A Style-Based Generator Architecture for Generative Adversarial Networks

CVPR 2019

Tero Karras, Samuli Laine, Timo Aila

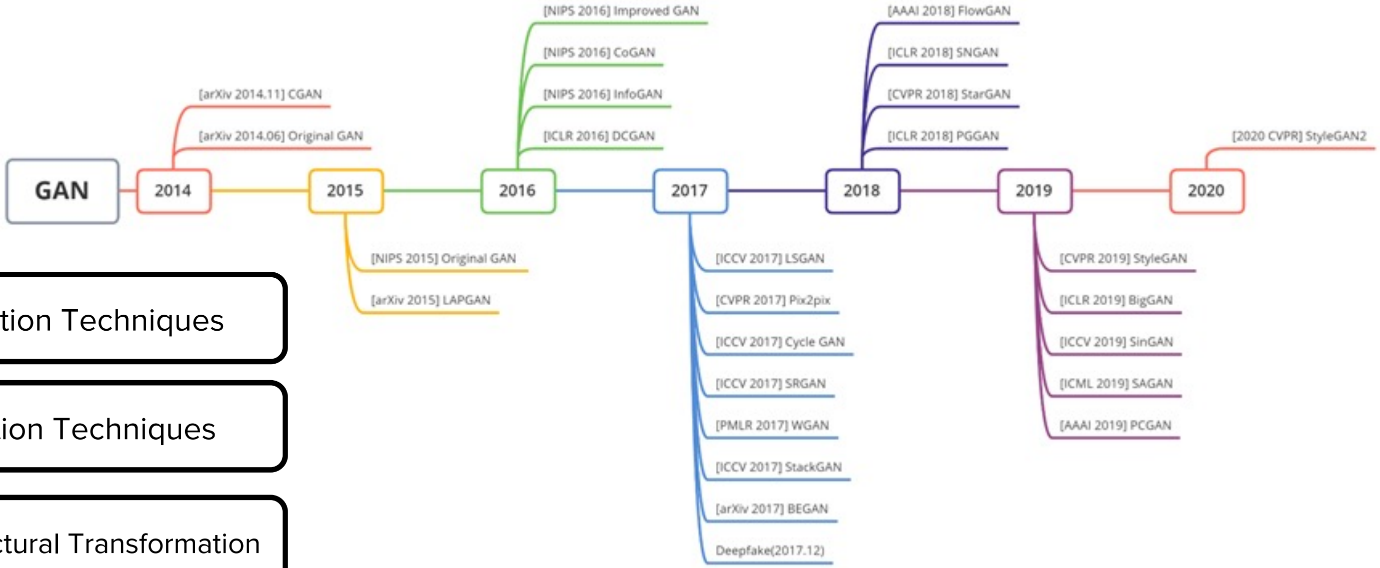
Presented by Beomsoon Park

Contents

Introduction

Introduction

❑ History of GAN



+ Normalization Techniques

+ Optimization Techniques

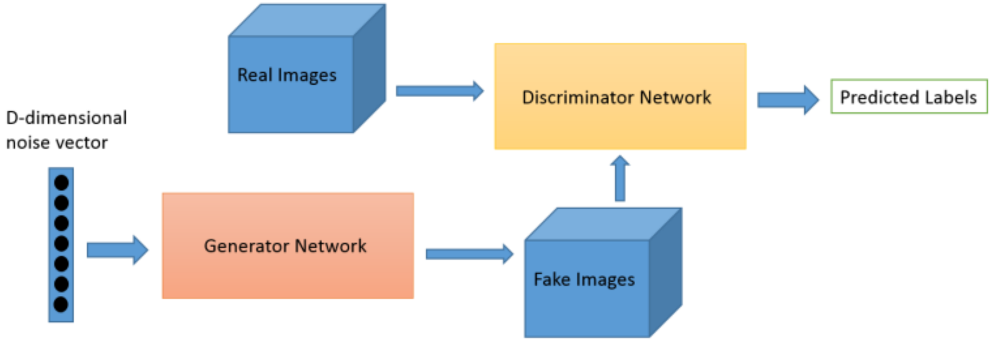
+ Auxiliary Structural Transformation

https://roytravel.tistory.com/109

Introduction

❑ History of GAN

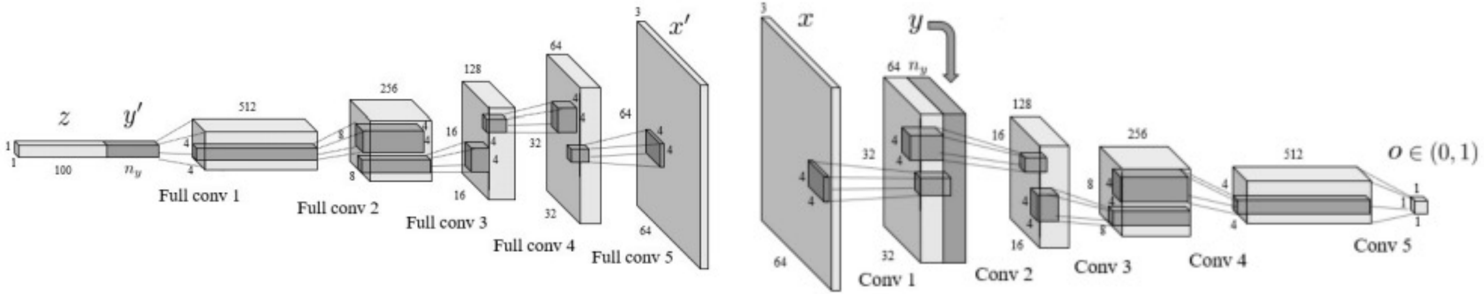
Original GAN



Introduction

❑ History of GAN

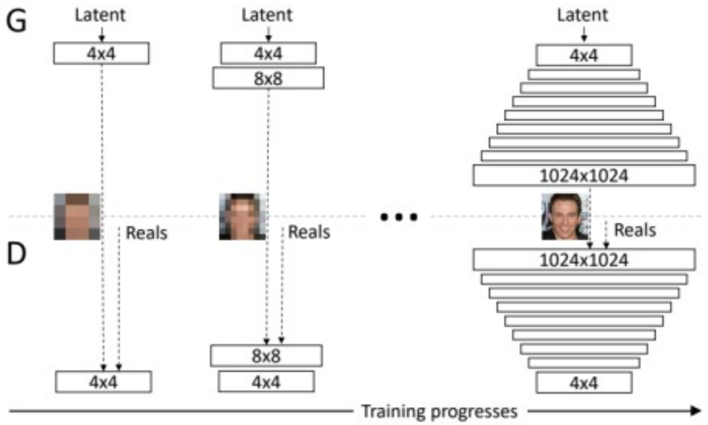
cGAN



Introduction

❑ History of GAN

ProGAN



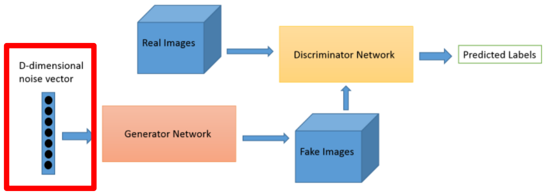
Introduction

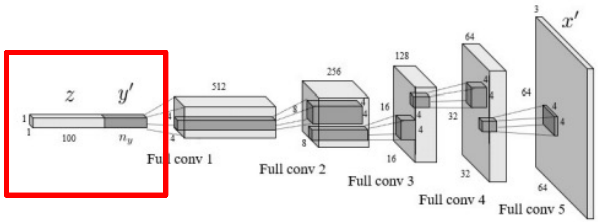
❑ Problem of previous models

Original GAN

cGAN

ProGAN





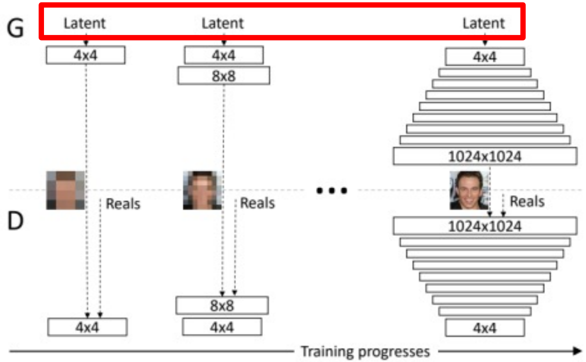
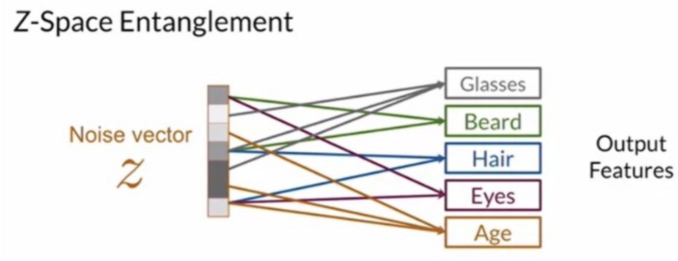


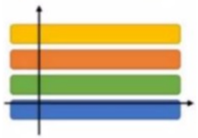
Image Generation based on **single latent space**

Introduction

❑ Entanglement in latent space



Actual Distribution



Desired
(Completely Distinguished)

Unexpected Impacts on other features

Less control over
the latent space & the generator

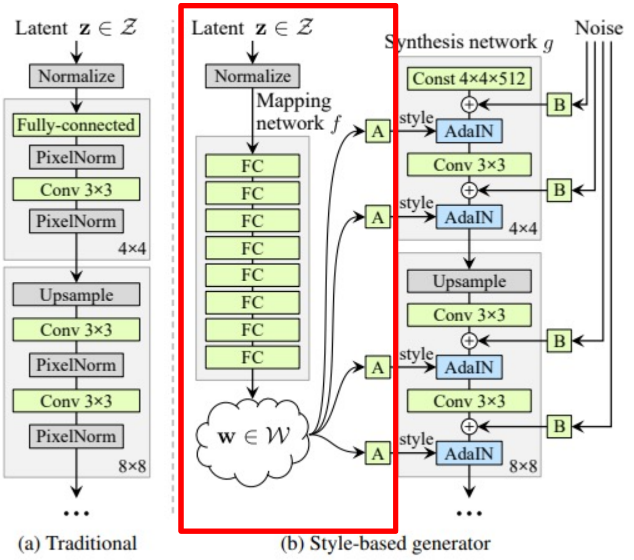
Motivation for style-based generator

https://velog.io/@tobigs-gm1/Disentanglement

Style-based Generator

Style-based Generator

❑ Model Structure



Mapping Network $f : \mathcal{Z} \rightarrow \mathcal{W}$ (8-layer MLP)

$w \in \mathcal{W}$: Intermediate Latent Vector

Learnable Affine Parameters

$$\mathbf{y} = (\mathbf{y}_s, \mathbf{y}_b)$$

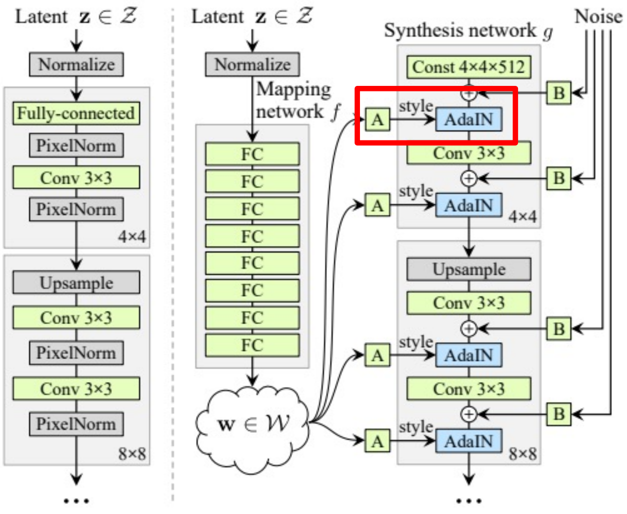
"Style"

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

"Adaptive Instance Normalization"

Style-based Generator

❑ Model Structure



(a) Traditional (b) Style-based generator

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

"Adaptive Instance Normalization"

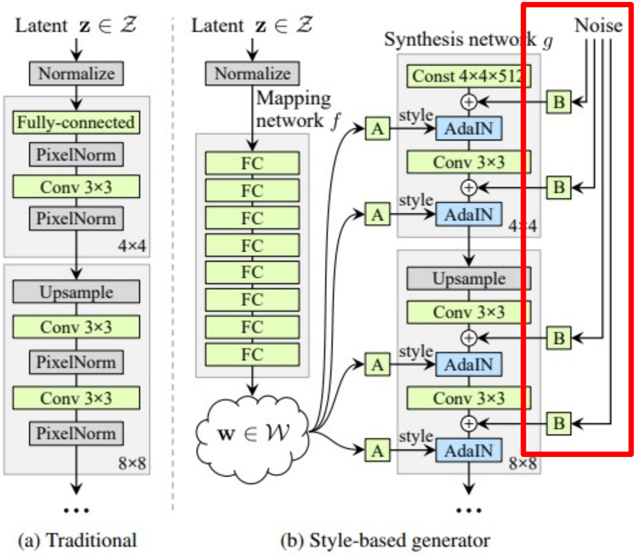
Normalization on each feature map

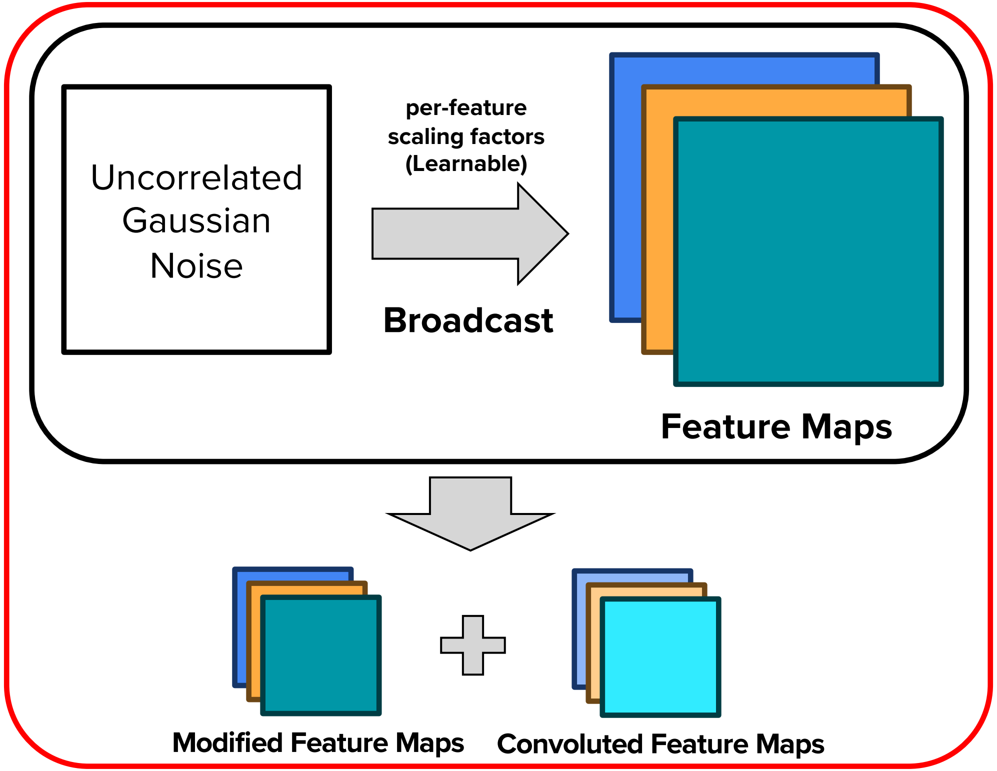
Scaling & Biasing with **"Style"**

Adding "Style" to the image that is being generated

Style-based Generator

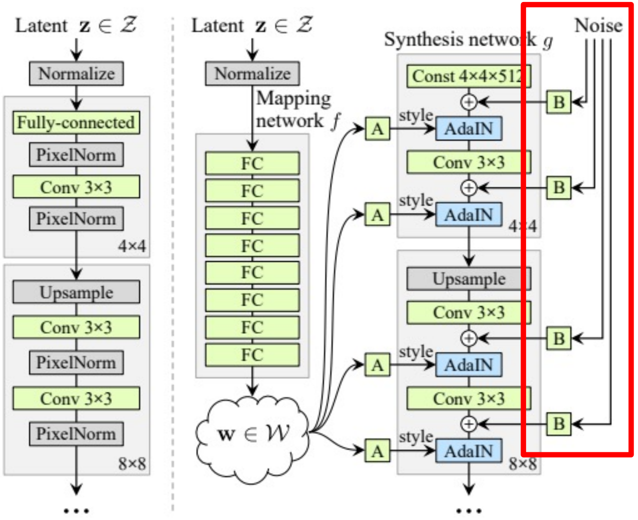
❑ Model Structure



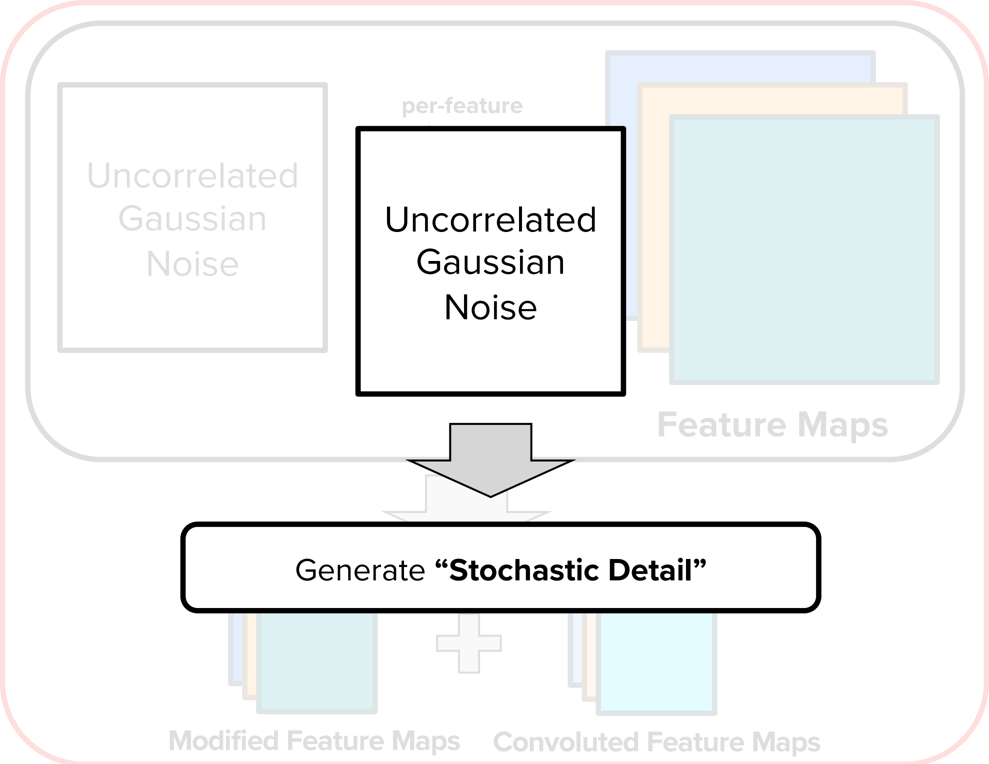


Style-based Generator

❑ Model Structure



(a) Traditional
(b) Style-based generator

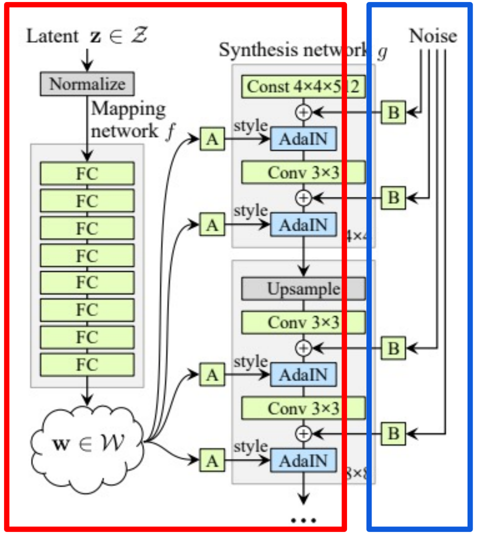


Properties of the Style-based Generator

Properties of the Style-based Generator

❑ Distinct Properties

Localized effects of styles



(b) Style-based generator

Stochastic Variation

Style Mixing

Properties of the Style-based Generator

❑ Localization of the effects of styles

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

Normalization of each feature vector

Scaling & Biasing based on the style

Achieve Localization

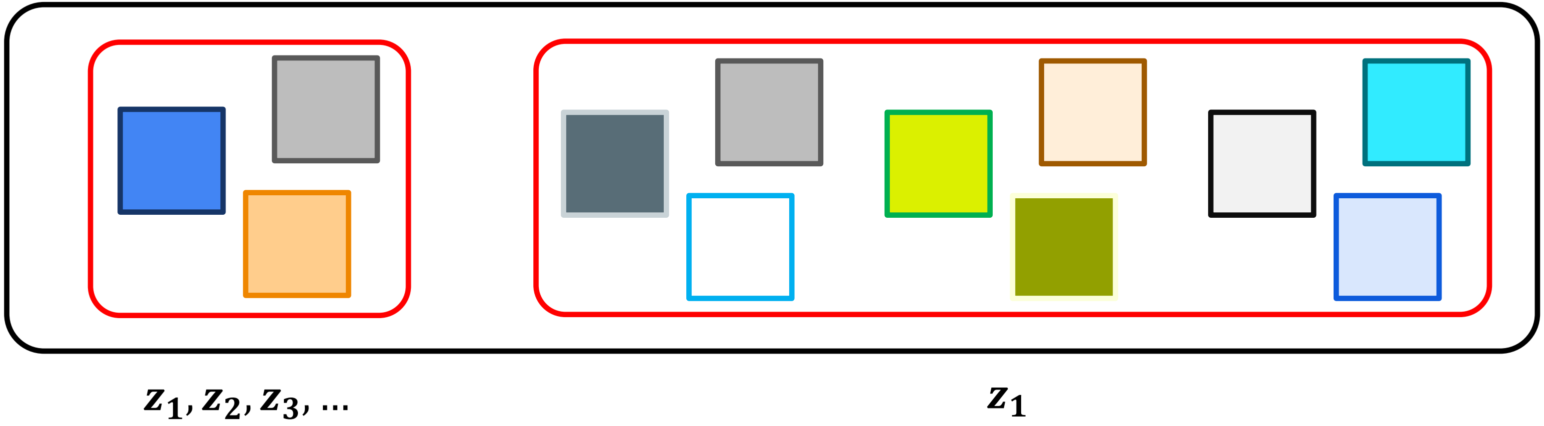
Force the convolution to be performed **only** based on the preceding features

Control the convolution in the desired direction as indicated in the style

Properties of the Style-based Generator

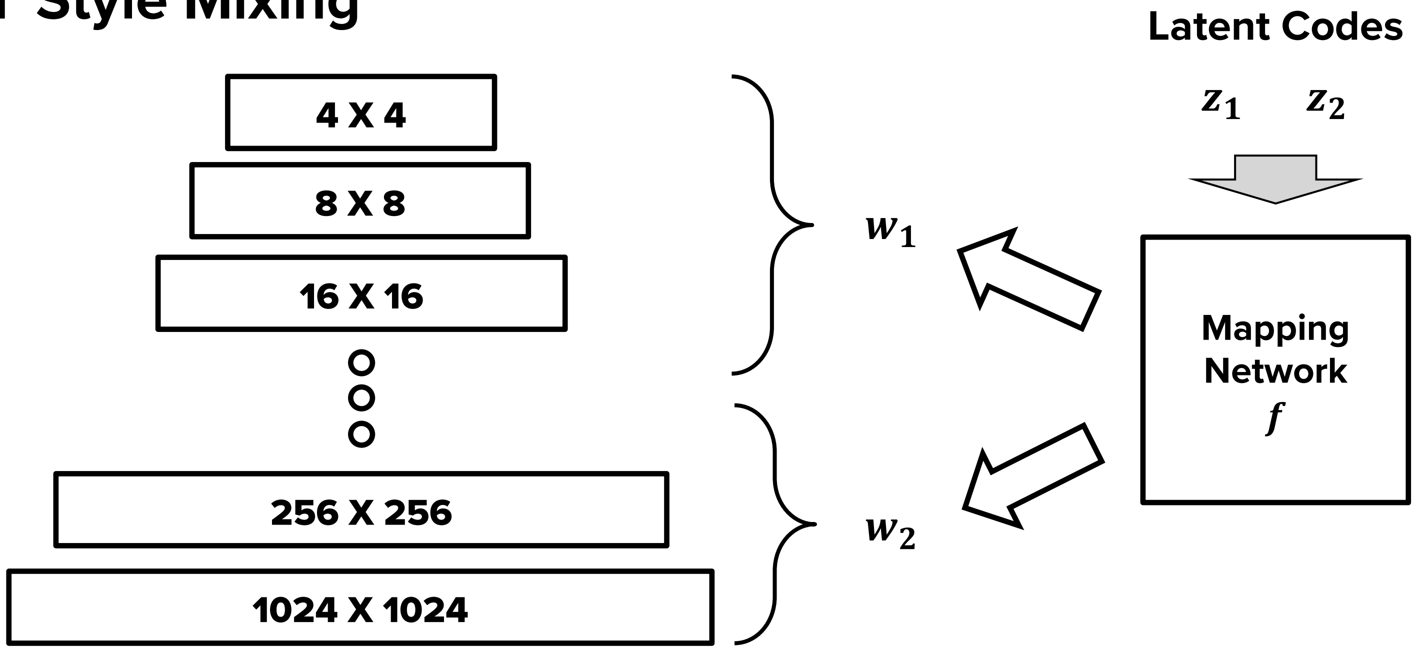
❑ Mixing Regularization

Generated Images



Properties of the Style-based Generator

- ❑ **Style Mixing**



Properties of the Style-based Generator

- ❑ **Style Mixing**

“Coarse Styles” from source B
“Fine Styles” from source A

“Middle Styles” from source B
“Middle Styles” from source A

“Fine Styles” from source B
“Coarse Stylse” from source A

Properties of the Style-based Generator

❑ Style Mixing

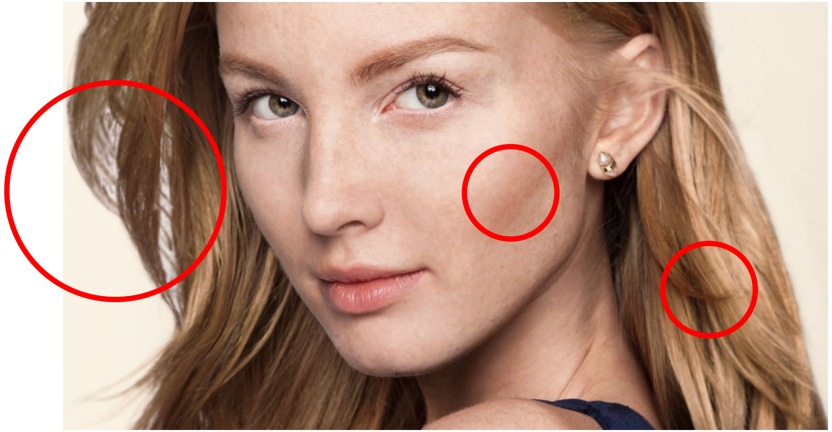
Mixing regularization	Number of latents during testing 1	2	3	4
E 0%	4.42	8.22	12.88	17.41
50%	4.41	6.10	8.71	11.61
F 90%	**4.40**	**5.11**	6.88	9.03
100%	4.83	5.17	**6.63**	**8.40**

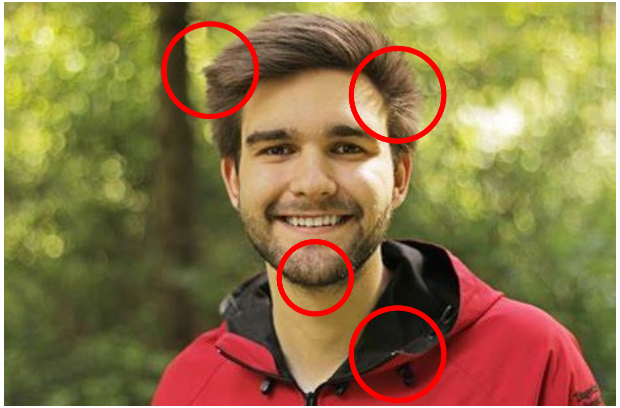
FID scores in FFHQ

*** FFHQ : New dataset of human faces created with style-based generator**

Properties of the Style-based Generator

❑ Stochastic Variation





Stochastic aspects in the picture of humans

Can be randomized
without affecting the perception

Properties of the Style-based Generator

❑ Stochastic Variation

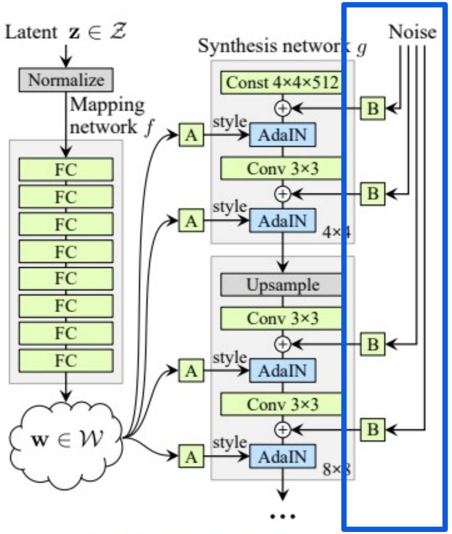
Difficult to achieve with traditional generators

- Latent information can be **supplied only through the input layer**
- **Hard to specify when and how many** spatially-varying pseudorandom numbers should be generated

Require more network capacity + Inevitable exposure of the periodicity of generated signal

Properties of the Style-based Generator

❑ Stochastic Variation



(b) Style-based generator

Add **per-pixel noise** after every convolution

With the help of per-feature scaling factors

Fresh noise is supplied to every layer (Features)

Only affects stochastic aspects (= Effect is localized)

Properties of the Style-based Generator

❑ Stochastic Variation

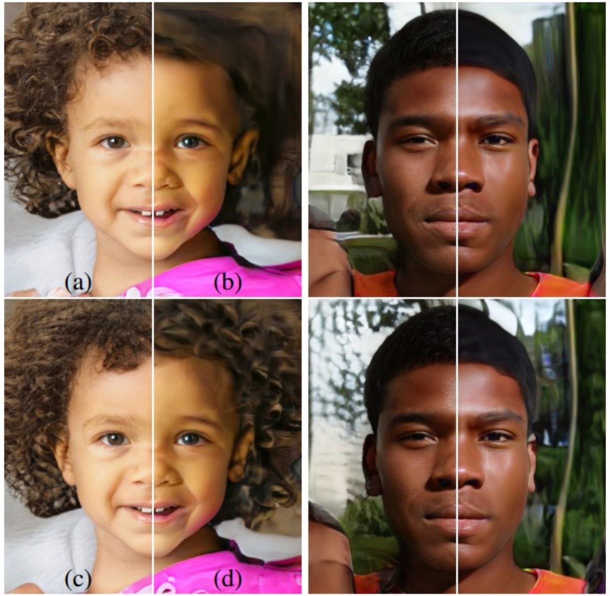
(a) Generated image (b) Stochastic variation (c) Standard deviation

White parts
➔ High Variations

Black parts
➔ Low Variations

Properties of the Style-based Generator

❑ Stochastic Variation



(a) : All layers
(b) : No noise
(c) : Fine Layers
(d) : Coarse Layers

Different Results according to the stage that noise is injected

Properties of the Style-based Generator

❑ Style & Stochasticity

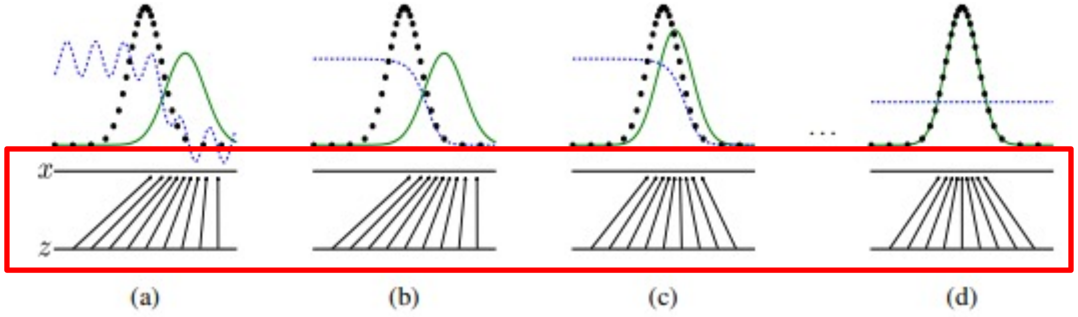
Style	Affect the entire image → Features are **controlled coherently** with $\mathbf{y} = (\mathbf{y}_s, \mathbf{y}_b)$
Stochasticity	- Noise added to all pixels **independently** → **Only handle** stochastic variation

⇨ Unsupervised learning with appropriate use of **global and local channels**

Disentanglement Studies

Disentanglement Studies

❏ **Entanglement in $\mathcal{Z}$**

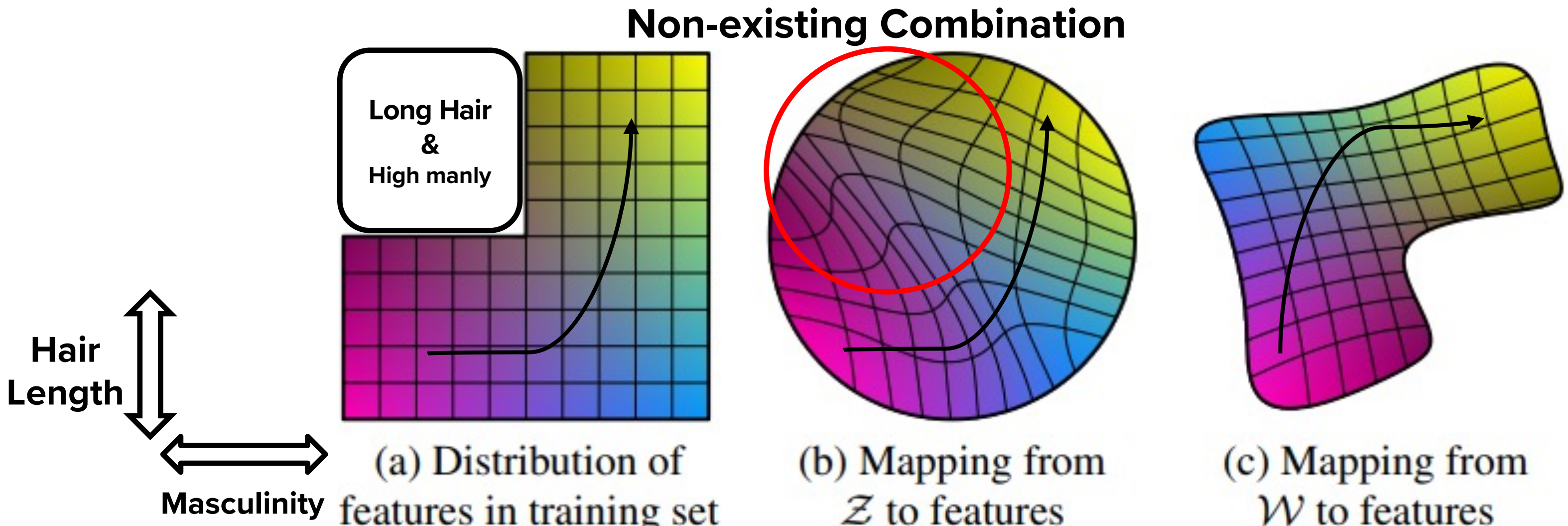


Sampling probability is forced to follow the density function of training data

I. Goodfellow et al. Generative Adversarial Networks. In NIPS, 2014.

Disentanglement Studies

- ❑ **Achieve Disentanglement with $\mathcal{W}$**



(a) Distribution of features in training set

(b) Mapping from $\mathcal{Z}$ to features

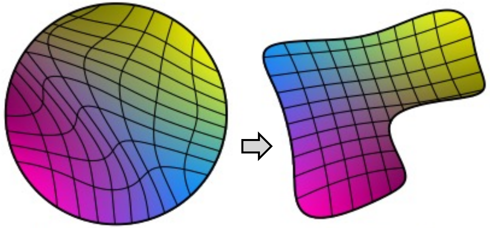
(c) Mapping from $\mathcal{W}$ to features

2 factors of variation (Masculinity & Hair Length)

Disentanglement Studies

- ❑ **Achieve Disentanglement with $\mathcal{W}$**

Non-linear **Linear**



(b) Mapping from $\mathcal{Z}$ to features (c) Mapping from $\mathcal{W}$ to features

$$z \rightarrow f(z) \rightarrow w$$

"Unwarp" $\mathcal{W}$ → Factors of variation become **more linear**

Mapping function → Help disentanglement through training

Disentanglement Studies

- **New metrics for quantifying disentanglment**

Previous suggested metrics : **Not suitable**

No encoder to yield latent codes

Perceptual Path Length

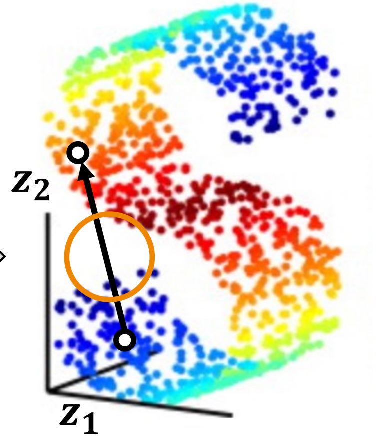
Linear Separability

Disentanglement Studies

❑ Perceptual Path Length

How to quantify disentanglement? → **Interpolation path in latent space**

Non-existing new feature in the latent space →



Entangled Latent Space

https://www.jkobject.com/blog/umap-explanation/

Disentanglement Studies

❑ Perceptual Path Length

How to quantify disentanglement? → **Interpolation path in latent space**

Proposed Metric

The **amount of change** while performing **interpolation in the latent space**

Disentanglement Studies

❑ Perceptual Path Length

Perceptually-based pairwise image distance → Weighted Difference between 2 VGG16 embeddings

Human-like Similarity Judgement on images

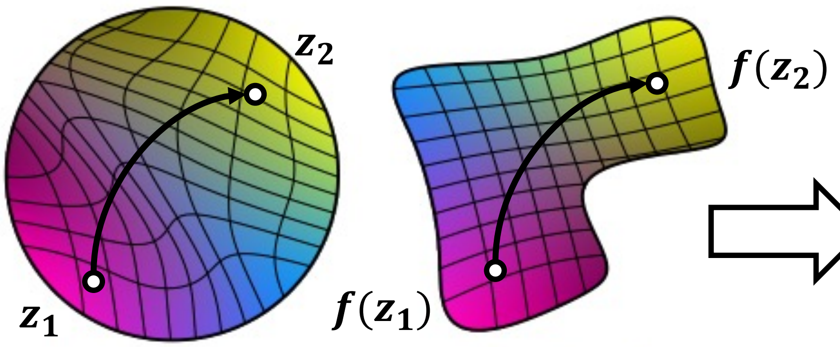
< Interpretation to the perceptual path length >

Low ⟷ **High**

Low	High
High Similarity → Smooth Change	Low Similarity → Drastic Change
“More disentangled”	**“Less disentangled”**

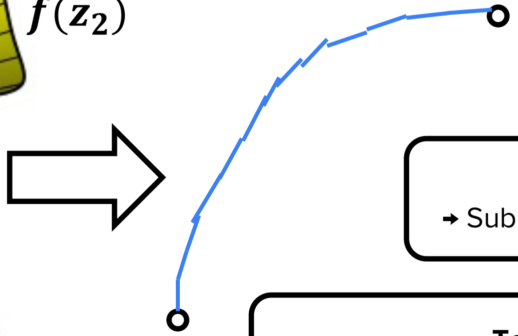
Disentanglement Studies

❑ Perceptual Path Length



(b) Mapping from $\mathcal{Z}$ to features

(c) Mapping from $\mathcal{W}$ to features



Interpolation path
➜ Subdivided into linear segments

Total perceptual length
= The sum of perceptual differences over each segment

Disentanglement Studies

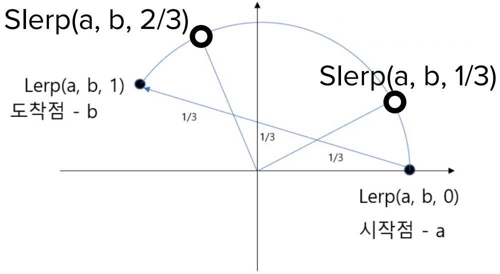
❑ Perceptual Path Length

$$l_{\mathcal{Z}} = \mathbb{E}\Big[\frac{1}{\epsilon^2} d\big(G(\mathrm{slerp}(\mathbf{z}_1, \mathbf{z}_2;\, t)), G(\mathrm{slerp}(\mathbf{z}_1, \mathbf{z}_2;\, t+\epsilon))\big)\Big]$$

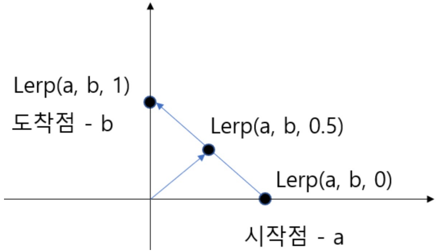
Perceptual Path Length in $\mathcal{Z}$

$$l_{\mathcal{W}} = \mathbb{E}\Big[\frac{1}{\epsilon^2} d\big(g(\mathrm{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2);\, t)), g(\mathrm{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2);\, t+\epsilon))\big)\Big]$$

Perceptual Path Length in $\mathcal{W}$



*** slerp : Spherical Interpolation**
*** lerp : Linear Interpolation**



Disentanglement Studies

❑ Perceptual Path Length

	Method		Path length full	Path length end	Separability
B	Traditional generator	$\mathcal{Z}$	412.0	415.3	10.78
D	Style-based generator	$\mathcal{W}$	446.2	376.6	3.61
E	+ Add noise inputs	$\mathcal{W}$	**200.5**	**160.6**	3.54
	+ Mixing 50%	$\mathcal{W}$	231.5	182.1	**3.51**
F	+ Mixing 90%	$\mathcal{W}$	234.0	195.9	3.79

Perceptual Path Length and Separability score depending on the architecture

	Method		FID	Path length full	Path length end	Separability
B	Traditional 0	$\mathcal{Z}$	5.25	412.0	415.3	10.78
	Traditional 8	$\mathcal{Z}$	4.87	896.2	902.0	170.29
	Traditional 8	$\mathcal{W}$	4.87	324.5	212.2	6.52
	Style-based 0	$\mathcal{Z}$	5.06	283.5	285.5	9.88
	Style-based 1	$\mathcal{W}$	4.60	219.9	209.4	6.81
	Style-based 2	$\mathcal{W}$	4.43	**217.8**	199.9	6.25
F	Style-based 8	$\mathcal{W}$	**4.40**	234.0	**195.9**	**3.79**

Results depending on the architecture and depth of mapping network

* full : $t \sim U(0,1)$
* end : $t \in \{0,1\}$

Disentanglement Studies

❑ Linear Separability

If sufficiently disentangled.. → Direction vectors consistently pointing to the corresponding factors of variation should exist

Proposed Metric

How well latent-space points can be separated into distinct 2 sets

= How clearly each set can represent binary attribute

Disentanglement Studies

❑ Linear Separability

Train auxiliary **classification** networks

Same as the discriminator in ProGAN

→ Obtain **binary attributes**

Generate 200000 images to evaluate one of the attributes

→ **Perform classification**

→ Leave **the most confident** samples (100000 samples)

Disentanglement Studies

❑ Linear Separability

$H(X|Y)$: Conditional Entropy

X : The class **predicted** by linear SVM based on $\boldsymbol{z}$

Y : The **true class** predicted by pre-trained classifier

The **amount of additional information** to determine the true class

$\exp(\Sigma_i H(X_i|Y_i))$: Final **Separability Score**

Disentanglement Studies

❑ Linear Separability

	Method		Path length full	Path length end	Separa-bility
B	Traditional generator	$\mathcal{Z}$	412.0	415.3	10.78
D	Style-based generator	$\mathcal{W}$	446.2	376.6	3.61
E	+ Add noise inputs	$\mathcal{W}$	**200.5**	**160.6**	3.54
	+ Mixing 50%	$\mathcal{W}$	231.5	182.1	**3.51**
F	+ Mixing 90%	$\mathcal{W}$	234.0	195.9	3.79

Perceptual Path Length and Separability score depending on the architecture

	Method		FID	Path length full	Path length end	Separa-bility
B	Traditional 0	$\mathcal{Z}$	5.25	412.0	415.3	10.78
	Traditional 8	$\mathcal{Z}$	4.87	896.2	902.0	170.29
	Traditional 8	$\mathcal{W}$	4.87	324.5	212.2	6.52
	Style-based 0	$\mathcal{Z}$	5.06	283.5	285.5	9.88
	Style-based 1	$\mathcal{W}$	4.60	219.9	209.4	6.81
	Style-based 2	$\mathcal{W}$	4.43	**217.8**	199.9	6.25
F	Style-based 8	$\mathcal{W}$	**4.40**	234.0	**195.9**	**3.79**

Results depending on the architecture and depth of mapping network

Results

Results

- ❑ **Comparing with other generators**

	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**

FID scores for various generators

Results

❑ Perceptual Path Length & Separability

	Method		Path length full	Path length end	Separability
B	Traditional generator	$\mathcal{Z}$	412.0	415.3	10.78
D	Style-based generator	$\mathcal{W}$	446.2	376.6	3.61
E	+ Add noise inputs	$\mathcal{W}$	**200.5**	**160.6**	3.54
	+ Mixing 50%	$\mathcal{W}$	231.5	182.1	**3.51**
F	+ Mixing 90%	$\mathcal{W}$	234.0	195.9	3.79

	Method		FID	Path length full	Path length end	Separability
B	Traditional 0	$\mathcal{Z}$	5.25	412.0	415.3	10.78
	Traditional 8	$\mathcal{Z}$	4.87	896.2	902.0	170.29
	Traditional 8	$\mathcal{W}$	4.87	324.5	212.2	6.52
	Style-based 0	$\mathcal{Z}$	5.06	283.5	285.5	9.88
	Style-based 1	$\mathcal{W}$	4.60	219.9	209.4	6.81
	Style-based 2	$\mathcal{W}$	4.43	**217.8**	199.9	6.25
F	Style-based 8	$\mathcal{W}$	**4.40**	234.0	**195.9**	**3.79**

Even the traditional model has **performed better** when $\mathcal{W}$ is introduced

Best results are derived from the proposed model → **$\mathcal{W}$ is less entangled than $\mathcal{Z}$**

Results

❑ Unresolved problem

Figure 6: Drop-like artefacts in generated images

Figure 7: Other visual artefacts in the CelebA dataset. Notice the lack of symmetry and visible marks in ears. There's also some artefacts in the eyes, glasses, and lips

Thank you