

STYLE GAN : A Style Based Generator Architecture for Generative Adversarial Networks

Vyshak Puthusseri

College of Engineering, Trivandrum
TVE17MCA054

23th Oct 2019

Introduction

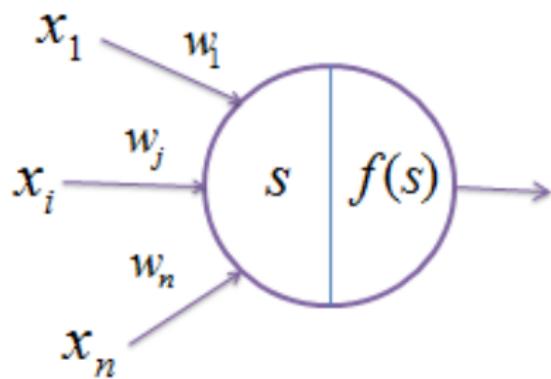
- 1 Deep Learning
- 2 Style Transfer
- 3 GAN : Generative Adversarial Network
- 4 Style Based Generator Architecture for GAN
- 5 Conclusion

Deep Learning - HOW?

LTU : Linear Threshold Unit

Building blocks of neural networks Proposed by Warren McCulloch and Walter Pitts

Only a concept, No learning strategy



Deep Learning - HOW?

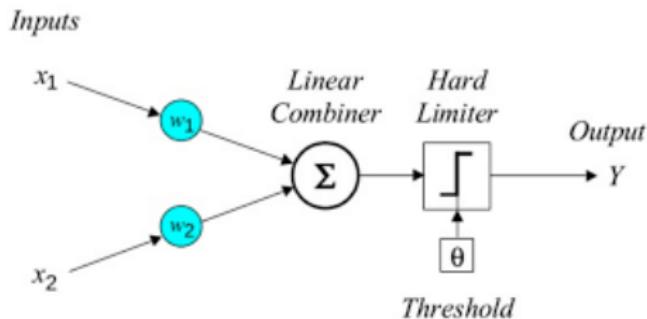
Perceptron

- LTU + Learning rule .

Deep Learning - HOW?

Perceptron

- LTU + Learning rule .
- Works only for binary classification



Deep Learning - HOW?

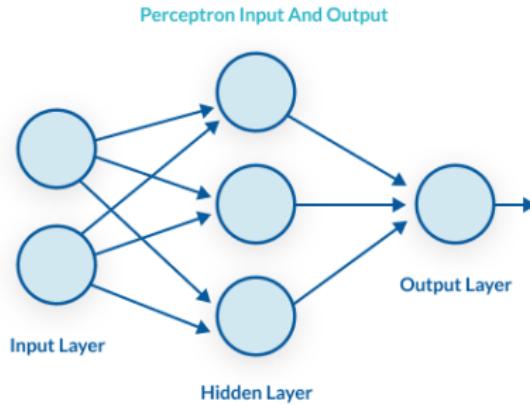
Multilayer Perceptron

- Multiple LTUs are stacked side by side and on top

Deep Learning - HOW?

Multilayer Perceptron

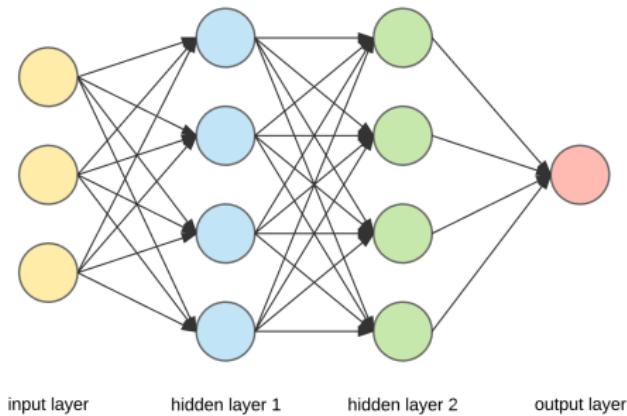
- Multiple LTUs are stacked side by side and on top
- Activation function : Sigmoid



Deep Learning - HOW?

Deep Neural Network

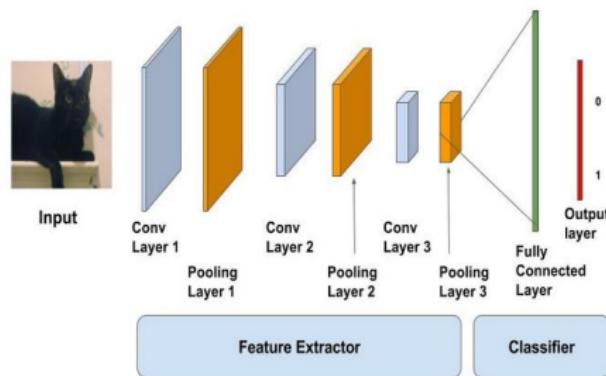
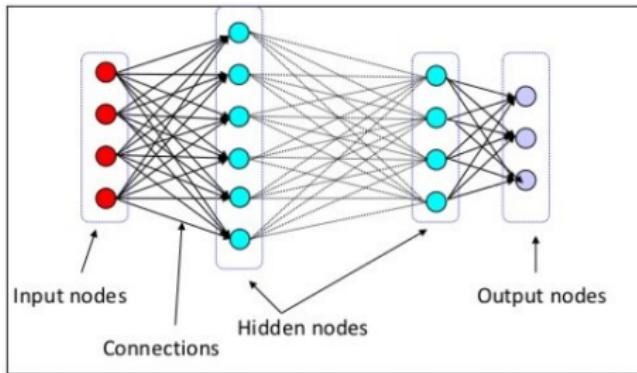
If there is only one hidden layer



Learning Weights of a deep neural networks is called as **deep learning**

Convolutional Neural Networks

Why



Neural Style Transfer

Style transfer relies on separating the content and style of an image. Given one content image and one style image, we aim to create a new, target image which should contain our desired content and style components:

- objects and their arrangement are similar to that of the content image (feature reconstruction)
- style, colors, and textures are similar to that of the style image (texture synthesis)

Neural Style Transfer

$$\text{content.image} + \text{style.image} = \text{new.imagewithstyle.transferred} \quad (1)$$



<https://github.com/puthusseri/styleTransfer.git>

Generative Adversarial Network

GANs are generative models: they create new data instances that resemble your training data.

eg: images that look like photographs of human faces, even though the faces don't belong to any real person.



GAN : Applications

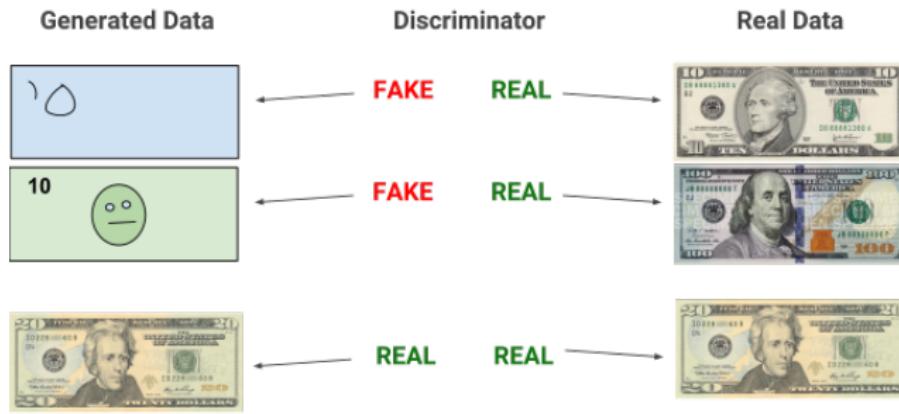
- Image to image translation (in unsupervised way)
- blue prints to real image
- photo to cartoon (Facebook AI research)
- photo of day to night (NVIDIA Research)
- Creating stimulated training set (eg : face recognition problem)
- for imitation learning

GAN : Overview

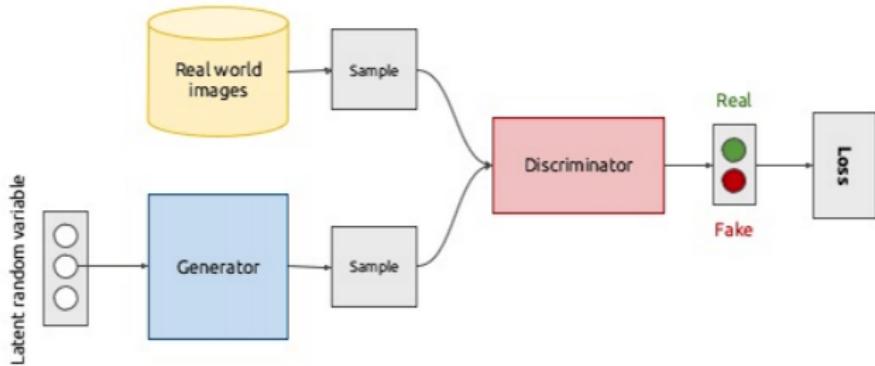
GANs has two parts:

- The generator : learns to generate plausible data. The generated instances become negative training examples for the discriminator.
- The discriminator: learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results.

GAN : Training



GAN : Architecture



Style Based GAN

- Introduced by NVIDIA
- Improved the efficiency of GAN by improving the generator
- Introduced new automated metrics - perceptual path length and linear seperability
- Result was : new dataset Flickr Face HQ (FFHQ) of size 2.56 TB

Style Based Generator

- The weights are studied through the 8 layer affine transformation.
- Feature maps are normalized using AdaIN
- Generate *stochastic details* by introducing the explicit noise for each layer.
- Final resulting feature maps are passed to the discriminator.

Style Based Generator : Architecture

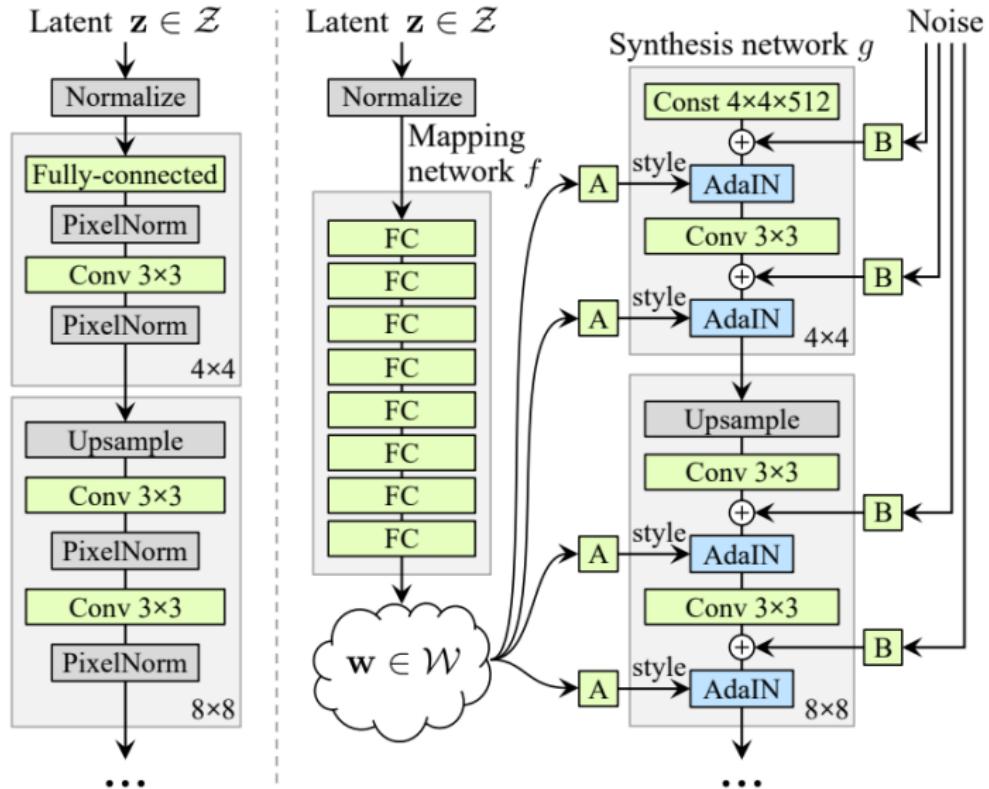
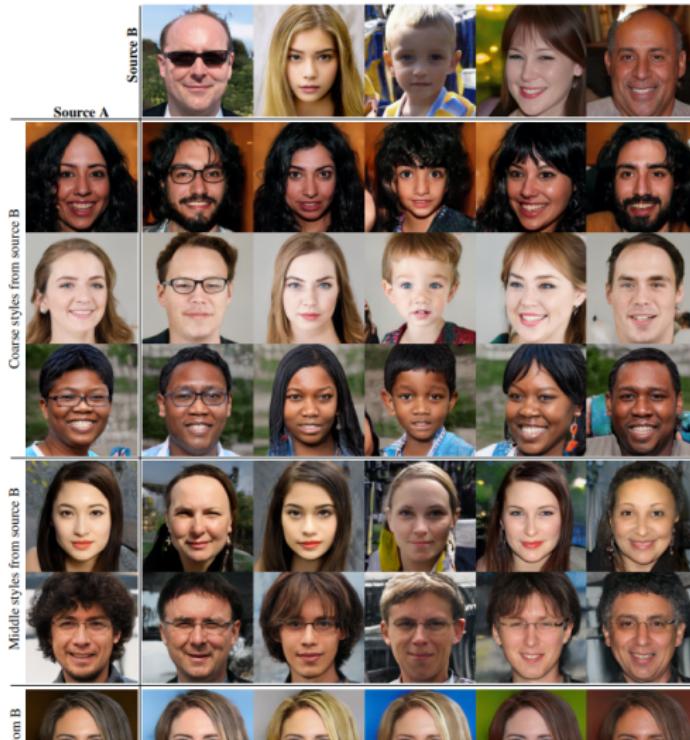


Image Quality

- Comparing with CelebA-HQ with FFHQ based on Frechet inception distances (FID) , a great improvement happens
- Used truncation trick
- Used 26.3M parameters for training
- Generated image is of 1024 * 1024 resolution

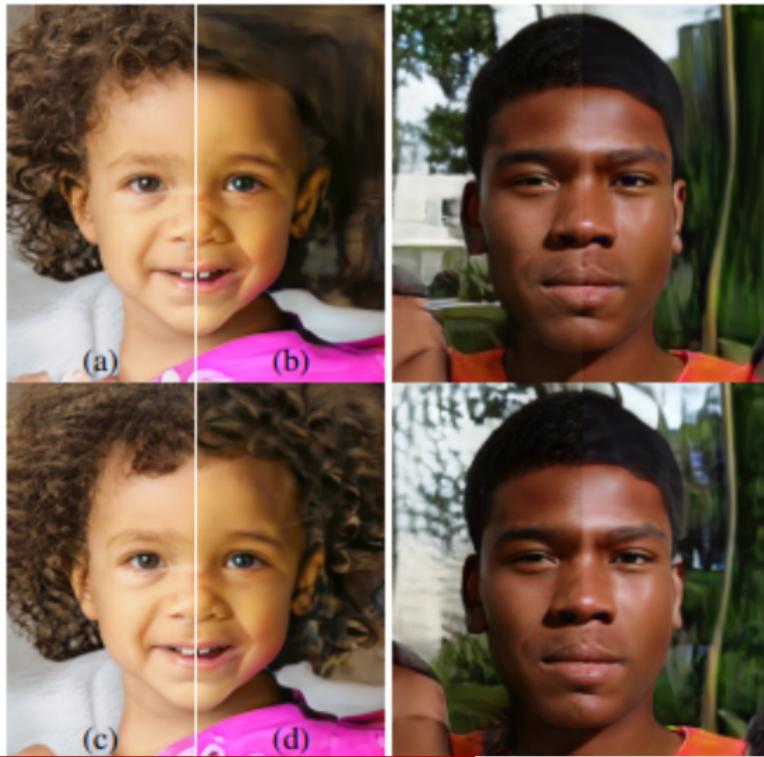
Style Based Generator : Properties

- Style mixing - mixing regularization



Style Based Generator : Properties

- Stochastic variation



Conclusion

- Reduced time complexity of the GAN
- FFHQ image database

Thank You!