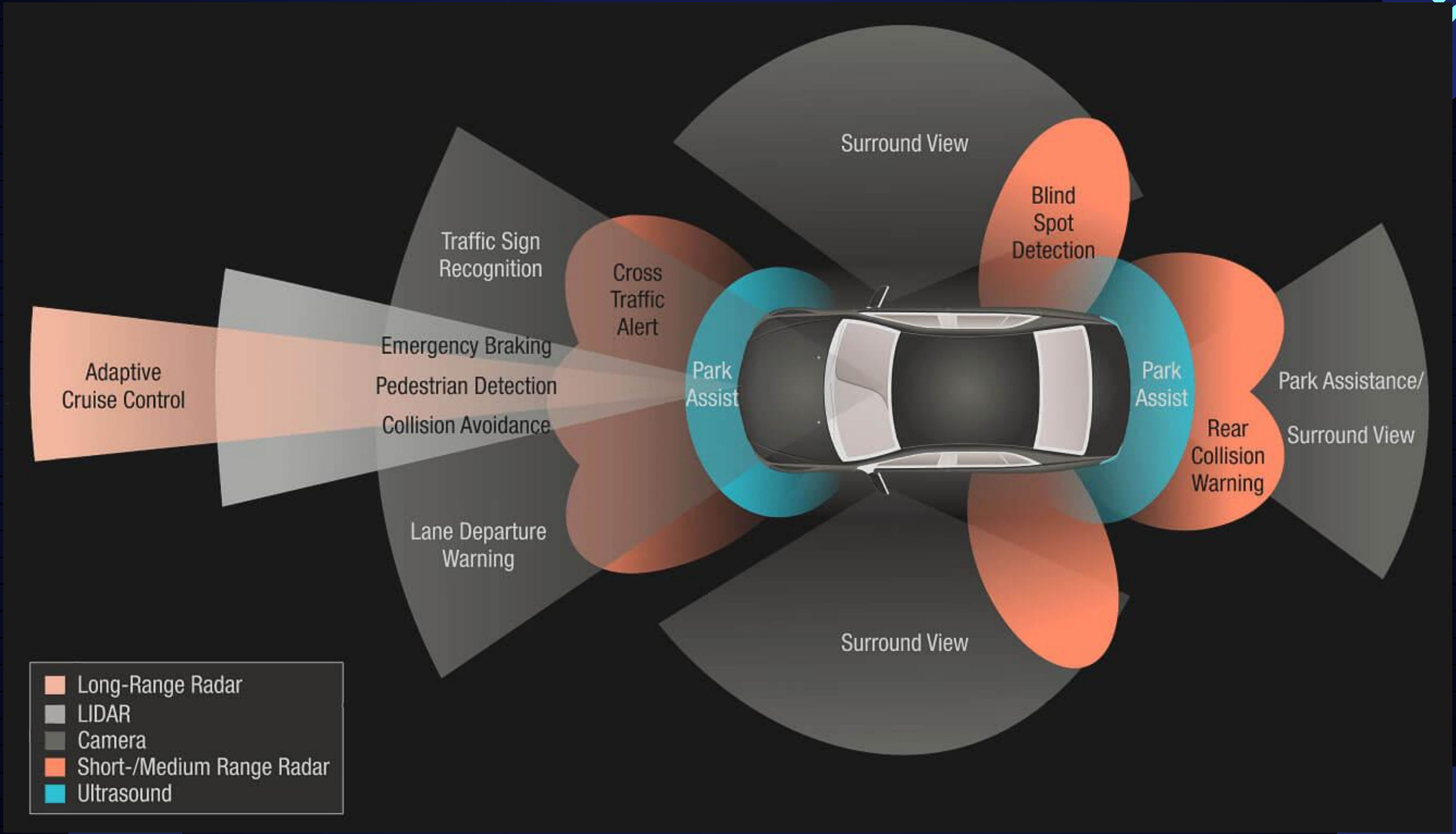
Advanced Driver-Assistance Systems

Presented by **Elyes Khechine**

GitHub

ADAS in Modern Autonomous Vehicles



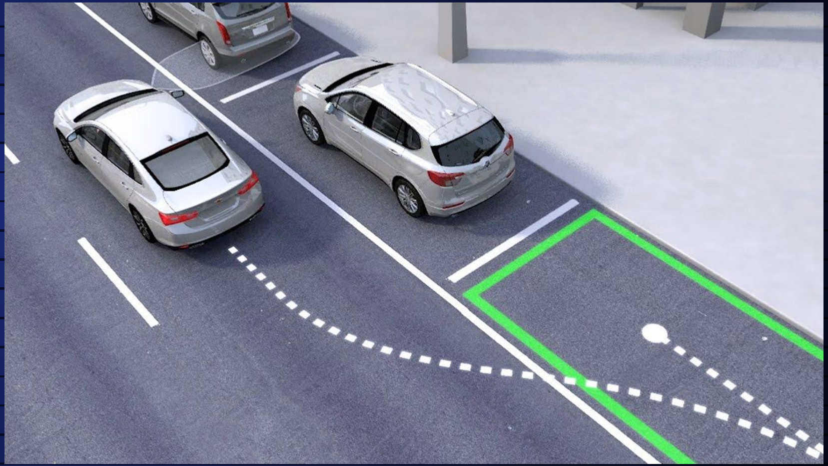
01

Park Assist

Also known as Automatic Parallel Parking (APP)

What is it?

Helps guide the driver into a parallel parking spot after searching and finding a viable option. The driver still is responsible for braking and monitoring their environment.



How does it work?

01

Activation

Advanced sensors read the gaps between cars in the area where the driver wants to park. The feature won't activate if there isn't sufficient room to parallel park, which helps ensure that the car won't bump into any nearby cars.

02

Wheel Take Over

When activated, this feature can then take over some of the vehicle's steering and acceleration functions needed to park.

03

Braking

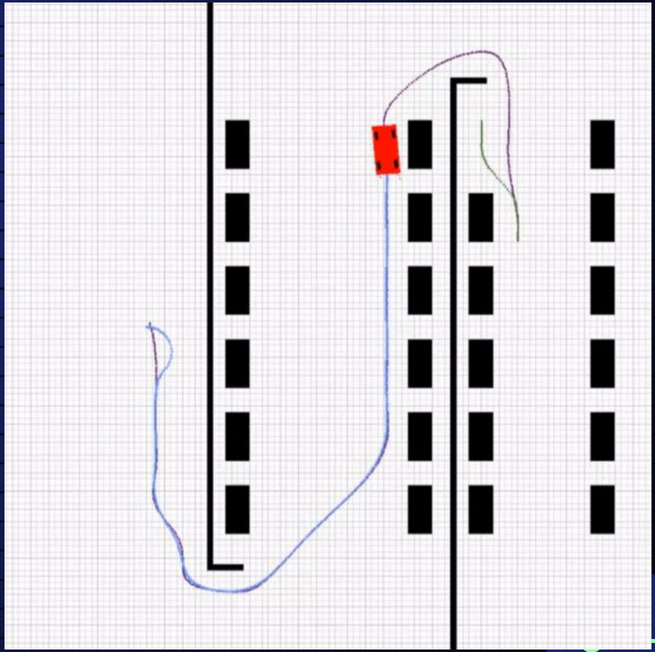
The driver is responsible for braking when using automatic parallel parking. They can override the automatic parallel parking maneuver by grabbing the steering wheel.

Automatic Parallel Parking Implementation Example

This repository contains a python implementation of an automatic parallel parking system in a virtual environment that includes path planning, path tracking, and parallel parking. The agent navigates its route through the environment and is directed to the assigned park location by the MPC controller.

Repo Link: https://github.com/Pandas-Team/Automatic-Parking

At first, the agent will find a path to park position then it will compute the arriving angle. Based on the arriving angle, the agent chooses a coordinate as ensure1. After that, the parking path is planned from ensure1 to ensure2 using 2 circle equations. MPC (Model Predictive Controller) controls the agent and car parks in ensure2 coordinate.



02

Lane Keeping Assist (LKAS)

What is it?

Lane keeping assist uses a video camera to detect the lane markings ahead of the vehicle and to monitor the vehicle's position in its lane. It then provides automatic steering and/or braking to keep the vehicle in its travel lane.

How does it work?

01

Lane Monitoring

The driver may receive an alert via a sound, flashing light or vibration if they drift out of the lane.

02

Wheel Take Over

If the driver doesn't take action, this feature may gently steer them back to the center of the lane.

03

Engage

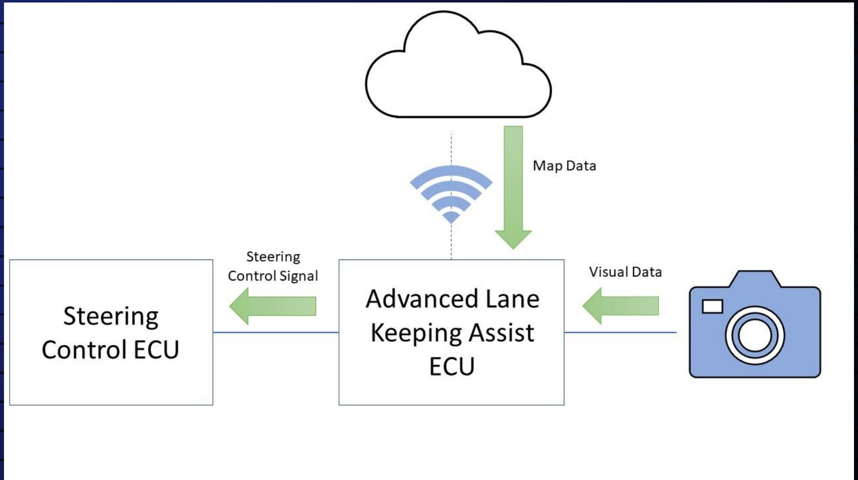
Slightly pulling the wheel will disable the lane keeping assist on most cars.

LKAS Implementation using Matlab and Simulink.

This project was implemented using Matlab and Simulink. The main algorithm used in the Matlab section of the project was the **Hough transform** as well as the **Sobel edge detection** algorithm.

Repo Link:
https://github.com/EmmaNiBhriain/Lane-Keeping-Assist

It takes data from external sensors such as cameras to analyze the road ahead. This data is combined with data from the cloud which indicates the type of road that the car is on, for example if the car is on a one-way road it does not need to calculate the position of two lanes as there will be no on-coming cars. In the event that the car moves onto a road where there is road markings visible, this system will turn off and another lane keeping assist algorithm can take over.

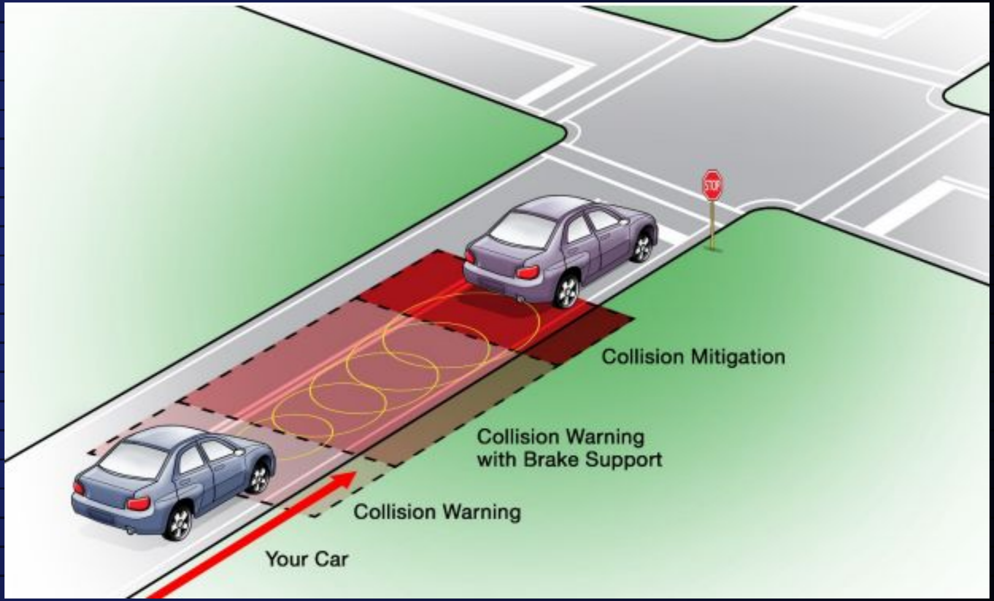


03

Automatic Emergency Braking (AEB)

What is it?

Automatic Emergency Braking (AEB) is a system that applies the brakes to their maximum to avoid a stopped car ahead or a car turning in front of the driver.



How does it work?

01

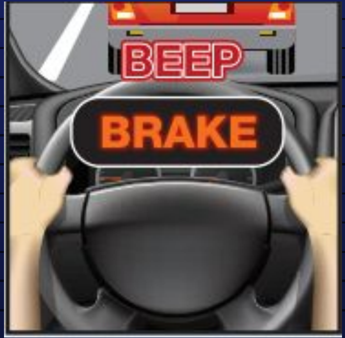
Always-On Scan

The system scans the road for objects and possible hazards.

02

Alert

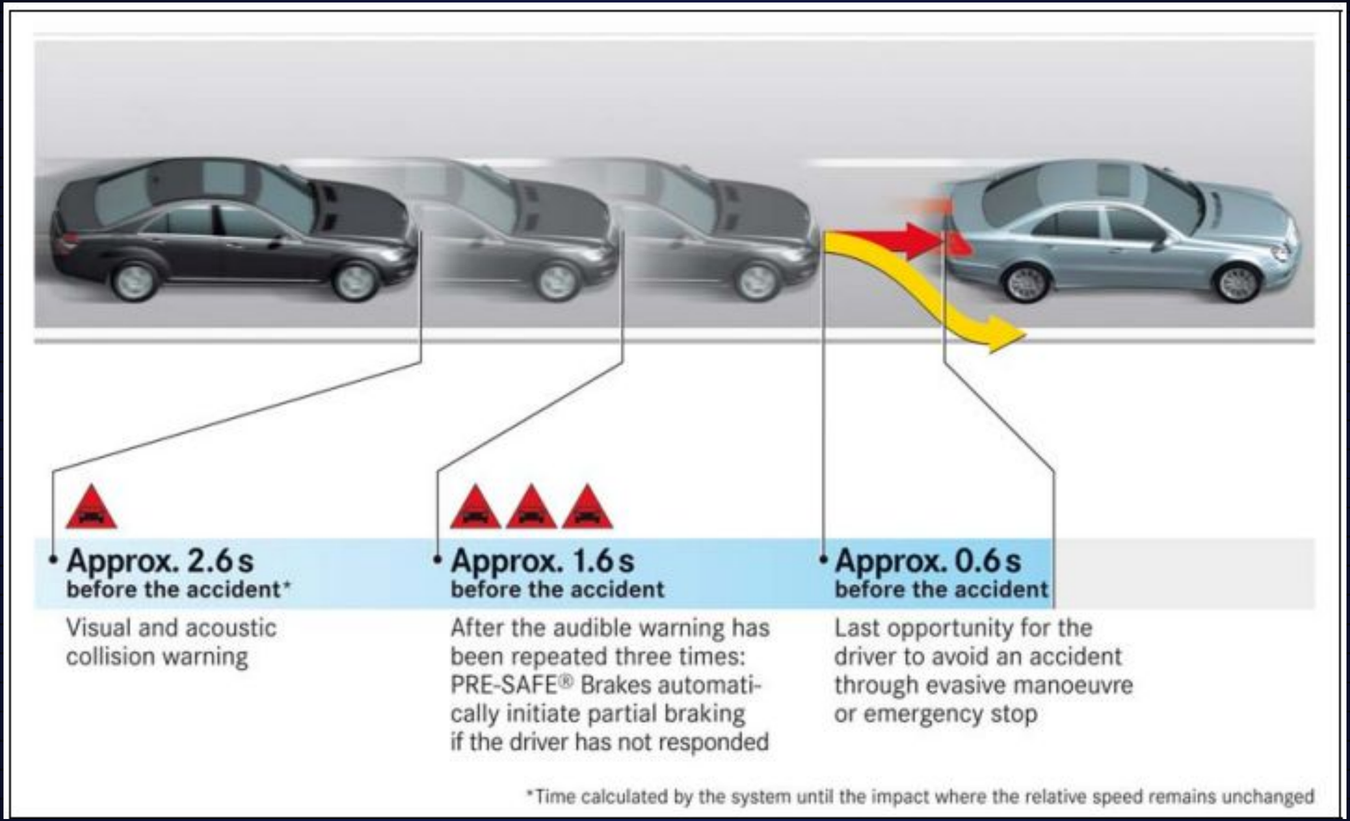
If collision is possible, warning tones and other visual alerts would activate to alert the driver.



03

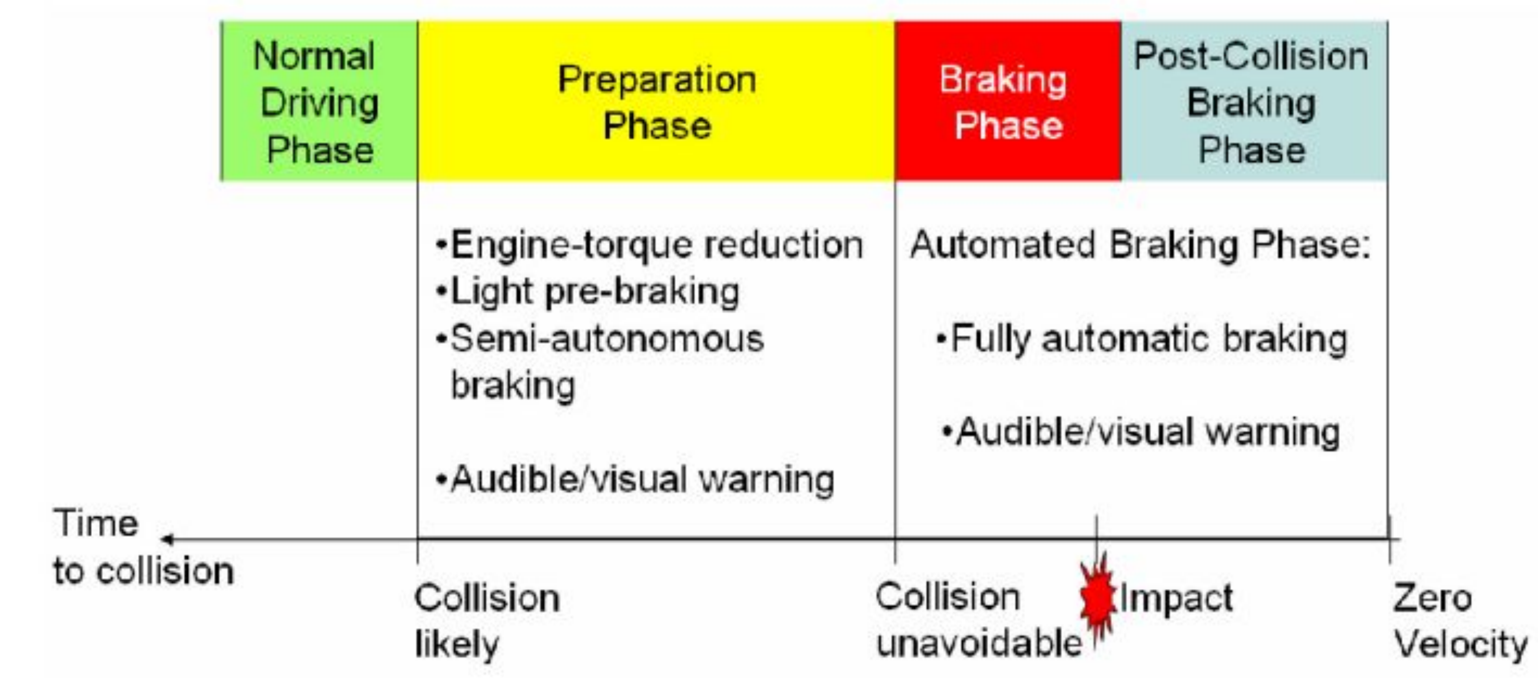
Take Action

If collision is imminent and the driver hasn't responded, the car will automatically brake hard.

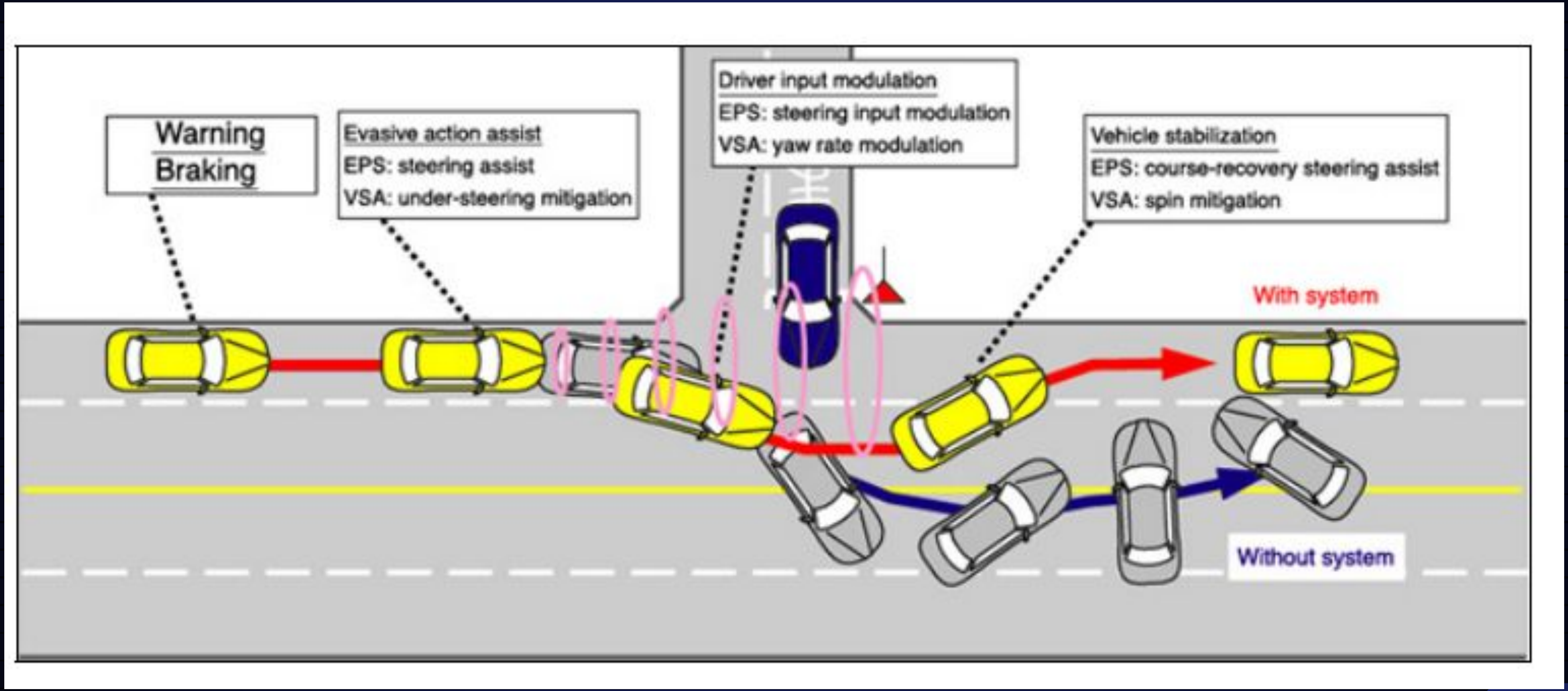


Warnings provided by PRE-SAFE Brake in a typical rear-end collision situation

Collision Mitigation Phases



Forward Obstacle Assistance System



AEB Implementation using Deep Learning

This repository contains code for a project whose goal is to Implement Automatic Emergency Braking System using a monocular camera. This model is trained on tusimple lane dataset.

Repo Link: https://github.com/kuldeepbishnoi/Automatic-Emergency-Braking

- **LaneNet**

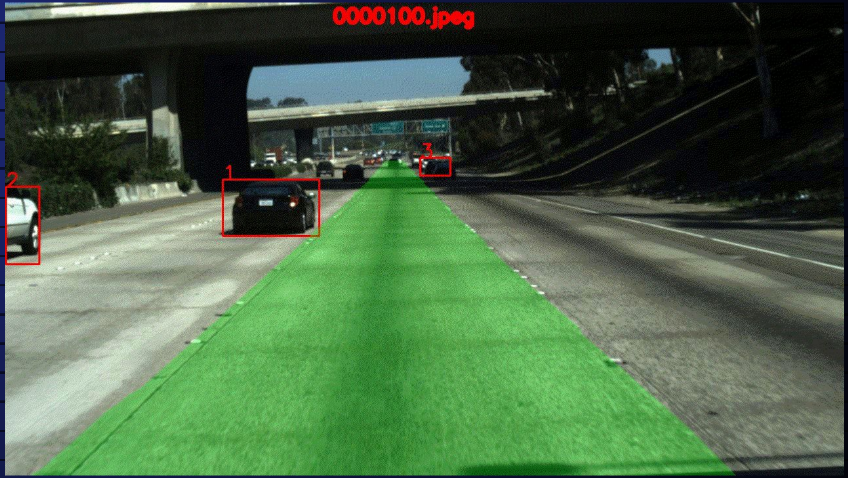
LaneNet algorithm is a state of art deep convolution neural network which is used to detect lanes and is implemented using tensorflow.

- **DeepSORT**

We use YOLO v3 algorithm to perform vehicle detections. Then we take this output feed it to DeepSORT in order to create a highly accurate vehicle tracker.

- **Automatic Emergency Braking System**

This feature can sense incoming(traffic coming to ego lanes) and slow(as well as stopped) traffic ahead and urgently apply the brakes.

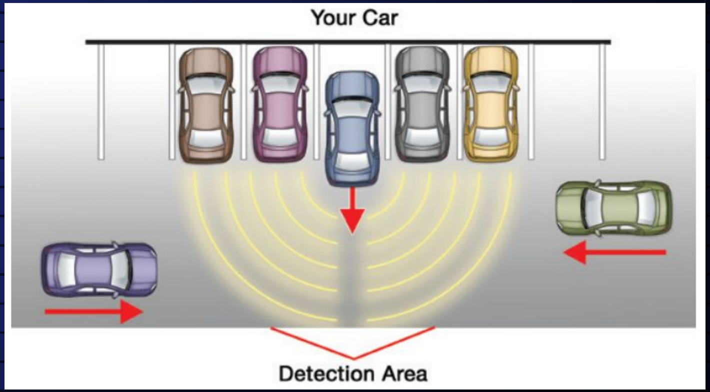


04

Rear Cross Traffic Alert (RCTA)

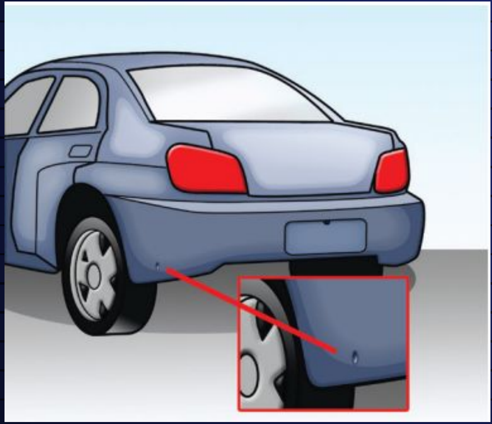
What is it?

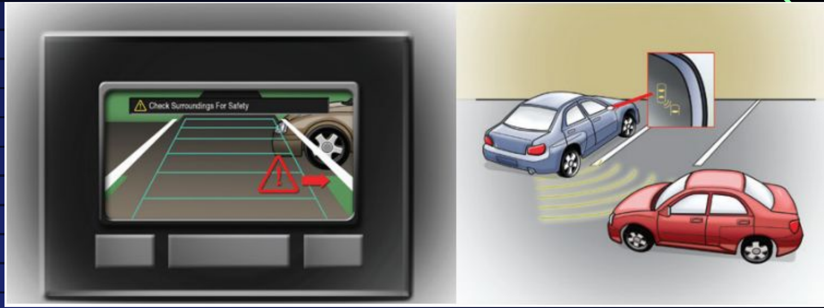
Rear Cross Traffic Alert is designed to help the driver back out of spaces where they may not see approaching traffic, as sometimes happens in parking lots. Rear cross traffic alert monitors two areas behind the driver for vehicles approaching from the right or left.



How does it work?

Sensors are located at each side of the vehicle near the rear bumper and look like buttons. Typically, the rear cross traffic alert will use the same sensors as a blind spot monitoring system.





Warning

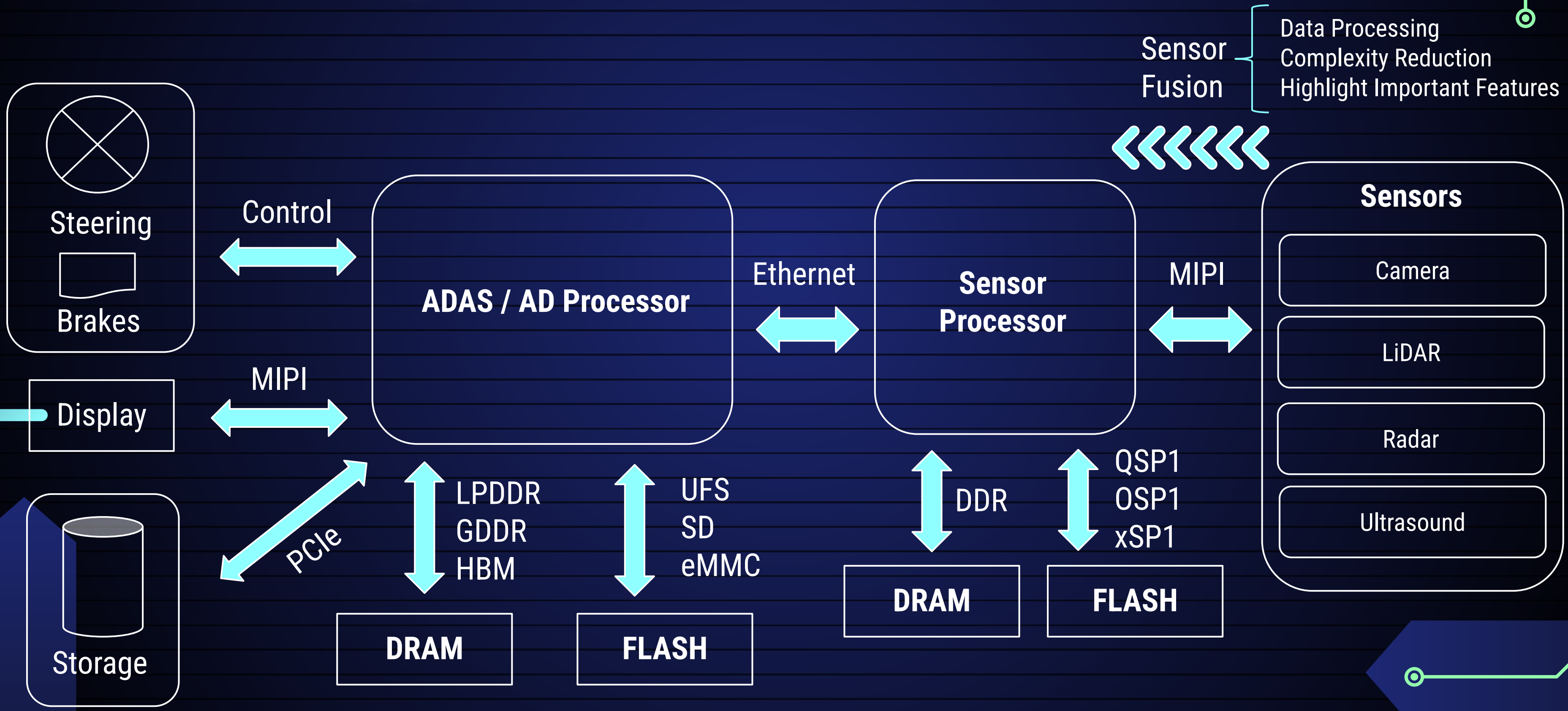
The RCTA system will alert the driver when other vehicles are detected near or in the backing path. Once the rear end of the vehicle is beyond adjacent objects and vehicles, the rear cross traffic alert system will then have a clear view of traffic approaching from the left or right.

Sensor Monitoring

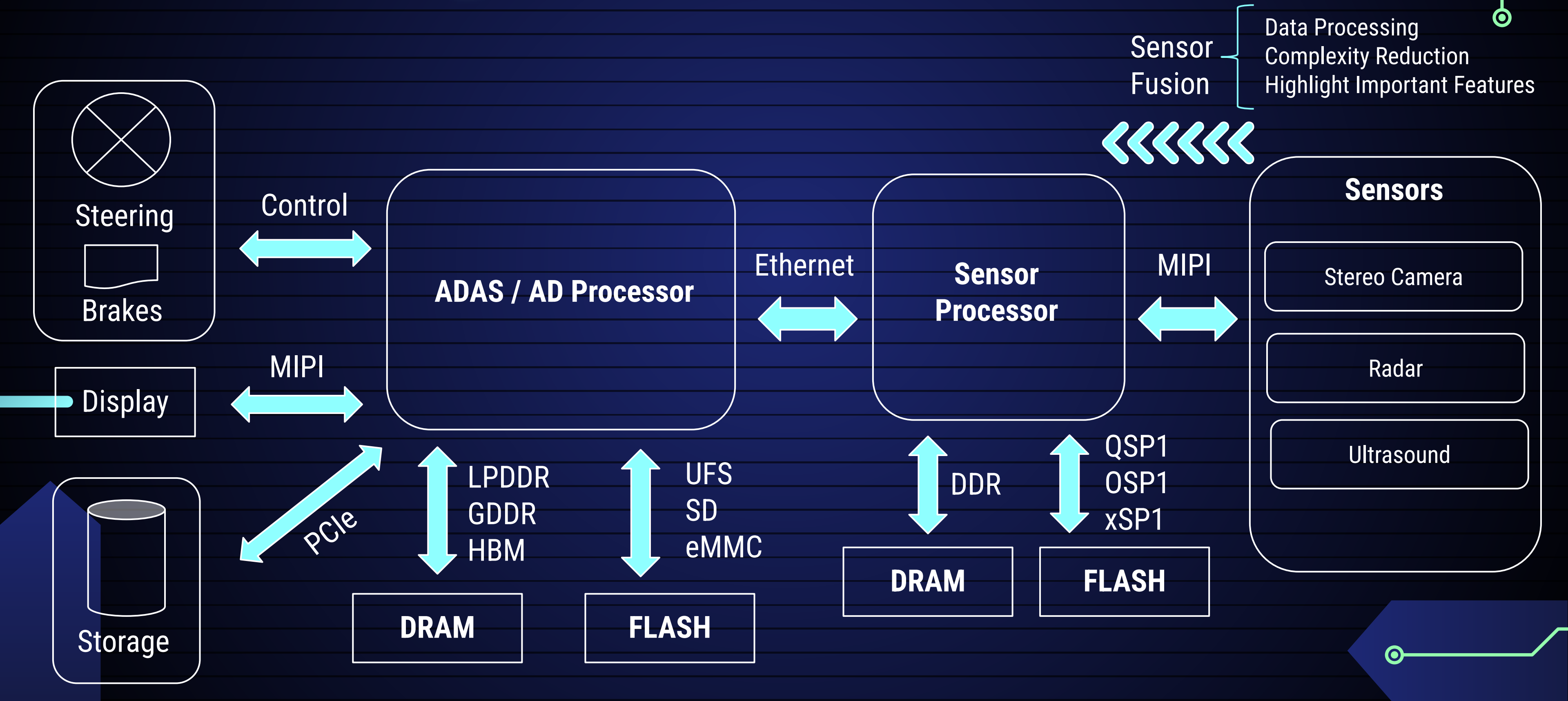
ADAS Sensors & Architecture

How can we implement these systems?

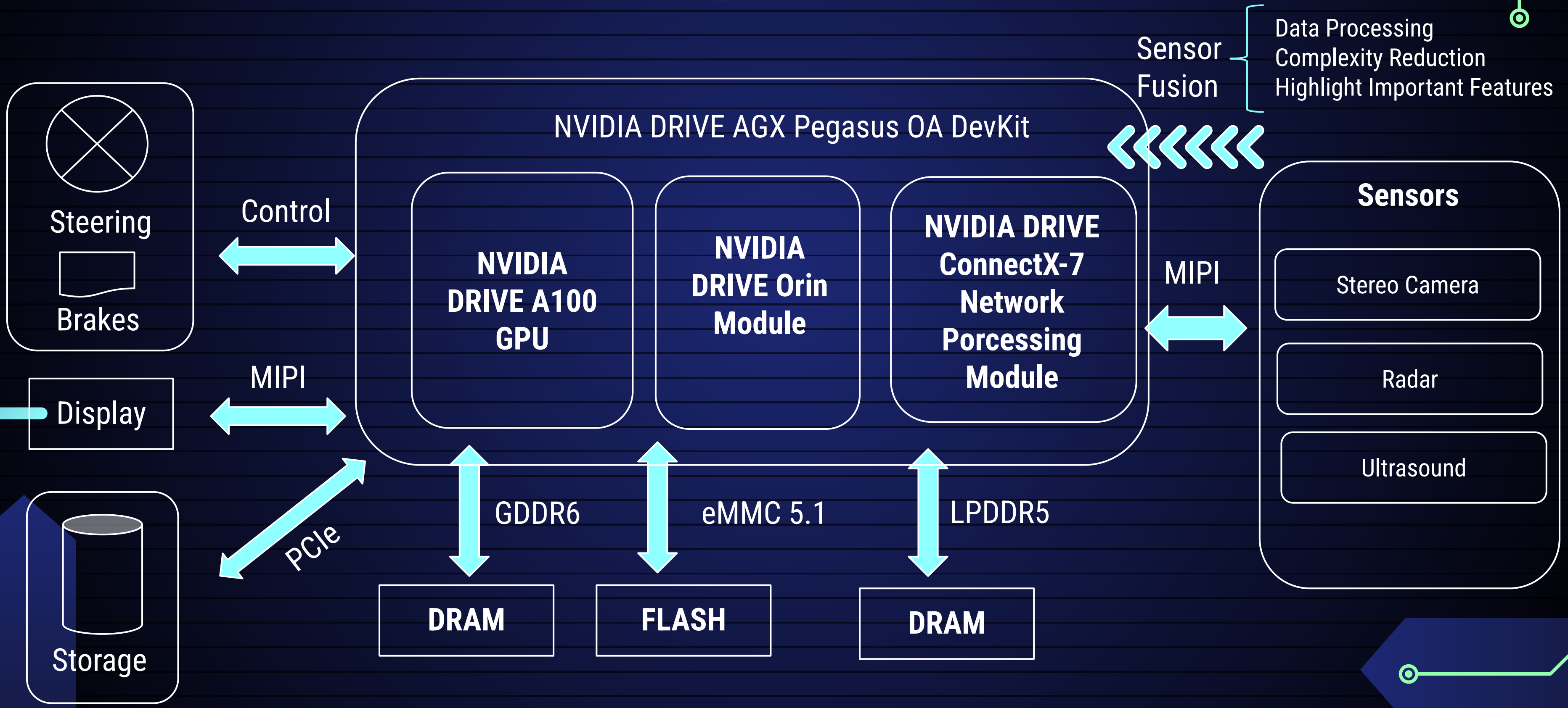
Integrated ADAS Architecture



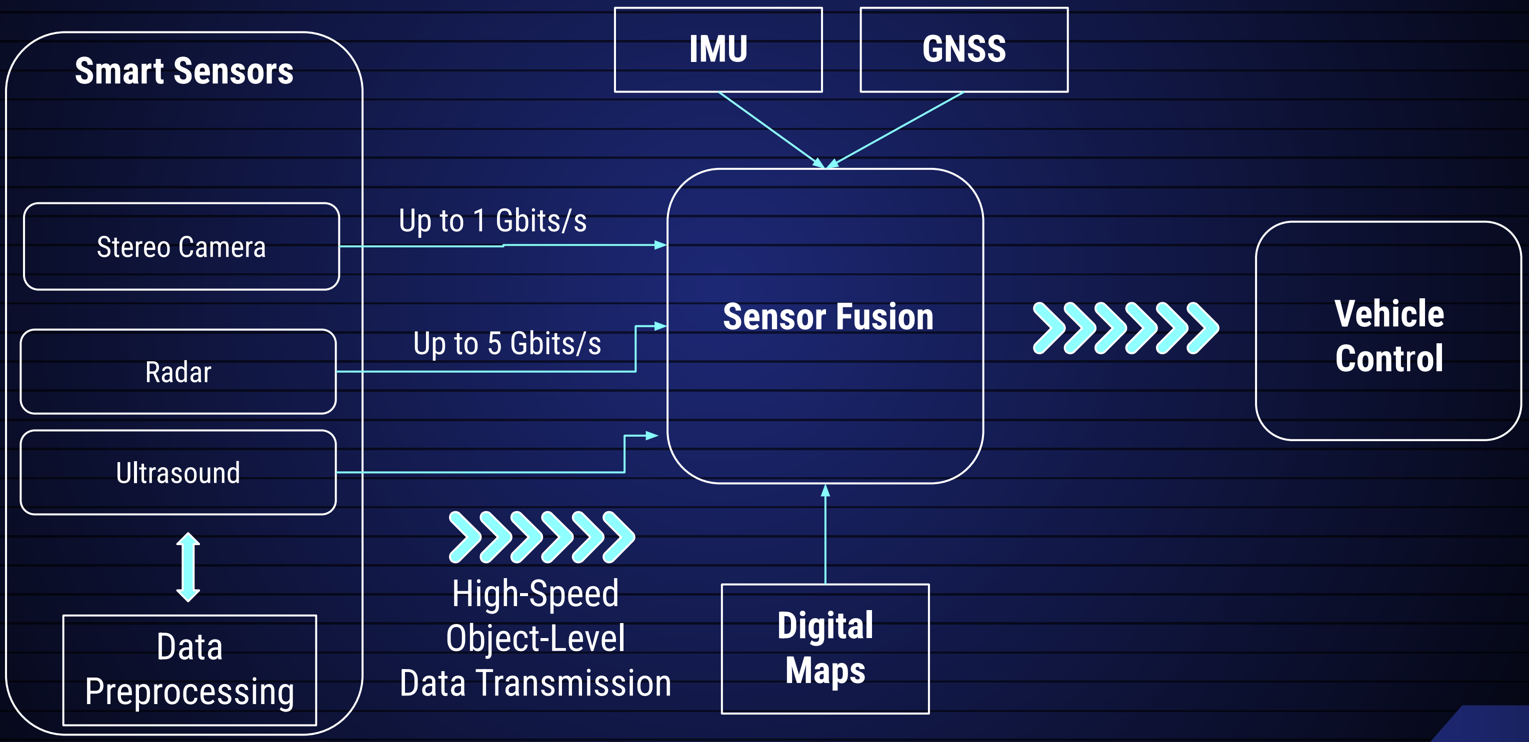
Integrated AV Architecture



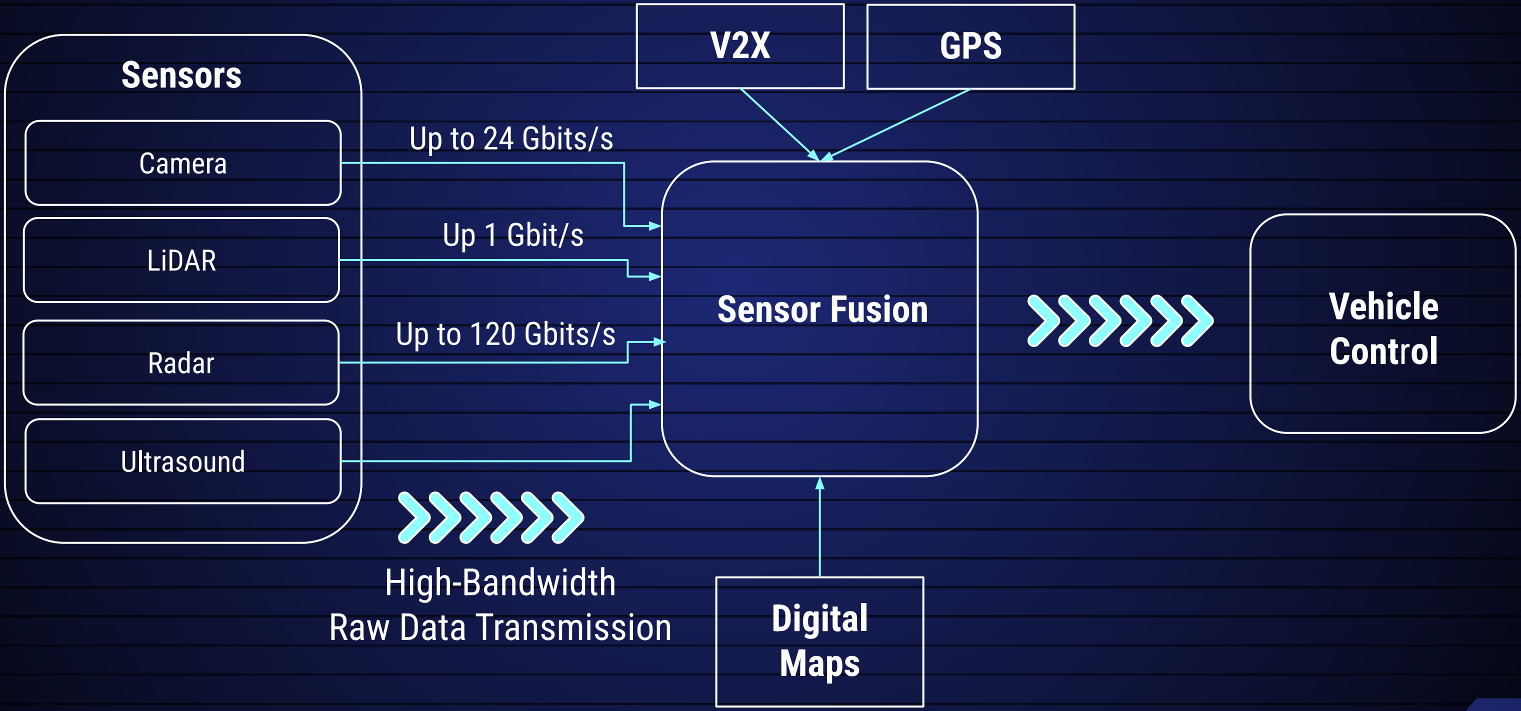
TUNMARS Integrated Architecture



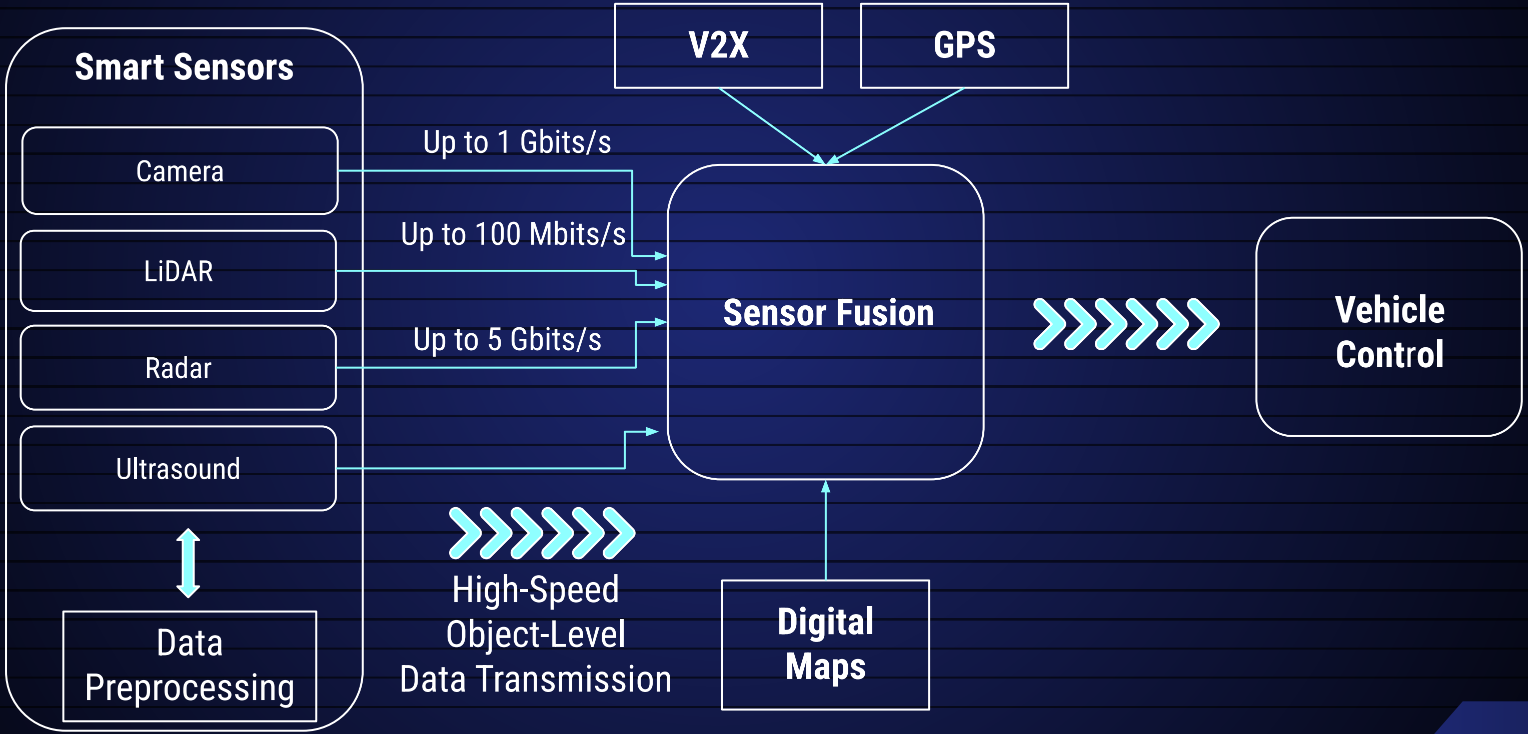
Hybrid Data Sensor Fusion Concept



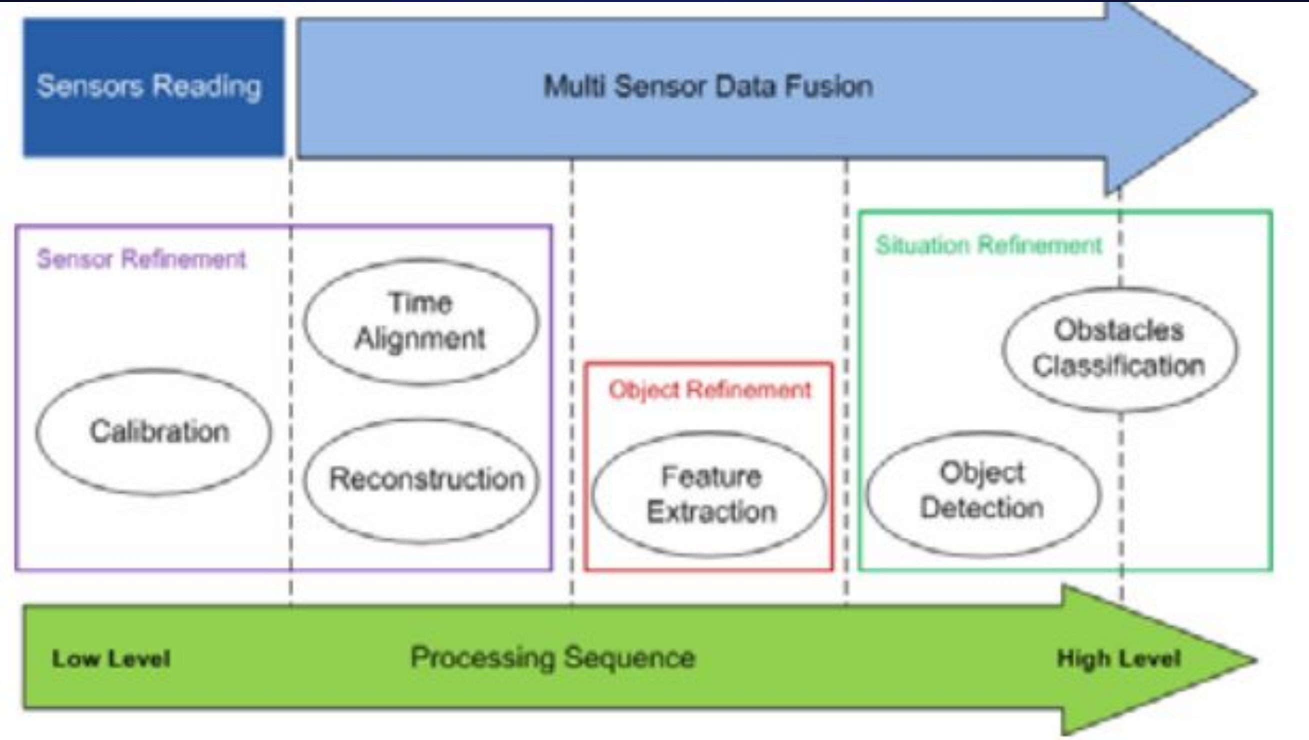
Raw Data Sensor Fusion Concept



Hybrid Data Sensor Fusion Concept



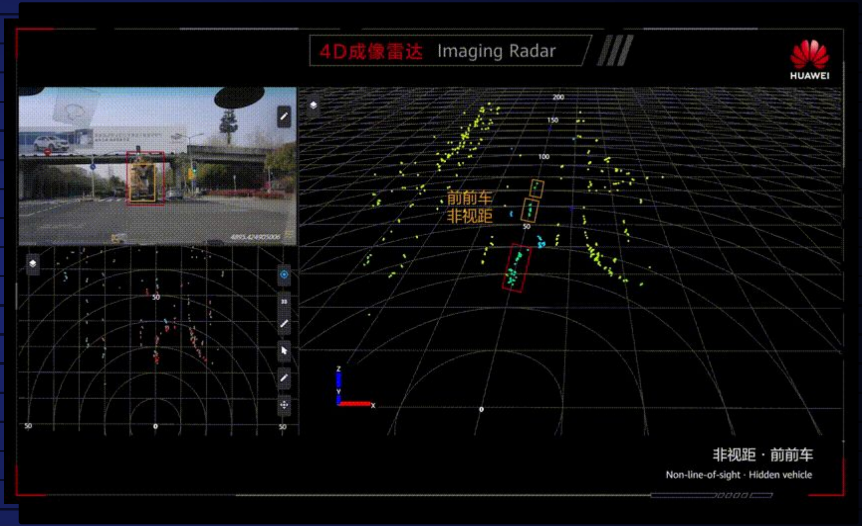
Sensor Fusion Processing



4D Imaging Radars

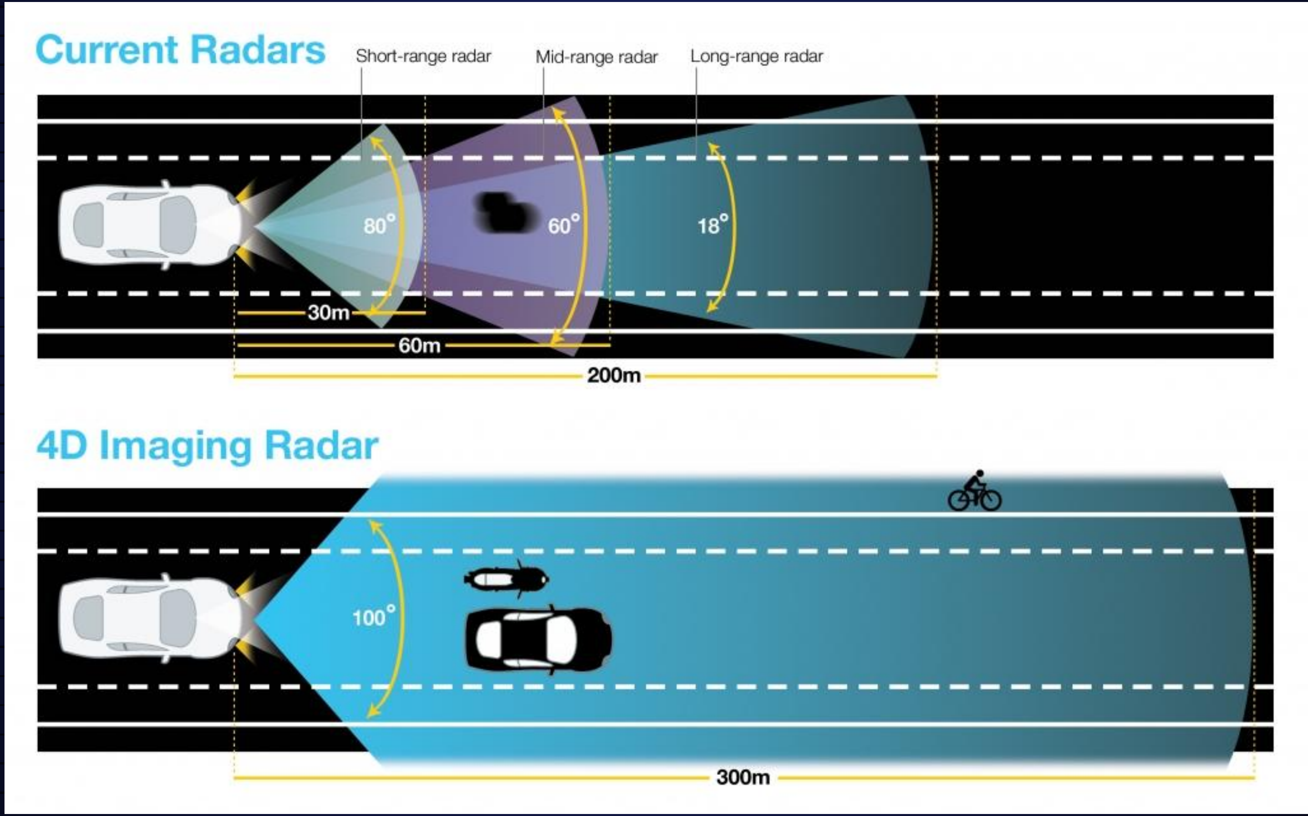
- 4D radar uses a large Multiple Input Multiple Output (MIMO) antenna array for echolocation. It sends signals that bounce off objects in the environment and captures the results to calculate the size, location, direction, speed, and elevation of objects in the environment.

- 4D radar replaces older technologies like cameras, radar, and Lidar that are used for computer vision and autonomous driving in vehicles.



- 4D radar has the advantage of being able to work in any weather and any level of lighting, detect elevation, speed, and direction accurately, and detect targets behind other objects in the environment.

Current Radars VS. 4D Imaging Radars



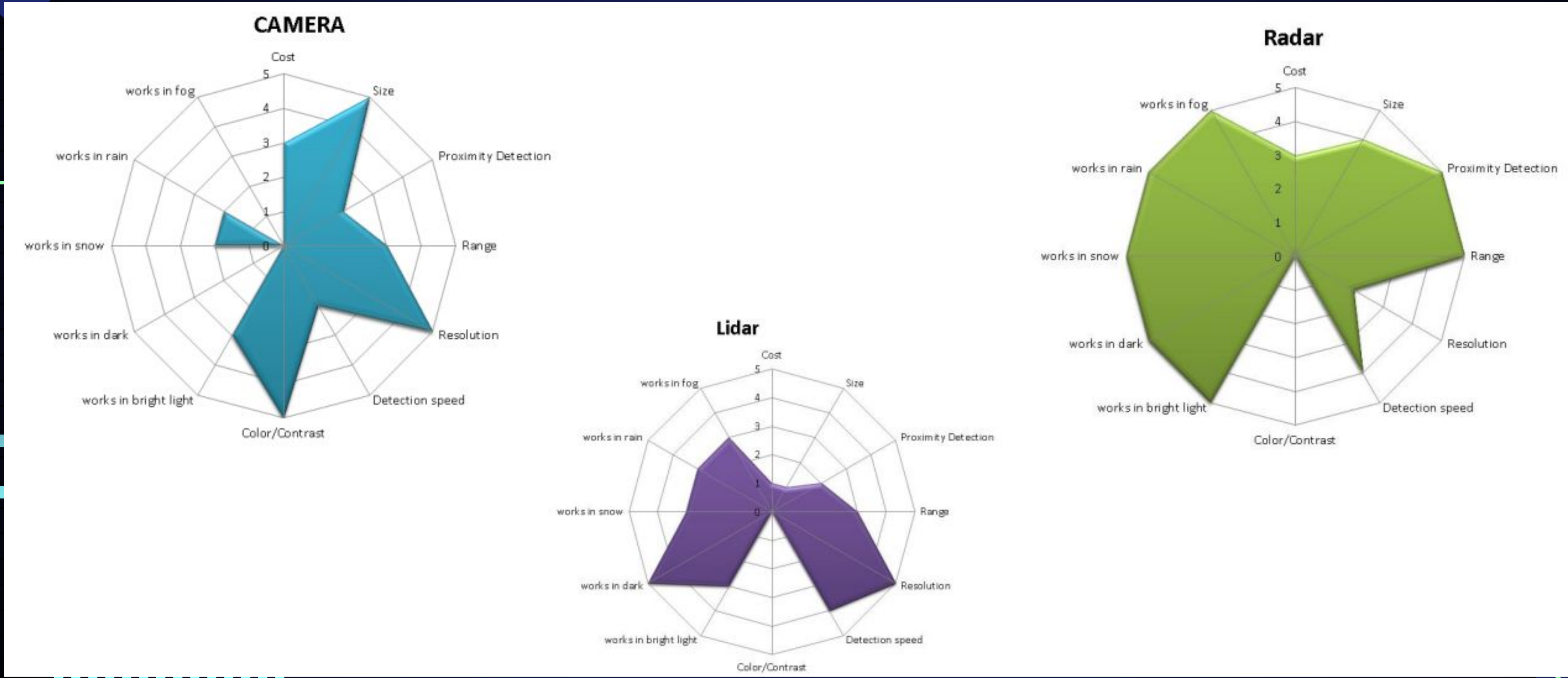
ADAS Sensors Comparison (1)

	Camera	3D Imaging Radars	LiDAR	4D Imaging Radar
Angular Resolution	<=0.1°	1.2°	0.1°	0.6°-1°
Range Accuracy	5cm	10cm	<=5cm	<=5cm
Maximum Range	250m	200m	100m	350m

Sensors Comparison

	Stereo Camera	3D Flash LiDAR	LiDAR	4D Imaging Radar
Angular Resolution	<=0.1°	<1.9°	0.1°	0.6°-1°
Range Accuracy	<50cm	<35cm	<=5cm	<=5cm
Maximum Range	300m	200m	100m	350m
Cost	$200+	$2400+	$1200+	

ADAS Sensors Comparison (2)



THANKS