PES University, Bangalore

(Established under Karnataka Act No. 16 of 2013)

MAY 2020: IN SEMESTER ASSESSMENT (ISA) B.TECH. IV SEMESTER

UE18MA251- LINEAR ALGEBRA

MINI PROJECT REPORT

ON

Colourization of Black and White Images using Neural Networks

Submitted by

1. Name: Sreejesh Saya SRN: PES1201800293

Branch & Section: Computer Science and Engineering, 4th Semester, I section

---

PROJECT EVALUATION

(For Official Use Only)

Sl. No.	Parameter	Max Marks	Marks Awarded
1	Background & Framing of the problem	4	
2	Approach and Solution	4	
3	References	4	
4	Clarity of the concepts & Creativity	4	
5	Choice of examples and understanding of the topic	4	
6	Presentation of the work	5	
	Total	25	

Name of the Course Instructor:

Signature of the Course Instructor:

Colorization of Black and White Images using Neural Networks

Sreejesh Saya

Department of Computer Science

and Engineering

PES University, Bangalore, India

saya.sreejesh@gmail.com

Abstract- **We present a Neural Network based system that attempts to colorize black and white photographic images without any human interference. We explore various network architectures, colour spaces. The final model we build, generates colorized images that are aesthetically pleasing to the human eye, demonstrating the ability of computers to bring life to grayscale images using colour that would otherwise be a tedious task even for human experts. We propose a fully automatic approach that produces near true and realistic colours.**

Keywords: Neural Networks, Deep Learning, Colorization of Images

I. INTRODUCTION

Colouring a grayscale image is easy task for even the most expert of humans. A human can easily recognize an object and attempt to recall its true colour from just looking at it, even in a black and white image, thus the human mind is free to hallucinate several plausible colours for said object (Excluding the complexities such as shadows, highlights, contrast etc.). This high level comprehension is precisely why the development of a fully automatic colorization algorithm still remains a challenge to this day. Automatic colorization serves as a proxy measure for our visual understanding. Our work makes this connection explicit; we use a type of a deep Neural Network architecture driving advances in image classification and object detection.

There are different ways of colouring an image using Deep Learning, including Automatic and User Guided, as well as video colorization. Automatic Colorization relies completely on the Neural Network to identify the colour and shadows, contrast etc. of a certain object present in the Black and White image. User guided Image Colorization takes on a different approach where the Neural Network is provided a colour-palette. The Neural Network identifies the objects present in the image and applies the appropriate colour to that object. This is useful in colouring manga comics and small sketches, where a large colour-palette is often not used. This paper dives into Automatic Colorization of human faces obtained from various sources online.

Black and White images can be represented in grids of pixels. Each pixel has a value that corresponds to its brightness. Colour images consist of three layers: a red layer, a green layer, and a blue layer. Take the example of a green leaf on a white background. Intuitively, we might assume that the leaf is present only in the green layer. But as shown in Fig 1.0, we observe that the leaf is in fact present in all three channels. The layers not only determine colour, but also brightness. To achieve the colour white, for example, we would need maximum value of all colours. By adding an equal amount of red and blue, we can make the green brighter. Thus, a colour image encodes the colour and the contrast using three layers: RGB. This is the standard colour space used widely.

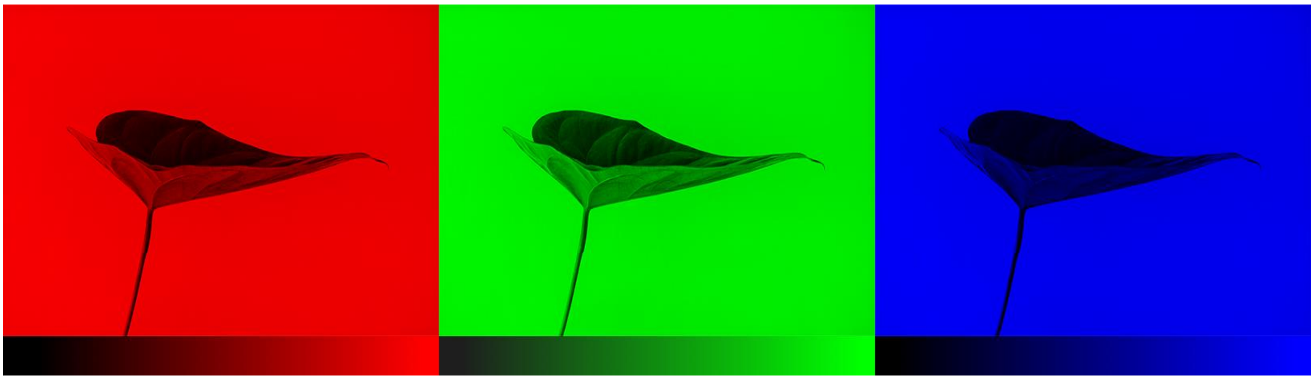


Fig 1.1 – The leaf is present in all three colour channels

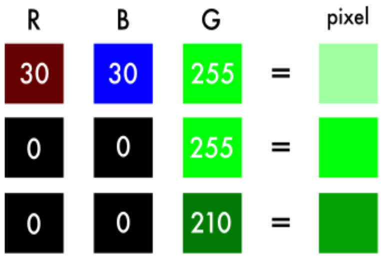


Fig 1.2 - Adding equal amounts of blue and red makes the green layer brighter

E layer in a colour image has a value from 0 to 255 (in 8-bit colour format). The value 0 means that is has no colour in this layer. If the value is 0 for all colour channels, then the image pixel is black.

II. RELATED WORKS

We derive our core idea from Zhang, R., Isola, P., & Efros, A. A. [1] i.e. using a popular colour space known as Lab over the standard RGB. Obtaining a dataset for training is very easy: any colour photo can be used as a training example, simply taking the L – Lightness channel as the input to the Neural Network and predicting the '**a**' and '**b**' channels. They observed that previous works of attempts to colour photos using Convolutional Neural Networks have resulted in de-saturated photos since they made use of loss functions that encouraged conservative predictions inherited from standards regression problems.

Futschik, D. [2] wrote a detailed paper regarding the structure of Neural Networks that deal with the automatic colorization method. He included multiple layers, more than we could include, in his Neural Network for detecting characteristics in an image to provide accurate colouring. Futschik heavily relied on Zhang's research paper [1]. We observed his approach of constructing a Neural Network and applied to ours as well.

Iizuka, S., Simo-Serra, E., & Ishikawa, H [3] conducted a different approach to that of others. Their Convolutional Neural Network was a complex model consisting of multiple layers, far more than what we could implement. This method of theirs, proved to output quite convincing results. In brief, they constructed a classifier that when fed a black and white image, would begin to classify various part of the image and output predicted labels. These labels were passed to the colorization network in the fusion layer along with the grayscale image which then coloured the image appropriately referring to the labels which resulted in convincing results.

Larsson, G., Maire, M., & Shakhnarovich, G. [4] used the RGB colour space instead of the favoured Lab colour space observed so far. They used architectures for sematic

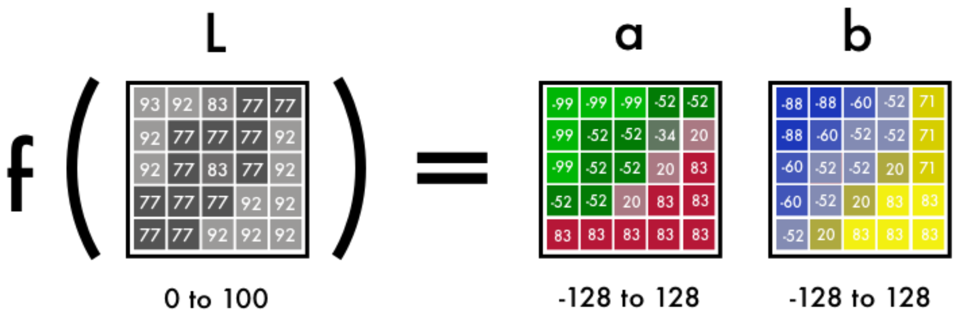


Fig 2.0 – Function that takes input, a grayscale image, and outputs the 'ab' layers of Lab colour space

segmentation and edge detection. They used a CNN as well, but their processing turned out to be quite costly. Their method was quite inoperable for our paper. They processed each pixel of the image separately and even with a bilinear interpolation technique, it required significant memory. This caused them trouble to use batches of images. They experimented with various output colour spaces and loss functions that were well beyond the scope of this paper.

Cheng, Z., Yang, Q., & Sheng, B. [5] dwelled deep into the colouring techniques for colouring black and white images that are usually implemented by human experts. They used feature descriptors to extract each pixel of the image and using multiple high level, returns an output that is the chrominance of the corresponding pixel which was passed to a joint bilateral filter to reduce noise. This was well beyond the scope of our approach. Their approach resulted in a much more appealing coloured image when compared to some of the state-of-the-art colouring techniques.

III. DATASET

Our model is trained on over a 1000 human faces extracted from various sources. This dataset consists of different skin colours, facial structures, angles, hair colour, providing a wide range of faces for the CNN to train itself. These images are 256x256 in size and are 8-bit RGB colour space which is adequate for our testing. The Neural Network once trained, is tested on 30 images that were not included in the training set to evaluate the performance.

IV. APPROACH

As we know, a Neural Network creates a relationship between an input value and output value. To be more precise with our colorization task, the network needs to find the traits that link grayscale images with coloured ones. In sum, we are searching for the features that link a grid of grayscale values to the two colour grids of the Lab colour space as shown in Fig 2.0. We start by making a simple version of our Neural Network to colour an image of a

Fig 3.1 – The left image is the grayscale image, the centre is the Neural Network result, the right is the true image

Fig 3.2 – Leftmost is the L layer, middle – '**a'** layer, rightmost – '**b'** layer

woman's face. This way, we can get familiar with the core logic of our model as we add features to it. Once our Neural Network is trained on the woman, we evaluate it using the same image. The results are as shown in Fig 3.1. Fig 3.2 show the Lab colour space for the same image.

A. *Colour Space*

Before feeding the image to our Neural Network, we need to change our colour space. Since, RGB requires three layers which can become complex, we use a popular colour space known as Lab. **L** – stands for lightness, and '**a'** and '**b'** represent the colour spectrums green-red and blue-yellow.

As shown in Fig. 3.1, a Lab encoded image has one layer for Lightness, this actually is our grayscale image that will be fed into our CNN, and we have packed three layers i.e. RGB into two layers '**a'** and **'b'.** This means that we can use the original grayscale image in our final prediction that is merge it with the output of the CNN to get our coloured image. Hence, we have only two channels to predict. It is a well-known fact that 94% of cells in our eyes determine brightness. That leaves only 6% of our receptors to act as sensors for colours. In the above fig., we observe that the grayscale image is a lot sharper than the colour layers. This is why we need to keep the grayscale image in our final prediction.

The last two layers in Fig 3.1 represent the colour spectrum red-green and blue-yellow. When these three layers are merged, we get our original image.

B. *B&W to Colour*

Our final prediction looks like Fig 4.0. We have a grayscale image that will be fed into the CNN which we want to predict two colour layers, the ab in **Lab.**

To turn one layer into two layers, we use convolutional filters. They are used to highlight or remove something to extract information from the picture. The network will either create a new image from a filter or combine several filters into one image.

For a convolutional Neural Network, each filter is automatically adjusted to help with the intended outcome. We'll start by stacking multiple filters and narrow them down into two layers, the '**a'** and '**b'** layers.

C. *Feature Extractor (Layers)*

Our Neural Network finds characteristics that link grayscale images with their coloured version. We look for simple pattern: a diagonal line, all black pixels and so on. We look for the same pattern in other parts of the image and remove if they don't match. We generate new images from our mini filters. To further understand the image, we decrease the image size by half. By combining our new image with lower level filters we can detect more complex patterns such as a dot, a line. We extract the patterns from the image again. This time we get 128 new images. After passing the image through multiple filters, the Neural Network begins to identify low level features such as edges and corners. This process is like most Neural Networks that deal with vision, known as Convolutional Neural Network. Convolution is basically combining several filtered images to understand the context in the image.

D. *Explanation*

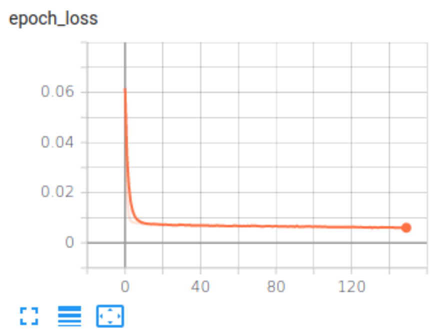
The input is a grid representing a black and white image. It outputs two grids with colour values. Between the input and output values, we create filters to link them together. We convert an image from RGB colour space to Lab colour space. The black and white layer is our input and the two coloured layers must be the output.

We map the predicted values and the real values within the same interval. This way, we can compare the values. To map the predicted values, we use a Tanh activation function. The true colour values for the Lab colour space range from -128 to 128. Dividing this by 128 gives us the interval [-1, 1]. After calculating final error, the network updates the filters to reduce total error. We repeat this until error is as low as possible.

The main difference from other visual networks is the importance of pixel location. In colouring networks, the image size or ratio stays the same throughout the network. In other networks, the image gets distorted the closer it gets to the final layer.

The max-pooling layers in classification networks increase the information density, but also distort the image. It only values the information, but not the layout of an image. In colouring networks we instead use a stride of 2, to decrease the width and height by half. This also increases information density but does not distort the image. Two further differences are up-sampling layers and maintaining the image ratio. Classification networks only care about the final classification. Therefore, they keep decreasing the image size and qualify as it moves through the network.

V. RESULTS



To train the model, we used batches of images of size 10. This means that the model will train over 10 images at a time until all the all the images are exhausted, per epoch. We trained the model for 1500 epochs since our model showed no improvement after around the 1500th epoch.

Fig 4.0 – Leftmost image is the grayscale image, the centre is the true image, and the rightmost image is the image produced by our model

The results are quite accurate. The Neural Network was able to detect certain features of the face and differentiate the foreground and the background. It was able to accurately colour the skin based on the lightness of grayscale image.

Fig 4.0 is shown. The leftmost image is the grayscale image that is fed into our model. The centre image is the true/original image from which we extract the grayscale image. The rightmost image is the final result after combining the output of the model with the grayscale image to get a colour image. As you can see, the model accurately detects characteristics such as hair, background, skin-tone and appropriately colours the image.

The skin-tone of the female face is pretty close to the true image. The advantage here is that, the entire process of colouring her face took about less than a second for the CNN whereas it would hours for a human to replicate.

Fig 5.1 – True Grayscale image of a sportsman

As shown in Fig 5.1 above, we have a grayscale image of a sportsman, the image is a black and white image in its true form, meaning that the image did not have any values in the '**a**' and '**b'** colour space of the Lab colour space. This was a true test of our Neural Network model.

Once the Neural Network processed the image, it returned an image, the **ab** part of our Lab colour space. It is not clearly visible to us since there is no brightness associated with these layers. This image will be combined with the grayscale image so that the lightness from the grayscale image fuses with the '**ab**' output producing colour. Since, these two images are matrices, they can easily be combined, serving as layers 1, 2 and 3 in the output matrix. Layer 1 represents L – Lightness, the 2nd and 3rd layers represent the red-green and blue-yellow layers.

Since this is a matrix representation of the image, we must convert it into a format that can be displayed easily. We use the lab2rgb function available in the sci-kit python package. Once converted, the final result is as shown in Fig 5.2 below. The model was able to detect the face accurately and attempted to colour the skin appropriately.

Fig 5.2 – Final result of our model. Producing a coloured image of the sportsman

VI. CONCLUSION

The image in Fig 5.2 is a convincing image of the sportsman. The average human was not able to conclude that it was an image coloured by a model.

As mentioned, this entire process took less than a second for our model to process. Compared, it would take an expert human, multiple hours, with the help of software such as Adobe Photoshop to even match the quality of our model let alone provide better quality. With an advanced iteration of our model, we can even colorize grayscale videos and bring life to movies made in the Black and White Film era for our pleasure.

REFERENCES

[1] Zhang, R., Isola, P., & Efros, A. A. (2016, October). Colorful image colorization. In European conference on computer vision (pp. 649-666). Springer, Cham.7

[2] Futschik, D. (2018). Colorization of black-and-white images using deep Neural Networks.

[3] Iizuka, S., Simo-Serra, E., & Ishikawa, H. (2016). Let there be color! Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. ACM Transactions on Graphics (ToG), 35(4), 1-11

[4] Larsson, G., Maire, M., & Shakhnarovich, G. (2016, October). Learning representations for automatic colorization. In European Conference on Computer Vision (pp. 577-593). Springer, Cham.

[5] Cheng, Z., Yang, Q., & Sheng, B. (2015). Deep colorization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 415-423).