

Lab 2: Classifying Remote Sensing Images with Data Augmentation and Transfer Learning

In this lab, you will perform data augmentation and transfer learning yourself to classify remote sensing images.

Task 1: Dataset download and preparation (code already included in the template). In Google Colab, use `wget` to download the remote sensing image dataset from http://weegeevision.ucmerced.edu/datasets/UCMerced_LandUse.zip, and then convert the *.tif images into *.jpg

Task 2: Training and test data splitting (10 pts). Use `keras.preprocessing.image_dataset_from_directory` to load the data, and split the data into 80% for training and 20% for testing (set the image size to 96 x 96). Plot out any 9 images of your choice from the training data to give a preview of the dataset.

Task 3: Data augmentation (40 pts). Use at least four different ways of your choice to augment the training dataset. Feel free to set your own probabilities for different augmentation methods (e.g., you may prefer to apply one image processing method more frequently than another). After the augmented dataset is generated, concatenate it with the initial training data.

Task 4: Train a simple CNN using the augmented data (30 pts). You can design the CNN of your own choice but it should contain at least 2 convolutional layers, 2 max pooling layers, 1 dropout layer (check out this link: <https://www.tensorflow.org/tutorials/images/classification#dropout>), and 1 dense layer. Use 'adam' optimizer, sparse cross entropy loss, and accuracy for metrics. Train your model on the augmented dataset for 10 epochs. Evaluate the performance of your CNN model on the test dataset by measuring its accuracy.

Task 5: Apply Inception-ResNet V2 to do transfer learning on augmented data (20 pts).

Find Inception-ResNet V2 model on Tensorflow Hub:

https://tfhub.dev/google/imagenet/inception_resnet_v2/feature_vector/4 . Use the same optimizer, loss function, and accuracy setting, as used in Task 4. Train your transfer learning model on the augmented dataset for the same number of epochs. Evaluate the performance of this transfer learning model on the test dataset by measuring its accuracy. If you want to learn more about the model architecture of Inception-ResNet V2, you can read this Google AI blog: <https://ai.googleblog.com/2016/08/improving-inception-and-image.html>

To submit:

- Put your Google Colab link in UBLearns and then submit