|  |  |
| --- | --- |
| A picture of a winding road and trees  Image Colorization  BAN 676 – Deep Learning | Submitted By:  Uttara Dabbiru (py4642)  Priyanka Shah (mt8964)  Under The Guidance Of:  Dr. Surendra Sarnikar |

Table of Contents

[INTRODUCTION 2](#_Toc89360664)

[DATA DESCRIPTION AND DATA PREPARATION 2](#_Toc89360665)

[MODEL DESCRIPTION 2](#_Toc89360666)

[ALTERNATIVE MODEL ANALYSIS 3](#_Toc89360667)

[MODEL OPTIMIZATION 4](#_Toc89360668)

[IMPLEMENTATION 5](#_Toc89360669)

[DISCUSSION 6](#_Toc89360670)

# **INTRODUCTION**

One of the most exciting applications of deep learning is colorizing black and white images. This task needed a lot of human input and hardcoding several years ago but now the whole process can be done end-to-end with the power of AI and deep learning.

Recoloring and restoring old photos are a painstaking process when done manually through some photo editing software. Coloring black and white images with deep learning has become an impressive showcase for the real-world application of neural networks in our lives.

Auto encoders and GANs are some of the techniques that provide solution for restoring and recoloring of old photos.

# **DATA DESCRIPTION AND DATA PREPARATION**

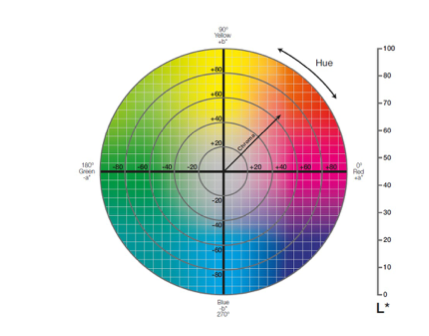
Data is a combination of different set of classes collected from Kaggle and combined. Data set contains 2797 images and the classes chosen are Cars, Persons and Flowers.

# **MODEL DESCRIPTION**

**Autoencoders**

In image colorization what we plan to do is taking a grey scale image and supplying a corresponding color image and training the algorithm / autoencoder and creating a model that learns how to colorize images.

By far, the most popular way of representing images is RGB. But there are also different approaches available, and one of them is Lab color space. The Lab color space expresses colors as 3 values:



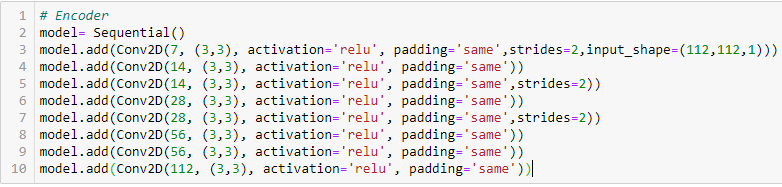
L starts for lightness. how light is the image. In L channel the image is in greyscale level., with values ranging from 0(black) to 100(white).

A channel is represented along the x-axis which goes from Green to Red color spectrum with values ranging from -128(green) to 127(red).

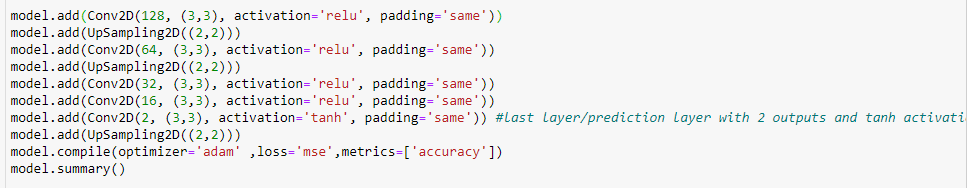
B channel is represented along the y-axis and goes from blue to yellow color spectrum, with values ranging from -128 (blue) to 127(yellow).

We have used Auto encoders to artificially colorize the grey scale images. Autoencoders reduce the dimensionality of the input data. They are built from two components:

**Encoder:** transforms the input data into a lower-dimensional representation. To achieve this goal, the encoder must learn only the important features of the data.



**Decoder**: recovers the input from the low-dimensional representation.



* model = Sequential() loads the network. The input shape is 112\*112\*1 (This is the L channel that goes in).
* The Activation function used here is relu.
* The last prediction layer has 2 outputs each corresponding to A and B layers.
* Since the A and B layers values ranges from -1 to 1. We use the activation function tanh in the prediction layer as it has the range between -1 to 1.
* UpSampling comes at the decoder stage because once the image goes down to 14\*14 we need to upscale them back to 112\*112
* Finally giving us an output of (112,112,2) which means A and B channels are reconstructed.

In Auto encoders we fit the model with (x, y) where x is the input from L channel and y is from A and B channel.



# **ALTERNATIVE MODEL ANALYSIS**

**Pix2Pix GAN**

There are other deep learning models such as auto encoders, convolution neural networks and GAN to implement this problem. GANs are an exciting and rapidly changing field, delivering on the promise of generative models in their ability to generate realistic examples and in generating photorealistic photos of objects, scenes, and people that even humans cannot tell are fake.

The input to our GAN model will be a black and white image and the output will be a colorized image. It is fantastic how a model is going to learn that a particular object needs to be colored yellow or green.

GAN makes generative models having two neural networks compete with each other. There are two major components:

**Generator:** It turns noise into an imitation of data to try to trick discriminator. The input of the generator is original greyscale image and generated output is RGB image. It has encoder decoder structure symmetrically. In our model, RGB images converting into greyscale image which acts as labels. For generator, minimizing loss means improves itself to some extent to fool discriminator. We are using encoder decoder where encoder takes greyscale image and produce latent dimension (Bottleneck) and decoder produce RGB image by enlarging latent dimension.

**Discriminator**: It tries to identify real and fake data. The discriminator is used generated RGB image and original RGB image to identify fake or real image. For discriminator, maximizing loss means classifying generated images accurately.

* Used LeakyRelu and Relu activation functions in Generator and Discriminator respectively.
* Used loss functions MeanSquaredError for Generator and binary\_crossentropy for Discriminator.
* Calculated Generator and Discriminator losses
* Used Optimizer Adam with learning rate 0.0005

Graphical user interface, application

Description automatically generated

# **MODEL OPTIMIZATION**

**RGB to Lab:**

A close-up of a flower

Description automatically generated

**Auto encoders without batch normalization:**

With 100 epochs the Model accuracy is 54%. Total parameters in training the model are 338,714. Below is the inference image without Batch normalization.

Chart

Description automatically generated

**Auto encoders with batch normalization:**

With 100 epochs the Model accuracy has improved to 59%. Total parameters in training the model are 339,974. Below is the inference image without Batch normalization.

Chart

Description automatically generated

The results improved by adding batch normalization to the model

# **IMPLEMENTATION**

The model has been implemented by developing a web app using streamlit. By uploading the image file to the file uploader, it can be observed that the resized output is displaying with traces of color.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

# **DISCUSSION**

Color is the primary factor responsible for making a photo feel exciting, lively, mysterious or perhaps melancholic. Coloring an image can help tell us story visually and it can be used to communicate on an emotional level.

By using this project, we can convert old B/W photos of grandparents into colorized photo making that more memorable. We can extend this project by colorizing the videos too as well as include various filters unlike greyscale.

# **RUBRICS**

**Meet Expectations Model :**

Autoencoders (analyzed 2 models including typical hyperparameters, batch normalization and identified which model is doing good by giving the better accuracy and better image output)

**Exceed Expectation model :**

* Developed Pix2Pix GAN model.
* Built Web app using streamlit.