Automated Colorization Of Black and White and Gray-Scale Images

Krishanth Kumar
CB.EN.P2CEN19007
Center for excellence in Computational Engineering & networking
Amrita Vishwa Vidyapeetham
Coimbatore, India
krishanthvs@gmail.com

Sai Kesav.R
CB.EN.P2CEN19010
Center for excellence in Computational Engineering & networking
Amrita Vishwa Vidyapeetham
Coimbatore, India
rachakondasaikesav@gmail.com

Bhanu Prakash.M
CB.EN.P2CEN19008
Center for excellence in Computational Engineering & networking
Amrita Vishwa Vidyapeetham
Coimbatore, India
bhanu0893@gmail.com

***Abstract*—Image colorization is an important process which enables any black and white or grey images to be retained back to their original color. The proposed algorithm uses Skip connections in CNN and leaky-ReLu activation function to analyze the colors across a set of color images and their respective black and white versions and the final output being given as a colourized photo. We have trained the model with datasets containing grey scale images and their respective colour images and finally found that we have achieved a good colourization.**

***Index Terms*—deep learning, Skip connections in CNN, Leaky ReLu**

INTRODUCTION

In recent days Deep Learning (DL) is getting very popular and is being used for solving research problems in a wide range of fields like bio-medical, cyber security, autonomous vehicles etc. Deep learning models are replacing traditional machine learning based models due the fact that DL models have the ability to automatically extract the useful features from the input while the traditional machine learning models require manual feature engineering. One of the most popular Deep learning (DL) architectures is the Convolutional neural network (CNN). CNN is very popular in the computer vision field as it uses convolution filters in order to extract the various location invariant features. Since CNN models work well with images, various CNN models and its variants are used in biomedical fields and image colorization applications using computer vision, which enables a visual appeal of the images with finer details and enhanced color accuracy, making the image more understandable by adding more relevant information.

From the chosen datasets, with which we run our architectures and analysed our outputs, and then through the Lab color space known as CIELAB color space, where ‘L’ is described as the lightness characteristic from black to white and ‘a’ from green to red and ‘b’ from blue to yellow, unlike the RGB and CMYK colour models, Lab is designed for aspiring perceptual uniformity and for approximate human vision.

From the chosen color images from the datasets we have trained each image in their black and white versions and using transfer learning method we have used them in VGG16 CNN architecture and the last layer gives Lab channel which enables the colorized output for the given input images. For the better understanding of the image colorization process we have referred to the standard architecture for colorization for the same problem statement proposed by Zhang[1].

We have structured two new architectures where one gave us a colored version of the input image but with not a great result and the second architecture which was structured as an encoder-decoder architecture with skip-connections fed from encoder to decoder with a Leaky ReLu activation function and a final RGB channel enabling us outputs with better colorization and accuracy.

OBJECTIVE

Our primary objective is to compare architectures to colorize a black and white or grayscale image. And also to propose a new architecture which will efficiently colorize a black and white image.

RELATED WORKS

CNN-based architectures are used in segmentation of images. There are many studies involving colourization of grey scale images. In[1], it was proposed that the dataset be converted into LAB colour space and only the L channel is used for training their CNN and predict the AB channel. In[7], they utilized a pre-trained CNN, which was originally designed for image classification, they were able to separate content and style of different images and recombine them into a single image, then proposed a method that can add colors to a grayscale image by combining its content with style of a color image having semantic similarity with the grayscale one. In[3], they proposed transfer learning with an autoencoder format architecture with pretrained imagenet weights and converting the dataset into LAB colour space and training with L channels and predicting the AB channels.

In [2], Classical U-Net architectures composed of encoders and decoders are very popular for segmentation of medical images, satellite images etc. The U-Net architecture uses skip connections to combine low-level feature maps with higher-level ones, which enables precise pixel-level localization. A large number of feature channels in the upsampling part allows propagating context information to higher resolution layers. This type of network architecture proved themselves in binary image segmentation competitions such as satellite image analysis and medical image analysis.

DATASET DESCRIPTION

A total of 3 datasets were used for different architectures, a dataset of randomly chosen 2500 images with no classes defined, another dataset is Linneaus 5 dataset with a total of 1200 training images and 400 testing images belonging to 5 classes.

And the last chosen dataset contained 5556 training images and testing images of different classes.

The Classes attributes in the last dataset are the large-scale image collection the details of which are in Table I.

Category Names	**Train**	**Test**
People's faces	**4596**	**960**
Coast		
Buildings		
Mountains		
Forests		
Open country		
Street		
City center		
Total	**5556**	

Table I: THE SUMMARY OF DATABASE AVAILABLE FOR COLORIZATION

OUR METHODOLOGIES

Lab Color Space:

The CIELAB color space describes mathematically all perceivable colors in the three dimensions L for lightness and a-b for the color components of green-red and blue-yellow respectively, where the RGB values range from (0,255) but for the model we have used we were procured to convert them to (0,1) range, by dividing the R,G,B values of each pixels by a maximum of (255) and converting them from RGB to Lab, where L is sustained through out the model dispensing on the convolutions and the layers added and a-b channels be the final layer giving the colorized outputs.

For the proposed architecture (KKB architecture), we have taken preprocessed data of numpy arrays of shape (width, height, 3),3 describes the RGB channels in the color space.

A. VGG16 Transfer Learning Model:

In VGG16 architecture there were two phases encoder and decoder phase, and the pretrained imagenet weights of the VGG16 model were taken until 5 convolutional layers the input image has been converted from RGB to Lab color space and after the fifth convolutional layer a custom decoder with fully connected layers and a ReLu activation function and is max pooled and is predicted by the L channel where the a and b channels for colorization is decided during the whole encoder and decoder process.

B. new CNN Architecture:

In this architecture we have taken the image as an input after its conversion from RGB to Lab color space where a and b channels are added before the output and L is along the whole convolutional procedure, similar to the VGG16 architecture, unlike taking the pretrained weights, we have taken the input image and run it through around 13 convolution layers with 3 up sampling layers after 9, 10, 13 convolution layers respectively

and then through the a and b color space, which gave us an partialled output of an colored image as there was no encoder and decoder used and with the weak output of the colorized image, in the next architecture we did propose skip connections, deconvolution and using Leaky ReLu activation functions for the better and clear rich colored output.

C.Proposed new Architecture (KKB Architecture):

After the partially colorized output of the previous architecture, we have taken preprocessed input data, where the black and white images and their respective color photos are in numpy array format and loaded them directly into the model.

The dataset chosen was the 3rd dataset mentioned above which comprised 5556 training and testing images of people's faces, Coast Lines, Buildings, Forests, Open Countries, Streets and City Centres, and CelebFaces attributed the large dataset with more than 2000 celebrity images.

The model was structured in an encoder-decoder format with skip connections fed from each layer of convolution belonging to the encoder to that of each layer of convolution belonging to the decoder.

For each convolution layer a Leaky ReLu activation function is used, unlike ReLu function when the functions value turns out to be 0 for negative outputs, there is a small negative slope of minimal value like 0.01 multiplied, the function computes for $f(x) = 1(x<0)(\alpha x)+1(x>=0)(x)$, where the α is a small assumed constant.

In the encoder set, there are a total of 7 convolution layers, after the convolution and the activation function a batch normalization layer is added for yielding better results and a dropout layer connected to the batch normalized layer before the next convolution layer so that it doesn't block any information further, this process continues until the 7th layer, where the model has completed its encoding.

In the decoder set,there are a total of 8 deconvolutional layers or better known as the transposed convolutional layers with a Leaky ReLu activation function and an additional empty layer of the same size of that of deconvolved output, which is connected from the encoder to each deconvolved layer of decoder until the 14th deconvolutional layer, after that the 15th layer of the model is again a convolutional layer and the RGB channels are predicted here colorizing the input grey scale image.

The empty layer in each encoder enables to skip certain features needed for the colorization of the image and the color intensity of each feature in the image from the corresponding decoder makes an accurate colorized output with better colorization of the given input. The block diagram of the proposed architecture can be seen in Figure 6.

Figure 3,4,5 show the output achieved by the proposed architecture and we can see that the output color image is better than those of the previously proposed architectures(Figure 1,2)

Input Output

Fig. 1.Input and Output images of VGG16 architecture

Input Output

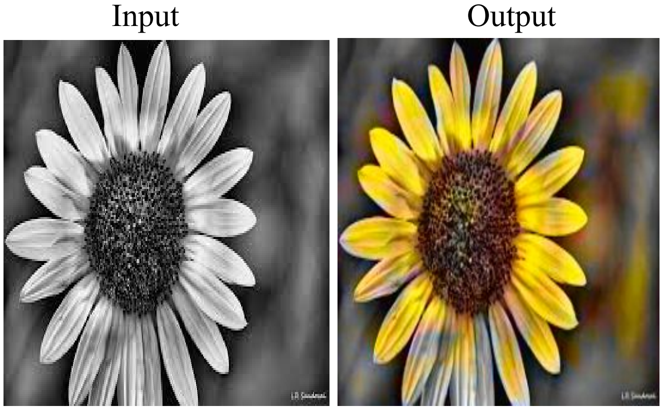

Fig. 2.Input and Output images of new CNN architecture.

Input Output

Fig. 3.Input and Output images of new architecture (KKB architecture)

Input Output

Fig. 4.Input and Output images of new architecture (KKB architecture)

Input Output

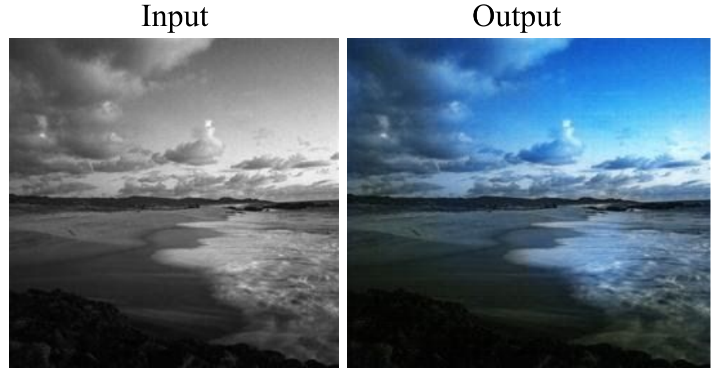

Fig. 5.Input and Output images of new architecture (KKB architecture)

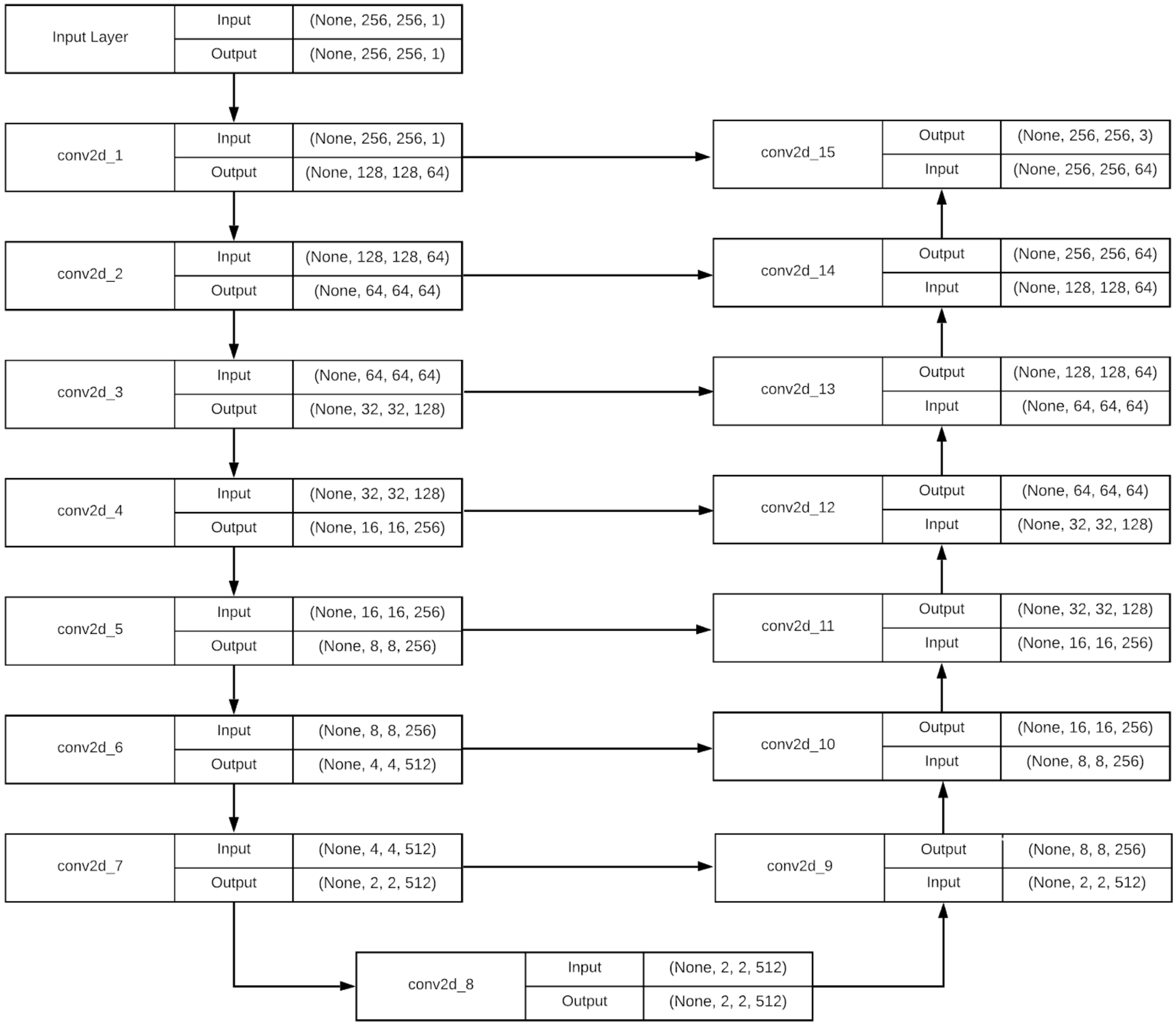

Fig. 6.Block Diagram of new architecture (KKB architecture)

Discussion

Our analysis shows that VGG-16 architecture and the new CNN architecture were not able to give us the better accurate colorized output, when compared to the proposed new architecture (KKB architecture), because the deconvolutions and convolutions play a major role in encoding and decoding a certain image’s features and colored output.

The challenges we faced were, in some datasets the input images were full of red color which was overshadowing the total output, so we had to search for a new dataset and while trying for VGG19 architecture the computational capacity wasn’t sufficient

Conclusion

As we go deeper in the architecture, the output is better.We have observed the previous statement to be true for 10 epochs by seeing the outputs of shallow and deep architectures as seen in the figure below.

Fig.7.Deep Architecture Output

Fig.8. Shallow Architecture Output

In this study of image colorization, we have tried to find out the best colorized output for a given input from the respective datasets used for the each architecture, but for the final proposed architecture the input images preprocessed by taking the black and white images with their respective color photos in numpy array format and loaded directly into the model, and when compared with the two of the architectures used VGG16 with transfer learning, and New CNN architecture, the proposed architecture (KKB architecture) with the structure of convolutions and deconvolutions acting as encoder and decoder gave the best output of the colorized image for their respective inputs.

References

[1] Richard Zhang, Phillip Isola, Alexei A. Efros. Colorful Image Colorization. preprint arXiv:1603.08511

[2] V Iglovikov, A Shvets. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation preprint arXiv:1801.05746

[3] Autoencoders using transfer learning - Image colorization from Python for Microscopists by Sreeni youtube channel

[4] https://www.xrite.com/blog/lab-color-space

[5]https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

[6] https://www.learnopencv.com/convolutional-neural-network-based-image-colorization-using-opencv/

[7] Tung Nguyen, Kazuki Mori, Ruck Thawonmas. Image Colorization Using a Deep Convolutional Neural Network

[8] https://www.machinecurve.com/index.php/2019/12/10/conv2d transpose-using-2d-transposed-convolutions-with-keras/

[9] https://towardsdatascience.com/batch-normalization-theory-and-how-to-use-it-with-tensorflow-1892ca0173ad

[10] https://www.kaggle.com/sravankumar9892/

[11]https://machinelearningmastery.com/upsampling-and-transpose-convolution-layers-for-generative-adversarial-networks/

[12] https://www.youtube.com/watch?v=EujccFRio7o&feature=youtu.be

[13] https://www.xrite.com/blog/lab-color-space

[14] https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

[15] http://chaladze.com/l5/