Indian Institute of Information Technology(IIIT), Design and Manufacturing, Jabalpur

CS314b : Machine Learning
Literature Review

Reverse-Oldification Of Images

Manav Goyal *2017138*
Aditya Baurai *2017307*
Rakshak *2017207*

We want to thank our course instructor Dr. Kusum Bharti(CSE faculty at IIIT Jabalpur) for giving us the opportunity to work on this project.

(I) Abstract :

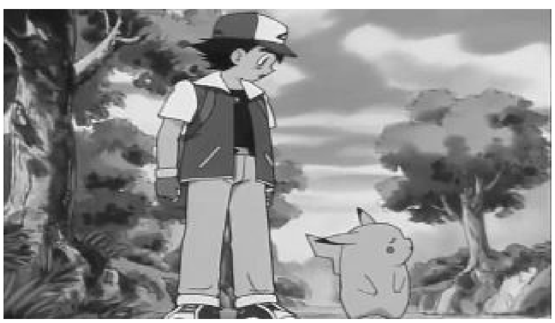
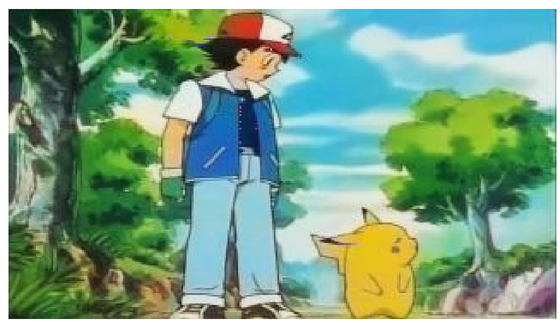
Here we aim to study and implement a deep learning architecture revolving around the application of a neural network in order to rejuvenate old images, that is by colorizing them, and hence making them *'alive'* again. Image restoration cum reconstruction has always been a topic of interest, with applications such as extracting useful information from the images of ancient historical artifacts(*after reverse-oldifying it to increase the color channels and hence, the amount of information encapsulated*), or even bringing a black and white snapshot from the 90s to this century(*applications in entertainment industry*), or colorizing the popular Mangas(Japanese comics), which are drawn without colors(mostly). The heavy process has expedited with the advent of the modern deep-learning/Big Data era, where GPUs and TPUs are getting more and more powerful as time progresses, along with a massive surge in the amount of data available to learn from.

(II) Research papers that aided in this project :

- **Colorization using ConvNet and GAN -** *by Quiwen Fu(Stanford University), Wei-Ting Hsu(Stanford University), Mu-Heng Yang(Stanford University)*

  In this paper, the aforementioned authors implemented Convolutional Neural Networks(CNNs) as well as Generative Adversarial Neural Networks(GANs) on a plethora of images taken from Manga(Japanese comics). Normally, Manga is drawn without colors(black and white) until they are made into animations and aired on television. In order to provide readers with more pleasant and comprehensive reading experience, the authors used deep learning to color the Manga images.

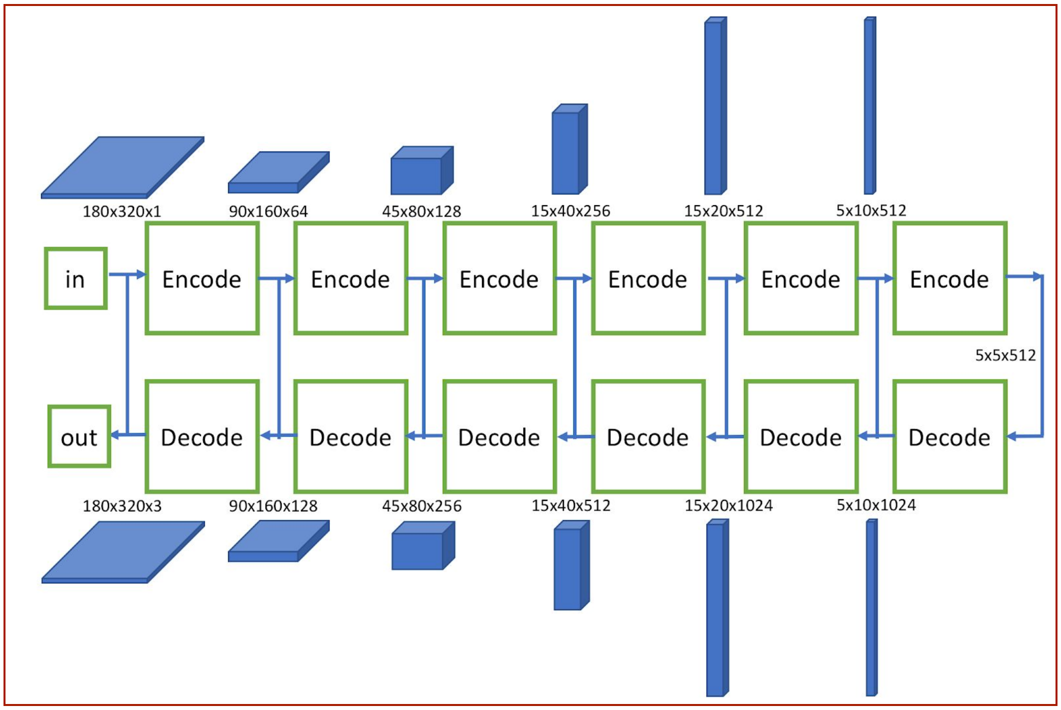
Following were the results observed :

Manga image (Black and White)	**Colored image using ConvNets as observed by the authors**
	
	
	

(Fig 1 || *source of the paper is given in the glossary section at the end of this document*)

Following methods were used by the authors in order to colorize the images and obtain the aforementioned results :

- **ConvNets :** This architecture uses convolving layers in an **encoder-decoder** fashion in order to generate colorized images from input grayscale or edge-only images and uses pixel wise L2 loss(*mean square error*) as the set loss function to optimize. According to the authors, the architecture used was symmetrical having six encoding and decoding layers, with increasing number of filters while encoding and decreasing number while decoding. In encoding we use 2-D convolutions with strides for image downsizing from a given to a target configuration. Whereas, in image decoding we perform 2-D convolution with stride one, along with 2-D upsampling operation in order to upsize and result the output image at the end. The model they used is depicted down below :

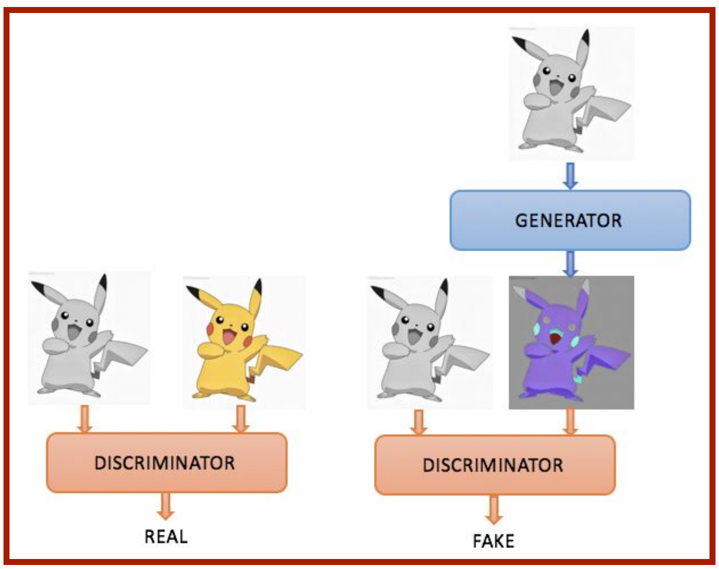


-

(Figure 2 || *Source of the paper at the end of this document.*)

- **Generative Adversarial Networks :** In part two of their paper, the authors deployed a Generative Adversarial Network, which is essentially two neural networks competing with each other over time, learning together and enhancing each other's performance. In a nutshell, a generator takes the input and generates a fake image. The discriminator gets images from both the generator and the label, along with grayscale input, and tries to tell which pair contains real colored images. At each iteration, the generator produces a more realistic image(***having relatively more accurate color filling on the input grayscale image***), while the discriminator gets better at distinguishing false/ fake images(ones with the wrong color distribution). Both models are trained in a min-max fashion until the generator is capable enough to '*outsmart*' the discriminator.

The architecture of the generator is same as that of the ConvNet, whereas that of the discriminator resembles the encoder part of the ConvNet architecture.



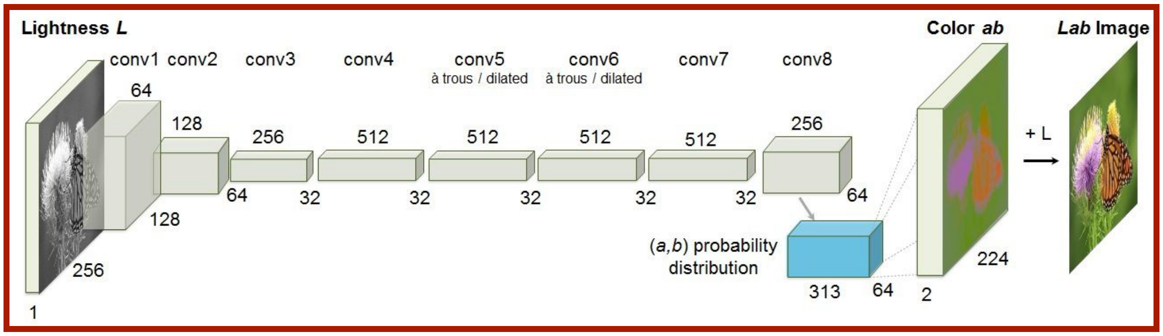
(Fig 3 || *discriminator trying to figure out whether the color filling seems real or fake. Paper's link at the end of the document.*)

The results are slightly better than those observed in Fig 1.

- **Colorful Image Colorization-** *by Richard Zhang(UC Berkeley), Phillip Isola(UC Berkeley), and Alexei A. Efros(UC Berkeley)*

This paper also attacks the image colorization challenge by producing a *plausible* color to an input grayscale image by using Self-supervised learning approach. The system is implemented as a feed-forward Convolutional Neural Network(CNN) at test time. At the crux of this paper, the authors take advantage of ***CIELab*** image format, which separates the luminosity completely from the color tradeoffs. The

Lightness channel '**L**' is fed into the architecture, corresponding to which channels '**a**' and '**b**' were predicted. An architecture similar to ConvNet - as mentioned in the previous paper - was used in order to extract the features from an input '**L**' channel and then predict '**a**' and '**b**' channels accordingly by deconvolution operations. The model architecture used is as follows :



(Fig 4 || *Each conv layer refers to a block of 2-3 repeated* ***conv*** *and* ***ReLU*** *layers, followed by* ***BatchNormalization****. The net has no* ***pool*** *layer. The image is taken from the research paper appended at the end of this document).*

The authors received stunning results as a result of the aforementioned approach.

(Fig 5 || *Results obtained by the authors. The input images are the ones in row 1, and the corresponding 'colorized' output are stacked in row 2. Full paper's link at the end of this document*).

(III) Insights and takeaways from the aforementioned researches :

- A **deep convolutional Auto Encoder**(*a neural network structure having a series of successive downsampling layers in order to extract useful features from a given input[**the encoder layer**], followed by a series of successive upsampling layers in order to construct an output based on the **features extracted**[**the decoder layer**]*) is to be used. So, a black and white image will be fed as an input to the architecture and its color channels will be predicted after the network breaks down the image in the middle in order to extract useful features from it.
- There are two approaches possible now :
    - Image can be converted into grayscale and the corresponding **R, G and B** (red, green, and blue) channels are predicted by the autoencoder neural network architecture.
    - Image can be first converted into **CIELab** format and then the '**L**' channel can be fed as input to the architecture, resulting in the prediction of '**a**' and '**b**' channels. The three channels will subsequently be combined into an **Lab** image, which is displayed after re-converting it into **RGB** format.
- Out of the two approaches mentioned above, we decided to proceed with the second one. The primary reason behind the approach is that in CIELab format, it completely separates luminosity(due to the 'L' channel) with the color tradeoffs(controlled by the 'a' and 'b' channel). Also, instead of predicting three channels as in approach one(mentioned above), we only need to predict two('a' and 'b' feature values).
- Choosing a data set may seem like a trivial task, however it's one of the driving factors towards the success of this project. A **consistent** collection of images belonging to the same **source distribution** is selected rather than a mix of a plethora of random images.

(IV) References :

- Colorful Image Colorization - *by Richard Zhang, Phillip Isola, and Alexei A. Efros from University of California, Berkeley* : **https://arxiv.org/abs/1603.08511**
- Colorization Using ConvNet and GAN : *by Quiwen Fu, Wei-Ting Hsu and Mu-Heng Yang from Stanford University :*

  https://www.semanticscholar.org/paper/Colorization-Using-ConvNet-and-GAN-Fu-Hsu/327f96c410ab390b2778ffb579d89632b210d337