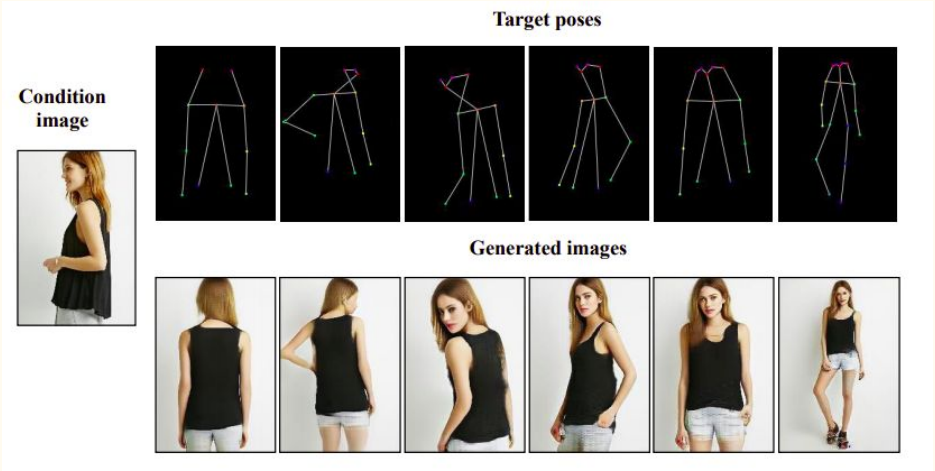
Human Pose Transfer

Prajval Nakrani, 17D070014
Parth Shettiwar, 170070021
Harekrissna Rathod, 17D070001
Utkarsh Bhalode, 17D070006

Introduction

Human Pose Transfer is a core computer vision problem where, as the name suggests, the task is to change the pose of a human given the target and input pose maps. This can be very useful in solving tasks like data augmentation for person re-identification. Furthermore, this can also be used in video generations with a sequence of poses.



Challenges

Human Pose transfer can be exceptionally challenging, particularly when given only the partial observation of the person. It is crucial for network to create each and every body part of human in correct configuration and at the same time should preserve the semantics of a normal human being. This is complicated by problems like occlusion, where the target pose is such that the human is required to rotate by a large angle resulting in network being asked to create some unobserved body parts. Finally, the texture on cloth and face regions have to be maintained in the output image. A GAN based network helps to achieve the above tasks since it can learn the underlying data distribution of the dataset and know the characteristic features of Human body.

Our Contribution

- We leverage the recent StyleGan Architecture to solve the problem of Human Pose Transfer using appropriate loses as mentioned in Progressive Attention Transfer Paper
- We show qualitative results on the Deep Fashion Dataset to evaluate the performance of our Pose Transfer Network

Related Work

The Pose Transfer problem is a relatively newer area of research. It is observed that in all the networks, a separate pipeline for Pose and Image is taken. The following are broadly the approaches which people have followed in past few years to solve this problem:

1)Pixel-level feature warping: Estimates a pixel wise flow map by leveraging Dense Key Point estimations. Computationally expensive technique.

2)Piecewise Affine Transform: Divide the human skeleton into rigid parts and approximates the warping function with piece-wise affine transformation

3)Using Progressive Attention: Let the network learn the mapping between input and target body parts and their poses using attention blocks. No computation of any flow maps.

4)Using AdaIn layer: Adaptive Instance normalization originally used for style Transfer. These methods model the Pose transfer problem as a Style Transfer one and incorporate the styles (encodings) information from pose to generate target image.

Problem Statement

Given a reference person image x and a target pose p , our goal is to generate a photorealistic image x’ for that person but in pose p.

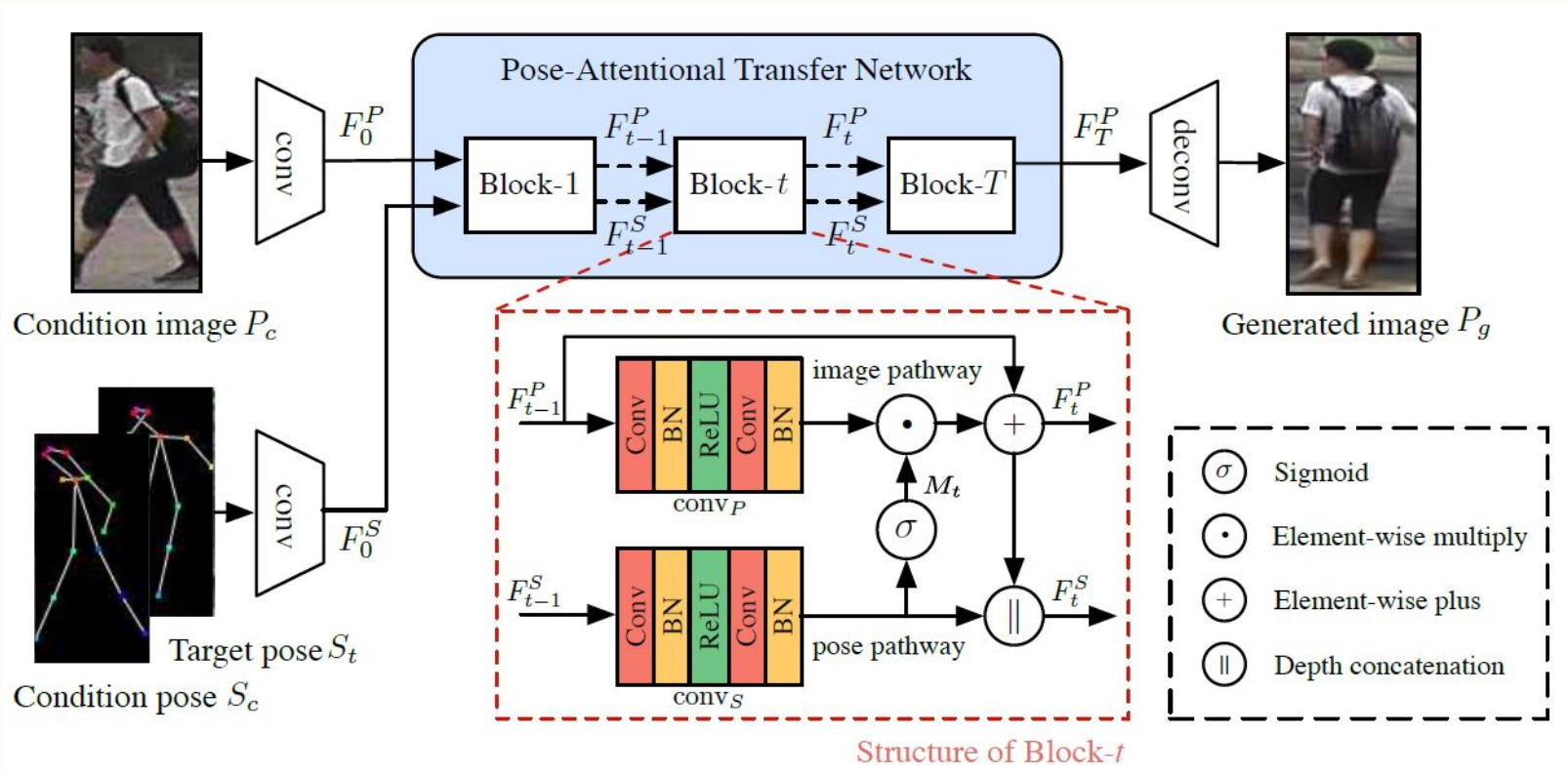
More formally, we consider the image pair (x1, x2) (source and target) with their corresponding 18 channel pose heatmaps (p1, p2). The model should takes the triplet (x1, p1, p2) as inputs and generate x2’.

Pose Information: Using a pose estimator (Cao et. al.), we extract 18 human keypoints. The keypoints are encoded into a 18-channel binary heatmap, where each channel is filled with 1 within a radius of 8 pixels around the corresponding keypoint and 0 elsewhere.

Progressive Pose Attention Transfer Network - PATN

- Perspective - images of all the possible poses and views of a certain person constitute a manifold in the image space
- Pose Transfer = traversal from a particular initial point p_x on manifold to another point p_y on the same pose manifold
- The challenges in the Pose Transfer problem can be attributed to the the complex nature and structure of the above mentioned pose manifold
- Insight - Local structure of the manifold can still be simple
- Idea - Break a pose transfer problem with large variation into smaller pose transfer problems with less pose variation in each step.
- Motivation - *Progressive* Pose transfer scheme for solving the problem

PATN - Architecture Details

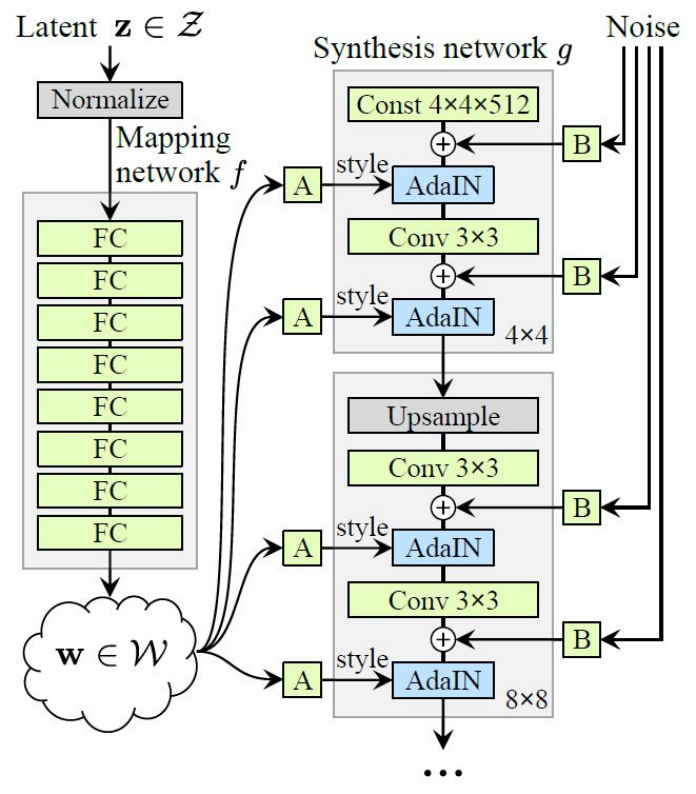


StyleGAN - An Overview

- A smartly designed Generator.
- The StlyeGAN work is orthogonal to the research of GAN loss functions, regularization, hyper-parameters etc.
- The proposed architecture leads to an automatically learned, unsupervised separation of high-level attributes of an image like pose, identity etc. and finer and stochastic details like freckles, hair etc.
- The architecture also enables an intuitive scale-specific control of synthesis
- Introduces a new method of style transfer at the level of feature maps via the use of Adaptive Instance Normalization - AdaIN.

$$\mathrm{AdaIN}(\mathbf{x}_i, \mathbf{y}) = \mathbf{y}_{s,i} \frac{\mathbf{x}_i - \mu(\mathbf{x}_i)}{\sigma(\mathbf{x}_i)} + \mathbf{y}_{b,i}$$

StyleGAN - Architecture

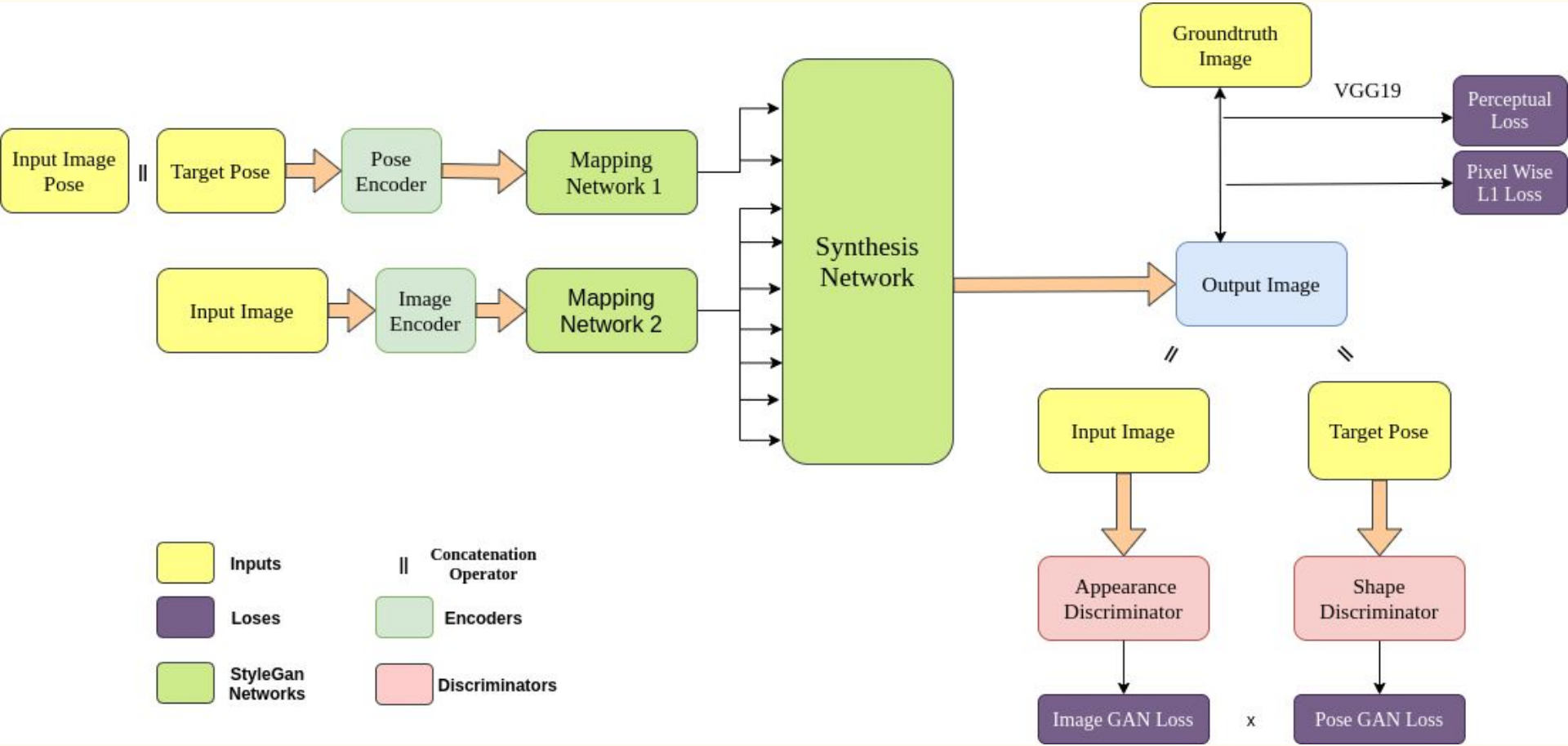


- The Generator takes a learnt constant input as content input and then adjusts the style of the image at various levels of representation at each conv layer.
- There is also a provision of addition of noise at each layer for controlling the more stochastic variations in the image.
- The architecture also uses an FC network for disentangled space representation of input latent space that might have unavoidable entanglements.

Our Approach - Combining PATN and StyleGAN

- We saw that PATN provides a way of transferring image from one pose to another progressively using a series of PAT blocks.
- However, Pose Transfer can be viewed as a form of Style Transfer. Where, the pose of the image is the main style that our architecture focuses on.
- Now, StyleGAN provides a superior method of style transfer by using disentangled representations of the input latent space and using AdaIN layers for effective style transfer at feature level
- Hence, we use the training concept and problem formulation as proposed by PATN however, we use the StyleGAN architecture for purpose of pose transfer.

Our Architecture



Architecture Overview

- Image Encoder - Learns a 512-dimensional latent code for the image space
- Pose Encoder - Learns a 512-dimensional latent code for the pose space
- Image Mapping Network - Learns a 512-dimensional disentangled representation of the image latent space
- Pose Mapping Network - Learns a 512-dimensional disentangled representation of the pose latent space
- Synthesis Network - Takes a learnt constant as the content input and applies styles via AdaIN on affine transformations of the disentangled representations
- Idea - The top two coarse levels are controlled by the pose latent space. The top seven finer levels or bottom 7 coarser levels controlled by the image latent space
- Human Pose Estimator (HPE) is used for estimation of 18 joint locations in human body that potentially represents the pose of a person.

Discriminators

- We train two distinct discriminators like PATN viz.
- 1) Appearance Discriminator (D_A) - It judges whether P_g (Generate Image) contains the same person present in P_c(Input Image)
- 2) Shape Discriminator (D_S) - It judges whether P_g contains the same pose as represented by S_t(Target Pose)
- We incorporate the scores from both the discriminators in our loss functions while training the network.

Loss Functions

- Full Loss Function - $$\mathcal{L}_{full} = \arg\min_{G}\max_{D} \alpha\mathcal{L}_{GAN} + \mathcal{L}_{combL1}$$

- Adversarial Loss - $$\begin{aligned}\mathcal{L}_{GAN} =& \mathbb{E}_{S_t\in\mathcal{P}_S,(P_c,P_t)\in\mathcal{P}}\{\log[D_A(P_c,P_t)\cdot D_S(S_t,P_t)]\}+\\ & \mathbb{E}_{S_t\in\mathcal{P}_S,P_c\in\mathcal{P},P_g\in\hat{\mathcal{P}}}\{\log[(1-D_A(P_c,P_g))\\ & \cdot(1-D_S(S_t,P_g))]\}.\end{aligned}$$

- Combined Perceptual and L1 loss - $$\mathcal{L}_{combL1} = \lambda_1\mathcal{L}_{L1} + \lambda_2\mathcal{L}_{perL1}$$

$$\mathcal{L}_{perL1} = \frac{1}{W_\rho H_\rho C_\rho}\sum_{x=1}^{W_\rho}\sum_{y=1}^{H_\rho}\sum_{z=1}^{C_\rho}\|\phi_\rho(P_g)_{x,y,z} - \phi_\rho(P_t)_{x,y,z}\|_1$$

Datasets

1. *In-shop Clothes Retrieval Benchmark* DeepFashion Dataset

   It comprises of human posed images with white clear background.

   It contains high resolution images (256x256).

   It is an easier dataset due to high resolution and clean background.

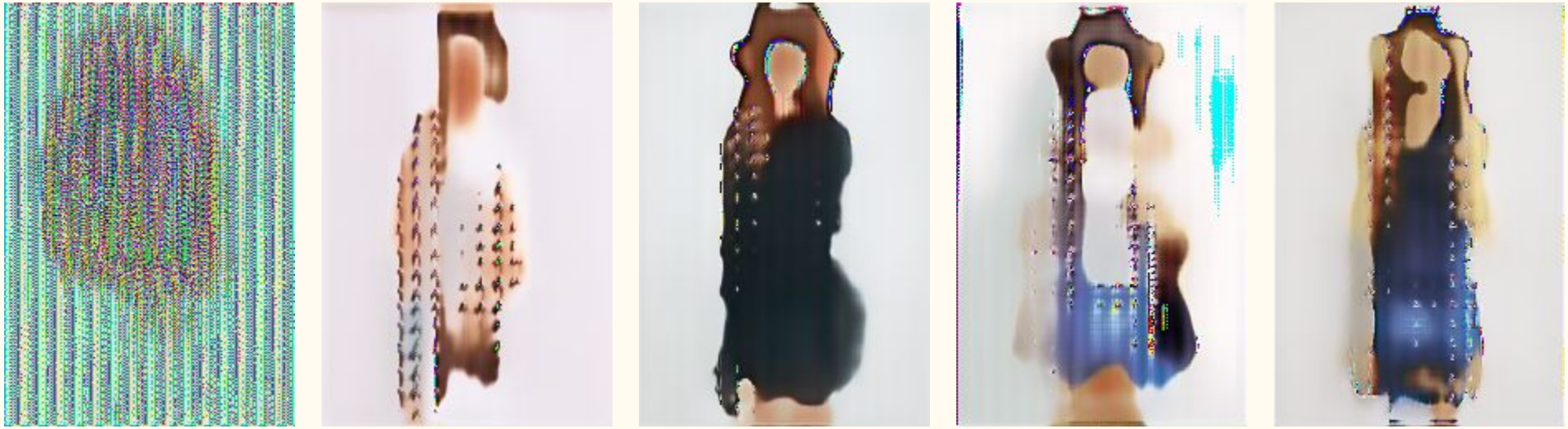
2. Market-1501 Dataset

   It comprises human pose images with regular background.

   It contains low resolution images (64x64).

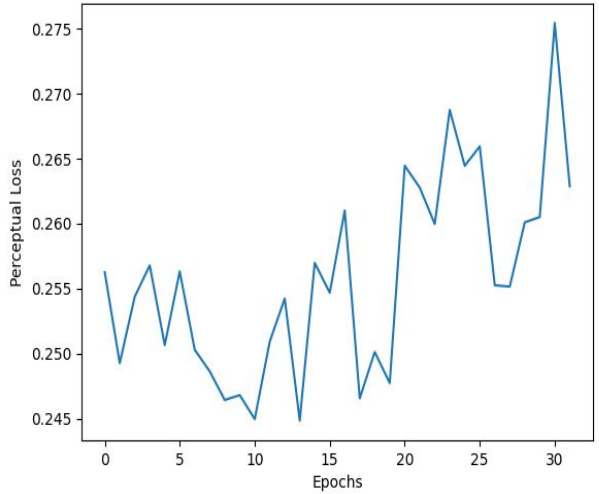
   It is a challenging dataset due to low resolution and non-uniform background.

Results

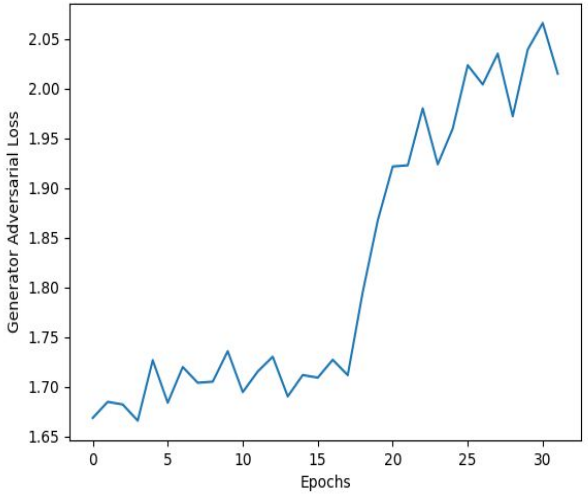


Epoch 1 | Epoch 11 | Epoch 19 | Epoch 24 | Epoch 32

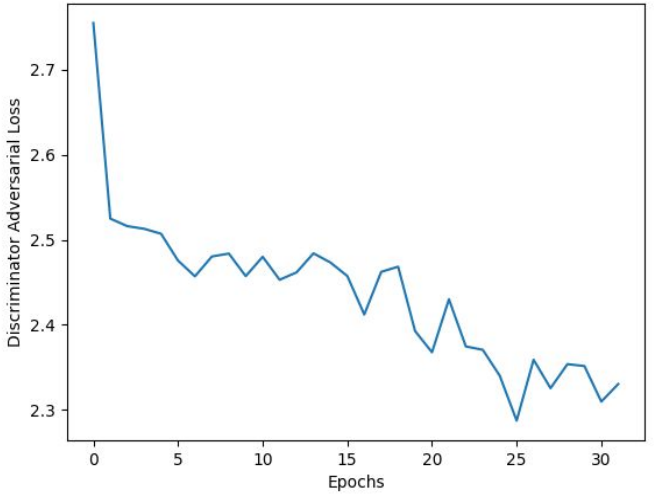
Training Curves



Perceptual Loss



Generator Adv Loss



Discriminator Adv Loss

Thank You