**COLORIZING B&W IMAGES USING GANs AND CNN**

### A Project Work Synopsis

*Submitted in the partial fulfillment for the award of the degree of*

# BACHELOR OF ENGINEERING

### IN

### BIG DATA ANAYLTICS

### Submitted by:

### DHRUV MISHRA

### 20BCS3844

### PUNEET CHAUDHARY

### 20BCS3848

### Under the Supervision of:

**Mr. Pulkit Dwivedi**



### CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,

**PUNJAB**

**OCTOBER, 2022**

# Table of Contents

Title Page

Abstract

List of Figures

#### INTRODUCTION\*

* 1. Problem Definition
  2. Project Overview/Specifications\* (page-1 and 3)
  3. Hardware Specification
  4. Software Specification

1. LITERATURE SURVEY
   1. Existing System
   2. Proposed System
2. [PROBLEM FORMULATION](#_TOC_250001)
3. RESEARCH OBJECTIVES
4. METHODOLY
5. CONCLUSION AND FUTURE SCOPE

#### TENTATIVE CHAPTER PLAN FOR THE PROPOSED WORK

1. **REFERENCES**

# ABSTRACT

Black and white image colorization by hand is time- consuming and ineffective. It has been tried with

Photoshop editing is challenging because it necessitates in- depth research and takes up to a month to colourize a photo.

Using sophisticated image colorization techniques is a practical way to approach the task. Since it lies at the intersection of two obscure fields, digital image processing and deep learning, the literature on picture colorization has attracted attention in the past ten years.

There have been initiatives to take advantage of transfer learning’s advantages and the ever-increasing accessibility of end-to-end deep learning models.

Deep learning models like Convolutional Neural Networks can be used to automatically extract image attributes from the training data.

**List of figures**

# Graph of stock price

# Lstm Architecture

# Flow chart

# INTRODUCTION

# PROBLEM DEFINITION

# In this project we attempt to implement deep learning approach to colorize old black and white images. Deep learning is effectively implemented in image transformation. The objective is to colorize old black and white images in order to make them 3 dimensional RGB images. The system must be able to access all images in the dataset, convert them into grey scale images. It must convert the grey scale images to the colored RGB images. Also provide the difference between the greyscale , colored and ground truth images.

# PROJECT OVERVIEW

# Colorizing contains of two models – Generator and Discriminator.

# ● Generative Adversial Networks(GANs) are a way to make a generative model by haiving two neutral networks. complete with each other.

# ● Recolorizing and restoring old photos is a painstaking process when done manually through some photo editing software.

* One solution to this problem is using GANs for restoring and recolorization of old photos.
* Colorizing black and white images with deep learning has become as impressive showcase for the real-world application if neutral networks in out lives.

# 

# Graph of stock price

# GANs example

# As seen in the picture the colorizing of the grayscale is being done. There are three images in one row which shows three types of images the grey scale, colorized and the ground truth in which the ground truth is the original image that we converted to grayscale and colorized output is the image formed after training the data.

# 

# HARDWARE SPECIFICATION

# 8 GB RAM

# Processor – Ryzen 7 4800H

# 1.4 SOFTWARE SPECIFICATION

# Python Compiler with required Libraries and Modules

# Language: Python

# Operating System: Windows 7/8/10/11

# Modules like:

# PIL

# GANs

# Keras

# Tensorflow

# Pandas

# Numpy

# CNN

# LITERATURE REVIEW

**2.1 BOOKS ABOUT GANs**

9 Books on Generative Adversarial Networks (GANs) by Jason Brownlee on August 21, 2019 in Generative Adversarial Networks Tweet Tweet Share Last Updated on August 21, 2019

1 The First GAN (2014) Goodfellow, Ian, et al. “Generative adversarial nets.” Advances in neural information processing systems. 2014.

In 2014, Ian Goodfellow was in a bar with his friends dis- cussing how to synthesize images using artificial intelligence. A typical talk over a beer. While his friends were discussing statistical approaches, he argued for using two neural networks to jointly learn “how to draw” and to “judge the drawing”. The former trains on the feedback of the latter, which trains to recognize real images from fake images, just to bash the former.

Initially, the idea was met with criticism: training one network is hard, training two is madness. Nonetheless, slightly drunk, Goodfellow went back home and coded it overnight. To his surprise, it worked. 2 StyleGAN (2019) Karras, Tero, Samuli Laine, and Timo Aila. “A style-based generator ar- chitecture for generative adversarial networks.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019. 3 Pix2Pix and CycleGAN (2017) Isola, Phillip, et al. “Image-to-image translation with conditional adversarial networks.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Zhu, Jun-Yan, et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks.” Proceedings of the IEEE international conference on computer vision. 2017.

Semi-Supervised Learning (2016) Salimans, Tim, et al. “Im- proved techniques for training gans.” Advances in neural information processing systems. 2016. 5 Anime GAN (2017) Jin, Yanghua, et al. “Towards the automatic anime characters creation with generative adversarial networks.” arXiv preprint arXiv:1708.05509 (2017).

2.1.3 RELATED WORK:

# Iizuka et al. In this paper he combined two networks, one to predict the global features of the input image and the other to specialize in local features of input images. The global features network is trained for image classification and directly concatenated to the local features network which are then trained for colorization of images using L2 Euclidean loss function. Richard Zhang [2] In this paper he introduced an optimized solution by taking a huge data-set and single feed- forward pass in CNN. They used a custom multinomial cross entropy loss with class rebalancing and by using humans as subjects they were able to fool 32used prior color distribution obtained from the training set to predict a distribution for each output pixel. Baldassarre et al. [3] In this paper he made a network model that combines a deep CNN architecture, that is trained from scratch, with a pre-trained model Inception- Resnetv2 for high level feature extraction. They train this network on a small subset of 60,000 images from ImageNet. This architecture is similar to that used by Iizuka et al. [1] and it also uses Euclidean (L2) loss function. In [4] deep convolutional neural network architectures used are inherited from the VGG16 network. They implemented two models: one as a regression model and other as a classification model. They use the CIE LUV colorspace for input and output. They posed it as a classification task that can produce colorized images which are much better than those generated by a regression- based model.

# 2.2 PROPOSED SYSTEM

# Our objective is to convert an input grayscale image, which has a single channel of image data, into a typical RGB image, which has three channels of image data. the colorspace used by CIE

# Since it distinguishes between the brightness (intensity) and colour components of a picture, is utilised to represent the input and output images of the model.

# To map the input grayscale image to a coloured Lab space image, which is ultimately transformed to an RGB image, we train CNN models. In more detail, the goal is to learn a mapping from an input L channel (grayscale) image X RHW1 to the two associated colour channels (a, b channels) Y RHW2, where H, W are image dimensions. the anticipated.The predicted colourized Lab space image is then created by combining colour channels a and b with the input L channel image. We employ the mean squared error loss (often referred to as the Euclidean loss L2 function) between expected and actual Lab images as the objective function.

# OBJECTIVES

# 1. To grayscale RGB images and apply model to it.

# 2. To develop generators and discriminator to train the model.

# 3. To apply Convolution layers and LeakyRelu layers to the model.

# 

# PROBLEM FORMULATION

# The technology is growing day-by-day and still there exist many factors that have the scope of improvement. The existing systems have many cons few of which are solved an improved by our system. Mentioned below are few cons of existing system and the solution that our system provides to rectify these problems of exiting system.

**1)** We use the StandardScaler, rather than the MinMaxScaler as you might have seen before. The reason is that stock prices are ever-changing, and there are no true min or max values. It doesn’t make sense to use the MinMaxScaler, although this choice probably won’t lead to disastrous results at the end of the day.

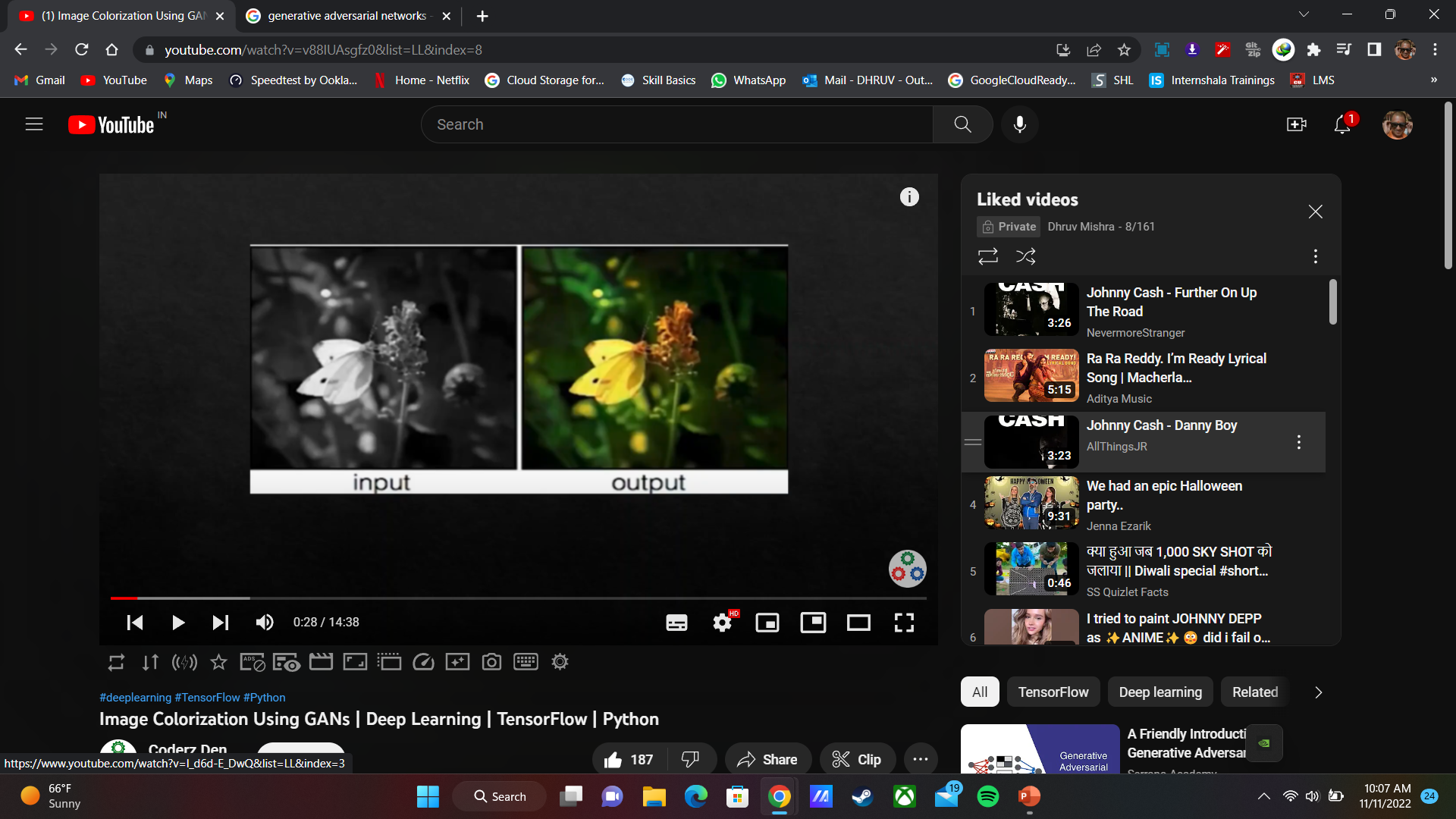
**2)** Stock price data in its raw format can’t be used in an LSTM model directly; we need to transform it using our pre-defined `extract\_seqX\_outcomeY` function. For instance, to predict the 51st price, this function creates input vectors of 50 data points prior and uses the 51st price as the outcome value.

# METHODOLOGY

Dataset: We will convert 3000 RGB photographs from different domains (mountains, forests, cities, erc) to grayscale so that they can serve as labels for our model.

* 1. We ran the generator once and the discriminator twice for a single step.
  2. For the discriminator, obtaining a good probability (closer to 1.0) for photos from the dataset and correctly classifying created images are both necessary to minimize its loss.
  3. The generator enhances itself to the point where it may deceive the discriminator by minimizing its loss. When the discriminator is tricked, created images will also produce probabilities (near to 1.0). 4. We’ll train the discriminator so that it produces probabilities for real photos that are more likely to be close to 1.0. (from our dataset) and output probabilities for images generated by the generator that are

1. closer to 0.0. 5. If the discriminator is sufficiently ”clever,” it will produce probability for genuine images that are closer to
2. 1.0 (coming from our database. Our generator is being trained to create such lifelike visuals that will make the even when the photos are fake, the discriminator’s output probabilities are closer to 1.0 (not from our dataset, but from the generator.



GANs generated image

**CONCLUSION AND FUTURE SCOPE**

**6.1** **CONCLUSION**

We implemented four different Deep Learning models for automatic colorization of grayscale images, two based on CNN and two based on GAN. We have shown that a simple CNN model outperforms large and complex Inception- resnetv2

based model on a small dataset. It was easily able to produce plausible colors for the high-level components in images like sky, mountains and forests but did not focus on smaller details. The inception-resnetv2 based model was too sophisticated for the simple dataset we collected which was one of the reasons for its poor performance, though [3] had shown this model works wellHowever, the Pix2Pix GAN models outperformed both CNN models providing high quality, nearly artifact free and vibrant output. Leveraging pre-trained Mobilenetv2 and Densenet121 allowed us to train high performance models with limited computational resources. Pix2Pix with MobileNetV2 was the most optimum in performance and compact enough to be deployed in a mobile device.

The generalisation power of the models is low due to the restricted data. Also, all these models perform poorly in complex scenes with fine details, though GAN models are still better than CNN based. A larger and diverse dataset would enable the model to learn a broader range of colour schemes increasing the generalisation power. Further, more complex models can be explored to mitigate the issues of colorizing small objects and fine details.

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks *. . .*”. Instead, try “R. B. G. thanks*. . .*”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

**6.2** **FUTURE SCOPE**

We will be able to colorize our grandparents old black and white images using this method.

# REFERENCES

1. *[1] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. ”Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification”. ACM Transaction on Graphics (Proc. of SIGGRAPH), 35(4):110, 2016.*
2. *[2] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” in ECCV, 2016.*
3. *[3] Baldassarre, F., Mor´ın, D. G., Rode´s-Guirao, L. (2017). Deep Koalarization: Image Colorization using CNNs and Inception-ResNet-v2. CoRR, abs/1712.03400. Retrieved from* [*http://arxiv.org/abs/1712.03400.*](http://arxiv.org/abs/1712.03400)
4. *[4] Jeff Hwang and You Zhou, “Image Colorization with Deep Convo- lutional Neural Networks”, Stanford University*
5. *[5] David Futschik, “Colorization of black-and-white images using deep neural networks”, January 2018.*
6. *[6] Alex Avery and Dhruv Amin, “Image Colorization”.CS230 – Winter 2018*
7. *[7] G. Larsson, M. Maire, and G. Shakhnarovich, “Learning represen- tations for automatic colorization”, CoRR, vol. abs/1603.06668, 2016.*
8. *[8] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to- image translation with conditional adversarial networks,” in Proceedings of the IEEE Conference on computer vision and pattern recognition, 2017, pp. 1125–1134.*
9. *[9] K. Nazeri, E. Ng, and M. Ebrahimi, “Image colorization using gen- erative adversarial networks,” in International Conference on Articulated Motion and Deformable Objects. Springer, 2018, pp. 85–94.*