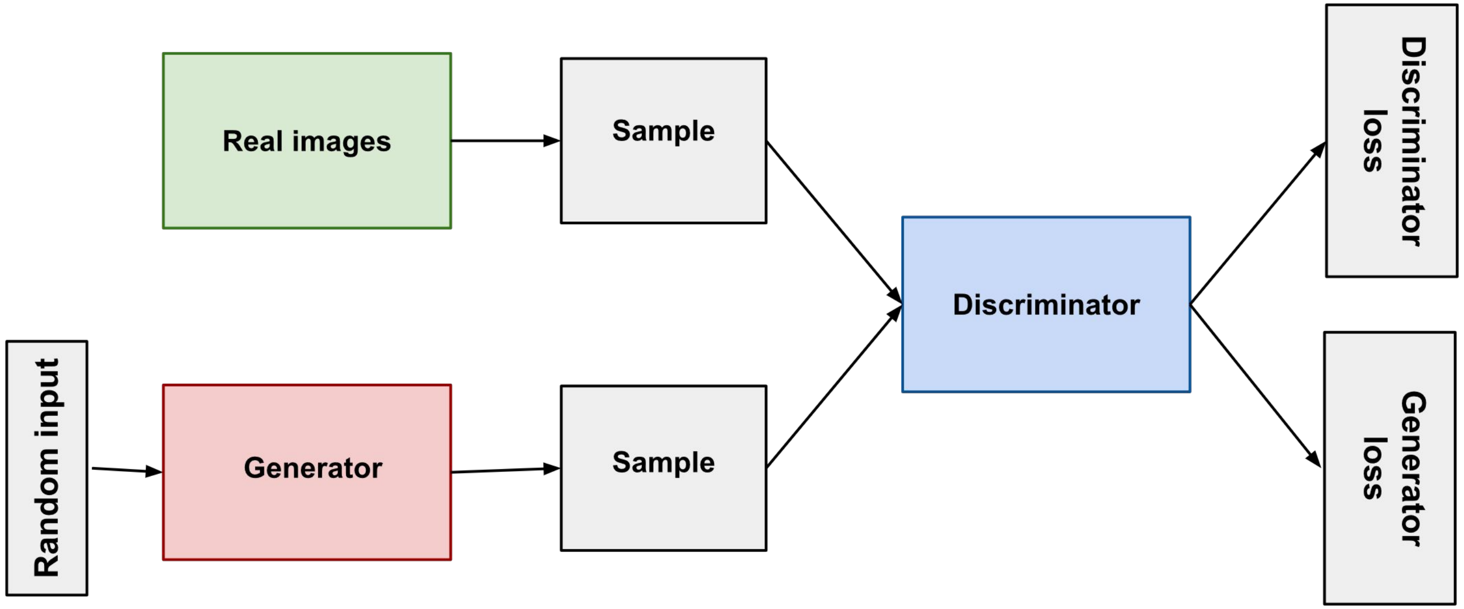
A Style-Based Generator Architecture for Generative Adversarial Networks

Tero Karras Samuli Laine Timo Aila

Agenda

1. Background
2. Motivation
3. Architecture & Properties
4. Disentanglement studies
5. Results
6. Conclusion
7. Q & A

Background – Generative Adversarial Networks (GAN)



Source: https://developers.google.com/machine-learning/gan/gan_structure

- Generator: Learns to map from the latent space, to the real image space
- Discriminator: Estimates the probability that a sample comes from the training data rather than the generator

Background – Image Style Transfer

Style image

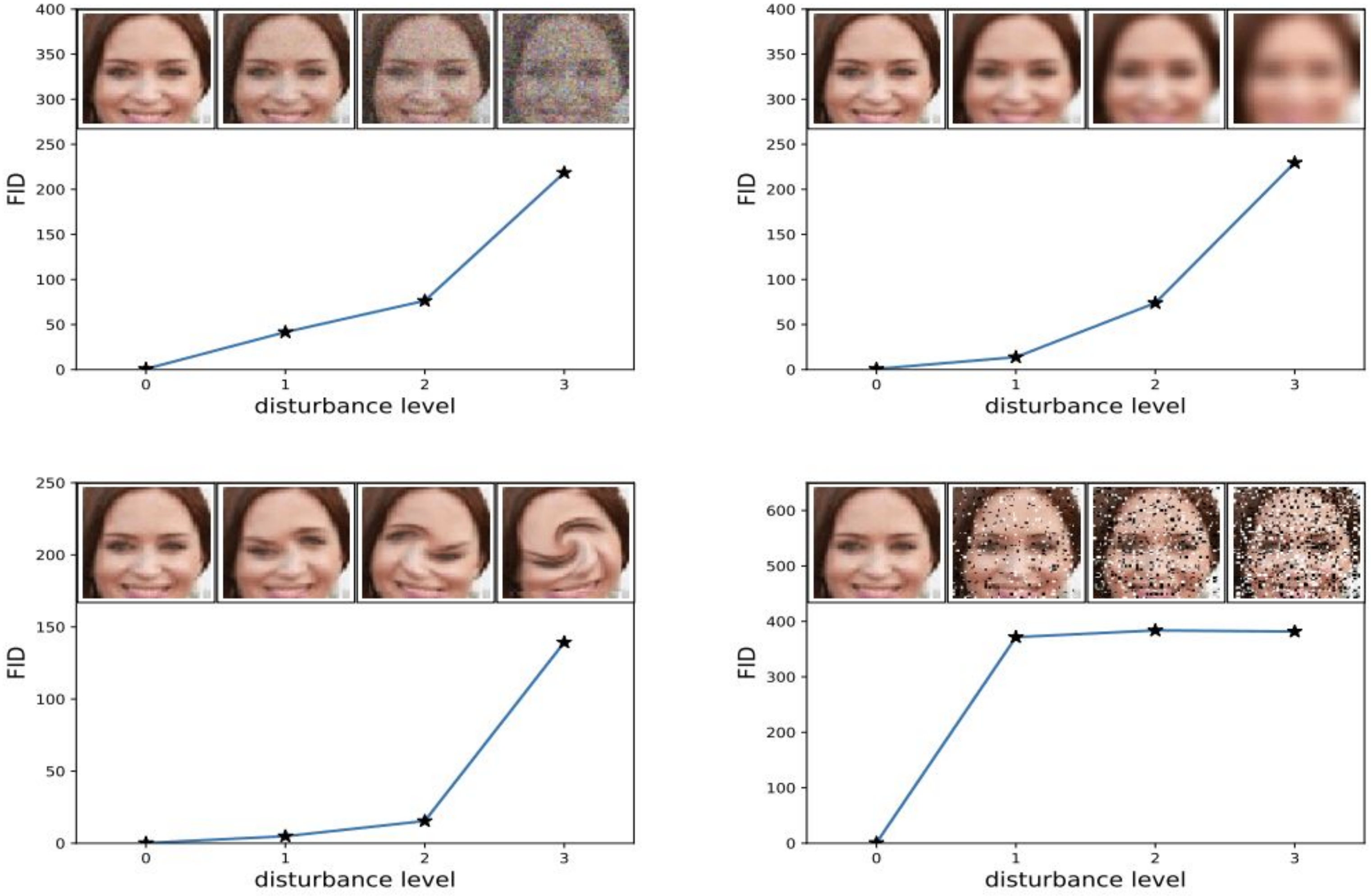
Content image

Synthesized image

Source: Image Style Transfer Using Convolutional Neural Networks

- Image = semantic object + style
- Transferring the style from one image onto another

Background – Fréchet Inception Distance (FID)



- A metric used to assess the quality of images created by a generative model
- Compare the distribution of generated images with the distribution of real images used to train the generator
- Features are generated from convolutional neural networks: Compare the mean and standard deviation of one of the deeper layers (assume Gaussian distribution)

Source: GANs trained by a two time-scale update rule converge to a local Nash equilibrium

Motivation

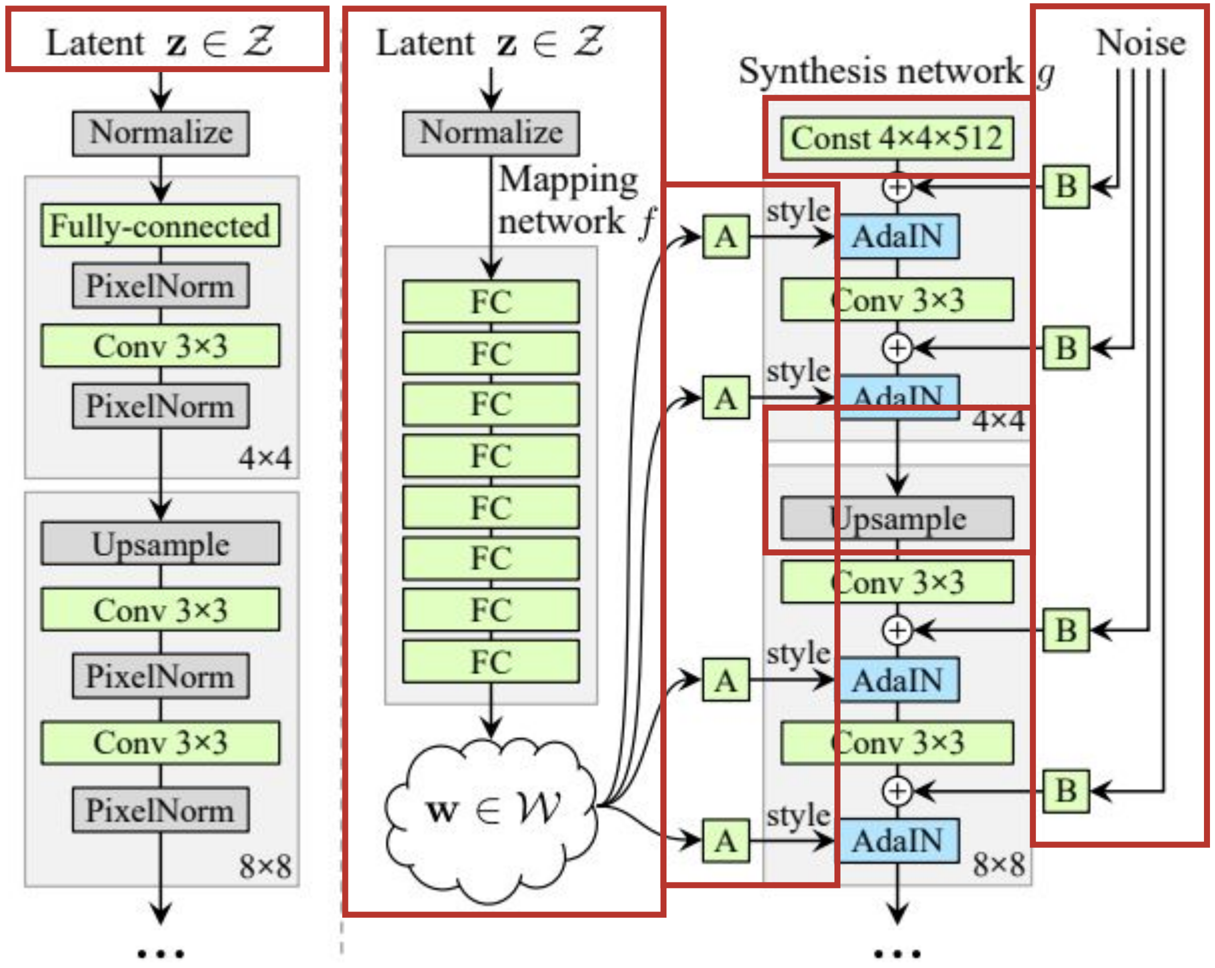
- The generators continue to operate as black boxes.
- The properties of the latent space are poorly understood.
- The commonly demonstrated latent space interpolations provide no quantitative way to compare different generators against each other.

Solution:

A style-based generator architecture that leads to

- **an automatically learned, unsupervised separation of high-level attributes and stochastic variation in the generated images**
- **better interpolation properties**
- **better disentanglement of the latent factors of variation**

Architecture & Properties



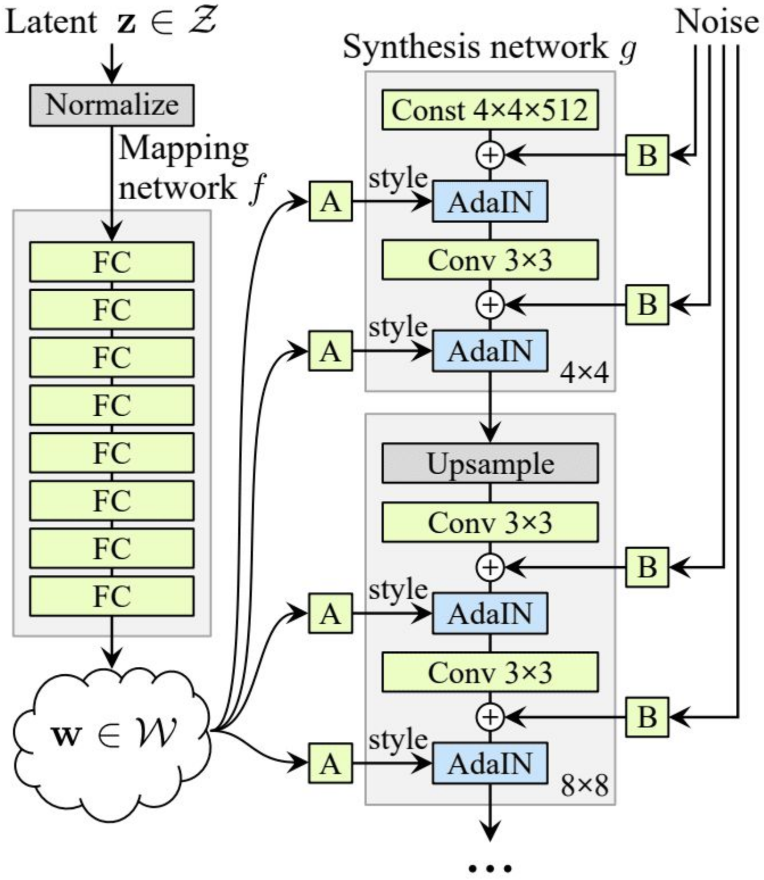
(a) Traditional (b) Style-based generator

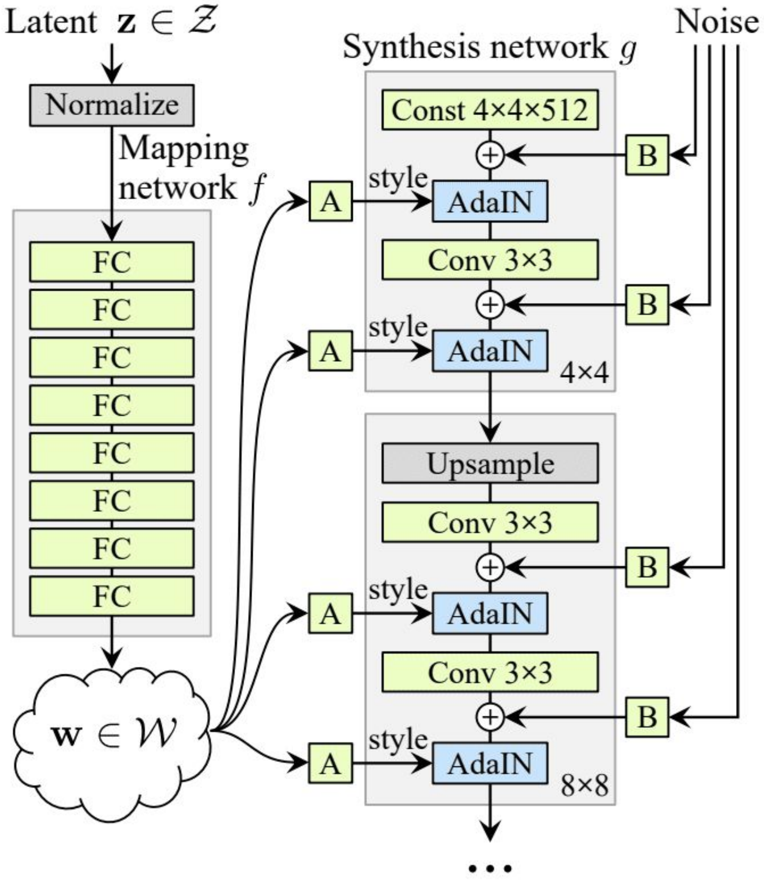
- Progressive growing GAN training method (Bi-linear sampling)
- Traditional generator
  - Provide the latent code to the generator through the first layer of a feedforward network
- Style-based generator
  - Map the latent input to an intermediate latent space W uisng an 8-layer MLP
  - A: Apply learned affine transformations that specialize w to styles $y = (y_s; y_b)$ which control adaptive instance normalization (AdaIN) operations after each convolution layer

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

  - B: Apply learned per-channel scaling factors to the noise input
  - Improvement: Remove the traditional input layer and start the image synthesis from a learned 4 × 4 × 512 constant tensor

Architecture & Properties

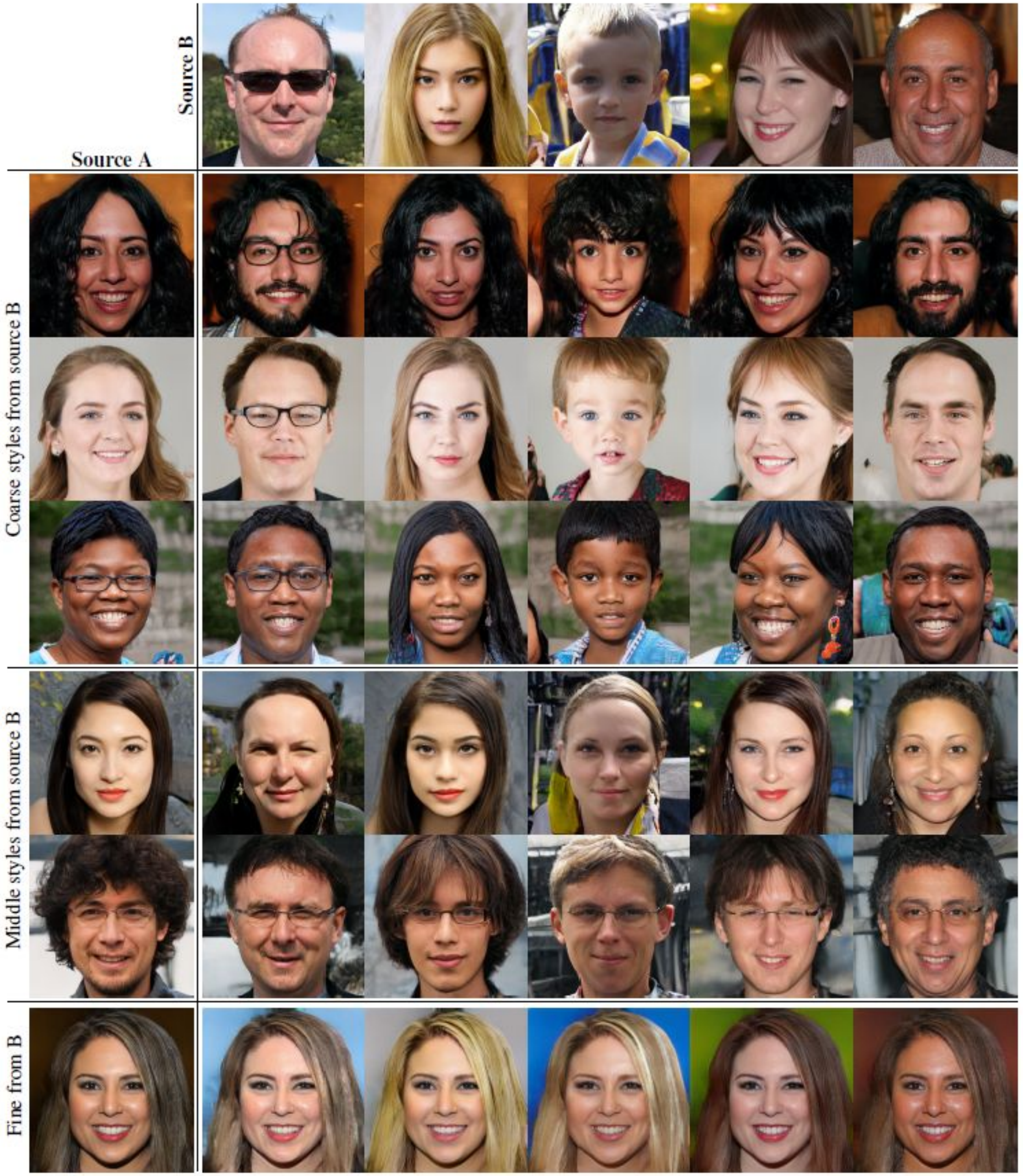




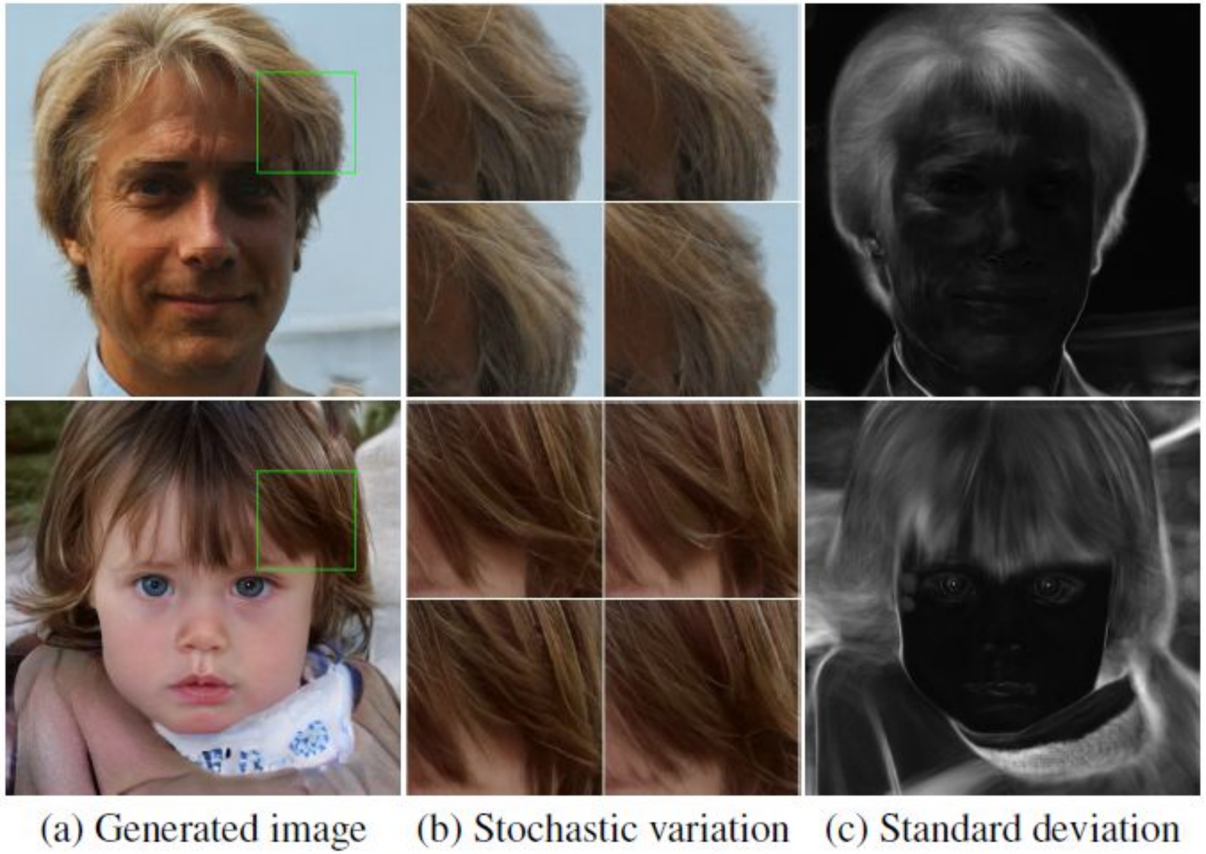
- Mixing regularization
  - When generating an image, switch from one latent code to another at a randomly selected point in the synthesis network.

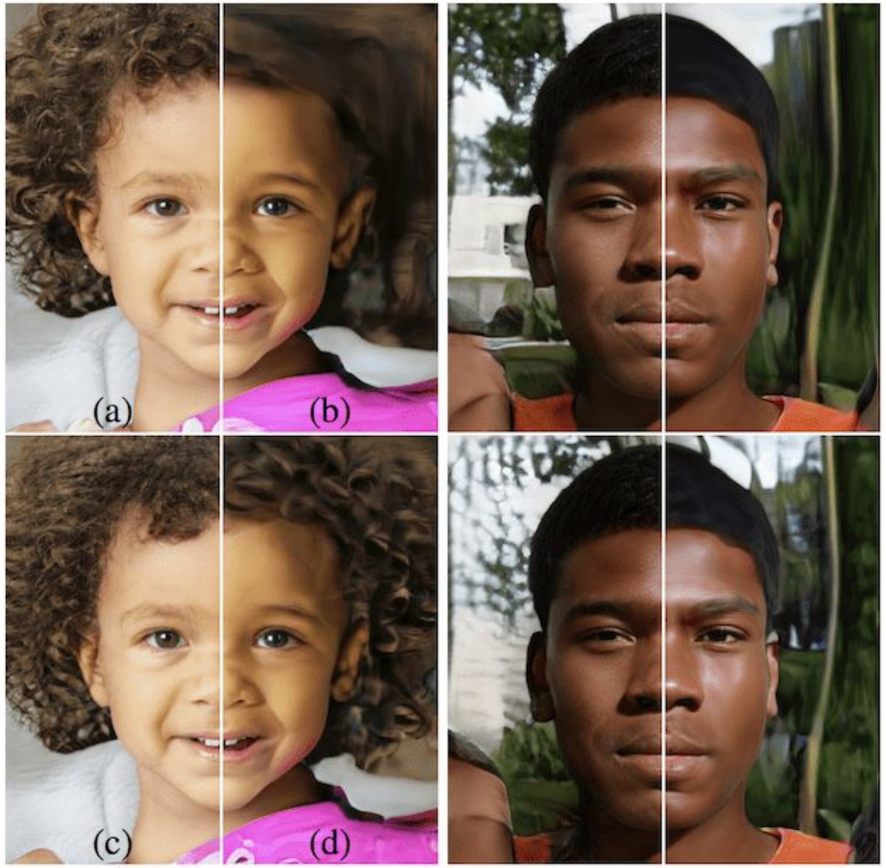
Architecture & Properties

- Mixing regularization
  - Coarse spatial resolutions (4^2 – 8^2) bring high-level aspects such as pose, general hair style, face shape, and eyeglasses from B.
  - Middle resolutions (16^2 – 32^2) bring smaller scale facial features, hair style, eyes open/closed from B.
  - High resolutions (64^2 – 1024^2) bring mainly the color scheme and microstructure from B.



Architecture & Properties





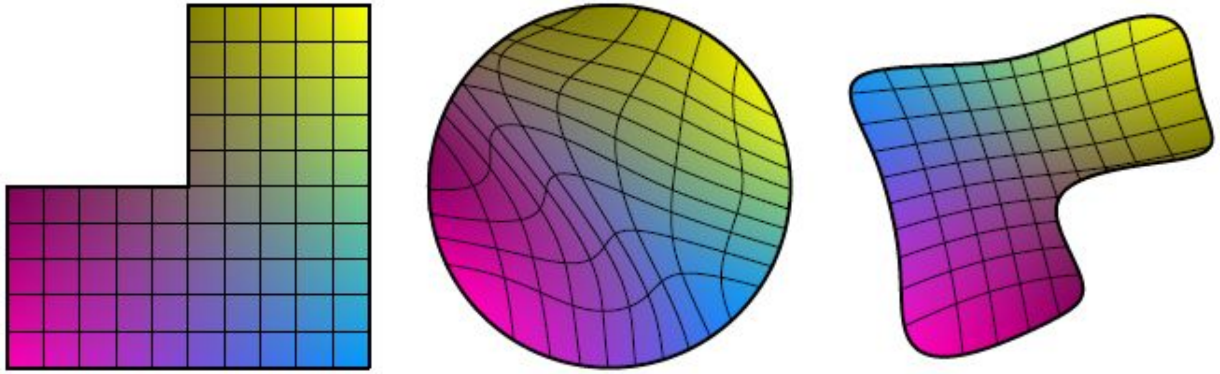
- Stochastic variation
  - Traditional generator: Invent a way to generate spatially-varying pseudorandom numbers from earlier activations whenever they are needed
  - Style-based generator: Add per-pixel noise after each convolution

Results

	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**

- Datasets
  - CELEBA-HQ: 30,000 high-quality celebrity images at 1024^2 resolution, each with 40 binary attributes annotations
  - Flickr-Faces-HQ (FFHQ): 70,000 high-quality images at 1024^2 resolution, more variation than CELEBA-HQ in terms of age, ethnicity and image background, and also much better coverage of accessories such as eyeglasses, sunglasses, hats, etc.
- Calculate the FIDs using 50,000 images drawn randomly from the training set, and report the lowest distance encountered over the course of training

Disentanglement studies

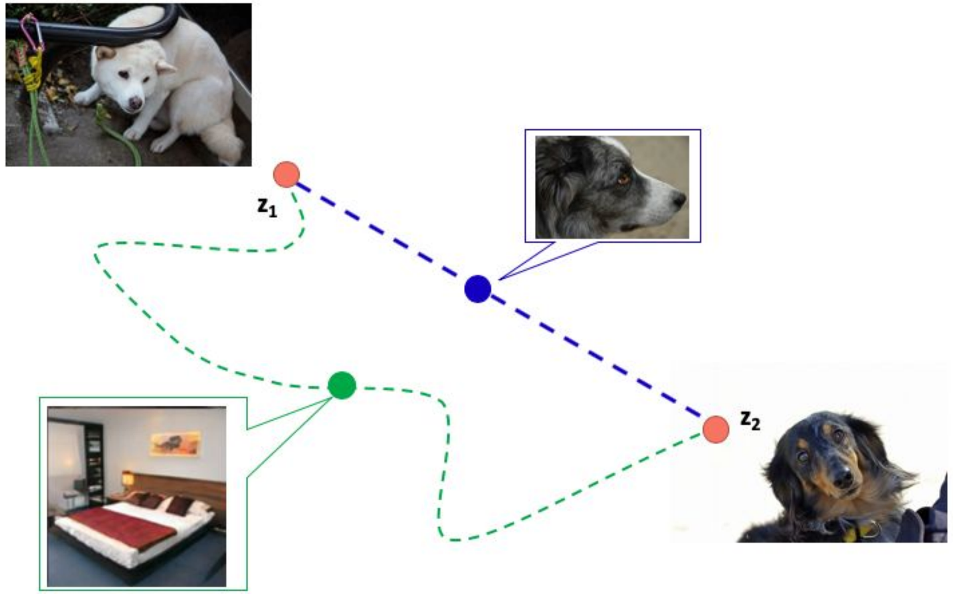


(a) Distribution of features in training set (b) Mapping from $\mathcal{Z}$ to features (c) Mapping from $\mathcal{W}$ to features

- A latent space that consists of linear subspaces, each of which controls one factor of variation
- There is pressure for the generator to unwarp W so that the factors of variation become more linear: It should be easier to generate realistic images based on a disentangled representation than based on an entangled representation

Disentanglement studies

- Perceptual path length
  - Measure how drastic changes the image undergoes as we perform interpolation in the latent space
  - A less curved latent space should result in perceptually smoother transition than a highly curved latent space.



Source: https://medium.com/analytics-vidhya/from-gan-basic-to-stylegan2-680add7abe82

- Linear separability
  - Measure how well the latent-space points can be separated into two distinct sets via a linear hyperplane, so that each set corresponds to a specific binary attribute of the image
  - If a latent space is sufficiently disentangled, it should be possible to find direction vectors that consistently correspond to individual factors of variation.

Results

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**

- The intermediate latent space is perceptually more linear than the latent space.
- Style mixing appears to distort the intermediate latent space somewhat.
- Both traditional and style-based generators benefit from having a mapping network in terms of FID, separability, and path length.
- A deeper mapping network generally performs better than a shallow one.

Conclusion

- The traditional GAN generator architecture is in every way inferior to a style-based design.
- The investigations to the separation of high-level attributes and stochastic effects, as well as the linearity of the intermediate latent space will help improve the understanding and controllability of GAN synthesis.
- The average path length metric could be used as a regularizer during training, and perhaps some variant of the linear separability metric could act as one, too.

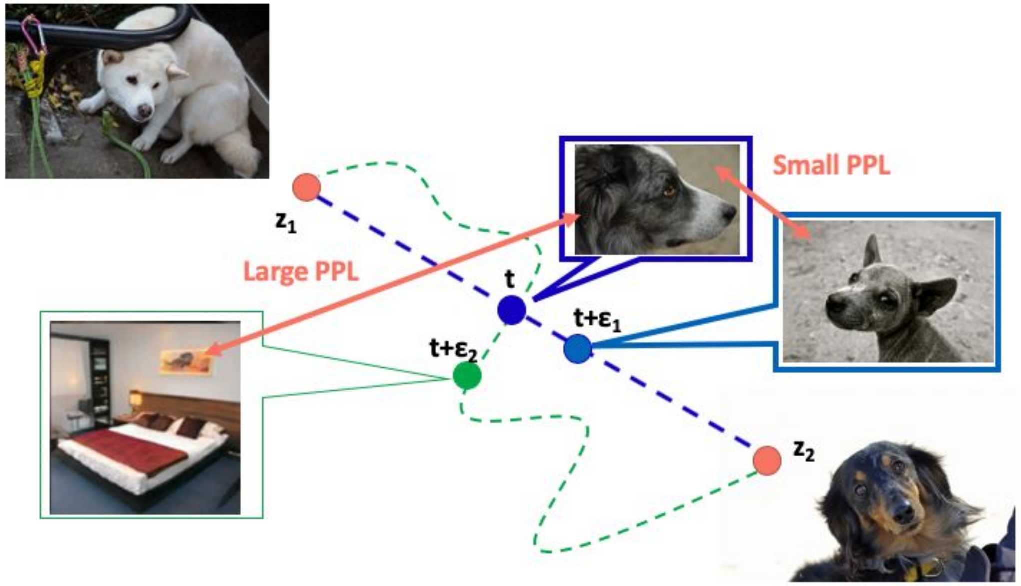
Q & A

Appendix

- Perceptual path length

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

  - g: Generator
  - f: Mapping network
  - d: Perceptual distance
  - lerp: Linear interpolation



- Linear separability
  - Train auxiliary classification networks for a number of binary attributes to label the generated images
  - For each attribute, fit a linear SVM to predict the label based on the latent-space point and classify the points by this plane
  - Compute the conditional entropy H(Y|X) where X are the classes predicted by the SVM and Y are the classes determined by the pre-trained classifier
  - Final separability score:

$$\exp(\textstyle\sum_i \mathrm{H}(Y_i|X_i))$$