**18CS3074S – Deep Learning**

**PROJECT BASED REPORT**

**ON**

# **PARKING OCCUPANCY AND PATTERN ANALYSIS**

Submitted in partial fulfillment of the requirement for the award of the degree of

**BACHELOR OF TECHNOLOGY**

**In**

**COMPUTER SCIENCE AND ENGINEERING**

**By**

**BATCH-4**

**KOTHAPALLI MOHAN SAI (180030360)**

******

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**K L (Deemed to be) University**

Green Fields, Vaddeswaram, Guntur District – 522 502

**(2020-2021)**

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



## **CERTIFICATE**

This is to certify that the project based laboratory report entitled “**Parking Occupancy and Pattern analysis**” submitted by **K.MOHAN SAI** bearing Regd. No. **180030360** to the Department of Computer Science and Engineering, KL University in partial fulfillment of the requirements for the completion of a project based Laboratory in “**DEEP LEARNING**” course in BTech VI Semester, is a bonafide record of the work carried out by them under my supervisionduringtheacademicyear2020–2021.

PROJECTSUPERVISOR HEAD OF THE DEPARTMENT

Dr. SAGAR IMAMBI Dr. HARI KIRAN VEGE

**DEPARTMENT OF COMPUTER SCIENCE ENGINEERING**



## **DECLARATION**

We hereby declare that this project based lab report entitled **“Parking Occupancy and Pattern Analysis”** has been prepared by us in partial fulfillment of the requirement for the award of degree “**BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE ENGINEERING**” during the academic year 2020-2021.

We also declare that this project based lab report is of our own effort and it has not been submitted to any other university for the award of any degree.

**Date: 02-04-2021**

**Place: Vaddeswaram**

**KOTHAPALLI MOHAN SAI (180030360)**

## **ACKNOWLEDGEMENT**

It is great pleasure for me to express my gratitude to our honorable President **Sri. Koneru Satyanarayana**, for giving the opportunity and platform with facilities in accomplishing the project based laboratory report.

I express the sincere gratitude to our principal Prof Dr.Deepak for his administration towards our academic growth.

I express sincere gratitude to HOD of CSE Dept. **Dr. HariKiran Vege** for his leadership and constant motivation provided in successful completion of our academic semester. I record it as my privilege to deeply thank for providing us the efficient faculty and facilities to make our ideas into reality.

I express my sincere thanks to our project supervisor **Dr. SAGAR IMAMBI** for his novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to venture this project successfully.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **PAGE NO** |
| 1 | Abstract | 06 |
| 2 | Introduction | 07 |
| 3 | Methodology | 09 |
|  |  |  |
|  |  |  |
|  | **ABSTRACT** |  |
|  |  |  |

Parking issues have been receiving increasing attention. An accurate parking occupancy prediction is considered to be a key prerequisite to optimally manage limited parking resources. However, parking prediction research that focuses on estimating the occupancy for various parking lots, which is critical to the coordination management of multiple parks (e.g., district-scale or city-scale), is relatively limited. This project aims to analyse the performance of different prediction methods with regard to parking occupancy, considering different images of busy and free where busy means there’s no place and free means there’s place to park the vehicle.We want to implement this using CNN and Mobilenet v2 architecture as parking occupancy can be easily checked using mobile.

**INTRODUCTION**

## **Convolutional Neural Networks (CNNs / ConvNets):**

Convolutional Neural Networks are very similar to ordinary Neural Networks from the previous chapter: they are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. The whole network still expresses a single differentiable score function: from the raw image pixels on one end to class scores at the other. And they still have a loss function (e.g. SVM/Softmax) on the last (fully-connected) layer and all the tips/tricks we developed for learning regular Neural Networks still apply.

In this project, the dataset contains images and using these images we must predict whether there’s parking available or not.The dataset contains Full , Free images and we need to find occupancy of the parking.

**Sample Images** :

Free:



Full :



**METHODOLOGY**

We are using Convolutional Neural Networks and MobileV2NET architecture to solve this project

**WHERE DO THE DATASET COLLECTED**

We collected the dataset from “[www.kaggle.com](http://www.kaggle.com)”. The dataset link is “https://www.kaggle.com/daggysheep/find-a-car-park”

**PREPROCESSING THE TRAIN AND TEST DATA**

Because it is all about images, it is enough to standardize images and using imagedatagenerator we can create some more images.

**CODE**

import numpy as np

import pandas as pd

import os

for dirname, \_, filenames in os.walk('/content/drive/MyDrive/data'):

  for filename in filenames:

        print(os.path.join(dirname, filename))

from tensorflow.compat.v1 import ConfigProto

from tensorflow.compat.v1 import InteractiveSession

import tensorflow as tf

from tensorflow.keras.preprocessing.image import ImageDataGenerator

train\_datagen = ImageDataGenerator(rescale= 1./255,shear\_range= 0.2,zoom\_range=0.2,horizontal\_flip = True)

training\_set = train\_datagen.flow\_from\_directory('/content/drive/MyDrive/data',target\_size = (64,64),batch\_size= 32,class\_mode = 'binary')

from tensorflow.keras.layers import Conv2D

cnn = tf.keras.models.Sequential()

cnn.add(tf.keras.layers.Conv2D(filters=32, padding= 'same', kernel\_size = 3, activation = 'relu', input\_shape = [64,64,3]))

cnn.add(tf.keras.layers.MaxPool2D(pool\_size=2, strides =2))

cnn.add(tf.keras.layers.Conv2D(filters= 32, padding='same', kernel\_size=3, activation = 'relu'))

cnn.add(tf.keras.layers.MaxPool2D(pool\_size =2, strides = 2))

cnn.add(tf.keras.layers.Flatten())

cnn.add(tf.keras.layers.Dense(units=128, activation = 'relu'))

cnn.add(tf.keras.layers.Dense(units=1, activation = 'sigmoid'))

cnn.summary()

cnn.compile(optimizer = 'adam', loss = 'binary\_crossentropy', metrics= ['accuracy'])

cnn.fit(x=training\_set, epochs=1)

import numpy as np

from tensorflow.keras.preprocessing import image

test\_image = image.load\_img('/content/drive/MyDrive/data/Free/img\_1006075057.jpg', target\_size = (64,64))

test1 = image.img\_to\_array(test\_image)

test1 = test1/255

test1 = np.expand\_dims(test1, axis=0)

result = cnn.predict(test1)

result

if result[0] <= 0.4:

    print('There is no parking available')

else:

    print('Parking available')

import matplotlib.pyplot as plt

import matplotlib.image as img

plt.title('%.2f%% Free' % (result[0] \* 100))

plt.imshow(test\_image)

import numpy as np

from tensorflow.keras.preprocessing import image

test\_image1 = image.load\_img('/content/drive/MyDrive/data/Full/img\_1002004102.jpg', target\_size = (64,64))

test2 = image.img\_to\_array(test\_image1)

test2 = test2/255

test2 = np.expand\_dims(test2, axis=0)

result1 = cnn.predict(test2)

result1

if result1[0] <= 0.4:

    print('Parking Available')

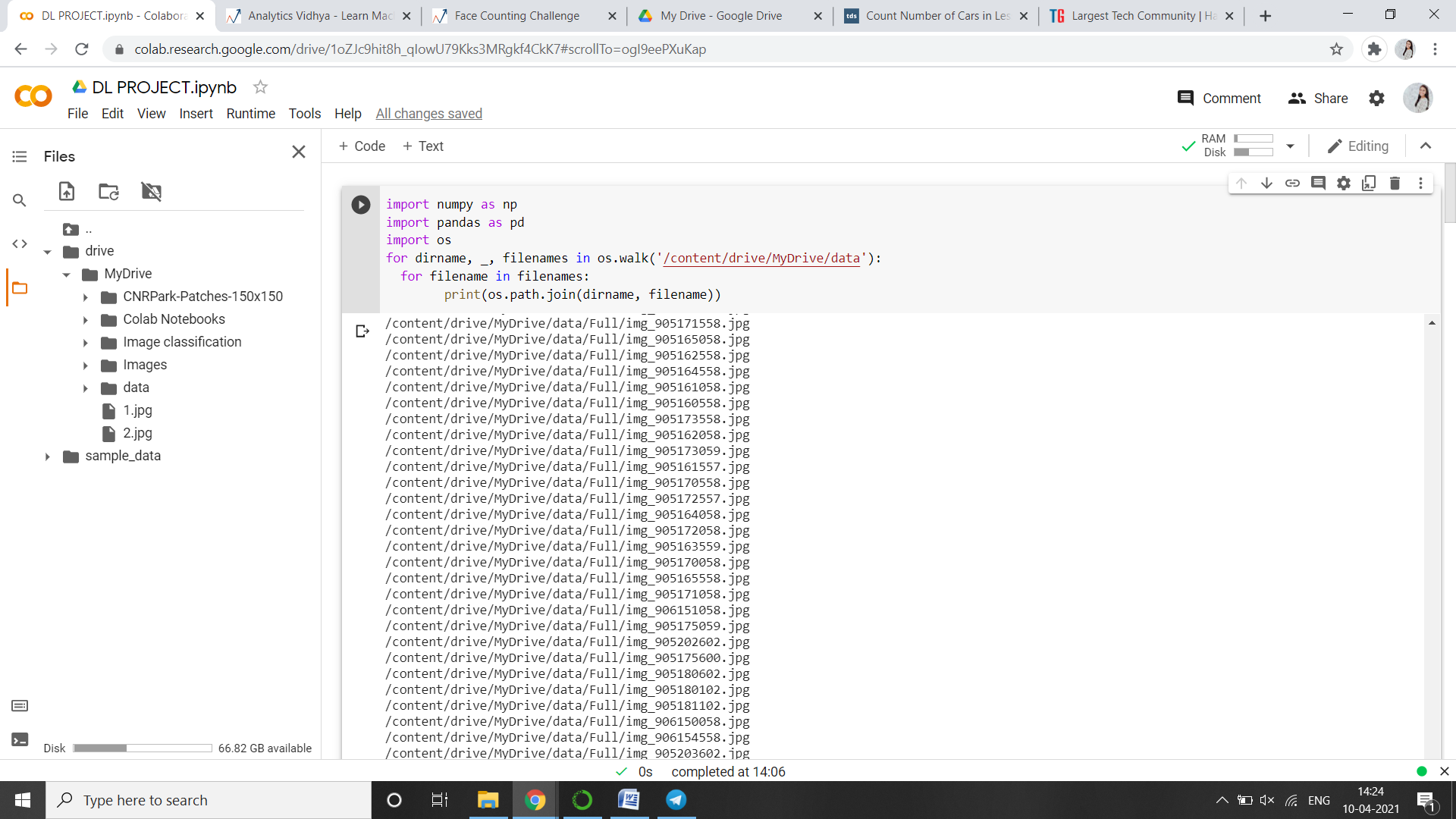
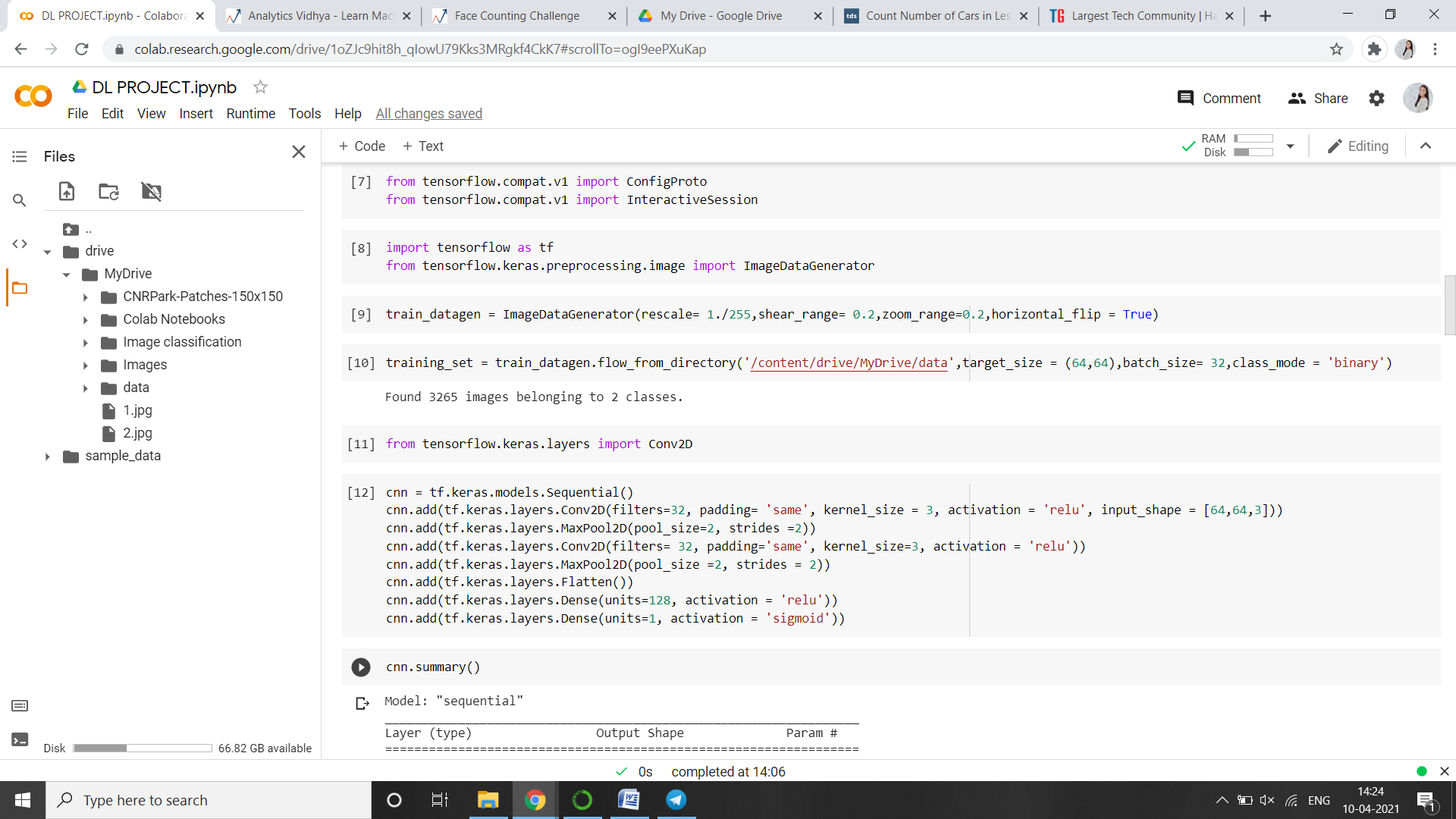
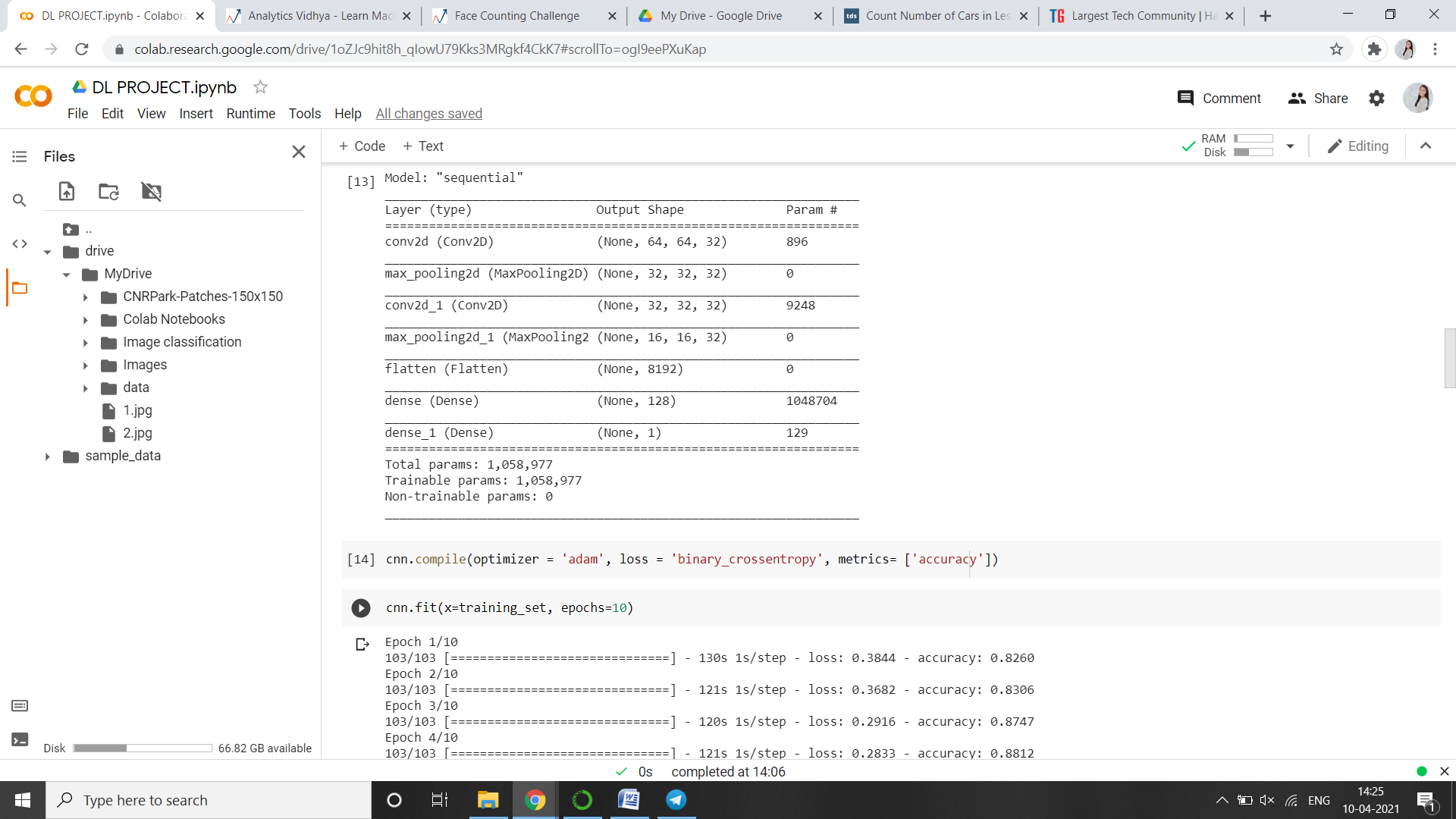
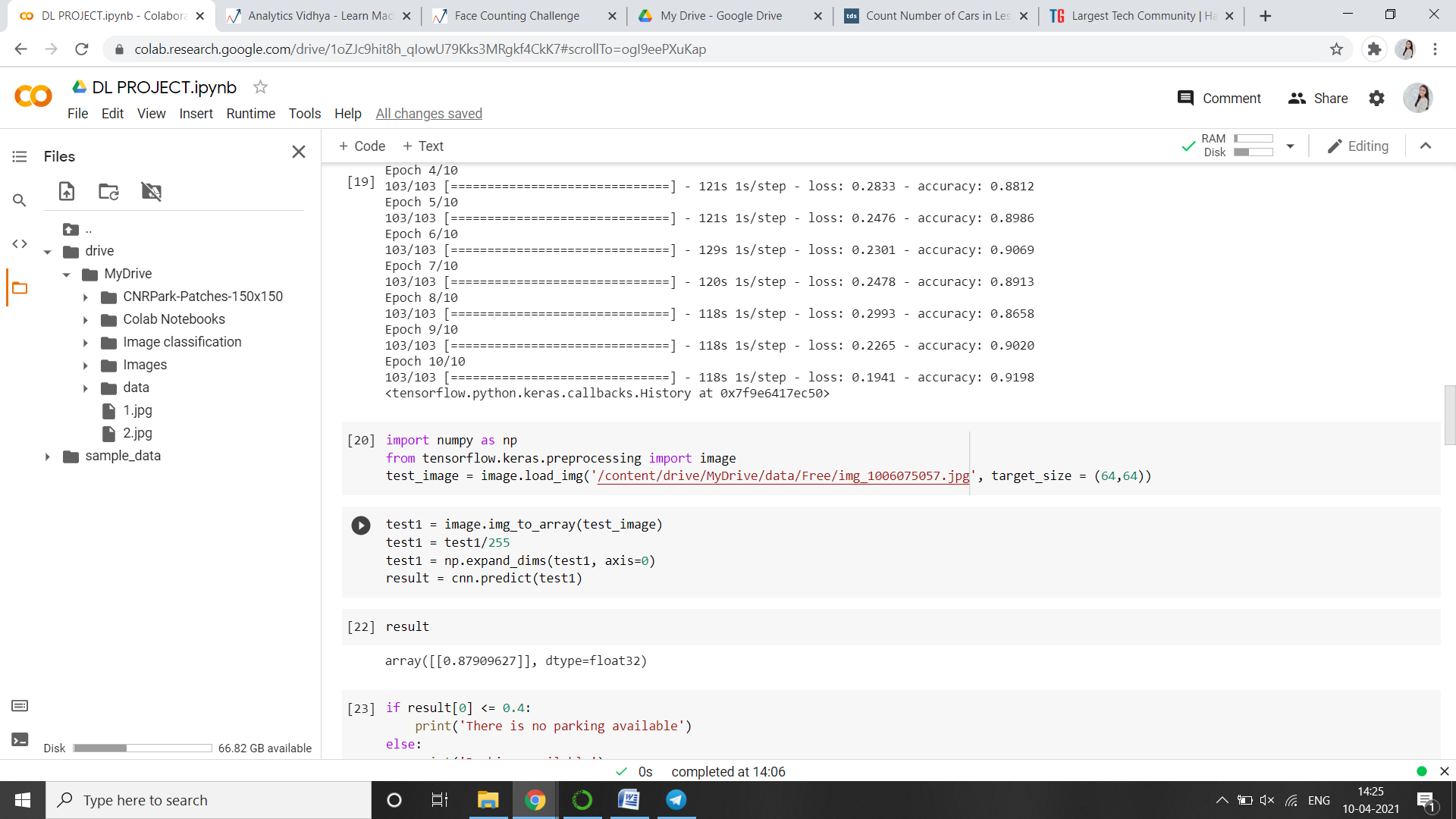
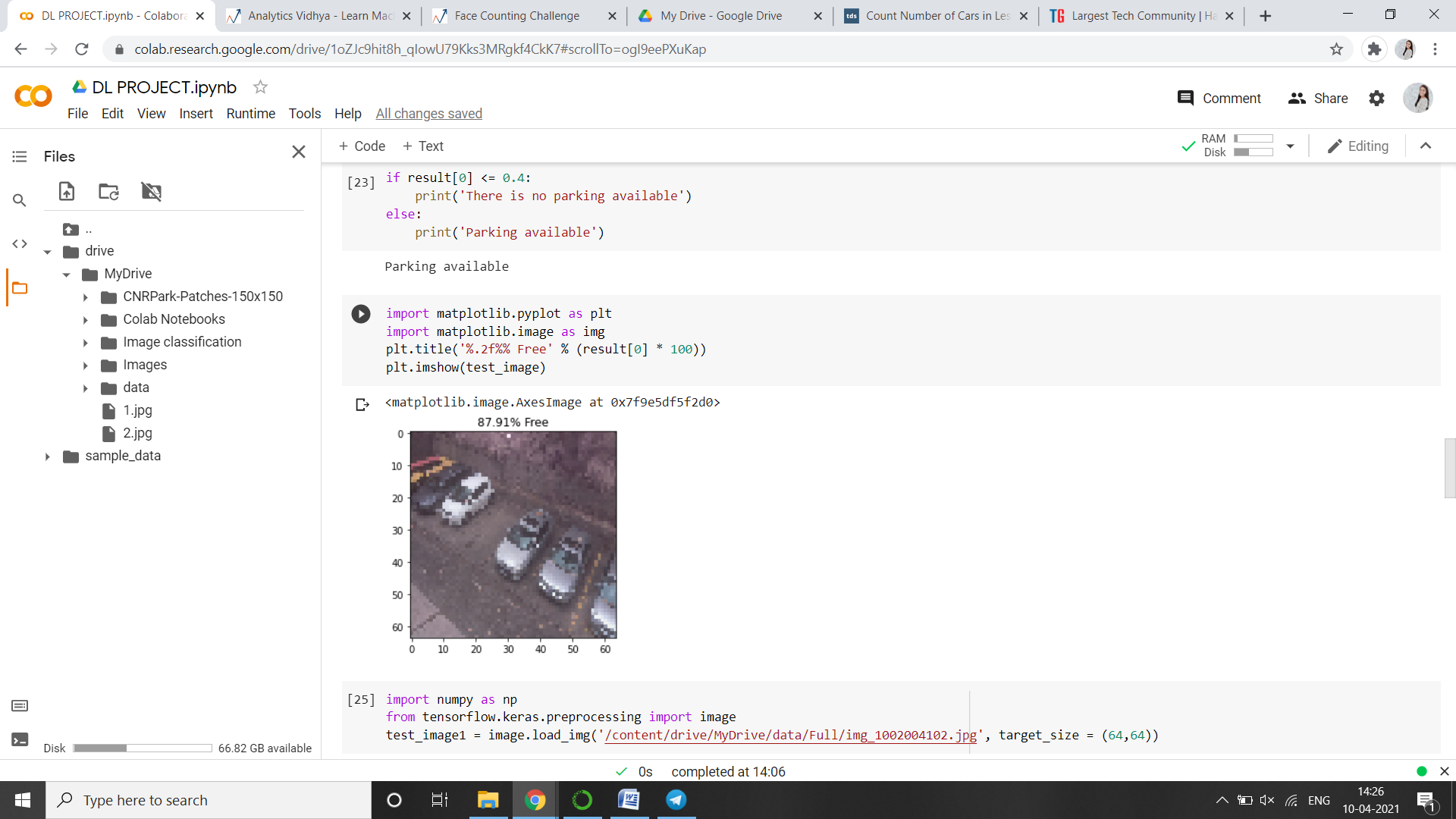
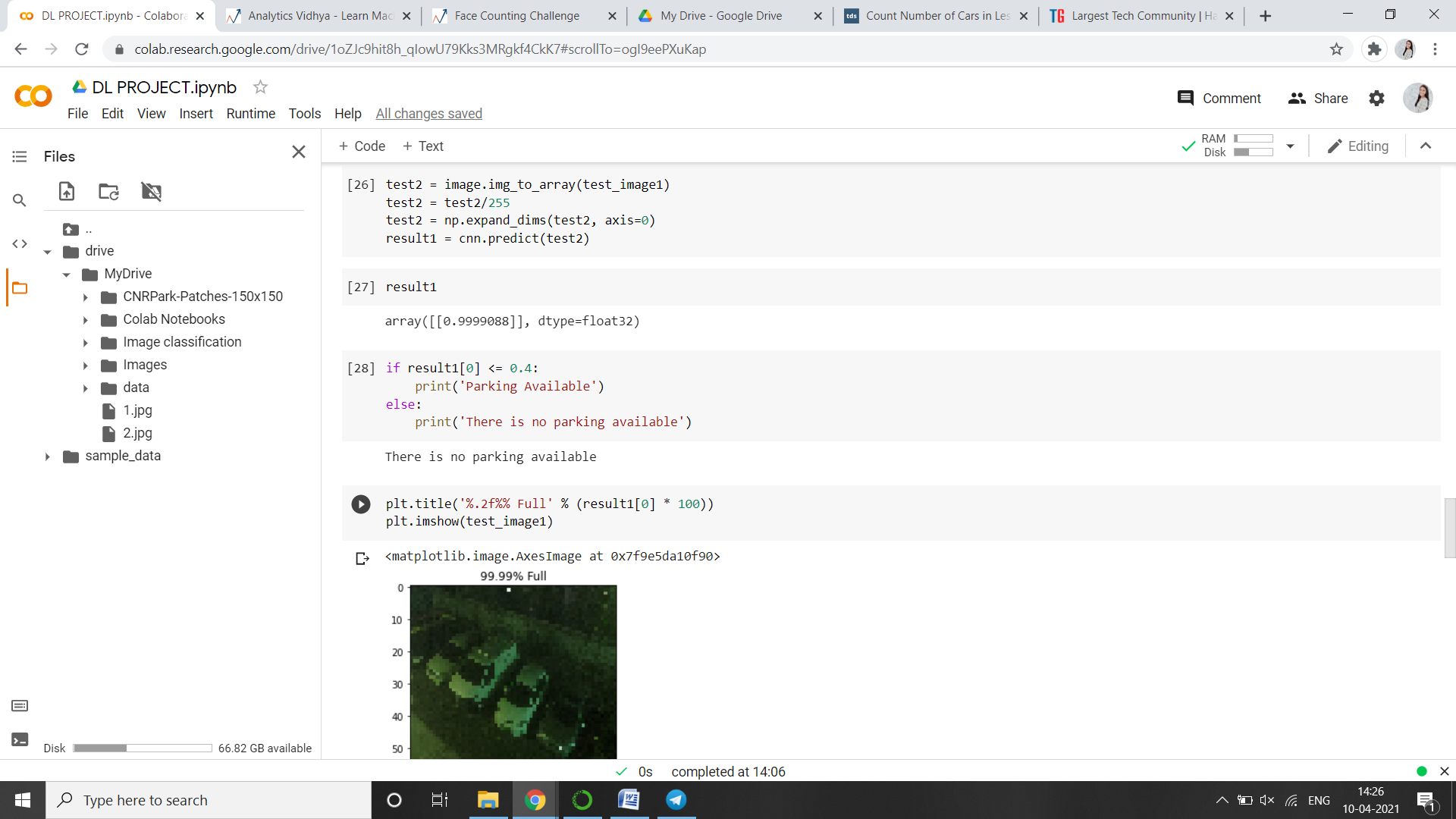
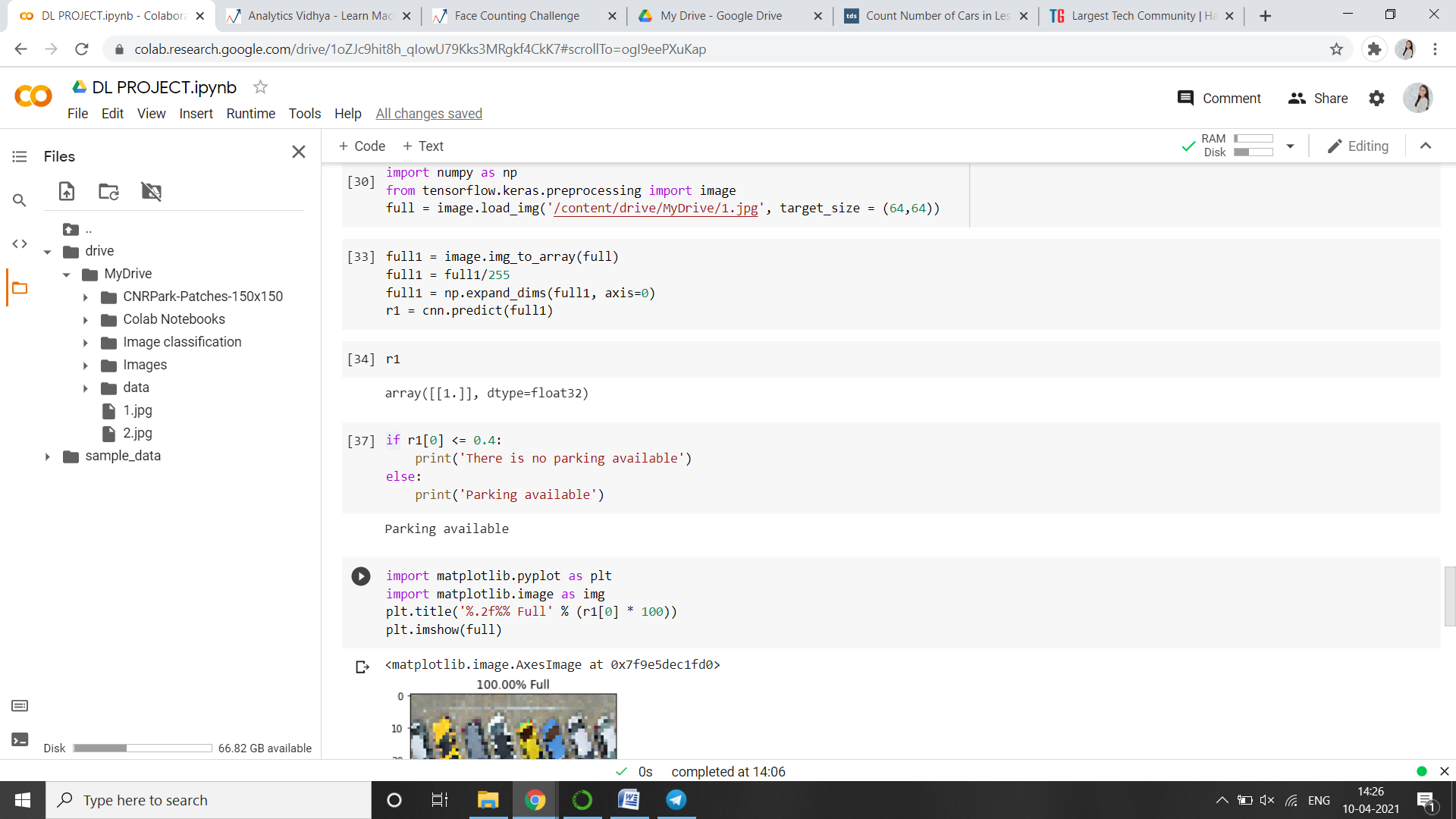
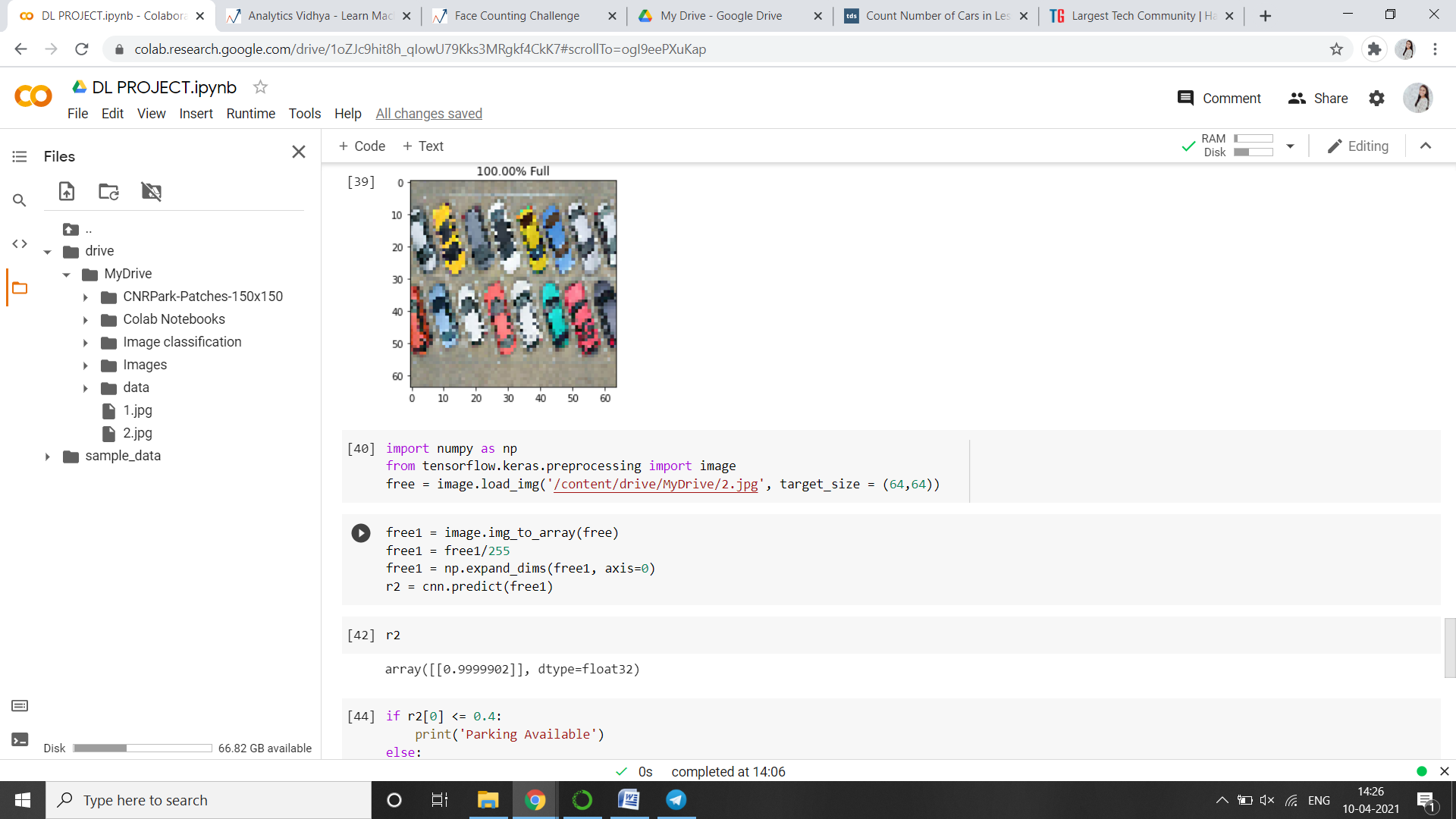
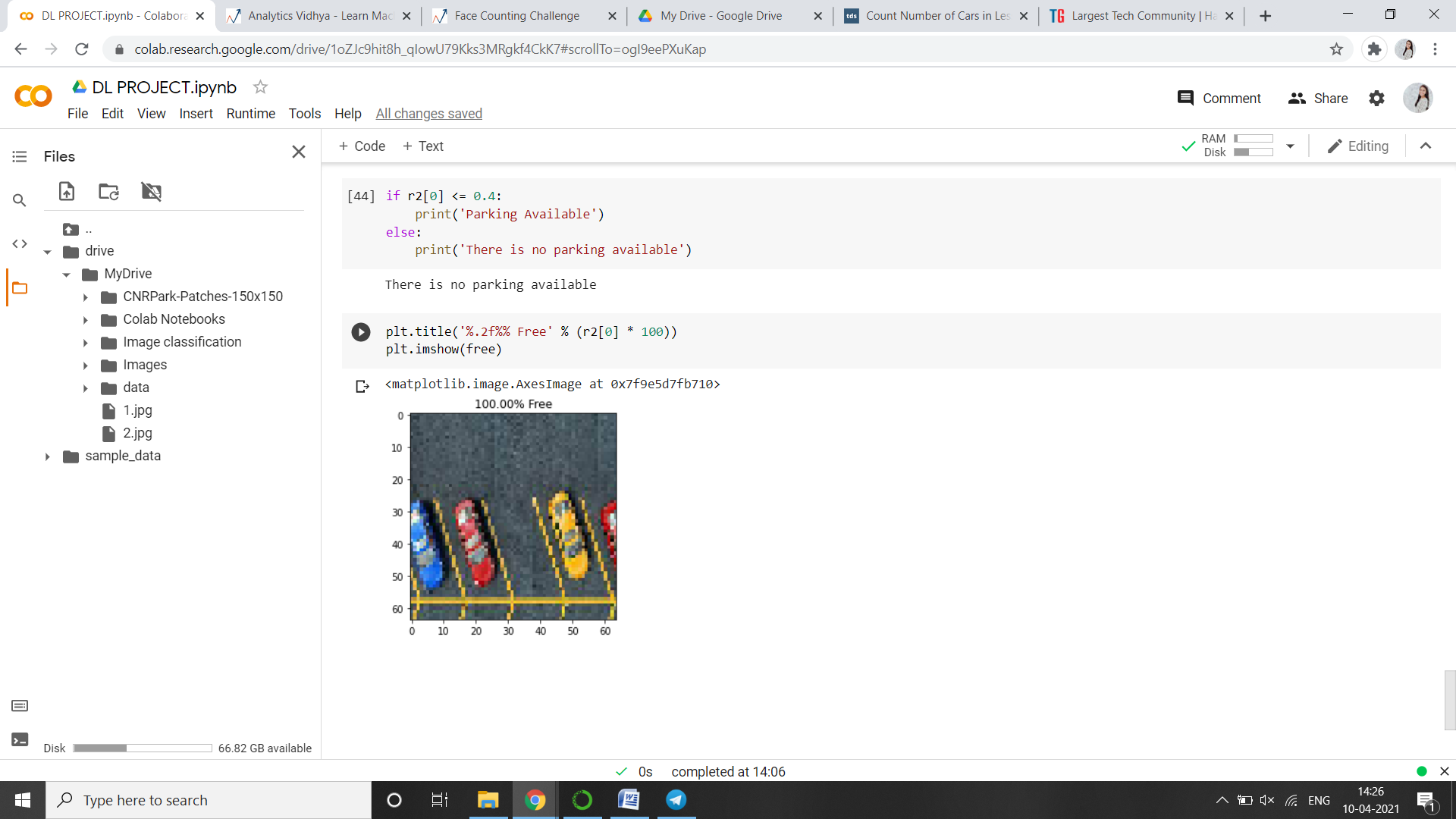
else:

    print('There is no parking available')

plt.title('%.2f%% Free' % (result1[0] \* 100))

plt.imshow(test\_image1)

**SCREENSHOTS**

**        **

**CONCLUSION**

This project predicts whether there is parking available or not. And checks how many slots are left free in a test image.

**REFERENCES**

1. <https://www.hindawi.com/journals/jat/2020/5624586/>
2. <https://www.sciencedirect.com/science/article/abs/pii/S0967070X15300858#:~:text=Parking%20occupancy%20is%20an%20important,capacity%20for%20a%20given%20price>.
3. https://www.kaggle.com/jonghyunlee1993/kaggle-park-lot-full-free-class-activation-mapping