Image Colorization using Neural Networks

Abhay tomer ◻ Sonakshi Singh ◻ Aman Kumar ◻ Avisheek Bhowmick

Apex Institute of Technology, Chadigarh University

Abstract— Image colorization is inherently an ill-posed problem with multi-modal uncertainty for colorization of gray-scale abstract images. In this paper, we trained 3 models on different sets of images viz. abstract geometrical images and abstract fluid gradient images. Then we compared these models to a model trained on both sets of images.

Keywords:— Image Colorization, Abstract Images, Autoencoders, Neural Networks.

1 Introduction

Colored images contain more information than grayscale images. They are clearer and appeal more to the viewers. Colored images contain three channels about the color of the image known as RGB which means RED, GREEN, and BLUE. Although the gray image is 1-Dimensional as it contains only gray. Colored image to be converted to grayscale. But if the opposite is true, it is not an easy task. The reason for this is that many colors can lead to a single gray level but if we go backwards, we cannot determine which color corresponds to the particular gray level we are trying to convert to. This is not a viable solution but efforts can be made to create a solution to this problem. In many cases, color correction may be needed to minimize the difference between images in order to further the process.

The main purpose of Colorization is to make the gray images attractive to the viewer as the colors are added to fit the photo essence. The coloring process works to color each pixel in a gray image with a specific color. Coloring techniques can be divided into three types: hand color, in which a great human effort is required. In particular, photo editing software such as Paint or Adobe Photoshop are used to convert a gray image to a color image but this process is time-consuming. The second type is a color-based scribble and color-based modeling. The scribble-based methods used in this genre require a lot of effort from the user to add a good scribble to gray images. The problem of adding gradient color to a gray image is not the correct method. The proposed method is a completely automated method. We suggest using the reference color image to help transfer colors from the index image to the gray image. The reference image is converted to a Lab color space, while the grayscale image is adjusted according to the light channel L. The gray image corresponds to both channels a and b, before being converted to an RGB image.

2 Methodology

In this paper, we will train 3 CNN, first one on abstract geometrical images. Second one will be trained on abstract Fluid gradient images and the third one will be trained on both i.e., abstract and gradient images. Then we will be comparing the result to the original images and also against each other to find out if a model trained on both image sets could potentially perform better.
The models will be trained using autoencoder. The autoencoder will be trained on 350 images over 150 epochs (indicates the number of passes of the entire training dataset the machine learning algorithm). However, the third model will be trained on 700 images over 100 epochs.

2.1 Autoencoder

It is a specific type of neural network in which both input and output are the same. The purpose of the autoencoder is to learn lower-dimensional representation for high-dimensional data, especially by reducing the dimensionality, by training the network to capture the most important parts of the input image(figure1).

2.1.1 Encoder

Encoder transforms the input into a low-dimensional latent vector.

2.1.2 Bottleneck

It contains the compressed knowledge representations of the data that is being fed.

2.1.3 Decoder

It reconstructs the input from the latent vector as much as possible.

2.2 CIE LAB Color Space

CIE lab also knowns as LAB color space has 3 channels where L stands for lightness ranging from 0(black) to 100(white), A stands for red/green value [green(-ve) to red(+ve)] and B stands for blue/yellow [blue(-ve) to yellow(+ve)].

$$L^{\star} = 116\, f\left(\frac{Y}{Y_{\mathrm{n}}}\right) - 16$$

$$a^{\star} = 500\left(f\left(\frac{X}{X_{\mathrm{n}}}\right) - f\left(\frac{Y}{Y_{\mathrm{n}}}\right)\right)$$

$$b^{\star} = 200\left(f\left(\frac{Y}{Y_{\mathrm{n}}}\right) - f\left(\frac{Z}{Z_{\mathrm{n}}}\right)\right)$$

Using lab color space we can easily get gray-scale images as in the first channel only L(lightness) is there. It also reduces color channels to only 2 (green-red, blue-yellow) with respect to RGB colorspace which has 3.

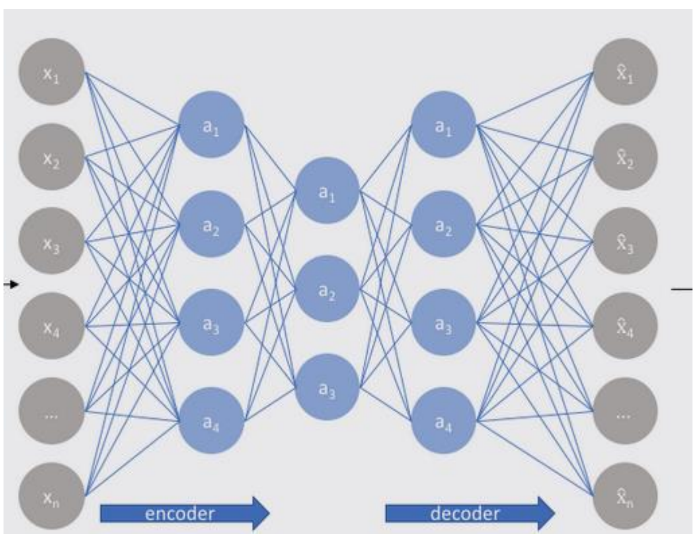

Figure 1: internal visualization of an autoencoder. (image credit: https://www.v7labs.com/blog/autoencoders-guide#autoencoders-intro)

3 Results

The Graphs between The Training loss and Validation loss decreases significantly reaching as low as 0.04. The opposite can be said about Graphs of Accuracy where the highest accuracy achieved is **0.822.**

3.1 Model V1

This model was trained extensively over 150 epochs on geometrical abstract images

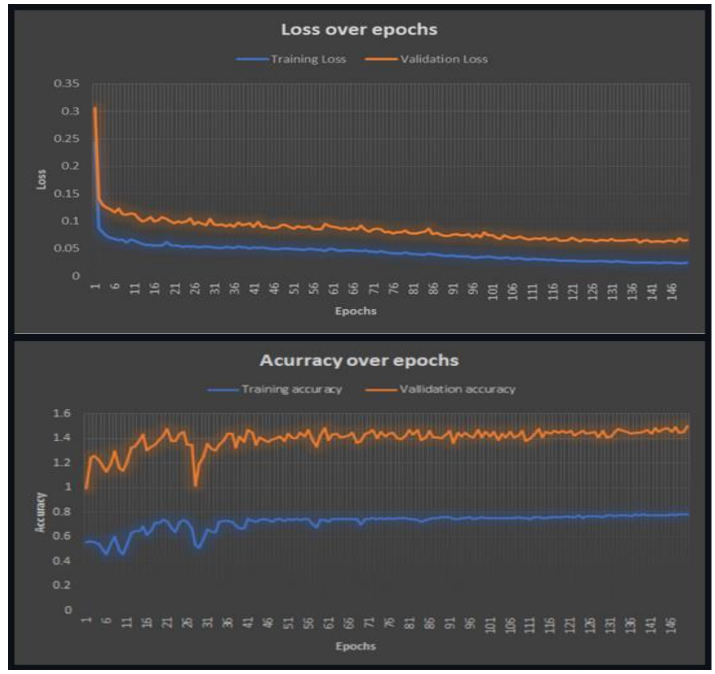

Figure 2: Training and Validation, Loss and Accuracy over epochs of Model V1.

3.2 Model V2

This model was trained extensively over 150 epochs on Fluid gradient abstract images.

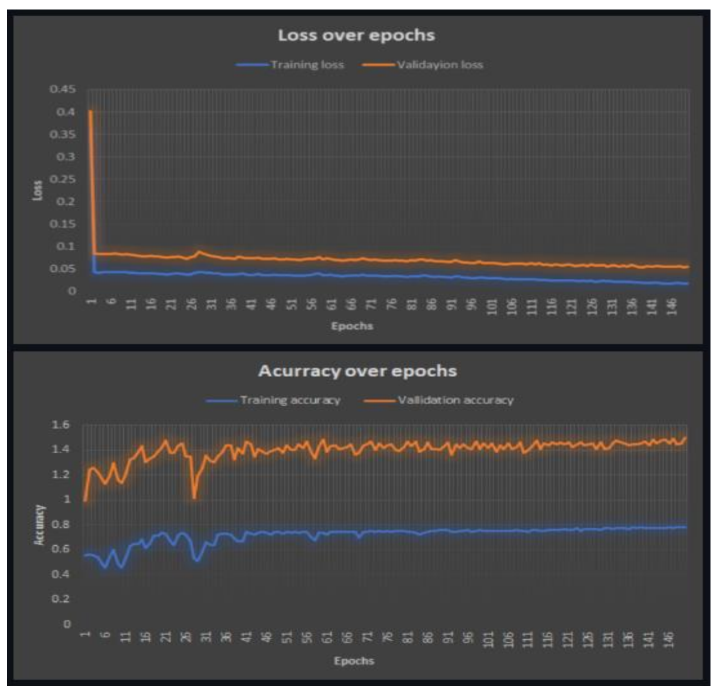

Figure 3: Training and Validation, Loss and Accuracy over epochs of Model V2.

3.3 Model V12

This model was trained extensively over 100 epochs on Fluid gradient as well as geometrical abstract images.

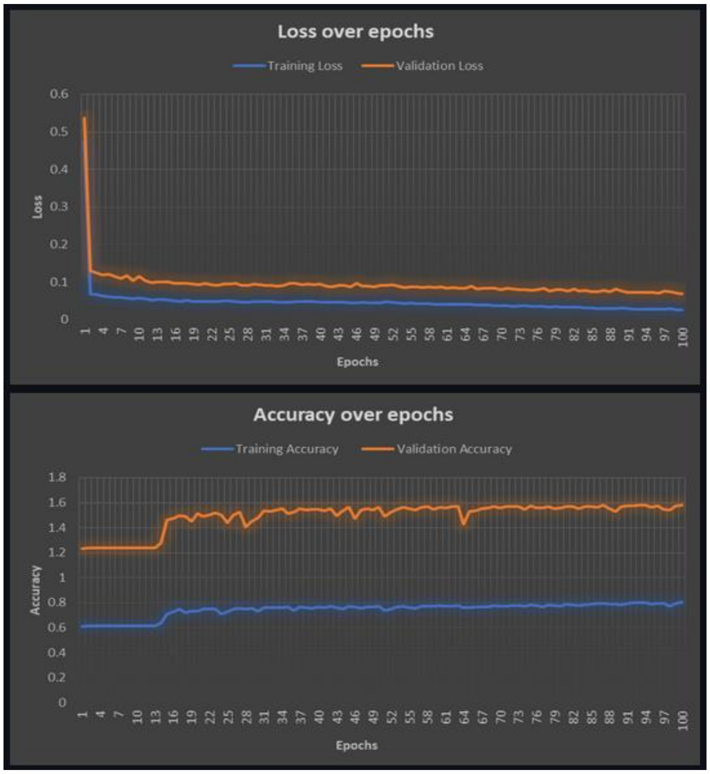

Figure 4: Training and Validation, Loss and Accuracy over epochs of Model V12

4 Comparison

If you compare the images from the different models to the original images you will be able to observe that all the trained models do an excellent job in reproducing the images from greyscale to colorized images.

Model V1 does good job coloring geometrical images as it was trained on geometrical images yet is able to produce acceptable results when exposed to fluid images.
Model V2 does an outstanding job in colorization of Fluid gradient images but on the contrary when subjected to geometrical abstract images the results produced are displeasing.
Model V12 being trained on both the data sets proves excellent in producing fluid gradient images but lacks finesse when colorizing abstract geometrical images. But provides a better result than model V2.

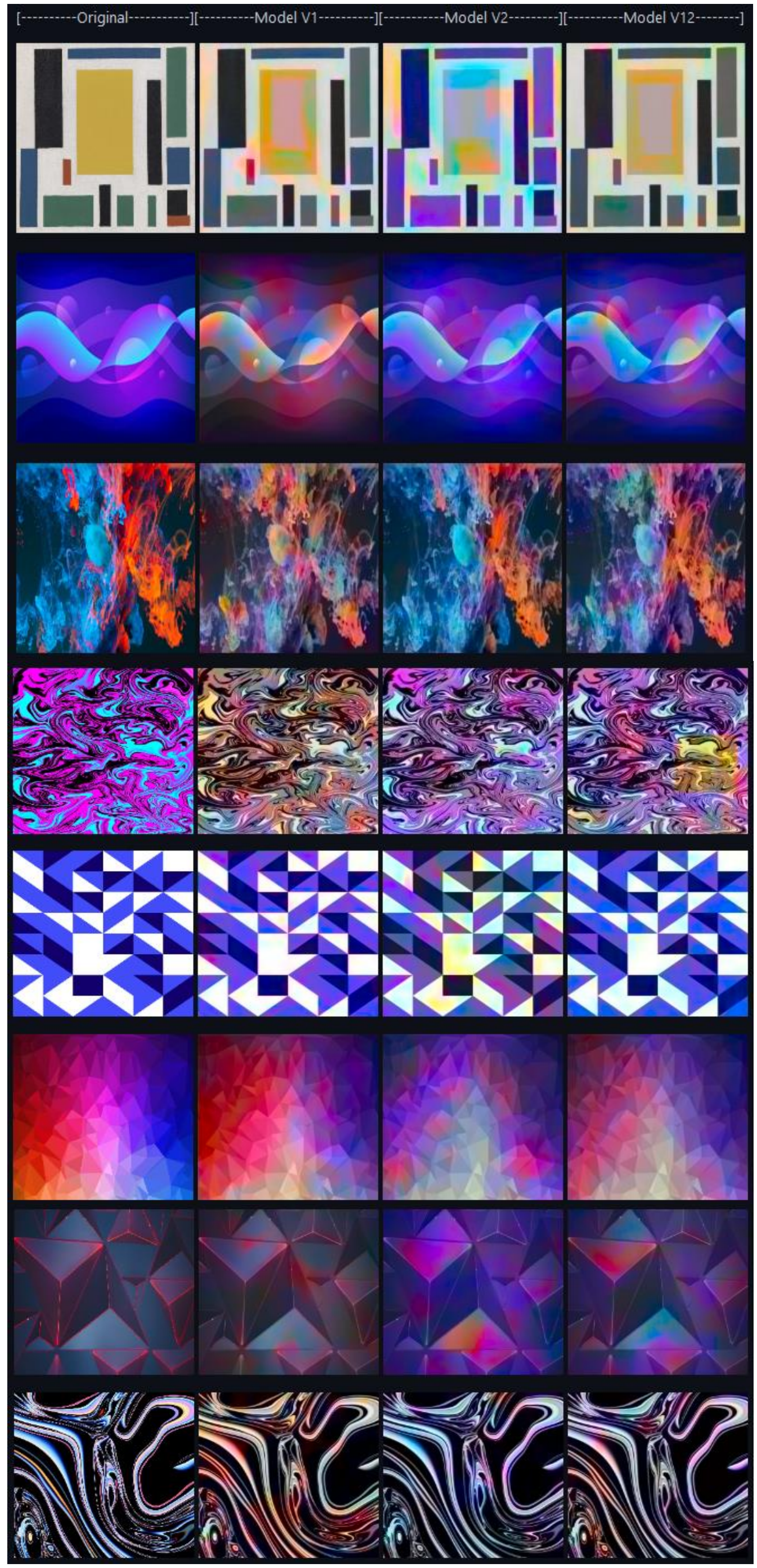

Figure 5: Comparison of the images produced. The result images are displayed in this format.

Original > Model V1 > Model V2 > Model V12

5 Conclusion

5.1 Model V1

Figure 6: Best result from Model V1.

Highest Validation Accuracy achieved on this model is: **0.7494.** Which is palatable considering error margins.

5.2 Model V2

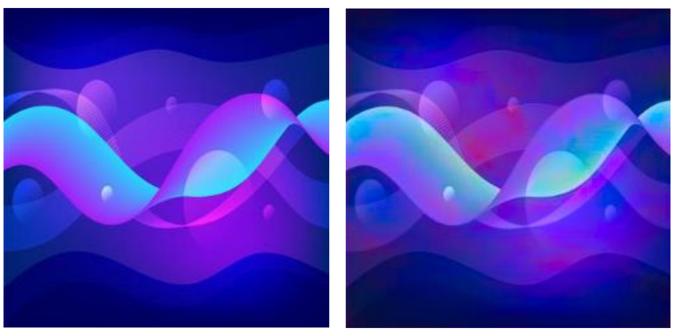

Figure 7: Best result from Model V2.

Highest Validation Accuracy achieved on this model is: **0.8215.** Which is almost as good as you can achieve from a CNN.

5.3 Model V12

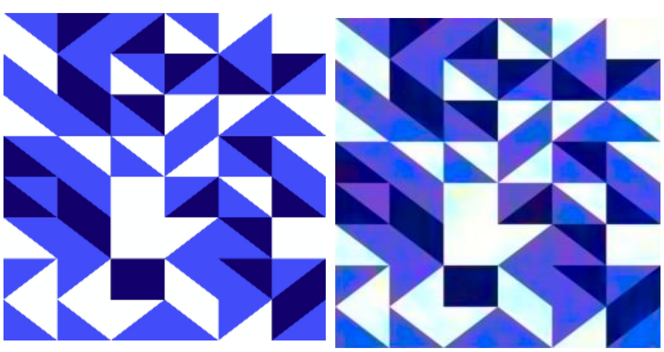

Figure 8: Best result from Model V12.

Highest Validation Accuracy achieved on this model is: **07983.** Which is more than expected for a model trained on both data sets.

By training our model on coloring abstract images we tried to maximize the output of colored images depending on the CIE color lab space. Since we also allow to select tone and saturation to which image must colorize, it makes the model more efficient hence obtaining maximum output and minimum loss.

In conclusion, Training a model with both data sets turned out to be great with validation **accuracy reaching close to 80%,** which is more accurate than the average of model V1 and model V2, therefore training networks on curated similar data-set is not beneficial over mixed data-set.

6 Acknowledgments

The research paper was written in accordance to the minor project "Image_colorizer" which colorizes images and has additional functionality of adjusting attributes such as Saturation, Brightness and Contrast.

We would like to express my gratitude to my Supervisor, Vipin Tiwari, who guided me through the project. We would also like to thank all the members of the group who supported and helped in different modules and parts of this project and research paper.

We would like to acknowledge the help provided by Chandigarh University and their staff. We would like to show our deep appreciation to Dilpreet Kaur for finalizing this project on Image colorization.

(Image_colorizer: https://github.com/Herobrine25mcpe/Image_Colorizer)

7 References

[1] Image Colorization Using a Deep Convolutional Neural Networt Tung Nguyen Graduate School of Information Science and Engineering, Ritsumeikan University, Kazuki Mori , Ruck Thawonmas College of Information Science and Engineering, Ritsumeikan University

[2] ChromaGAN: Adversarial Picture Colorization with Semantic Class Distribution Patricia Vitoria, Lara Raad and Coloma Ballester Department of Information and Communication Technologies University Pompeu Fabra, Barcelona, Spain

[3] https://blog.floydhub.com/colorizing-b-w-photos-with-neural-networks/

[4] Baldassarre, F., Morín, D. G., & Rodés-Guirao, L. (2017). Deep koalarization: Image colorization using cnns and inception-resnet-v2. arXiv preprint arXiv:1712.03400.

[5] https://lukemelas.github.io/image-colorization.html

[6] https://ezyang.github.io/convolution-visualizer/index.html

[7] https://github.com/vdumoulin/conv_arithmetic