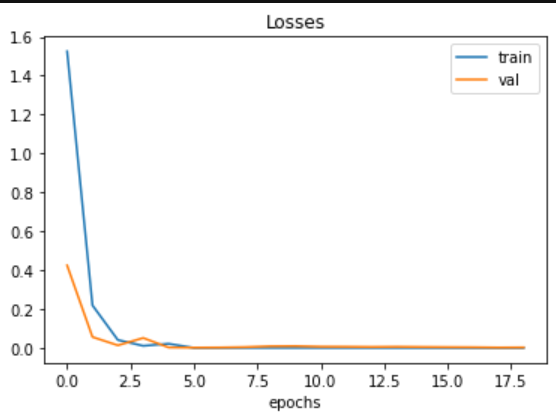
**Report**

For this task of classifying images into two classes: “fields” and “roads”. In training sets, I got 153 images divided into 2 classes not balanced (110 images for roads & 43 images for fields). I trained a simple CNN model with 3 convolutional layers and 1 fully connected to perform this classification. After testing it, I got a train accuracy of 86% (a not so satisfactory result), which I expected because there is not much data to train a CNN model and get successful result.

So, I adopted transfer learning to train data by implementing a pre-trained VGG16 model with a total of 16 convolutional layers (with 3x3 filters) in the "features" part, followed by 3 fully connected layers in the "classifier" part, for a total of 19 layers, and fine-tuned it by replacing the last fully connected layer with two output nodes instead of one.

I trained the model for 20 epochs using binary cross-entropy loss as the loss function and Adam optimizer with a learning rate of 0.001. I also applied data augmentation techniques like rotation, random horizontal flipping, normalizing, and resizing our dataset (242, 242, 3) to prevent overfitting.

Une image contenant graphique

Description générée automatiquementThe results in ***Figure 1*** show that our model achieved good accuracy on both the validation and test sets. After 6 epochs, the model achieved an accuracy of 100% on the training set and 100% on the validation set. The loss curves in ***Figure 2*** for both training and validation sets decreased consistently throughout the training process, indicating that our model is learning from the data and not overfitting.

**Figure 1: Accuracies**

**Figure 2: Losses**

To evaluate the performance of our model on a new dataset, I separated testing data into two fields (images 2,3,5,6,7,8 for “roads” and images 1,4,9,10 for “fields”). The test accuracy achieved a good result considering the limited amount of testing data (10 images) we had for this task.

Une image contenant carré

Description générée automatiquementI used a confusion matrix ***Figure 3*** to assess my model. We observe that the model is good at classifying “roads”, and it struggles with fields. It can be explained by the fact that the model learned to classify roads better than fields, and this is due to the unbalanced training data.

**Figure 3: Heatmap of confusion matrix**

Another tool used to evaluate the performance, classification reports. The precision for class “fields” (0.50) indicates that when the model predicted class “fields”, it was correct only 50% of the time. The recall for class “fields” (1.00) indicates that the model correctly identified all instances of this class in the data.

The precision for class “roads (1.00) indicates that when the model predicted this class, it was correct 100% of the time. The recall for class “roads” (1.00) indicates that the model correctly identified all his instances in the data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| 0.0 | 0.50 | 1.00 | 0.67 | 2 |
| 1.0 | 1.00 | 0.75 | 0.86 | 8 |
| Accuracy |  |  | 0.8 | 10 |
| Macro avg | 0.75 | 0.88 | 0.76 | 10 |
| Weight avh | 0.90 | 0380 | 0.82 | 10 |

**Table 1: classification reports**

Overall, the pre-trained VGG16 model with fine-tuning and data augmentation techniques proved to be effective in classifying images into two classes - Field and Road. The results show that our model achieved good accuracy on both the validation and test sets and is capable of correctly classifying new images as well.

**N.B**: In the training data, there were 2 road images (number 3&5) in the fields folder, so I transferred them to the roads field.