Exploring Unpaired Image-to-Image Translation

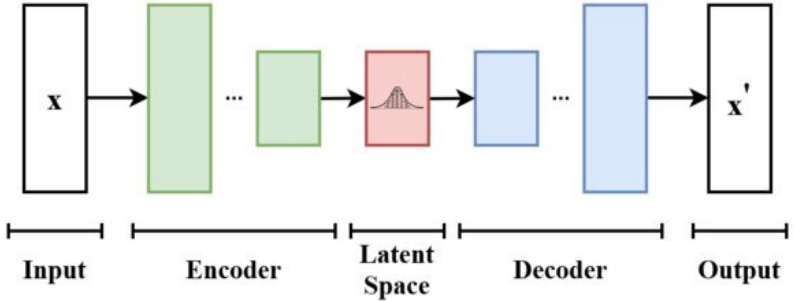
EE698R - Course Project

Course Instructor: Dr. Vipul Arora

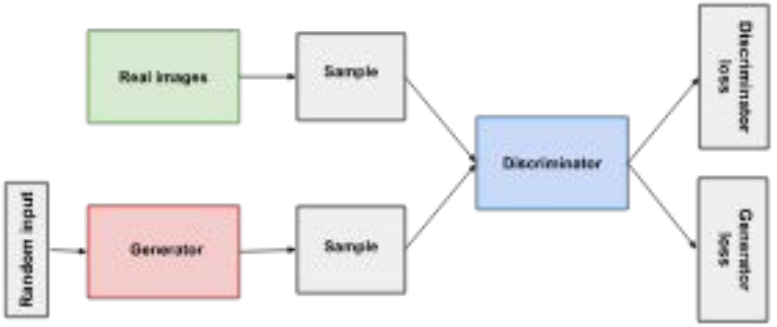
Rishi Agarwal (210849)
Shivam Sharma (210983)

Introduction

The primary aim of this project is to explore the robustness of relatively old breakthrough models in the field of generative learning on newer, more complex data and to understand the behind-the-scenes working of popular model architectures like VAEs and GANs taught in the class by implementing them from scratch.

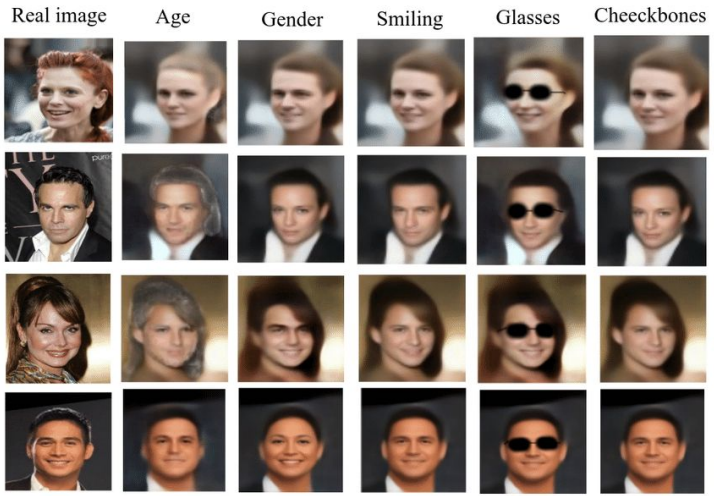


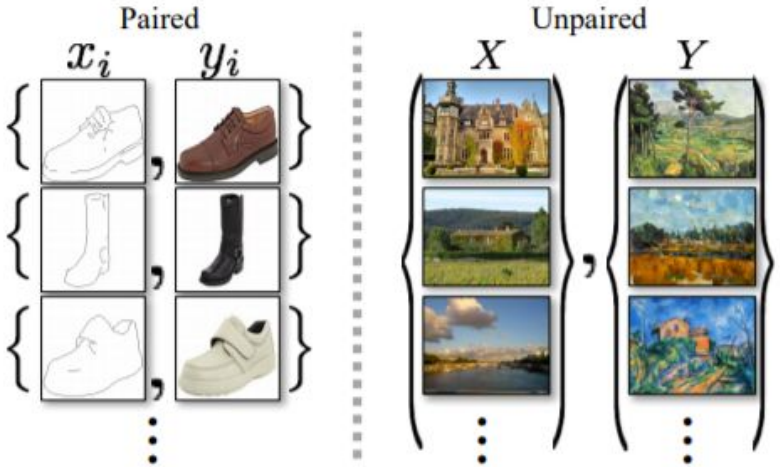
Sample VAE Architecture (Credit: Wikipedia)



Sample GAN Architecture (Credit: Google)

Problem Statement





The problem statement we have chosen is **unpaired image-to-image translation**. We aim to first verify the accuracy and results of models on originally used datasets, and then choose a **completely new and recent dataset** to perform a different translation task using these old models by tweaking the architecture accordingly to get the best results. Then, we do a comparison of these models to draw conclusions on their capability.

VAE/GAN
(Autoencoding beyond Pixels, 2016)

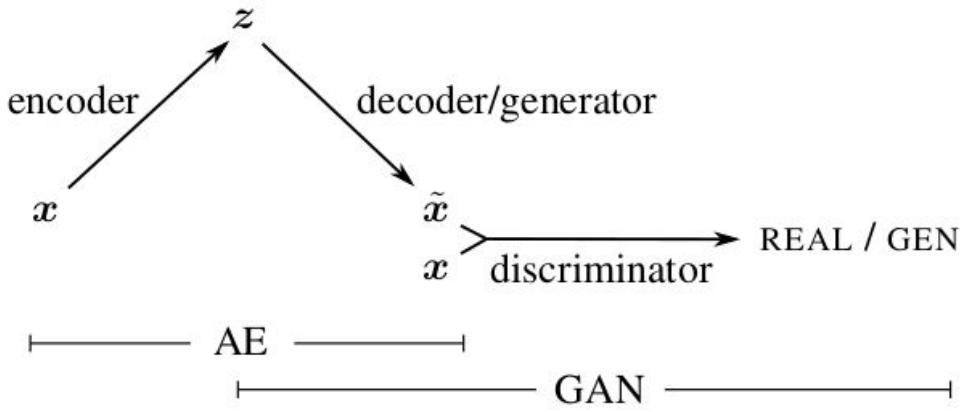


Figure 1. Overview of our network. We combine a VAE with a GAN by collapsing the decoder and the generator into one.

Algorithm 1 Training the VAE/GAN model

$\boldsymbol{\theta}_{\text{Enc}}, \boldsymbol{\theta}_{\text{Dec}}, \boldsymbol{\theta}_{\text{Dis}} \leftarrow$ initialize network parameters
repeat
  $\boldsymbol{X} \leftarrow$ random mini-batch from dataset
  $\boldsymbol{Z} \leftarrow \text{Enc}(\boldsymbol{X})$
  $\mathcal{L}_{\text{prior}} \leftarrow D_{\text{KL}}(q(\boldsymbol{Z}|\boldsymbol{X})\|p(\boldsymbol{Z}))$
  $\tilde{\boldsymbol{X}} \leftarrow \text{Dec}(\boldsymbol{Z})$
  $\mathcal{L}_{\text{llike}}^{\text{Dis}_l} \leftarrow -\mathbb{E}_{q(\boldsymbol{Z}|\boldsymbol{X})}\left[p(\text{Dis}_l(\boldsymbol{X})|\boldsymbol{Z})\right]$
  $\boldsymbol{Z}_p \leftarrow$ samples from prior $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
  $\boldsymbol{X}_p \leftarrow \text{Dec}(\boldsymbol{Z}_p)$
  $\mathcal{L}_{\text{GAN}} \leftarrow \log(\text{Dis}(\boldsymbol{X})) + \log(1 - \text{Dis}(\tilde{\boldsymbol{X}})) + \log(1 - \text{Dis}(\boldsymbol{X}_p))$
  // Update parameters according to gradients
  $\boldsymbol{\theta}_{\text{Enc}} \overset{+}{\leftarrow} -\nabla_{\boldsymbol{\theta}_{\text{Enc}}}(\mathcal{L}_{\text{prior}} + \mathcal{L}_{\text{llike}}^{\text{Dis}_l})$
  $\boldsymbol{\theta}_{\text{Dec}} \overset{+}{\leftarrow} -\nabla_{\boldsymbol{\theta}_{\text{Dec}}}(\gamma\mathcal{L}_{\text{llike}}^{\text{Dis}_l} - \mathcal{L}_{\text{GAN}})$
  $\boldsymbol{\theta}_{\text{Dis}} \overset{+}{\leftarrow} -\nabla_{\boldsymbol{\theta}_{\text{Dis}}}\mathcal{L}_{\text{GAN}}$
until deadline

Training workflow and model overview of the VAE/GAN model. The idea is to replace the generator of a GAN with a VAE so that the reconstruction objective of the VAE is replaced from the L2 norm between the recreated and original image, which is not practically sensible with a more sensible GAN loss which enables better attribute reconstruction. The final loss function looks like:

$$\mathcal{L} = \mathcal{L}_{\text{prior}} + \mathcal{L}_{\text{llike}}^{\text{Dis}_l} + \mathcal{L}_{\text{GAN}} \ .$$

Enc	Dec	Dis
5×5 64 conv. ↓, BNorm, ReLU	8·8·256 fully-connected, BNorm, ReLU	5×5 32 conv., ReLU
5×5 128 conv. ↓, BNorm, ReLU	5×5 256 conv. ↑, BNorm, ReLU	5×5 128 conv. ↓, BNorm, ReLU
5×5 256 conv. ↓, BNorm, ReLU	5×5 128 conv. ↑, BNorm, ReLU	5×5 256 conv. ↓, BNorm, ReLU
2048 fully-connected, BNorm, ReLU	5×5 32 conv. ↑, BNorm, ReLU	5×5 256 conv. ↓, BNorm, ReLU
	5×5 3 conv., tanh	512 fully-connected, BNorm, ReLU
		1 fully-connected, sigmoid

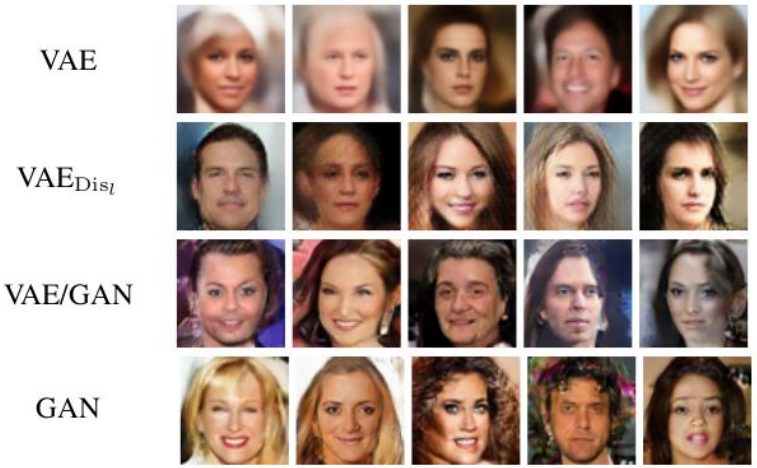


Figure 3. Samples from different generative models.

Model architecture of the original paper and a comparison of the VAE/GAN with normal VAE, VAE when the VAE is not optimized according to the discriminator, when it is tweaked acc. To the discriminator and a pure GAN model respectively.

We can see that the VAE/GAN is a huge improvement from normal VAE in terms of image clarity but still struggles to capture fine details and semantics correctly as compared to a pure GAN

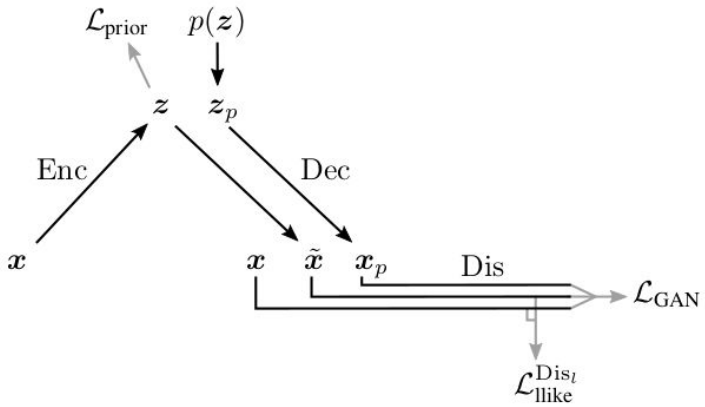
Figure 5. Using the VAE/GAN model to reconstruct dataset samples with visual attribute vectors added to their latent representations.

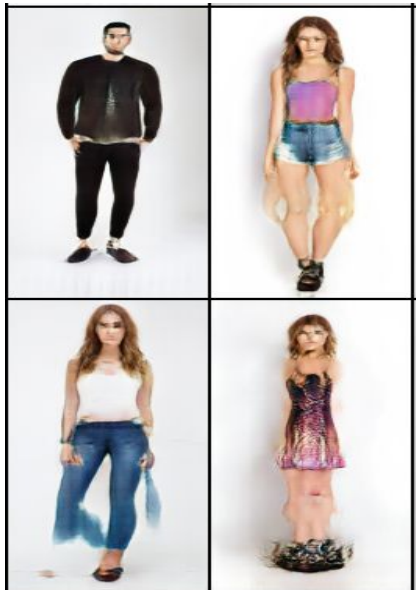
Style transfer task using VAE/GAN by averaging the positive and negative latent representations for each label

Our Additions

We have performed the following tasks for this model:

1. Implemented from scratch in pytorch
2. Verified the original results from the paper on the same data
3. Modified the model architecture and performed experiments to test the performance on a recent dataset (DeepFashion, 2022)
4. Compared the performance between the datasets and qualitative analysis of the same





Average latent representation of MEN_Jacket, WOMEN_Shorts, WOMEN_Jeans and WOMEN_Skirts respectively

Encoder	Decoder	Discriminator
5×5 64 conv. ↓, BNorm, L-ReLU	32·16·256 fully-connected, BNorm, L-ReLU	5×5 32 conv., L-ReLU
5×5 128 conv. ↓, BNorm, L-ReLU	5×5 256 conv. ↑, BNorm, L-ReLU	5×5 128 conv. ↓, BNorm, L-ReLU
5×5 256 conv. ↓, BNorm, L-ReLU	5×5 128 conv. ↑, BNorm, L-ReLU	5×5 256 conv. ↓, BNorm, L-ReLU
1024 fully-connected, BNorm, L-ReLU	5×5 32 conv. ↑, BNorm, L-ReLU	5×5 256 conv. ↓, BNorm, L-ReLU
	5×5 3 conv., tanh	512 fully-connected, BNorm, L-ReLU
		1 fully-connected, sigmoid

Table 2: Architectures for the three networks that comprise DeepFashion VAE/GAN. ↓ and ↑ represent down- and upsampling respectively. L-ReLU represents Leaky ReLU with linear region slope of 0.2

Original vs Recreated vs Rec + Random noise

MAN_Shirt

WOMAN_Skirt

Original

MAN_Sweater

WOMAN_Blouses

WOMAN_Cardigan

WOMAN_Shorts

WOMAN_Jacket

WOMAN_Dress

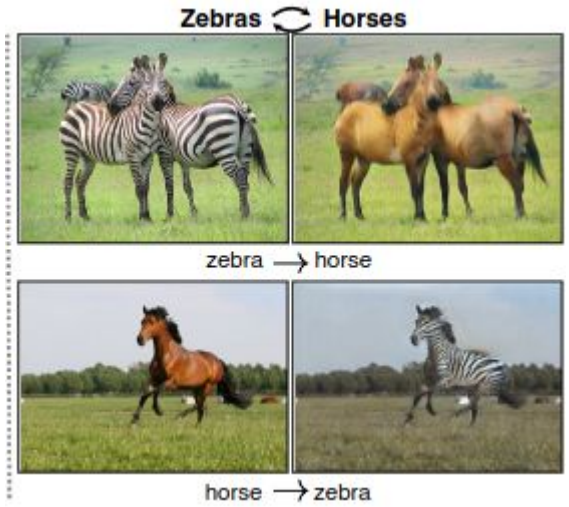
MAN_Sweater

Experiments and Conclusions

1. Increasing the latent space dimension from 128 to 256 introduced some improvement but further increase had minimal impact. This is probably because the dataset was not very extensive and also because the number of parameters required to be learnt increased significantly with z_size, hence the model simply did not train well.
2. We tried replacing the arithmetic of the latent representation feature translation from $\mathbf{Z \mathrel{+}= Z_{yes} - Z_{no}}$ to $\mathbf{Z \mathrel{+}= (Z_{yes} - Z_{no})_{target\_image} - (Z_{yes} - Z_{no})_{given\_image}}$ but that did not seem to have any effect.
3. We note that the reconstruction as well as the translation is less clear and more blurry for the DeepFashion data. This is expected as the model is trained on a much more diverse and varying data of clothes with different poses and clothing items, and also the image input size is significantly larger.
4. The attribute vector decoding and translation show that the model is still able to grasp the attributes correctly from the average encoding for this data.

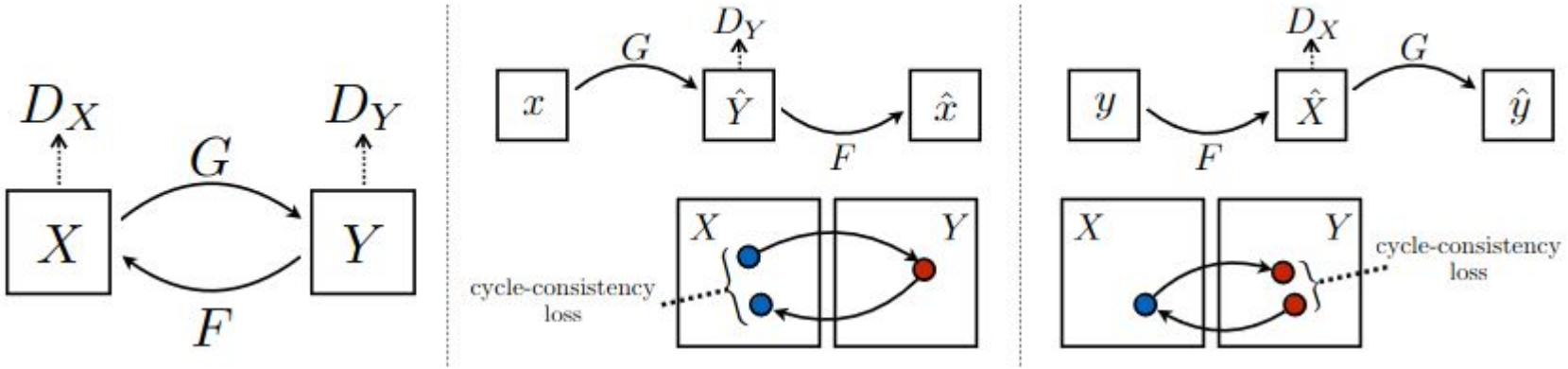
CycleGAN (Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, 2017)

Overview of CycleGAN



The paper modifies the original GAN architecture to learn specific characteristics from 2 sets of images and also to translate an image from one set to another

Method



Model has two generators G and F and two discriminators Dx and Dy

Training

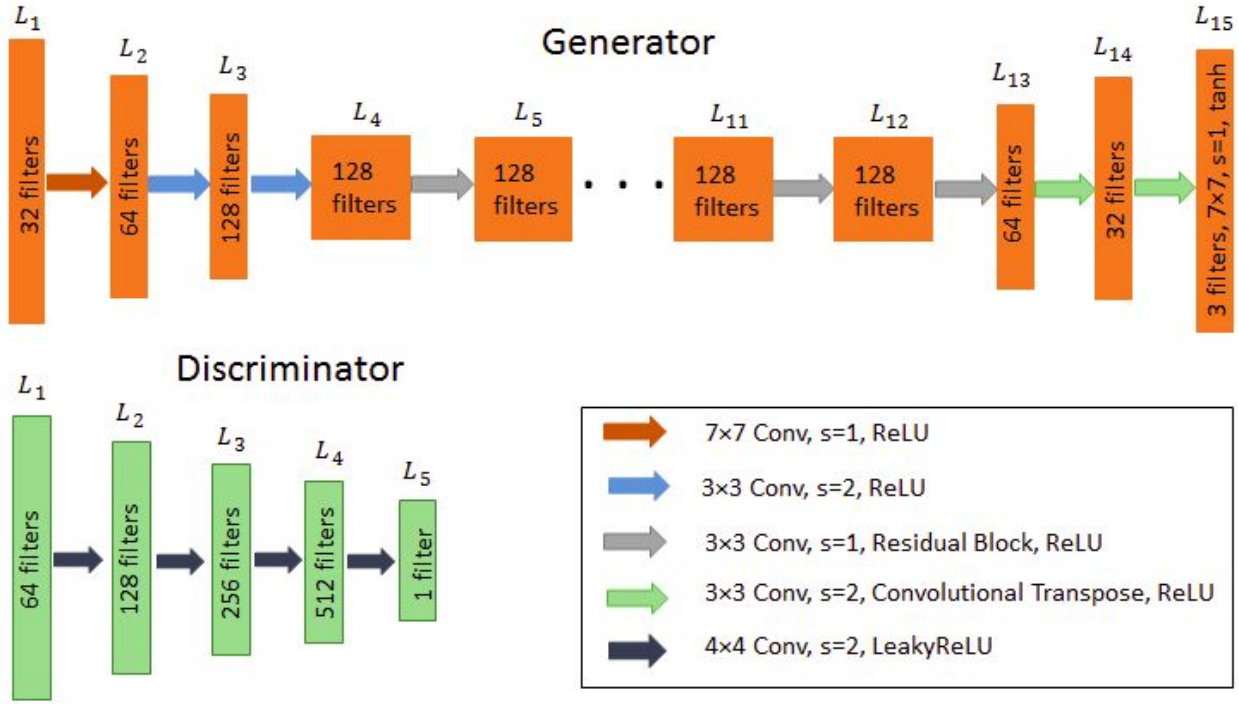
- Adversarial Loss

$$\mathcal{L}(G, D_Y) = \mathbb{E}_{y \sim p(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p(x)}[\log(1 - D_Y(G(x)))]$$
$$\mathcal{L}(F, D_X) = \mathbb{E}_{x \sim p(x)}[\log D_X(x)] + \mathbb{E}_{y \sim p(y)}[\log(1 - D_X(F(y)))]$$

- Cycle Consistency

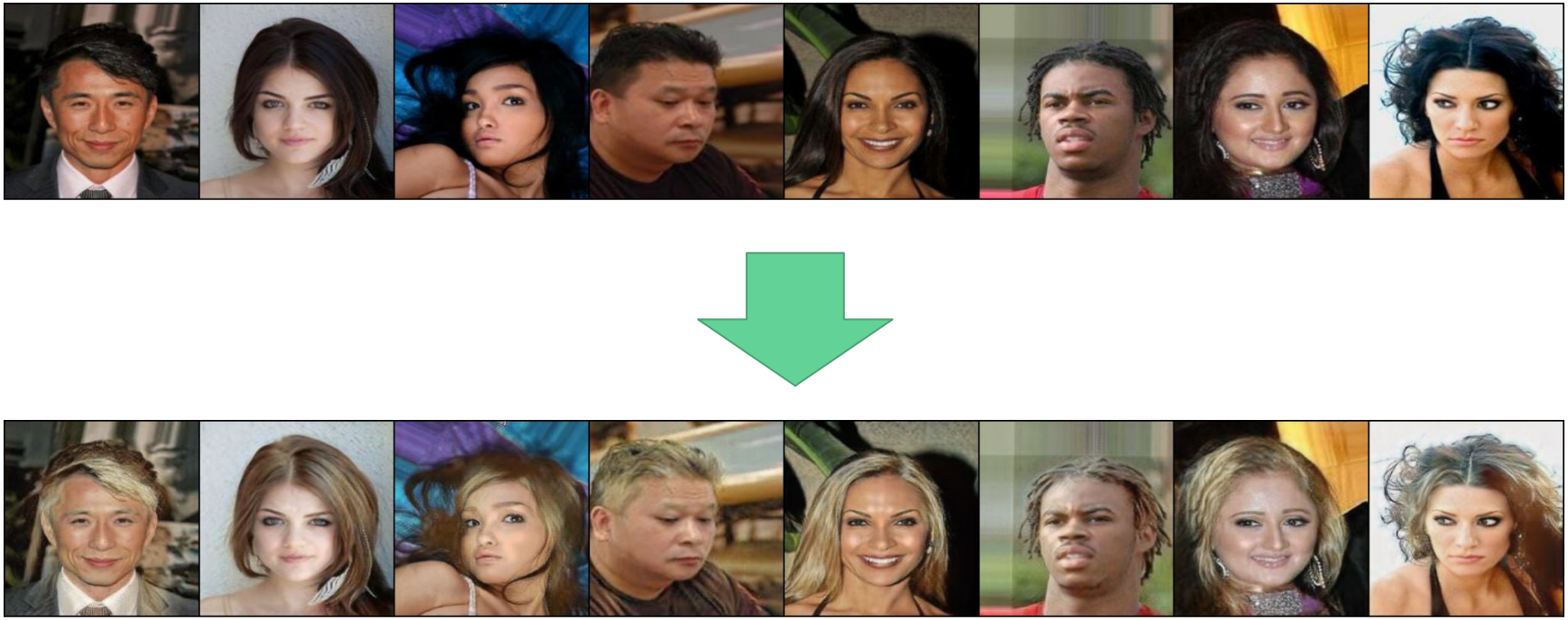
$$\mathcal{L}(G, F) = \mathbb{E}_{x \sim p(x)}[\|F(G(x)) - x\|] + \mathbb{E}_{y \sim p(y)}[\|G(F(y)) - y\|]$$

Network Architectures



Our Implementation and Experiments with CycleGAN

Experiments on CelebA



Experiments on CelebA

Experiments on DeepFashion

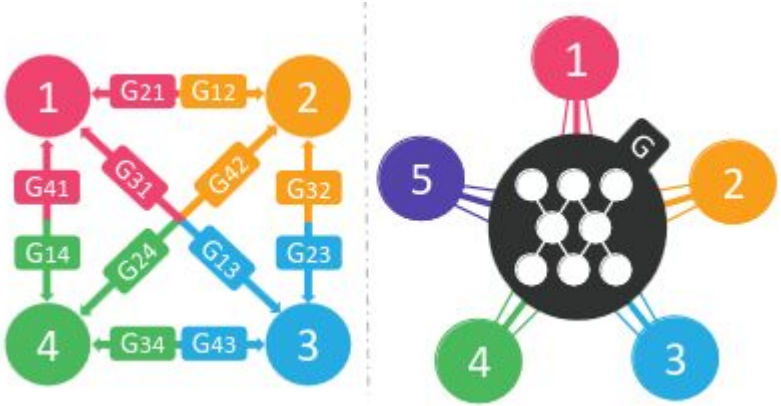
Experiments on DeepFashion

An Interesting Note...

A similar effect is observed in many attempts to generate sleeved clothing.

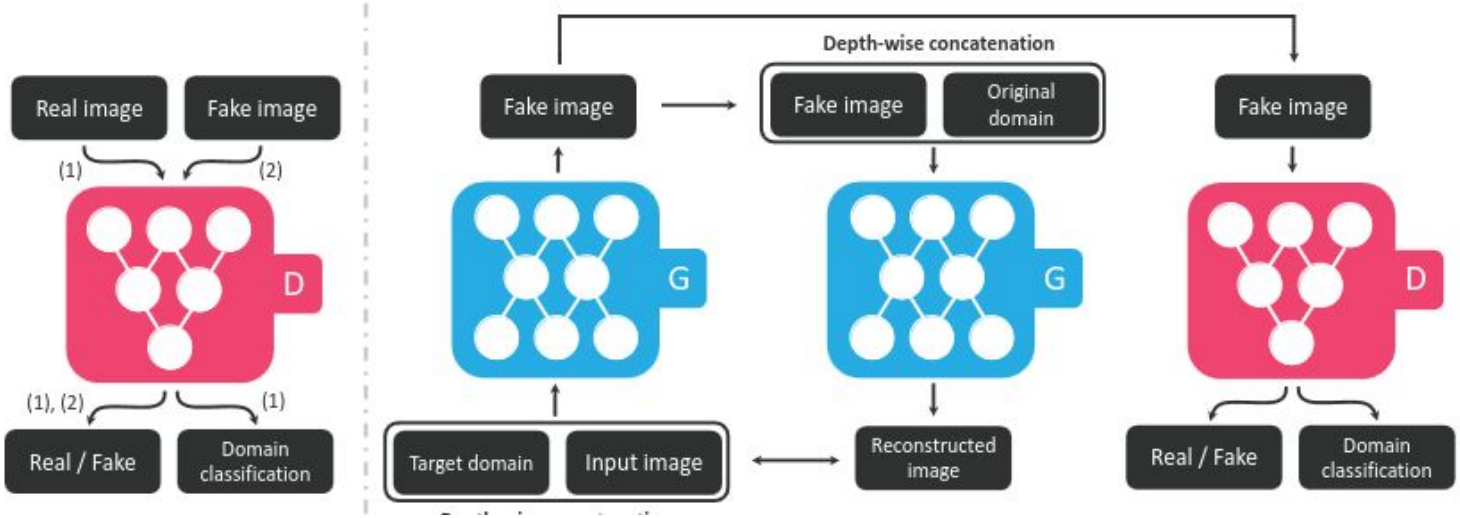
StarGAN
(StarGAN, 2018)

Overview



The paper aims to simplifies the architecture and reduce the compute to train models for doing multi domain image to image translation

Method



Generator takes as input the image and the target attributes

Discriminator outputs the probability of image being real/fake as well as the probabilities of target labels.

Training

- Adversarial Loss

$$\mathcal{L}_{adv}(G, D) = \mathbb{E}_x[\log D(x)] + \mathbb{E}_{x,c}[\log(1 - D(G(x, c)))]$$

- Domain Classification Loss

$$\mathcal{L}_{cls}(D) = \mathbb{E}_{x,c_r}[-\log D(c_r|x)]$$
$$\mathcal{L}_{cls}(G) = \mathbb{E}_{x,c}[-\log D(c|G(x, c))]$$

- Cycle Consistency

$$\mathcal{L}_{cyc}(G) = \mathbb{E}_{x,c_r,c}[\|x - G(G(x, c), c_r)\|]$$

Our Experiments and Results

Single Attribute Change

- Black to Blond Hair

- Blond to Black Hair

Multi Attribute Change

- (Male, Not Smiling) -> (Female, Black Hair, Smiling)

Bad Results Involving Complex Changes

- (Black Hair, No Bangs, Smiling) -> (Blond Hair, Bangs, No Smile)

Comments and Comparison

- VAE based model was comparatively easier to train in terms of hyperparameter adjustment, but also failed to capture the little details and generated blurry results.
- GAN based models were better at creating higher quality images but involved more hyperparameters which had to be tuned according to the dataset.

Future Work

- Due to time constraint and GPU availability, the StarGAN model could be tested on the DeepFashion dataset.
- Experimenting with more state of the art methods like denoising models and normalizing flows for image to image translation

THANK YOU