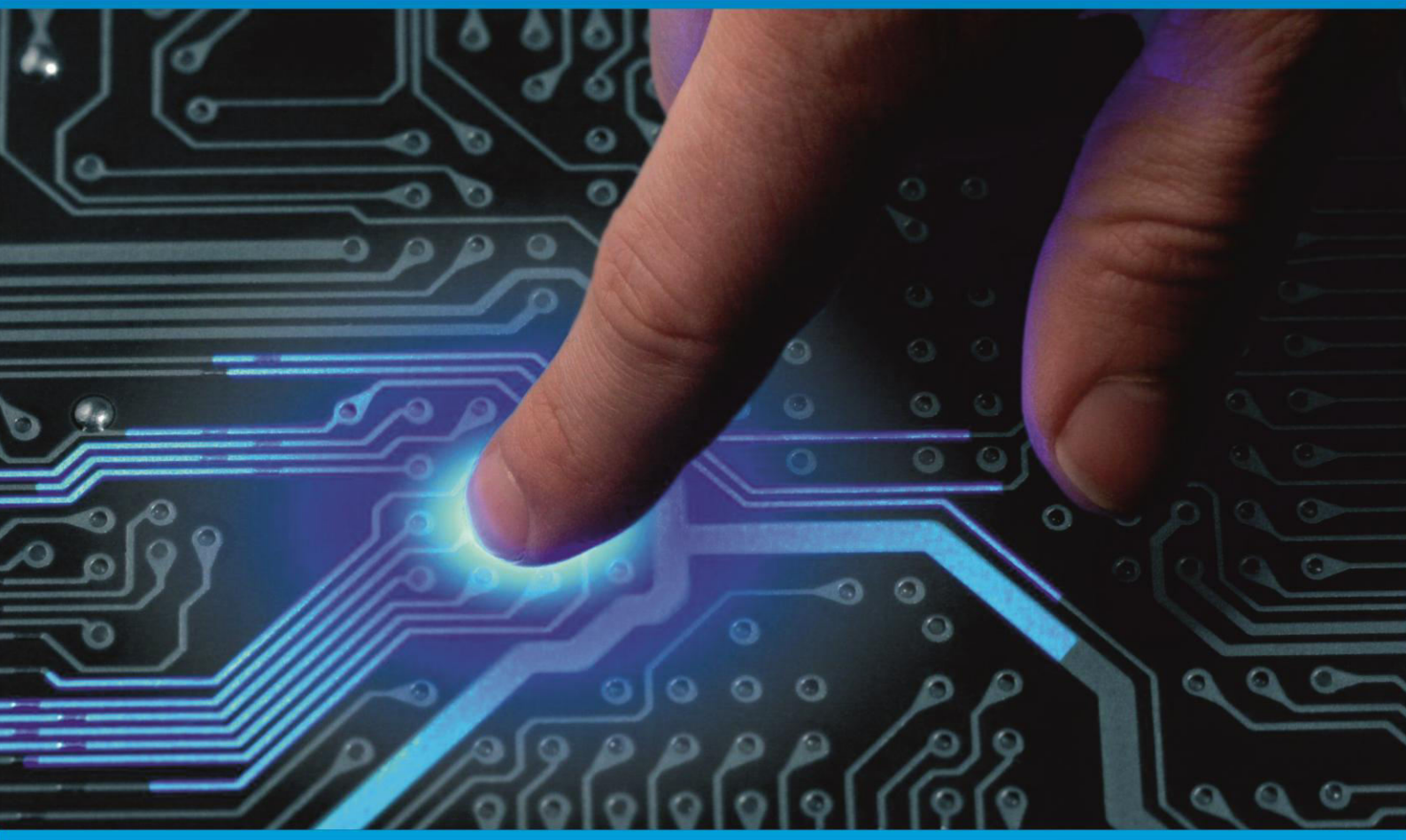
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 7, July 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 8.165

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

Image Colorization: Bringing Grayscale Images to Life using Deep Learning &Computer Vision

Harsh Kumar Shrivastava[1], Avikant Srivastava[2], Sanyam Kumar Singh[3], Suguna M K[4]

BE Students, Department of Information Science and Engineering, Sir M Visvesvaraya Institute of Technology, Bengaluru, India[1,2,3]

Assistant Professor, Department of Information Science and Engineering, Sir M Visvesvaraya Institute of Technology, Bengaluru, India[4]

ABSTRACT: The field of machine learning with deep learning has been used to improve so many fields, be it healthcare, education, technology, etc. This is because of the extensive research being done and creative and unimaginable solutions coming forth every day. Image colorization is also one such technique that uses the power of computer vision along with deep learning to solve a previously unimaginable task. Image colorization is a technique for applying color to a grayscale image to make the image more beautiful and sharper. Grayscale graphics, also known as "black and white" images, are common in older multimedia content. In this paper, a fully automated data-driven deep learning solution for coloring grayscale photos using the capabilities of convolutional neural networks is implemented taking inspiration from the encoder-decoder neural network architecture.

KEYWORDS: Grayscale, Colorization, neural network architecture

I. INTRODUCTION

Image colorization is the technique of applying colors to a grayscale image to make it more aesthetically pleasing and more defined. Grayscale graphics, also known as "black and white" images, are common in older multimedia. Color quality, regardless of its luminosity, is referred to as chromaticity. As a result, appropriately adding chromatic data to photos has become a major research topic.

Automating the task of coloring images is a challenging problem due to the different image conditions that must be handled with one or more algorithms. The problem is also very constrained since two of the three dimensions of the image are lost. Here, there is a need for a technique with better efficiency and reduced time to keep up with the fast-moving world. Today, there are many deep learning/computer vision algorithms that help bring in more potential ideas for tackling these tasks. The aim of this paper is to test deep learning techniques in a comprehensive manner by deriving results through the models and comparing their accuracy.

Fig 1.1 The colorization process using different approaches

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

Traditional colorization requires significant user interaction whether in the form of placing numerous color scribbles, looking at related images, or performing segmentation. In a faster, better, and optimized way, to leverage the semantic information to, for example, color the dusk sky or the human skin—all without requiring human interaction is what is needed.

RGB vs L*a*b

When we load an image, we get a rank-3 (height, width, color) array with the last axis containing the color data for our image. These data represent color in RGB color space and there are 3 numbers for each pixel indicating how much Red, Green, and Blue the pixel is. In the following image, you can see that in the left part of the "main image" (the leftmost image) we have blue color so, in the blue channel of the image, that part has higher values and has turned dark.

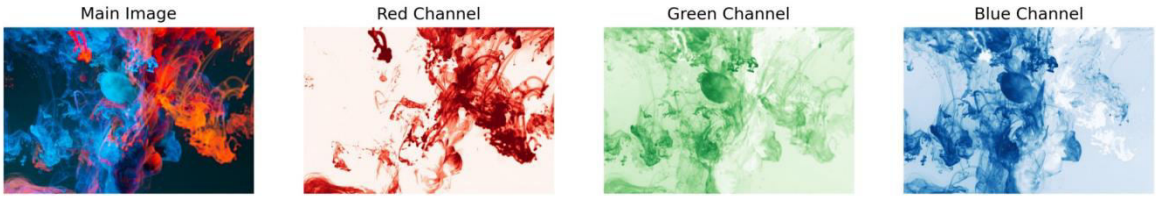


Fig 1.2 Red, Green, and Blue channels of an image

In L*a*b color space, we have again three numbers for each pixel, but these numbers have different meanings. The first number (channel), L, encodes the Lightness of each pixel and when we visualize this channel (the second image in the row below) it appears as a black and white image. The *a and *b channels encode how much green-red and yellow-blue each pixel is, respectively. In the following image, you can see each channel of L*a*b color space separately.

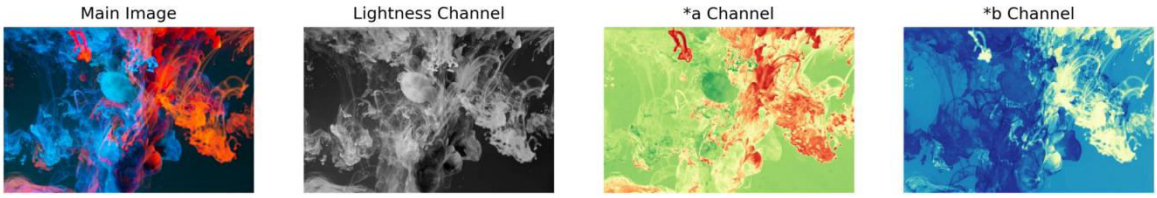


Fig 1.3 Lightness, *a, and *b channels of Lab color space for an image

Most of the approaches use L*a*b color space instead of RGB to train the models. There are a couple of reasons for this choice, but I'll give you an intuition of why we make this choice. To train a model for colorization, we should give it a grayscale image and hope that it will make it colorful. When using L*a*b, we can give the L channel to the model (which is the grayscale image) and want it to predict the other two channels (*a, *b) and after its prediction, we concatenate all the channels, and we get our colorful image. But if you use RGB, you must first convert your image to grayscale, feed the grayscale image to the model, and hope it will predict 3 numbers for you which is a way more difficult and unstable task due to the many more possible combinations of 3 numbers compared to two numbers. If we assume we have 256 choices (in an 8-bit unsigned integer image this is the real number of choices) for each number, predicting the three numbers for each of the pixels is choosing between 256^3 combinations which are more than 16 million choices, but when predicting two numbers we have about 65000 choices (actually, we are not going to wildly choose these numbers like a classification task and I just wrote these numbers to give you an intuition).

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

II. METHODOLOGY

Traditionally and conventionally, images are displayed in Red-Green-Blue (RGB) and each color has a layer. The image is displayed in a pixel grid with pixel values ranging from 0 (black) to 255 (white). CIE Lab examined the color space, Where the L channel represents the luminance and a & b channels represent the green-red and blue-yellow color spectra. The H x W size RGB input image is first converted to the L * a * b color space, and the L channel (luminance) is the input to the model. This provides the model with semantic information about the image, including but not limited to objects and their textures. The task is to predict the other two-color channels. By merging the brightness and the predicted color channels, the model guarantees a high level of detail in the final color image. To color the image, the neural network creates a correlation between the grayscale input image and the colored output image.

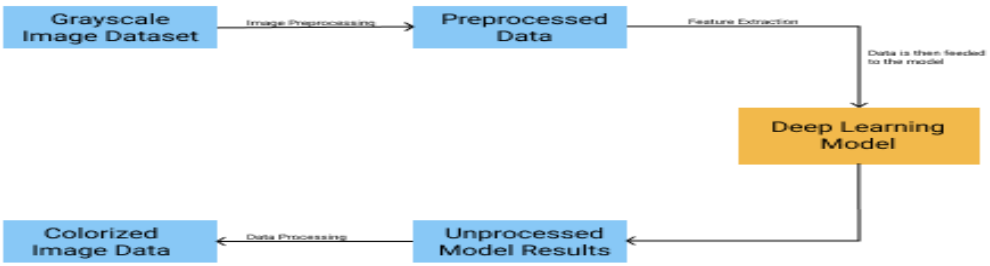


Fig 2.1 High level flow for understanding the colorization process

A. *SYSTEM ARCHITECTURE:*

The convolutional neural network architecture implemented here is inspired by the Encoder-Decoder network architecture. Given the grayscale or the Luminance channel of an image, the model tries to predict the a and b channels of the image and then merge the luminance and the predicted channels to give a colorized image as output. The L channel of size 256x256x1 is provided as input to the Encoder and the output of the encoder is fed to the Decoder which outputs the ab channel prediction of size 256x256x2.

1. ENCODER

The Encoder network consists of 8 2D convolutional layers each having a kernel size of 3 and padding of 1 is applied on each side to maintain the layer's input size. The first, third, and fifth convolutional layers also have a stride of 2. All 8 layers are followed by the ReLU activation function. The output size of this network is 8x8x256 (HxWxD).

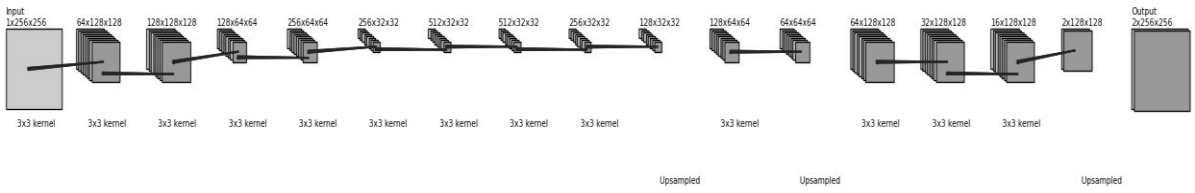


Fig 2.2 Visualization of the convolutional neural network architecture implemented, inspired by the encoder-decoder architecture.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

2. DECODER

The Decoder network consists of 8 layers, 5 convolutional layers each having a kernel size of 3 and padding of 1 is applied on each side to maintain the layer's input size and 3 upsampling layers with a scale factor of 2. All the convolutional layers are followed by the ReLU activation function. The output size of this network is 256x256x2 (HxWxD).

3. OPTIMIZER

The optimizer used in this task is the Adam optimizer implemented in the torch library. the learning rate for the optimizer was initiated with a value of 5e-4 which was decayed as the training went ahead using the Step Learning Rate Scheduler

4. LOSS FUNCTION

The model utilizes MSELoss loss function, which calculates the mean squared error (squared L2 norm) between each element in the input x and target y.

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1         [-1, 64, 128, 128]             640
              ReLU-2         [-1, 64, 128, 128]               0
            Conv2d-3        [-1, 128, 128, 128]          73,856
              ReLU-4        [-1, 128, 128, 128]               0
            Conv2d-5          [-1, 128, 64, 64]         147,584
              ReLU-6          [-1, 128, 64, 64]               0
            Conv2d-7          [-1, 256, 64, 64]         295,168
              ReLU-8          [-1, 256, 64, 64]               0
            Conv2d-9          [-1, 256, 32, 32]         590,080
             ReLU-10          [-1, 256, 32, 32]               0
           Conv2d-11          [-1, 512, 32, 32]       1,180,160
             ReLU-12          [-1, 512, 32, 32]               0
           Conv2d-13          [-1, 512, 32, 32]       2,359,808
             ReLU-14          [-1, 512, 32, 32]               0
           Conv2d-15          [-1, 256, 32, 32]       1,179,904
             ReLU-16          [-1, 256, 32, 32]               0
           Conv2d-17          [-1, 128, 32, 32]         295,040
             ReLU-18          [-1, 128, 32, 32]               0
         Upsample-19          [-1, 128, 64, 64]               0
           Conv2d-20           [-1, 64, 64, 64]          73,792
             ReLU-21           [-1, 64, 64, 64]               0
         Upsample-22         [-1, 64, 128, 128]               0
           Conv2d-23         [-1, 32, 128, 128]          18,464
             ReLU-24         [-1, 32, 128, 128]               0
           Conv2d-25         [-1, 16, 128, 128]           4,624
             ReLU-26         [-1, 16, 128, 128]               0
           Conv2d-27          [-1, 2, 128, 128]             290
             Tanh-28          [-1, 2, 128, 128]               0
         Upsample-29          [-1, 2, 256, 256]               0
================================================================
Total params: 6,219,410
Trainable params: 6,219,410
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.25
Forward/backward pass size (MB): 127.50
Params size (MB): 23.73
Estimated Total Size (MB): 151.48
----------------------------------------------------------------
```

Fig 2.3 The summary of the CNN model

B. *PREPROCESSING:*

The image processing part mainly includes converting and processing the image data into Lab colorspace instead of keeping it to the commonly known RGB colorspace in which almost all the visual data is available.
The image data is resized to 256x256x3 before converting the colorspace. The images are then split into 2. The L (Luminance) channel and the ab (green-red and blue-yellow color spectra) alpha and beta channels. Then the dataset is divided into the training and the test dataset. The input size of the network is 256x256x1, as it takes the Luminance channel (single channel) as input. At last, the images are converted to Tensors and then finally fed to the deep learning architecture model and the results are obtained.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

C. *POSTPROCESSING:*

These results in the form of Tensors of the size 256x256x2 which denote the protected ab color channels are then processed using image processing techniques that include first converting the Tensors into numpy format and then merging the Luminance channel of the corresponding image. The predicted "image" in LAB format is finally transformed into its intended format i.e., the RGB colorspace (BGR format in case of using OpenCV to view and save results).

III. EXPERIMENTAL RESULTS

The project compares the encoder-decoder network trained and validated at 10, 20, and 50 epochs with a batch size of 40.

Dataset:

The dataset used here corresponds to the real-world photos from Flickr described in the paper of CartoonGAN. The dataset contains the following type of landscape-based pictures:

- landscapes (900 images)
- landscapes mountain (900 images)
- landscapes desert (100 images)
- landscapes sea (500 images)
- landscapes beach (500 images)
- landscapes island (500 images)
- landscapes Japan (900 images)

The dataset is taken from Kaggle and contains a total of 4319 pictures from which, 4200 of the images are used for the training and validation process of the deep learning Encoder-Decoder network architecture.

Training:

The model is trained and validated at 10, 20, and 50 epochs. The model is not initialized with any pre-trained weights. Adam optimizer is used at a learning rate of 5e-4 and no momentum parameters.

Adam optimizer is used because it converges faster and provides lower loss values. The loss function used is the MSELoss criterion, which calculates the mean squared error between each pixel in the input and the output.

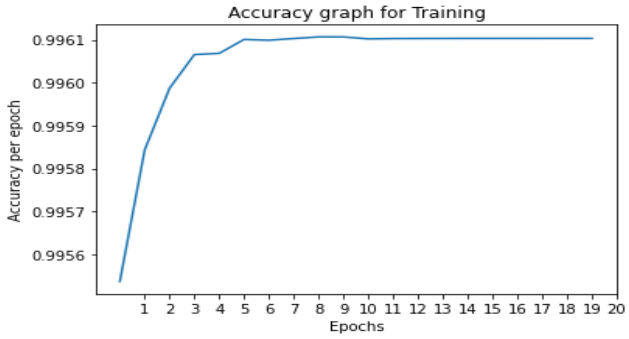


Fig. 3.1 A line graph for epoch vs accuracy when network is set to training mode

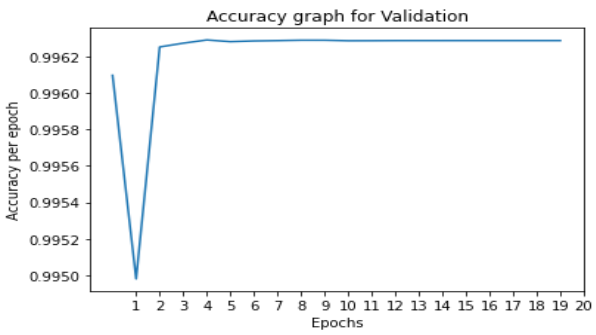


Fig. 3.2 A line graph for epoch vs accuracy when network is set to validation mode

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

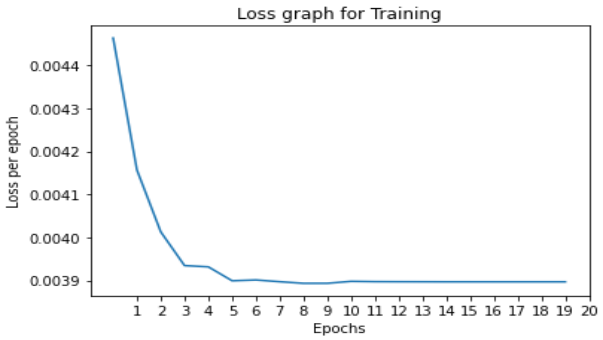


Fig. 3.3 A line graph for epoch vs loss when network is set to training model

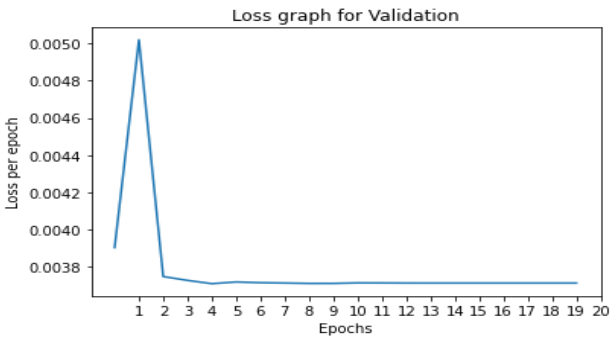


Fig. 3.4 A line graph for epoch vs loss when network is set to validation mode

Results:

The network implemented achieves a stable accuracy after just 10-15 epochs but can't properly colorize the test image dataset which leaves room for improvement in the network architecture.

The loss and accuracy for model20 corresponding to the encoder-decoder network trained for 20 epochs have been graphed.

Fig. 3.5 Original colored image from unsplash

Fig. 3.6 Grayscaled Image

Fig. 3.7 Image predicted after the model

Fig. 3.8 Image predicted after the model

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

was trained for 10 epochswas trained for 20 epochs

Fig. 3.9 Image predicted after the model was trained for 50 epochs

IV. CONCLUSION

In some instances, colorization using a Deep Convolutional Neural Network and an objective function can provide results that are indistinguishable from "real" color images. Colorization of single images is a subject of research with important real-world applications. Deep convolutional algorithms for image colorization have seen tremendous expansion as a result of deep learning approaches' extraordinary success. Various ways utilizing network architecture, training methodologies, and learning paradigms, among other things, are offered based on exciting innovations. Black-and-white material, such as historical photographs, movies, and sketches, can be colorized more effectively and on a larger scale. This improvement can be further bolstered by utilizing a much larger dataset, such as ImageNet, which contains 1.4 million images divided into 1000 classes, allowing the model to be generalized to a wide range of images. Furthermore, increasing the number of training steps improves the colorization of the model.

V. FUTURE SCOPE

Scope:

The very nature of this project allows for multiple techniques to be integrated together as modules and their results can be combined to increase the accuracy of the final result. Another approach of instance and context-based learning can also be integrated into the project as that would help on increasing the efficiency of a model. This provides a great degree of modularity and versatility to the project. Colorizing videos is also a stretch goal of this project.

While the project might not be able to reach the universal goal of any research project in the field of artificial intelligence and machine learning of 100% accuracy or perfect results, it might end up creating a system that can, with enough time and data, get very close to becoming practically and commercially usable.

While the project might not be able to reach the universal goal of any research project in the field of artificial intelligence and machine learning of 100% accuracy or perfect results, it might end up creating a system that can, with enough time and data, get very close to becoming practically and commercially usable.

Dataset is also an immense factor in deciding how the results look like. The precision of the algorithms increases when the size in quality of the dataset is increased, or when the dataset is less ambiguous and truer to what something might generally be perceived as in terms of color, shape, size, and texture. That is where image processing comes in and it can help in improving the dataset considerably. Hence, quality data will surely make the model more accurate and reduce the number of false positives and false negatives.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007052 |

Contributions to Society:

The real aspect of this project comes to play when its advantages for society are illustrated briefly. This project intends to propose an enhanced approach that might give more insight into this topic to further the exploration of new techniques that might not just be useful to this specific area of image colorization but in general, be useful to the computer vision and deep learning community for other vision research areas. There are also many practical uses of image colorization where this project will be able to contribute, for example, (forensics, movies).

Limitations:

- Low quality and/or the resolution of visual data.
- Already edited i.e., unnatural image data.
- Ambiguous and complex image data might affect the results.
- Comparisons might not be fair.
- Lack of proper black and white legacy image data.

Fig 7.1 Limitations of Image colorization

REFERENCES

[1] Cheng, Z. et al. Deep Colorization. , 2015 IEEE International Conference on Computer Vision (ICCV). (2015)
[2] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," ACM transactions on graphics (TOG), vol. 23, no. 3, pp. 664– 672, 2004.
[3] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in European conference on computer vision. Springer, 2016, pp. 649– 666.
[4] F. M. Carlucci, P. Russo, and B. Caputo, "(DE)2 CO: Deep depth colorization," IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 2386–2393, 2018.
[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in IEEE conference on computer vision and pattern recognition, 2009, pp. 248– 255.
[6] Anwar, Saeed, Muhammad Tahir, Chongyi Li, Ajmal S. Mian, Fahad Shahbaz Khan and Abdul Wahab Muzaffar. "Image Colorization: A Survey and Dataset." ArXiv abs/2008.10774 (2020): n. Pag.
[7] JasonAntic. jantic/deoldify: https://github.com/jantic/DeOldify, 2019.
[8] Bana, Tejas, JatanLoya and Siddhant Vijay Kulkarni. "ViT-Inception-GAN for Image Colourising." ArXiv abs/2106.06321 (2021): n. pag.
[9] Nguyen, Tung Duc, Kazuki Mori and Ruck Thawonmas. "Image Colorization Using a Deep Convolutional Neural Network." ArXiv abs/1604.07904 (2016): n. pag.
[10] Baldassarre, F., Morín, D. G., &Rodés-Guirao, L. (2017). Deep koalarization: Image colorization using cnns and inception-resnet-v2. arXiv preprint arXiv:1712.03400.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.165

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details