HW4 - Image Classification - Boat MNIST
Due 7/21/2024

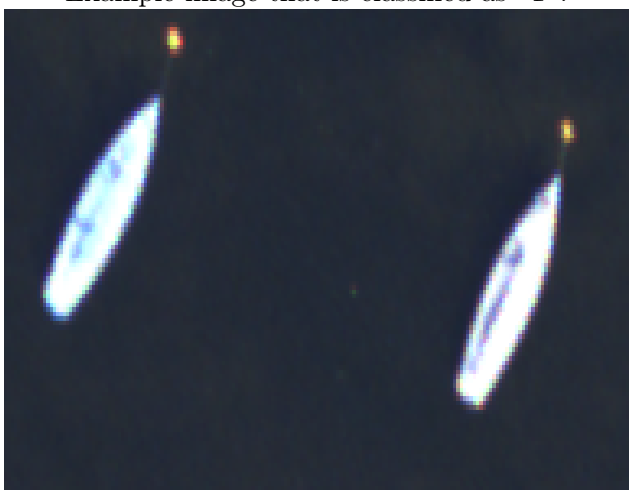
1 Introduction

The Boat MNIST data set is used as a test data for binary image classification and is provided by Lookout. It aims at providing a simple hands-on benchmark to test small neural networks. There are the following two classes:

1 - if the image contains any watercraft instance including boats, ships, surfboards, ... ON the water

0 - all the rest, i.e. just water or anything on the land (could also be boats)

Example image that is classified as "1".



Your job is to download the code that I modified so that is can work on Discovery. It is fully automated and already has a simple NN built in. You will need to **answer a series of 8 questions** that I have marked as TODO explaining what each line of code is doing. You will also need to modify the code to **create your own Neural Net work to achieve an accuracy above 90%.** The code is uploaded on canvas under HW4 -> HW4.ipynb. The code that is uploaded is designed to work for the default partition -> "courses". You can change any of the training hyper parameters in def main(): I encourage you to do so!

Notes: You will have to pip install torch. I still do not have global working solution for that yet.

If you want to run a partition as "courses-gpu" you will need to select cuda/12.1 as the System-wide CUDA Module and change "no_cuda" variable in the code to "True". I did not see any significant difference between running a "courses" partition vs a "courses-CUDA" partition.

I selected 8 CPUs and 16 GBs of Memory and it takes less than a minute to run 1 epoch how the NN is currently designed right now.

The original code was created to run as a script file. I do not expect you to run it as a script file, but there are several parts that you will have to modify if you are wanting to run it as a script file yourself. I left the code that is needed to parse a script file commented out. There are several lines that you will need to either modify or un-comment to get it to work as a script file.

The original code that this assignment is designed for was created to save an ONNX file to compete in the online competition. I have uploaded the original challenge.pdf that details the steps you need to do if you want to want to upload your model as well. This is not part of the assignment.

2 Grading (Out of 100 points)

- 10 Points : Use appropriate comments throughout your code.
- 30 Points : Answer each of the 8 questions that are marked in the code. (write your answers at the end of your python code as a block comment.)
- 25 Points : Create your own NN architecture.
- 25 Points : Explain your NN architecture.
- 10 Points : Achieve an accuracy above a 90%

3 How to turn in

You must run and leave your code on your Discovery. The TAs will be able to access your file and grade on Discovery.

Good Luck Cat!