

## Homework #0

### Problem 1

The ResNet101 model was run as in the hw0p1.py file. Each of the predictions of the associated images can be observed by exporting the “HW0” zipped archive using the same file structure and modifying line 21 to be any of the images within the “randomResNetPics” folder. For example “randomResNetPics/pug.jpg.” Five different images were passed into the model and the top five predictions were observed, along with the percentage likelihood associated with each. Tables 1-5 below demonstrate the prediction/percentage of each image, and a discussion for reasons the prediction probability may be low, or if the prediction was incorrect wholly. The following code was used as a reference [dlwpt-code/2\\_pre\\_trained\\_networks.ipynb at master · deep-learning-with-pytorch/dlwpt-code · GitHub](#).

Pug	
Prediction	Percentage (%)
Pug	99.09
French Bulldog	0.30
Brabancon Griffon	0.15
Norwegian Elkhound	0.15
Chow	0.06

Table 1: Pug Image Accuracy ResNet101

Barn	
Prediction	Percentage (%)
Barn	47.17
Church	21.42
Mobile Home	5.19
Boathouse	3.55
Monastery	3.16

Table 2: Barn Image Accuracy ResNet101

Like many of the other images, ResNet101 model correctly predicts that the input is a barn; however, with a rather low confidence which is likely caused by the non-traditional nature of the barn given.

Fan	
Prediction	Percentage (%)
Electric Fan, blower	99.90
Space Heater	0.095
Spotlight	0.0014
Radio	0.0003
Hand Blower	0.00024

Table 3: Fan Image Accuracy ResNet101

Guitar	
Prediction	Percentage (%)
Acoustic Guitar	99.96
Electric Guitar	0.021
Pick, plectrum	0.009
Cello	0.0022
Whistle	0.00097

Table 4: Guitar Image Accuracy Resnet101

Water Bottle	
Prediction	Percentage (%)
Water Bottle	83.47
Water Jug	8.98
Cocktail Shaker	5.25
Nipple	0.70
Petri Dish	0.48

Table 5: Water Bottle Image Accuracy ResNet101

## Problem 2

The trained ResnetGen network was utilized in feeding in five random input images of horses to generate an output image with the horses having a zebra pattern. Similar to problem 1, the output images can be observed by selecting a horse image from the contained "horsepics" folder within the "hw0p2.py" file.

Table 6 below shows the input and output images used with the model.


ResNetGen	
Input	Output
	
	
	



Table 6: Input and Output of ResNetGen

As seen in the table above, the ResNetGen model correctly modifies the input images of horses to be zebra-patterned with relatively high accuracy. In the second image, it appears as though the model is unable to apply the zebra pattern to a very pure white section of the horse. Similarly in the last image, the model applies the pattern to the horse pictured, but also to segments of the grass in the foreground, and mountains in the background. This is likely being caused by the color of the grass appearing very similar to that of many horses, and the mountains having a similar shape to that of a horse.

### Problem 3

The ptflops script was used in observing the MACs and number of parameters of both of the evaluations ran on the ResNet101 model and the ResnetGen model in problems #1 and #2. The results of each are below in Table 7. This output can be observed by running either the “hw0p1.py” or the “hwop2.py” programs and observing the results.

	ResNet101	ResnetGen
Number of MACs	7.85 GMac	43.53 GMac
Number of Parameters	44.55 M	11.38 M

Table 7: Complexity and Parameters of ResNet101 and ResnetGen

## Problem 4

Similar to problem 1, five of the same images were chosen and the pretrained MobileNet model was run in evaluation mode to observe a prediction and percentage probability of correct prediction. The method of executing the prediction on any of the images is the same as conducted in problem 1 but with the "hw0p4.py" file. The following tables 8-12 below show the results of various image predictions of the MobileNet model.

Pug	
Prediction	Percentage (%)
Pug	95.38
French Bulldog	2.20
Brabancon Griffon	1.08
Bull Mastiff	0.22
Boston bull, Boston Terrier	0.21

Table 8: Pug Image Accuracy MobileNet

Barn	
Prediction	Percentage (%)
Church Building	24.76
Palace	16.49
Monastery	11.26
Boathouse	9.35
Barn	6.42

Table 9: Barn Image Accuracy MobileNet

Unlike when using the ResNet101 model, the MobileNet model is unable to correctly predict the input image that was given as shown above in Table 8. The top 1 prediction from MobileNet for this image was a church; however, even this prediction gives a 24 percent confidence. The low confidence and erroneous prediction is likely caused by the barn image is not that of a typical barn's appearance. Furthermore, the prediction of a barn was actually the least likely option when this model was evaluated.

Fan	
Prediction	Percentage (%)
Electric Fan, Blower	99.99

Space Heater	0.0001
Spotlight	0.0004
Table Lamp	0.00009
Lampshade	0.00003

Table 10: Fan Image Accuracy MobileNet

Guitar	
Prediction	Percentage (%)
Acoustic Guitar	99.95
Electric Guitar	0.05
Cello	0.0005
Violin	0.0002
Banjo	0.00016

Table 11: Guitar Image Accuracy MobileNet

Water Bottle	
Prediction	Percentage (%)
Water Bottle	45.63
Cocktail Shaker	41.11
Nipple	8.91
Water Jug	1.33
Perfume, essence	1.12

Table 12: Water Bottle Image Accuracy MobileNet

Other than the barn image, the Mobilenet model was able to correctly identify all of the input images and with relatively high confidence (80%+). The ptflops library was used as in problem #3 in measuring the computational complexity and number of parameters of the MobileNet model. These results are below table 13.

	MobileNet
Number of MACs	0.32 GMac

Damian Hupka  
801091092  
ECGR 4090 - Real-Time AI

<b>Number of Parameters</b>	3.5M
-----------------------------	------

Table 13: MobileNet MACs and Parameters