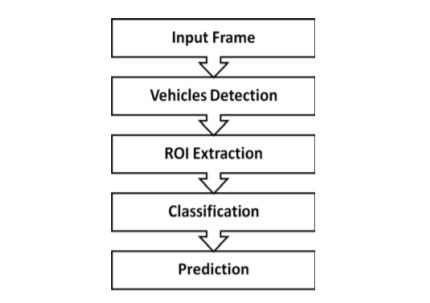
**Project no.8 : Car Parking Classification**

**( Team members : Omkar3.Chavan , Sandeep6.Dubey , Girish.Pillai )**

**1.Architecture:**

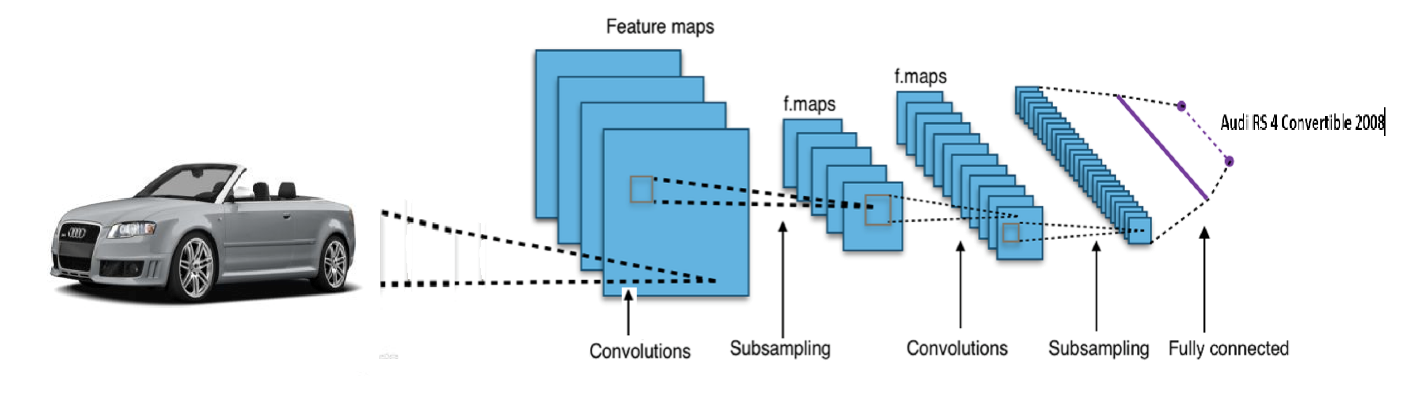
Following is the flow of the project.



**2.Approach of the solution**

1) Car detection :- The image frames collected from the surveillance camera are processed by object detection algorithm and the snapshot of the car is passed further to the image classification module.

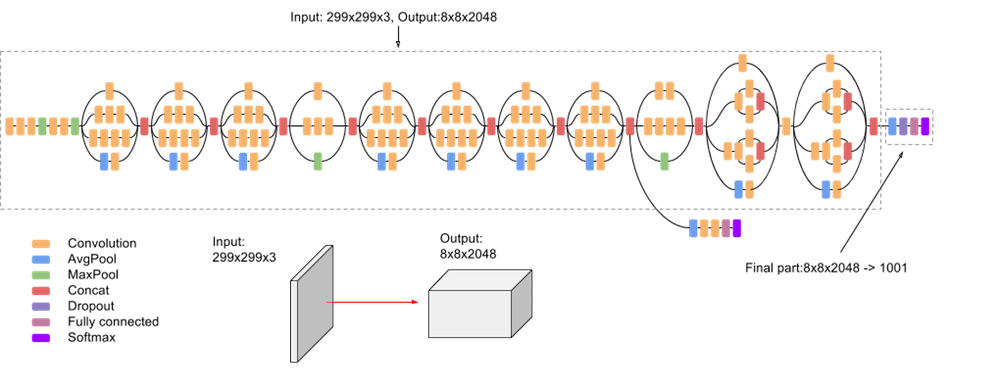
2) Image classification :-



This layer consists of Convolutional Neural Network (CNN) in order to classify the car image into various classes. Following is the architecture of CNN on basis of which we developed are own model.

Inception V3

Inception module uses the concept of adding multiple layers before pooling layer in the convolutional layer and concatenates the output of the whole process to pass it to the full connected neural network layer.



**3.Reason behind using various Opensource libraries/ Frameworks**

Why did we use tenserflow ?

TensorFlow can be used to build deep learning algorithms. We can build deep learning neural networks from TensorFlow with more than 1 hidden layer. We can use this for image recognition, object recognition and various classification problems.

TensorFlow is the bigger library to perform various operations both for normal and deep Neural networks.

TensorFlow provides multiple APIs.

Tensorflow has a very strong community support and a very well written documentation

**4.Instructions to deploy and run the code**

>>>>Make sure you have python 3.x and TensorFlow installed.

>>>>open the zip and run the below 2 file one after the other

**1) Run *retrain.py***

Open retrain.py in your chosen Python editor and give it a quick skim.

Depending on your computer’s power, the training process will probably take 5-8 minutes.

**2) Run *test.py***

In open test.py in your chosen Python editor. Verify the file and directory paths at the top are correct for your computer, then run the script. The script will open each image in TEST\_IMAGE\_DIR in an OpenCV window and show the classification results on standard out.

**3)** After running the script go to the folder containing the code, a csv file will be created named **dict.csv** which contains the model name and its count.