Pixel Dawgs: Image Classifier 10/12/15

Nick Rosetti, Shreya Gupta, Satyanarayana Reddy, Manish Borse

## Data Collection

Flickr was used as a source for training images. Each image is automatically tagged by a machine learning algorithm. A training set of images and tags is publically available from Flickr’s servers. The set contains a text file which contains automatically generated tags along with an associated image id. There is another set of text files which contain an image id and an associated URL. Each person ran a script to parse the text containing image URLs and downloaded a total of ~300,000 images. Each image was saved with its associated id. The text file containing the tags is then split into three separate 5GB files so that a dictionary can be used to find an images tags. These tags are saved into a separate text file that contains images ids which are used for processing later.

## Data Analysis

The text file which contains image ids and tags is reformatted by using tags as an index. This means that each tag has an associated list of images. The top tags are calculated by rank and a selection of the top 0.5% is used to seed general tags. Each RGB image is converted to Hue Saturation Value space. A histogram for each image within a tag is calculated and a composite histogram is computed. This operation is completed for each tag. The composite histogram is written to a file by tag. This histogram is loaded back into memory to use to tag images.

In order to tag images, a list of valid images is loaded into memory. Histogram back projection is used to “back project” an image onto a histogram. This produces a resulting image that contains a per pixel representation of the probability that each pixel belongs to the tagged composite histogram. The current output of this algorithm is a 0 to 1 probability that the image is tagged correctly. A probability threshold will be used to select “valid” tags.

## Future Work

Current work represents a backbone for the image classification algorithm. There is an end-to-end method in place and algorithms need to be added to provide greater accuracy. Image Segmentation is an area which is vital to a functional image classifier. The next step is to insert a segmentation algorithm into the composite histogram generation. It is currently assumed that each tag represents an entire image. The selected algorithm to implement is K-Means clustering with histogram recombination analysis. Each image will be broken into N “like” layers and layers will be combined if they are similar enough. In order to assign tags to each layer manual input will be taken. Manual input is only required to create the initial tag histograms.