Image_Classification_Neural_Nets

August 17, 2024

0.1 Image classification with neural networks

By: Haden Loveridge, Alex Parson, Biagio Alessandrello

Importing Packages

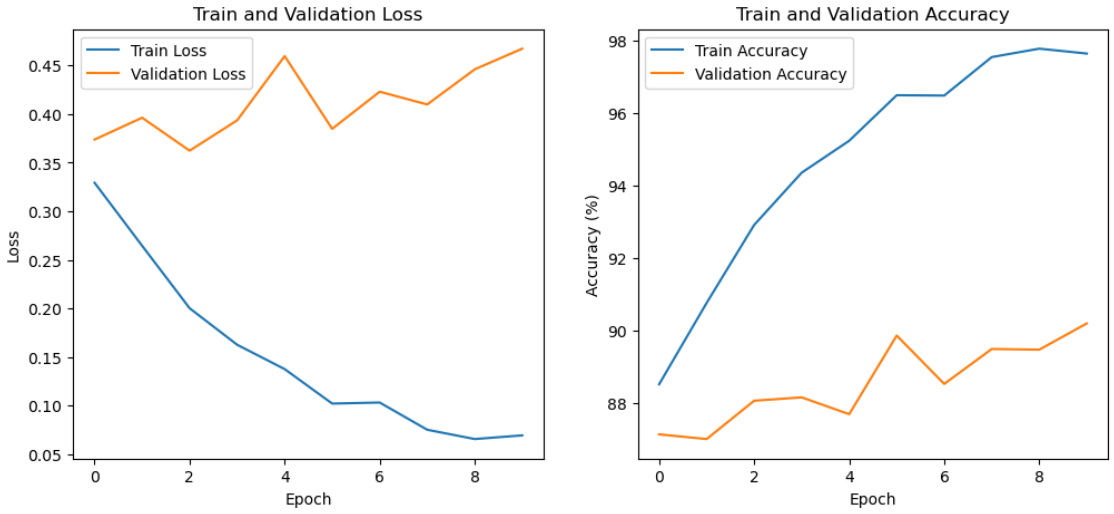
Loading in the data and creating the 80/20 testing splits

```
SimpleCNN(
  (conv1): Conv2d(3, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (conv2): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (conv3): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (pool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1,
ceil_mode=False)
  (fc1): Linear(in_features=8192, out_features=128, bias=True)
  (fc2): Linear(in_features=128, out_features=10, bias=True)
  (relu): ReLU()
)
```

```
Epoch [1/10]
Train Loss: 0.3294, Train Accuracy: 88.53%
Validation Loss: 0.3738, Validation Accuracy: 87.15%
--------------------------------------------------
Epoch [2/10]
Train Loss: 0.2646, Train Accuracy: 90.77%
Validation Loss: 0.3963, Validation Accuracy: 87.02%
--------------------------------------------------
Epoch [3/10]
Train Loss: 0.2003, Train Accuracy: 92.92%
Validation Loss: 0.3624, Validation Accuracy: 88.07%
--------------------------------------------------
Epoch [4/10]
Train Loss: 0.1628, Train Accuracy: 94.36%
Validation Loss: 0.3936, Validation Accuracy: 88.17%
--------------------------------------------------
Epoch [5/10]
Train Loss: 0.1377, Train Accuracy: 95.24%
Validation Loss: 0.4597, Validation Accuracy: 87.70%
--------------------------------------------------
```

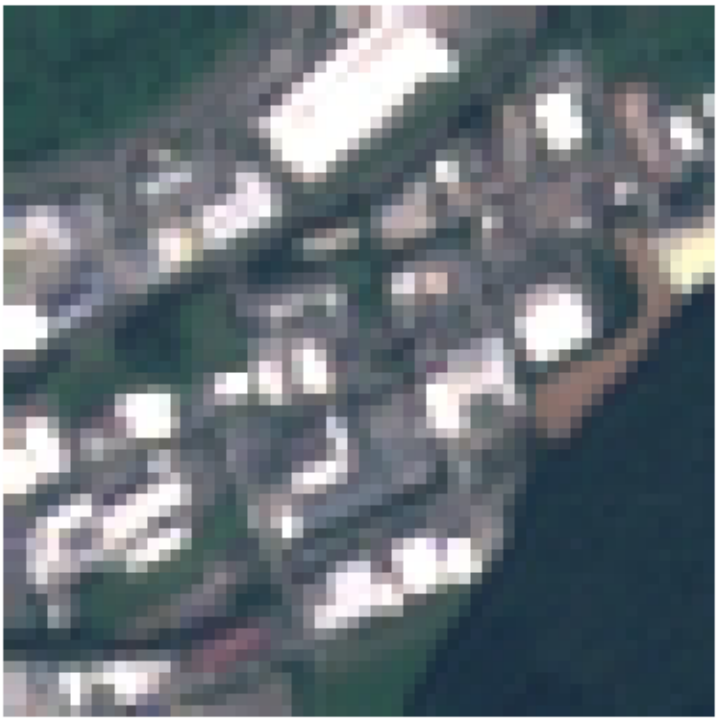
```
Epoch [6/10]
Train Loss: 0.1022, Train Accuracy: 96.49%
Validation Loss: 0.3848, Validation Accuracy: 89.87%
--------------------------------------------------
Epoch [7/10]
Train Loss: 0.1033, Train Accuracy: 96.48%
Validation Loss: 0.4230, Validation Accuracy: 88.54%
--------------------------------------------------
Epoch [8/10]
Train Loss: 0.0753, Train Accuracy: 97.54%
Validation Loss: 0.4099, Validation Accuracy: 89.50%
--------------------------------------------------
Epoch [9/10]
Train Loss: 0.0658, Train Accuracy: 97.77%
Validation Loss: 0.4460, Validation Accuracy: 89.48%
--------------------------------------------------
Epoch [10/10]
Train Loss: 0.0695, Train Accuracy: 97.64%
Validation Loss: 0.4673, Validation Accuracy: 90.20%
--------------------------------------------------
```

Using PyTorch to clasify the pictures in EuroSAT_RBG pictures into a 80/20 split. After running the model with epoch of 10, it ended up giving the following results on the final Epoch of Train Loss: 0.0695, Train Accuracy: 97.64%, Validation Loss: 0.4673, Validation Accuracy: 90.20%.



As you can see above on the Train Loss and Accuracy charts, the two are slowly diverging. As the model trains more and more on the training data due to overfitting, the loss increases.

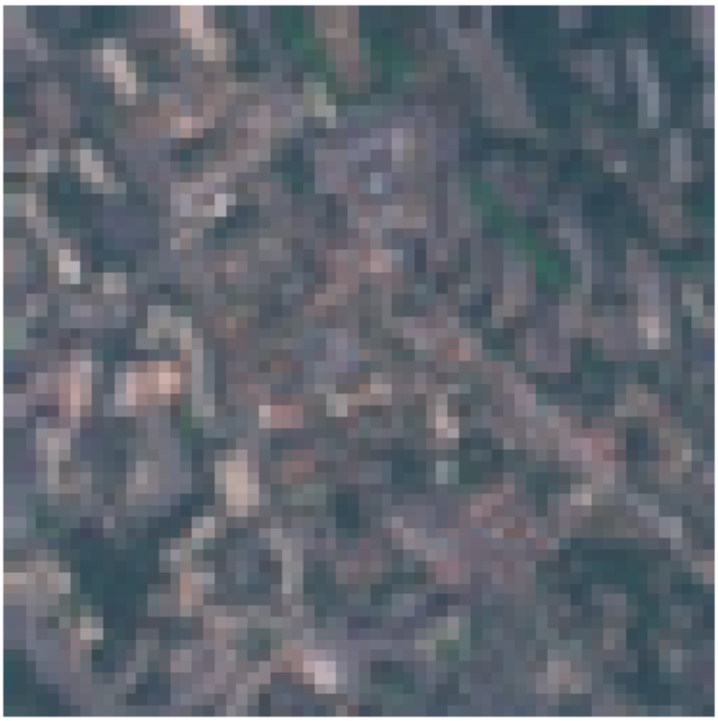
Filename: Industrial_110.jpg
Predicted class: Industrial



Filename: PermanentCrop_502.jpg
Predicted class: PermanentCrop

Filename: Pasture_6.jpg
Predicted class: Pasture

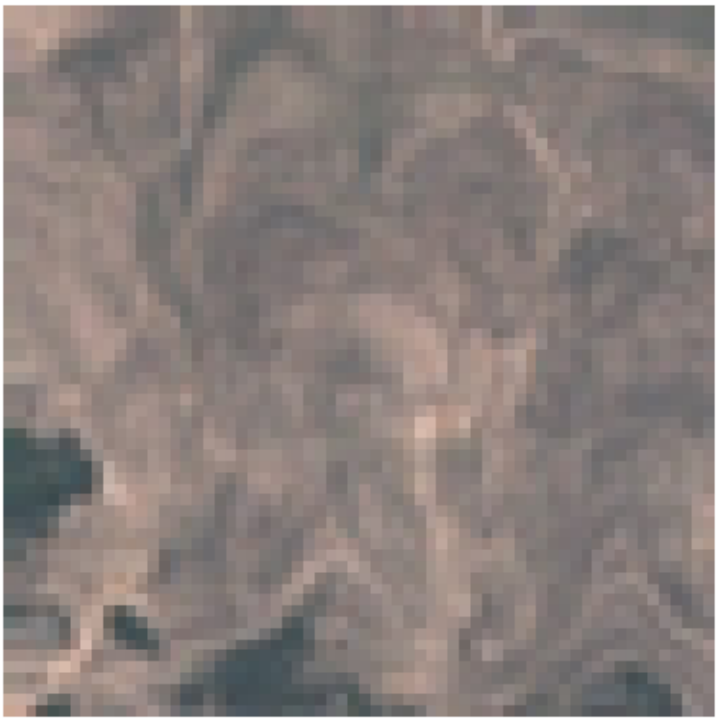
Filename: Residential_693.jpg
Predicted class: Residential



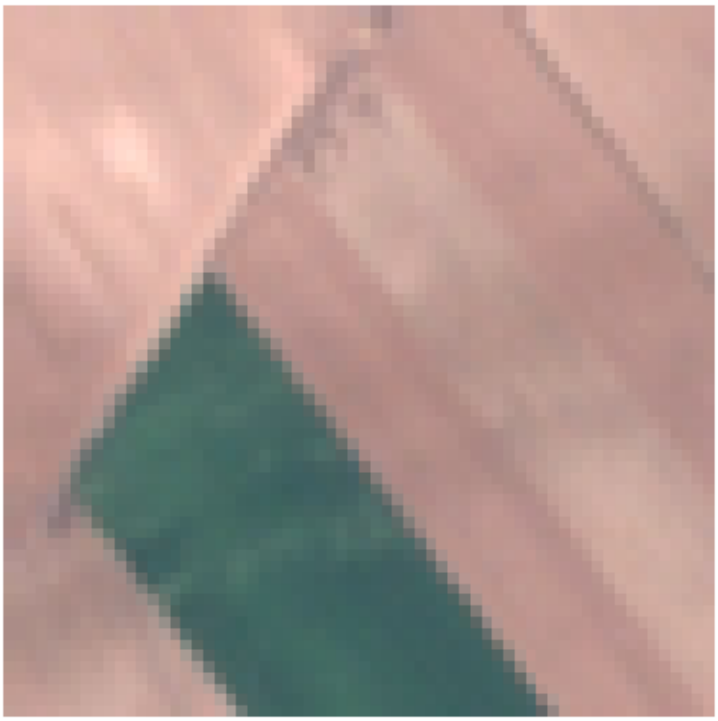
Filename: Forest_2907.jpg
Predicted class: Forest

Filename: SeaLake_910.jpg
Predicted class: SeaLake

Filename: HerbaceousVegetation_2986.jpg
Predicted class: HerbaceousVegetation



Filename: AnnualCrop_2983.jpg
Predicted class: AnnualCrop



Filename: Highway_6.jpg
Predicted class: Highway

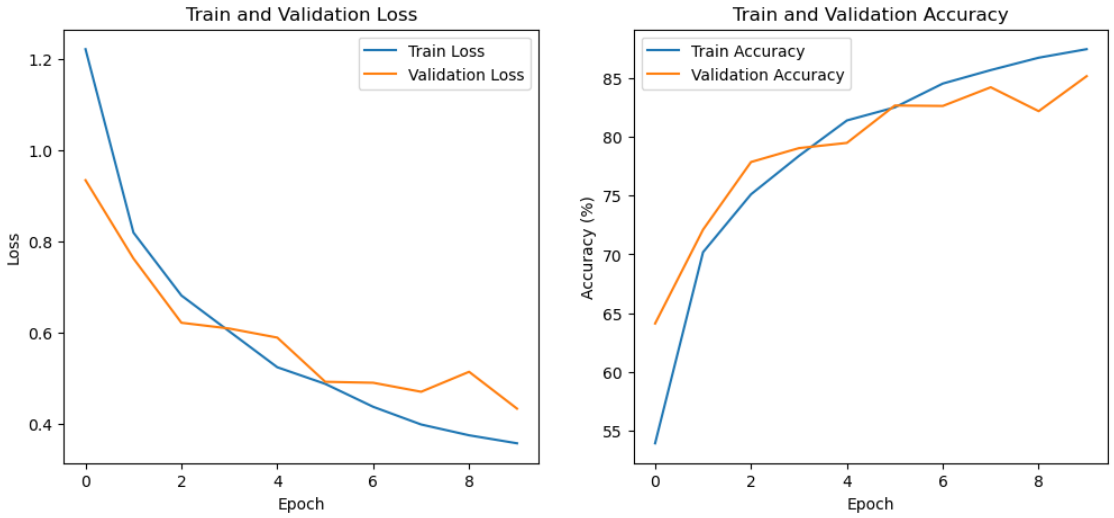
As shown in the pictures above, This instance of the model got all of the selected pictures correct.

```
Number of classes: 10

Epoch [1/10]
Train Loss: 1.2219, Train Accuracy: 53.90%
Validation Loss: 0.9348, Validation Accuracy: 64.11%
--------------------------------------------------
Epoch [2/10]
Train Loss: 0.8200, Train Accuracy: 70.19%
Validation Loss: 0.7633, Validation Accuracy: 72.11%
--------------------------------------------------
Epoch [3/10]
Train Loss: 0.6822, Train Accuracy: 75.12%
Validation Loss: 0.6224, Validation Accuracy: 77.87%
--------------------------------------------------
Epoch [4/10]
Train Loss: 0.6035, Train Accuracy: 78.39%
Validation Loss: 0.6098, Validation Accuracy: 79.06%
--------------------------------------------------
Epoch [5/10]
Train Loss: 0.5250, Train Accuracy: 81.41%
Validation Loss: 0.5899, Validation Accuracy: 79.50%
```

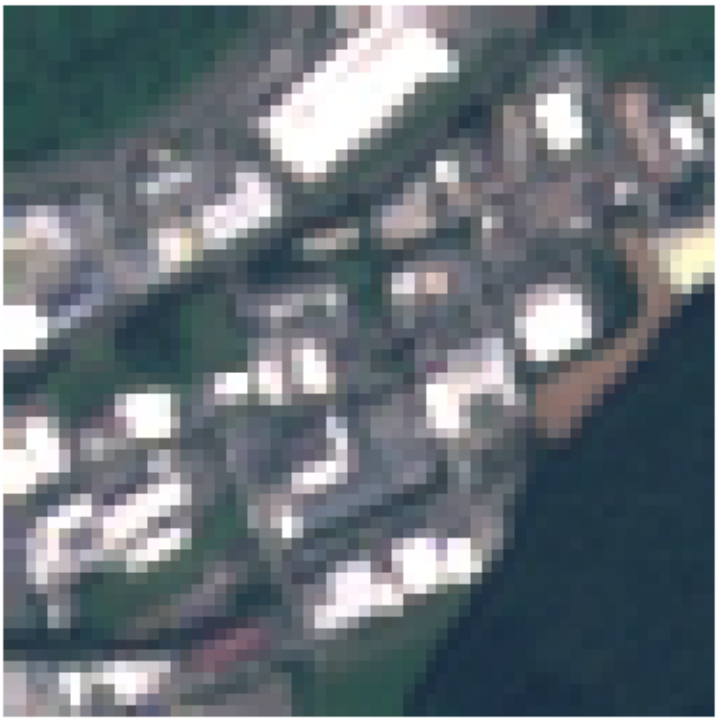
```
--------------------------------------------------
Epoch [6/10]
Train Loss: 0.4886, Train Accuracy: 82.53%
Validation Loss: 0.4931, Validation Accuracy: 82.69%
--------------------------------------------------
Epoch [7/10]
Train Loss: 0.4383, Train Accuracy: 84.56%
Validation Loss: 0.4910, Validation Accuracy: 82.65%
--------------------------------------------------
Epoch [8/10]
Train Loss: 0.3996, Train Accuracy: 85.70%
Validation Loss: 0.4713, Validation Accuracy: 84.24%
--------------------------------------------------
Epoch [9/10]
Train Loss: 0.3759, Train Accuracy: 86.76%
Validation Loss: 0.5150, Validation Accuracy: 82.20%
--------------------------------------------------
Epoch [10/10]
Train Loss: 0.3584, Train Accuracy: 87.49%
Validation Loss: 0.4343, Validation Accuracy: 85.19%
--------------------------------------------------
```

I decided to try to improve the model past the 90.20% with Data Augmentation and Learning Rate Schedule, the results did not come up to the inital run but it did lead to some good metrics.



As shown on the graph, the testing accuracy is following closely to the training accuracy which was not the case in the original attempt. If we increased the Epoch to more than 10 it will improve the predictability of the model substantially.

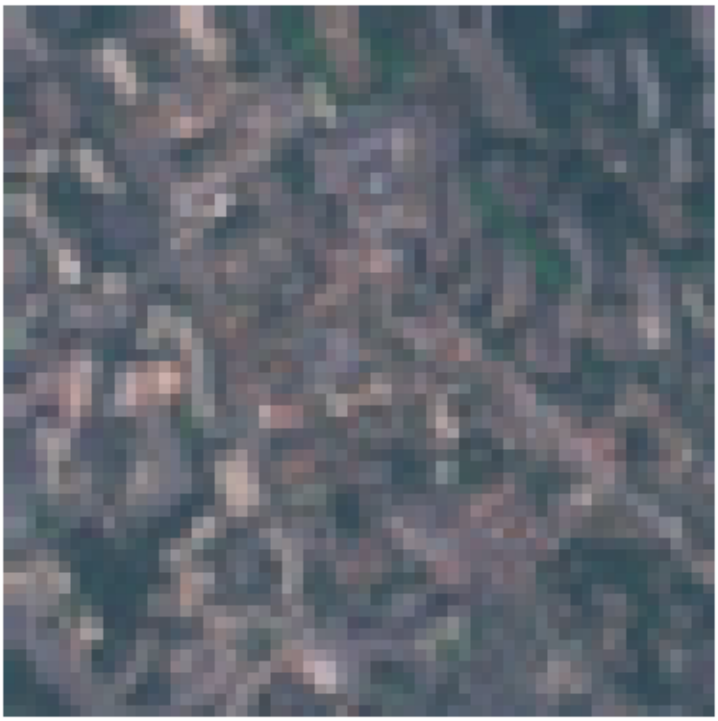
Filename: Industrial_110.jpg
Predicted class: Industrial



Filename: PermanentCrop_502.jpg
Predicted class: HerbaceousVegetation

Filename: Pasture_6.jpg
Predicted class: Pasture

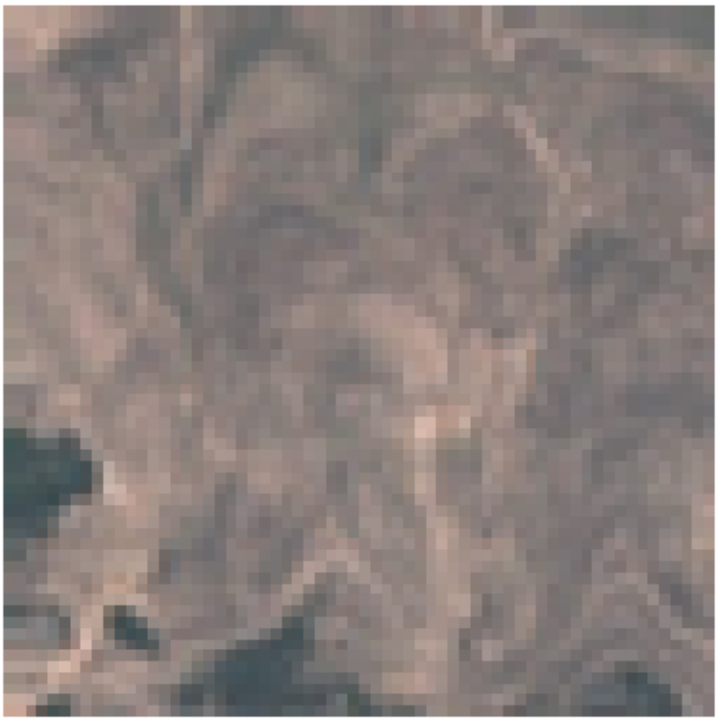
Filename: Residential_693.jpg
Predicted class: Residential



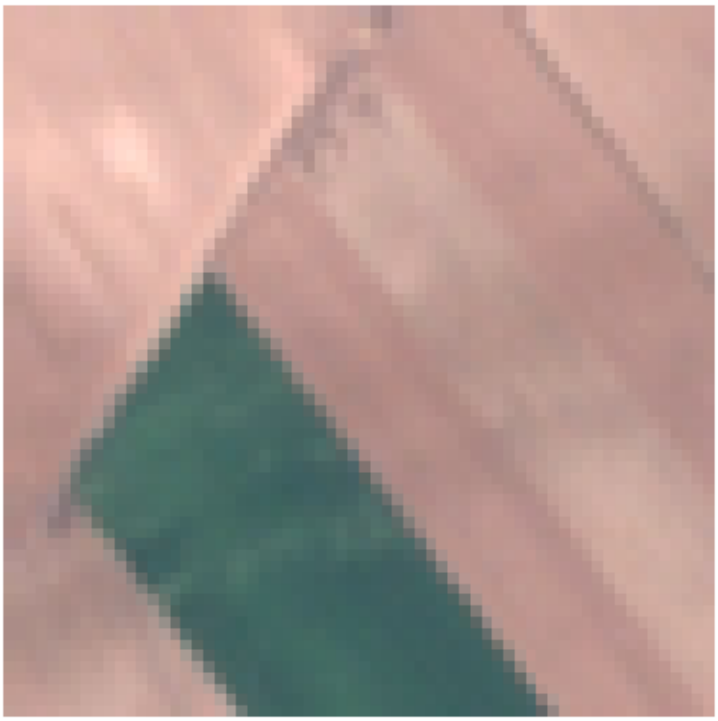
Filename: Forest_2907.jpg
Predicted class: Forest

Filename: SeaLake_910.jpg
Predicted class: SeaLake

Filename: HerbaceousVegetation_2986.jpg
Predicted class: HerbaceousVegetation

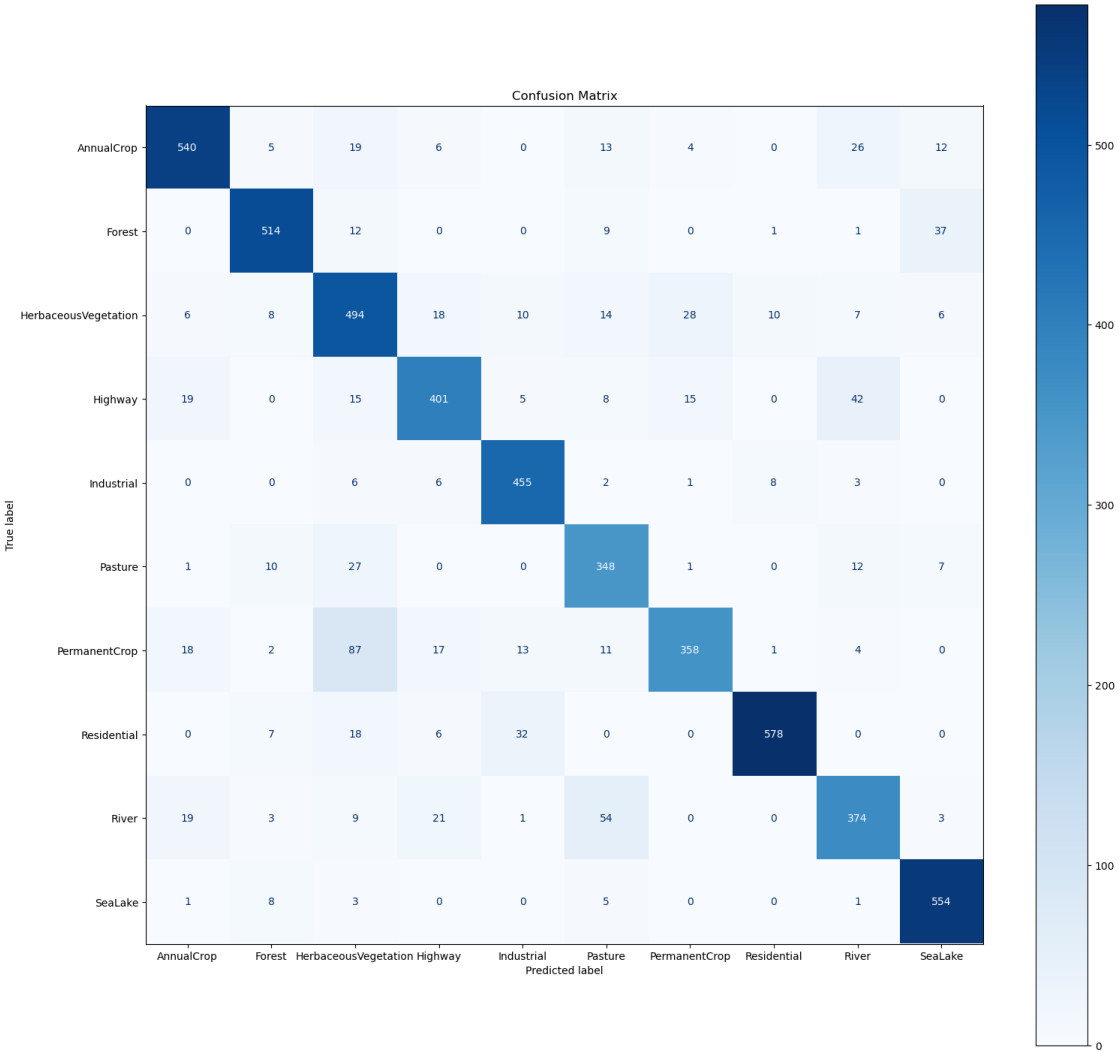


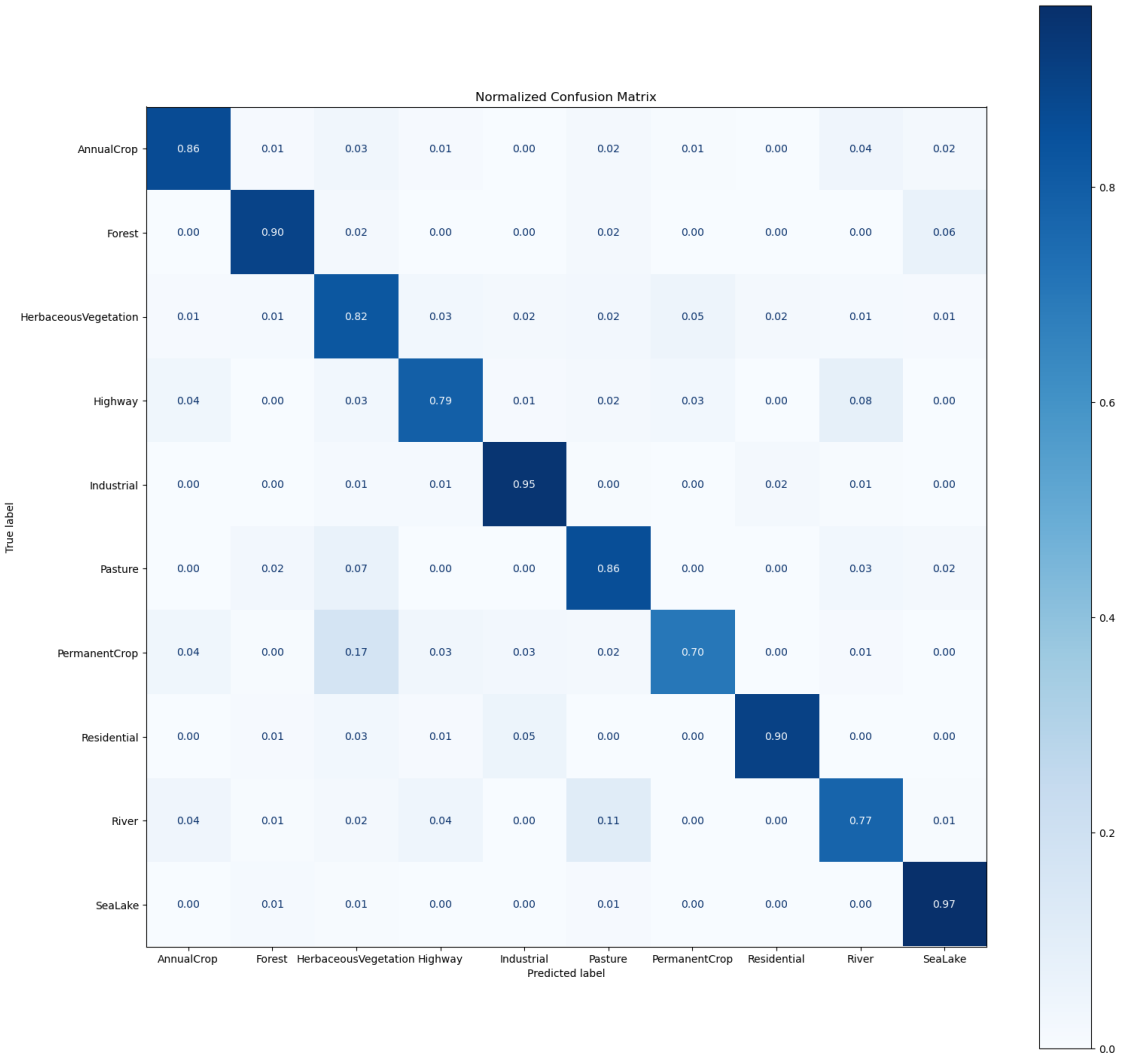
Filename: AnnualCrop_2983.jpg
Predicted class: AnnualCrop



Filename: Highway_6.jpg
Predicted class: Highway

Above are the pictures predicted. The model did get some wrong in this instance due to the lower performance of the training.


Confusion Matrix
AnnualCrop 540 5 19 6 0 13 4 0 26 12
Forest 0 514 12 0 0 9 0 1 1 37
HerbaceousVegetation 6 8 494 18 10 14 28 10 7 6
Highway 19 0 15 401 5 8 15 0 42 0
Industrial 0 0 6 6 455 2 1 8 3 0
Pasture 1 10 27 0 0 348 1 0 12 7
PermanentCrop 18 2 87 17 13 11 358 1 4 0
Residential 0 7 18 6 32 0 0 578 0 0
River 19 3 9 21 1 54 0 0 374 3
SeaLake 1 8 3 0 0 5 0 0 1 554
True label
AnnualCrop Forest HerbaceousVegetation Highway Industrial Pasture PermanentCrop Residential River SeaLake
Predicted label
500
400
300
200
100
0



Above you can see the conusion metrix from the analysis.