Analysis of StyleGAN and StyleGAN2

Sangeeta Yadav, Arijit Dey
Indian Institute of Science

Overview:

- StyleGAN
- StyleGAN2
- Quality Assessment and Comparison
- Results

- ➔ **Goal of the project**
- ➔ To understand and analyze recently proposed Style based image generators, popularly known as StyleGANs

- ➔ **Main Contributions**
- ➔ Compared the network architecture of StyleGAN and StyleGAN2
- ➔ Introduced Unbiased error metrics to quantitatively compare these two styleGANs
- ➔ Validated the improvements of StyleGAN2 over StyleGAN
- ➔ Implemented the error metrics from scratch
- ➔ Introduced Deep Metrics such as LPIPS which tend to explain GANs better than conventional metrics
- ➔ Analyzed the performance of both StyleGANs

StyleGAN:

	Method	CelebA-HQ	FFHQ
A	Baseline Progressive GAN [30]	7.79	8.04
B	+ Tuning (incl. bilinear up/down)	6.11	5.25
C	+ Add mapping and styles	5.34	4.85
D	+ Remove traditional input	5.07	4.88
E	+ Add noise inputs	**5.06**	4.42
F	+ Mixing regularization	5.17	**4.40**

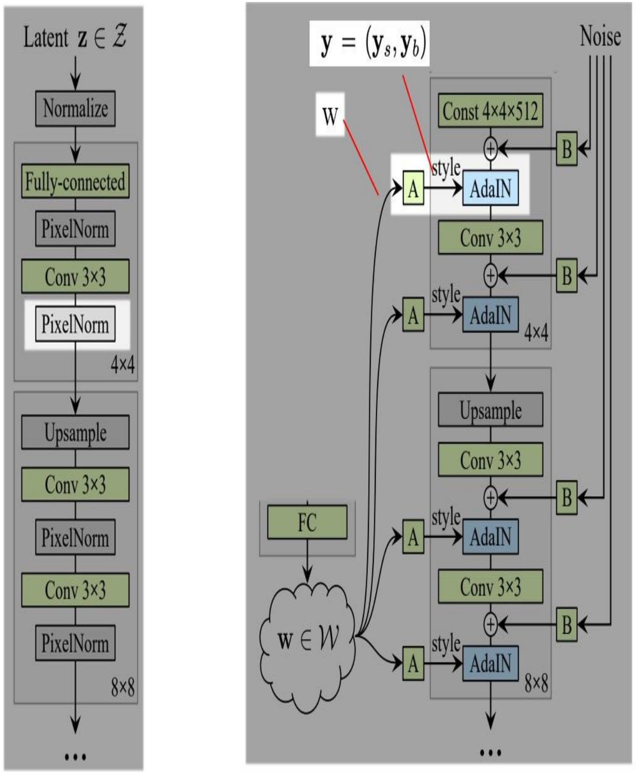
feature map i

the mean & standard deviation of the feature map i

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

normalize the feature map value

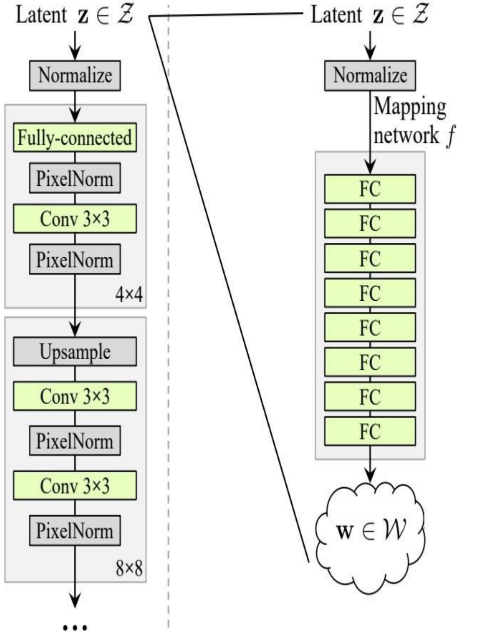
(instance normalization)



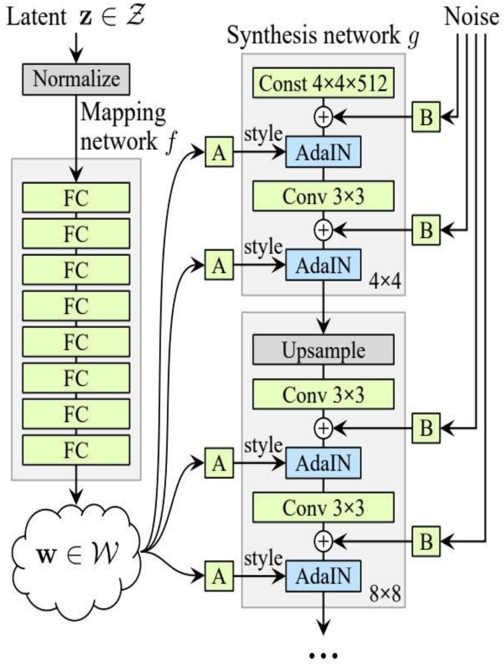
(a) Traditional

(b) Style-based generator

Traditional and StyleGAN: Network Architecture

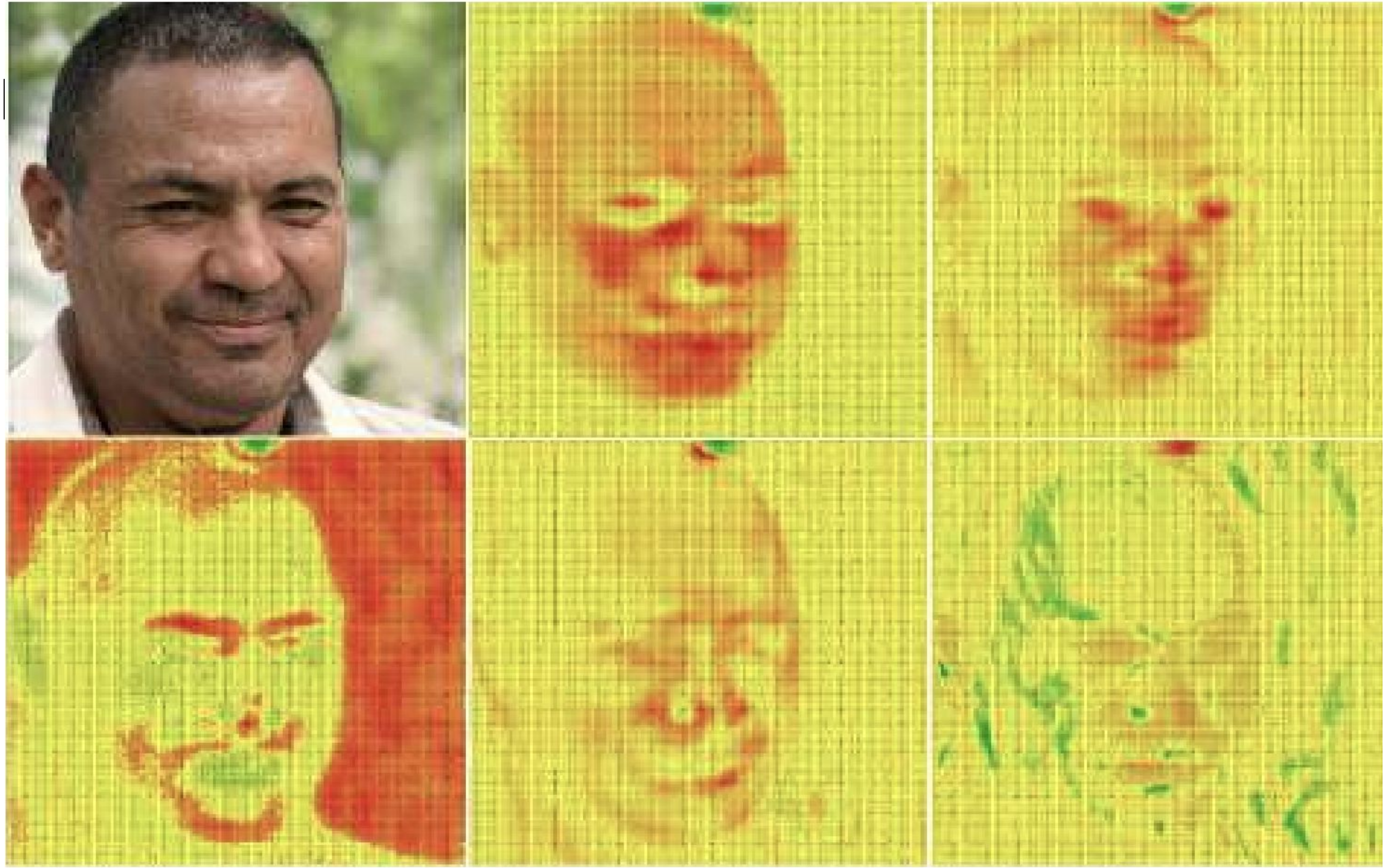


(a) Traditional (b) Style-based generator



(b) Style-based generator

(a) Generated image (b) Stochastic variation



StyleGAN & StyleGAN2: Network Architecture

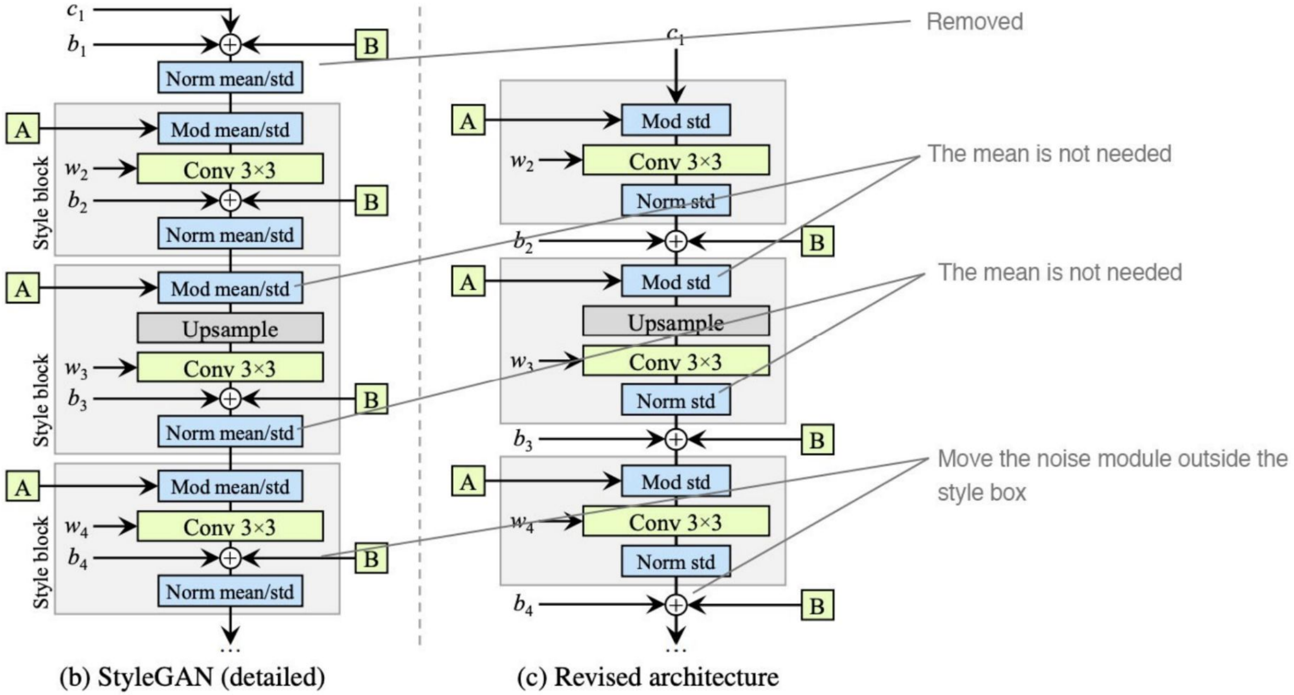
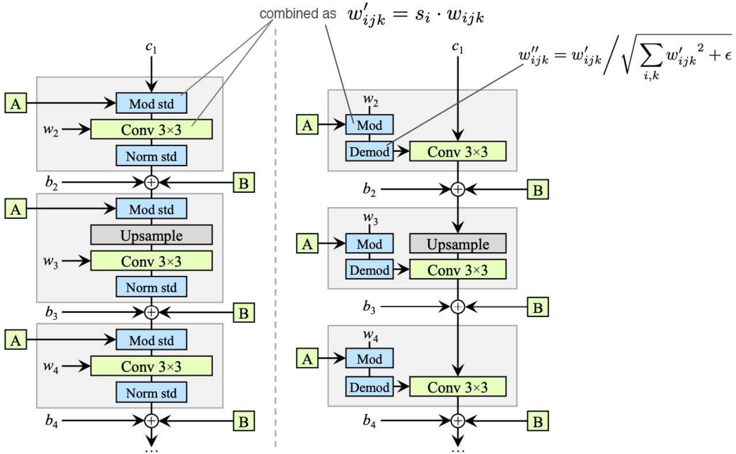


Figure 3: StyleGan2: Network architecture

StyleGAN2: Improve upon StyleGAN

- *Lacking Control Over Synthesized Images*

    - Weight Demodulation
    - Path Length Regularization
    - Deep Fake Detection via Projection
    - Removal of Latent Point Input
    - Perceptual Path Length



(c) Revised architecture
(d) Weight demodulation

$\mathbb{E}$

Error Metrics

We compute the following error metrics for both StyleGAN and StyleGAN2 images over a dataset of 2000 images

- Frechet inception distance (FID) and $\overline{FID}_\infty$

$$FID = ||M_t - M_g||_2^2 + Tr(C_t + C_g - 2(C_t C_g)^{\frac{1}{2}})$$

- Perceptual Path Length (PPL)
- Inception Score (IS)

$$\mathrm{IS}(G) = \exp\left(\mathbb{E}_{\mathbf{x}\sim p_g} D_{KL}(\, p(y|\mathbf{x}) \parallel p(y) \,) \right),$$

- Learned Perceptual Image Patch Similarity (LPIPS)

Calculation of FID_∞ and IS_∞

Algorithm 1 Evaluating $\overline{\text{FID}}_\infty$ for a Generator

G: Generator
I: Inception Network
n: Number of images we want to generate
t: Number of FIDs we compute for extrapolation
m_t: Precomputed groundtruth activations mean
c_t: Precomputed groundtruth activations covariance

procedure EVALUATE(G, I, n, t, m_t, c_t)
    FID = {}
    $z \leftarrow$ SOBOLSAMPLER(n)
    activations $\leftarrow I(G(z))$

    ▷ 5000 is the min no. of images we evaluate with
    batchSizes = LINSPACE(5000, n, t)
    for batchSize in batchSizes **do**
        SHUFFLE(activations)
        activations$_i$ $\leftarrow$ activations[:batchSize]
        FID.insert(CALCULATEFID(activations$_i$, m_t, c_t))
    end for
    reg $\leftarrow$ LINEARREGRESSION(1/batchSizes, FID)
    $\overline{\text{FID}}_\infty \leftarrow$ reg.predict(0)
    return $\overline{\text{FID}}_\infty$
end procedure

Comparison of StyleGAN and StyleGAN2

Figure : Images generated by StyleGAN

Figure : Images generated by StyleGAN2

	FFHQ 1024x1024		LSUN Car 1024x1024		LSUN Cat 128x128	
Error Metric	StyleGAN	StyleGAN2	StyleGAN	StyleGAN2	StyleGAN	StyleGAN2
FID	49.99	**44.36**	33.34	**30.68**	55.89	**50.67**
PPL	1234	**1086**	1865	**813**	1067	**998**
$\overline{FID}_\infty$	**49.992**	51.360	29.87	**28.65**	49.87	**45.43**
$\overline{IS}_\infty$	3.537	**3.697**	2.47	**2.95**	2.18	**2.64**
LPIPS	0.18	**0.16**	0.15	**0.13**	0.17	**0.16**

Conclusion

- *StyleGANs continue to be the State-of-the art architecture for generating human like faces. Despite the artifacts observed in StyleGAN, it is observed that StyleGAN is somewhat easier to accomodate in other pipelines e.g. Image2StyleGAN and Image2StyleGAN++ (although modified versions use StyleGAN3)*