

AWS Recognition : Vehicle Detection

Abstract

Amazon Web services has become expert in cloud computing. AWS offers many different tools and solutions for enterprises and software developers that can be used in data centers in up to 190 countries. Groups such as government agencies, education institutions, nonprofits and private organizations can use AWS services.

This part of project uses AWS Rekognition to detect Vehicles in the images. This part is further exteded to detect Vehicles in Live videos thus give and on-demand run time processing and detection.



Group Members

Name	Roll	Email	Project Guide
Aradhita Menghal	04	menghalar@rkneec.edu	
Urvi Negi	16	negiu@rkneec.edu	
Akshat Chandak	35	chandaka@rkneec.edu	Dr. M. B. Chandak
Shreyas Nemani	77	nemanism@rkneec.edu	
Shubham Jha	78	jhasm@rkneec.edu	

Importing Important Libraries

For AWS

- **boto3** : Boto3 is the Amazon Web Services (AWS) Software Development Kit (SDK) for Python, which facilitate in writing software that makes use of services like Amazon S3 and Amazon EC2. It allows directly create, update, and delete AWS resources from Python scripts.

In [1]:

```
import boto3
```

For Image Processing and Display

- **PIL** : Python Imaging Library is a free and open-source additional library for the Python programming language that adds support for opening, manipulating, and saving many different image file formats.
- **IPython** : IPython is a command shell for interactive computing in multiple programming languages, originally developed for the Python programming language, that offers introspection, rich media, shell syntax, tab completion, and history.

In [2]:

```
from PIL import Image, ImageDraw
from IPython.display import display
```

For Utilities

- **io** : Python io module allows us to manage the file-related input and output operations.
- **pprint** : The pprint module provides a capability to “pretty-print” arbitrary Python data structures in a form which can be used as input to the interpreter. If the formatted structures include objects which are not fundamental Python types, the representation may not be loadable.

In [5]:

```
import io
from pprint import pprint
```

Setting Environment and Credentials for AWS (Configuration)

Part 1

1. Signing in to Management console in AWS ...
2. Opening IAM Console ...
3. Creating a user ...
4. Assigning policies to the user `AWSLambdaExecute` and `AmazonRekognitionFullAccess` ...

Part 2

Obtaining User Credentials keys from management access keys

Note : For security reasons, Keys are set to XXXXXXXXXXXX

In [6]:

```
key = "AKIARFVQGHVXB7KVJN2"
secret_key = "dN3mCQAEuTA06nfhnldmpxffKyf4v+p6610ob607"
```

Part 3 :

Setting up region.

Let us take us-west-2 at Oregon, United States



In [8]:

```
region = 'us-west-2'
```

Creating API Caller

To call AWS API's on our behalf, we make use of Abstracted API.

We want to connect to low-level client interface. Therefore we will use `boto3.client()`

Service(s) to call : `rekognition`

In [11]:

```
client = boto3.client('rekognition',
                      aws_access_key_id=key,
                      aws_secret_access_key=secret_key,
                      region_name=region)
```

Setting up and reading Image

In [56]:

```
image1 = 'tests/sample1.jpg'

with open(image1, 'rb') as source_image:
    source_bytes=source_image.read()

image = Image.open(image1)
display(image)
```



Calling API and Storing Response

In [34]:

```
response= client.detect_labels(Image={'Bytes':source_bytes},MaxLabels=10,MinConfidence=95,)
```

In [35]:

```
pprint(response)
```

```
{'LabelModelVersion': '2.0',
 'Labels': [{('Confidence': 99.80648803710938,
    'Instances': [{('BoundingBox': {'Height': 0.40231195092201233,
        'Left': 0.1429692506790161,
        'Top': 0.36501234769821167,
        'Width': 0.31121718883514404},
       'Confidence': 99.80648803710938},
      {'BoundingBox': {'Height': 0.43399977684020996,
        'Left': 0.5247434973716736,
        'Top': 0.3640315532684326,
        'Width': 0.3463401198387146},
       'Confidence': 99.69676971435547},
      {'BoundingBox': {'Height': 0.0950193703174591,
        'Left': 0.34746429324150085,
        'Top': 0.2663646340370178,
        'Width': 0.09636733680963516},
       'Confidence': 99.63018035888672},
      {'BoundingBox': {'Height': 0.22876952588558197,
        'Left': 0.7084001302719116,
        'Top': 0.28796088695526123,
        'Width': 0.2678349018096924},
       'Confidence': 99.60955810546875},
      {'BoundingBox': {'Height': 0.2196170538663864,
        'Left': 0.017233550548553467,
        'Top': 0.295086532831192,
        'Width': 0.25215578079223633},
       'Confidence': 99.58589172363281},
      {'BoundingBox': {'Height': 0.12044242024421692,
        'Left': 0.5149036049842834,
        'Top': 0.24897067248821259,
        'Width': 0.09709720313549042},
       'Confidence': 99.28630828857422},
      {'BoundingBox': {'Height': 0.06947677582502365,
        'Left': 0.6187900304794312,
        'Top': 0.2510042190551758,
        'Width': 0.10943432152271271},
       'Confidence': 99.18661499023438},
      {'BoundingBox': {'Height': 0.08283356577157974,
        'Left': 0.8217981457710266,
        'Top': 0.24189375340938568,
        'Width': 0.13719157874584198},
       'Confidence': 98.6993408203125},
      {'BoundingBox': {'Height': 0.07161980122327805,
        'Left': 0.2460605353116989,
        'Top': 0.2534158229827881,
        'Width': 0.09537138789892197},
       'Confidence': 98.09568786621094},
      {'BoundingBox': {'Height': 0.07633476704359055,
        'Left': 0.02131439372897148,
        'Top': 0.24749447405338287,
        'Width': 0.14701640605926514},
       'Confidence': 96.96759033203125}],
 'Name': 'Car',
 'Parents': [{('Name': 'Vehicle'), ('Name': 'Transportation')}],
 'Confidence': 99.80648803710938,
 'Instances': [],
 'Name': 'Transportation',
```

```
'Parents': [],
{'Confidence': 99.80648803710938,
'Instances': [],
'Name': 'Vehicle',
'Parents': [{'Name': 'Transportation'}]},
{'Confidence': 99.80648803710938,
'Instances': [],
'Name': 'Automobile',
'Parents': [{'Name': 'Vehicle'}, {'Name': 'Transportation'}]},
{'Confidence': 98.87216186523438,
'Instances': [],
'Name': 'Sports Car',
'Parents': [{'Name': 'Car'},
{'Name': 'Vehicle'},
{'Name': 'Transportation'}]},
{'Confidence': 96.90312194824219,
'Instances': [],
'Name': 'Coupe',
'Parents': [{'Name': 'Sports Car'},
{'Name': 'Car'},
{'Name': 'Vehicle'},
{'Name': 'Transportation'}]},
{'Confidence': 96.23088073730469,
'Instances': [],
'Name': 'Tire',
'Parents': []},
{'Confidence': 96.08903503417969,
'Instances': [],
'Name': 'Spoke',
'Parents': [{'Name': 'Machine'}]},
{'Confidence': 96.08903503417969,
'Instances': [],
'Name': 'Machine',
'Parents': []},
{'Confidence': 95.27862548828125,
'Instances': [{"BoundingBox": {"Height": 0.03222047537565231,
'Left': 0.3257557153701782,
'Top': 0.2874937653541565,
'Width': 0.014418458566069603},
'Confidence': 95.27862548828125},
{'BoundingBox': {"Height": 0.08022399991750717,
'Left': 0.834028422832489,
'Top': 0.4265576899051666,
'Width': 0.045549772679805756},
'Confidence': 88.8863754272461},
{'BoundingBox': {"Height": 0.07384871691465378,
'Left': 0.12992331385612488,
'Top': 0.42881715297698975,
'Width': 0.03561103343963623},
'Confidence': 79.31412506103516},
{'BoundingBox': {"Height": 0.1503514051437378,
'Left': 0.36699336767196655,
'Top': 0.5906511545181274,
'Width': 0.04326023906469345},
'Confidence': 76.84893798828125},
{'BoundingBox': {"Height": 0.05044828727841377,
'Left': 0.23733735084533691,
'Top': 0.38309040665626526,
'Width': 0.02053551748394966},
'Confidence': 76.37004852294922},
{'BoundingBox': {"Height": 0.04186691716313362,
```

```
        'Left': 0.7241848707199097,
        'Top': 0.3832460045814514,
        'Width': 0.024964427575469017},
    'Confidence': 74.71394348144531},
    {'BoundingBox': {'Height': 0.09900222718715668,
                    'Left': 0.528217077255249,
                    'Top': 0.47771865129470825,
                    'Width': 0.019963646307587624},
     'Confidence': 70.58234405517578},
    {'BoundingBox': {'Height': 0.0336938239634037,
                    'Left': 0.6227752566337585,
                    'Top': 0.27907395362854004,
                    'Width': 0.016481494531035423},
     'Confidence': 59.97272872924805},
    {'BoundingBox': {'Height': 0.03389355540275574,
                    'Left': 0.9054529070854187,
                    'Top': 0.28893858194351196,
                    'Width': 0.019908523187041283},
     'Confidence': 58.91135787963867},
    {'BoundingBox': {'Height': 0.034510307013988495,
                    'Left': 0.059760548174381256,
                    'Top': 0.29119551181793213,
                    'Width': 0.021453510969877243},
     'Confidence': 56.00244140625},
    {'BoundingBox': {'Height': 0.031820040196180344,
                    'Left': 0.6686364412307739,
                    'Top': 0.28913164138793945,
                    'Width': 0.016746997833251953},
     'Confidence': 51.50978469848633}],
    'Name': 'Wheel',
    'Parents': [{"Name": "Machine"}]}],
'ResponseMetadata': {'HTTPHeaders': {'connection': 'keep-alive',
                                      'content-length': '4419',
                                      'content-type': 'application/x-amz-json-1.1',
                                      'date': 'Wed, 17 Mar 2021 10:24:38 GMT',
                                      'x-amzn-requestid': 'd326c8d5-4c39-4d1f-8da7-9e9aa0334f6f'},
                         'HTTPStatusCode': 200,
                         'RequestId': 'd326c8d5-4c39-4d1f-8da7-9e9aa0334f6f',
                         'RetryAttempts': 0}}
```

In [49]:

```
Label_list = response["Labels"]

for i in Label_list:
    if(i["Name"]=="Car"):
        instance_list=i["Instances"]

pprint(instance_list)

[{'BoundingBox': {'Height': 0.40231195092201233,
                 'Left': 0.1429692506790161,
                 'Top': 0.36501234769821167,
                 'Width': 0.31121718883514404},
 'Confidence': 99.80648803710938},
 {'BoundingBox': {'Height': 0.43399977684020996,
                 'Left': 0.5247434973716736,
                 'Top': 0.3640315532684326,
                 'Width': 0.3463401198387146},
 'Confidence': 99.69676971435547},
 {'BoundingBox': {'Height': 0.0950193703174591,
                 'Left': 0.34746429324150085,
                 'Top': 0.2663646340370178,
                 'Width': 0.09636733680963516},
 'Confidence': 99.63018035888672},
 {'BoundingBox': {'Height': 0.22876952588558197,
                 'Left': 0.7084001302719116,
                 'Top': 0.28796088695526123,
                 'Width': 0.2678349018096924},
 'Confidence': 99.60955810546875},
 {'BoundingBox': {'Height': 0.2196170538663864,
                 'Left': 0.017233550548553467,
                 'Top': 0.295086532831192,
                 'Width': 0.25215578079223633},
 'Confidence': 99.58589172363281},
 {'BoundingBox': {'Height': 0.12044242024421692,
                 'Left': 0.5149036049842834,
                 'Top': 0.24897067248821259,
                 'Width': 0.09709720313549042},
 'Confidence': 99.28630828857422},
 {'BoundingBox': {'Height': 0.06947677582502365,
                 'Left': 0.6187900304794312,
                 'Top': 0.2510042190551758,
                 'Width': 0.10943432152271271},
 'Confidence': 99.18661499023438},
 {'BoundingBox': {'Height': 0.08283356577157974,
                 'Left': 0.8217981457710266,
                 'Top': 0.24189375340938568,
                 'Width': 0.13719157874584198},
 'Confidence': 98.6993408203125},
 {'BoundingBox': {'Height': 0.07161980122327805,
                 'Left': 0.2460605353116989,
                 'Top': 0.2534158229827881,
                 'Width': 0.09537138789892197},
 'Confidence': 98.09568786621094},
 {'BoundingBox': {'Height': 0.07633476704359055,
                 'Left': 0.02131439372897148,
                 'Top': 0.24749447405338287,
                 'Width': 0.14701640605926514},
 'Confidence': 96.96759033203125}]
```

In [40]:

```
print(len(instance_list))
```

10

Inference

1. It has detected 10 cars in the image.
 2. Bounding box returns the co-ordinates which we will use to plot images
 3. Confidence indicates the confidence levels
 4. We only extracted the car object. We can extract other objects as well.
-

Plotting the Images

In [48]:

```
image=Image.open(image1)

imgWidth, imgHeight = image.size
draw = ImageDraw.Draw(image)

#Bounding Boxes

for i in instance_list:

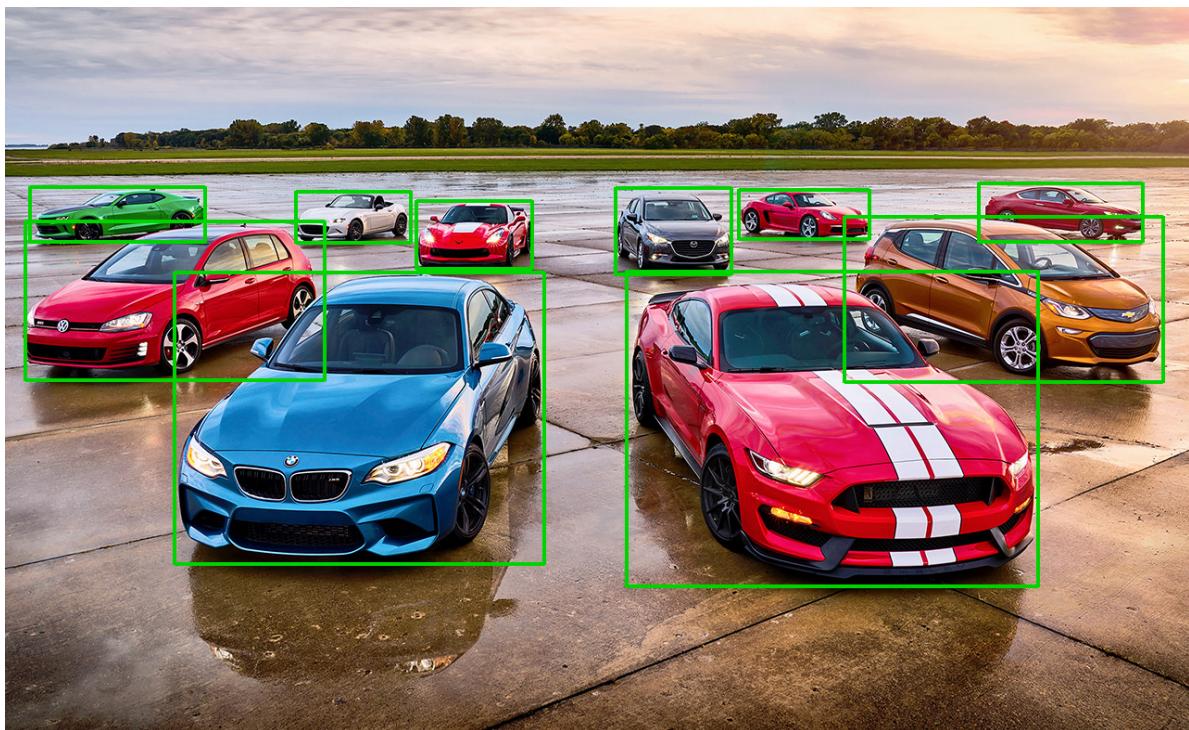
    ele=i["BoundingBox"]
    h=ele["Height"]
    l=ele["Left"]
    t=ele["Top"]
    w=ele["Width"]

    left = imgWidth * l
    top = imgHeight * t
    width = imgWidth * w
    height = imgHeight *h

    points = (
        (left,top),
        (left + width, top),
        (left + width, top + height),
        (left , top + height),
        (left, top)
    )

    draw.line(points, fill="#00d400", width=4)

display(image)
```



In [135]:

```
def detect(imx):
    with open(imx, 'rb') as source_image:
        source_bytes=source_image.read()

    response= client.detect_labels(Image={'Bytes':source_bytes},MaxLabels=10,MinConfidence=0.5)
    Label_list = response["Labels"]

    for i in Label_list:
        if(i["Name"]=="Car"):
            instance_list=i["Instances"]

    print(len(instance_list))

    img =Image.open(imx)

imgWidth, imgHeight = img.size
draw = ImageDraw.Draw(img)

#Bounding Boxes

for i in instance_list:

    ele=i["BoundingBox"]
    h=ele["Height"]
    l=ele["Left"]
    t=ele["Top"]
    w=ele["Width"]

    left = imgWidth * l
    top = imgHeight * t
    width = imgWidth * w
    height = imgHeight *h

    points = (
        (left,top),
        (left + width, top),
        (left + width, top + height),
        (left , top + height),
        (left, top)
    )

    draw.line(points, fill="#00d400", width=4)

display(img)

#change this name everytime

path = 'results/result9.jpg'
img.save(path)
```

Test 1 : Single Car

In [78]:

```
image2 = 'tests/sample2.jpg'  
detect(image2)
```

1

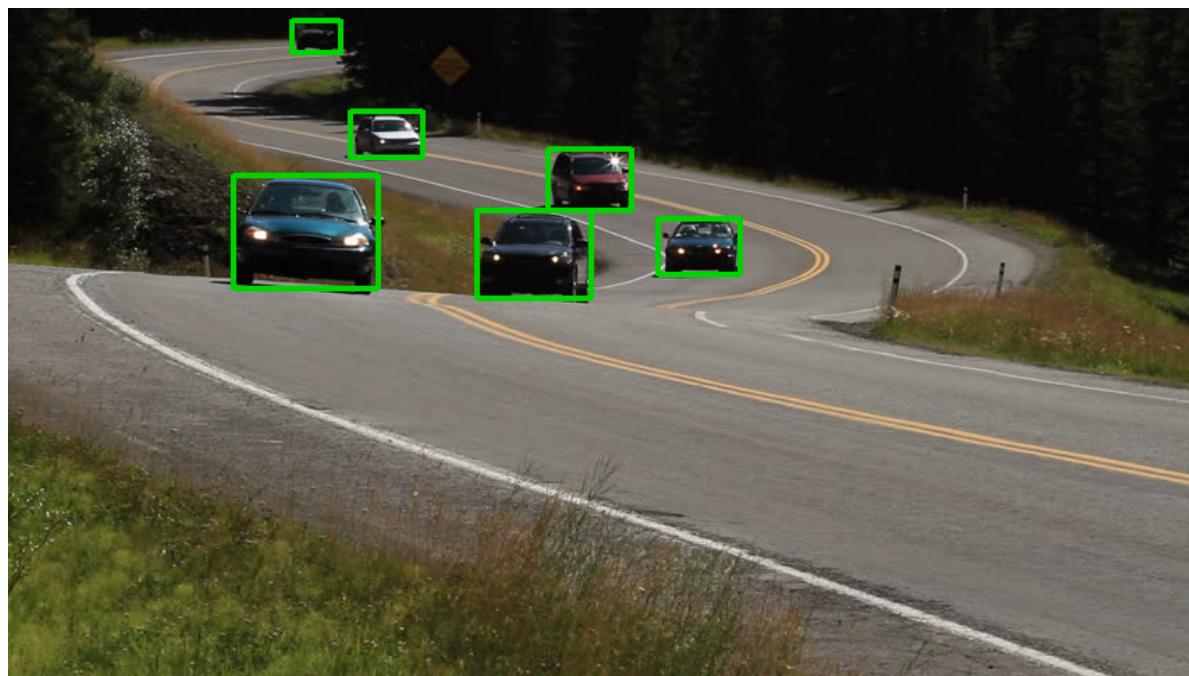


Test 2 : Multiple Cars

In [79]:

```
image3 = 'tests/sample3.jpg'  
detect(image3)
```

6

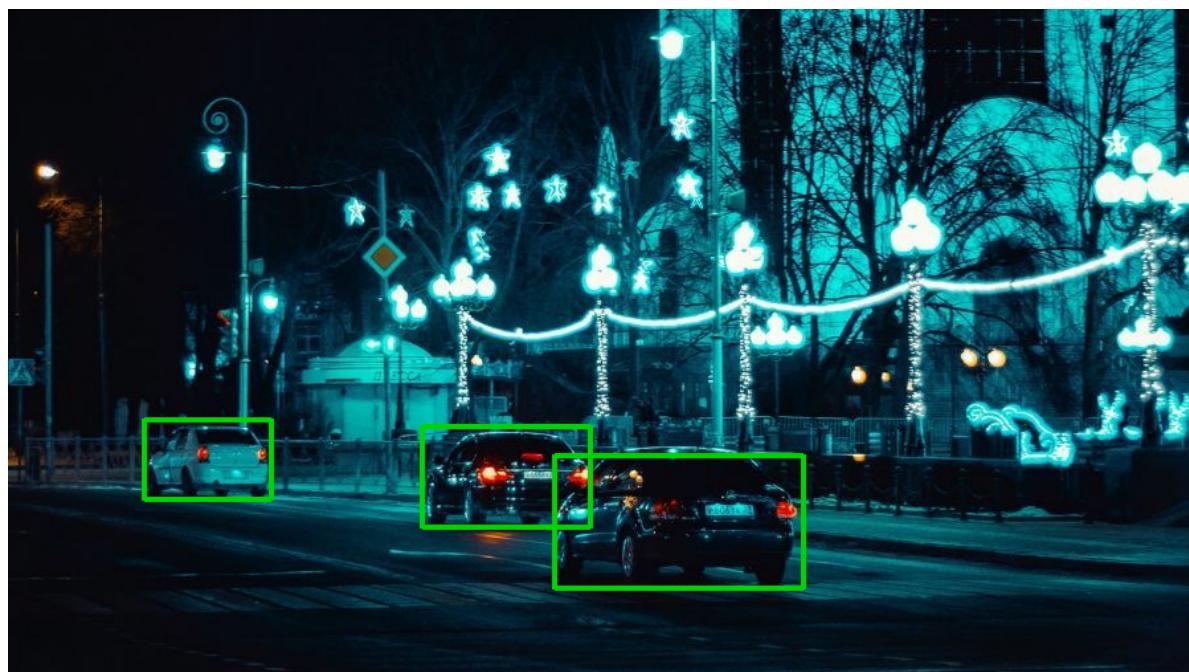


Test 3 : Night View

In [84]:

```
image4 = 'tests/sample4.jpg'  
detect(image4)
```

3

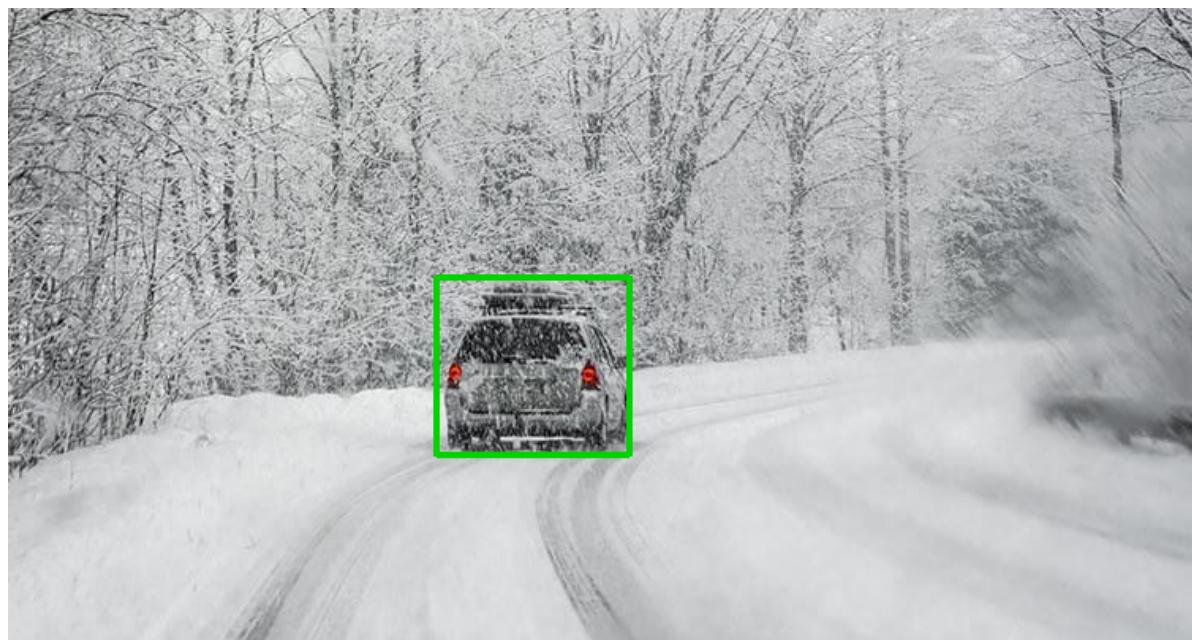


Test 5 : Snowfalls of Eupore

In [88]:

```
image5 = 'tests/sample5.jpg'  
detect(image5)
```

1

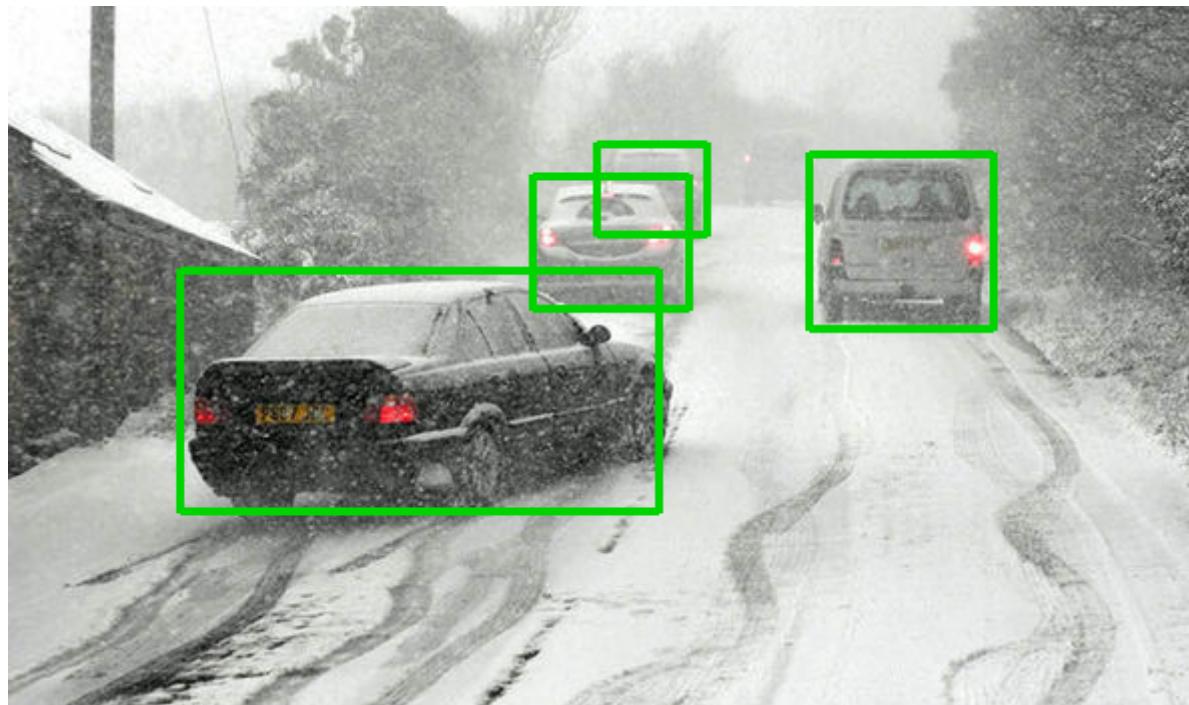


Test 6 : Chilling Winters of Canada

In [96]:

```
image6 = 'tests/sample6.jpg'  
detect(image6)
```

4

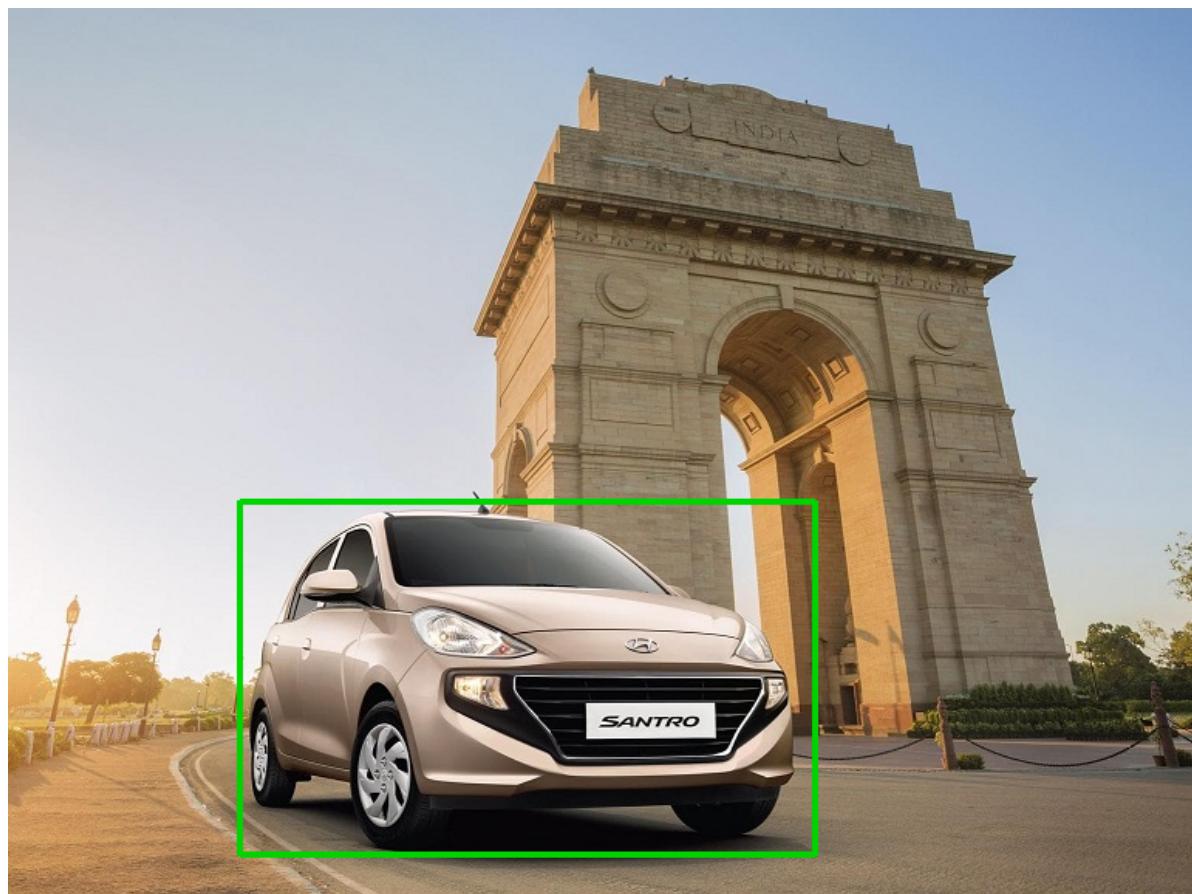


Test 7 : We heard Tesla comin' to India. Well.. So do we!

In [102]:

```
image7 = 'tests/sample7.jpg'  
detect(image7)
```

1



Test 8 : National Highways of India

In [134]:

```
image8 = 'tests/sample8.jpg'  
detect(image8)
```

7

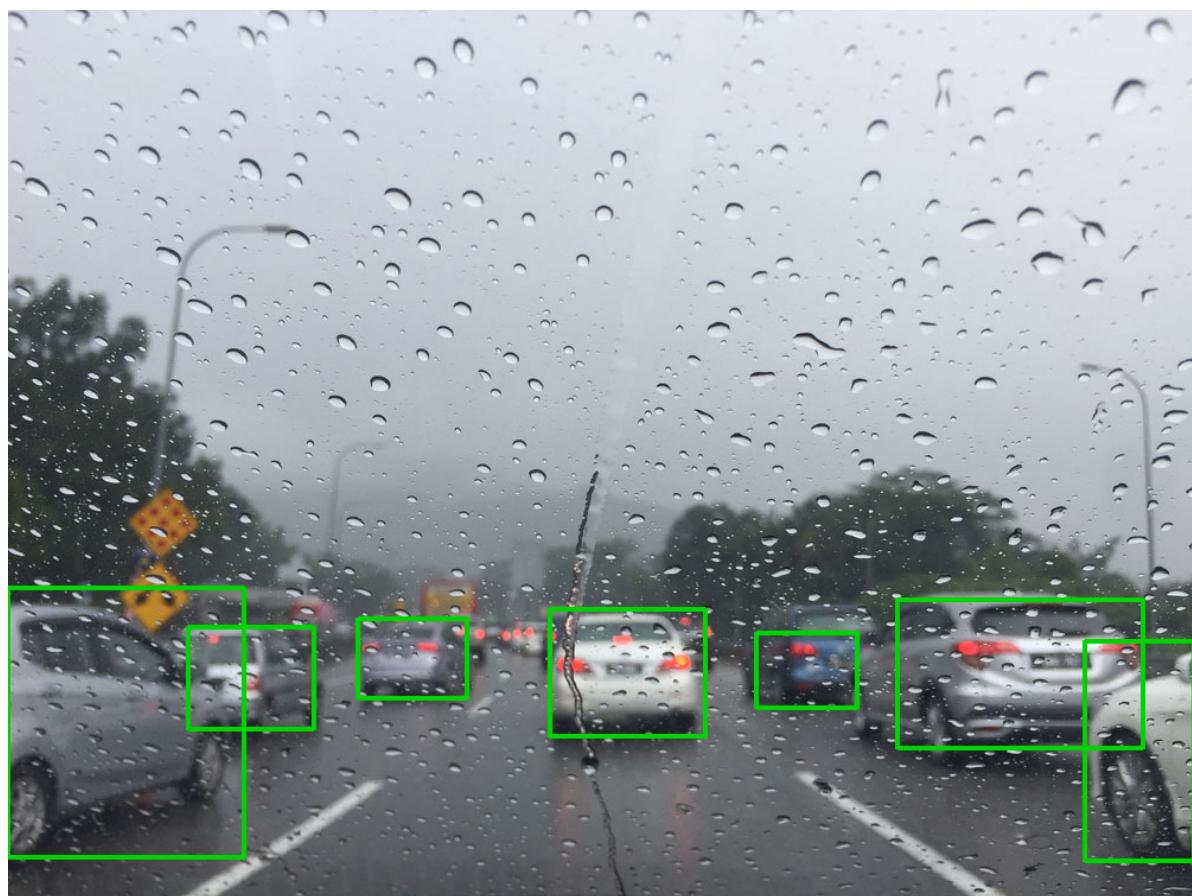


Test 9 : Can you spot the cars in Monsoons of Tropical India? Well... Our project can

In [136]:

```
image9 = 'tests/sample9.jpg'  
detect(image9)
```

7



Test 10 : A lot of cars

In [130]:

```
image10 = 'tests/sample10.jpg'  
detect(image10)
```

27

