Colorization Using Conditional-GANs

Team : HASHeD

Team Members

Arjun Singh
21111402
arjuns21@iitk.ac.in

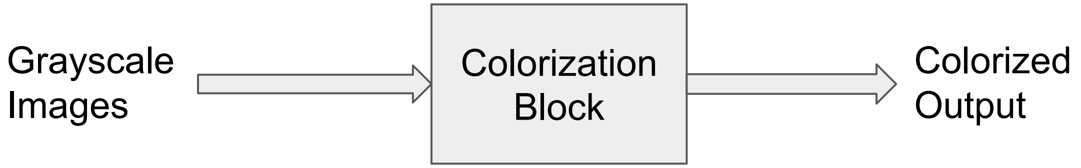
Debdeep Paul Chaudhuri
21111413
debdeeppc21@iitk.ac.in

Himanshu Lal
21111403
himanshul21@iitk.ac.in

Shivam Tripathi
21111408
shivamtr21@iitk.ac.in

INTRODUCTION

Image Colorization is a process of taking in grayscale images and producing colorized output of the same.



Motivation Behind Colorization: With advent of Deep Learning based approaches and rise in computational power, historical images, which are grayscale in nature can be readily converted to it's colorized equivalent. Colorization can find it's influence widely in reproduction of old astronomical images, electron microscopy and even b/w motion pictures for instance.

Literature Survey

For our project we had surveyed some of the seminal works in this domain, which provided us with direction to conduct experiments, addressing issues that're previously overlooked. We'll walkthrough the papers surveyed, what did they include and what did they missed.

1. **(Base Paper) Image-to-Image Translation with Conditional Adversarial Networks** (https://paperswithcode.com/paper/image-to-image-translation-with-conditional):

2. Image Colorization with Generative Adversarial Networks (https://paperswithcode.com/paper/image-colorization-with-generative)

3. Learning Representations for Automatic Colorization (https://paperswithcode.com/paper/learning-representations-for-automatic)

4. Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification (https://paperswithcode.com/paper/let-there-be-color-joint-end-to-end-learning)

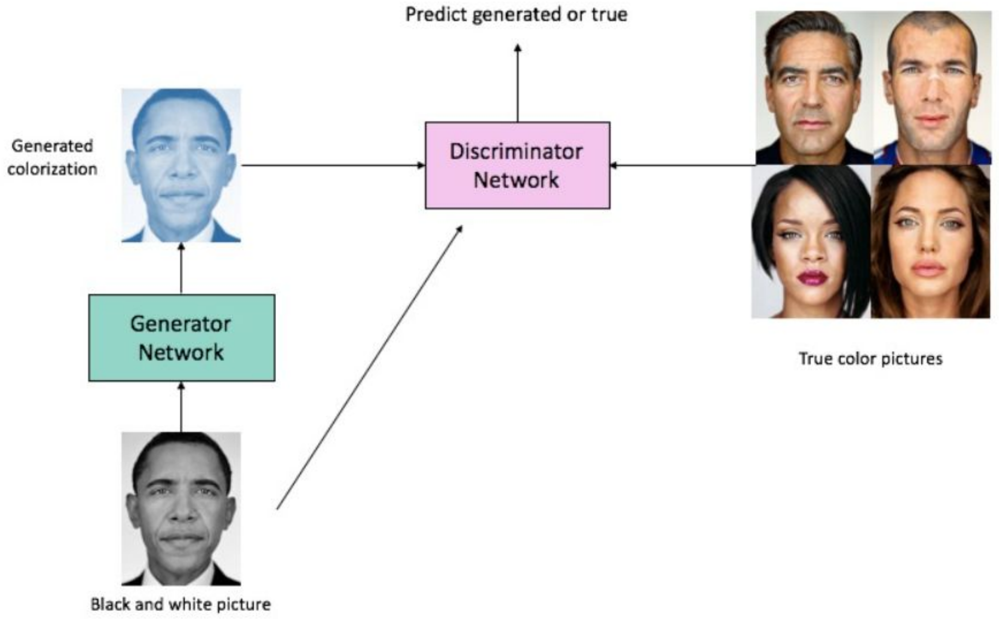
5. Color2Embed: Fast Exemplar-Based Image Colorization using Color Embeddings (https://paperswithcode.com/paper/color2style-real-time-exemplar-based-image)

6. ChromaGAN: Adversarial Picture Colorization with Semantic Class Distribution (https://paperswithcode.com/paper/chromagan-an-adversarial-approach-for-picture)

7. Deep Colorization (https://paperswithcode.com/paper/deep-colorization)

GAN : Generative Adversarial Network

- **GAN** consists of a **generator** and **discriminator** architecture, where the generator continuously tries to fool the discriminator, and the discriminator tries to find the truth.
- A **mini-max** game between generator and discriminator.
- The process ends with generator producing **realistic** output.
- This architecture fits in our solution as it produces colorized images on successful training.



Conditional-GAN (or c-GAN)

- **CGAN** or Conditional GAN's are GAN's + some added info that acts as a condition, grayscale images in our case.
- This facilitates learning as the **semantic nature** of grayscale image matches with that of GT image.
- The **new-generator** tries to map the grayscale image with some added noise to the output.
- Contrary to a vanilla GAN which just tinkered with the noise hoping to produce an output c-GAN has some **apriori information**.

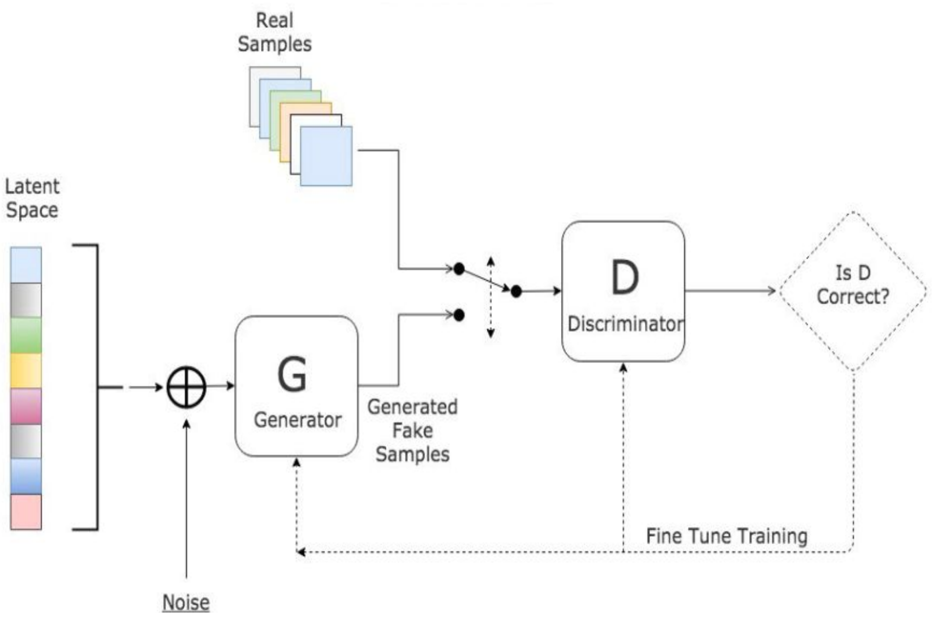
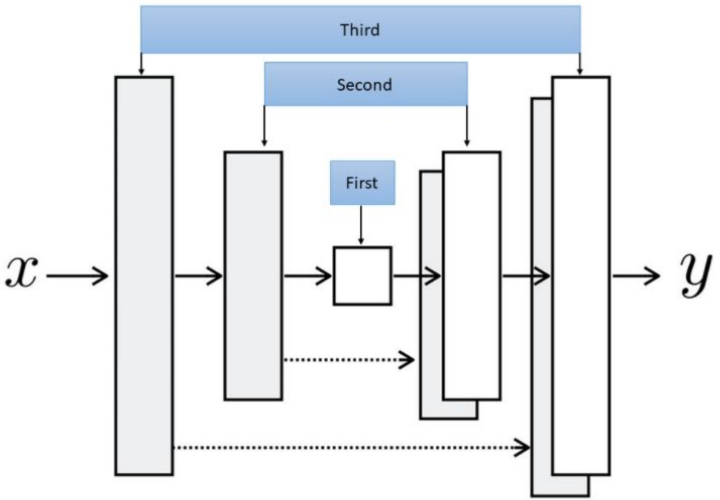
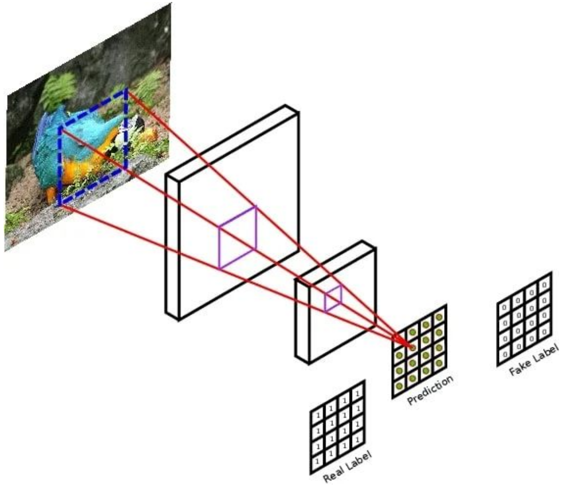


Image-to-Image Translation with c-GAN



Generator (U-Net)



Discriminator (PatchGAN)

Image-to-Image Translation with c-GAN

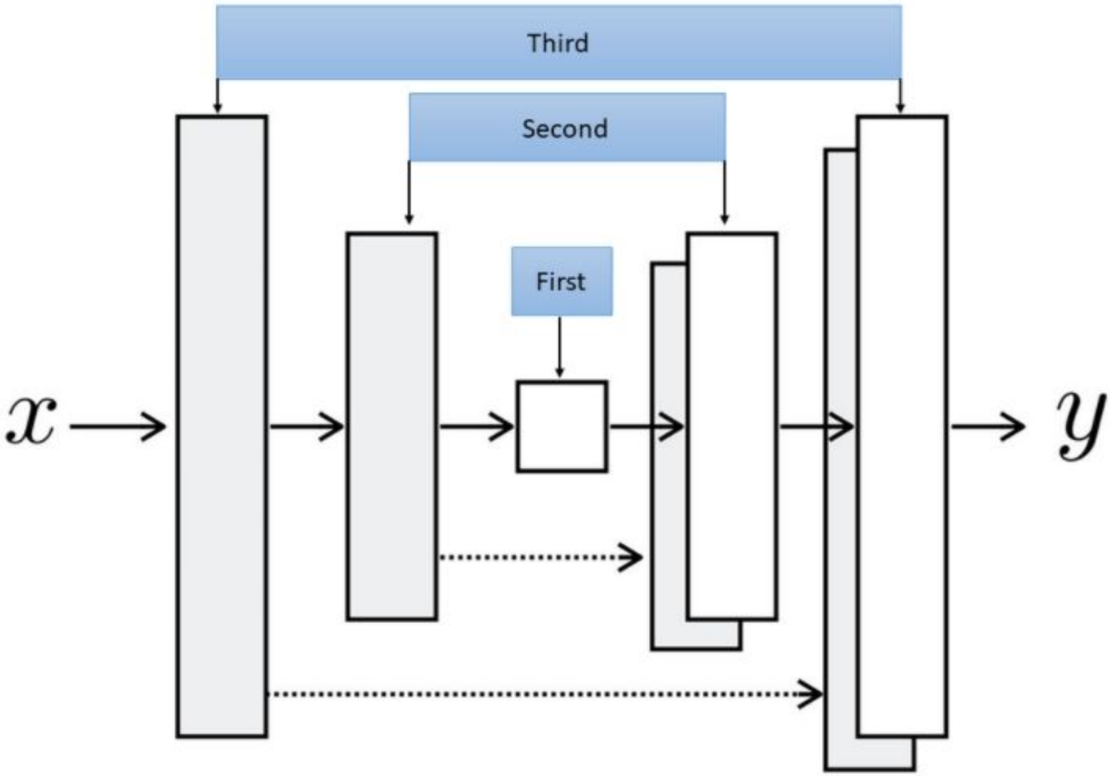
Adversarial Loss → $$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y}[log D(x, y)] + \mathbb{E}_{x,z}[log(1 - D(x, G(x, z)))]$$

L1 Loss → $$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[||y - G(x, z)||_1]$$

Final Objective → $$G^* = arg \min_G \max_D = \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G)$$

U-Net

- x is the grayscale input and y is the colorized output.
- It is a classic encoder-decoder framework with some additions like **skip-connections** b/w them.
- The skip connections enables sharing of **low level information** directly b/w layers aiding in image translation.
- In the case of **Image-colorization**, input and output share features viz. edges and hence skip connections makes sense.



Drawbacks of Existing Models

1. **Mis-colorization:** regions of images with high fluctuations are frequently colored green. This is likely caused by the large number of grassland images in the training set, thus the model leans towards green whenever it detects a region with high fluctuations in pixel intensity values.

2. **Low quality:** All models that we referred to in the literature survey did not increase the image quality. They are only colorizing the image without concerning about improving image quality.

Proposed Method

Loss Function:- Our loss function will have components of the **pix-to-pix** loss and content loss inspired from DeblurGAN and SRGAN methods. This might have a combined effect on the output generated, producing both colorization while also improving the quality of results produced, mitigating the erroneous colorization of low resolution images to some extent.

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y}[log D(x, y)] + \mathbb{E}_{x,z}[log(1 - D(x, G(x, z)))]$$

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[||y - G(x, z)||_1]$$

$$G^* = arg \min_G \max_D = \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G)$$

$$\mathcal{L}_X = \frac{1}{W_{i,j} H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^S)_{x,y} - \phi_{i,j}(G_{\theta_G}(I^B))_{x,y})^2$$

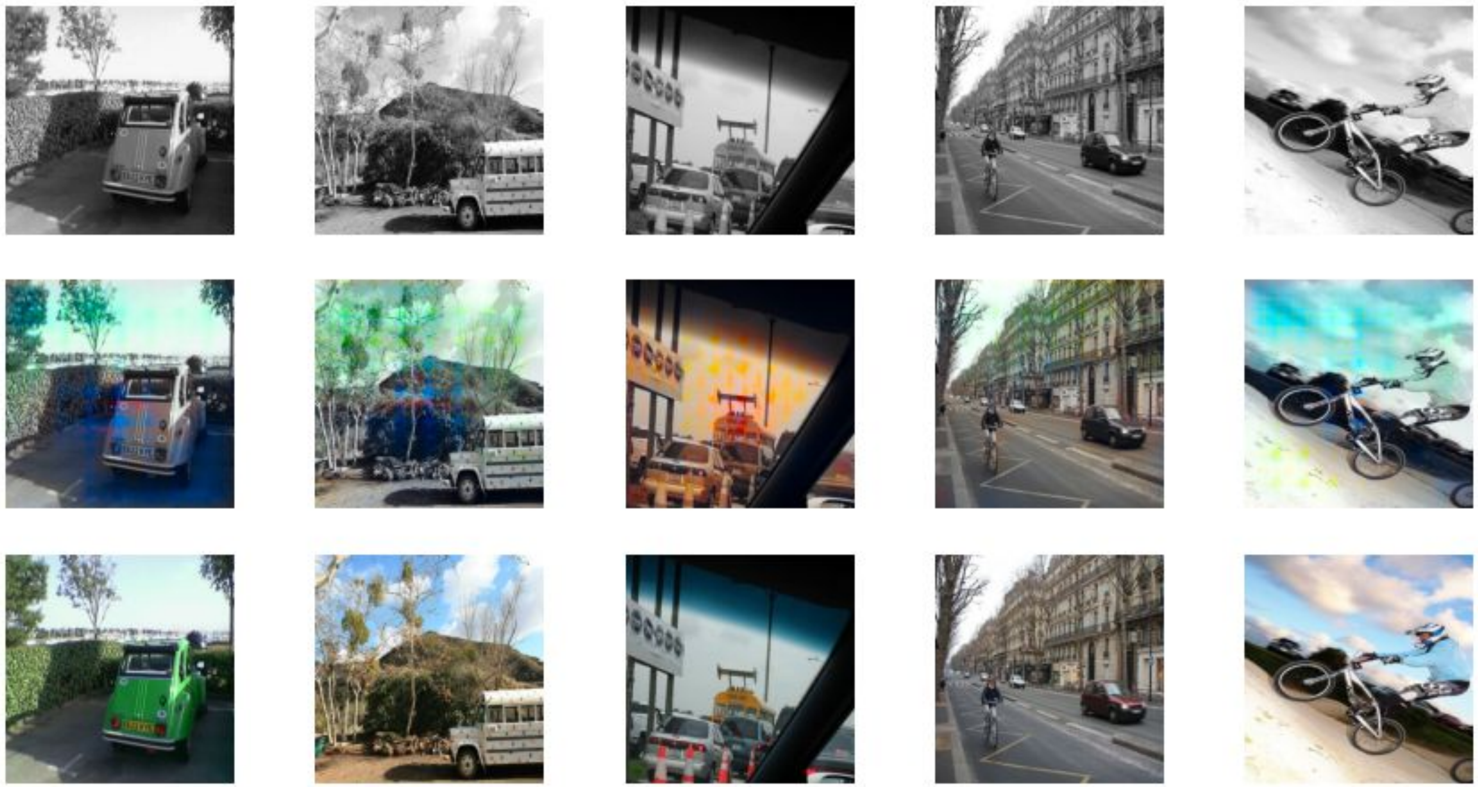
$$Loss\ Proposed = \gamma G^* + (1 - \gamma)\mathcal{L}_X$$

Preliminary Results

Grayscale Input

Generator Output

Ground Truth



Project Timeline

Week 1
Literature Survey

Week 2
Exploring GAN Based Methods

Week 3
Modifying Pix2Pix Image-to-Image Translation Mechanism for High-Quality Colorization

Week 4
Implementation of Pix2Pix

Week 5
Implementation of Proposed Method

Week 6
Fine Tuning

References

Thank You

Perceptual Loss

$$\mathcal{L}_X = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^S)_{x,y} - \phi_{i,j}(G_{\theta_G}(I^B))_{x,y})^2$$

Lx represents the perceptual loss to be used in our proposed method, where Φi,j indicates the feature map obtained by the j-th convolution (after activation) before the i-th max-pooling layer within the VGG19 network.

Lx calculates the mean euclidean distance between each pixel in the output feature maps produced by the VGG19.

In the proposed loss function Lx will calculate mean squared error between the feature maps of ground-truth image and generator result.

Evaluation Metric that can be used for colorization problems

MAE(Mean absolute Error) : Mean can be taken from predicted pixel values to ground truth pixel values. A certain threshold can also be included. Similarly MSE/RMSE loss can also be used.
This metric is being used in :

- Learning Representations for Automatic Colorization (RMSE)
- Image Colorization using generative Adversarial Networks.

PSNR(Peak Signal to Noise Ratio) : PSNR is the log of ratio of maximum possible value to the RMSE
This metric is being used in :

- ChromaGan
- Learning Representations for Automatic Colorization
- Super Resolution Nets

Accuracy : % of the correct pixels(with certain Threshold) can also be used as a valuable evaluation Metric
This metric is being used in :

- Deep Colorization