time. Usually, the location and speed of the detected objects are determined and sometimes the type of objects also can be identified. In this block, filters can be applied to eliminate the objects that the PAEB system is not interested in. The Video Preprocessing block will finish the preprocessing for video images. Different cameras usually generate video images with different format or quality, so in this block, the input video images should be converted to the desired format.

   The radar sensor can provide a list of potential objects with their motion and location information but it may not tell whether the detected objects are pedestrians or not with certainty. The monocular camera provides the video frames that may contain many pedestrians, but finding out all the pedestrians by searching a whole frame is extremely complex and time consuming. Even pedestrians are classified from the video frame, the pedestrians motion and location information still cannot be obtained by doing image processing. So in this stage, the developers should use some fast sensor fusion algorithms to project each target detected by radar sensor into the areas of the video frame to obtain a set of Region of Interest (ROI). Each ROI is paired with a target detected by the radar sensor. And these pairs will be transmitted to the next stage.

In this stage, two more issues should be considered. One is the output frequency of the radar sensor and camera sensor. The output frequencies for radar sensor and camera sensor are usually different. In this stage, the data from the radar sensor and camera sensor should be synchronized. The second is the mounting locations of sensors on the vehicle. When doing sensor fusion, the locations of radar sensor and camera sensor should also be considered. Since both the radar sensor and camera sensor have their own coordinate systems, they should be transformed into the same coordinate system when doing sensor fusion.

2. Sensor Fusion after Image Processing

   In this structure, the sensor fusion for the radar sensor and camera sensor will be after the image processing. The Sensory Data Preprocessing Stage will firstly preprocess the sensory data separately for each type of sensor. However, no sensor fusion is done in this stage. The Pedestrian Detection Stage will do the sensor fusion to obtain the robust verified data of detected objects. If this structure is used, then two types of input parameters are required: (1) the input data from radar sensor; (2) the video image generated by camera sensor.

   The Radar Preprocessing block will do the preprocessing for radar input. Radar sensor can detect many objects at a time. Usually, the location and speed of the detected objects are determined and sometimes the type of objects also can be identified. In this block, filters can be applied to eliminate the objects that the PAEB system is not interested in. Additionally, different types of radar sensors usually generate output information at different frequencies, so this stage should also be able to handle the variation of the frequency of the input information from sensors.

The Video Preprocessing block will finish the preprocessing for video images. Different cameras usually generate video images with different format or quality. In this block, the input video images should be converted to the desired format. Additionally, the stage usually processes raw camera data using simple cues and fast algorithms to identify potential pedestrian candidates. This stage needs to have high detection rate even at the expense of allowing false alarms. The Pedestrian Detection stage then applies more complex algorithms to the candidates from the Sensory Data Preprocessing Stage in order to separate genuine pedestrians from false alarms.

3. Sensor Fusion with Processed Image Data Input

   In this structure, the camera sensor is required to generate processed data, not only the video image. Similar with the radar sensor, the camera sensor that has data processing capability usually can identify the type of detected objects, and obtain their location, speed.

   The goal of this Sensory Data Preprocessing Stage is preprocessing the input data from radar sensor and camera sensor separately, and preparing the input data for future use. For example, it is possible that different camera sensors or radar sensors have different capabilities and different output information. Sometimes the data generated by sensors is not in the desired format, so this stage is responsible for reorganizing this data into desired format. Additionally, some filters can be applied to the input data from radar sensor and camera sensor to eliminate the unnecessary information.

Table 5.8 shows the input parameters for the Sensory Data Preprocessing Stage for structure C described in Figure 5.6. Since the camera sensor can provide the processed data, the image processing based pedestrian classification is not essential. When the simulation test is running, these parameters will be periodically provided to this stage.

Table 5.8. The Input Parameters of Sensory Data Preprocessing Stage.

Item	**Description**
Radar Data	The radar data is described in Table 5.2.
Camera Data	The camera data is described in Table 5.3. If we use the processed camera data, then the video image is not necessary.

Table 5.9 shows the output parameters for the Sensory Data Preprocessing Stage. These output parameters will be fed to the Pedestrian Detection Stage. Preprocessed Sensory Data is actually a big two dimensional array. And the size of each item in this table is Max Num. of Radar Detected Objects plus Max Num. of Camera Detected Objects. If less objects are detected, the unused signals are reported as 0. If N objects are detected, they are reported as the first N elements if this array.

Current Implementation

Figure 5.7 shows how this study implemented this stage. Currently we have implemented the third case of the main three possible PAEB simulations structures described in Figure 5.6. This stage accepts the processed data from radar sensor and camera sensor. Then it will preprocess these two sets of input sensory data separately. Since both the Radar Sensor Model and Camera Sensor Model can generate the processed data in the format we need, we do not need to reorganize the format of the input data. However, in this study, we configure the radar sensor and camera with different output frequencies. The output frequency of the radar sensor is 25 Hz and the output of the camera is 30 Hz. We need to synchronize the information from sensors before using them. PreScan provides us a library function for transmitting the frequency of input data to the desired frequency.

Table 5.9. The Output Parameters of Sensory Data Preprocessing Stage.

Item	**Description**
Object ID [Num]	The id used for identifying this object.
Object Type ID [Num]	The Type ID of the detected object. The types are defined in Table 5.5.
Object Range [m]	Range at which the target object has been detected. The distance to the nearest point is returned.
RangeX [m]	The X component of the Range, in sensor coordinates.
RangeY [m]	The Y component of the Range, in sensor coordinates.
RangeZ [m]	The Z component of the Range, in sensor coordinates.
DopplerVelocity [m/s]	The velocity of target point, relative to the sensor, along the line-of-sight between sensor and target point.
Doppler Velocity X/Y/Z [m/s]	The velocity of target point, relative to the sensor, along the line-of-sight between sensor and target point, decomposed into X, Y, Z of the sensor's coordinate system.
Theta (θ) [deg]	Azimuth angle in the sensor's coordinate system at which the target is detected.
Phi (ϕ) [deg]	Elevation angle in the sensor's coordinate system at which the target is detected.
Width [m]	The width of the detected object.
Height [m]	The height of the detected object.
Confidence [%]	A confidence value is used for indicating how sure this object has been identified.
Sensor Type [Num]	This parameter is used for indicating which sensor has detected this pedestrian. 1=Radar Sensor; 2=Camera Sensor; 3=Both of them.

5.3.2 Pedestrian Detection

Goal and Problems Description

As mentioned above, there are three different structures can be applied for the blocks that locating in the blue rectangle in Figure 5.3. The implementation of this

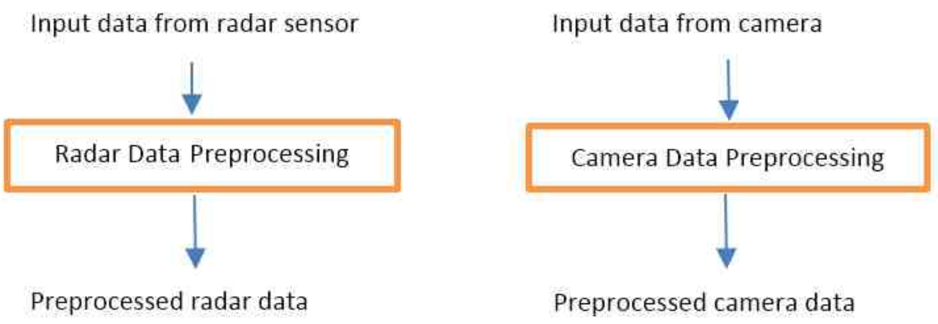


Fig. 5.7. Current implementation of Sensory Data Preprocessing Stage.

stage will also have three different versions due to the variation of structures. The following paragraphs describe these different structures in detail.

1. Sensor Fusion in Image Processing

   The previous Sensory Data Preprocessing stage provides a set of ROIs in a video frame. This stage would use classifiers to distinguish pedestrians from non-pedestrian objects for each ROI. Usually the input to the classifier is a vector of raw pixel values of the ROIs or features extracted from them, and the output is the decision showing whether there is a pedestrian or not. In many cases, the confidence values are also returned.

   The classifiers are usually trained using a number of positive and negative examples to determine the decision boundary between them. After training, the classifier processes unknown samples and decides the presence or absence of the object based on which side of the decision boundary the feature vector lies. Some of the classifiers that are used for pedestrian detection are the following: support vector machines (SVMs), various types of neural networks, and statistical learning classifiers such as AdaBoost. SVM finds a hyperplane decision boundary based on maximizing the minimum separation between classes which can be generalized to find non-linear boundaries by the use of kernel functions. Artificial Neural Networks or Machine Learning uses multiple layers of neuro to obtain highly non-linear decision boundaries between classes based on the training samples given to this classifier. Adaboost combines a number of weak classifiers into strong classifier using weighted averaging. Weights are iteratively learned based on mis-classified samples. Classifier cascade optimizes performance and speed by combining multiple classifiers by feeding output of fast but less discriminative classifier to the input of slow but more discriminative classifier.

After the classification and verification, all the non-pedestrian objects will be discarded. The output of this stage should be a list of pedestrians with their information. One part of information is from the radar sensor such as their motion and location information. The other part of information such as the size and cloth color can be obtained by applying image processing algorithms. In last stage each ROI has been paired with a target that detected by radar sensor, so these two parts of information can be easily matched and combined together.

2. Sensor Fusion after Image Processing

   The difference from the first structure is that the image processing based pedestrian detection is not assisted by the radar sensor. The image processing will search the whole picture to find pedestrians. Sometimes the location and speed of the detected pedestrians can also be obtained by image processing.

   Once the image processing based pedestrian is finished, then the sensor fusion algorithms can be applied. The same pedestrian can be detected by both the camera sensor and radar sensor, and being described differently. We just need to fuse the radar data and camera data to obtain the robust verified data of detected objects.

3. Sensor Fusion with Processed Image Data Input

   The previous Sensory Data Preprocessing stage provides the objects that detected by radar sensor and camera sensor. There are many types of objects that can be detected by the radar and camera, and each of them provides a list of detected objects. And this list contains not only pedestrians, but also other types of objects. Currently, the V2V-PAEB system is designed to be only focus on pedestrians. So all the non-pedestrian objects should be eliminated. This stage would use simple filters to distinguish pedestrians from non-pedestrian objects. We fuse the lists of objects from the radar sensor and camera together to obtain a robustly identified list of pedestrians.

Table 5.10 presents the input parameters of the Pedestrian Detection Stage. Except for the Vehicle State, all the other four input parameters are from the previous Sensory Data Preprocessing Stage.

Table 5.10. The Input Parameters of Pedestrian Detection Stage.

Item	**Description**
Preprocessed Sensory Data	The preprocessed sensory data from previous stage of Sensory Data Preprocessing.
Vehicle State	The vehicle state information that described in Table 5.1.

Table 5.11 shows the output parameters of the Pedestrian Detection Stage. This stage has only one output parameter, Detected Pedestrians. The Detected Pedestrian is an array sized Max Num. of Radar Detected Objects * Number of Parameters per Pedestrian.

Current Implementation

Currently, we have implemented the structure of Sensor Fusion with Processed Image Data Input, so the goal of this stage is using the radar-camera fusion algorithms to obtain the robust verified data of detected objects. This is due to the fact that the data is derived from two separate sensors and fused, matched and then approved. In this thesis, the V2V-PAEB simulation model uses both the two types

Table 5.11. The Output Parameters of Pedestrian Detection Stage.

Item	Description
Ped_ID [Num]	The pedestrian ID that uniquely deciding a specific pedestrian.
Ped_X [m]	The X coordinate of this pedestrian in the global coordinate system of this simulation.
Ped_Y [m]	The Y coordinate of this pedestrian in the global coordinate system of this simulation.
Ped_Z [m]	The Z coordinate of this pedestrian in the global coordinate system of this simulation.
Ped_Lat [deg/min/sec]	The GPS latitude location of this pedestrian.
Ped_Long [deg/min/sec]	The GPS longitude location of this pedestrian.
Ped_Alt [m]	The GPS altitude location of this pedestrian.
Ped_Heading [deg]	The heading direction of this pedestrian comparing to the North.
Ped_Speed [m/s]	The moving velocity of this pedestrian.
Ped_Width [m]	The width of this pedestrian.
Ped_Height [m]	The height of this pedestrian.
Ped_Confidence [%]	this value indicates how confident this pedestrian is determined.
Ped_Range [m]	The range between this pedestrian and the host vehicle.
Ped_RangeX [m]	The X directional range between this pedestrian and the host vehicle.
Ped_RangeY [m]	The Y directional range between this pedestrian and the host vehicle.
Ped_RangeZ [m]	The Z directional range between this pedestrian and the host vehicle.
Ped_DopplerVelocity [m/s]	The Doppler velocity of this pedestrian.
Ped_DopplerVelocityX [m/s]	The X directional Doppler velocity of this pedestrian.
Ped_DopplerVelocityY [m/s]	The Y directional Doppler velocity of this pedestrian.
Ped_DopplerVelocityZ [m/s]	The Z directional Doppler velocity of this pedestrian.
Ped_Theta [deg]	The Theta angle between the pedestrian and the host vehicle.
Ped_Phi [deg]	The Phi angle between the pedestrian and the host vehicle.

of sensors (one millimeter wave radar sensor and one object camera) as a combination to detect pedestrians. Both the two sensors are mounted at the front of vehicle and forward looking (They are mounted at the same location, so they have the same sensor coordinate system).

As has been mentioned, both the radar sensor and the camera sensor are providing their processed data, not just the raw data, so we dont need to dig the input parameters for more information. We just need to fuse the radar data and camera data to obtain the robust verified data of detected objects.

Currently, a very simple algorithm is applied for the fusion of radar data and camera data. This algorithm fuse the sensory data based on the location, object type and moving speed of detected objects. In this thesis, the radar sensor and camera sensor are equipped on the same location and they have the same coordinate system. So the fusion algorithms can be simplified significantly. Since radar and camera use different technologies to detect objects, and there are always some error of the output information for both of them, the same object can be described differently. Currently, some threshold values are used to tolerate and calibrate such errors. One threshold is THRESHOLD_POSITION with the value 0.2 m, and another threshold is THRESHOLD_SPEED with the value 0.2 m/s. If the difference of two locations is smaller than THRESHOLD_POSITION, they will be considered as the same location. Similarly, if the difference of two speeds is no larger than THRESHOLD_SPEED, these two speeds will be considered as the same one.

Figure 5.8 is the flow diagram that shows how this stage has been implemented. All the objects in the Radar Data will be compared with all the objects that in the Camera Data to find out the objects that detected by both radar and camera. If two objects found respectively in the radar data and the camera data have the same location, object type and moving speed, then they will be considered as the same object and it will be considered to be verified with 100 percent of confidence. Otherwise, if an object can be found only in the Radar Data or the Camera Data, then it will be considered to be identified with 50 percent of confidence.

If an object is identified with 100 percent of confidence, then the output information of this object should be fused by both the information from Radar and Camera Data. Currently, the average value is used as the output information. For example, if the speed of one object are respectively 5.8 m/s in the Radar Data and 5.9 m/s in the Camera Data, then the output speed will be (5.8 + 5.9)/2 = 5.85 m/s. If an object is identified with 50 percent of confidence, then the information of this object will not be fused and will be used directly.

Since the input data from the radar sensor and camera contains the type of each detected object, so the pedestrians can be easily identified by examining the Object Type ID. If the Object Type ID of an object is 4, then this object is a pedestrian. Other this object is a non-pedestrian object.

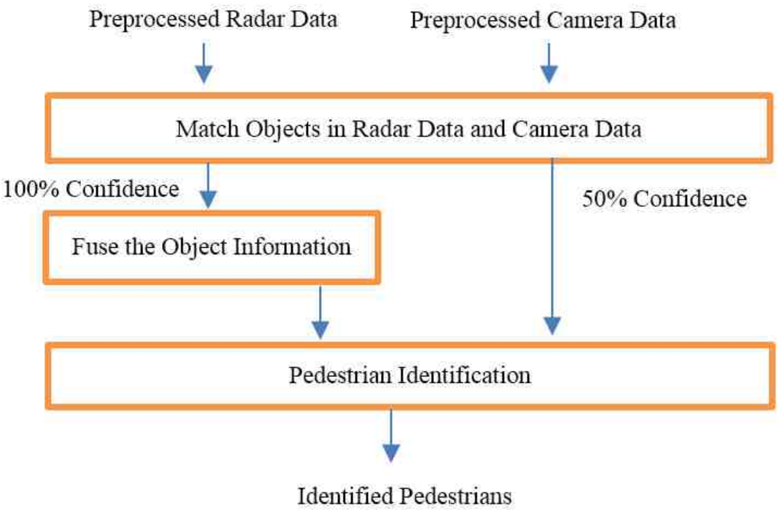


Fig. 5.8. Current implementation of Pedestrian Detection Stage.

5.3.3 Tracking (1)

Goal and Problems Description

The sensors can only provide the state information of the pedestrians, but they do not provide their trajectories. So this stage should track all pedestrians overtime to get their trajectories for the Potential Collision Prediction stage to detect potential collisions. We can describe the trajectory of detected in both global coordinate systems (such as GPS Coordinate System and Experiment Axis Coordinate System) and local Vehicle Coordinate System.

If we need to include the trajectory information of detected pedestrians in the V2V-PAEB Message and send the message to other vehicles, we should describe the trajectory in global coordinate system. The location of pedestrian is tracked using its GPS location in the GPS Coordinate System or its coordinate in the Experiment Axis Coordinate System. We can periodically record pedestrians location with a proper sample rate. Additionally, we do not need to track a pedestrian all the time since it is detected at the first time, because the V2V-PAEB system usually cares about the last several seconds. For example, we can only generate the trajectory of the pedestrian only for the latest three seconds, and the older trajectories will not be considered. Usually the trajectory can be described using a polynomial equation of GPS coordinates or axis coordinates. If a vehicle receives a V2V-PAEB Message, it will extract the trajectory of each pedestrian, and then put the vehicle itself in the global coordinate system and check if there is a potential collision between them.

Table 5.12. The Input Parameters of Tracking (1) Stage.

Item	**Description**
Detected Pedestrians	The pedestrian information that provided by the previous Pedestrian Detection Stage.
Vehicle State Data	The current state information of host vehicle provided by Vehicle State Model.

While for the on board PAEB system, it can also use the Vehicle Coordinate System to describe pedestrians trajectories in order to predict the collision between the vehicle and the pedestrian conveniently. The radar sensor can provide the range and Doppler velocity of detected pedestrians in the Radar Sensor Coordinate System (can be easily converted to Vehicle Coordinate System). The range and Doppler velocity can be tracked over time and calculate the relative distance and relative speed between the vehicle and the pedestrian. Then the relative distance and relative speed can be used to predict collision and calculate Time to Collision between them.

Table 5.12 describes the input parameters of Tracking (1) Stage. The input parameters of this stage are the output parameters of Pedestrian Detection Stage as well as the Vehicle State information.

Table 5.13 describes the output parameters of Tracking (1) Stage. As can be seen, comparing with the input parameters, just four items are appended to the end to describe the trajectory. In the global coordinate system, the trajectory can be represented as a second order polynomial equation of GPS coordinates. In this study, we use Ped_Trajectory_Coef_A, Ped_Trajectory_Coef_B and Ped_Trajectory_Coef_C as the coefficients of the equation. Ped_Trajectory_Coef_A is the coefficient of the second order item; Ped_Trajectory_Coef_B is the coefficient of the first order item; and Ped_Trajectory_Coef_C is the constant item. In the vehicle local coordinate system, we use Relative Distance and Relative Speed to abstract the trajectory of pedestrian relative to the vehicle.

Current Implementation

In this study, the trajectories of the detected pedestrians are updated every half second if they are represented using the global coordinate system. Currently, the output frequency of the Vehicle State Model is 25 Hz. That means we have 25 sample points per second and we can use these points to calculate the location of each pedestrian. Then we use a Matlab function polyfit to get the equation that presents the trajectory of pedestrian. This function has three parameters. The first two parameters are vectors containing the GPS latitude and GPS longitude respectively (or X coordinate and Y coordinate). The third parameter is the degree of the polynomial, and in this study we use degree 2. The return value of poyfit function is a vector

Table 5.13. The Output Parameters of Tracking (1) Stage.

Item	**Description**
Ped_X [m]	The X coordinate of this pedestrian in the global coordinate system of this simulation.
Ped_Y [m]	The Y coordinate of this pedestrian in the global coordinate system of this simulation.
Ped_Z [m]	The Z coordinate of this pedestrian in the global coordinate system of this simulation.
Ped_Lat [deg/min/sec]	The GPS latitude location of this pedestrian.
Ped_Long [deg/min/sec]	The GPS longitude location of this pedestrian.
Ped_Alt [m]	The GPS altitude location of this pedestrian.
Ped_Heading [deg]	The heading direction of this pedestrian comparing to the North.
Ped_Speed [m/s]	The moving velocity of this pedestrian.
Ped_Range [m]	The range between this pedestrian and the host vehicle.
Ped_RangeX [m]	The X directional range between this pedestrian and the host vehicle.
Ped_RangeY [m]	The Y directional range between this pedestrian and the host vehicle.
Ped_RangeZ [m]	The Z directional range between this pedestrian and the host vehicle.
Ped_DopplerVelocity [m/s]	The Doppler velocity of this pedestrian.
Ped_DopplerVelocityX [m/s]	The X directional Doppler velocity of this pedestrian.
Ped_DopplerVelocityY [m/s]	The Y directional Doppler velocity of this pedestrian.
Ped_DopplerVelocityZ [m/s]	The Z directional Doppler velocity of this pedestrian.
Ped_Theta [deg]	The Theta angle between the pedestrian and the host vehicle.
Ped_Phi [deg]	The Phi angle between the pedestrian and the host vehicle.
Ped_Trajectory_Coef_A	The coefficient value of the second order item in the polynomial equation of pedestrian trajectory.
Ped_Trajectory_Coef_B	The coefficient value of the first order item in the polynomial equation of pedestrian trajectory.
Ped_Trajectory_Coef_C	The coefficient value of constant item in the polynomial equation of pedestrian trajectory.

with three elements. They are respectively the coefficients of second order, first order and constant value items. These coefficients describe the track of the pedestrians and they are described in Table 5.13 in detail.

The PAEB system in our V2V-PAEB does not use global coordinate system to represent the trajectory of pedestrians. PreScan provides some libraries for calculating the Relative Distance and Relative Speed between the vehicle and pedestrian based on the vehicle state information and data from radar sensor. In this thesis, we use Relative Distance and Relative Speed to track each pedestrian and predict potential collision. Additionally, if there is a potential collision between them, the Time to Collision (TTC) can be easily calculated using Relative Distance dividing Relative Distance.

5.3.4 Send V2V-PAEB Message

Goal and Problems Description

This stage will construct the V2V-PAEB Message and let the DSRC Transmitter Model to send it to other vehicles. A V2V-PAEB message is created only if at least one pedestrian is detected. The format of V2V-PAEB message has been defined in Table 5.4. The information of pedestrians contained in this message should include their GPS location, speed, heading direction, color, size, and so on. Usually, the information of the pedestrian is calculated based on the input parameters from previous Track (1) Stage and the Vehicle State Data. For example, the GPS location of pedestrian should be calculated based on the GPS location of vehicle and their relative location between the vehicle and pedestrian. Additionally, the color and size of pedestrian are usually obtained by applying image processing algorithms to the video images from camera sensor. However, since pedestrian model in PreScan also provides the requested pedestrian information, we can directly use this when constructing the V2V-PAEB Messages. This is similar with the Pedestrian to Vehicle communication.

Table 5.14 shows the input parameters of the stage. The Detected pedestrians After Track is the output parameter of previous stage Track (1). And Vehicle State is the real time state information of the host vehicle.

Table 5.14. The Input Parameters of Send V2V-PAEB Message Stage.

Item	**Description**
Detected Pedestrians After Track	This is the output data of Stage Track (1).
Vehicle State Data	The current state information of host vehicle.

Table 5.15 is shows the output parameters of this state. The single output is a V2V-PAEB Message which has been described in Table 5.4 in detail.

Table 5.15. The Output Parameters of Send V2V-PAEB Message Stage.

Item	**Description**
V2V-PAEB Message	The formation of V2V-PAEB Message is described in Table 5.4.

In addition, the V2V-PAEB message requires a timestamp which can be used by the receivers to extrapolate and synchronize the information contained in the messages. This timestamp comes from GPS time clock which is received by all vehicles in the V2V network. All the vehicles in the V2V network should have the same time clock, otherwise, the messages will be unreliable, and there we be a mess up.

The V2V-PAEB Message contains both the host vehicles and the pedestrians GPS locations. The accuracy of these GPS locations are crucial to the V2V-PAEB system, because a vehicle would use this locations to track the pedestrians and make safety decisions after receiving a V2V-PAEB Message. However, most of vehicular GPS devices have significant positioning errors [26] and latency [27]. This is a challenging problem, but some approaches have been proposed to reduce the error of GPS devices.

This message can be either broadcast to the V2V network or directly sent to the selective vehicles that truly need this message. The former approach will cause the message explosion if there are too many vehicles and pedestrians in a small area. The computing resources would be exhausted and the network traffic will be jammed. Thats because the V2V-PAEB Message is a special type of message and it has to be generated and sent out periodically. It is different from other types of V2V message such as the lane changing alarm that described in DSRC protocol. The lane changing alarm is only generated and sent out when a vehicle is trying to change lane. So the amount of this message is not large. However, the V2V-PAEB Message is not the same story. For example, if there are 10 vehicles and 10 pedestrians in a small area. Each vehicle can see all these 10 pedestrians. And they will send V2V-PAEB messages to each other. If the V2V-PAEB generating interval is 25 ms which means 40 messages will be sent out from each vehicle per second. So each pedestrian will handle 9*40*10 = 3600 pedestrians per second which is quite significant amount of computations. Apparently, the amount of messages can be reduced if a larger interval for sending the V2V-PAEB Message. However, a larger interval means that the receiver cant obtain the real time information of the pedestrians and the receiver will be blind before the next message comes. So if broadcasting is used, the interval of V2V-PAEB Message should be properly chosen.

If a V2V-PAEB Message is directly sent to the vehicles that truly need this message. The amount of messages will be reduced but it needs the host vehicle to do some extra job to select the destinations of this message. Whats more, in the case, it needs the V2V communication protocol to support unicast and multicast. However, sometimes the sender may not be able to see the destination vehicles that indeed need this message. So unicast is usually eligible to the senders that always have a vast view field and hardly blocked by other objects. For example, the traffic lights that located at intersections and equipped with sensors and V2V components.

If there are too many pedestrians, and cant be put in one single packet, then this message can be separated into multiple packets with sequenced packet identifier. Once this message is constructed, it will be sent to nearby vehicles.

Current Implementation

PreScan provides a DSRC Message Transmitter Model that used for transmitting the V2V-PAEB Message. Currently, the V2V-PAEB Model packs the messages according to the protocol defined in this study and uses broadcast mode to send out the V2V-PAEB Messages. The broadcasting interval is 100 ms. Every 100 ms, the Send V2V-PAEB Message Stage generates a V2V-PAEB Message and then let the Message Transmitter Model send out this message.

There is a restriction of the Message Transmitter Model that it always sends a message with the fixed length of 200 items. In this thesis, if the length of message is less than 200, then all the unused items will be set to zero. As has been mentioned, the V2V-PAEB Message consist of two parts: the vehicle information and the pedestrian information. The length of the vehicle information in the V2V-PAEB Message is 10, and each pedestrian has 10 parameters. So the max number of pedestrians contained in a single message is (200-10)/10=19. If the host vehicle detects more than 19 pedestrians, we can put them into multiple messages and send them out through different messages separately.

Since the Vehicle State Model provides the real-time state information of the vehicle, so we can use this information directly to construct the vehicle information part of the V2V-PAEB Message. In this study, there are two ways to obtain the pedestrian information. One way is using the vehicle state information and sensory data to calculate the pedestrian information, such as its GPS location, speed and heading direction. The PreScan software provides the libraries for calculating these parameters, so these parameters are easily obtained. Another way is getting the pedestrian information directly from the pedestrian model. PreScan provides the simulation model of each pedestrian in the experiment and this model can generate the real-time state information of this pedestrian. If we use this method, we can add some noise or shift to the parameters obtained from pedestrian model to mimic the GPS errors or sensor errors. In this study, both of the above methods have been implemented.

5.3.5 V2V-PAEB Message Preprocessing

Goal and Problems Description

The goal of this stage is preprocessing the incoming V2V-Messages and prepare for future use. Usually, the preprocessing operations include message extraction, message filtering and information synchronization.

This stage has two input parameters (See Table 5.16). The V2V-PAEB Message that received from other vehicles, and the Vehicle State information. The detail information of them is described respectively in Table 5.4 and Table 5.1.

Table 5.16. The Input Parameters of V2V-PAEB Message Preprocessing Stage.

Item	**Description**
V2V-PAEB Message	This is the V2V-PAEB Message that described in Table 5.4.
Vehicle State Data	The current state information of host vehicle from Vehicle State Model. The detail descriptions are shown in Table 5.1.

Table 5.17 demonstrates the output parameters of this stage. Each item in this table is an array sized Max Num. of Radar Detected Objects * Max Num. of Vehicles Supported. For example, if the radar sensor can detect at most 2 objects and there are at most 2 vehicles that can send V2V-PAEB Message to the host vehicle, then the array size would be 2*2 = 4. It means that this stage can handle at most 4

pedestrians at a time. The first two elements are the pedestrian information from vehicle, and the last two elements of this array are the pedestrian information from vehicle two, and so on and so forth.

Table 5.17. The Output Parameters of V2V-PAEB Message Preprocessing Stage.

Item	**Description**
Pedestrians Detection Flag [Num]	The elements of this array are basically used as a flag used to indicate if other parameters have a significant value at the same location. For example, if Pedestrians Detection Flag (i) = 1, then Pedestrians Speed (i) will also has a meaningful value, which is the moving speed of a specific pedestrian. Additionally, if sums all the elements of Pedestrian Detection Flag up, then the total number of the observed pedestrians can be obtained.
Pedestrians Confidence [Num]	This value describes how confident the pedestrian is identified.
Pedestrians Size [Num]	The size of the pedestrian.
Pedestrians Color [Num]	The color of the pedestrian.
Pedestrians Speed [m/s]	The moving speed of the pedestrian.
Pedestrians Heading [deg]	The heading direction of the pedestrian. It is the angle comparing to the North in the clockwise direction.
Pedestrians Acc [m/s^2]	The acceleration of the pedestrian.
Pedestrians Range [m]	The distance from the host vehicle to the pedestrian.
Pedestrians Theta [deg]	The angle between the forward direction of host vehicle and the pedestrian.
Pedestrians dRotZ [deg]	The differential value of the rotate angle in the Z direction between the host vehicle and the pedestrian.
Pedestrians dx [m]	The differential distance in the X direction between the host vehicle and the pedestrian.
Pedestrians dy [m]	The differential distance in the Y direction between the host vehicle and the pedestrian.
Pedestrians dz [m]	The differential distance in the Z direction between the host vehicle and the pedestrian.

As has been discussed before, if the V2V-PAEB model utilizes broadcasting to send V2V-PAEB Messages, there will be a message explosion if there are too many vehicles and pedestrians in a small area. A receiver can receive many V2V-PAEB Messages at a very short period of time. However, it might be that many of them are useless to the receiver. In this sense, the receiver should eliminate such useless messages as soon as possible. Otherwise, it will waste the computing resources of the receiver. Usually, the useless messages can be eliminated by applying proper filters. All the received V2V-PAEB Message should be firstly fed to the filters. Table 5.18 shows the filtering methods that can be used in the V2V-PAEB simulation model.

If a received message passes all the above filters, then it should be put into a message queue. This stage would periodically read a number of messages from the queue and extract the pedestrian information in them. The size of the queue should be large enough to hold the max number of messages that the simulation experiment can generate. So the queue size should be Max Num. of Vehicles Supported * Length per Message.

Since there is always an uncertain time delay for messages transmitting on the V2V network, the current locations of all the pedestrians contained in the messages might have been changed a lot by the time the messages arrive at the host vehicle. So current locations of these pedestrians need to be predicted and calibrated using the timestamp information contained in the messages. In addition, these messages were usually generated and arrived at different time, and they need to be synchronized before using them. Since the pedestrian location information in V2V-PAEB Messages are described using the world coordinate system (GPS location), they need to be converted to the host vehicles local coordinate system.

Additionally, the V2V network is an open network and it can be hacked. Some vehicles in the network may send messages with inaccurate or even false information of pedestrians. They may also send out messages at an extremely high frequency and try to jam the traffic V2V network. This stage should provide some mechanism to

Table 5.18. The Message Filters of V2V-PAEB Message Preprocessing Stage.

Filter	**Description**
Message Type Filter	The V2V Message Receiver Model can receive many types of V2V messages at the same time. Then all of these messages will be fed to the V2V-PAEB Message Preprocessing Stage. However, the V2V-PAEB system currently needs only V2V-PAEB Messages. So all the non-V2V-PAEB Messages should be discarded.
Sender ID Filter	The V2V network usually has a blacklist which contains the vehicles that have done some malicious behaviors. All the messages that from the senders that contained in the blacklist should be denied.
Location Filter	This filter would check the location of the sender and the pedestrians contained in the message. If they are far away from the host vehicle, or even they are on the different street, this message should be discarded because they do not post any potential collision with the host vehicle.
Event Time Filter	In the V2V network, there is always some uncertain time of delay when a message traveling from the sender to the receiver. This filter would examine the event time of a message, and compare it with a threshold. If the event time of this message is earlier than this threshold, it will be considered to be expired and then eliminated.

detect such misbehavior [24]. Otherwise, serious safety problems can be caused by such type of misbehavior.

Current Implementation

Figure 5.9 shows the current implementation of the V2V-PAEB Message Processing Stage in V2V-PAEB simulation model. As mentioned above, there are many problems in the V2V-PAEB Message Processing Stage. However, in this thesis, we only focus on four basic and typical problems for providing a fundamental structure of this this block. If needed, we can easily update the algorithms of this block in the future. The implementation of these four steps are listed as following.

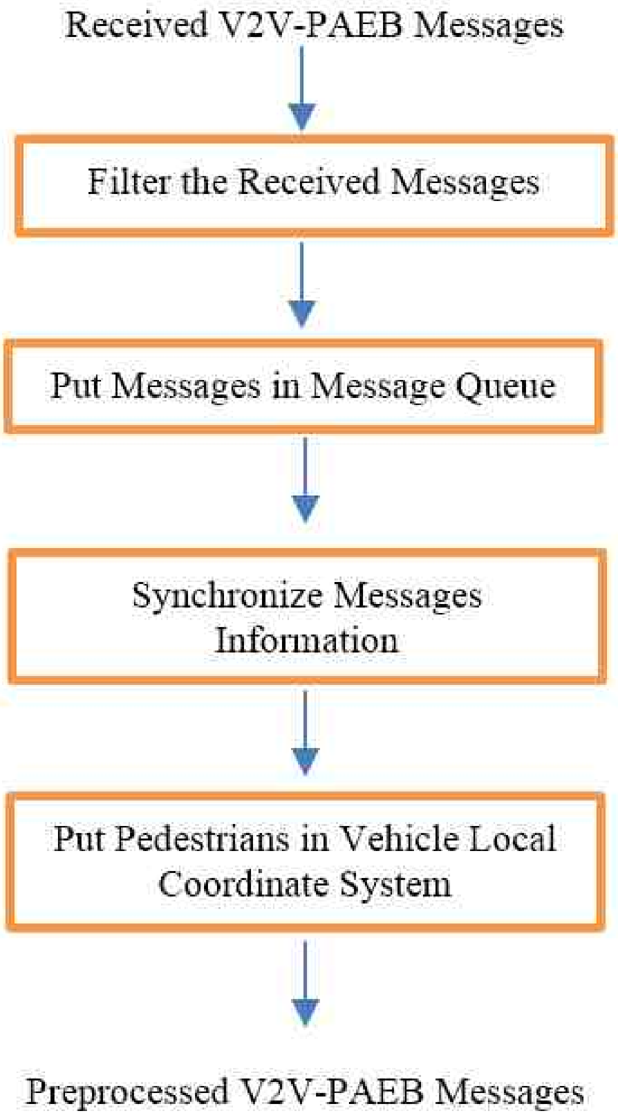

Received V2V-PAEB Messages
Filter the Received Messages
Put Messages in Message Queue
Synchronize Messages Information
Put Pedestrians in Vehicle Local Coordinate System
Preprocessed V2V-PAEB Messages

Fig. 5.9. Current implementation of the V2V-PAEB Message Processing Stage.

The step Filter the Received Messages will filter the incoming messages. All the filters mentioned in Table 5.18 have been implemented in V2V-PAEB simulation model. For each received message, the Sender ID Filter will check if the sender is in the blacklist. If it is in the blacklist, then this message will be discarded. If the message is accepted by the Sender ID Filter, then it will go to the Message Type Filter. In this filter, the Message Type will be examined and only the V2V-PAEB Message can pass this filter. If this message is accepted by the Message Type Filter, then it will be fed to the Event Time Filter. This filter will compare the Event Time of this message with current time. We use a threshold value to determine if this message is too old and should be discarded. Currently, this threshold value is set to be 1.5 s. That means if this message was generated more than 1.5 s ago, it will be discarded. Sender Location & Pedestrian Location Filter is the last filter that will filter the message based on the locations of both the sender and the pedestrians contained in the message. Currently, if the distance between the sender and the host vehicle is more than 100 meters, then this message will be discarded. In addition, if their distance is within 100 meters, then this filter will continue to examine the distance between the host vehicle and each pedestrian contained in this message. If the distance between the host vehicle and any pedestrian is bigger than 50 meters, then this pedestrian will be discarded. If one message has passed all the above filters successfully, then this message and the pedestrians contained in this message will be accepted finally.

The step Put Messages in Message Queue will temporarily store the received V2V-PAEB Messages. Currently, the size of the queue is Max Num. Vehicles Supported. It means that this message queue can store at most Max Num. Vehicles Supported V2V-PAEB Messages. Thats enough because the V2V-PAEB Messages are being sent out by the sender and processed by the receiver at the same frequency. Once a message has been processed by the receiver, this message will be deleted from the message queue.

The step Synchronize Message Information will predict the current information for each pedestrian contained in this V2V-PAEB Message. Message information synchronization is especially necessary if V2V-PAEB Message Frequency is low, or the message transmission delay is big. In current implementation, the V2V-PAEB Message Frequency is 100, which means every 10 ms, a V2V-PAEB Message will be generated and sent out. In addition, currently the time delay for message transmission is set to zero which means no time delay. So we can assume that within such a short period of time, the information of each pedestrian does not change. So we do not apply message information synchronization here.

The step Put Pedestrians in Vehicle Local Coordinate System will project each pedestrian into the local coordinate system of the host vehicle. Since the location information for each pedestrian is a GPS coordinate, the host vehicle should convert it to the local coordinate system. PreScan provides the tools GPS2XYZ for converting GPS coordinate to local coordinate.

5.3.6 V2V-PAEB Message Merge

Goal and Problems Description

The previous stage provides N sets of pedestrians that obtained from the received messages. It is possible that a pedestrian is detected by different vehicles at the same time, so there are usually many duplicated pedestrians here. In addition, due to the inaccuracy of different sensors, the information of the same pedestrian can be reported differently from messages that from different vehicles and even from different messages of the same vehicle. It is also possible that different pedestrians are mapped to the same location by different vehicles. In some extreme conditions, the host vehicle may receive messages that containing many false pedestrians. This stage serves for merging all these message data together to obtain a list of pedestrians without duplicate ones. This stage will also calibrate the information of each pedestrian should as accurate as possible.

Table 5.19 describes the input parameters of this stage. The Preprocessed V2V-PAEB Message is the output data of the V2V-PAEB Message Preprocessing Stage. This is a multiple dimensional vector containing many preprocessed V2V-PAEB Mes-

Table 5.19. The Input Parameters of V2V-PAEB Message Merge Stage.

Item	**Description**
Preprocessed V2V-PAEB Messages	The preprocessed V2V-PAEB Messages that described in detail in Table 5.17.
Vehicle State Data	The current state information that described in detail in Table 5.1.

sages. The Vehicle State Data is the real time state information from the Vehicle State Model.

Table 5.20 presents the output parameters of this stage. This is a multiple dimensional vector containing the information of many pedestrians. The V2V-PAEB Message Merge Stage has merged and calibrated the information of pedestrians extracted from the received messages, and all the duplicated pedestrians have been eliminated.

Current Implementation

Figure 5.10 shows the current implementation of V2V-PAEB Message Merge Stage. Currently, a very simple algorithm is applied for the merge of V2V-PAEB Messages. This algorithm merge the V2V-PAEB Messages based on the location and moving speed of pedestrians contained in the messages.

Since there are always some error of the output information for both of the radar sensor and camera sensor, the same pedestrian can be described differently by deficient vehicles. Currently, some threshold values are used to tolerate and calibrate such errors. One threshold is THRESHOLD_POSITION with the value 0.5 m, and another threshold is THRESHOLD_SPEED with the value 0.3 m/s. If the difference of two locations is smaller than THRESHOLD_POSITION, they will be considered as the same location. Similarly, if the difference of two speeds is no larger than THRESHOLD_SPEED, these two speeds will be considered as the same one. In ad-

Table 5.20. The Output Parameters of V2V-PAEB Message Merge Stage.

Item	**Description**
Pedestrians Detection Flag [Num]	The elements of this array are basically used as a flag used to indicate if other parameters have a significant value at the same location. For example, if Pedestrians Detection Flag (i) = 1, then Pedestrians Speed (i) will also has a meaningful value, which is the moving speed of a specific pedestrian. Additionally, if sums all the elements of Pedestrian Detection Flag up, then the total number of the observed pedestrians can be obtained.
Pedestrians Confidence [Num]	This value describes how confident the pedestrian is identified.
Pedestrians Size [Num]	The size of the pedestrian.
Pedestrians Color [Num]	The color of the pedestrian.
Pedestrians Speed [m/s]	The moving speed of the pedestrian.
Pedestrians Heading [deg]	The heading direction of the pedestrian. It is the angle comparing to the North in the clockwise direction.
Pedestrians Acc [m/s^2]	The acceleration of the pedestrian.
Pedestrians Range [m]	The distance from the host vehicle to the pedestrian.
Pedestrians Theta [deg]	The angle between the forward direction of host vehicle and the pedestrian.
Pedestrians dRotZ [deg]	The differential value of the rotate angle in the Z direction between the host vehicle and the pedestrian.
Pedestrians dx [m]	The differential distance in the X direction between the host vehicle and the pedestrian.
Pedestrians dy [m]	The differential distance in the Y direction between the host vehicle and the pedestrian.
Pedestrians dz [m]	The differential distance in the Z direction between the host vehicle and the pedestrian.

dition, a simple logic was used to determine if two pedestrians from different messages are the same pedestrian. Currently, if two pedestrians have the same location and speed, they will be considered as the same pedestrian.

As mentioned above, a pedestrian can be described differently in different V2V-PAEB Messages, so the information of this pedestrian should be calibrated after it is matched and verified. In this thesis, we calculate the average value for each parameter of the pedestrian and use this average value to describe this pedestrian. For example, a pedestrian is contained in 5 V2V-PAEB Messages, and its speed in these messages are described respectively as 1.45 m/s, 1.50 m/s, 1.58 m/s, 1.45 m/s and 1.60 m/s. So the speed of this pedestrian will be determined as (1.45+1.50+1.58+1.45+1.60) = 1.516 m/s. Other parameters such as the location, heading direction, Range and Theta are also calculated in the same way.

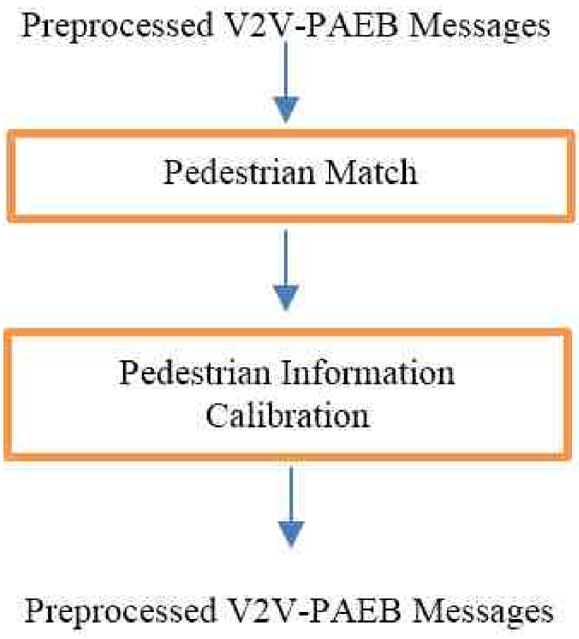

Preprocessed V2V-PAEB Messages
Pedestrian Match
Pedestrian Information Calibration
Preprocessed V2V-PAEB Messages

Fig. 5.10. Current implementation of V2V-PAEB Message Merge Stage.

5.3.7 Tracking (2)

Goal and Problems Description

Similar to the purpose of Track (1) stage, all the pedestrians detected from the received V2V-PAEB Messages should also be tracked over time to get their trajectories in order to predict their future locations and detect potential collisions.

Since the locations of pedestrians extracted from the V2V-PAEB Messages are presented using GPS coordinates, so the trajectories of pedestrians should be also presented using Global Coordinate System (GPS Coordinate System or Experiment Axis Coordinate System). Usually, the trajectory of one pedestrian can be obtained if at least two V2V-PAEB Messages that contain this pedestrian arrive at the receiver. The more V2V-PAEB Messages contain this pedestrian received, the more accurate the trajectory will be.

Table 5.21 shows the input parameters of this stage. The Merged V2V-PAEB Messages is the output parameter of V2V-PAEB Message Preprocessing Stage. The duplicate pedestrian information containing in the Merged V2V-PAEB Messages have been eliminated and calibrated. The Vehicle State Data is the output data of Vehicle State Model. It provides the real time state information of the vehicle.

Table 5.21. The Input Parameters of Tracking (2) Stage.

Item	**Description**
Merged V2V-PAEB Messages	The output parameter of V2V-PAEB Message Preprocessing Stage.
Vehicle State Data	The current state information that described in detail in Table 5.1.

Table 5.22 shows the output parameters of this stage. The trajectory of pedestrian is also presented as a second order polynomial equation. In this study, we use Ped_Trajectory_Coef_A, Ped_Trajectory_Coef_B and Ped_Trajectory_Coef_C as

the coefficients of the equation. Ped_Trajectory_Coef_A is the coefficient of the second order item; Ped_Trajectory_Coef_B is the coefficient of the first order item; and Ped_Trajectory_Coef_C is the constant item.

Current Implementation

In this study, once a V2V-PAEB Message containing one specific pedestrian arrives, the trajectory of pedestrian will be updated. Currently, the V2V-PAEB Message is generated and sent out every 100 ms. That means we usually have 10 (the number can vary due to transmission delay and packet loss) sample points per second and we can use these points to calculate the location of each pedestrian. Then we use a Matlab function polyfit to get the equation that presents the trajectory of pedestrian. This function has three parameters. The first two parameters are vectors containing the GPS latitude and GPS longitude respectively. The third parameter is the degree of the polynomial, and in this study we use degree 2. The return value of polyfit function is a vector with three elements. They are respectively the coefficients of second order, first order and constant value items.

5.3.8 Pedestrian Information Merge

Goal and Problems Description

Now there are two sets of detected pedestrians: one is detected from the on-board sensor systems of host vehicle; and the other one is detected from the received V2V-PAEB Messages. This stage is responsible for merging them together to obtain a complete set of pedestrians surrounding the host vehicle.

It is possible that some of the pedestrians that reported by other vehicles can also be detected by host vehicle's on-board sensor systems. So in this stage, all the duplicated pedestrians should be eliminated. It is also possible that the on-board sensors of host vehicle and the V2V-PAEB Messages describe the same pedestrian

Table 5.22. The Output Parameters of Tracking (2) Stage.

Item	**Description**
Pedestrians Detection Flag [Num]	The elements of this array are basically used as a flag used to indicate if other parameters have a significant value at the same location. For example, if Pedestrians Detection Flag (i) = 1, then Pedestrians Speed (i) will also has a meaningful value, which is the moving speed of a specific pedestrian. Additionally, if sums all the elements of Pedestrian Detection Flag up, then the total number of the observed pedestrians can be obtained.
Pedestrians Confidence [Num]	This value describes how confident the pedestrian is identified.
Pedestrians Size [Num]	The size of the pedestrian.
Pedestrians Color [Num]	The color of the pedestrian.
Pedestrians Speed [m/s]	The moving speed of the pedestrian.
Pedestrians Heading [deg]	The heading direction of the pedestrian. It is the angle comparing to the North in the clockwise direction.
Pedestrians Acc [m/s^2]	The acceleration of the pedestrian.
Pedestrians Range [m]	The distance from the host vehicle to the pedestrian.
Pedestrians Theta [deg]	The angle between the forward direction of host vehicle and the pedestrian.
Pedestrians dRotZ [deg]	The differential value of the rotate angle in the Z direction between the host vehicle and the pedestrian.
Pedestrians dx [m]	The differential distance in the X direction between the host vehicle and the pedestrian.
Pedestrians dy [m]	The differential distance in the Y direction between the host vehicle and the pedestrian.
Pedestrians dz [m]	The differential distance in the Z direction between the host vehicle and the pedestrian.
Ped_Trajectory_Coef_A	The coefficient value of the second order item in the polynomial equation of pedestrian trajectory.
Ped_Trajectory_Coef_B	The coefficient value of the first order item in the polynomial equation of pedestrian trajectory.
Ped_Trajectory_Coef_C	The coefficient value of constant item in the polynomial equation of pedestrian trajectory.

differently, so the information of duplicate pedestrians should also be calibrated and verified. Usually, the information from on-board sensors is usually more reliable than that from the received messages. So the information of pedestrians from the on-board sensor systems can be used to calibrate or verify the pedestrian information from messages.

Table 5.23 demonstrates the input parameters of this stage. One parameter is the tracked pedestrians from Track (1) Stage, and another one is the tracked pedestrians from the Track (2) Stage. The input parameters from Track (1) and Track (2) should have the similar format, so that they can be merged in this stage.

Table 5.23. The Input Parameters of Pedestrian Information Merge Stage.

Item	**Description**
Pedestrians From Track (1) Stage	This input parameter is from the output of the Track (1) Stage. The detail information of this parameter is described in Table 5.13.
Pedestrians From Track (2) Stage	This input parameter is from the output of the Track (2) Stage. The detail information of this parameter is described in Table 5.22.
Vehicle State Data	The current state information that described in detail in Table 5.1.

Table 5.24 shows the output parameters of the Pedestrian Information Merge Stage. The items of the output are basically the same as the output parameters of Tracking (1) and Tracking (2) only with some minor differences. One difference is the number of pedestrians containing in this output parameter. That is because the duplicated pedestrians have been eliminated. Another difference is the value for each parameter containing in this output data. That is because the value of each parameter might have been calibrated based on the input from Tracking (1) and Tracking (2) stages.

Table 5.24. The Output Parameters of Pedestrian Information Merge Stage.

Item	**Description**
Pedestrians Detection Flag [Num]	The elements of this array are basically used as a flag used to indicate if other parameters have a significant value at the same location. For example, if Pedestrians Detection Flag (i) = 1, then Pedestrians Speed (i) will also has a meaningful value, which is the moving speed of a specific pedestrian. Additionally, if sums all the elements of Pedestrian Detection Flag up, then the total number of the observed pedestrians can be obtained.
Pedestrians Confidence [Num]	This value describes how confident the pedestrian is identified.
Pedestrians Size [Num]	The size of the pedestrian.
Pedestrians Color [Num]	The color of the pedestrian.
Pedestrians Speed [m/s]	The moving speed of the pedestrian.
Pedestrians Heading [deg]	The heading direction of the pedestrian. It is the angle comparing to the North in the clockwise direction.
Pedestrians Acc [m/s^2]	The acceleration of the pedestrian.
Pedestrians Range [m]	The distance from the host vehicle to the pedestrian.
Pedestrians Theta [deg]	The angle between the forward direction of host vehicle and the pedestrian.
Pedestrians dRotZ [deg]	The differential value of the rotate angle in the Z direction between the host vehicle and the pedestrian.
Pedestrians dx [m]	The differential distance in the X direction between the host vehicle and the pedestrian.
Pedestrians dy [m]	The differential distance in the Y direction between the host vehicle and the pedestrian.
Pedestrians dz [m]	The differential distance in the Z direction between the host vehicle and the pedestrian.
Ped_Trajectory_Coef_A	The coefficient value of the second order item in the polynomial equation of pedestrian trajectory.
Ped_Trajectory_Coef_B	The coefficient value of the first order item in the polynomial equation of pedestrian trajectory.
Ped_Trajectory_Coef_C	The coefficient value of constant item in the polynomial equation of pedestrian trajectory.

Current Implementation

Figure 5.11 shows the current implementation of Pedestrian Information Merge Stage. Currently, a very simple algorithm is applied for the merge of pedestrian information that from the on-board sensor systems and V2V-PAEB Messages. This algorithm merges the pedestrian information based on the location and moving speed of pedestrians contained in the messages. This is quite similar with the implementation of V2V-PAEB Message Merge Stage.

Since there are always some error of the output information for both of the radar sensor and camera sensor, the same pedestrian can be described differently by deficient vehicles. Currently, some threshold values are used to tolerate and calibrate such errors. One threshold is THRESHOLD_POSITION with the value 0.3 m, and another threshold is THRESHOLD_SPEED with the value 0.2 m/s. If the difference of two locations is smaller than THRESHOLD_POSITION, they will be considered as the same location. Similarly, if the difference of two speeds is no larger than THRESHOLD_SPEED, these two speeds will be considered as the same one. In addition, a simple logic was used to determine if two pedestrians from different messages are the same pedestrian. Currently, if two pedestrians have the same location and speed, they will be considered as the same pedestrian.

As mentioned above, a pedestrian can be described differently in different V2V-PAEB Messages, so the information of this pedestrian should be calibrated after it is matched and verified. In this thesis, we calculate the average value for each parameter of the pedestrian and use this average value to describe this pedestrian. For example, a pedestrian is contained in 5 V2V-PAEB Messages, and its speed in these messages are described respectively as 1.45 m/s, 1.50 m/s, 1.58 m/s, 1.45 m/s and 1.60 m/s. So the speed of this pedestrian will be determined as (1.45+1.50+1.58+1.45+1.60) = 1.516 m/s. Other parameters such as the location, heading direction, Range and Theta are also calculated in the same way.

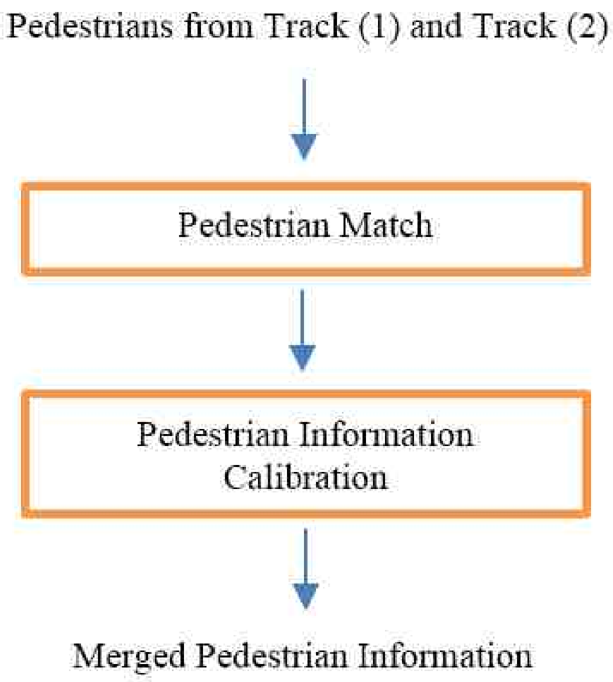

Pedestrians from Track (1) and Track (2)
Pedestrian Match
Pedestrian Information Calibration
Merged Pedestrian Information

Fig. 5.11. Current implementation of Pedestrian Information Merge Stage.

5.3.9 Potential Collision Detection

Goal and Problems Description

This stage would project the current trajectories of the pedestrians and the host vehicle into future and determine the possibility of collision based on geometric computations. There are many approaches for predicting the collision between the host vehicle and pedestrians. Usually the speed and trajectory of the pedestrians as well as the host vehicle are assumed do not change significantly during that time. Some approaches assume the vehicle and pedestrians as points when predicting the probability of collision. This way is convenient for computation but it will lose some precision. So the dimensions and sizes of both the host vehicle and the pedestrians should also be considered when predicting the potential collision.

Once a potential collision is detected, then the TTC should be calculated to show when the potential collision happen. In addition, a Collision Confidence value should be given for indicating how sure this potential collision is identified.

Table 5.25 shows the input parameters of Potential Collision Prediction Stage. In this stage, the potential collision is predicted based on the current state information of both the host vehicle and the detected pedestrians.

Table 5.25. The Input Parameters of Potential Collision Prediction Stage.

Item	**Description**
Detected pedestrians	This is from the output of Pedestrian Information Merge Stage. The detail information of this parameter is described in Table 5.20.
Vehicle State Data	The current state information that described in detail in Table 5.1.

Table 5.26 shows the output parameters of the Potential Collision Prediction Stage. Each potential collision is presented by the TTC and its confidence. The TTC is used for indicating how soon this potential collision will occur. And its confidence is used for specifying how sure the potential collision is.

Table 5.26. The Output Parameters of Potential Collision Prediction Stage.

Item	**Description**
Time to Collisions (TTCs) [s]	The time to collisions for each of the detected potential collision.
Collision Confidences [%]	The confidence for each detected potential collision.

Current Implementation

Figure 5.12 shows the current implementation of this stage. In this thesis, both the vehicle and the pedestrians are considered as points. So the calculation of the potential collision can be simplified significantly. This stage will check the detected pedestrians one by one to determine if there will be a potential collision between the vehicle and each pedestrian. The prediction of potential collision can be finished in the following two steps.

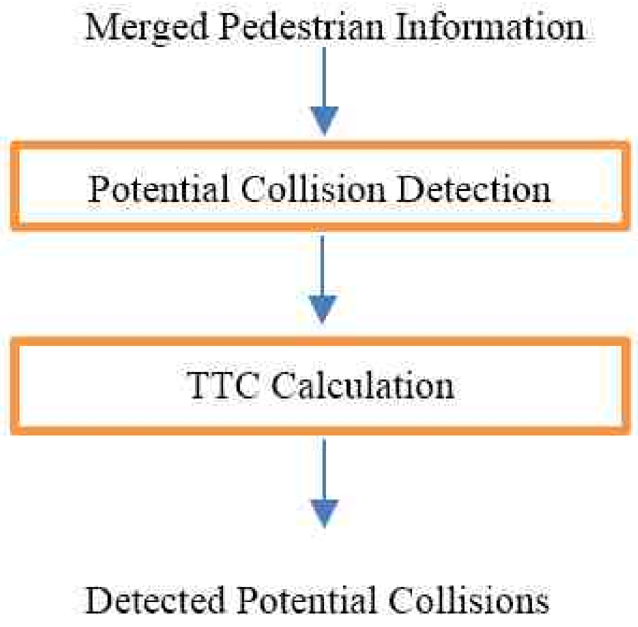

Merged Pedestrian Information
Potential Collision Detection
TTC Calculation
Detected Potential Collisions

Fig. 5.12. Current implementation of Potential Collision Prediction Stage.

The first step is examine the trajectories of both the pedestrian and host vehicle, and check if they can meet at some point in the future. If there is no cross point between their trajectories, then there will be no potential collision between them. Otherwise, there might be a potential collision. Then the second step is checking if they can reach the cross point at the same time. If they can reach this cross point at the same time, there will be a potential collision with them. Otherwise, there will be no potential collision. So after these two simple steps, a potential collision is determined. However, since the vehicle and the target pedestrian may not travel with constant speed, and the trajectories may also change suddenly, so the potential collision may not be determined with 100 percent of confidence. Currently, we assume that both the vehicle and the pedestrian travel with constant velocity and do not make sharp turns. So the potential collision is determined with 100 percent of confidence.

5.3.10 Decision Making

Goal and Problems Description

This stage is responsible for making proper safety decisions when potential collisions are detected. For each potential collision, the Time to Collision (TTC) should be used for evaluating its emergency level [25]. In the case of high level of emergency, the proper driver warning should be triggered to enable corrective actions. If the collision is imminent, the automatic braking could also be started to avoid or mitigate the potential collision.

If the automatic braking is decided, then the braking pressure should be calculated. In different conditions, the braking pressure can be different. Sometimes full braking is essential and sometimes partial braking is fine. Some approach supports automatic steering when the braking is going on. In this case, the decision strategy for automatic steering should also be implemented in this stage [5].

Table 5.27. The Input Parameters of Decision Making Stage.

Item	**Description**
Detected Potential Collisions	The detected potential collisions from the previous stage. They are described in details in Table 5.26.
Vehicle State Data	The current state information that described in detail in Table 5.1.

Table 5.28. The Output Parameters of Decision Making Stage.

Object Type ID	**Description**
Pedestrian Detection Flag [Y/N]	This flag is used to indicate whether any pedestrians have been detected. If so, this parameter will be set to Y. Otherwise, it will be set to N.
Driver Warning Flag [Y/N]	This flag is used to indicate if the driver warning should be triggered. If this parameter is set to Y, then a driver warning will be triggered immediately. Otherwise, if this parameter is set to N, then the vehicle will do nothing.
Automatic Braking Flag [Y/N]	This flag is used to indicate if the automatic braking should be started. If it is set to Y, then the automatic braking will be started immediately. Otherwise, if this parameter is set to N, then the vehicle will do nothing.
Brake Pressure [bar]	This value is used to control the deceleration of the vehicle when the automatic braking is started. Once the Automatic Braking Flag is set to Y, then the Brake Pressure should be assigned a value between zero and the Max Braking Pressure. Otherwise, if the Automatic Braking Flag is set to N, then this parameter should be zero.
Automatic Steering Flag [Y/N]	This flag is used to indicate if the automatic steering control should be started. If it is set to Y, then the automatic steering will be started immediately. Otherwise, if this parameter is set to N, then the vehicle will do nothing.
Automatic Steering Angle [deg]	This value is used to specify the steering wheel status. A positive value means turning right, and minus value means turning left. Once the Automatic Steering Flag is set to Y, then this parameter should be assigned with a meaningful value. Otherwise, it should be zero.
Time To Collision [s]	Time to Collision represents for how much time left before the collision occurs. If many pedestrians are detected at the same time, and each one has its TTC. Then this parameter will be set to the smallest of them.

Table 5.27 describes the input information of Decision Making Stage. The Detected Potential Collisions are the output of previous stage Potential Collision Prediction, and the Vehicle State Data is the real time state information of the host vehicle. The reason for transmitting the Vehicle State information to this stage is that the vehicle usually makes different decisions according to its own state.

Table 5.28 describes the output parameters of the Decision Making Stage. As can be seen in Figure 5.2, these output parameters will be transmitted to other models of vehicle for either taking actions or displaying the simulation results.

Current Implementation

Usually, a safety decision will be made based on both the detected potential collisions and current state of the host vehicle. The logic of the decision making process can be extremely complicated. However, in this thesis, only some simple logic are used to demonstrate the decision making process and the use of this stage.

Pedestrian Detection Flag is calculated based on the input parameter Detected Potential Collisions. As has been mentioned above, if a potential collision with pedestrian is detected, the Collision Confidences will be set to 100. So the Collision Confidences value can be used to determine the Pedestrian Detection Flag. If the Collision Confidences has the value 100, then set the Pedestrian Detection Flag to Y. Otherwise, it will be N.

Many potential collisions can be detected with different TTCs at the same time. Then the potential collision that has the smallest TTC will be chosen to determine the Driver Warning Flag and the Automatic Braking Flag. If current TTC is smaller than the value of THRESHOLD_WARNING, then the Driver Warning Flag will be set to Y. Otherwise, it will be N. In addition, if current TTC is smaller than the value of THRESHOLD_BRAKING, then the Automatic Braking Flag will be set to Y. Otherwise, it will be N.

Once the Automatic Braking Flag is set to Y, then the output parameter Brake Pressure should be calculated immediately. Usually, the Braking Pressure will be different under different emergency level and the vehicle state. The max value of Brake Pressure is the predefined global variable Max Braking Pressure. Currently, the V2V-PAEB does not support automatic steering, so the output parameter Automatic Steering Flag will be always set to N and Automatic Steering Angle will be always zero.

Table 5.29. Threshold values used for making decisions.

Item	**Description**
THRESHOLD_WARNING	A threshold value that used for determine the Driver Warning Flag. This value is usually calculated real-time according to the state of vehicle or the emergency level of potential collision.
THRESHOLD_BRAKING	A threshold value that used for determine the Automatic Braking Flag. This value is also usually calculated real-time according to the state of vehicle or the emergency level of potential collision.

From the year 2013 to 2014, two different 2013 model year sedans with PAEB capability were tested by IUPUI TASI group, and 400 test runs were for one vehicle and 350 for the other one. We collect two sets of data that describing the PAEB performance of both the vehicles. Finally, a decision making simulation model for each vehicle was developed based on the testing data. In paper [26], the implementation of the decision making model is presented in detail.

In this study, the V2V-PAEB simulation model currently does not support automatic steering. So the output parameter Automatic Steering Flag will be always set to N and Automatic Steering Angle will be always zero.

6. SIMULATION TEST OF V2V-PAEB MODEL

The proposed V2V-PAEB simulation model has been implemented and tested in PreScan environment. PreScan comprises several modules that together provide everything the V2V-PAEB simulation model needs. The intuitive graphical user interface (GUI) allows us to build the experiment scenario and model the requested sensors, while the Matlab/Simulink interface enables us to develop and test the V2V-PAEB simulation model. The following sections present how to develop and test the V2V-PAEB simulation model using PreScan environment.

6.1 Build Experiment Scenario

Most of the V2V-PAEB scenarios for improving pedestrian safety in [4] are in the extreme conditions that the PAEB systems usually have very poor performances. Figure 6.1 is an example scenario chosen from paper [4] for testing the V2V-PAEB simulation model. In this scenario, five vehicles and one pedestrian are at the intersection. The traffic light changes from green to red when the pedestrian is crossing street. At the same time, vehicle 5 is approaching this intersection quickly and the driver sees the traffic changes to green, so the driver does not stop and keeps driving. Both the pedestrian and vehicle 5 could not see each other, because their views are blocked by vehicle 2. However, vehicle 1 and vehicle 2 can see this pedestrian.

For this experiment scenario, two cases will be run separately to check if the V2V-PAEB system works better than the PAEB system.

Case 1: Only vehicle 5 is equipped with V2V-PAEB system. So vehicle 5 cannot receive any V2V-PAEB Messages from other vehicles, and it uses only PAEB system to detect pedestrians. Since the line of sight of on-board sensors is blocked by vehicle

2, the PAEB system may have a poor performance. In this case, we will examine if the potential collision can be avoided without the assistance of V2V-PAEB Messages.

Case 2: Vehicle 1, 2 and 5 are equipped with V2V-PAEB system, and vehicle 1 and 2 can detect this pedestrian and they will report this pedestrian to vehicle 5 through V2V-PAEB Message. Theoretically speaking, vehicle 5 can detect this pedestrian much earlier through the received V2V-PAEB Message than through its on-board sensors. The performance of the V2V-PAEB simulation model can be evaluated by examining if the collision between the pedestrian and vehicle 5 can be avoided or mitigated.

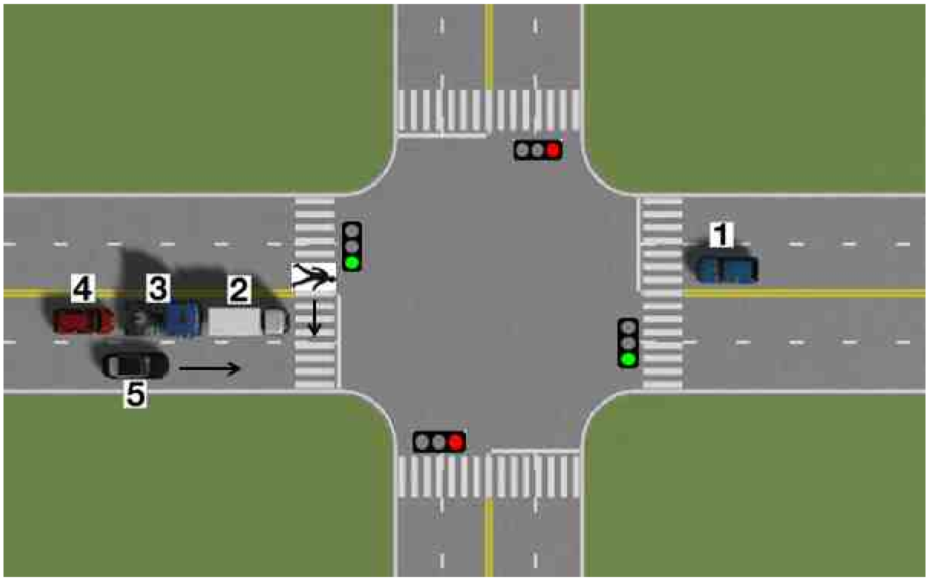


Fig. 6.1. The experiment scenario for testing the V2V-PAEB model.

This experiment can be easily built in PreScans GUI by using drag and drop actions to the library elements of road sections, infrastructure components (trees, buildings, traffic signs), actors (cars, trucks, bikes and pedestrians), sensors (radars, cameras, lidars), weather conditions (such as rain, snow and fog) and light sources (such as the sun, headlights and lampposts). Figure 6.2 shows the built experiment in the GUI. This experiment scenario is exactly the same as the one described in the above figure.

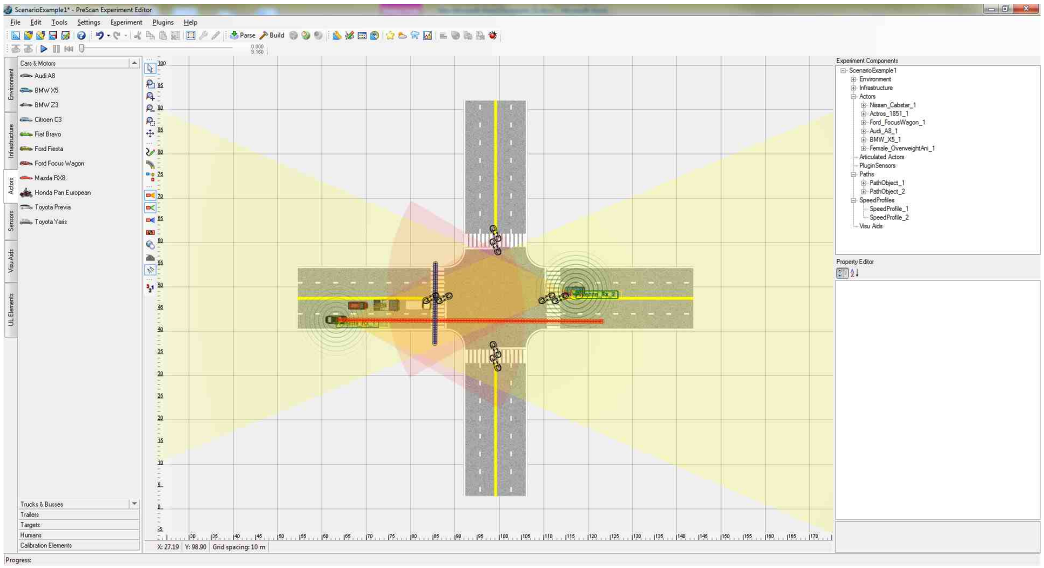


Fig. 6.2. The built experiment scenario in PreScans GUI.

As has been mentioned in section 5, the V2V-PAEB simulation model uses two sensors (one radar sensor and one camera sensor) for pedestrian detection. PreScan has provided the simulation models of both the radar sensor and camera sensor, and both of them can be configured with different performances. Table 6.1 and Table 6.2 present the configuration of Radar Sensor Model and Camera Sensor Model respectively.

Table 6.1. The configuration of Radar Sensor Model.

Parameter	**Configuration**
Scan Pattern	Line Scan
Number of Beams	1
Beam Type	Elliptical Cone
Beam Range [m]	40
Beam $\Delta\theta$ [deg]	60
Beam $\Delta\phi$ [deg]	9
Capture Frequency [Hz]	25
Max Number of Objects	10

Table 6.2. The configuration of Camera Sensor Model.

Parameter	**Configuration**
Stereo Vision	Disabled
Horizontal Resolution [pixel]	500
Vertical Resolution [pixel]	375
Frame Rate [Hz]	50
Color/Monochrome	Monochrome
Intensity Factor [RGB]	1/1/1
CCD Parameters	Enabled
Focal length	7.5
CCD Chip Size [mm]	1/2 (6.4*4.8)

After the experiment scenario is built in PreScans GUI, then the experiment is compiled into a dedicated MATLAB/Simulink Engineering Workspace (see Figure 6.3). The models of vehicles and pedestrians as well as the sensors are generated automatically and ready to use.

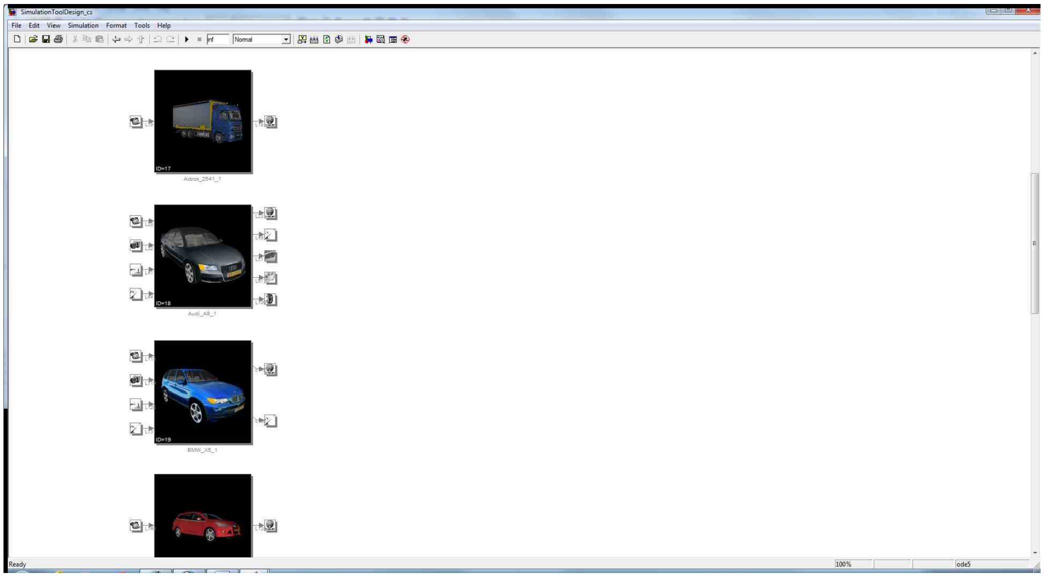


Fig. 6.3. The simulation models of the experiment.

6.2 Add V2V-PAEB Simulation Model to Vehicle Model

When click on one of this vehicle model (Audi_A8_1) in Figure 6.3, we can go to inside of the vehicle model where all the supporting models required by the V2V-PAEB simulation model are readily available. Note that all the vehicles have not been equipped with the V2V-PAEB model yet, so we need to add the V2V-PAEB model to the requested vehicles. Figure 6.4 shows that the V2V-PAEB model is added to a vehicle model and connected with its supporting models. For the sake of simplicity only vehicle 1, vehicle 2 and vehicle 5 are equipped with a V2V-PAEB model.

As has been mentioned before, the V2V-PAEB simulation model accepts four types of input parameters (The inputs from Vehicle State Model, Radar Sensor Model, Camera Sensor Model and DSRC Receiver Model). Figure 6.4 has shown how to connect the V2V-PAEB model with these input simulation models. Additionally, the output of V2V-PAEB simulation model should also connect with the DSRC transmitter model or actuator models.

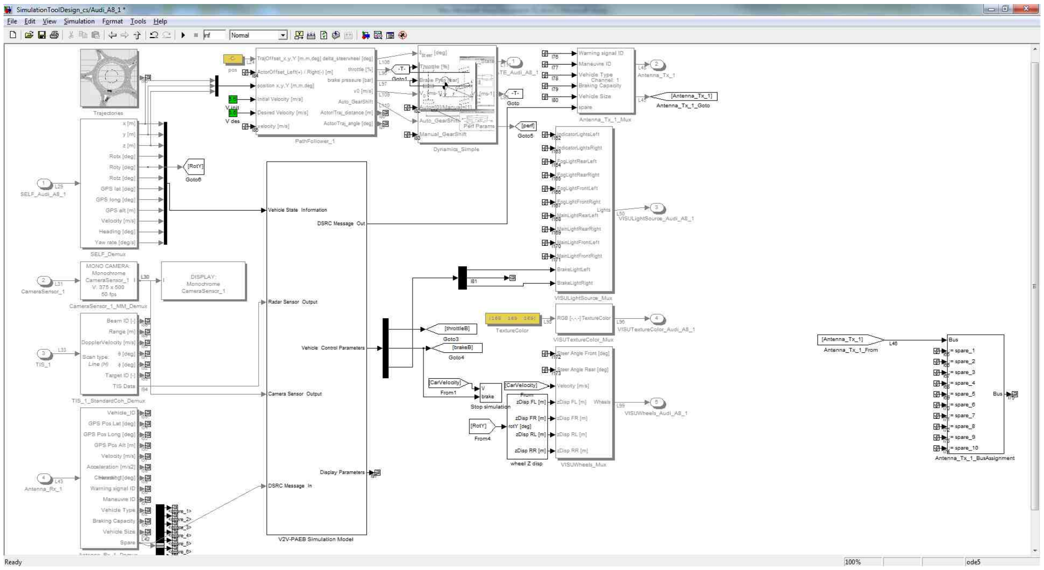
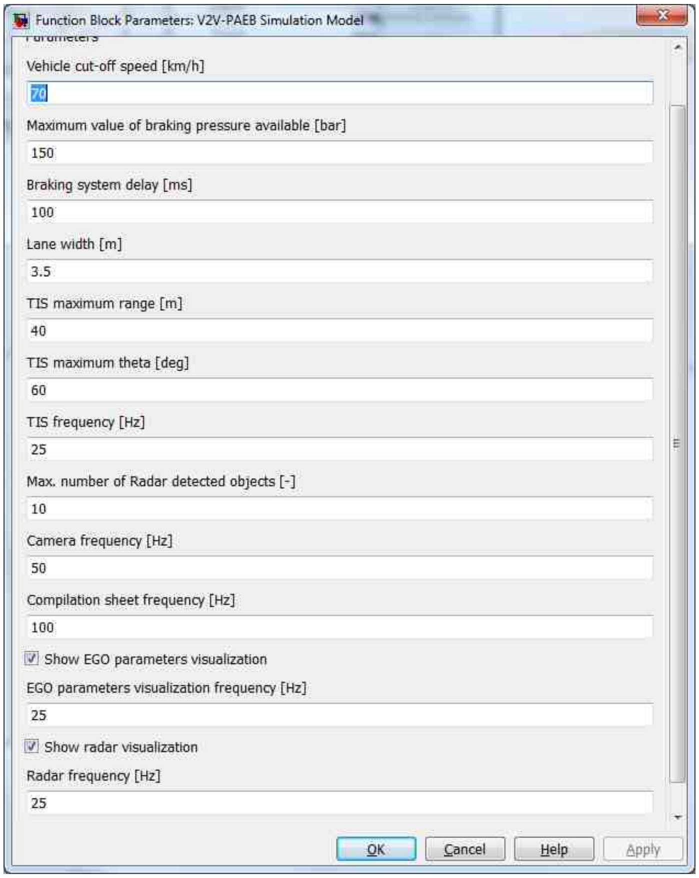


Fig. 6.4. The internal of Audi_A8_1 simulation model.

6.3 Configuration of V2V-PAEB Model

In section 5 we have discussed how the algorithms in each block have been implemented, so in this section we do not describe the internal algorithms of V2V-PAEB simulation model again. Since current implementation is not mature, we may need to modify the algorithms for some blocks of V2V-PAEB model in the future. When one block is modified, other blocks are not affected. So the algorithms can be easily modified and evaluated V2V-PAEB simulation model.

Before running the simulation, the V2V-PAEB simulation model should be configured properly. In section 5.1, all the configurations of the V2V-PAEB model have been presented. We use a graphic interface to configure these parameters. Figure 6.5 shows the configurations in this study.



Function Block Parameters: V2V-PAEB Simulation Model

Parameters

Vehicle cut-off speed [km/h]
70

Maximum value of braking pressure available [bar]
150

Braking system delay [ms]
100

Lane width [m]
3.5

TIS maximum range [m]
40

TIS maximum theta [deg]
60

TIS frequency [Hz]
25

Max. number of Radar detected objects [-]
10

Camera frequency [Hz]
50

Compilation sheet frequency [Hz]
100

☑ Show EGO parameters visualization

EGO parameters visualization frequency [Hz]
25

☑ Show radar visualization

Radar frequency [Hz]
25

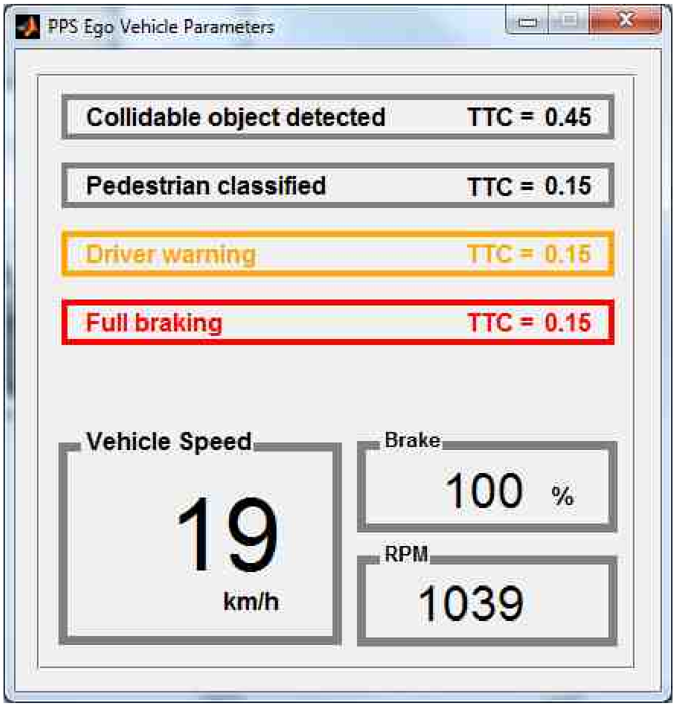
Fig. 6.5. The configurations of the V2V-PAEB simulation model.

In this study, two cases will be run separately to proof the V2V-PAEB system has a better performance than PAEB system. In this simulation experiment, vehicle 1, 2 and 5 are equipped with V2V-PAEB simulation model. When run the simulation for case 1, the V2V capability on vehicle 1, 2 and 5 is disabled, so they cannot send out V2V-PAEB Messages. And vehicle 5 cannot receive any V2V-PAEB Messages and it can only use PAEB system to handle the potential collision with the pedestrian. When run the simulation for case 2, the V2V capability on vehicle 1, 2 and 5 will be enabled and they can share the V2V-PAEB Messages with each other. So vehicle 5 can use not only its PAEB system and the received V2V-PAEB Messages to detect pedestrian and make safety decisions. In the V2V-PAEB simulation model, we have implemented a switch to enable and disable the V2V capability.

6.4 Simulation Result

Figure 6.6 is the simulation result of vehicle 5 for case 1. It shows that there was collision between vehicle 5 and the pedestrian with collision speed 19 km/h. The pedestrian was detected by the PAEB system when TTC was 0.45. However, at that time the PAEB system did not know this object was a pedestrian, and this pedestrian was classified when TTC was 0.15. Once the pedestrian was identified, both the driver warning and automatic braking were applied by the PAEB system at TTC equals to 0.15. Since it was too late and the PAEB system has no enough time to react, the collision was not avoided.

Figure 6.7 is the simulation result of vehicle 5 for case 2. It shows that the potential collision between vehicle 5 and the pedestrian was avoided successfully. The pedestrian was detected when TTC equaled to 1.99s. When TTC equaled to 1.57s the driver warning was triggered and the automatic full braking was started at TTC equaled to 0.59s. So the simulation results for vehicle 5 in case 1 and case 2 vary signicantly.



PPS Ego Vehicle Parameters

Collidable object detected TTC = 0.45

Pedestrian classified TTC = 0.15

Driver warning TTC = 0.15

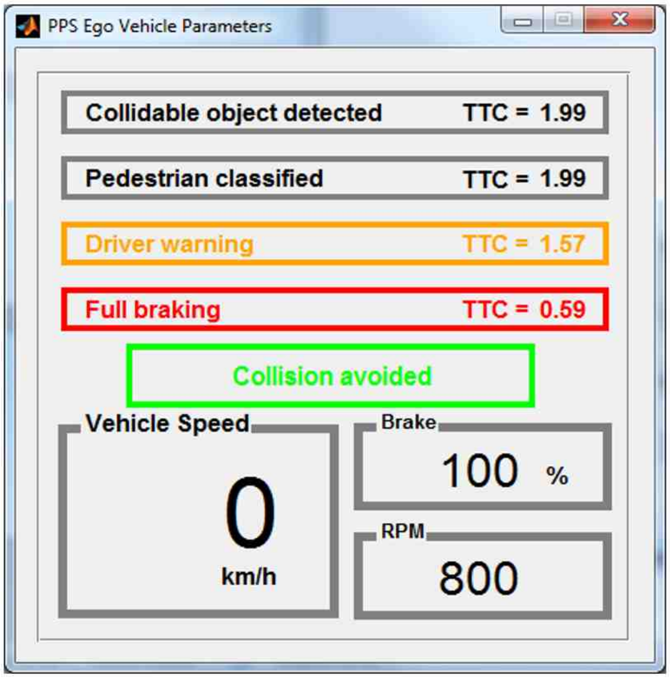
Full braking TTC = 0.15

Vehicle Speed: 19 km/h

Brake: 100 %

RPM: 1039

Fig. 6.6. The simulation results of case 1.



PPS Ego Vehicle Parameters

Collidable object detected TTC = 1.99

Pedestrian classified TTC = 1.99

Driver warning TTC = 1.57

Full braking TTC = 0.59

Collision avoided

Vehicle Speed: 0 km/h

Brake: 100 %

RPM: 800

Fig. 6.7. The simulation results of case 2.

Table 6.3 shows the comparison of the simulation results of case 1 and case 2. We can see that in case 2 the V2V-PAEB system on vehicle 5 can detect pedestrians much earlier than that in case 1. And the V2V-PAEB system can have more reaction time and has a better performance than PAEB system only. So the V2V-PAEB system can compensate the limitations of PAEB system, and improve pedestrian safety significantly.

Table 6.3. The comparison of the simulation results for case 1 and case 2.

Item	**Case 1**	**Case 2**
Pedestrian Detected before collision [s]	0.45	1.99
Pedestrian Classified before collision [s]	0.15	1.99
Warning generated before collision [s]	0.15	1.57
Full Braking generated before collision [s]	0.15	0.59
Collision Avoided [Yes/No]	No	Yes
Collision Speed [km/h]	19	NA

Usually, there are many factors can affect the simulation results. For example, the accuracy of various types of sensors or the speed for data processing. If the sensor accuracy is lower or the data processing slower, then the performance of this simulation model will be worse. In this study, in order to show that the V2V-PAEB simulation model is working, and to proof the V2V-PAEB system can improve the performance of PAEB system, we assume that the information provided by sensors are accurate and there is no delay for data processing. So the simulation results obtained in this study are the best performance that the V2V-PAEB system have.

7. CONCLUSION AND FUTURE WORK

This study discussed the advantages and limitations of PAEB system. The limitations of PAEB system is its short detection range and unable to detect objects being obscured. To further improve the performance of PAEB system, we proposed an idea that integrating the capabilities of V2V and PAEB to allow the information of pedestrians detected by PAEB system to be shared in the V2V network. Theoretically, the V2V-PAEB system may have better performance than PAEB system because other vehicles can report the pedestrians that one vehicles PAEB system failed to detect.

In this thesis, we defined the structure and information processing stages of V2V-PAEB system. This system consists of ten blocks and each block is designed to solve some specific problems. In additional, the input and output parameters of the V2V-PAEB system as well as its each block are also defined. For further studying the V2V-PAEB system, we also developed a V2V-PAEB simulation tool using PreScan software. Based on this simulation tool, we also developed the basic internal control algorithms of V2V-PAEB simulation model. And the simulation results show that the V2V-PAEB system can improve pedestrian safety significantly.

Since the V2V-PAEB system is quite complicated, this study cannot handle all the problems, and only part of the problems are solved. As a result, current implementation of the V2V-PAEB simulation has many limitations and can only work properly under some specific conditions. So one of the direction to improve the V2V-PAEB simulation model is developing the robust internal algorithms. Especially the algorithms for solving the problems of message information inaccuracy, message explosion, and message fusion and so on. Based on the predefined architecture and blocks of V2V-PAEB model, the internal algorithms are easily modified and upgraded. When one block is modified, other blocks are not affected.

In addition, current V2V-PAEB system is designed only for improving the road safety of pedestrians. However, there are many other types of objects that the V2V-PAEB system can benefit the safety. For example, the vehicles have no V2V-PAEB system, bicyclist, and animals and so on. The key point of this work is developing the algorithms for identifying type of the object. It is similar with pedestrian identification, but usually different classifiers are used.

Besides those, current V2V-PAEB system shares the pedestrian information only among vehicles. However, there are many other types of objects (such as infrastructures, pedestrians) have DSRC communication capability. So another direction to improve the V2V-PAEB system is to expend it to V2X-PAEB system. The "X" means everything.

REFERENCES

REFERENCES

[1] World Health Organization, "More than 270 000 pedestrians killed on roads each year," Retrieved from http://www.who.int/mediacentre/news/notes/2013/make_walking_safe_20130502/en/, Last Date Accessed: November 24, 2015.

[2] E. Coelingh, A. Eidehall, and M. Bengtsson, "Collision warning with full auto brake and pedestrian detection - a practical example of automatic emergency braking." Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference, 19-22 September, 2010.

[3] J. Harding, G. R. Powell, R. Yoon, J. Fikentscher, C. Doyle, D. Sade, M. Lukuc, J. Simons, and J. Wang, *Vehicle-to-Vehicle communications: Readiness of V2V technology for application.* Washington, DC: National Highway Traffic Safety Administration, 2014.

[4] M. Liu, S. Chien, and Y. Chen, "Improve road safety using combined v2v and pre-collision systems." The 24th International Technical Conference on the Enhanced Safety of Vehicles (ESV2015), 8-11 June, 2015.

[5] M. Heesen, M. Dziennus, T. Hesse, A. Schieben, C. Brunken, C. Loper, J. Kelsch, and M. Baumann, "Interaction design of automatic steering for collision avoidance: challenges and potentials of driver decoupling." IET Intelligent Transport Systems, 22 January, 2015.

[6] J. Tsuchida, S. Tokoro, H. Fujinami, and M. Usami, "The advanced sensor fusion algorithm for pre-crash safety system." SAE Technical Paper 2007-01-0402, 2007.

[7] T. Kawasaki, J. Tsunekawa, K. Harada, and K. Suzuki, "Development of front-side precrash safety system for intersection collisions." 16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services, 21-25 September, 2009.

[8] EURO NCAP Website, "Autonomous emergency braking," Retrieved from http://www.euroncap.com/en/vehicle-safety/the-rewards-explained/autonomous-emergency-braking/, Last Date Accessed: November 24, 2015.

[9] IIHS, "U.s. dot and iihs announce historic commitment from 10 automakers to include automatic emergency braking on all new vehicles," Retrieved from http://www.iihs.org/iihs/news/desktopnews/u-s-dot-and-iihs-announce-historic-commitment-from-10-automakers-to-include-automatic-emergency-braking-on-all-new-vehicles, Last Date Accessed: November 24, 2015.

[10] Wikipedia, “Radar, From Wikipedia, the free encyclopedia,” Retrieved from https://en.wikipedia.org/wiki/Radar, Last Date Accessed: November 24, 2015.

[11] M. Pijpers, “A literature study on the working principles and characteristics of frequently applied sensors in advanced driver assistance systems,” Retrieved from https://www.utwente.nl/ctw/aida/education/Rapport_MP.pdf, Last Date Accessed: November 24, 2015.

[12] Wikipedia, “Lidar, From Wikipedia, the free encyclopedia,” Retrieved from https://en.wikipedia.org/wiki/Lidar, Last Date Accessed: November 24, 2015.

[13] Wikipedia, “Thermographic camera, From Wikipedia, the free encyclopedia,” Retrieved from https://en.wikipedia.org/wiki/Thermographic_camera, Last Date Accessed: November 24, 2015.

[14] Wikipedia, “Image Sensor, From Wikipedia, the free encyclopedia,” Retrieved from https://en.wikipedia.org/wiki/Image_sensor, Last Date Accessed: November 24, 2015.

[15] Wikipedia, “Ultrasound, From Wikipedia, the free encyclopedia,” Retrieved from https://en.wikipedia.org/wiki/Ultrasound, Last Date Accessed: November 24, 2015.

[16] U.S. Department of Transportation, “Vehicle to vehicle communication technology,” Retrieved from www.safercar.gov/staticfiles/safercar/v2v/V2V_Fact_Sheet_101414_v2a.pdf, Last Date Accessed: November 24, 2015.

[17] SAE International, “DSRC implementation guide,” Retrieved from http://www.sae.org/standardsdev/dsrc/DSRCImplementationGuide.pdf, Last Date Accessed: November 24, 2015.

[18] P. Website, “Automotive vehicle-to-everything (v2x) communications market 2014-2024: V2v, v2i, v2p, v2h, in-v, its, car-to-x,” Retrieved from http://www.prnewswire.com/news-releases/automotive-vehicle-to-everything-v2x-communications-market-2014-2024-v2v-v2i-v2p-v2h-in-v-its-car-to-x-278426231.html, last Date Accessed: November 24, 2015.

[19] Tass International Website, “PreScan Overview,” Retrieved from https://www.tassinternational.com/prescan-overview, Last Date Accessed: November 24, 2015.

[20] M. Tideman and M. van Noort, “A simulation tool suite for developing connected vehicle systems.” 2013 IEEE Intelligent Vehicles Symposium (IV), 23-26 June, 2013.

[21] T. Gandhi and M. M. Trivedi, “Pedestrian protection systems: Issues, survey, and challenges.” IEEE Transactions on Intelligent Transportation Systems, 04 September, 2007.

[22] Wikipedia, “Mobileye, From Wikipedia, the free encyclopedia,” Retrieved from https://en.wikipedia.org/wiki/Mobileye, Last Date Accessed: November 24, 2015.

[23] GPS.gov Official Website, “GPS Accuracy,” Retrieved from http://www.gps.gov/systems/gps/performance/accuracy/, Last Date Accessed: November 24, 2015.

[24] O. Trullols-Cruces, M. Fiore, and J. M. Barcelo-Ordinas, “Understanding, modeling and taming mobile malware epidemics in a large-scale vehicular network.” IEEE 14th International Symposium and Workshops, 4-7 June, 2013.

[25] Y. Chen , K. Shen, and S. Wang, “Forward collision warning system considering both time-to-collision and safety braking distance.” IEEE 17th Industrial Electronics and Applications (ICIEA) Conference, 19-21 June, 2013.

[26] B. Tang, S. Chien, and Y. Chen, “Obtain a simulation model of a pedestrian collision imminent braking system based on the vehicle testing data.” IEEE 17th International Conference on Intelligent Transportation Systems (ITSC), 8-11 October, 2014.