**Self-driving cars platform**

* HD Maps
* Localization – How the car determines it’s location. Uses laser and radar data & compares what it sees through these sensors to a HD Map to single digit centimeter level accuracy
* Perception – How the car sees the world, fully convolution network. Deep learning **CNN** is used for classification, detection & segmentation etc. Work with data from several different self-driving car sensors including cameras, radar, lidar
* Prediction – predict how other vehicles and pedestrians might move. One approach **RNN**(recurrent neural network) uses movement data over time and uses this time series data to predict
* Planning
* Control – steering, throttle, break to execute **plan**

**How self-driving cars work:**

A self-driving car basically has

Computer vision

Localization Path planning Control

Sensor fusion

* **Computer Vision :** 
  + To figure out what the world around us looks like
  + Use of Deep Neural Nets is incorporated here
  + Like identify other cars around us
* **Sensor Fusion :** 
  + How we incorporate data from other sensors
  + Use of Lidar is incorporated here
  + A lidar(array of lasers) does a 360 degree sweep of the world to map it
* **Localization :** 
  + Once we know what the world around us looks like using the two above we then find where we are in the world
  + GPS are a possible option but are not because their precision is upto 1-2 metres and it is very high to make a car crash into the sidewalk
  + Here we define sophisticated mathematical algorithms for single digit cm level accuracy
* **Path planning :** 
  + Once we have done knowing the world around us and where we are in this world we then define a path to chart through the world to get us to where we’d like to go
  + Define a path for the car to follow, change movement based on predicting other movements like cars, people around us
* **Control :** 
  + How we turn the steering, press throttle, press brakes to move through this path

**Reference vehicle:**

* Base vehicle that can be **controlled electronically**
* This type of vehicle is called **“Drive-By-Wire”** vehicle

**Reference hardware:**

* **Controller Area Network(CAN)** is the vehicle’s internal communication network
* CAN card is how the computer connects to the cars internal network to send **Signals** for acceleration, brake and steering
* **GPS** receives signals from satellites to determine our location
* **Inertial Measurement Unit(IMU)** measures the vehicles movement and position by tracking position, speed, acceleration and other factors
* **LiDAR** is an array of pulse lasers, the LiDAR used in SDCs can scan 360 degrees around the vehicle. The reflection of these laser beams builds the point cloud that software uses to build the environment.
* **Camera** captures the image data, computer vision is used to understand the environment using these images
* **Radar** is used for detecting obstacles. Although Radar has low resolution to determine what object is detected but advantage of Radar is economical, good at measuring speed of other vehicles and works in all environments

A picture containing text, map

Description automatically generated

Fig. SDC hardware reference (Source: researchgate)

**Software stack:**

* **Real Time Operating System(RTOS)** :
  + Example combination of Ubuntu and a Kernel. Remember Ubuntu is just an OS but not real time but adding a separate kernel we make it RTOS
  + RTOS is used to produce timely calculations, analysis and execute corresponding tasks in short time after the car collects data from outside world
* **Runtime Framework:**
  + Example: Optimized version of ROS(Robot OS) is a runtime framework
  + ROS divides autonomous systems into multiple modules based on function
  + Modules are independent and communicate through runtime framework
  + Shared memory reduces the need to copy data if different modules require access(write once read multiple)
  + Decentralization solves problem of single point of failure
  + Different nodes sends messages to each other to communicate
  + The data sent across nodes should be compatible with other nodes else it will cause failure
* **Modules:**

These are independent chunks of code with their own algorithms communicating with each other to produce a single complex working environment

**Cloud service:**

Applications that run on cloud outside of the vehicle

Put electronic data in the cloud and access it from anywhere with internet and authentication

Cloud services can contain simulations, data, security, Over The Air(OTA) updates

**Simulations**  are environment replica of a real world for testing modules and the simulations are optimized using the real world data to improve the execution