CSCI 5622 Project Proposal: Colorizing Grayscale Images

Prathyusha Gayam, Tarunianand M., Chandler Baggett, Rakesh Shivanand Margoor, Poorwa Hirve

1 Aim of the Project

Our project aims to build and analyze several models to convert grayscale images to color using Machine Learning techniques.

2 Motivation

Grayscale images are widely used in image processing. They scale the RGB colors to roughly one-third the size of the original image. Thus the storage of the grayscale image is less expensive. In this process, however, the original image is lost. Our project aims to reclaim the original image properties using Machine Learning.
An interesting application of our project is its role in black and white movies, as well as reclaiming the colors of black and white photographs. Subreddits like /r/colorization involve people who wish to colorize black and white photos as before colored photography and movies came into picture, only black and white images and films existed. In addition to these, MRI scans and satellite imagery are areas in which this can be applied. Also, it allows better interpretation of modern grayscale images such as those from CCTV cameras, astronomical photography, or electron microscopy. Manual way of colorizing images would take a lot of time and if it involves colorizing a video it would be a Herculean task. Automating the conversion of grayscale images to color using machine learning techniques would save a lot of time and effort.

3 DataSet

There are several datasets which have been used earlier to test the gray to color image conversion. Some of which are:

- Berkeley Segmentation Dataset which contains 100 test images and 200 train images:
  `https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html`
- ImageNet which contains millions of images organized in a hierarchical way.
  `http://www.image-net.org/`
- Computational Visual Cognition Laboratory which contains few hundred images in several categories.
  `http://cvcl.mit.edu/database.htm`

All these datasets are publicly available. Initially, we would be using the Berkeley Segmentation dataset for all our modelling purpose. If we are able to bring a good model, we would probably use some part of the Imagenet database for better performance of the model as it contains large training data.

4 Machine Learning Methods

Predicting the actual color of an image automatically, involves necessary learning from different train images. The learning can be done by using different features of an image such as object detection, RGB channels etc which are explained more in the next section. Hence building a model which predicts the color would definitely involve supervised learning. Based on literature survey and research, we think some of the popular supervised machine learning algorithms that can be applied to this model are discussed below.

- Linear Logistic Regression
  We would use a linear logistic regression model with a One vs. All approach as opposed to true multinomial regression. This would provide us with a good baseline. [1]

- SVM
  We can use an SVM classifier in order to predict the color of an input image. The core principle would be to build a One vs. All multilabel classification where the labels would be the possible colors for a pixel and the input would be a pixel's feature. We could run the classification on all labels and choose the best probable color for the given pixel. [2, 3]

- CNN
  We want to train our model in an end to end fashion by applying a CNN architecture which will learn both local and global priors. We also want to design the model which could process an image of any resolution. CNN architecture layers - Normalization, Convolution and Pooling layers

- K means clustering usually involves grouping several parts of your data (in this case your image) based on its similarities to a centroid feature. This centroid feature is mostly predetermined. This can yield new results for every run as they are randomly determined (using the same initial parameters). To overcome this, we can use an initially seeded randomizing function, and end up with similar results for every run.[4]
  The value of k, is an important factor here, in the sense that this will determine how many features similar to the centroid can be determined.
  For our task of image conversion, K means clustering will help to cluster parts of an image with similar brightness and hues. This in turn leads to edge detection,, making it easier to bring out finer details in the image (again, this would depend on the value of k). The issues we will face here are determining the initial centroids and getting an optimal value of k to recalculate all further values of the centroid.

5 Image Features

The probable features in an image that we will be considering are:

- Edges:
  Edges denote the boundary (edge) between to image regions. The shape, smoothness or gradient value of the edge would be a feature.

- Texture Recognition
  Texture is characterized by repetition of basic elements. Local Binary Patterns (LBP) descriptors can be used to extract texture information. [5]

- Hue-Saturation-Value
  It is the closest model used in image analysis which is similar to the way humans perceive colors. [6]

- SIFT and SURF
  Scale-Invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF) are object detection algorithms used for images of different scales and rotations. [7, 8, 9]

6 Evaluation of Results

One of the probable evaluation metric for verifying the correctness/accuracy of the model is to compare the pixel values between the actual output and the expected output. The closeness of the actual to the original color can be quantified using L2 norms of the differences between their pixels. Another approach could be the ratio of matched pixels with the total pixels. The matching can be defined by considering a threshold in the differences.

Evaluating an unlabeled data is quite ambiguous, as there is no expected output to verify the correctness. In this scenario, the aesthetic sense of colorization comes into the picture. If a user is unable to relate the color and the content, then we can say that the model did not classify the pixels appropriately. The same approach has been used by others (scholars) in their research for evaluation of their model.

References

[1] `https://cs.uwaterloo.ca/~zfrenett/CS886-Project.pdf`

[2] `http://cs229.stanford.edu/proj2015/163_report.pdf`

[3] `http://cs229.stanford.edu/proj2013/KabirzadehSousaBlaes-AutomaticColorizationOfGrayscaleImages.pdf`

[4] Zarezadeh, A & Alizadeh Ghazijahani, Hamed & Judi, Arash. (2012). Color Image Segmentation with K-Means Clustering and Edge Detection Methods.

[5] Local Binary Patterns. Wikipedia, Wikimedia Foundation, 25 Nov. 2017

[6] Bear, Jacci Howard. Understanding the HSV Color Model. ThoughtCo, 13 Jan. 2018, www.thoughtco.com/what-is-hsv-in-design-1078068.

[7] Scale-Invariant Feature Transform. Wikipedia, Wikimedia Foundation, 28 Feb. 2018

[8] Speeded up Robust Features. Wikipedia, Wikimedia Foundation, 28 Feb. 2018

[9] http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_surf_intro/py_surf_intro.html

[10] http://hi.cs.waseda.ac.jp/~iizuka/projects/colorization/en/