Enhancement Of Image Quality Resolution Using CNN, Deep Learning

Jyothi Bhavani Patibandla, Praneeth Bonagiri, Nikitha Arumilli

Abstract

Automatic image colorization is a challenging task in computer vision that aims to generate plausible color versions of grayscale images. Traditional approaches often rely on manual intervention or complex hand-crafted features, making them laborious and time-consuming. In recent years, deep learning techniques, particularly Generative Adversarial Networks (GANs), have demonstrated remarkable success in various image synthesis tasks, including image colorization.This paper introduces a novel approach for image quality resolution utilizing convolutional neural networks (CNNs) and the power of Generative Adversarial Networks (GANs). Our proposed method aims to achieve high-quality colorized images with complex backgrounds through a fully automatic process, minimizing the need for user interaction.

Introduction

Image quality enhancement is kind of new and one of the most popular applications that are actively being researched, Image colorization is a complex and fascinating process aimed at transforming grayscale images into visually appealing and meaningful representations by attributing appropriate colors to them. This intricate task often demands a comprehensive understanding of the image content and the application of manual adjustments to achieve artifact-free quality. Given the numerous color possibilities for various objects, finding a unique solution to this problem is a challenging endeavor. Broadly, there are two primary methods for image colorization: one that necessitates user input to designate colors to specific regions, subsequently extrapolating this information to the entire image and another that learns each pixel's color from a similar image with color information. The proposed method provides an effective solution to the challenging problem of automatic image colorization, demonstrating its potential in applications such as photo restoration, digital media enhancement, and artistic rendering.

Dataset description

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is a comprehensive image database that organizes its content hierarchically. Each node within the hierarchy represents a specific category and is illustrated by hundreds or even thousands of images. The size of the imagenet database makes it take a considerable amount of time to train a model. An alternative is to use a subset of imagenet. We have curated a subset of the dataset by randomly choosing 100 classes, with each class containing 100 images. As a result, the subset consists of 10,000 images, offering a more manageable size for various applications.By considering this way we can achieve Reduced space and time complexity by taking into account the subset of ImageNet [1].

Project description:

Description

our proposed model contains a generator and a discriminator, we use General Adversarial Networks used to make a generative model by having two neural networks compete with each other. Our dataset comprises 10,000 RGB images originating from diverse categories such as mountains, forests, cities, and more. These RGB images will be converted into grayscale, serving as labels for our model. In a single iteration, we execute the generator once and the discriminator twice. Optimizing the discriminator's loss entails accurately classifying generated images and achieving a high probability (nearing 1.0) for images from the dataset. By minimizing its loss, the generator enhances its performance to deceive the discriminator, which implies generating probabilities (approaching 1.0) even for images created by the generator. We will train the discriminator to produce probabilities closer to 1.0 for genuine images (from our dataset) and probabilities nearing 0.0 for images generated by the generator. If the discriminator is sufficiently intelligent, it will yield probabilities closer to 1.0 for authentic images (from our dataset). Consequently, we will train our generator to craft such realistic images that prompt the discriminator to produce probabilities near 1.0, even when the images are fabricated (originating from the generator rather than our dataset).

The generator takes the input image and generates a grayscale synthetic image. We have used 3 convolutional 2d layers. Our generator denoted as G, accepts a grayscale image x as input and generates an RGB image, G(x). The input x is a tensor with the shape (batch size, 120, 120, 1), while the output G(x) possesses a shape of (batch size, 120, 120, 3). Our generator employs an encoder-decoder configuration akin to the UNet architecture. Furthermore, we incorporate dilated convolutions to expand the receptive field. To facilitate better information flow from the encoder to the decoder, we introduce skip connections within our model. Discriminator:4 convolutional 2d layers with relu activation were employed. The real picture y (from the training data) and the generated image G(x) (from the generator) are inputs to the discriminator model, denoted as D, which then outputs two probabilities. We hone the discriminator's abilities so that it can distinguish between created and actual photos. As a result, we train the model so that y produces a 1 and G(x) produces a 0.

Main references used for the project

Colorizing grayscale images is a complex task aimed at enhancing visual appeal and making images more perceptually meaningful. It typically requires prior knowledge about the image content and manual fine-tuning to achieve artifact-free quality. With multiple ways to assign colors to image pixels, no single solution exists for this issue.

Two main methods are used for image colorization: one that requires users to designate colors to specific regions, and another that learns pixel colors from a color image with similar content. In this study, the latter approach is employed, where color information is extracted from one image and transferred to another. Deep learning, particularly convolutional neural networks (CNNs), has recently garnered significant interest in the computer vision and image processing fields.

Advancements in computational resources, such as GPU computing power, have facilitated the training of very deep CNNs, leading to remarkable outcomes. Examples include a deep CNN surpassing human-level performance on ImageNet classification and an adversarial network generating convincing images by training two CNNs simultaneously. These impressive achievements inspire further exploration of CNNs' potential in addressing the image colorization challenge. The authors used 1000 iterations on a Torch implementation to colorize images of the Japanese art style known as ukiyo-e. When they examined the stochastic gradient descent (SGD) and L-BFGS optimization techniques, they found that L-BFGS produced somewhat superior images without the need for manual parameter adjusting. The visual quality of the generated photos for both methods was enhanced by lowering the style weight during iterations.

Another approach is The suggested colorization method emphasizes how crucial semantic data is throughout the colorization process. The system requires knowledge of the localization and semantic composition of the scene to colorize photos successfully. The pre-trained VGG-16 CNN model is employed in this study, with a few minor tweaks, as it contains a significant quantity of semantic data from having been trained on the ILSVRC 2012 classification dataset with more than a million photos [4].

An RGB picture with a fixed size of 224x224 is used as the input for the VGG-16 model. The input grayscale picture for single-channel grayscale images is concatenated three times. The initial half of the recovered layers from VGG-16 contain all layers up until the third pooling layer, whereas the second part is disregarded. The layers of the first half are upscaled to the input size of VGG-16 and concatenated, producing a matrix of the dimensions 224x224x451, designated by T.

This work uses multi-stage features to integrate complementary data from a previous layer to achieve a fuller representation. Performance in both prior works and this one has been enhanced by these characteristics. To preserve a 224x224 resolution, two convolutional layers are added after Matrix T. These layers' outputs are combined to create a matrix of the dimensions 224x224x144, or Q. The anticipated U and V channels are produced by the second of two additional convolutional layers that come after Matrix Q. The region of matrix Q associated with the second convolutional layer in the first stage is used during the backpropagation method [2].

Difference in approach between our project and the main projects of our references

In our reference paper, they used VGG16 for the colorization of images their approach was to Convert all training images from the RGB color space to the Lab color space. where they Use the L channel as the input to the network and train the network to predict the ab channels and Combine the input L channel with the predicted ab channels. finally, they Convert the Lab image back to RGB [3].

In our Project, we have implemented GAN consisting of a generator model and a discriminator model. The generator model takes grayscale images as input and generates colorized images. The discriminator model classifies whether an image is a real (ground truth) or generated (colorized by the generator) image. The generator and discriminator models are trained together, where the generator learns to produce more realistic colorized images, and the discriminator becomes better at distinguishing between real and generated images.

The generator model takes a grayscale image (120 X 120) as input. It consists of three sets of convolutional layers followed by a bottleneck layer, and then a series of transposed convolutional layers to upsample the feature maps and produce a colorized image (120 X 120).LeakyReLU activation functions are used in the convolutional layers, while ReLU activation functions are used in the transposed convolutional layers. and Skip connections are created using concatenation layers, which combine feature maps from earlier layers with feature maps from later layers. This helps the model to learn better by maintaining spatial information from earlier layers.

Another difference is the dataset they used is entirely different from ours We have chosen the imagenet dataset that consists of 100 classes with 100 photos each at random from the dataset. Consequently, the size drops to 10,000 images. and the images in the dataset contains all combination of images like black and white, grayscale, and colored RGB images.

Further, we have computed the metrics such as Mean Squared Error (MSE), Frechet Inception Distance (FID), and Structural Similarity Index Measure (SSIM) to evaluate the performance of the model. and we Saved the generated images and ground truth images to separate directories. Finally Calculated the accuracy of the model was based on the average SSIM score and a predefined threshold value [6].

Difference in accuracy between our project and the main projects of our references

our reference paper demonstrates the effectiveness of the proposed VGG-16-based method for automatic image colorization, it lacks specific quantitative details regarding its performance. There were no special calculations for testing how accurate the model was but they have just shown the generated colored instances. But to assess the effectiveness of our model, we calculated metrics including Mean Squared Error (MSE), Frechet Inception Distance (FID), and Structural Similarity Index Measure (SSIM). Based on the scores of SSIM we have taken those average scores and set up a threshold for filtering out colorized images that don't meet a certain quality level. However, SSIM and threshold-based accuracy measure different aspects of image quality. While SSIM measures similarity between the generated and ground truth images, the threshold-based accuracy compares pixel-wise differences. finally, we have got an accuracy of 85% for the considerable amount of 50 epochs of data.

To draw a direct comparison between the two methods in terms of performance. While both Models work on making images visually appealing colorizations, the lack of specific quantitative results in the VGG-16 paper and the differences in datasets and evaluation metrics make it challenging to determine which method performs better overall. However, both methods demonstrate the potential of CNNs for automatic image colorization and contribute valuable insights and techniques to the field. Also Using a pre-trained VGG-16 model for feature extraction might not be the most efficient choice, as it was originally designed for image recognition tasks, not colorization. Hence the U-Net architecture captures both low-level and high-level features from the input images. So we present a novel approach for automatic image colorization that combines the strengths of CNNs and GANs, resulting in realistic and visually appealing colorizations while maintaining the original structure and details of the input images.

Analysis

Techniques used to enhance the performance of GAN

1. Used Soft & Noisy Labels: Earlier models used hard labels in the range of 0.0 - 1.0. However, this would make the model less reliable in accuracy prediction because the range is very high, and this kind of label makes it hard and reduces model performance. Therefore, we implemented the usage of soft labels that are binary: 0 indicates fail, and 1 indicates success.
2. Used the Adam Optimizer: Instead of stochastic gradient descent, we used the Adam optimizer. This optimization technique can handle sparse gradients on noisy problems. Additionally, we can decrease the number of function evaluations to improve the efficiency of the algorithm and obtain better results.
3. Used 'relu' Activation Function: While training the discriminator, we utilized the 'relu' activation function and avoided the issue of sparse gradients.

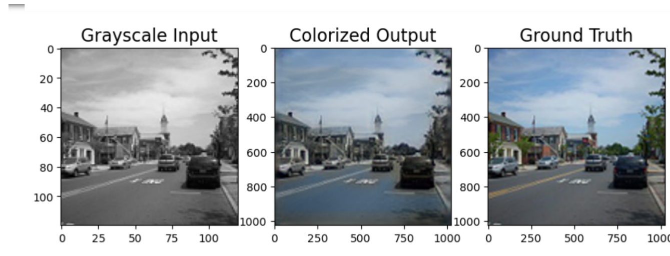

Figure 1: Image1

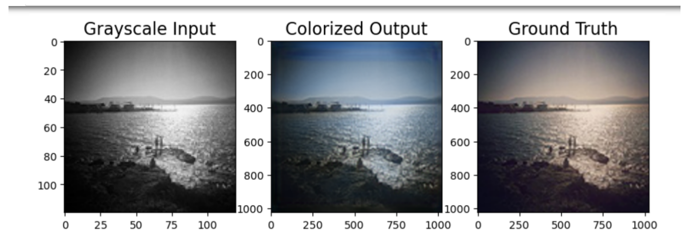

Figure 2: Image2

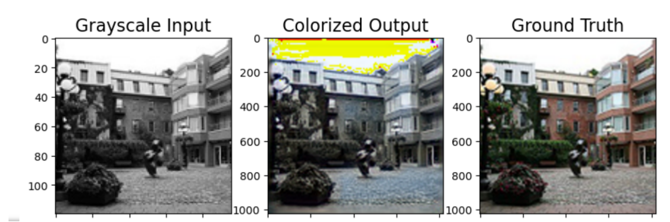

Figure 3: Image3

What could have been done better: U-Net Architecture: The unet is a fully convolutional network architecture and is used for fast image segmentation. we could have modified the architecture to get better output, but since it has outperformed all the best methods, we haven't made any changes to the architecture.

Future Work

As of now, we Implemented solutions for the colorization of images and their enhancement, similarly, colorization videos can also be done, though it is complex we achieve it by overcoming computational power and better optimization algorithms can be used [5]. In recent times, deep learning has

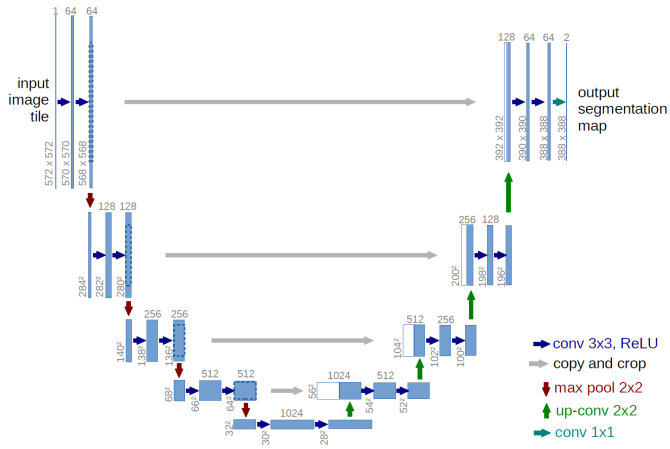

Figure 4: UNet Architecture

emerged as a powerful technique for automatically colorizing images while enhancing their quality. The proposed approach leverages the capabilities of neural networks to understand and process complex image patterns, enabling the transformation of images into vibrant, full-color versions. Deep learning models, such as convolutional neural networks (CNNs) and generative adversarial networks (GANs), have become popular tools for image colorization tasks. These models can effectively learn the intricate relationships between image features and their corresponding color information, resulting in more accurate and visually appealing colorized images [**m1**]. Currently, we have implemented solutions for image colorization and enhancement. Similarly, the colorization of videos can be achieved, although it is more complex. By addressing computational power challenges and employing improved optimization algorithms, we can successfully colorize videos as well. To further enhance image quality during colorization, deep learning models can also integrate techniques such as semantic segmentation and object recognition. This enables the model to understand the context of the image and apply more accurate and consistent colors across different objects and scenes. As research in this area continues to progress, we can expect even more sophisticated and efficient colorization methods to emerge in the future.

Conclusion

The paper proposes a reliable method for colorizing grayscale images using convolutional neural networks, we take the original RGB and grayscale images as input, and generator functions generate colored output this is used to verify our original rgb image using a discriminator. The results show we can enhance the quality of images with better colorization and a highly reliable model that can be further used for larger pixel images with high computational power. Extensive experiments are conducted on Imagenet datasets to evaluate the proposed method's performance. The results are compared against previous works in the field, showcasing the superiority of our approach in producing high-quality colorized images, especially when dealing with complex backgrounds. The experiments demonstrate the effectiveness of our multi-stage U-Net architecture, highlighting its ability to preserve fine details and enhance the overall image quality.Deep learning has greatly influenced the field of automatic image colorization and quality enhancement. Utilizing advanced neural networks architectures like CNNs, GANs, and encoder-decoder models, it is now possible to create vivid, realistic color images from grayscale versions. The paper proposes a reliable method for colorizing images using convolutional neural networks, we take the original RGB and grayscale images as input, and generator functions generate colored output this is used to verify our original rgb image using a discriminator. since the core of our framework is a multi-stage U-Net architecture-based CNN. and the adversarial training process facilitates the generator in producing visually appealing and realistic colorizations while improving the discriminator's ability to differentiate between real and generated high-resolution images. By leveraging the power of deep learning techniques, including CNNs and GANs, our proposed method offers a fully automatic image quality resolution process. The ability to achieve high-quality colorized images with minimal user interaction opens up new possibilities in various domains such as photo restoration, digital media enhancement, and visual content generation.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Communications of the ACM* 60.6 (2017), pp. 84–90.

[2] K Nimala, G Geetha, and J Godwin Ponsam. "Colorization of Black & White Videos & Photographs". In: *2022 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*. IEEE. 2022, pp. 1–6.

[3] K Kavi Niranjana and M Kalpana Devi. "RGB to lab transformation using image segmentation". In: *image* 3.11 (2015).

[4] Changyuan Qiu et al. "Automatic Image Colorization with Convolutional Neural Networks and Generative Adversarial Networks". In: ().

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer. 2015, pp. 234–241.

[6] Domonkos István Varga and Tamás Szirányi. "Convolutional Neural Networks for automatic image colorization". In: (2017).