**Solutions:**

**Print the configurations of all the layers in Alexnet, or print the configurations of**

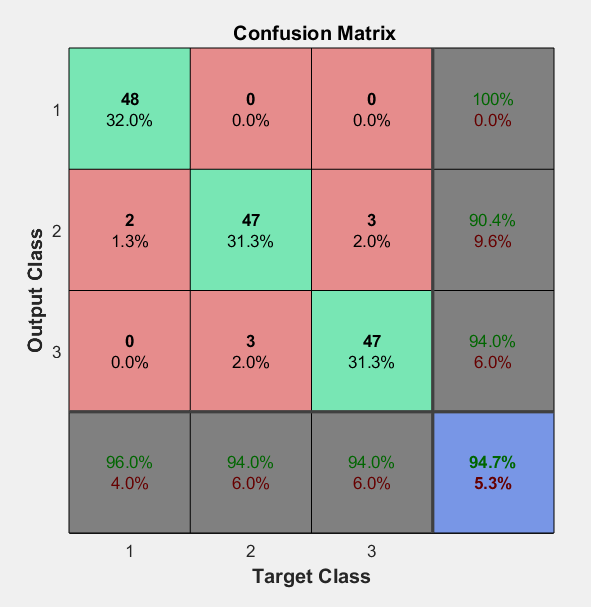
**all the layers in your own CNN.**

A screenshot of a cell phone

Description generated with very high confidence

**Print the confusion matrix of your classification result, and what is the accuracy**

**of classification result (it can be either on training data or test data)?**



websave('C:\Users\Sai\Desktop\saint thomas classes ppts\Machine Learning\imagenet-caffe-alex.mat',...

'http://www.vlfeat.org/matconvnet/models/beta16/imagenet-caffe-alex.mat');

% Load MatConvNet network into a SeriesNetwork

c = helperImportMatConvNet('C:\Users\Sai\Desktop\saint thomas classes ppts\Machine Learning\imagenet-caffe-alex.mat');

c.Layers %Display All CovNet layers

w = c.Layers(2).Weights;

w = mat2gray(w);

w = imresize(w,5);

figure, montage(w)

%% Set up image data

dataFolder = 'C:\Users\Sai\Desktop\saint thomas classes ppts\Machine Learning\Iris\_Imgs';

categories = {'setosa', 'versicolor','virginica'};

train = imageDatastore(fullfile(dataFolder, categories), ...

'LabelSource', 'foldernames');

train.ReadFcn = @(filename)readAndPreprocessImage(filename);

feaLayer = 'fc7';

trainFeatures = activations (c, train, feaLayer, ...

'MiniBatchSize', 32, 'OutputAs', 'columns');

x = train.Labels;

y = trainFeatures';

%[training, testSet] = splitEachLabel(trainingSet, 0.7, 'randomize');

% Train multiclass SVM

classifier = fitcecoc(y , x);

options = statset('UseParallel',true);

cvmdl = crossval(classifier,'KFold',10);

fprintf('kFold CV accuracy: %2.2f\n',1-cvmdl.kfoldLoss);

oofLabel = kfoldPredict(cvmdl,'Options',options);

ConfMat = confusionmat(x,oofLabel);

% Convert the integer label vector to a class-identifier matrix.

[~,grpOOF] = ismember(oofLabel,isLabels);

oofLabelMat = zeros(nLabels,n);

idxLinear = sub2ind([nLabels n],grpOOF,(1:n)');

oofLabelMat(idxLinear) = 1; % Flags the row corresponding to the class

[~,grpY] = ismember(x,isLabels);

YMat = zeros(nLabels,n);

idxLinearY = sub2ind([nLabels n],grpY,(1:n)');

YMat(idxLinearY) = 1;

figure;

plotconfusion(YMat,oofLabelMat);

h = gca;

h.XTickLabel = [num2cell(isLabels); {''}];

h.YTickLabel = [num2cell(isLabels); {''}];

**read and preprocessImage.m**

function Iout = readAndPreprocessImage(filename)

I = imread(filename);

% Some images may be grayscale. Replicate the image 3 times to

% create an RGB image.

if ismatrix(I)

I = cat(3,I,I,I);

End

% Resize the image as required for the CNN.

Iout = imresize(I, [227 227]);

end