SRS FOR COLLISION AVOIDANCE SYSTEM USING MACHINE LEARNING ALGORITHM

GROUP- 40

Requirements Analysis

* Environment Simulator – Unity/Unreal Engine
* Camera, sensor
* Machine Learning Algorithm – R-CNN, Hough transform, SVM, Random Forest, etc.
* Platform- Python, C++
* Operating System- windows 10 64-bit

Function Specification

* The function of the system is to alert or warn the driver beforehand to avoid collision.
* The distance and the speed are directly proportional, i.e. the speed should reduce with reducing distance.
* If the distance is not reducing and the driver is not in condition of applying breaks then the system applies breaks, this happens when distance of object is closer to vehicle.
* Speed will reduce according to distance reaches to certain threshold value.

External Interface Specifications

1. User Interface

* The user interface will be deployed in the actual car of the customer.
* It will tell the driver information which is not available on the dashboard such as distance with the vehicle in front of it.
* It will consist of a screen which will help the driver give alerts or warnings.

1. Communication Protocol

* Communication interface will communicate the result acquired by the software to the person driving the car, this can be done by fitting a buzzer or an alarm which will get activated when there is possibility of a mishap happening.
* The screen in the car will also help to communicate by displaying alerts signs on the screen. Or message such as “Vehicle needs to slow down”.

1. Hardware Interface

* The collision avoidance project of ours works in a virtual environment, for actual implementation of this project various other mechanical aspects would be needed to be taken into consideration.
* The implementation may involve other mechanical complexities such as actually using a car, being able to fix the screen in the car without excess difficulties, the car being able to take physical actions on indication of the software.

1. Database Backend

* The backend database will be a CSV file which will contain parameters such as speed, distance, acceleration, braking power, etc.
* The algorithms will train this database and calculate the possibility of an accident.
* The database will help to achieve maximum accuracy.

Technical specifications

1. Hardware Details :

* Ram : 8GB
* GPU : Graphics Card with DX9 support
* Hard Disk Storage : 1TB
* CPU : 2.5GHz

1. Operating System :

* Microsoft Windows

1. Performance :

* Processing time required by the system to process the input and take action for possible collision should be less than 3 seconds.
* Video input around 10 frames per second.
* Handle large amount of data generated in form of images from the video input.

1. Programming Languages and technologies :

* C#/C++
* Python
* Unity/Unreal

1. Versions of components used :

* Microsoft Windows 10 - 64-bit
* Unity 2018.x.x
* Unreal Engine 4