Grenze International Journal of Engineering and Technology, Jul Issue

Rear View Object Detection based on Camera Sensors

[1]Dushyanth S and [2]Kavitha M
[1-2]Siddaganga Institute of Technology / Department of Computer_Science and Engineering, Tumkur, India
Email: dushyanthrajdushi@gmail.com, kavitham@sit.ac.in

***Abstract*—The US American government introduced a law to mandatorily equip passenger vehicles with rear view cameras. Furthermore, US NCAP presented a test for passenger vehicles to brake on pedestrians while back up. These two circumstances lead to main motivation of the development of the Backup Assist(BUA). Nevertheless, more and more passenger cars in general are being equipped with rear view cameras. Rear view system(RVS) allows to deliver a rear view camera system including a braking functionality which is intended to make the driving mission safer and reduce the number of accidents in backup driving situations. RVS also focus on vehicle safety by reducing acceleration of the vehicle gradually in order to avoid jerk and increase the ride comfort.**

***Index Terms*— US NCAP, Backup Assist(BUA), Rear view system(RVS), braking functionality, vehicle safety, acceleration, jerk, ride comfort.**

I. Introduction

One of the goals of the Backup Assist is to meet the goals of the US NCAP test. However, the functionality of the Backup Assist shall not be reduced to the particular test case defined by US NCAP. The scope of this paper is to brake on a vast variety of objects which are stationary as well as moving. As far as practicable, the function shall be developed independently of the used sensor principle and adaptable to several microprocessors. Backup assist is an addition to the rear view camera demanded by US law. It performs emergency braking on objects in the driving path. The driver is in full responsibility to perform the driving task and monitor the environment. Fall back for malfunctions of the backup assist is the driver. To support controllability by the driver and other involved persons and reduce severity of consequences, the speed range is limited. The focus of the US NCAP lies on a stationary pedestrian standing in the ego vehicle path as shown in fig. 1. The ego vehicle coasts backwards on a straight path. The goal is to avoid a collision with the pedestrian.
Automated driving systems (ADSs) are described in this study as promising a safe, enjoyable, offers a smooth driving experience The number of fatalities in vehicles equipped with ADSs, on the other hand, is rising. Unless the existing state-of-the-art in ADS robustness is enhanced, the full potential of ADSs will not be realized. The difficulties that have still to be resolved as well as the technological components of automated driving are examined. A complete analysis of evolving techniques and critical features such as localization, mapping, perception, planning, and human machine interfaces was conducted after an overview of available datasets and tools for ADS development.
Various Advanced Driver Assistance (ADAS) have been created in order to address growing social issues such as traffic congestion and accidents. Parking Assistance System (PAS) is one of the ADAS applications that improves vehicle safety and driver comfort during the parking procedure [1]. Currently, ultrasonic sensors and back cameras are used in PAS, and numerous commercial systems have been released. The detecting range and

Grenze ID: 01.GIJET.7.2.511

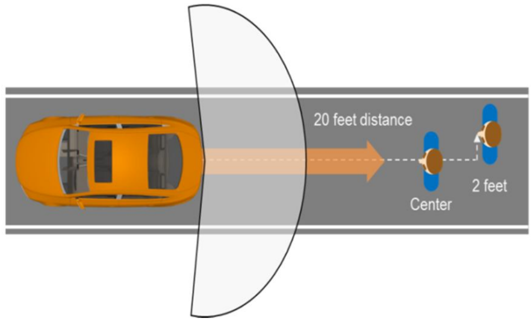

Figure 1. Stationary pedestrian standing in ego vehicle path

precision of ultrasonic sensors, on the other hand, are limited. Because the visible field of the back camera is confined to the self-backside vehicle's, it’s difficult to detect parallel parking lots.

The amount of acceleration and jerk (the amount of acceleration change), are the factors that contribute to the unpleasant ride. The inertia force is applied to the passenger and the body as a result of the vehicle's acceleration. The passenger feels more discomfort as the magnitude of the force or the change in force increases. As a result, the longitudinal braking situation is controlled in this study to improve ride comfort [2].

The following is the principle behind the jerk that occurs when the vehicle comes to a halt. Even if the vehicle decelerates slightly while coming to a halt, the acceleration drops to zero as soon as the vehicle comes to a halt since the vehicle's relative motion with the ground ceases. As a result, the amplitude of the vehicle's acceleration is considerably reduced in a discontinuous manner, resulting in a very big jerk. Due of the abrupt shift in acceleration, the passenger experiences the uneasy sensation of turning back. In this study, From immediately before stopping, the amount of abruptly shifting acceleration is gradually reduced, allowing the vehicle's instantaneous acceleration to converge to zero. If control is achieved and successful behaviour is achieved, acceleration and velocity converge smoothly to zero at the same moment, and the jerk is reduced greatly [3].

II. PROPOSED SYSTEM

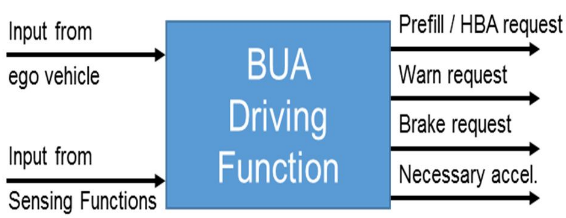

Figure 2. Inputs and outputs of the BUA driving function

The BUA function calculates if an object represents a collision threat. Therefore inputs from the ego vehicle CAN and data from Sensing Functions are evaluated by the BUA function, which is shown in fig. 2. Outputs are the requests for a prefill, warnings or braking.

Prefill / HBA Request

A prefill or HBA request shall lead to a preconditioning of the brakes. The signal is Boolean.

Warn Request

A warn request shall lead to a warning of the driver. This can be realized in various ways in the vehicle. A warning of the driver could be implemented as an acoustic, visual and/or haptic information. The signal is Boolean.

Brake Request

A brake request shall lead to an autonomous braking of the ego vehicle. The brake request might be applied in several steps/levels of different maximum decelerations. The later a braking of the vehicle is requested, the higher the acceleration has to be to avoid a collision. The signal is Boolean.

Necessary Acceleration

The necessary acceleration resp. deceleration to stop before colliding with the potential threat in m/s² is another output which will be calculated within the BUA Driving Function. Usually it is used together with HBA and/or the brake request to determine which deceleration should be applied. The signal leads to the deceleration realized by the brakes of the car. The signal is a float.

Warn Request Timing

The timing of the warning strongly depends on its manner. A visual warning could be requested earlier than an acoustic or haptic warning, due to the fact that it is less prominent. The reaction time of the driver has to be considered in the timing of the warning. On the other hand, a warning which is timed to early might lead to an increased number of false warnings. The final application of the warn request timing has to be aligned with the stakeholders respectively customers.

Brake Request Timing

Similar to the warn request timing the timing for the brake request depends on its characteristics. The brake request could be realized as a comfort brake with low deceleration, in different steps/levels (with different maximum decelerations) or as an emergency brake. Again, an early timed braking might lead to an increased number of false activations, which makes an alignment with the stakeholders indispensable. Furthermore, an early comfortable braking might lead to misuse of the functionality by the driver. If the risk exists that driver relies on the functionality while backing up, a misuse detection has to be implemented, since BUA is planned as a Level 2 functionality. Another possible characteristic of the function might be that it is not designed to always only brake down to standstill, but to reduce respectively limit the velocity. The use case for this implementation could be a possible threat which is not within the ego vehicle path but close to it.

Misuse Prevention

A misuse of the function by the driver must be prevented. Depending on the design of the function several options to avoid the misuse are possible. If the brake request, described in the previous chapter, is timed very late (emergency brake) the driver will rather not rely on the brake function to stop in front of an obstacle. Depending on the used hardware on which the algorithm runs on or customer demands, the object detection has to be limited to certain classes. If the function is designed to brake only for pedestrians a misuse becomes more risky and thus more unlikely.

On the other hand, if the function is designed very comfortable (comfort brake) and all obstacles are classified to brake on, the risk of a misuse of the function increases. In this case a misuse detection needs to be implemented that informs the driver to not rely on the function. If it is recognized that the driver does use the function to stop comfortably, e.g. by using a counter, the functions shall turn off and inform the driver [4].

A. SIL Simulation In Carmaker

"Software-in-the-loop testing," or SIL testing, is a test methodology in which executable code, such as algorithms (or even a whole controller strategy), is tested within a modelling environment that can help prove or test the software, which is usually created for a specific mechatronic system. It is frequently necessary to simulate aspects of the system that will eventually be present but are not required or would be too difficult to use, such as mechatronic hardware and sensors, in order to be able to test versions of the software. To conveniently simulate software, it is common to run the code on a laptop using a locally-hosted compiler, and then deploy the code on the target controller using a cross-compiler once the code has been adequately tested. Some signals can often be judged and dismissed as inconsequential, while others must be present. The judgement of what is required against what is not is normally made by the testing team, but it can be aided by using tools such as simulated and stimulated values. Throughout these software testing stages, using modelling tool chains like MATLAB Simulink from The Math Works can greatly improve the ability to integrate software code execution into an entire simulated environment. If necessary, third-party toolboxes can be employed to model specific mechanical systems. In addition to simulating the environment, a desktop GUI application like VISUALCONNX may connect and control components of the modelling environment, known as the 'plant model.' Providing stimuli can also be beneficial. such as loading, wear, noise, and so on, using a desktop GUI tool, such as VISUALCONNX, which can connect and control aspects of the modelling environment. Carmaker for Simulink is a complete Carmaker integration as shown in fig. 3, MATLAB and Simulink modelling and simulation environments with IPG's open integration and test platform. Carmaker is a sophisticated software tool for vehicle dynamics modelling, testing, and development for automobiles, motorcycles, and trucks. Carmaker can be used to develop automotive control systems such as ABS or ESP, ACC, EPB, ARS, AFS, as well as allwheel drive and engine control systems. Carmaker features were introduced to the Simulink environment using S-functions, MATLAB, and Simulink API functions. The combination of Carmaker with Simulink is not a loosely connected simulation, but rather a closely coupled combination of the two programmes that results in a quick and stable simulation environment.

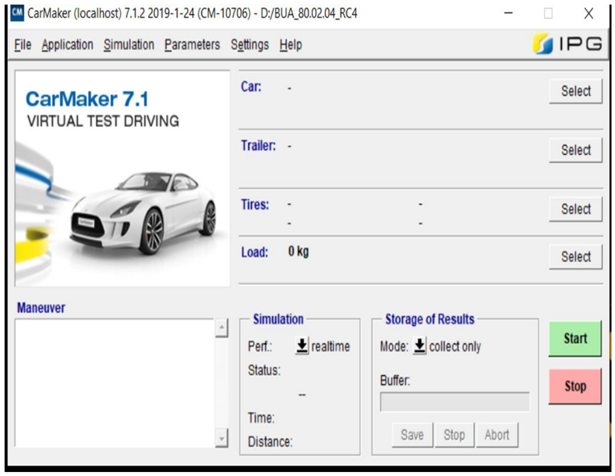

Figure 3. CarMaker

In general, SIL testing and simulation do not necessitate real-time execution because the code algorithms and functions are being demonstrated to execute various tasks. It may even require specific drivers or hardware to function in real-time. Due to the code's simplicity, the software will sometimes run much quicker than real-time, and at other times it will run significantly slower. If realistic timing information is needed, it can be reproduced numerically or pseudo real-time blocks can be employed in the modeling environment to pace the model as needed. Today is the day to put the vehicles of the future to the test! Our software products assist you in using the front-loading principle in your development processes, allowing you to plan development cycles more quickly. Tests may be transferred to the virtual world at an early stage – even before a real-world prototype is built – and are 100% repeatable thanks to our simulation technologies.

Our products' maneuver event, and scenario-based testing approaches reflect our goal of mimicking real-world test runs as accurately and reproducibly as feasible. Visualization features for a clear and trustworthy analysis of your test run, as well as the option of test automation and automated analysis, round out our virtual test driving package. A wide selection of product extensions is also available for your specific application scenarios if needed.

B. Smart Car Cantrol Model

A quick deceleration, for example, will make passengers uncomfortable in a unique car follows model centred on passenger comfort. The automobile following brake comfort model was created based on the relationship between vehicle deceleration and passenger comfort levels. By analysing the distance between the controlled car and the preceding automobile, as well as the controlled car's velocity, the model determines the controlled car's acceleration. The model is able to keep riders comfortable by managing the car's acceleration. The coefficient of friction between the car and the road surface is also taken into account. Experiments reveal that the model is extremely accurate in real-world scenarios.

Intelligent transportation systems are being developed with the purpose of improving driver comfort, reducing traffic accidents, and improving traffic flow. In real-world traffic circumstances, the traffic simulator is a useful tool for studying human dynamics, traffic systems, and vehicle designs. In order to develop a realistic and intelligent traffic environment, driver behaviour models must be investigated. The three key subjects of modern driver behaviour modelling research were driver performance and capacity, longitudinal driver behaviour models, and driver skill models [5].

Modeling driver behaviour relies heavily on car following models. Almost every traffic simulation has them. Within the same lane, car following depicts the interactions between an automobile and its preceding cars. Several car following models are currently available. When compared to other driving activities, car follows remains a point of attention due to its comparatively simple components. It's an important aspect of driving that's been well-represented by mathematical models. Car follows research aids not just in the overall understanding of traffic flow, but also in the prevention of highway collisions. Ride comfort is affected by vibration, acoustic sound, fragrance, temperature, visual stimulation, humidity, and seat design. There has been a lot of research into human vibrations and ride comfort, particularly in the area of vertical vibrations that cause ride pain. Despite the fact that ride discomfort induced by the use of emergency brakes is a common occurrence, little research has been done on the subject. A car follows model directs the driver's actions in relation to the vehicle in front of them in the same lane. A diagram depicting a car follow scenario is as shown in fig. 4.

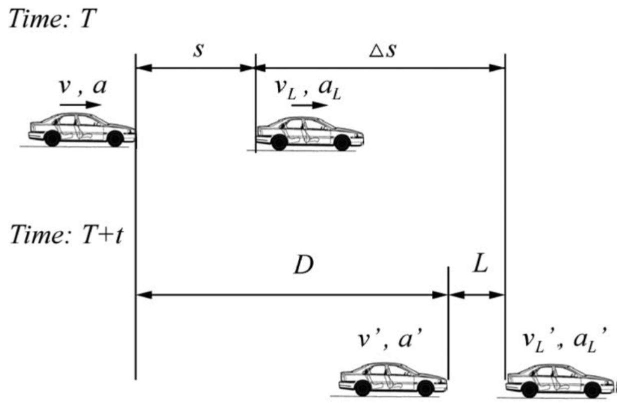

Figure 4. Car follows a model based on velocity and space

Assume that a driver will utilise a safety headway to follow a car, allowing him to stop even if the previous car uses its emergency brake. We also assume that the car's braking deceleration is consistent over all brake times, and that the drivers' reaction times are consistent as well. With a velocity of v and a space headway of s, the following automobile follows the preceding car. At time *T*, The previous automobile applies its emergency brake, and the following car driver knows that he needs to brake to reach his intended speed at time T + and begins to do so. The brake acceleration of the following vehicle is a. In the worst-case scenario, we assume the following car will have to come to a complete stop. The spacing headway is L when the car in front of you stops. From time T + to the instant the following automobile comes to a halt T + t, the displacement of the following car and the preceding car is D and s, respectively.
The proposed model is based on the preceding scenario. The (1) describes the displacements of the two cars at the precise moment when the following car stops.

$$D + L = s + \Delta s. \qquad (1)$$

C. Parking Lot Detection And Tracking

The Parking Assistance System (PAS) is a helpful tool for new or inexperienced drivers in difficult urban parking situations. In recent years, a number of automakers have proposed ultrasonic sensor-based PAS and rear-view camera-based PAS. Ultrasonic sensors, on the other hand, have a detection range of less than 3 metres, therefore the data can't be used to extract further information like obstacle recognition. In circumstances where a wider view is required, like as parallel parking, rear-view camera-based systems are ineffective. In this research, we propose a parking lot recognition technique based on surround view. An efficient tracking strategy was presented to address the tracking problem when recognised parking places were falling out of the surround view. The proposed algorithm's usefulness was demonstrated by experimental findings in both a simulation and a real-world setting.
Various Advanced Driver Assistance System (ADAS) have been created in order to address growing social issues such as traffic congestion and accidents. Parking Assistance System (PAS) is one of the ADAS applications that improves driver safety and comfort during the parking procedure.
Currently, ultrasonic sensors and back cameras are used in PAS, and numerous commercial systems have been released. The detecting range and precision of ultrasonic sensors, on the other hand, are limited.
Because the visible field of the back camera is limited to the rear side of the self-vehicle, it is difficult to detect parallel parking lots. A good answer to these issues is a surround view system as shown in fig. 5. Surround view adjusts for fisheye distortion in original photos acquired by four cameras on four sides of the vehicle, converts the undistorted views into bird's-eye view images, and then assembles surround view by matching four bird's-eye

view photographs. Surround view technology creates a virtual image that intuitively displays the situation all around the car, eliminating blind spots.
Because of the surround view image prevents perspective distortion of items attached to the ground surface, the geometrical features of parking lot lines can be employed to aid in the recognition of parking spaces painted on the ground surface. Furthermore, surround view is offered for the self-vehicle, allowing the driver to readily watch the ongoing parking operation [6]. Fig. 6 depicts the processing flowchart for the proposed surround view based parking lot recognition and tracking algorithm. The construction of a surround view using four cameras and video input will be discussed in greater detail. A line filter and probabilistic Hough transform will extract all candidate line segments as prospective parking lot line markers in our pre-processing section.

Figure 5. Surround view system

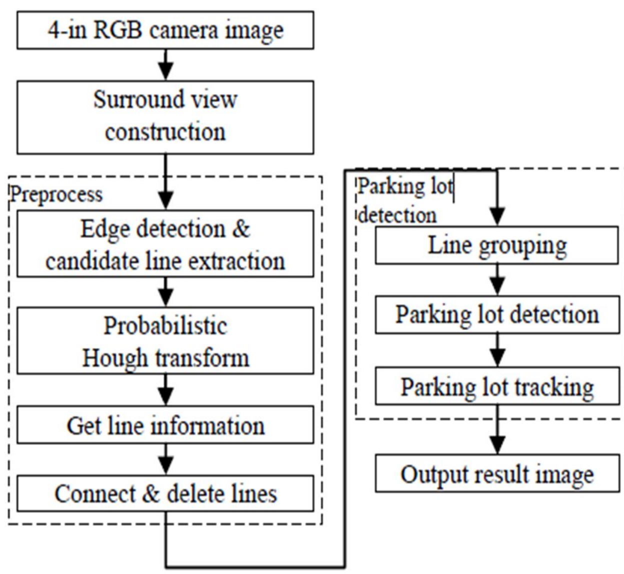

Figure 6. Surround view system block diagram

Surround view systems were brought to the market to let drivers see in all directions with no blind spots. Surround view system shows a virtual image taken from the sky that depicts the situation around the vehicle within a certain range. It corrects for fisheye distortion in four-camera input photos, converts the undistorted images to bird's-eye view images, and then matches four bird's-eye view photographs to create the self-around vehicle's view. Tracking is a big part of parking assistance. The movement of the surround view picture can be conceived of as a two-dimensional rotation and translation because the surround view will remain constant if the ground is considered plane. As a result, the surround view image tracking technique can be realised if the two-dimensional rotation and translation of the object in the current and previous frames can always be provided.

D. Intelligent Driver Assistance System

The "intelligent driver," also known as "EyeDriver," is a driver assistance system that uses four distinct components to analyse and provide real-time driver assistant data. The main components are drowsiness detection and head pose estimates, over-speed detection, lane departure, and front collision avoidance. This little bundle includes a Raspberry Pi board, a USB camera module, a Pi camera, and a TFT LCD. Because the "EyeDriver" is Sri Lanka's first affordable aftermarket solution, it can be installed and customised in any car with no effort and without the need for technical knowledge. The component for sleepiness detection and head pose estimates will track the driver's eyes and identify if his or her head posture is inconsistent or out of the optimal range. The vehicle's actual speed is compared to the recommended speed for the road, and if it is higher than the allowed, the system delivers a warning. This is handled by the over-speed detecting component. The lane departure component assists in maintaining vehicle stability in the selected lane on the road. When the motorist makes an intended lane change, the system also sends a notification. The front collision avoidance component will detect the road's frontal obstruction and issue a pre-collision/proximity warning signal. The notice is based on the vehicle's speed and the object's distance from the cars. The Raspberry Pi 3 Model B+ board serves as the foundation for the entire system, which was built using OpenCV and Python.

The "EyeDriver" system can detect drowsiness, overspeed, lane departure statuses, and forward crashes by estimating the driver's attentiveness. The "EyeDriver" system can assist drivers with these duties in order to make driving safer and more efficient. In the realm of automobiles, there are some aftermarket and built-in devices that can help the driver. All of these are out of reach for most people. They've also only covered a few specific characteristics. In the actual world, there is no such thing as an all-in-one product that anyone can instal and set up. The "EyeDriver" driver assistance system is simple to instal in any car and is also very cost effective.

A Raspberry Pi-based cost-effective car collision avoidance system was created to detect obstacles in real time [7]. Using OpenCV and Python, the system is implemented on the Raspberry Pi Module. For object identification and image capturing, a Logitech USB webcam is used. For effective edge detection, Laplacian, Sobel, and Canny edge detection techniques are used. When there are barriers in front of the car and unseen regions of the vehicle, a predictive vehicle collision avoidance system employing Raspberry-PI sends out warning notifications to the driver. Radar and one dash camera are used as vehicle detecting techniques for collision avoidance systems.

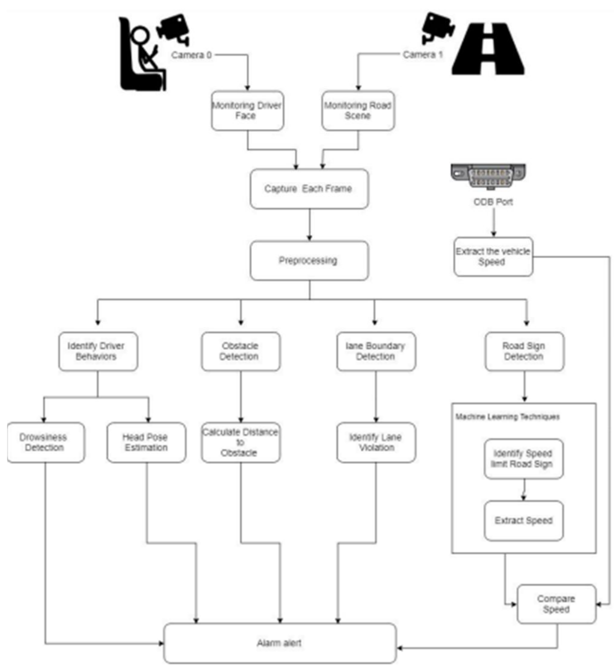

Figure 7. Eyedriver system

When the car's front end approaches another vehicle, the radar system detects that vehicle. The device then uses sound and visual cues to inform the driver. The object detection approach was used in the majority of the aforementioned forward collision warning (FCW) studies. However, they are unable to determine the distance to the object. Our FCW system can detect the distance between the vehicle and the objects and issue a correct and timely warning notice. Our system also features a user-friendly interface, and the driver will be notified step by step about the dangerous condition, with notifications such as sounding an alarm and displaying a warning alert. The system is separated into two parts to permit high efficiency and accuracy driver assistant functionalities:

1. Detection of drowsiness and estimation of head posture.
2. Detection of speed
3. Departure from the lane
4. Detection of forward collisions

A quick rundown of the "EyeDriver" system's operation is as shown in fig. 7.

The technology senses drowsiness and calculates head pose. A pi camera is used in the "EyeDriver" system to capture the driver's face. Camera calibration determines the focal length of the camera, as well as skew, distortion, and picture centre factors. The camera matrix values and camera distortion are calculated using photographs of the chessboard taken with the Pi camera from various angles and distances. These data will be used by the system to extract the driver's face's 3D object points. To collect 2D points of the essential face structures, the algorithm must first recognise the driver's facial landmarks (eyes, eyebrows, nose, mouth, and jawline). The system first uses Haar's Cascade method to locate the driver's face. The system then uses the facial landmark detector idea to extract the important facial components from the face.

III. Results

Every year, commuters in the United States travel almost 20 billion miles by bus and 9 billion miles by commuter rail. Riding comfort during the national average 24.4 minute commute to work is influenced by a number of factors. Some of these factors are noise, temperature, humidity, and motion. The factor of motion, including acceleration and jerk, will be investigated as a transit vehicle comes to a complete stop. The acceleration varies quite quickly when a vehicle reaches zero speed and does not reverse direction (almost discontinuously). As a result of the quick change in acceleration, all elements in motion with the vehicle, including its occupants, are subjected to a significant shock. Jerk is also known as jolt, which is an equally appropriate description of this action. Various automatic driving control approaches have recently been proposed with the goal of achieving a fully automated car. However, these methodologies did not adequately address passenger comfort while riding. In automatic driving, needless abrupt handling and sudden acceleration/deceleration may occur. The passengers are nervous and uncomfortable.

The amount of acceleration and jerk, or the variation in acceleration, are two aspects that contribute to the uncomfortable ride. As a result of the vehicle's acceleration, the inertia force is applied to the passenger and the body. As the magnitude of the force or the change in force rises, the passenger becomes increasingly uncomfortable. As a result, in order to improve ride comfort, the longitudinal braking situation is managed in this study. By specifying the discomfort index for the ride comfort, you can create a desirable acceleration trajectory that minimises the discomfort index. Furthermore, for riding comfort, an acceleration controller capable of keeping up with this is planned to conduct brake control [8]. Ride comfort is affected by vibration, acoustic sound, fragrance, temperature, visual stimulation, humidity, and seat design. There has been a lot of research into human vibrations and ride comfort, particularly in the area of vertical vibrations that cause ride pain. Despite the fact that ride discomfort caused by emergency brake use is prevalent, little research has been done on the subject.

The principle behind the jerk that occurs when the car comes to a halt is as follows. Even if the vehicle decelerates significantly while coming to a complete stop, the acceleration drops to zero as soon as the vehicle's relative motion with the ground ceases [9]. As a result, the vehicle's acceleration amplitude is drastically reduced in a discontinuous manner, resulting in a violent jerk. Because of the sudden change in acceleration, the passenger experiences the uneasy sensation of turning back. In this study, the amplitude of the abruptly changing acceleration is gradually lowered, beginning shortly before stopping, such that the vehicle's instantaneous acceleration can be reduced to zero [10]. If control and behaviour are both successful, acceleration and velocity converge smoothly to zero at the same time, and the jerk is considerably reduced.

IV. Conclusions

The intelligent transportation system is designed to improve driver comfort, prevent accidents, and improve traffic flow. The automation of the cars by including various functions in rear view camera to assist the driver thereby reducing the collisions while moving backwards and improving the ride comfort. The speed of the vehicle is decelerated while travelling backwards, and this research proposes efficient and adorable ways for driver assistance by recognising the obstruction using a camera module. This improves the comfort and safety of the driver by reducing vehicle jerk.

REFERENCES

[1] https://www.trafitec.dk/sites/default/files/publications/braking%20distance%20%20friction%20and%20driver%20behaviour.pdf

[2] https://peer.asee.org › an-investigation-of-acceleration-and-jerk-profiles- of public transportation vehicles

[3] https://www.researchgate.net/publication/234054306_Modeling_of_driver_following_behavior_based_on_minimum-jerk_theory

[4] https://www.iso.org/standard/7612.html: ISO 2631-1:1997.

[5] A smart car control model for brake comfort based on car following "*zhaohui wu, senior member, IEEE, yanfei liu, and gang pan, member, IEEE", no.1 pp.42-46,2008.*

[6] Surround View based Parking Lot Detection and Tracking *"Kazukuni Hamada*, Zhencheng Hu*, Mengyang Fan** and Hui Chen**",2015 IEEE Intelligent vehicles symposium pp.1106-1111, 2015.*

[7] EyeDriver: Intelligent Driver Assistance System "*Prabas Gayadeepth, Tharindu Pushpitha Baddewithana, Kalpana Vijintha Pannegama, Chamari Sandarekha Samarakkody, Dr. Pradeepa Samarasinghe, Ms. Samanthi Siriwardana"* 2019 IEEE 14th International Conference on Industrial and Information Systems (ICIIS), 18-20 Dec., Peradeniya, Sri Lanka.

[8] Araki, Takuma, Hidetaka Ohta, Yudai Fukuta, and Masahiro Oya. "Driver Braking Control Model for Vehicles." In *2018 International Conference on Information and Communication Technology Robotics (ICT-ROBOT)*, pp. 1-4. IEEE, 2018.

[9] Martin, Delton, and Dale Litwhiler. "An investigation of acceleration and jerk profiles of public transportation vehicles." In *2008 Annual Conference & Exposition*, pp. 13-194. 2008.

[10] Lee, Jonghyup, and Seibum Choi. "Braking control for improving ride comfort." In *MATEC Web of Conferences*, vol. 166, p. 02002. EDP Sciences, 2018.