Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

Mid Term Report

on

"AUTOMATIC COLORIZATION OF GRAYSCALE IMAGE"

COMP 484

(For partial fulfillment of 4th Year/ 1st Semester in Computer Engineering)

Submitted by:

Nischal Bhandari (Roll No. 10)

Babin Khatri (Roll No. 26)

Amar Kumar Mandal (Roll No. 58)

Neha Verma (Roll No. 59)

Aarush Timalsina (Roll No. 61)

Submitted to:

Dr. Bal Krishna Bal

Department of Computer Science and Engineering

Submission Date:

24th August, 2022

Introduction

Image colorization is the process of adding colors to a grayscale picture using as a source a colored image with similar content. Colorization techniques are widely used in astronomy, MRI scans, and black-and-white image restoration. Given a grayscale image, there is no unique correct colorization in the absence of any information from the user. Even though many implementations require the user to specify initial colors in certain regions of the picture, our project focuses on automatic image colorization, without any additional input from the user. In this report we describe our first attempts at colorizing a gray image using a similar colored picture. We begin by implementing a simplistic method that transfers colors between pixels based on the similarity in their luminosity. We improve our approach by considering a larger feature space that takes into account the provenance of each pixel and strives for spatial consistency. Then we apply PCA and KNN in the feature space, followed by a confidence voting that labels our gray pixels and determines from which colored pixels we have to transfer the chromatic channels.

Methods and Experiments

Our colorization algorithm is as follows: Our datasets consist of test and training examples that consist of color images of varieties of an object. These objects will be the input to our model. We begin by converting the given image into a n-dimensional numpy array where each pixel point on the image is matched to its corresponding RGB value.

We then normalize the image between the range of [0-1] using min-max scaling. After the image is converted into its equivalent numerical value, we then set up the Sequential model (i.e model that feeds our inputs sequentially from Input to Output). In the next steps, we design our Convolution Neural Network (CNN) model with a multi-layers hidden unit. In the hidden layers, the summed activation of the node is then transferred to the activation function (RELU and Tanh) which then outputs the results.

At this stage, the output given by the model is not very accurate, so it must go through multiple iterations to reach the optimal level of accuracy. For this purpose we tend to

minimize the value of the loss Function. Since we want to predict the value in a continuous range(color code) we use Mean Squared Error as our loss function. To optimize the loss function we use a gradient-based optimization technique called **rmsprops** which was proposed by Geoffrey Hinton (Father of back-propagation). This normalization balances the step size (momentum), decreasing the step for large gradients to avoid exploding and increasing the step for small gradients to avoid vanishing. Another benefit of using rmprops optimizer is that it uses an adaptive learning rate instead of treating the learning rate as a hyperparameter. This means that the learning rate changes over time.

Finally, the model is compiled and is prepared for training and testing.

So far we have done the following experiments.

1. Compiled the test and training datasets required for our model. As proposed, we had used Berkeley Segmentation Dataset BSDS 300 (https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/) and collected some similar images manually.
2. Convert our images to its corresponding-dimensional array in order to be fed in our model.
3. Completed the initial phase of our model development.

Design of upcoming experiments

Till now we have been working on specific classifications of our data samples. Strictly speaking, we have been training and validating our model using images of people only. Thus moving forward we will be training our model using all types of the images and evaluating our model based on the whole dataset.

- We plan to complete the final stage of model development.

  Rather than relying on a series of independent pixel-level decisions, by transferring color from a segmented example image (training example), we can account for the higher-level context of each pixel. In the training stage, the

luminance channel of the reference image along with the accompanying partial segmentation were provided as a training set for a supervised learning algorithm. After iteratively training the model, the algorithm has been ready to be tested with test datasets.

- Test the model using our Test Datasets.

  After training the model, we will be further testing and validating the model with test datasets to get the idea of how accurate our model is.

- If desired accuracy is obtained. We will save the model.

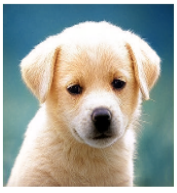
  After the above steps, if we get satisfactory accuracy of colorized images, then we begin to finalize the model to develop the API.

- To make it usable, we plan to develop an API.

  We will then develop an API for our model to make interaction between the colorizing model and other applications possible by fetching and retrieving images. Rather than for the feature, we will be doing this to connect with the frontend.

- And integrate it with the front end so that everyone can use it.

  The final step of our project will be to integrate with the frontend so that other users can easily use the application by uploading two images: one with grayscale image and other with similar colored image and obtaining a colorized grayscale image. The platform will be built upon the foundation of our model combined with user friendly frontend web pages as shown below.



Plan of upcoming experiments

Following are the activities each project members are assigned to for the upcoming experiments and activities.

Member's Name	Work
Nischal Bhandari	- Validating the model.
Babin Khatri	- Training the model using the training examples consisting of different categories of images.
Amar Kumar Mandal	- Evaluating and adjusting model accuracy.
Neha Verma	- Develop an UI for the colorization of the greyscale images - Testing the trained model
Aarush Timalsina	- Develop an API and integrate it with the front end - Testing the model and enhancing the model's accuracy

References

- *Automaticcolorizationofgrayscaleimages - Stanford University*. (n.d.). Retrieved August 24, 2022, from http://cs229.stanford.edu/proj2013/KabirzadehSousaBlaes-AutomaticColorizationOfGrayscaleImages.pdf
- *Colorize pictures: Turn black and white photos to color with ai*. Hotpot Design. (n.d.). Retrieved August 24, 2022, from https://hotpot.ai/colorize-picture
- The berkeley segmentation dataset and benchmark. (n.d.). Retrieved August 24, 2022, from https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/