% CNN for object recognition

close all;

clear all;

clc;

outputFolder=fullfile('caltech101'); % used to create file path

rootFolder=fullfile(outputFolder,'101\_ObjectCategories');

categories = {'airplanes', 'ferry', 'laptop'};

imds=imageDatastore(fullfile(rootFolder,categories),'LabelSource','foldernames');

tbl=countEachLabel(imds)

minSetCount=min(tbl{:,2})

imds=splitEachLabel(imds,minSetCount,'randomize')

countEachLabel(imds)

airplanes= find(imds.Labels =='airplanes',1);

ferry= find(imds.Labels =='ferry',1);

laptop= find(imds.Labels =='laptop',1);

figure

subplot(2,2,1);

imshow(readimage(imds,airplanes));

subplot(2,2,2);

imshow(readimage(imds,ferry));

subplot(2,2,3);

imshow(readimage(imds,laptop));

net=resnet50();

figure

plot(net)

title('Architecture of ResNet-50')

set(gca, 'YLim', [150 170]);

net.Layers(1)

net.Layers(end)

numel(net.Layers(end).ClassNames)

[trainingSet, testSet]=splitEachLabel(imds,0.3, 'randomize');

imageSize=net.Layers(1).InputSize

augmentedTrainingSet=augmentedImageDatastore(imageSize,trainingSet,'ColorPreprocessing','gray2rgb');

augmentedTestSet=augmentedImageDatastore(imageSize,testSet,'ColorPreprocessing','gray2rgb');

w1=net.Layers(2).Weights;

w1=mat2gray(w1);

figure

montage(w1);

title('First Convolutional Layer Weight')

featureLayer='fc1000';

trainingFeatures=activations(net, augmentedTrainingSet, featureLayer, 'MiniBatchSize', 32, 'OutputAs', 'columns');

trainingLables= trainingSet.Labels;

classifier=fitcecoc(trainingFeatures, trainingLables,'Learner','Linear','Coding','onevsall','ObservationsIn','columns');

testFeatures=activations(net, augmentedTestSet, featureLayer, 'MiniBatchSize', 32, 'OutputAs', 'columns');

predictLabels=predict(classifier, testFeatures,'ObservationsIn','columns');

testLables=testSet.Labels;

confMat=confusionmat(testLables,predictLabels);

confMat=bsxfun(@rdivide, confMat, sum(confMat,2));

mean(diag(confMat));

A=imread(fullfile('test101.jpg'));

ds=augmentedImageDatastore(imageSize,A,'ColorPreprocessing','gray2rgb');

imageFeatures=activations(net, ds, featureLayer, 'MiniBatchSize', 32, 'OutputAs', 'columns');

label=predict(classifier, imageFeatures,'ObservationsIn','columns');

sprintf('The loaded image belongs to %s class',label)

B=imread(fullfile('test102.jfif'));

ds1=augmentedImageDatastore(imageSize,B,'ColorPreprocessing','gray2rgb');

imageFeatures1=activations(net, ds1, featureLayer, 'MiniBatchSize', 32, 'OutputAs', 'columns');

label1=predict(classifier, imageFeatures1,'ObservationsIn','columns');

sprintf('The loaded image belongs to %s class',label1)

C=imread(fullfile('test103.jpg'));

ds2=augmentedImageDatastore(imageSize,C,'ColorPreprocessing','gray2rgb');

imageFeatures2=activations(net, ds2, featureLayer, 'MiniBatchSize', 32, 'OutputAs', 'columns');

label2=predict(classifier, imageFeatures2,'ObservationsIn','columns');

sprintf('The loaded image belongs to %s class',label2)

**OUTPUT:**

tbl =

3×2 table

Label Count

\_\_\_\_\_\_\_\_\_ \_\_\_\_\_

airplanes 800

ferry 67

laptop 81

minSetCount =

67

ans =

3×2 table

Label Count

\_\_\_\_\_\_\_\_\_ \_\_\_\_\_

airplanes 67

ferry 67

laptop 67



ans =

'The loaded image belongs to airplanes class'

ans =

'The loaded image belongs to laptop class'

ans =

'The loaded image belongs to ferry class'

**FIGURES:**





