Lab 6 Exercise - Reflections on transfer learning

1. Finetuning

The way I finetune the pretrained ResNet model includes 4 steps:

(1) Load the model with pretrained parameters: the pretrained model may include a well-trained set of parameters so that it performs well in extracting a variety of features (may include those we need) underlying in the images.

(2) Modify the model: replace the last layer, the original fully connected layer for classification, with a new one whose output unit number is equal to the number of class of our dataset—this is obvious.

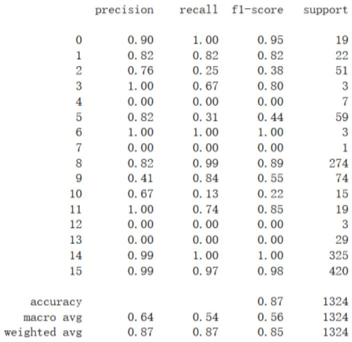
Additionally, for this application, the penultimate layer in the default network architecture, a 2D average pooling layer (from 7×7 to 1×1) will be not applicable since it drops the width-height ratio of activation out. Therefore, it is replaced by a 2-d adaptive average pooling layer where the activation shape will contain. Experimentally, it outperforms a 2-d adaptive max pooling layer by about 1% test accuracy after 30 epochs training.

(3) Freeze other parameters having been trained since we only need the last FC layer to be trained.

(4) Start training. With learning rate being 3×10^(-3), it turns out 82.7% accuracy in the training set and 78.5% test accuracy after 20 epochs. Since test accuracy begins to go down when training goes, this model has been a fully trained model.

2. Reflect on the two different approaches

The multiclass SVM I trained overwhelms the finetuned model: it ends up with 87.2% test set accuracy. Furthermore, it is extremely fast—172 times faster than the finetuned model. However, because of the high bias in the class distribution, it still does not do well to predict lesser-sample classes like how the finetuned model does.



```
              precision    recall  f1-score   support

           0       0.90      1.00      0.95        19
           1       0.82      0.82      0.82        22
           2       0.76      0.25      0.38        51
           3       1.00      0.67      0.80         3
           4       0.00      0.00      0.00         7
           5       0.82      0.31      0.44        59
           6       1.00      1.00      1.00         3
           7       0.00      0.00      0.00         1
           8       0.82      0.99      0.89       274
           9       0.41      0.84      0.55        74
          10       0.67      0.13      0.22        15
          11       1.00      0.74      0.85        19
          12       0.00      0.00      0.00         3
          13       0.00      0.00      0.00        29
          14       0.99      1.00      1.00       325
          15       0.99      0.97      0.98       420

    accuracy                           0.87      1324
   macro avg       0.64      0.54      0.56      1324
weighted avg       0.87      0.87      0.85      1324
```

Figure 1: SVM’s performance for each class