Colorization of Gray Scale Image

CMPE 258 - Deep Learning

Dr. Harry Li

TEAM MEMBERS

Name	SID	Email	Affiliation
Moxank Patel	016041064	moxankprakashbhai.patel@sjsu.edu	MS CMPE
Meet Patel	015971098	meetdineshbhai.patel@sjsu.edu	MSSE
Pratik Shah	016626779	pratikrajnikant.shah@sjsu.edu	MSSE
Swapnil Wagh	016161600	Swapnilsubhashrao.wagh@sjsu.edu	MSSE

TEAM COORDINATOR: Moxank Patel

ABSTRACT

Computer Vision has the exciting challenge of coloring grayscale images. Researchers have explored various methods to accomplish this task, such as the Scribble-based method and simple neural networks using Convolutional Neural Networks (CNN). One of the neural network-based approaches is Generative Adversarial Networks (GAN) [1], which involves a min-max game between a generator and a discriminator to minimize error. This project focuses on using GANs to colorize grayscales. This project aims to utilize GAN networks to colorize a given image. To improve training, we added an attention layer [2] concentrating on features and directly reducing generator errors. In addition, we added bump normalization and residuals to the network [3] to solve the missing gradient problem. After training for 100 epochs on V4 GPU Google Colab, we got a Mean Absolute Error of .15 on test data, which further can be reduced with training.

OBJECTIVE

The main objective of the project is to colorize a given grayscale image. To do this, we implemented a GAN network and trained the model on a dataset available on Kaggle [4]. In our architecture, we have incorporated residual CNN to deal with vanishing gradients, and along with that, to boost the training, we included an attention layer.

TECHNICAL CHALLENGES

1. Time-Consuming - A lot of time is needed during training because of the size of the image dataset and the architecture's almost 4M parameter count.
2. High Computational Power - A large dataset that cannot be stored in memory simultaneously, so the images to train were fetched ad hoc. The model was prepared using a V4 GPU with 16 GB capacity.
3. Overfitting: A primary challenge was training the model on train data without overfitting, so to deal with it, we added Gaussian noise, rotated images, increased brightness in the pictures, and randomly cropped them.

METHODOLOGY AND SOFTWARE/HARDWARE PLATFORM

1. We employed the GAN architecture for image colonization, which uses a generator to create the images and a discriminator to separate them. We have also used residual CNN modules to deal with the vanishing gradient issue.

2. The GAN network was implemented using tesnsorflow library. The other libraries used are cv2 for image processing, numpy for matrix calculation, sklearn for preprocessing and matlpotlib for visualization.
3. We used the V4 GPU of a Google Colab to train our model. It took around 10 hours to complete 100 epochs.
4. The loss function used is Min Max described in GAN paper [1] and in addition to it Mean Absolute Error is also being used as loss function for Generator.
   Min Max Function:

$$E_x[log(D(x))] + E_z[log(1 - D(G(z)))]$$

5. The metrics used for final evaluation is Mean Absoulte Error.

RESULTS AND DELIVERABLES

1. Feeding the input grayscale images to the trained GAN architecture-based model that can generate colorized images from input images.
2. After training, the generator loss is 0.76, while the discriminator loss is 1.39.
3. The error in terms of Mean Absolute Error was 0.15.

EXPERIENCE GAINED

1. We could comprehend the GAN architecture's complexity and the significance of maintaining a balance between the generator and discriminator models.
2. We have obtained knowledge using cutting-edge methods like residual dense modules and attention layers to boost the model's accuracy and the training process.

CONTRIBUTION

Moxank Patel	Researched GAN and Resnet architecture. Implemented Generator and hyper-tuned parameter. Contributed to the report.	25%
Meet Patel	Researched GAN and Resnet architecture. Implemented Discriminator architecture and hyper-tuned the parameters. Contributed to the report.	25%
Pratik Shah	Researched GAN and Resnet architecture. Implemented Loss function for Discriminator and Preprocessed Dataset. Contributed to the report.	25%
Swapnil Bagh	Researched GAN and Resnet architecture. Implemented Training Function and Test Function. Contributed to the report.	25%

REFERENCES

[1]. F. Marra, D. Gragnaniello, D. Cozzolino and L. Verdoliva, "Detection of GAN-Generated Fake Images over Social Networks," 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, USA, 2018, pp. 384-389, doi: 10.1109/MIPR.2018.00084.

[2]. Subakan, C., Ravanelli, M., Cornell, S., Bronzi, M., & Zhong, J. (2021). Attention Is All You Need In Speech Separation. ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021-, 21–25. https://doi.org/10.1109/ICASSP39728.2021.9413901

[3]. B. Li and Y. He, "An Improved ResNet Based on the Adjustable Shortcut Connections," in IEEE Access, vol. 6, pp. 18967-18974, 2018, doi: 10.1109/ACCESS.2018.2814605.

[4]. https://www.kaggle.com/datasets/theblackmamba31/landscape-image-colorization