TensorFlow skill testing project: Image Recognition in TensorFlow

Objective: Tackle an open problem in image recognition: the Google Street View Housing Numbers dataset.

Architecture: Your code will consist of three modules. This document contains the specification for the first module.

Module 1: `data_preprocessing.py`

This module should pre-process your data, which will come in the form of 3 X 32 X 32 images. This means the images are 32 by 32 pixels in size, and each pixel registers intensities in 3 different colour channels (red, green and blue, or RGB).

You'll have to figure out a few things here: first, how to normalize the pixel values (they go from 0 to 255, but you should consider converting this to a distribution with zero mean and unit variance). Second, you'll have to decide which makes more sense: to feed in all three RGB values to your neural net, to average the RGB values at each pixel before feeding them in, or to choose just one of those values (red, green **or** blue) to feed into your network. To figure this out, ask yourself how information about colour may or may not be relevant to the problem at hand.

This module should contain a function with the following signature:

`normalize(images) => normalized_images`

`images`	a list of images from the Google Street View House Numbers dataset, which you can find here: http://ufldl.stanford.edu/housenumbers/. **`images`** should be read in from `test_32x32.mat` or `train_32x32.mat`, the files containing the testing and training data.
`normalized_images`	the input images normalized to zero mean and unit variance, either having had their dimensionality reduced to 1 X 32 X 32 (due to averaging over RGB values, or from selecting one among R, G and B) or 3 X 32 X 32 (if you conclude that feeding in R, G and B makes more sense).
`normalize`	the function that normalizes the image pixel intensity values and returns the result

Your module can contain additional functions, but you must have **`normalize()`** somewhere in your script.

Once you've completed this exercise, send it to **@yazabi** and we'll give you feedback on your code!