250 Frames Data

July 26, 2018

```
In [3]: import pandas as pd
        import numpy as np
        import matplotlib
        import os
        import glob
        import subprocess

In [7]: path="C:/Users/asd/Desktop/sort-tracker/India 2/250 frames"

In [14]: def height(ybr,ytl):
              return abs(ybr-ytl)
         def width(xbr,xtl):
              return abs(xbr-xtl)
         def avg(arr,total):
             npArr=np.asarray(arr)
             average=np.sum(npArr)/total
             return average

In [16]: totalFrames=0
         totalID=0
         videoNo=0
         totalbbox=0
         totalPeds=0
         occPeds=0
         totalBus=0
         occBus=0
         totalTwo=0
         occTwo=0
         totalCar=0
         occCar=0
         totalTruck=0
         occTruck=0
         totalAuto=0
         occAuto=0
         totalCyclist=0
         occCyclist=0
         totalAnimal=0
         occAnimal=0
```

```python
allClasses=['Car', 'Two-Wheeler', 'Pedestrian', 'Auto-Rickshaw', 'Truck', 'Cyclist',
irrcount=0
ijkPedCross=[]
ijkPark=[]
ijkStop=[]
ijkRight=[]
ijkU=[]
pedWidth=[]
pedHeight=[]
carWidth=[]
carHeight=[]
busWidth=[]
busHeight=[]
autoWidth=[]
autoHeight=[]
truckWidth=[]
truckHeight=[]
twoWidth=[]
twoHeight=[]
cyclistWidth=[]
cyclistHeight=[]
animalWidth=[]
animalHeight=[]
uniqPeds=0
uniqCars=0
uniqTrucks=0
uniqBuses=0
uniqCyclists=0
uniqTwos=0
uniqAnimals=0
uniqAutos=0
#i represents the outer folders representing different persons' video folders
list2=glob.glob(path+"/*.csv")
for item in list2:
    fullPath=item
    readFile=pd.read_csv(fullPath)
    totalbbox=totalbbox+len(readFile)
    pedsIDuniq=[]
    carsIDuniq=[]
    busesIDuniq=[]
    trucksIDuniq=[]
    cyclistsIDuniq=[]
    twosIDuniq=[]
    animalsIDuniq=[]
    autosIDuniq=[]
    totalFrames=totalFrames + max(readFile.frameNumber)#total frames added
    ids=readFile.id
    idSet=set(ids)
```

```python
totalID=totalID+len(idSet)#total number of dstinct ids or total number of objects
videoNo=videoNo+1#total number of videos
for k in range(len(readFile)):
    if(readFile.label[k]=='Pedestrian'):
        totalPeds=totalPeds+1
        pedWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
        pedHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
        if(not(readFile.id[k] in pedsIDuniq)):
            uniqPeds=uniqPeds+1
            pedsIDuniq.append(readFile.id[k])
        if(readFile.isOccluded[k]==1):
            occPeds=occPeds+1
    elif(readFile.label[k]=='Car'):
        totalCar=totalCar+1
        carWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
        carHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
        if(not(readFile.id[k] in carsIDuniq)):
            uniqCars=uniqCars+1
            carsIDuniq.append(readFile.id[k])
        if(readFile.isOccluded[k]==1):
            occCar=occCar+1
    elif(readFile.label[k]=='Truck'):
        totalTruck=totalTruck+1
        truckWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
        truckHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
        if(not(readFile.id[k] in trucksIDuniq)):
            uniqTrucks=uniqTrucks+1
            trucksIDuniq.append(readFile.id[k])
        if(readFile.isOccluded[k]==1):
            occTruck=occTruck+1
    elif(readFile.label[k]=='Bus'):
        totalBus=totalBus+1
        busWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
        busHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
        if(not(readFile.id[k] in busesIDuniq)):
            uniqBuses=uniqBuses+1
            busesIDuniq.append(readFile.id[k])
        if(readFile.isOccluded[k]==1):
            occBus=occBus+1
    elif(readFile.label[k]=='Two-Wheeler'):
        totalTwo=totalTwo+1
        twoWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
        twoHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
        if(not(readFile.id[k] in twosIDuniq)):
            uniqTwos=uniqTwos+1
            twosIDuniq.append(readFile.id[k])
        if(readFile.isOccluded[k]==1):
            occTwo=occTwo+1
```

```python
            elif(readFile.label[k]=='Auto-Rickshaw'):
                totalAuto=totalAuto+1
                autoWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
                autoHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
                if(not(readFile.id[k] in autosIDuniq)):
                    uniqAutos=uniqAutos+1
                    autosIDuniq.append(readFile.id[k])
                if(readFile.isOccluded[k]==1):
                    occAuto=occAuto+1
            elif(readFile.label[k]=='Cyclist'):
                totalCyclist=totalCyclist+1
                cyclistWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
                cyclistHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
                if(not(readFile.id[k] in cyclistsIDuniq)):
                    uniqCyclists=uniqCyclists+1
                    cyclistsIDuniq.append(readFile.id[k])
                if(readFile.isOccluded[k]==1):
                    occCyclist=occCyclist+1
            elif(readFile.label[k]=='Animal'):
                totalAnimal=totalAnimal+1
                animalWidth.append(width(readFile.x_BR[k],readFile.x_TL[k]))
                animalHeight.append(height(readFile.y_BR[k],readFile.y_TL[k]))
                if(not(readFile.id[k] in animalsIDuniq)):
                    uniqAnimals=uniqAnimals+1
                    animalsIDuniq.append(readFile.id[k])
                if(readFile.isOccluded[k]==1):
                    occAnimal=occAnimal+1
            elif(readFile.label[k]=='Pedestrian Crossing'):
                irrcount=irrcount+1
                #ijkPedCross.append(str(i)+"-"+str(j)+"-"+str(k))
            elif(readFile.label[k]=='No Parking'):
                irrcount=irrcount+1
                #ijkPark.append(str(i)+"-"+str(j)+"-"+str(k))
            elif(readFile.label[k]=='No Stopping'):
                irrcount=irrcount+1
                #ijkStop.append(str(i)+"-"+str(j)+"-"+str(k))
            elif(readFile.label[k]=='Right Turn'):
                irrcount=irrcount+1
                #ijkRight.append(str(i)+"-"+str(j)+"-"+str(k))
            elif(readFile.label[k]=='U-Turn'):
                irrcount=irrcount+1
                #ijkU.append(str(i)+"-"+str(j)+"-"+str(k))
            elif(not(readFile.label[k] in allClasses)):
                allClasses.append(readFile.label[k])

In [17]: print("Total number of frames",totalFrames)
         print("Total number of distinct ids or total number of distinct objects",totalID)
         print("Total sequences or videos",videoNo)
```

```python
print("Total Number of Bounding Boxes", totalbbox)
print("Total Number of pedestrians", totalPeds )
print("occluded pedestrians",occPeds)
print("number of unique pedestrians",uniqPeds)
print("Total Number of cars", totalCar )
print("occluded cars",occCar)
print("number of unique Cars",uniqCars)
print("Total Number of buses", totalBus )
print("occluded buses",occBus)
print("number of unique buses",uniqBuses)
print("Total Number of Trucks", totalTruck )
print("occluded Trucks",occTruck)
print("number of unique trucks",uniqTrucks)
print("Total Number of Animals", totalAnimal )
print("occluded Animals",occAnimal)
print("number of unique animals",uniqAnimals)
print("Total Number of Cyclists", totalCyclist )
print("occluded Cyclists",occCyclist)
print("number of unique cyclists",uniqCyclists)
print("Total Number of two-wheelers", totalTwo )
print("occluded two-wheelers",occTwo)
print("number of unique two-wheelers",uniqTwos)
print("Total Number of Auto-Rikshaws", totalAuto )
print("occluded Auto-Rikshaws",occAuto)
print("number of unique autos",uniqAutos)
print("totalbbox", totalPeds+totalCar+totalBus+totalTruck+totalCyclist+totalTwo+totalA
print("Total number of bounding boxes for the 8 class objects", totalbbox-irrcount)
print("All the classes of data that are present",allClasses)
avgPedWidth=avg(pedWidth,totalPeds)
avgPedHeight=avg(pedHeight,totalPeds)
avgCarWidth=avg(carWidth,totalCar)
avgCarHeight=avg(carHeight,totalCar)
avgTruckWidth=avg(truckWidth,totalTruck)
avgTruckHeight=avg(truckHeight,totalTruck)
avgBusWidth=avg(busWidth,totalBus)
avgBusHeight=avg(busHeight,totalBus)
avgAnimalWidth=avg(animalWidth,totalAnimal)
avgAnimalHeight=avg(animalHeight,totalAnimal)
avgTwoWidth=avg(twoWidth,totalTwo)
avgTwoHeight=avg(twoHeight,totalTwo)
avgCyclistWidth=avg(pedWidth,totalCyclist)
avgCyclistHeight=avg(pedHeight,totalCyclist)
avgAutoWidth=avg(pedWidth,totalAuto)
avgAutoHeight=avg(pedHeight,totalAuto)
print("average pedestrian width",avgPedWidth)
print("average pedestrian Height",avgPedHeight)
print("average Car width",avgCarWidth)
print("average Car Height",avgCarHeight)
```

```python
        print("average Bus width",avgBusWidth)
        print("average Bus Height",avgBusHeight)
        print("average Auto width",avgAutoWidth)
        print("average Auto Height",avgAutoHeight)
        print("average Cyclist width",avgCyclistWidth)
        print("average Cyclist Height",avgCyclistHeight)
        print("average Two Wheeler width",avgTwoWidth)
        print("average Two Wheeler Height",avgTwoHeight)
        print("average Truck width",avgTruckWidth)
        print("average Truck Height",avgTruckHeight)
        print("average Animal width",avgAnimalWidth)
        print("average Animal Height",avgAnimalHeight)
        print("Number of road sign boxes",irrcount)
        print("The person-video-frame info for annotation of Pedestrian Crossing marking \n",
        print("The person-video-frame info for annotation of Parking Sign marking \n",ijkPark)
        print("The person-video-frame info for annotation of Stop Sign marking \n",ijkStop)
        print("The person-video-frame info for annotation of Right Turn marking \n",ijkRight)
        print("The person-video-frame info for annotation of U-Turn marking \n",ijkU)
```

```
Total number of frames 2911
Total number of distinct ids or total number of distinct objects 231
Total sequences or videos 12
Total Number of Bounding Boxes 19216
Total Number of pedestrians 3743
occluded pedestrians 902
number of unique pedestrians 58
Total Number of cars 8934
occluded cars 2682
number of unique Cars 96
Total Number of buses 420
occluded buses 354
number of unique buses 3
Total Number of Trucks 263
occluded Trucks 9
number of unique trucks 2
Total Number of Animals 0
occluded Animals 0
number of unique animals 0
Total Number of Cyclists 167
occluded Cyclists 2
number of unique cyclists 3
Total Number of two-wheelers 3428
occluded two-wheelers 626
number of unique two-wheelers 46
Total Number of Auto-Rikshaws 2233
occluded Auto-Rikshaws 283
number of unique autos 21
totalbbox 19188
```

```
Total number of bounding boxes for the 8 class objects 19188
All the classes of data that are present ['Car', 'Two-Wheeler', 'Pedestrian', 'Auto-Rickshaw',
average pedestrian width 41.4060913706
average pedestrian Height 88.1514827678
average Car width 113.866241325
average Car Height 72.3588538169
average Bus width 118.369047619
average Bus Height 76.3333333333
average Auto width 69.4057321988
average Auto Height 147.761307658
average Cyclist width 928.041916168
average Cyclist Height 1975.75449102
average Two Wheeler width 61.9302800467
average Two Wheeler Height 78.8380980163
average Truck width 80.5323193916
average Truck Height 82.8669201521
average Animal width nan
average Animal Height nan
Number of road sign boxes 28
The person-video-frame info for annotation of Pedestrian Crossing marking
 []
The person-video-frame info for annotation of Parking Sign marking
 []
The person-video-frame info for annotation of Stop Sign marking
 []
The person-video-frame info for annotation of Right Turn marking
 []
The person-video-frame info for annotation of U-Turn marking
 []


c:\python36\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarning: invalid value encountere
  import sys
```