

```

1 from tkinter import *
2 import csv
3 import cv2
4 from roboflow import Roboflow
5 from Drone import Drone
6
7 from Master import Master
8 from utils.Utils import Utils
9
10
11 # main.py
12 # authors: Diogo Rosário, João Raposo
13 # Description: This file retrieves the trained Roboflow models using the provided APIs and performs predictions
14 # on a specified test (image_path). Subsequently, the master agent is invoked to compile and process all images
15 # without overlap, ensuring the best possible identification of each corresponding drone.
16 # Additionally, the file manages the post-processing steps by saving the master agent's results in a CSV file named
17 # "mastersResults.csv" for future reference. Furthermore, it provides real-time visibility by printing the obtained
18 # results directly to the console during execution.
19
20
21 image_path = "mastersTests/test-10.jpg"
22 predicitonA_path = "predictions/predicitonA.jpg"
23 predicitonB_path = "predictions/predicitonB.jpg"
24 predicitonC_path = "predictions/predicitonC.jpg"
25 predictionM_path = "predictions/predictionM.jpg"
26 confidence = 51
27 overlap = 0
28
29 #Loading models of each drone
30 rfA = Roboflow(api_key="usQXRh13NGjn2HK6BwFP")
31 project1 = rfA.workspace().project("drone-a-tb89u")
32 modelA = project1.version(2).model
33
34 rfB = Roboflow(api_key="usQXRh13NGjn2HK6BwFP")
35 project2 = rfB.workspace().project("drone-b-xkvry")
36 modelB = project2.version(1).model
37
38 rfC = Roboflow(api_key="usQXRh13NGjn2HK6BwFP")
39 project3 = rfC.workspace().project("drone-c-hytyd")
40 modelC = project3.version(5).model
41
42 # Drones Reputation / confidence
43 drone_A_confidence = 0.1
44 drone_B_confidence = 0.5
45 drone_C_confidence = 0.5
46
47 # Open a csv file called "mastersResults.csv" and write the results of the identifications processed by the master.
48 # In this format: 'A - Cars', 'A - Houses', 'A - Trees', 'B - Cars', 'B - Houses', 'B - Trees', 'C - Cars', 'C - Houses', 'C - Trees', 'A Reputation', 'B
49
50 csv_filename = 'masterResults.csv'
51 with open(csv_filename, 'w', newline='') as file:
52     writer = csv.writer(file)
53     writer.writerow(['A - Cars', 'A - Houses', 'A - Trees',
54                     'B - Cars', 'B - Houses', 'B - Trees',
55                     'C - Cars', 'C - Houses', 'C - Trees',
56                     'A Reputation', 'B Reputation', 'C Reputation'])
57
58 #####
59 ##### DRONE A #####
60
61 predictA = modelA.predict(image_path, confidence=confidence, overlap=overlap)
62 drone_A = Drone("A", drone_A_confidence, predictA.json())
63 predictA.save(predicitonA_path)
64 drone_A.saveInCsv()
65
66 #####
67 ##### DRONE B #####
68
69 predictB = modelB.predict(image_path, confidence=confidence, overlap=overlap)
70 drone_B = Drone("B", drone_B_confidence, predictB.json())
71 predictB.save(predicitonB_path)
72 drone_B.saveInCsv()
73
74 #####
75 ##### DRONE C #####
76
77 predictC = modelC.predict(image_path, confidence=confidence, overlap=overlap)
78 drone_C = Drone("C", drone_C_confidence, predictC.json())
79 predictC.save(predicitonC_path)
80 drone_C.saveInCsv()
81
82 #####
83 ##### CALCULATION #####
84 #####
85
86 master = Master(drone_A, drone_B, drone_C)
87
88 clone_img = cv2.imread(image_path)
89 masterImage = clone_img.copy()
90
91 # Draw the result identifications
92 for box in master.identifications:
93     Utils.addLabel(masterImage, box.x, box.y, box.width, box.height, box.class_type, box.confidence, box.drone)
94
95 cv2.imwrite(predictionM_path, masterImage)
96 print(str(len(master.identifications)))

```