DIY Projects

Colorize Black and White Images

Expected time to finish **8-14 Days**

Difficulty - **Advanced**

Introduction

Our aim is to obtain colorized images from greyscale images using GANs.

We are going to be implementing the Image Colorization with Generative Adversarial Networks Paper using PyTorch. The results turn out to be great and it serves as a great introduction to Generative Adversarial Networks (or GANs in short). ! Example:

Tech Stack

- **Python**
- **PyTorch**

Pre-Requisites

- Basic knowledge of PyTorch
- Knowledge about Convolutional Neural Networks

Resources

- Anaconda (to set up Python env):
  Install Anaconda and Jupyter on Windows (2020)

- Python with VSCode (or any IDE of your choice):
  How to Run Python in Visual Studio Code on Windows 10 2022 B...

- Deep Learning with PyTorch:
  Deep Learning With PyTorch - Full Course

- Concepts of CNN:
  MIT 6.S191: Convolutional Neural Networks

Introduction to GANs -

- Google Course: Introduction to GAN

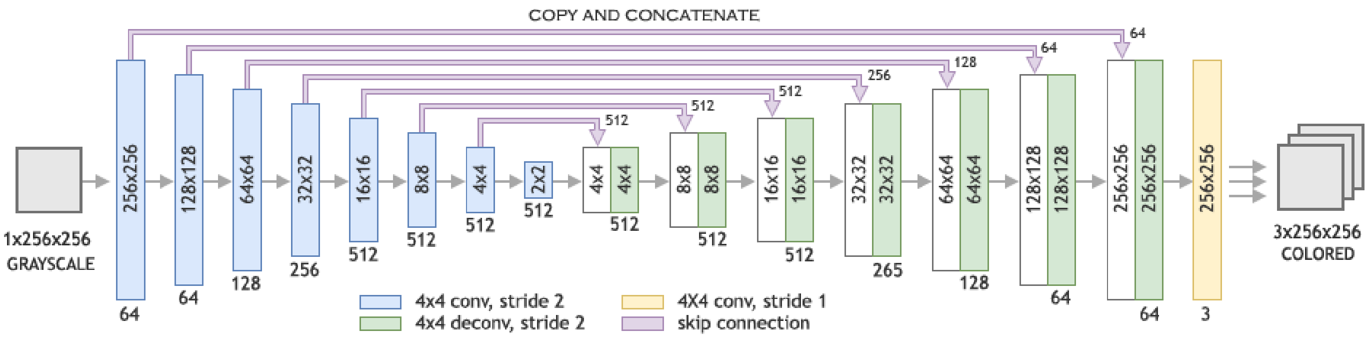
- Building a GAN From Scratch With PyTorch | Theory + Implement...

If you want to dig deeper into GANs:

- Generative Adversarial Networks paper
- Wasserstein GAN paper
- Coursera GAN specialization course

Implementation

Make sure to read the DCGAN paper (https://arxiv.org/abs/1803.05400) and understand the architectures of the Generator, Discriminator, and the Loss Functions.



The model by the authors was trained on datasets CIFAR-10 and Places365. You can use the Image Colorization dataset on Kaggle to make it more feasible for personal use.

The training might take some time, even in Google Collab, to get awesome results.

Additional Resources

https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html (understanding DCGAN in PyTorch)

https://github.com/ImagingLab/Colorizing-with-GANs (implementation in tensorflow)

https://github.com/harshitbansal05/Image-Colorization (implementation in PyTorch)

Note:

This project is not very easy in itself and requires reading up and a lot of patience while training and debugging. But that's the entire beauty of Deep Learning! With patience and resources, you will get astounding results! You may refer to the code from the links above but do not blindly copy the code. You should be able to understand each block of the architecture. All the best!

Submission

Form Link - https://forms.gle/sHmvzS1gT4LBxXa28