SinGAN: Learning a Generative Model from a Single Natural Image

Tamar Rott Shaham
Technion

Tali Dekel
Google Research

Tomer Michaeli
Technion

Technion
Israel Institute of Technology

Presented by Qing Lyu

April 1st, 2020

ICCV 2019
Seoul, Korea

What they did?

Generate realistic natural images using an unconditional GAN by only training **one image**

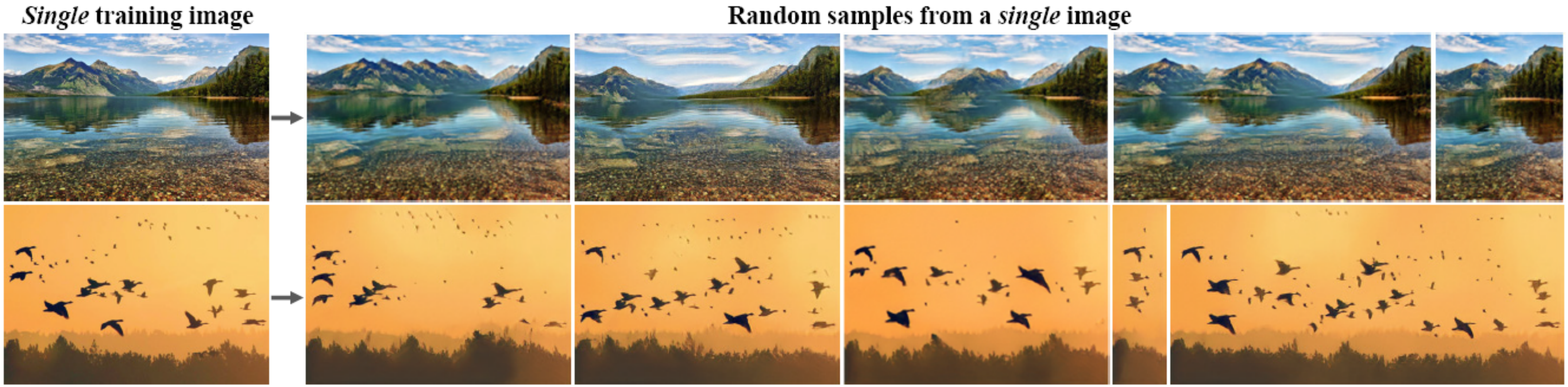
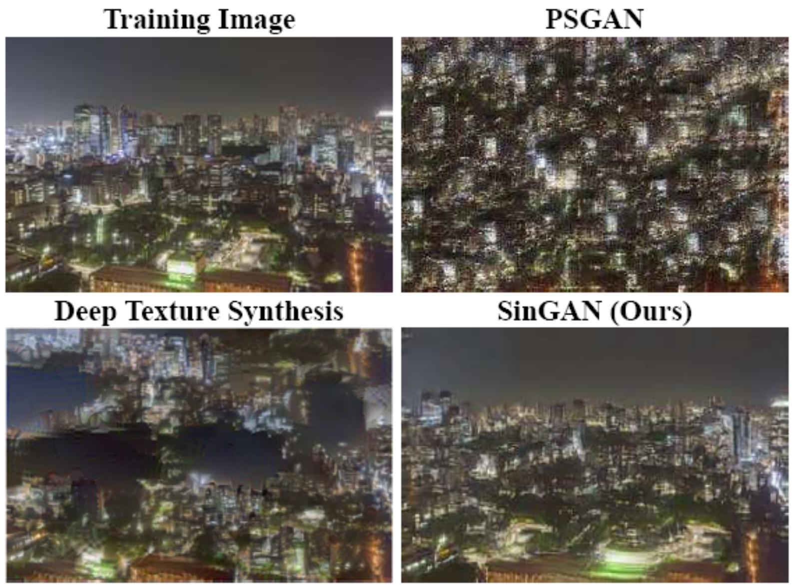


Figure 1: **Image generation learned from a single training image.** We propose *SinGAN*–a new unconditional generative model trained on a *single natural image*. Our model learns the image's patch statistics across multiple scales, using a dedicated multi-scale adversarial training scheme; it can then be used to generate new realistic image samples that preserve the original patch distribution while creating new object configurations and structures.

Why they conduct the experiment?

- Despite GANs have shown remarkable success in generating realistic images, capturing the **distribution of highly diverse datasets** with multiple object classes requires **conditioning the generation** on another input signal or training the model for a specific task
  - e.g. super-resolution, inpainting, retargeting
- These conditioning GAN models requires **large amount** of training dataset and can only be trained to implement a **single task**
- Current unconditional single image GANs do not generate meaningful samples when trained on non-texture images

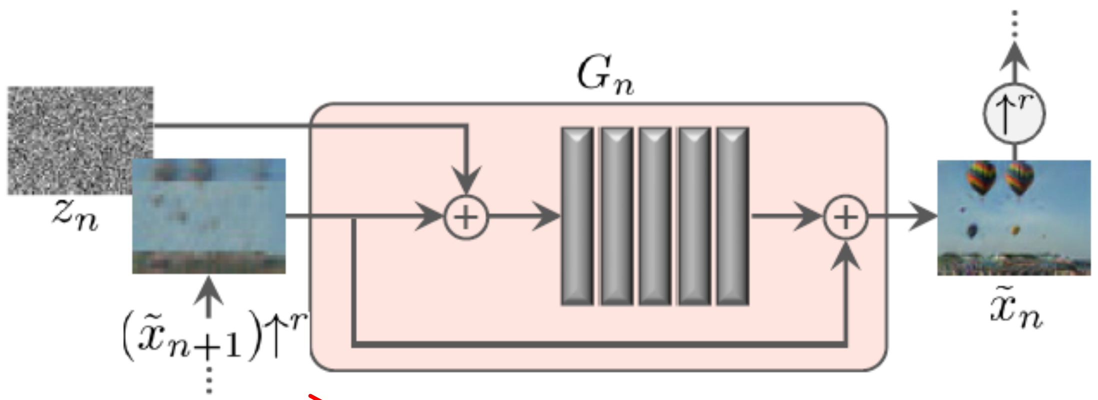


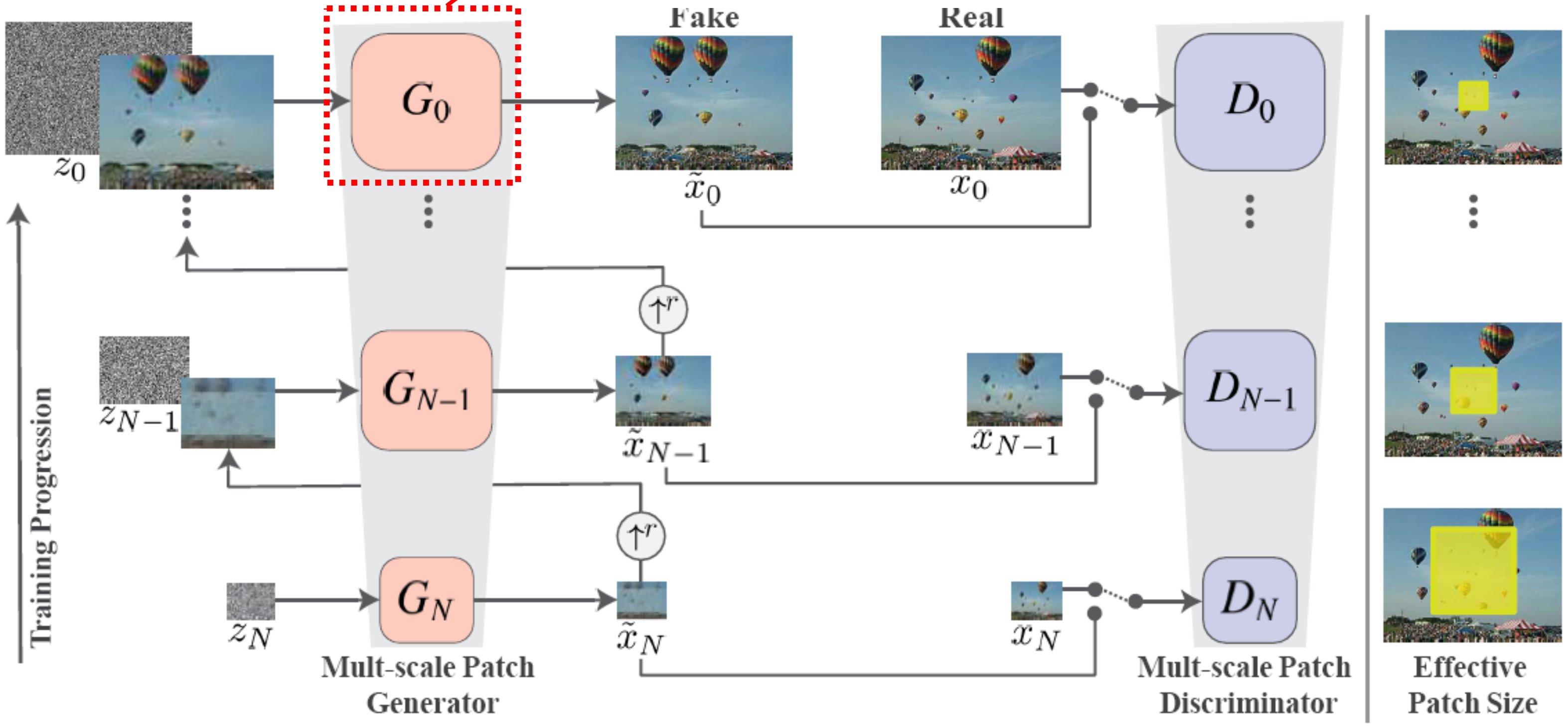
Their contributions

- The authors presented how to train an unconditional GAN model based on a single natural image
- It is shown that the **internal statistics of patches** within a single natural image typically **carry enough information** for learning a powerful generative model
- Once trained, their model can implement **multiple tasks**

How they did?

- Multi-scale architecture
- The model was trained sequentially





Same architecture
Same receptive field

Method in details

$$\tilde{x}_N = G_N(z_N)$$

$$\tilde{x}_n = G_n\left(z_n, (\tilde{x}_{n+1}) \uparrow^r\right), \qquad n < N$$

G_n :

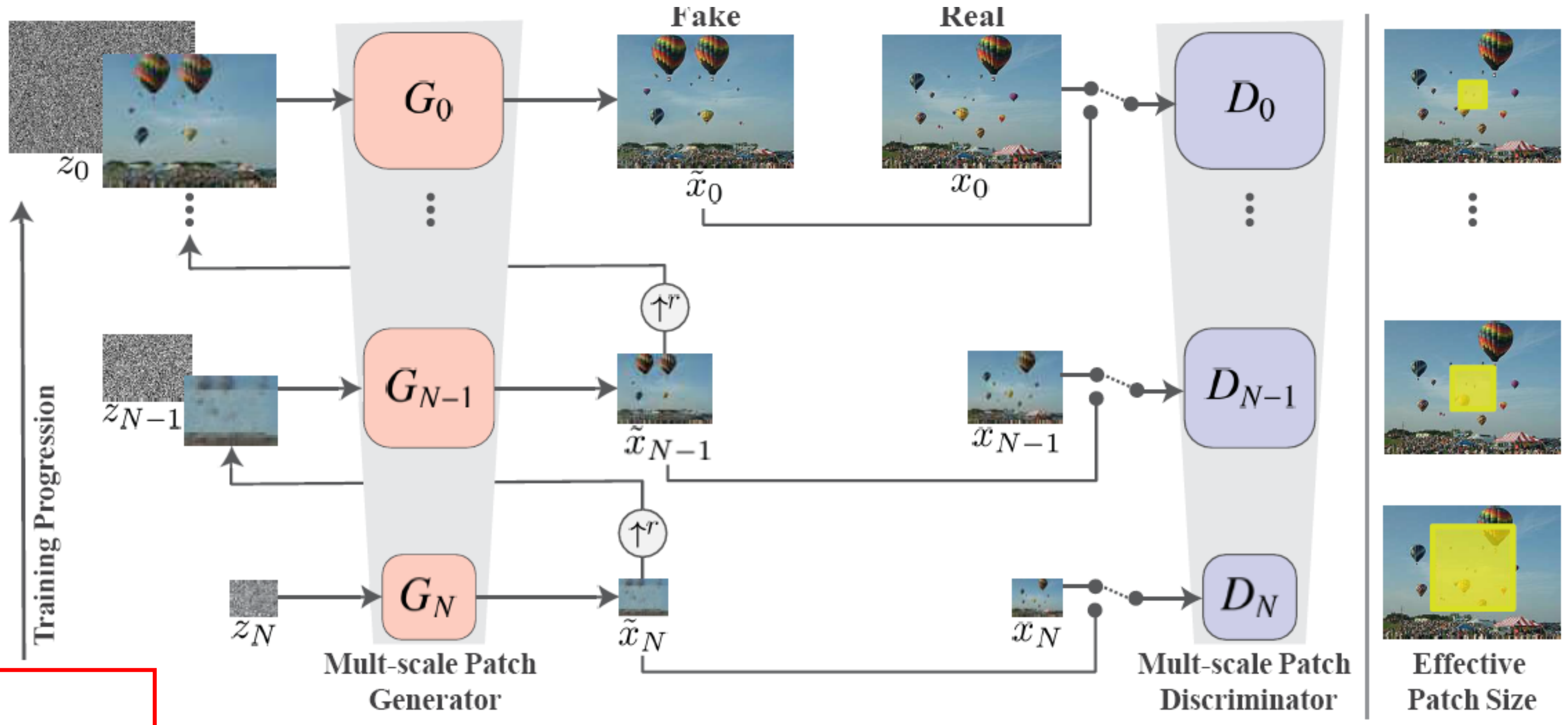
$$\tilde{x}_n = (\tilde{x}_{n+1}) \uparrow^r + \psi_n\left(z_n + (\tilde{x}_{n+1}) \uparrow^r\right)$$

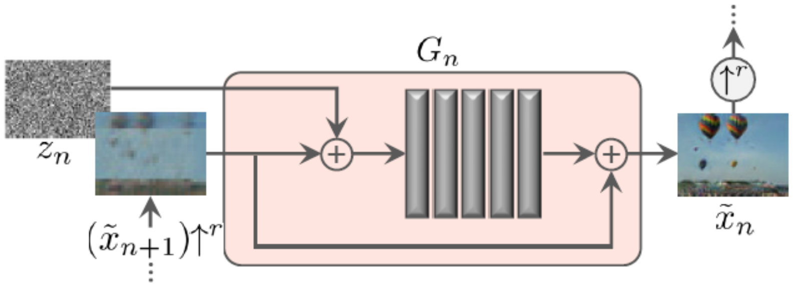
$$\min_{G_n} \max_{D_n} \mathcal{L}_{\text{adv}}(G_n, D_n) + \alpha \mathcal{L}_{\text{rec}}(G_n)$$

For n<N $\quad \mathcal{L}_{\text{rec}} = \|G_n(0, (\tilde{x}^{\text{rec}}_{n+1}) \uparrow^r) - x_n\|^2$

For n=N $\quad \mathcal{L}_{\text{rec}} = \|G_N(z^*) - x_N\|^2$

$$\{z^{\text{rec}}_N, z^{\text{rec}}_{N-1} \dots, z^{\text{rec}}_0\} = \{z^*, 0, \dots, 0\},$$



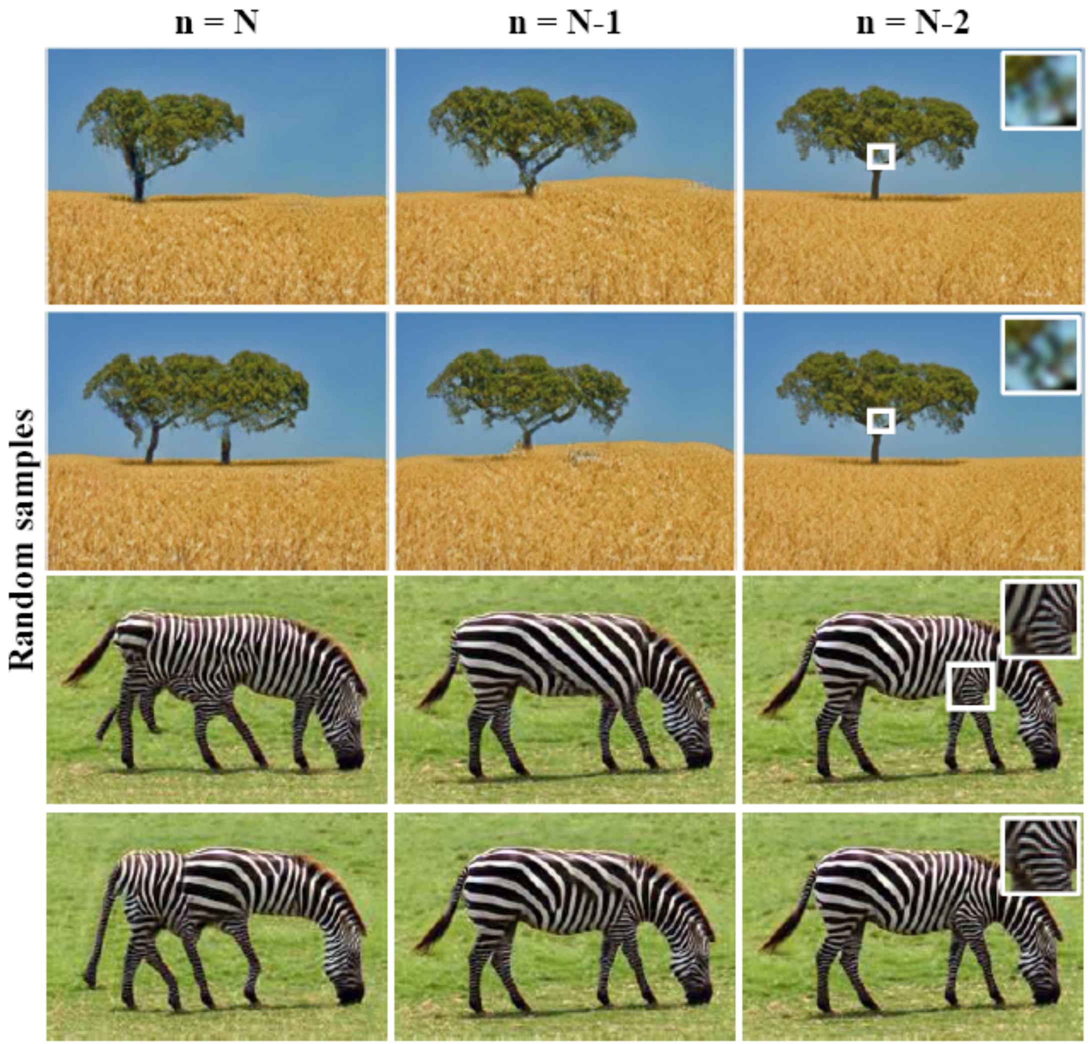


Method in details

- Receptive fields: 11 by 11
- Image size gradually increase by a factor $r \approx {}^{4}/_{3}$
- Image size in the coarsest level: 25 by 25

Result: Effect of scales at test time

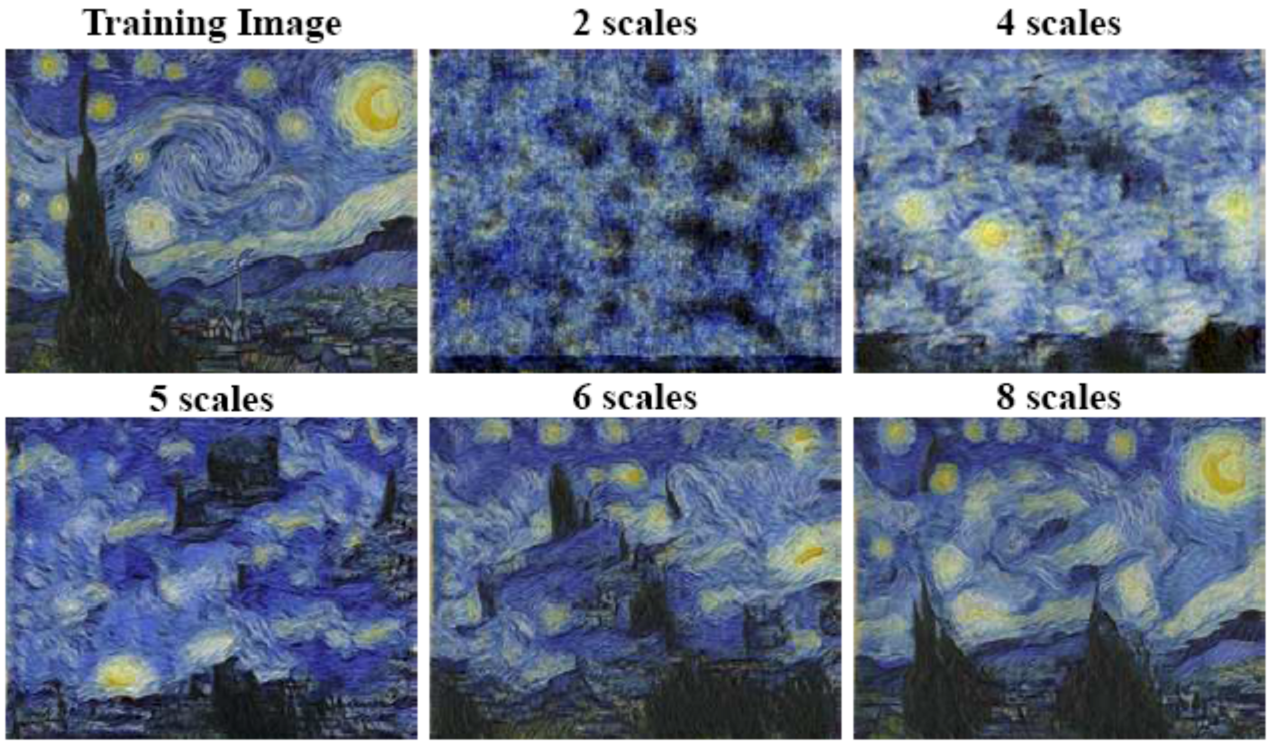
- During the testing, starting from different level



- Starting from finer scales enables to keep the global structure intact, while altering only finer image features

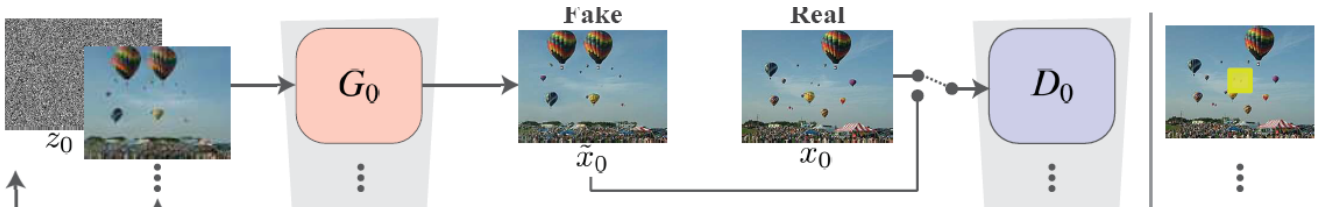
Result: Effect of scales during training

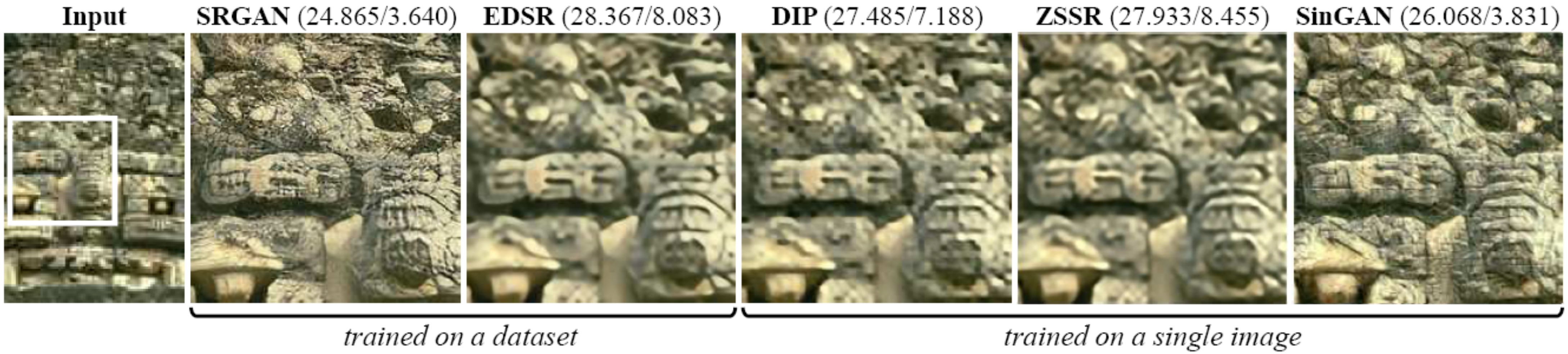
- As the number of scales increases, structures of larger support emerge, and the global object arrangement is better preserved



Application: super-resolution

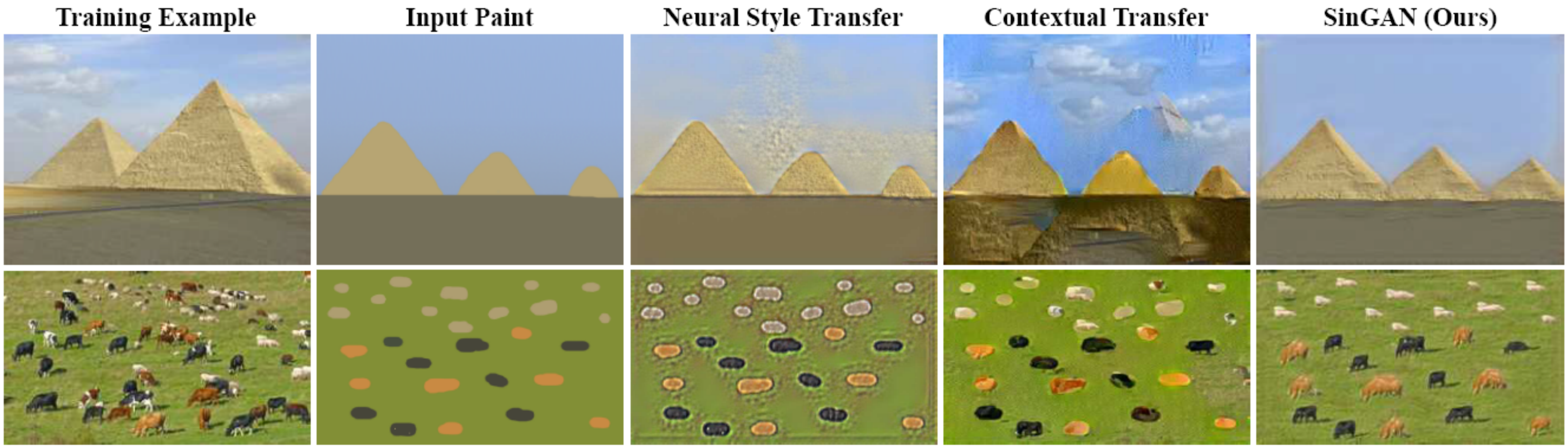
During the testing, upsample the LR image by a factor of *r* and inject it to the last generator, G_0. We repeat this procedure several times to obtain the final high-res output





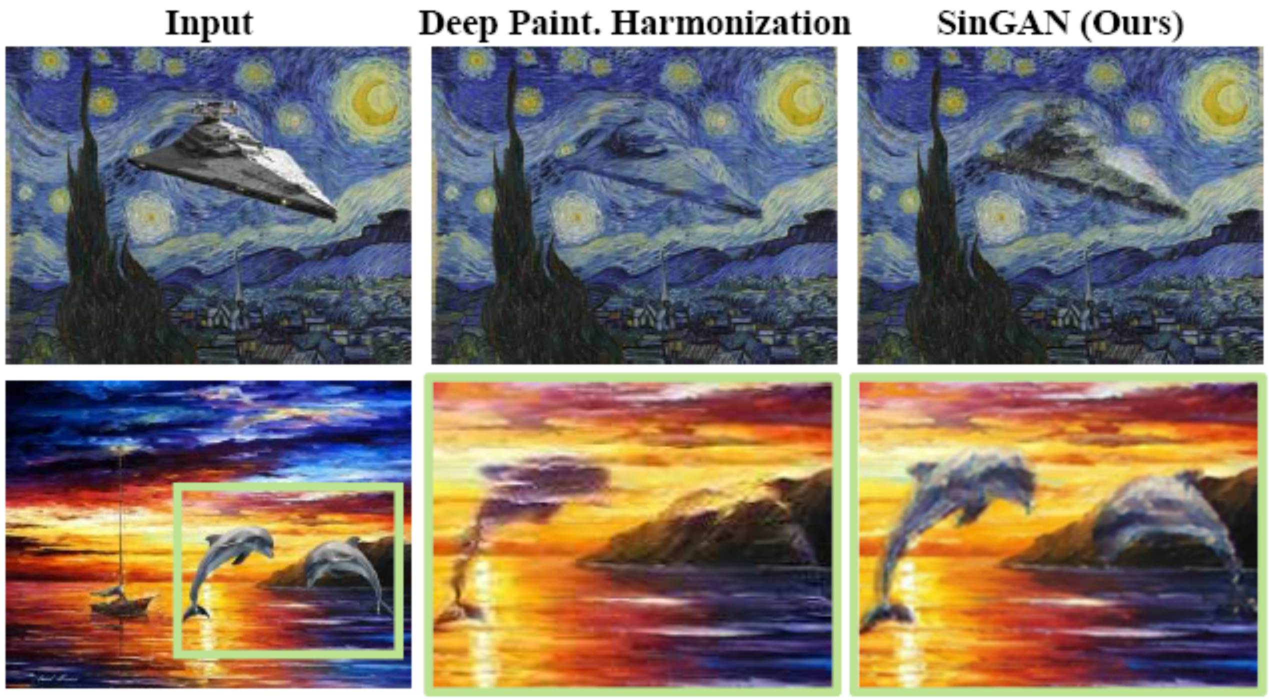
Application: Paint-to-image

Downsampling the clipart image and feeding it into one of the coarse scales (e.g. N-1 or N-2)



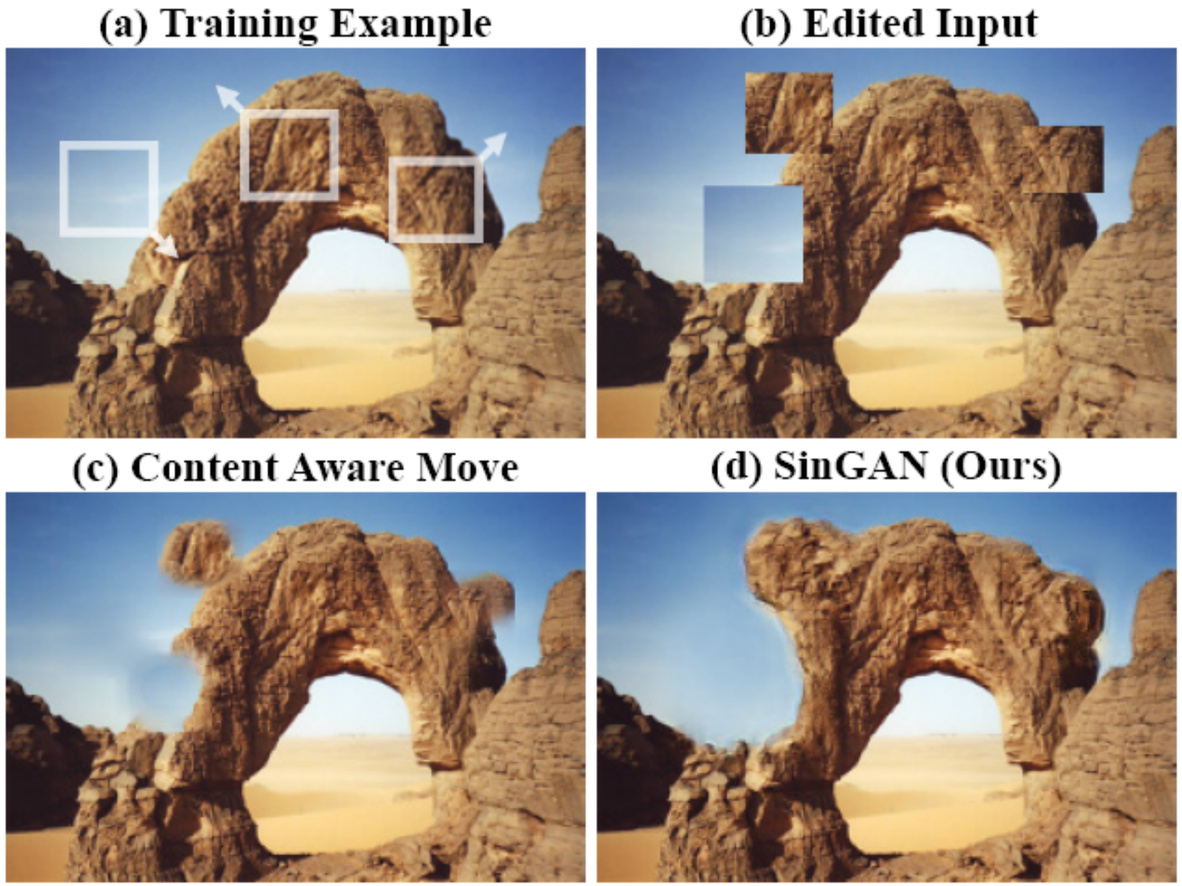
Application: Harmonization

Train the model on the background image, and inject a downsampled version of the naively pasted composite at test time



Application: Editing

Injecting a downsampled version of the composite into one of the coarse scales



More resources

https://www.youtube.com/watch?v=OvZVxqUmE3g
Start from 1:23:20