CSE616 Project

Automatic Image Colorization from Greyscale to RGB

Using convolutional neural networks

Name: Robeir Samir George

ID: 1901823

Introduction:

In image colorization, our goal is to produce a colored image given a greyscale input image. The problem in this task is that a single grayscale image may correspond to many colored images. As a result, traditional models often relied on significant user input alongside a grayscale image.

Recently, deep neural networks have shown remarkable success in automatic image colorization going from grayscale to color with no additional human input. This success may in part be due to their ability to capture and use semantic information in colorization.

This project implements a deep convolutional neural network for automatic colorization, the problem of converting grayscale input images into colored images. The model is based on the ResNet-18 classifier and trained on the MIT Places database of landscapes and scenes.

The Problem:

We aim to infer a full-coloured image, which has 3 values per pixel (lightness, saturation, and hue), from a grayscale image, which has only 1 value per pixel (lightness only). For simplicity, we will only work with images of size 256×256, so our inputs are of size $256 \times 256 \times 1$ (the lightness channel) and our outputs are of size $256 \times 256 \times 2$ (the other two channels).

Rather than work with images in the RGB format, as people usually do, we will work with them in the LAB colour space (Lightness, A, and B). This colour space contains exactly the same information as RGB, but it will make it easier for us to separate out the lightness channel from the other two (which we call A and B). We'll make a helper function to do this conversion.

We'll try to predict the colour values of the input image directly (we call this regression).

Dataset:

We are going to use subset of MIT Places dataset (containing places, landscapes, and buildings). You can download this dataset from this link.

You can go directly to MIT Places page through this link: MIT Places Database for Scene Recognition

Dataset is split into 90% training, and 10% test (validation).

Papers:

The work in this project is mainly based on those two papers:

[1603.08511] Colorful Image Colorization (arxiv.org)

[PDF] Let there be color! | Semantic Scholar

Also, I found this paper very useful to take its work into consideration: Combining Deep Convolutional Neural Networks with Markov Random Fields for Image Colorization

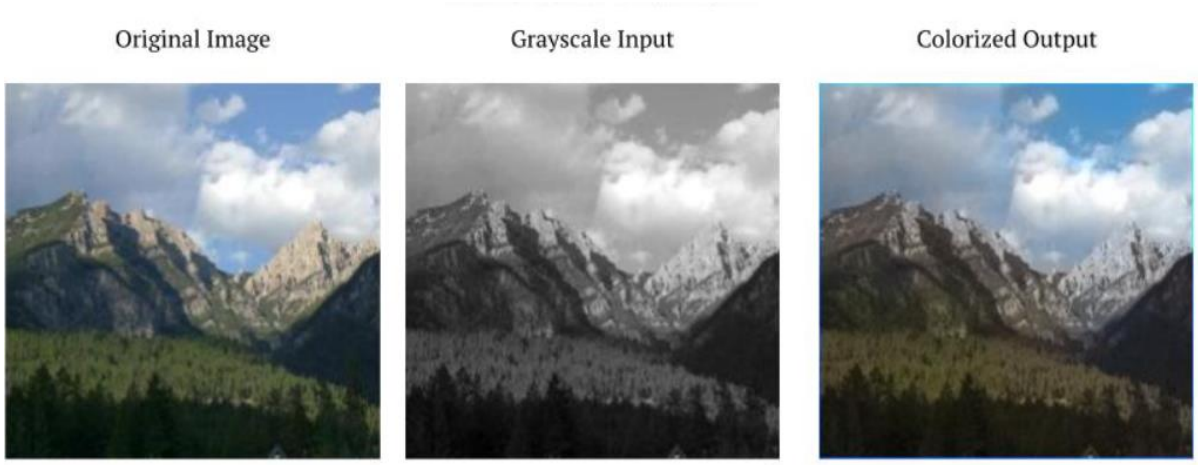
Overview:

In our project, we build and train a deep convolutional neural network for automatic image colorization, the task of generating a colored image from grayscale input image.

We reproduce and build upon recent CNN colorization work by the first paper in the papers section.

The third paper introduces using a Markov random Field-based model for inferring a final colored image from pixel-level color distributions.

First of all, given the brightness channel of image (a 1 x H x W input), the objective is to infer the corresponding chrominance and Hue channels (a 2 x H x W output). Image colorization poses a significant challenge to traditional computer vision techniques because a single black and white image may have a multitude of colorizations.



Model:

First, we conv neural network to directly predict the color channels to an input grayscale input image, The backbone of our network is the ResNet-18 classification network - an image classification network with 18 layers and residual connections- which we trained on grayscale images.

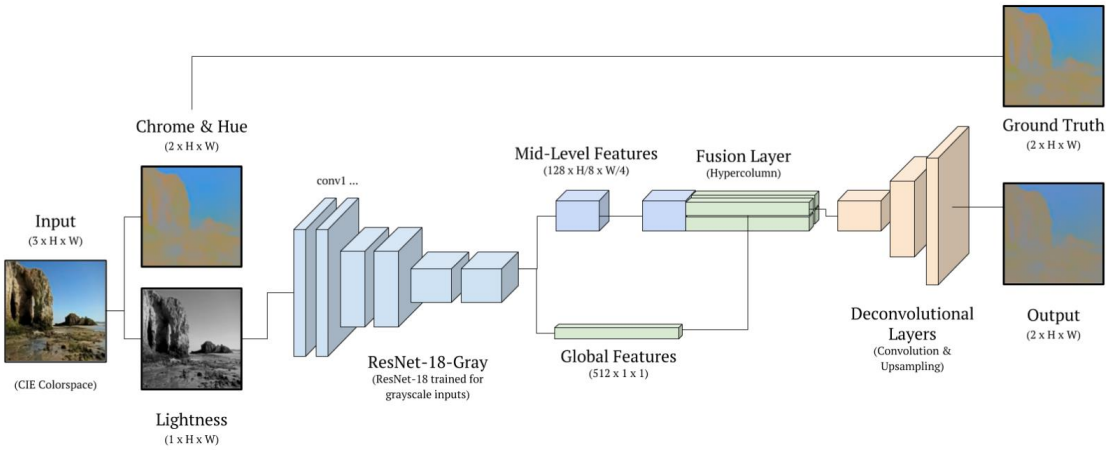
We first apply a number of convolutional layers to extract features from our image, and then we apply deconvolutional layers to upscale (increase the spacial resolution) of our features.

We will modify the first layer of the network so that it accepts grayscale input rather than colored input, and we will cut it off after the 6th set of layers.

Our model is based upon the network proposed in the first paper by Iizuka et al. but takes advantage of the feature extracting abilities of the ResNet classifier rather than training feature extractors from scratch.

Our network is an end-to-end network consists of:

- Classification network for grayscale images.
- Fusion layer for combining mid-level and global features (a learned linear combination of features)
- Colorization network for computing pixel-by-pixel colour (chroma and hue) distributions.
- Markov random field with mean field variation inference for computing the final colorization.



CNN+MRF Model: Second, we modified our network to output a distribution of color values for each pixel in the image rather than a single-color value. We infer the final color values from this distribution using a Markov Random Field.

Loss Function:

Since we are doing regression, we'll use a mean squared error loss function: we minimize the squared distance between the color value we try to predict, and the true (ground-truth) color value.

This loss function is slightly problematic for colorization due to the multi-modality of the problem.

For example, if a gray dress could be red or blue, and our model picks the wrong color, it will be harshly penalized. As a result, our model will usually choose desaturated colors that are less likely to be "very wrong" than bright, vibrant colors. There has been a recent research on this issue, but we will stick to our simple loss function for today.

Optimizer:

We will optimize our loss function (criterion) with the Adam optimizer.

Training:

We trained our model on Google Co-lab session for 100 epochs.

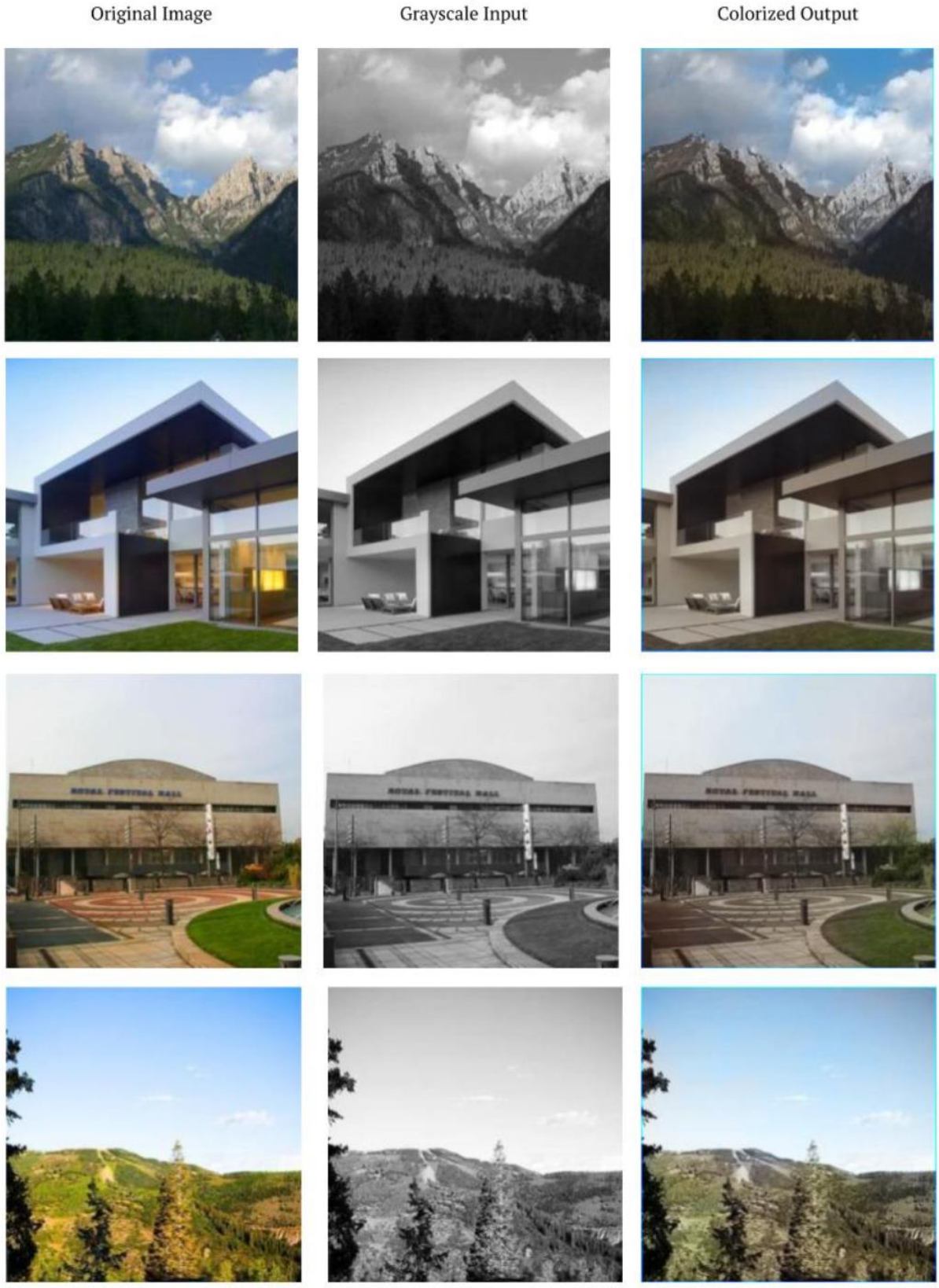
Code:

All code and files can be found on the following github repo:

RobsGeorge/ImageColorization-Deep-CNN (github.com)

Results:

Some of the results of the trained network, can be found in the following figures:



End Of Project …. Thanks 8 May 2022