Homework 11: Image Classification

Dear 109ers,

This homework assignment requires very little coding but it will take up to 3x3 hours to train convolutional neutral networks (CNN) on your laptop. We shift the due date back by three days to give you a bit more time. You might let your laptop do the hard work while you sleep.

We assume you have completed this week's lab assignment and have a working code that allows you to classify arbitrary set of images by training CNNs.

From bCourses please download and unzip these files, which will require 2x2.8 GB of disk space:

```
241795449 Nov 5 12:53:02 2025 images_for_homework_bikes_and_scooters_8_0.zip
218199307 Nov 5 12:54:47 2025 images_for_homework_bikes_and_scooters_8_1.zip
226916067 Nov 5 12:56:31 2025 images_for_homework_bikes_and_scooters_8_2.zip
233071021 Nov 5 12:58:20 2025 images_for_homework_bikes_and_scooters_8_3.zip
119862601 Nov 5 12:27:07 2025 images_for_homework_buildings_8_4.zip
115042029 Nov 5 12:28:38 2025 images_for_homework_buildings_8_5.zip
109132335 Nov 5 12:30:06 2025 images_for_homework_buildings_8_6.zip
140166644 Nov 5 12:46:55 2025 images_for_homework_cars_8_0.zip
138271039 Nov 5 12:48:18 2025 images_for_homework_cars_8_1.zip
134039977 Nov 5 12:49:41 2025 images_for_homework_cars_8_2.zip
139030129 Nov 5 12:51:10 2025 images_for_homework_cars_8_3.zip
 87193429 Nov 5 12:36:04 2025 images_for_homework_doors_8_7.zip
 91827745 Nov 6 10:42:54 2025 images_for_homework_doors_8_8.zip
 92671145 Nov 5 12:39:21 2025 images_for_homework_doors_8_9.zip
189395912 Nov 5 12:20:17 2025 images_for_homework_plants_8_4.zip
195202545 Nov 5 12:22:06 2025 images_for_homework_plants_8_5.zip
192165496 Nov 5 12:23:44 2025 images_for_homework_plants_8_6.zip
 99044475 Nov 5 12:41:14 2025 images_for_homework_windows_8_7.zip
 91011663 Nov 6 10:41:23 2025 images_for_homework_windows_8_8.zip
101060249 Nov 5 12:44:03 2025 images_for_homework_windows_8_9.zip
```

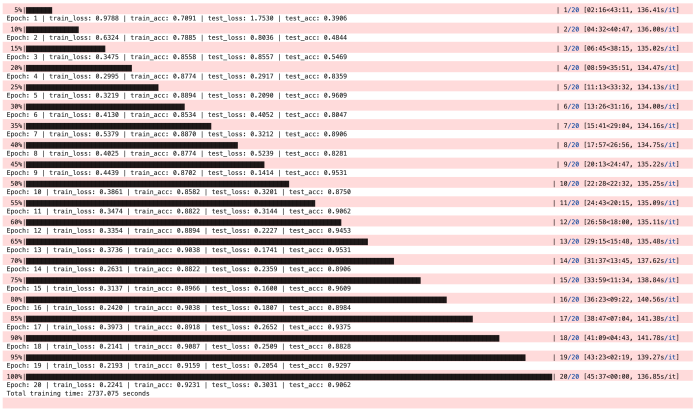
(1) Depending on the last digit of your student ID, you uploaded images for two categories:

ID = 0, 1, 2, or 3	Cars vs bicycles/scooters
ID = 4, 5, or 6	Plants vs buildings
ID = 7, 8, or 9	Doors vs windows

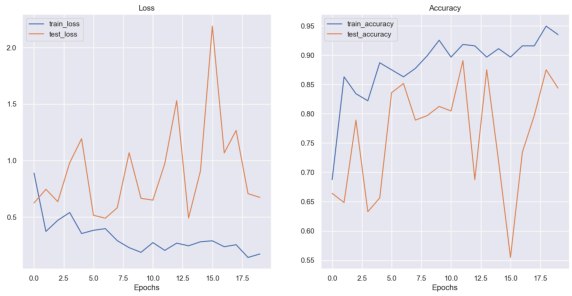
These zip files above were generated by selecting images at random from your submissions. If your images were submitted on time and their format were found to be appropriate, you will likely recognize some of your own. They were made a bit smaller and converted to the .jpg format with quality factor 6 to save you all some disk scape.

From the two .zip files to which you likely contributed to (if you last ID were **6**, this would be images_for_homework_plants_8_**6**.zip and images_for_homework_buildings_8_**6**.zip), select **2x(200+50) images** of your choice and copy them into usual training/testing folder structure:

```
$dd/training_set/plants    <<< 200
$dd/training_set/buildings <<< 200
$dd/test_set/plants        <<<  50
$dd/test_set/buildings     <<<  50
```



Then start the training of your CNN with 20 epochs, which may take between 40 minutes and 3 hours (see above). **Only if one epoch takes longer than 9 minutes on your laptop, you are permitted to reduce the number of images from the standard number of 2x(200+50) so that the whole training process does not take you much longer than 3 hours.** However, if you reduce the image number even more, points will be taken off most likely. Once the training is complete, generate the usual loss/accuracy graph:



You will be asked to submit your three notebooks for this assignment but we are also asking you to submit a separate *results.docx* file to make the grading process more efficient. **Please insert loss/accuracy graph into your results file and state which two image categories it corresponds to.**

(2) Now that you have trained you CNN, we want to subject it to a real test with many more images than you have used in your test set before. To make this process efficient we want you to first **write a function** that uses the existing CNN to classify all images in a given folder and then returns two numbers that specify how many images were predicted to be in each class. Your function may look like this:

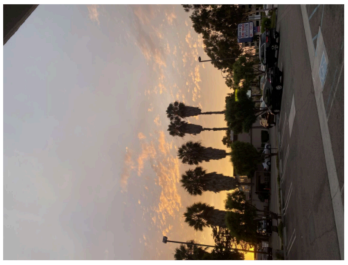
```
def ClassifyImagesInFolder(folder):
    # 1. Get all image paths (* means "any combination")
    image_path_list = glob.glob(folder+"/*")
    print(image_path_list)

    numberOfImagesPerClass = np.zeros((2)) # number of images identified in each class

    i = 0
    for custom_image_path in image_path_list:
        i += 1
        name = os.path.basename(custom_image_path) # name of image
        print(i,numberOfImagesPerClass,name)

        ...

        numberOfImagesPerClass[custom_image_pred_label.cpu()] += 1

        ...

    return numberOfImagesPerClass
```

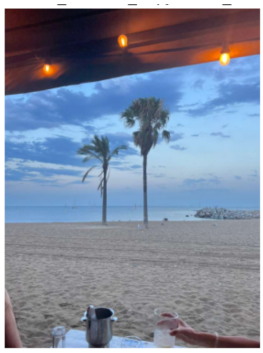
(3) For the current image classes like planets/buildings, we want to you to copy all files from the 2x2 other folders to two new folders. If your ID was 6 this would be images in folders `images_for_homework_plants_8_7`, `images_for_homework_buildings_8_8`, `images_for_homework_buildings_8_7` and `images_for_homework_plants_8_8`. Then classify all these images by calling the function above. In your *results* files, please report how many images in each class were identified correction and what their fractions were. So you should have two outputs for this format, one for each class:

```
[367.  28.]
[0.92911392 0.07088608]
```

(4) Modify the function above so that it automatically displays three images that were classified incorrectly. **Insert these three images into your *results* file.** Say if you see any reason or not why they were misclassified. Please also upload your final notebook.

(5) If you find any images that in your opinion should not have been placed inside any of these folders, **please report their names (one per line) in a separate text file named *problematic_images.txt* that we ask you to submit with your homework** so that we can remove them next year. Here are some problematic examples:



(6) Repeat every steps 1-5 for the two remaining pairs of images categories. If you ID was 6, this would planets vs buildings and cars vs bikes/scooters. Insert your findings into your *results* file and upload your two final notebooks as well.

(7) Add a concluding paragraph to your *results* file where you compare the results of step 3 for all three pairs of image categories. Say which classification problem was the easiest and which one was the hardest.

(If you know of a way to modify the existing Python code to make the training more robust, please contact the instructor.)