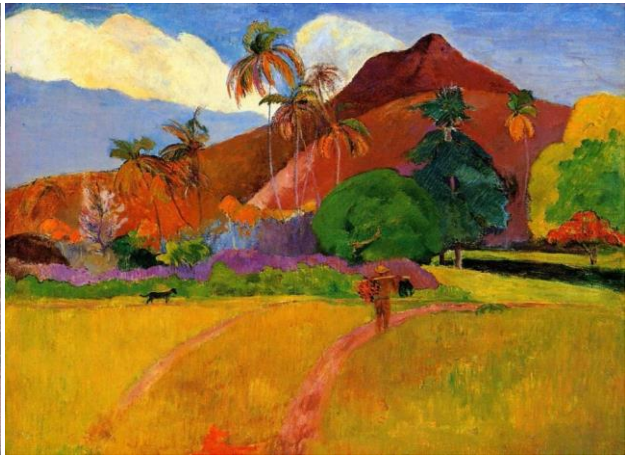
GauGAN/SPADE
Semantic Image Synthesis with Spatially Adaptive Normalization

Ming-Yu Liu

NVIDIA

Image credit: https://www.smithsonianmag.com/history/journey-oldest-cave-paintings-world-180957685/

Cave painting



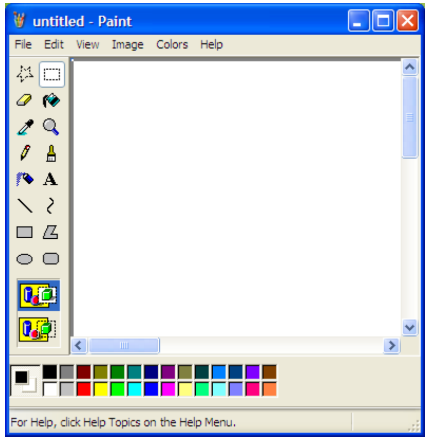
By Gauguin

By fabulouswalrus
using MS Paint

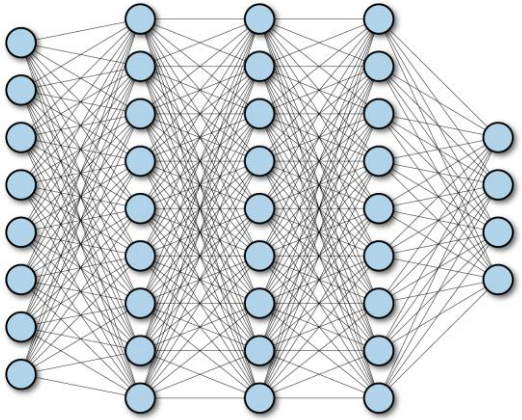
By Pablo Munoz Gomez
using NVIDIA GauGAN

Rock

Brush



Digital revolution

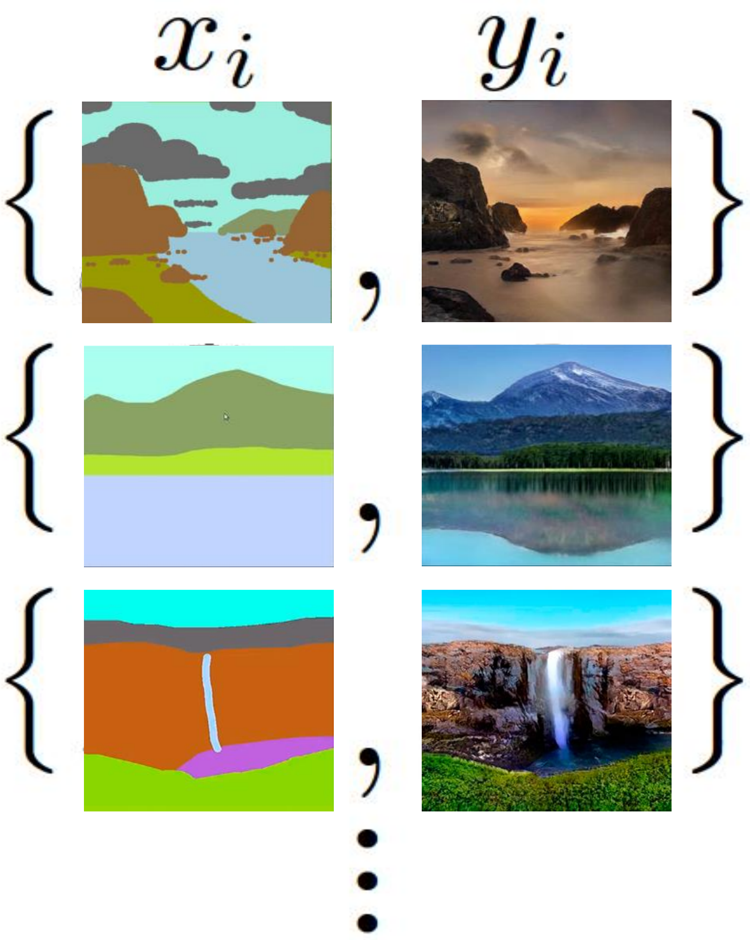


AI revolution

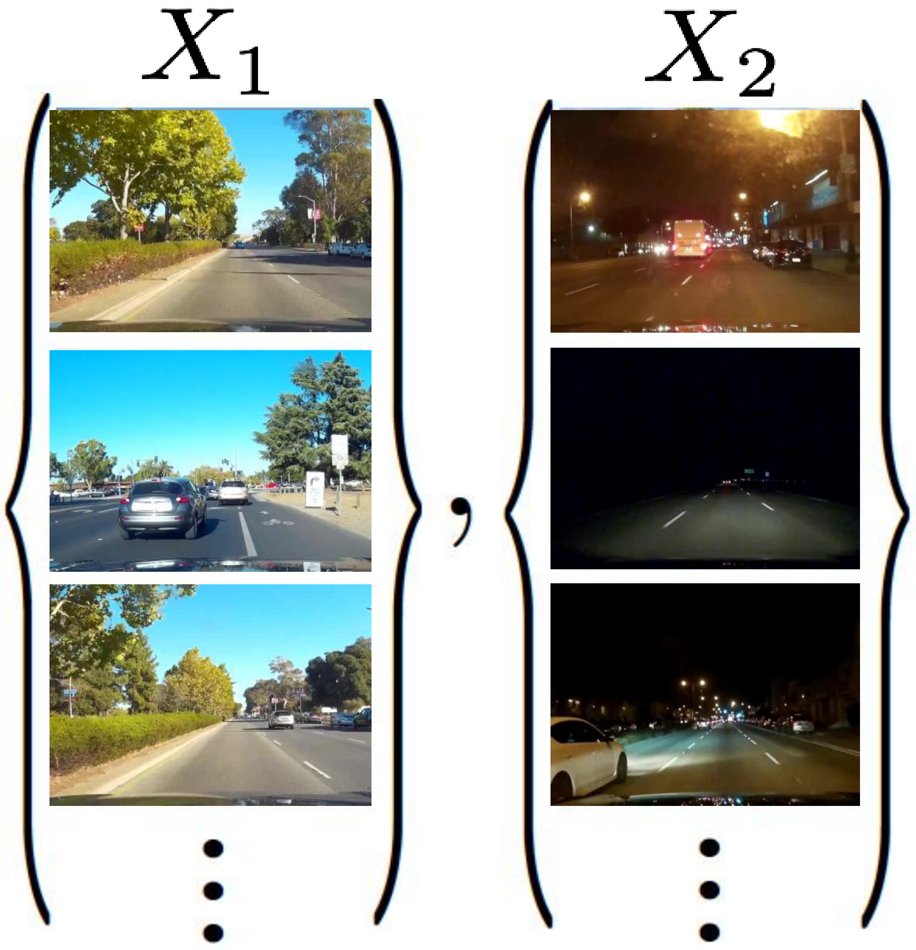
Caveman cartoon image credit: https://canchamthailand.org/caveman-classic-tees-off-april-6th-reserve-spot-now/

Supervised vs Unsupervised

Supervised/Paired/Aligned/Registered



Unsupervised/Unpaired/Unaligned/Unregistered

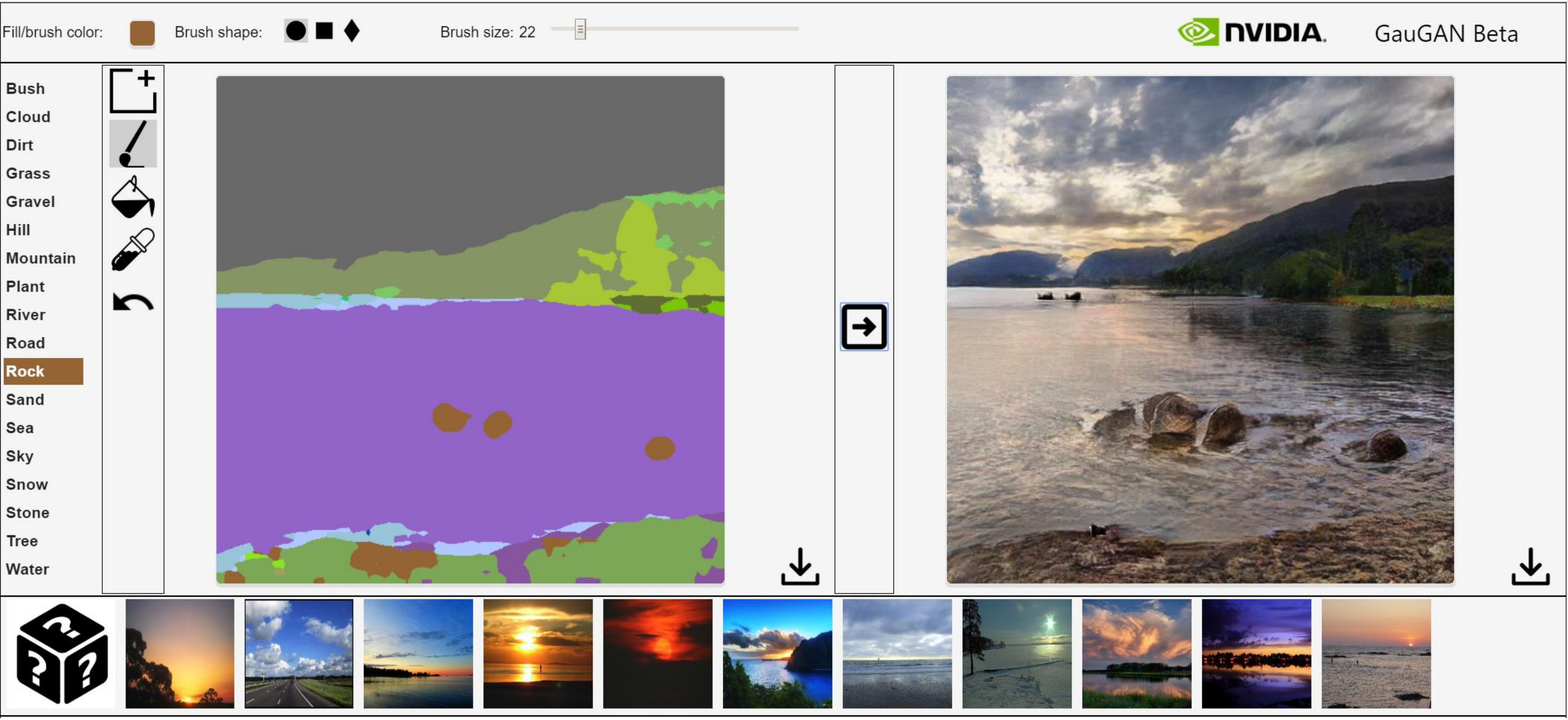


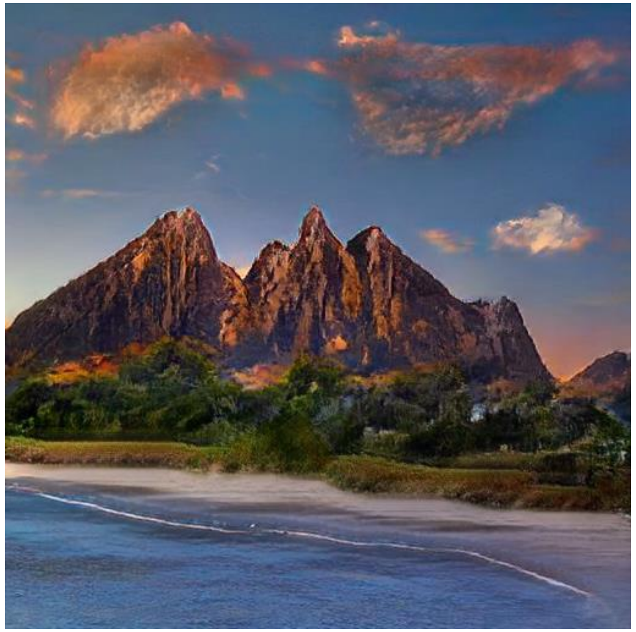
- Supervised/Paired/Aligned/Registered
  - Image Analogy (Hertzmann et. al. 2001)
  - pix2pix (Isola et. al. 2017)
  - CRN (Chen et. al. 2017)
  - BicycleGAN (Zhu et. al. 2017)
  - pix2pixHD (Wang et. al. 2018)
  - SIMS (Qi et. al. 2018)
  - SPADE (Park et. al. 2019)
  - …

$$D_1 \xrightarrow{F} D_2$$

- Unsupervised/Unpaired/Unaligned/Unregistered
  - CoupledGAN (Liu et. al. 2016)
  - DTN (Taigman et. al. 2017)
  - DiscoGAN (Kim et. al. 2017)
  - CycleGAN (Zhu et. al. 2017)
  - SimGAN (Shrivastava et. al. 2017)
  - DualGAN (Yi et. al. 2017)
  - UNIT (Liu et. al. 2017)
  - MUNIT, 2018 (Huang et. al. 2018)
  - DRIT (Lee et. al. 2018)
  - XGAN (Royer et. al. 2018)
  - GANimorph (Gokaslan et. al. 2018)
  - OST (Benaim et. al. 2018)
  - FUNIT (Liu et. al. 2019)
  - …

$$D_1 \underset{F_{2\to1}}{\overset{F_{1\to2}}{\rightleftarrows}} D_2$$


Fill/brush color:
Brush shape:
Brush size: 22
NVIDIA.
GauGAN Beta
Bush
Cloud
Dirt
Grass
Gravel
Hill
Mountain
Plant
River
Road
Rock
Sand
Sea
Sky
Snow
Stone
Tree
Water

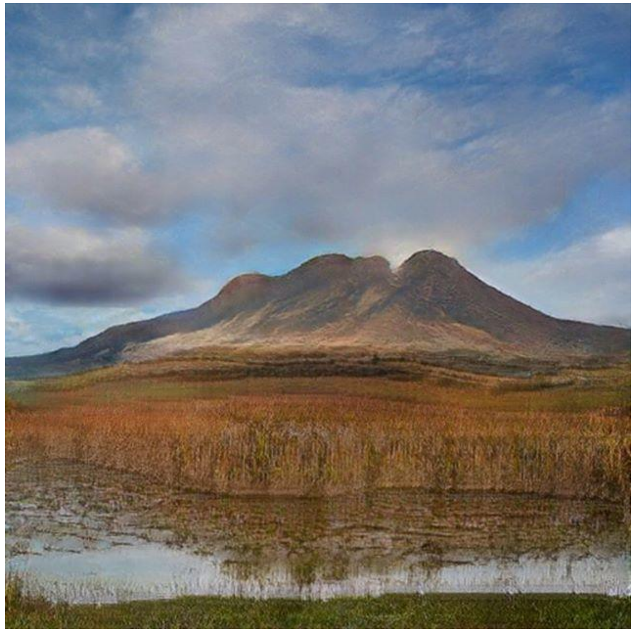


@Soerenpepp

@jonathanfly

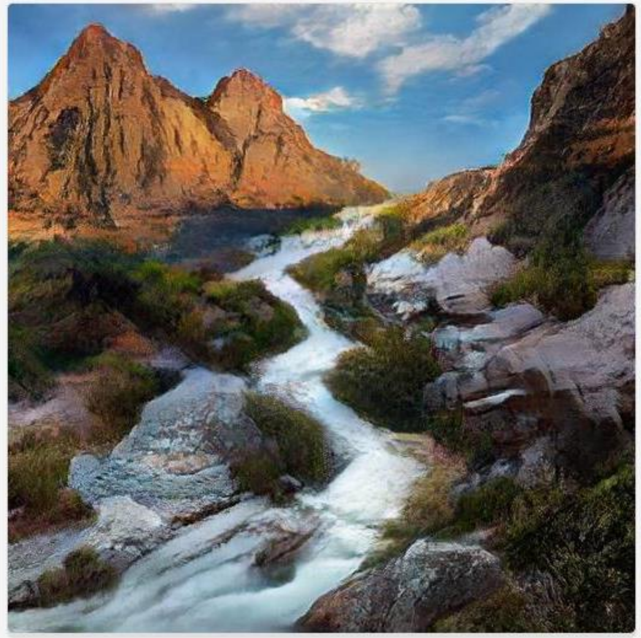
@torans_photo123

@inning0



@frasSmith

@seahcb



@LipComeralla

Tyler Schatz

@coliewertz

@darekzabrocki

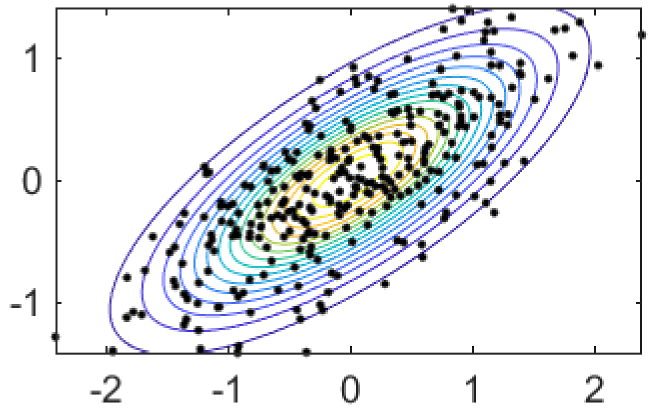
AI generated image

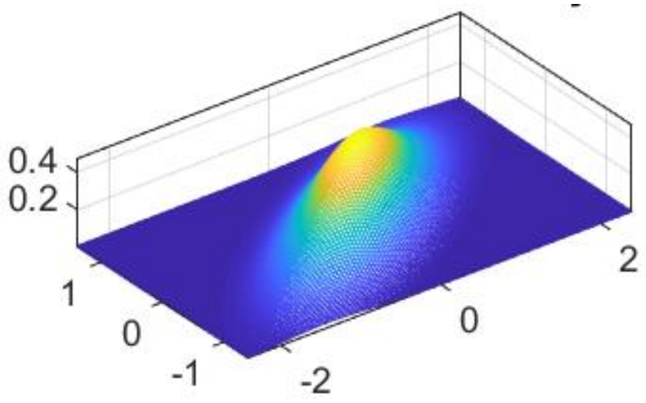
By Jay Axe

By Neil Bickford

How we achieve it?

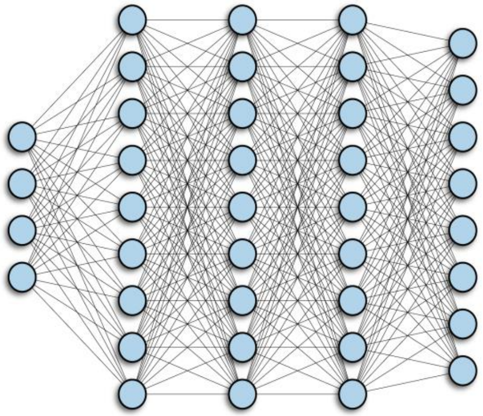
Deep Generative Modeling





Generate new 2D Gaussian samples

$$f(x|\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^d \det(\Sigma)}} \exp\left(-\frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}\right)$$



$$f(z) = \frac{1}{2\pi^{\frac{d}{2}}} e^{\frac{-z^2}{2}}$$

Generate new images

Generative Adversarial Networks

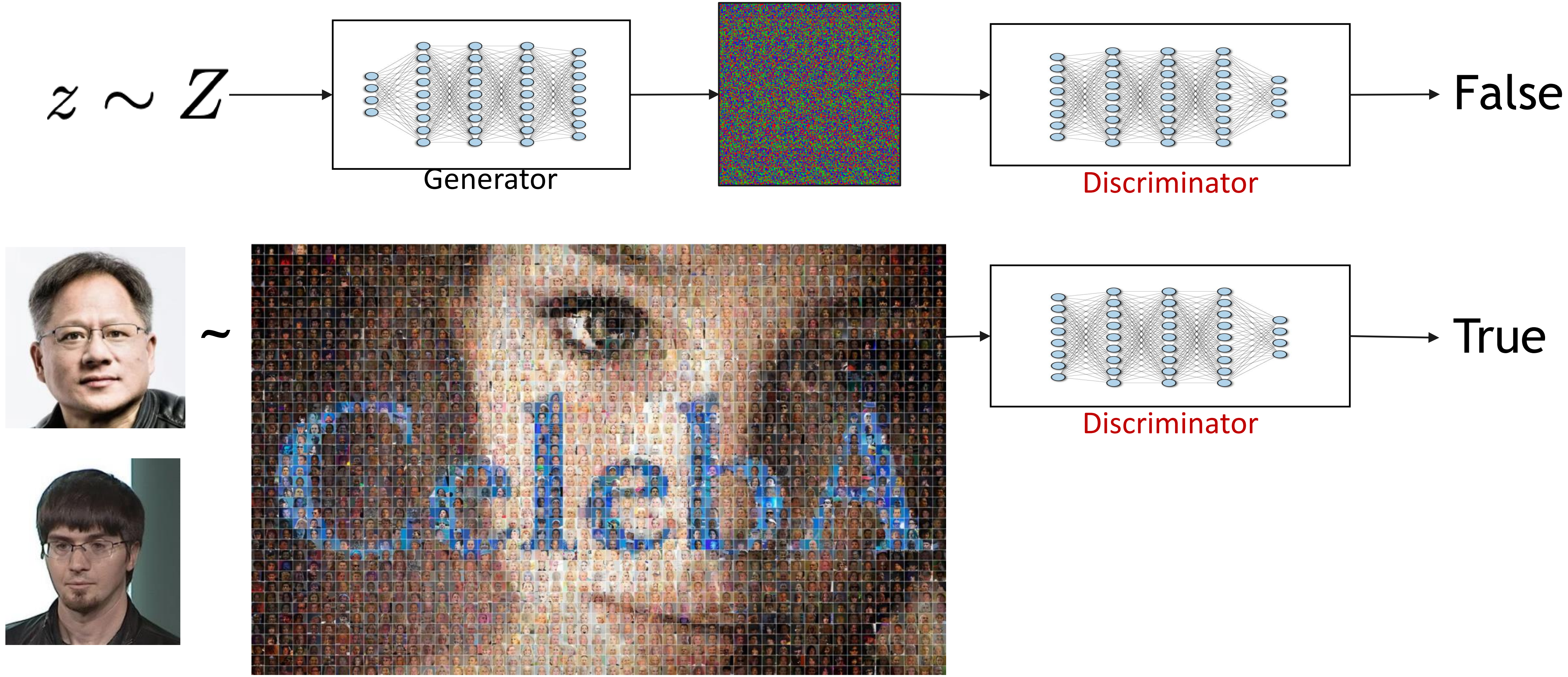
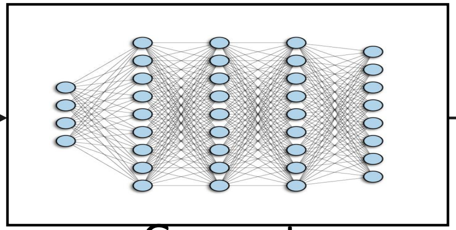


Image credit: Celebrity dataset, Jensen Huang, Founder and CEO of NVIDIA, Ian Goodfellow, Father of GANs.

After training the model for a while

$$z_1, z_2, z_3, \ldots$$



Generator

Image credit: NVIDIA StyleGAN, CVPR 2019

Conditional Generative Adversarial Networks

modeling $p_{X|Y}$

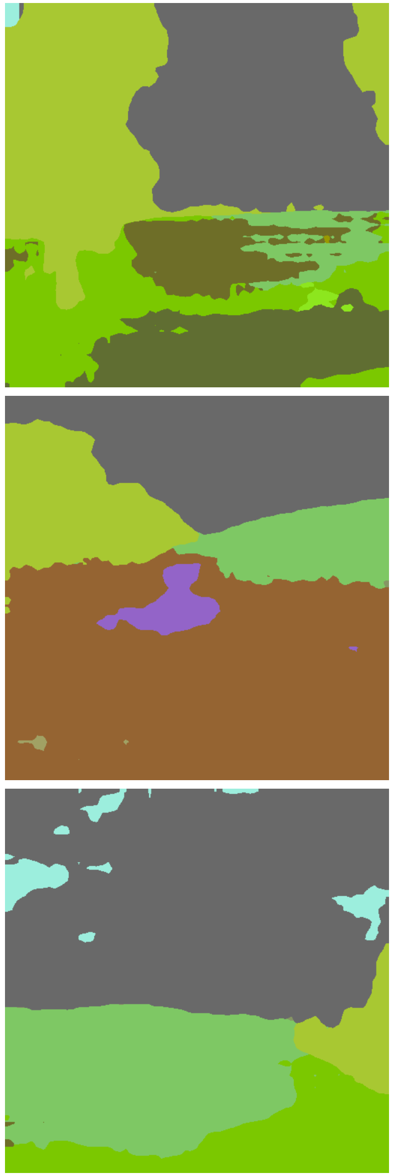
sampling $z \sim Z, y \sim Y$

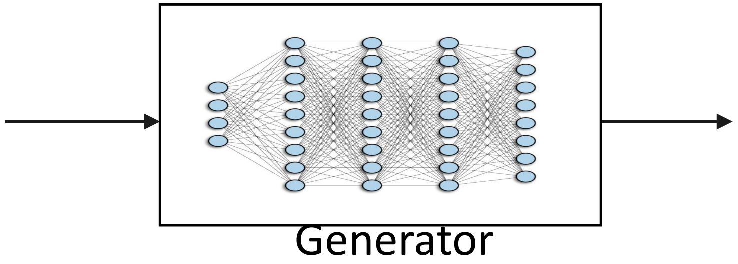
Segmentation Mask–Conditional GANs

$z \sim Z, y_1 \sim Y$

$z \sim Z, y_2 \sim Y$

$z \sim Z, y_3 \sim Y$





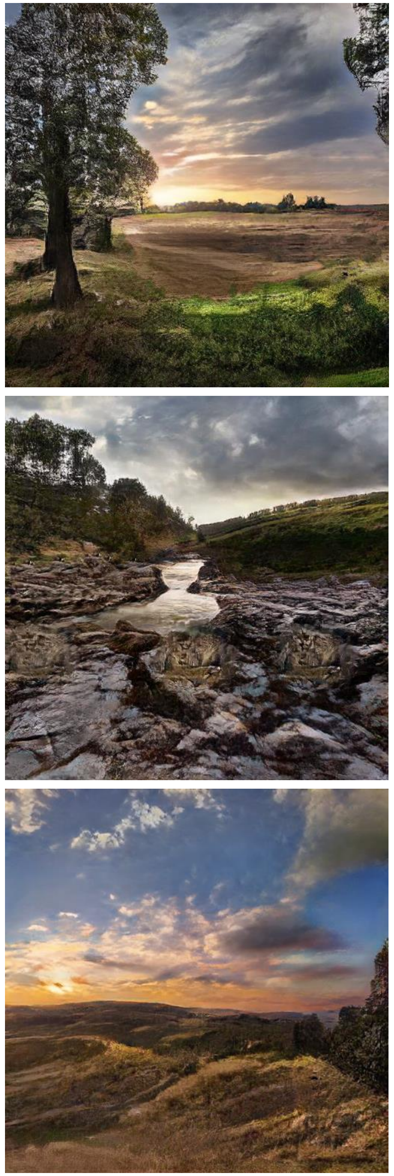
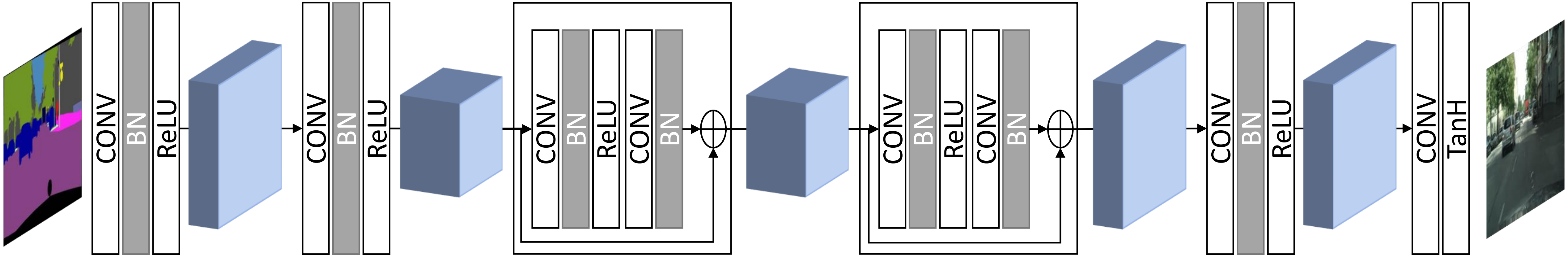


Illustration of pix2pixHD Generator Design

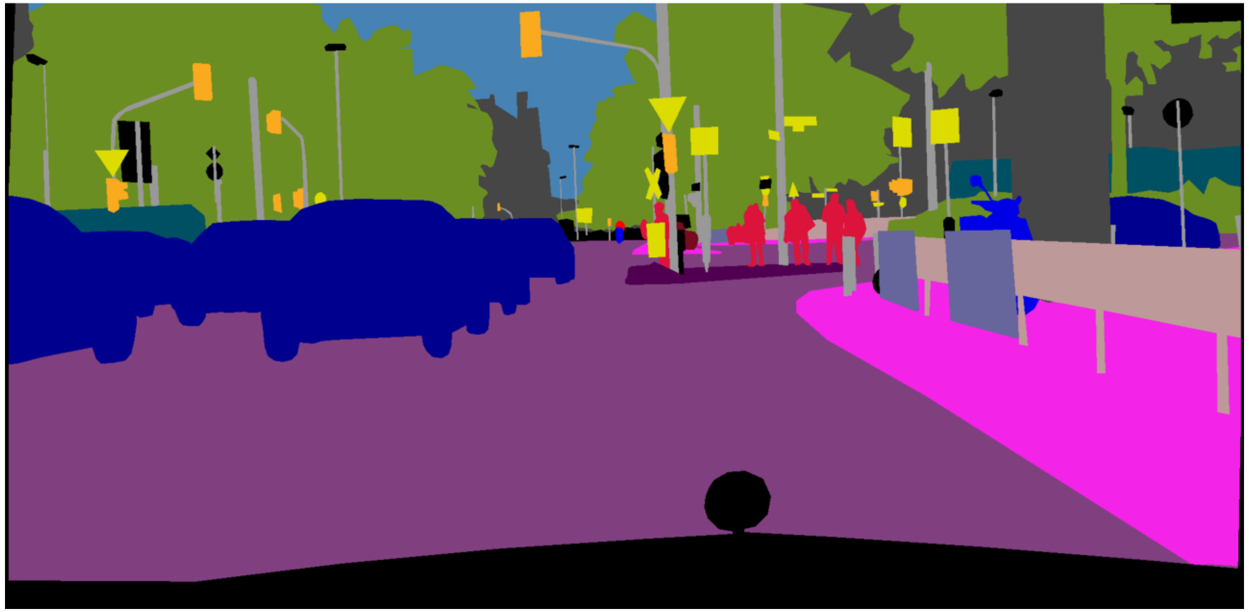


- Previous SOTA method for GAN-based semantic image synthesis
- ResNet-based encoder—decoder architecture
- Work nicely only on highly constrained scenes
- Utilize BatchNorm (BN) or InstanceNorm (IN) in the generator

[pix2pixHD: High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs]()
Ting-Chun Wang, **Ming-Yu Liu**, Jun-Yan Zhu, Andrew Tao, Jan Kautz, Bryan Catanzaro
Conference on Computer Vision and Pattern Recognition (**CVPR**) Oral 2018, Salt Lake City, Utah

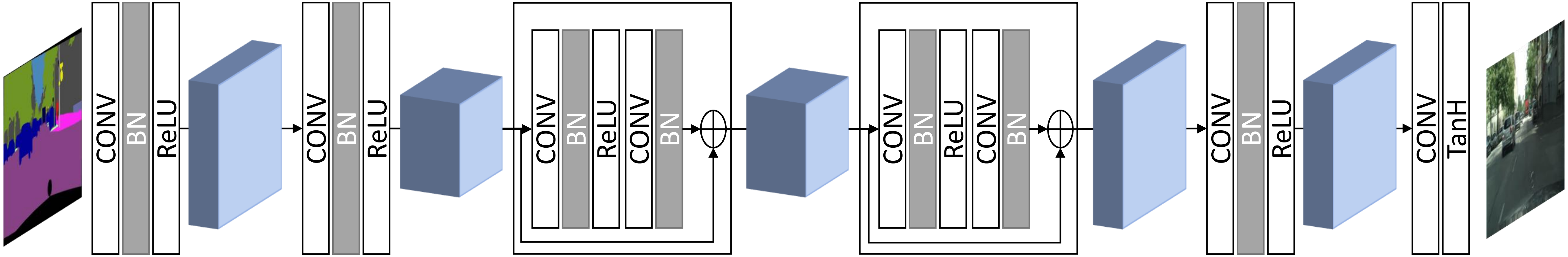
pix2pixHD results

Input labels



Synthesized image

BN: Batch Normalization



$$\tilde{h}^{(l)}_{n,c,y,x} = \boxed{\gamma^{(l)}_c} \frac{h^{(l)}_{n,c,y,x} - \mu^{(l)}_c}{\sigma^{(l)}_c} + \boxed{\beta^{(l)}_c}$$

$$\mu_c^{(l)} = \frac{1}{NHW} \sum_{n,y,x} h^{(l)}_{n,c,y,x}$$

$$\sigma_c^{(l)} = \left(\frac{1}{NHW} \sum_{n,y,x} \left(h^{(l)}_{n,c,y,x}\right)^2\right) - {\mu_c^{(l)}}^2$$

Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift

Sergey Ioffe, Christian Szegedy

Internaltional Conference on Machine Learning (ICML) 2015, Lille, France,

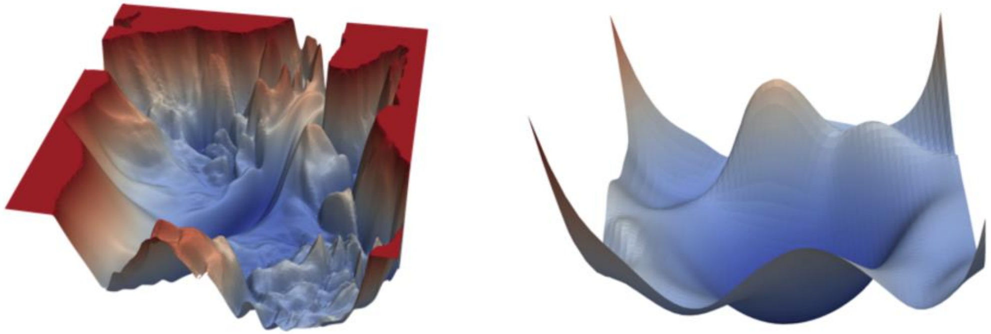
Why Batch Normalization?

- Initial hypothesis: reducing covariance shift in internal activations

Why Batch Normalization?

- ~~Initial hypothesis: reducing covariance shift in internal activations~~
- New hypothesis #1: leading to smoother optimization landscape
- New hypothesis #2: leading to length-direction decoupling of the weight space -> faster convergence rate

Loss landscape illustration



Without BN

With BN

$$\tilde{w} = \boxed{\gamma}\boxed{\frac{w}{||w||_s}}$$

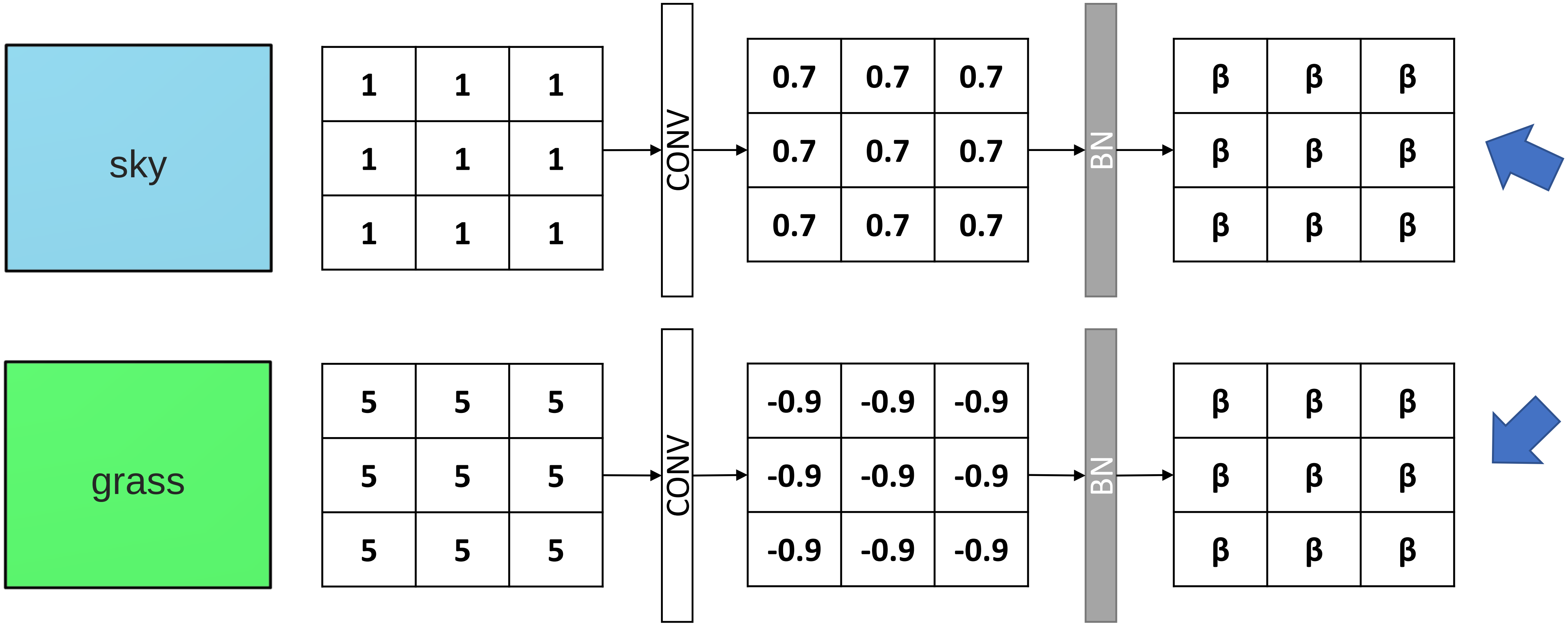
length

direction

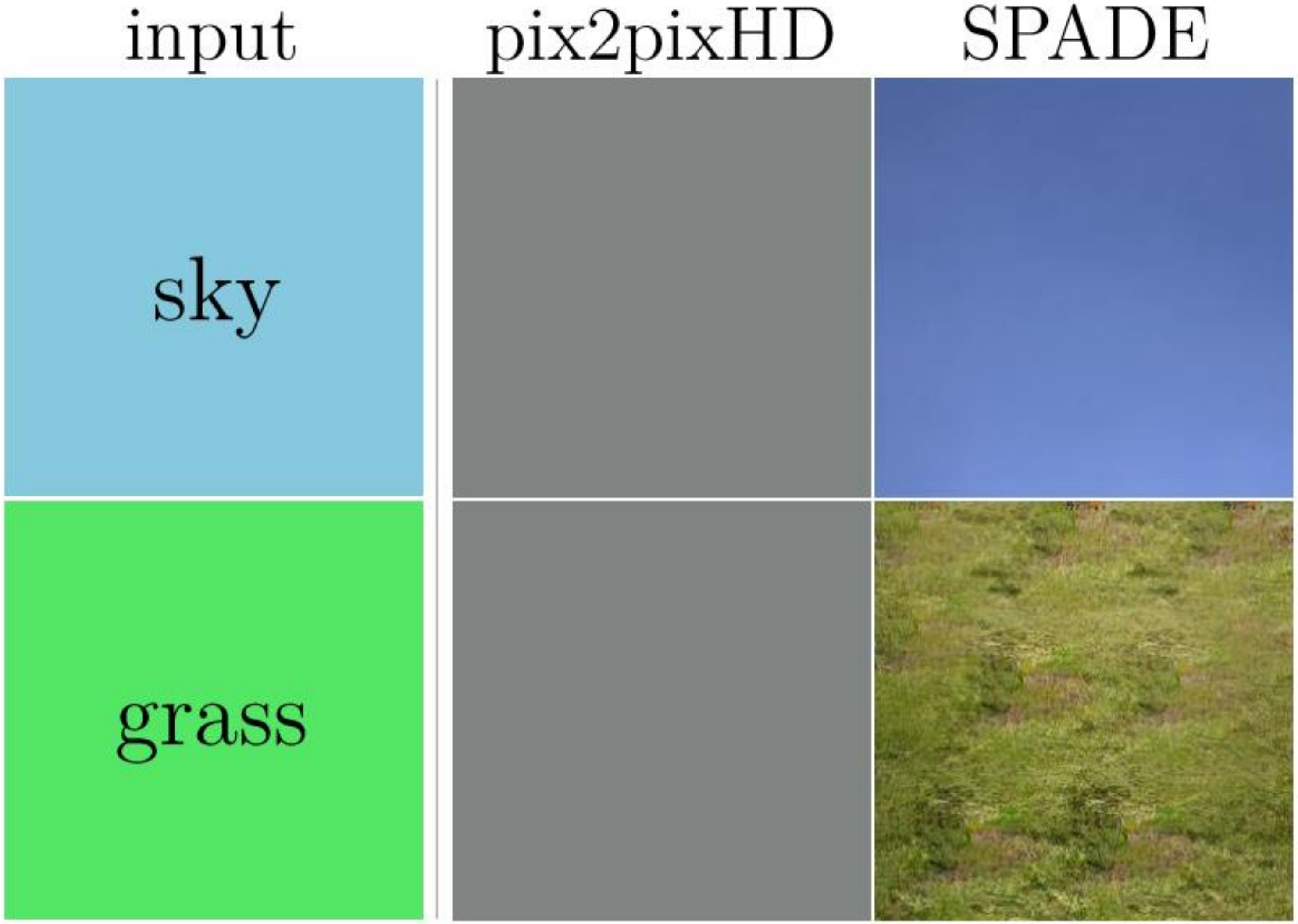
Issue with using Batch Normalization for Semantic Image Synthesis

- It tends to wash away semantic input information.

$$\tilde{h}^{(l)}_{n,c,y,x} = \gamma^{(l)}_c \frac{h^{(l)}_{n,c,y,x} - \mu^{(l)}_c}{\sigma^{(l)}_c} + \beta^{(l)}_c$$



Become identical, all the semantic information is gone.


input
pix2pixHD
SPADE
sky
grass

Segmentation masks often contains large uniform regions

SPADE: SPatially Adaptive DEnormalization

BN
$$\tilde{h}^{(l)}_{n,c,y,x} = \gamma^{(l)}_c \frac{h^{(l)}_{n,c,y,x} - \mu^{(l)}}{\sigma^{(l)}_c} + \beta^{(l)}_c$$

SPADE
$$\tilde{h}^{(l)}_{n,c,y,x} = \boxed{\gamma^{(l)}_{c,y,x}(s)} \frac{h^{(l)}_{n,c,y,x} - \mu^{(l)}}{\sigma^{(l)}_c} + \boxed{\beta^{(l)}_{c,y,x}(s)}$$

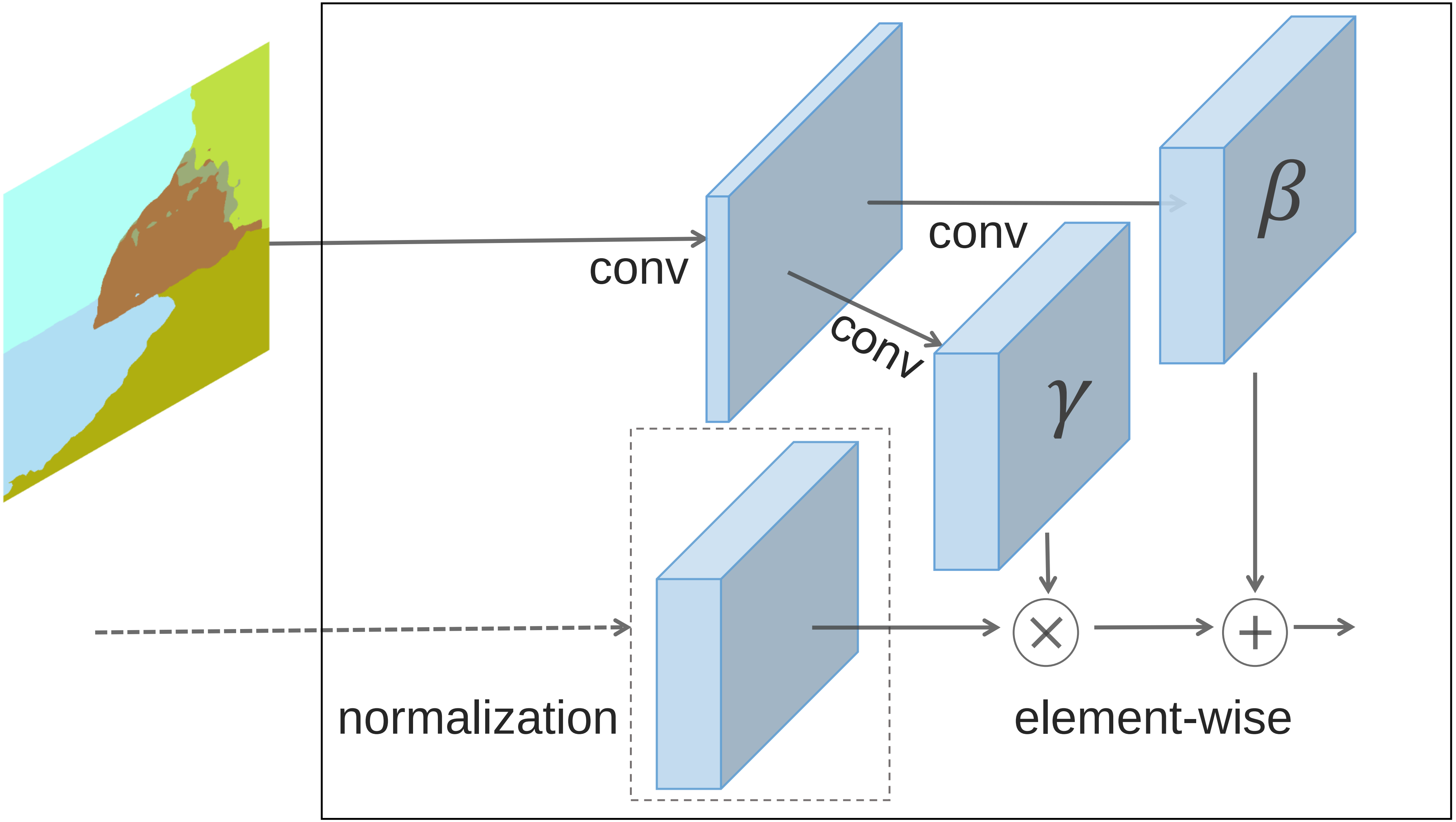
Spatially varying quantity

Depending on the input segmentation mask s

Information removed by normalization can be added back by gamma and beta

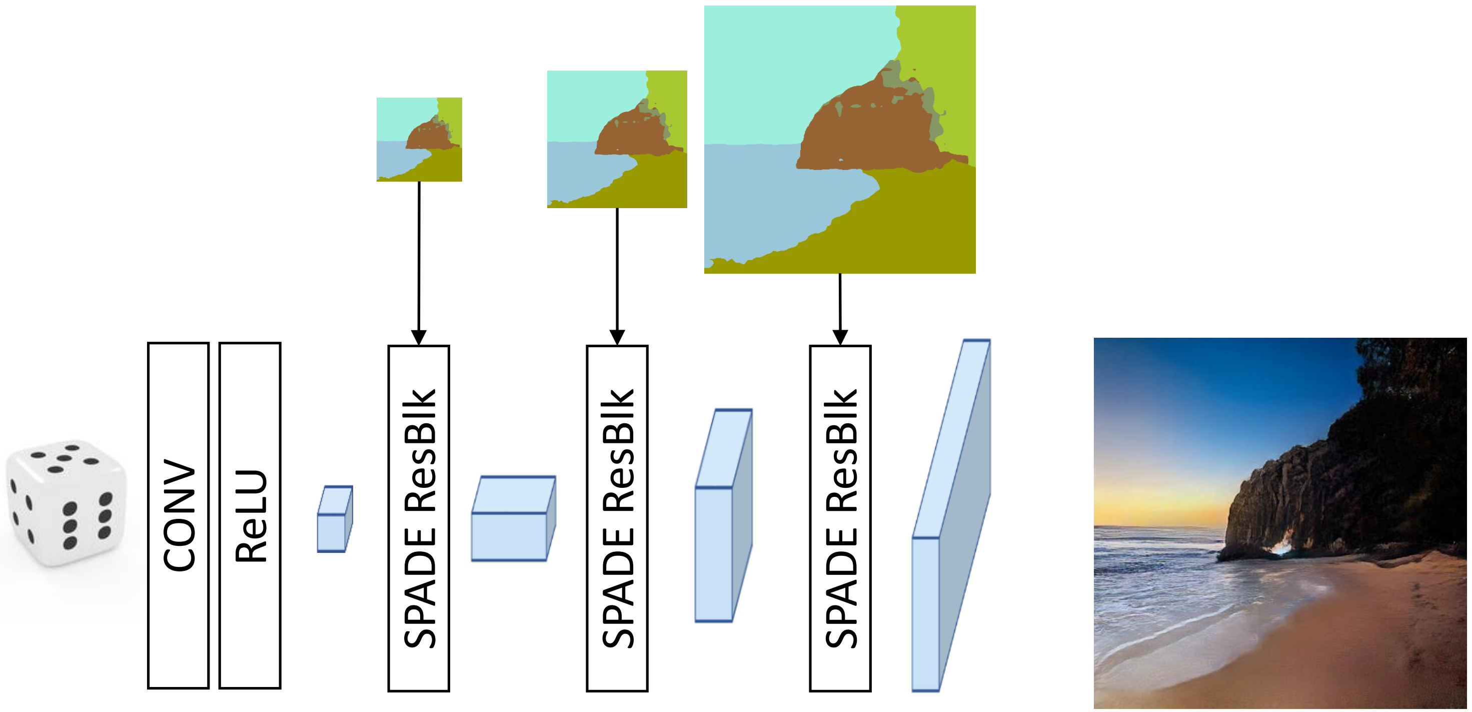
SPADE

$$\tilde{h}^{(l)}_{n,c,y,x} = \gamma^{(l)}_{c,y,x}(s)\frac{h^{(l)}_{n,c,y,x} - \mu^{(l)}}{\sigma^{(l)}_{c}} + \beta^{(l)}_{c,y,x}(s)$$

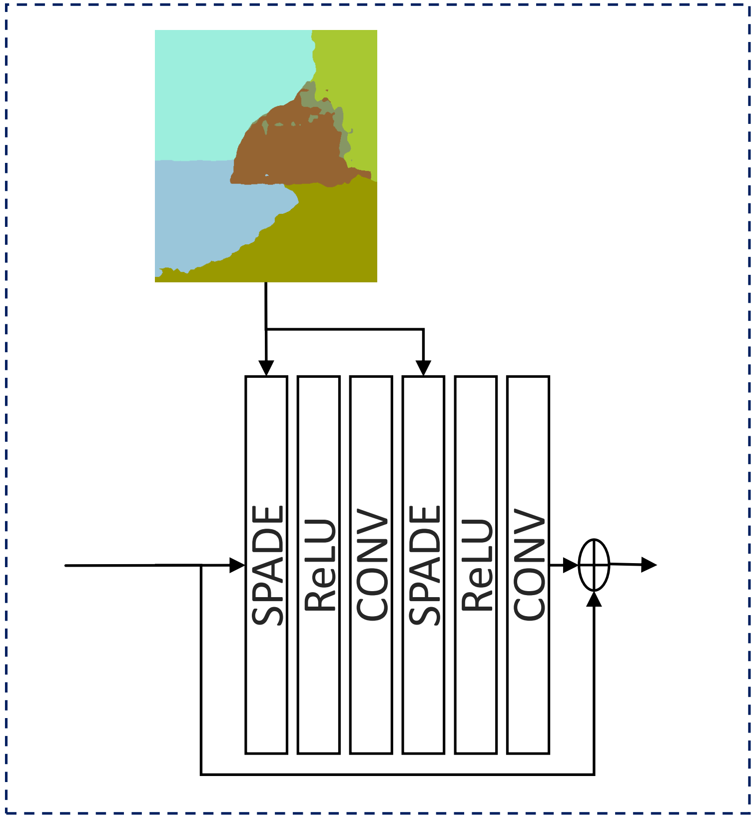


SPADE Module

SPADE-based Generator

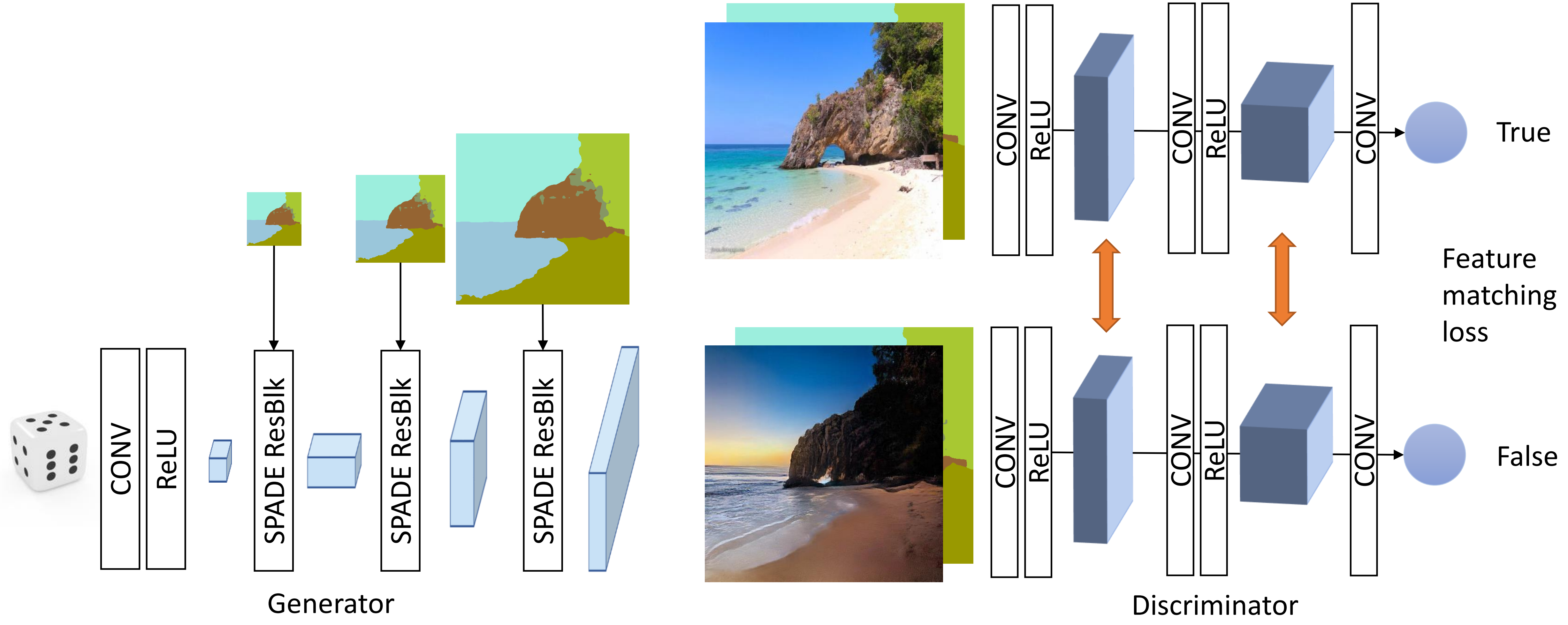


Generator

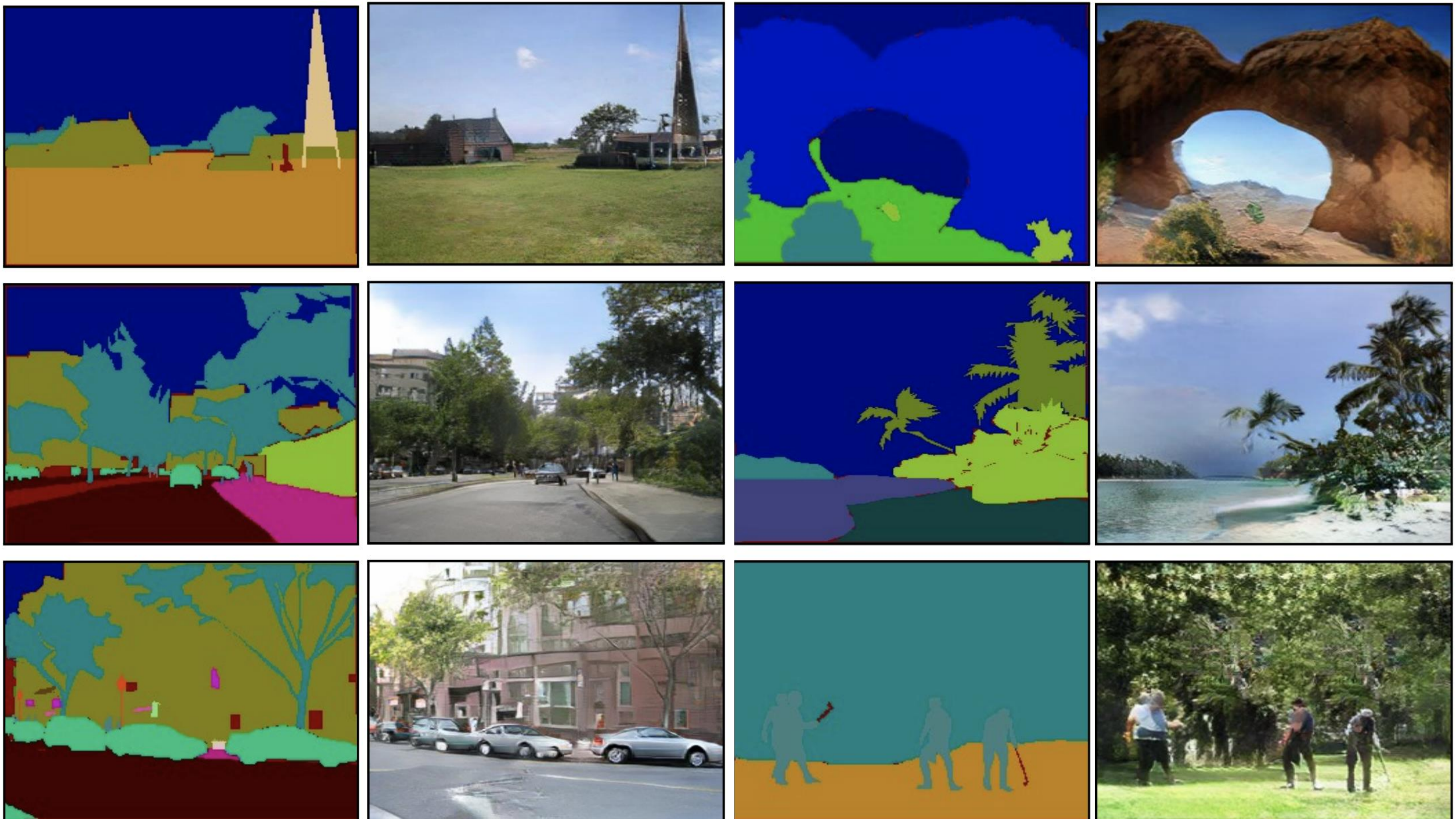


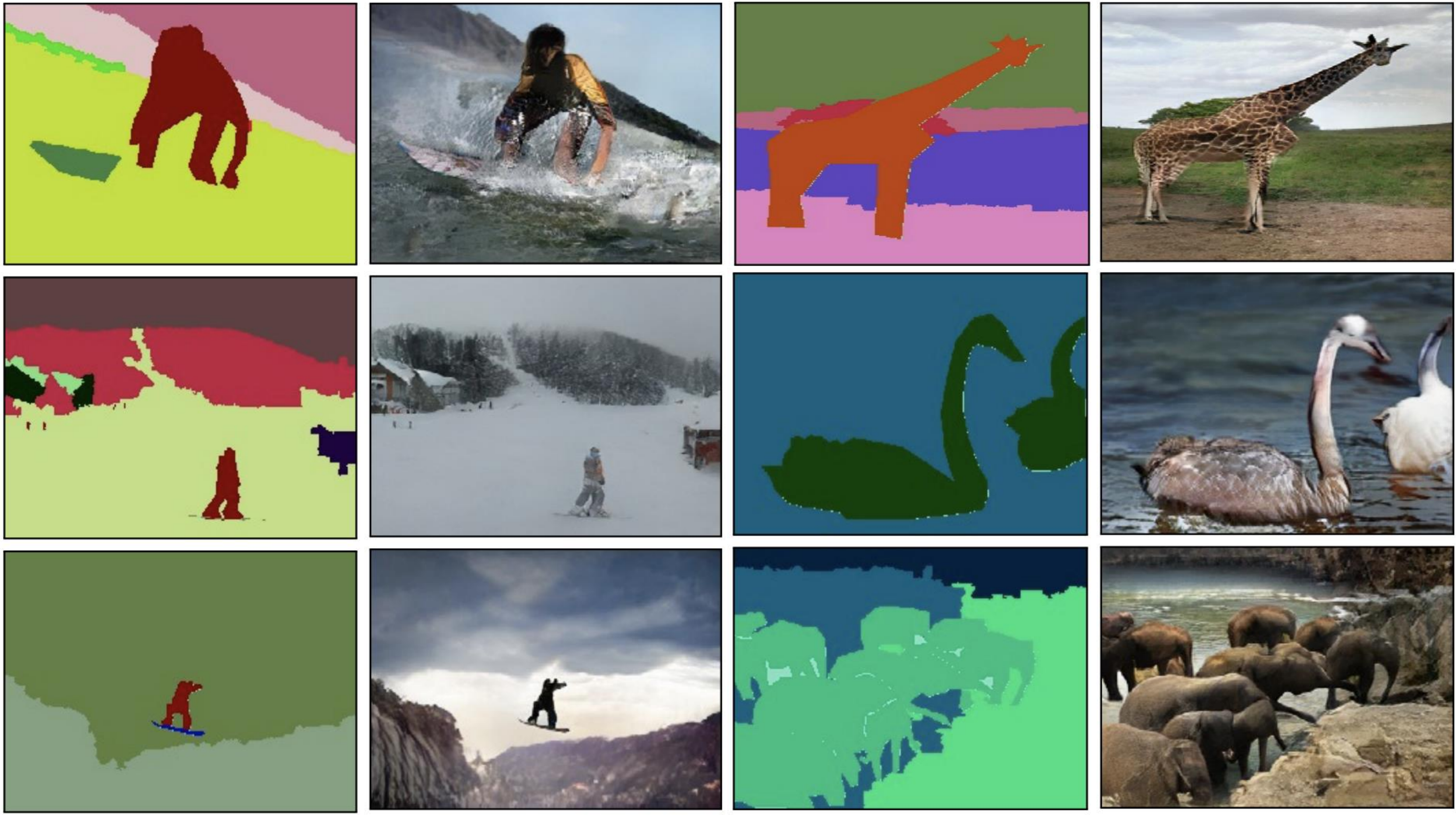
SPADE ResBlk

GauGAN Framework

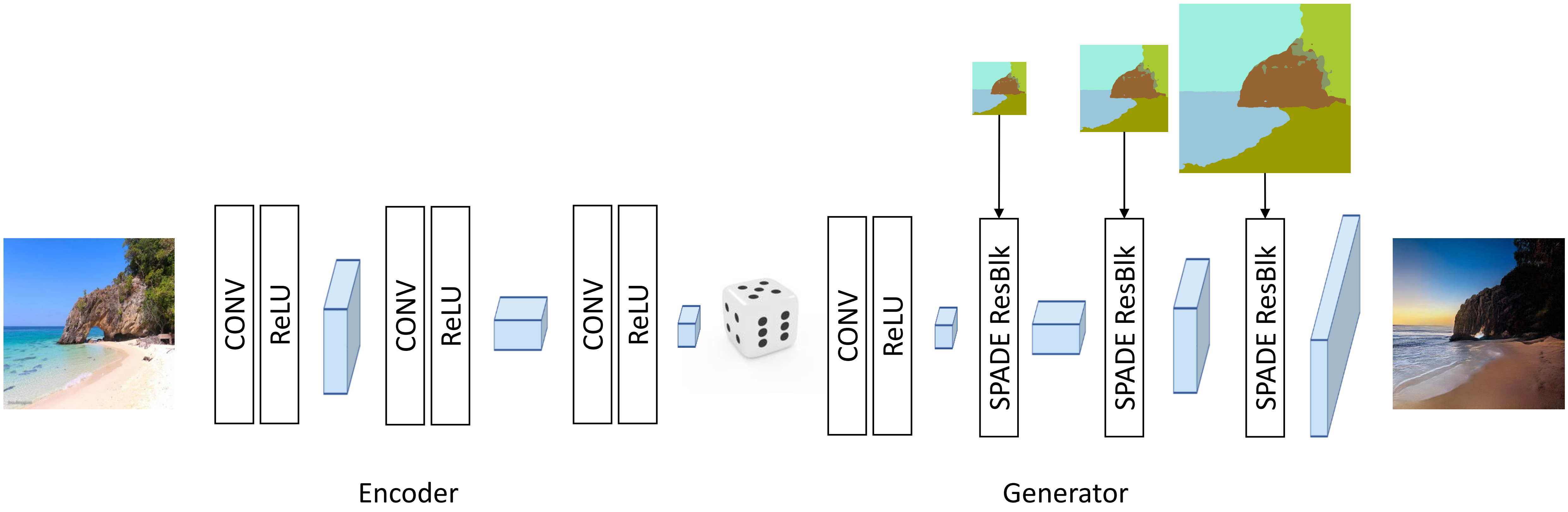


Results

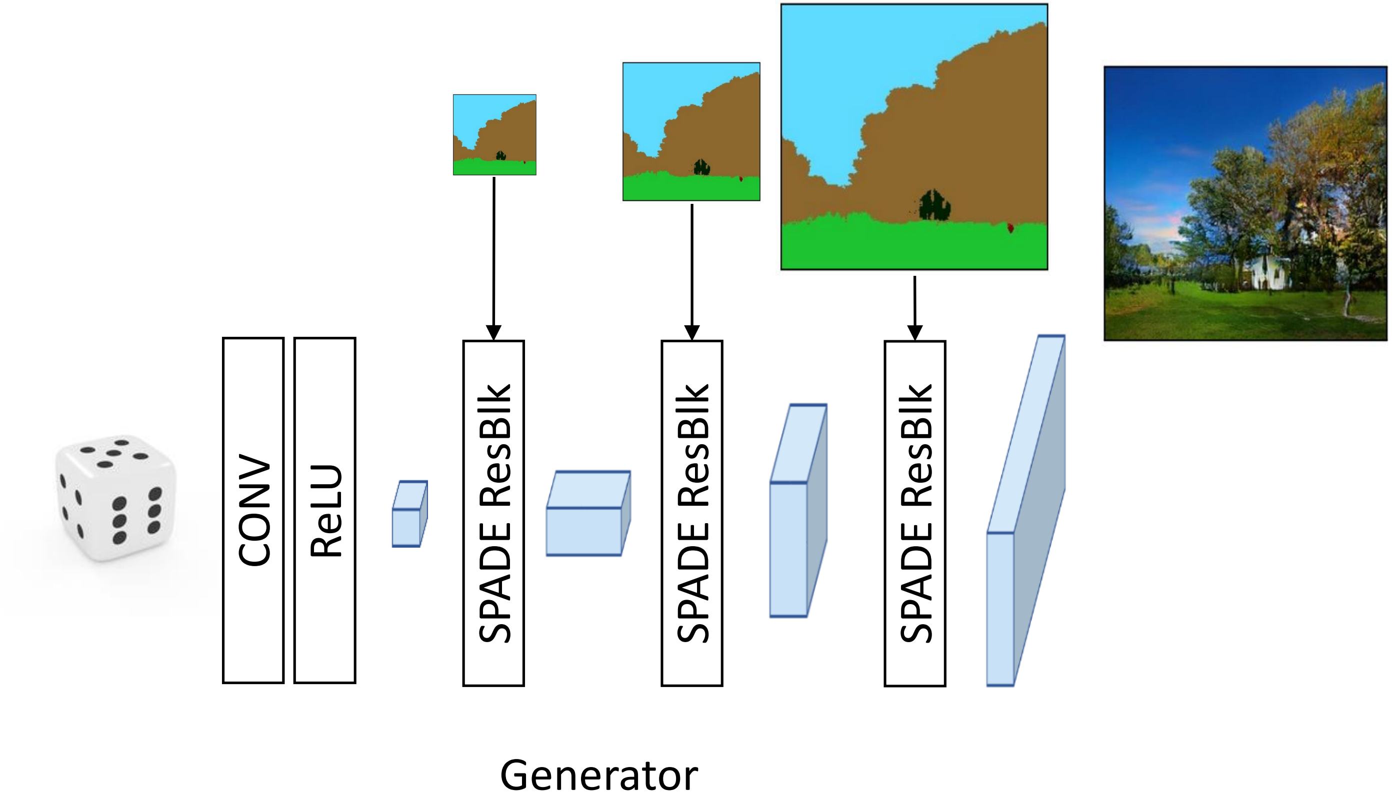




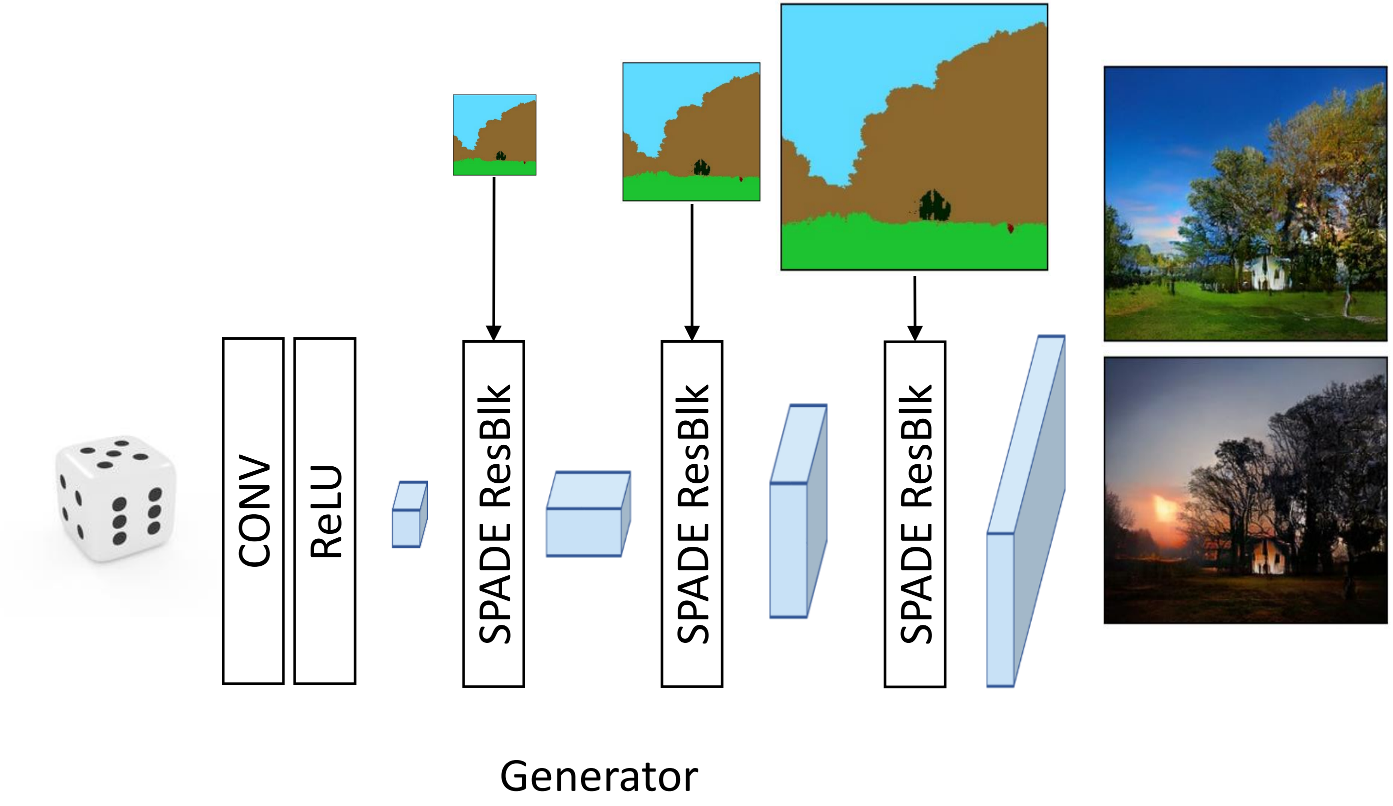
Style Control Learning via a Variational Learning Framework



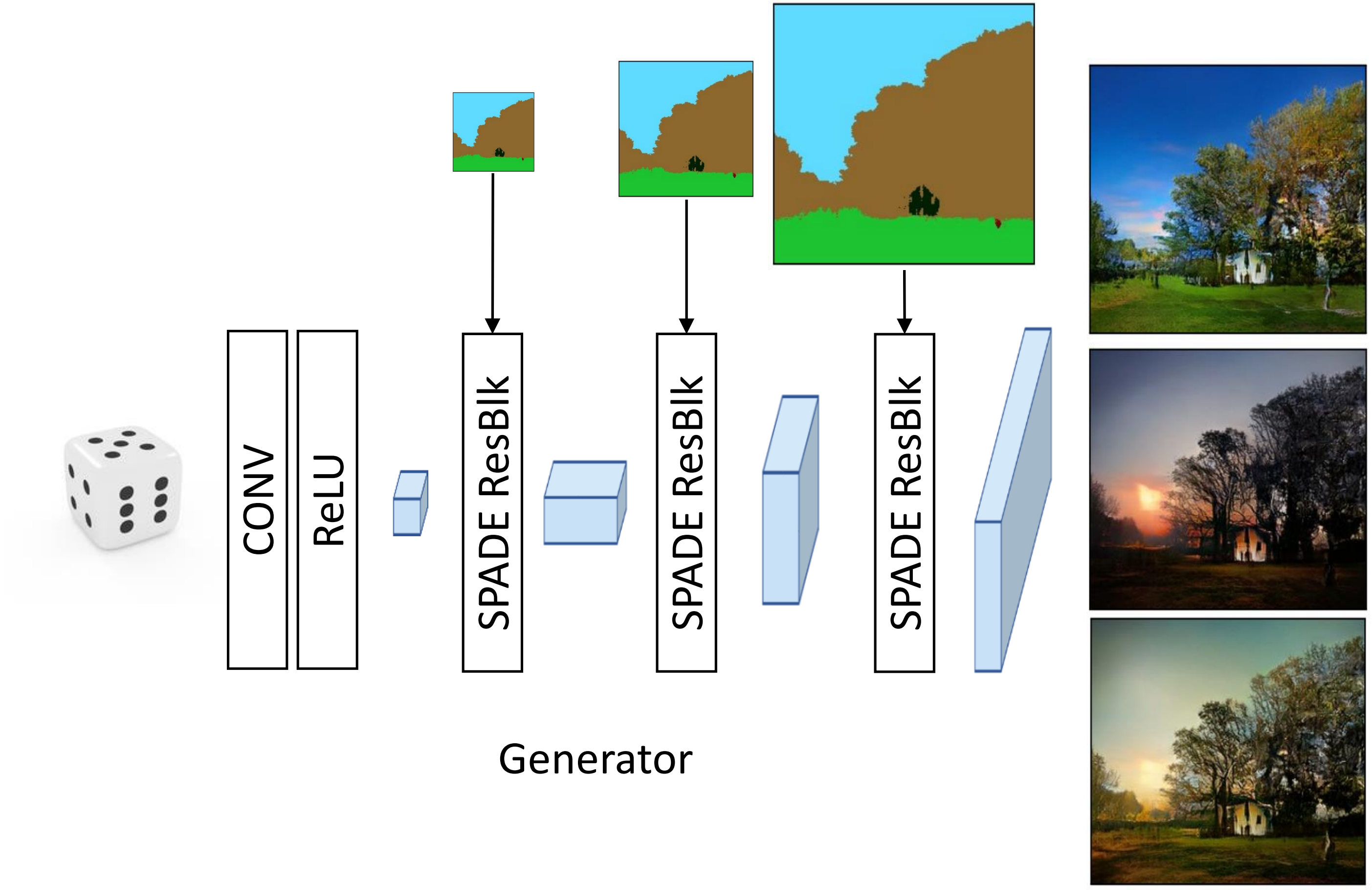
Style Control Learning via a Variational Learning Framework

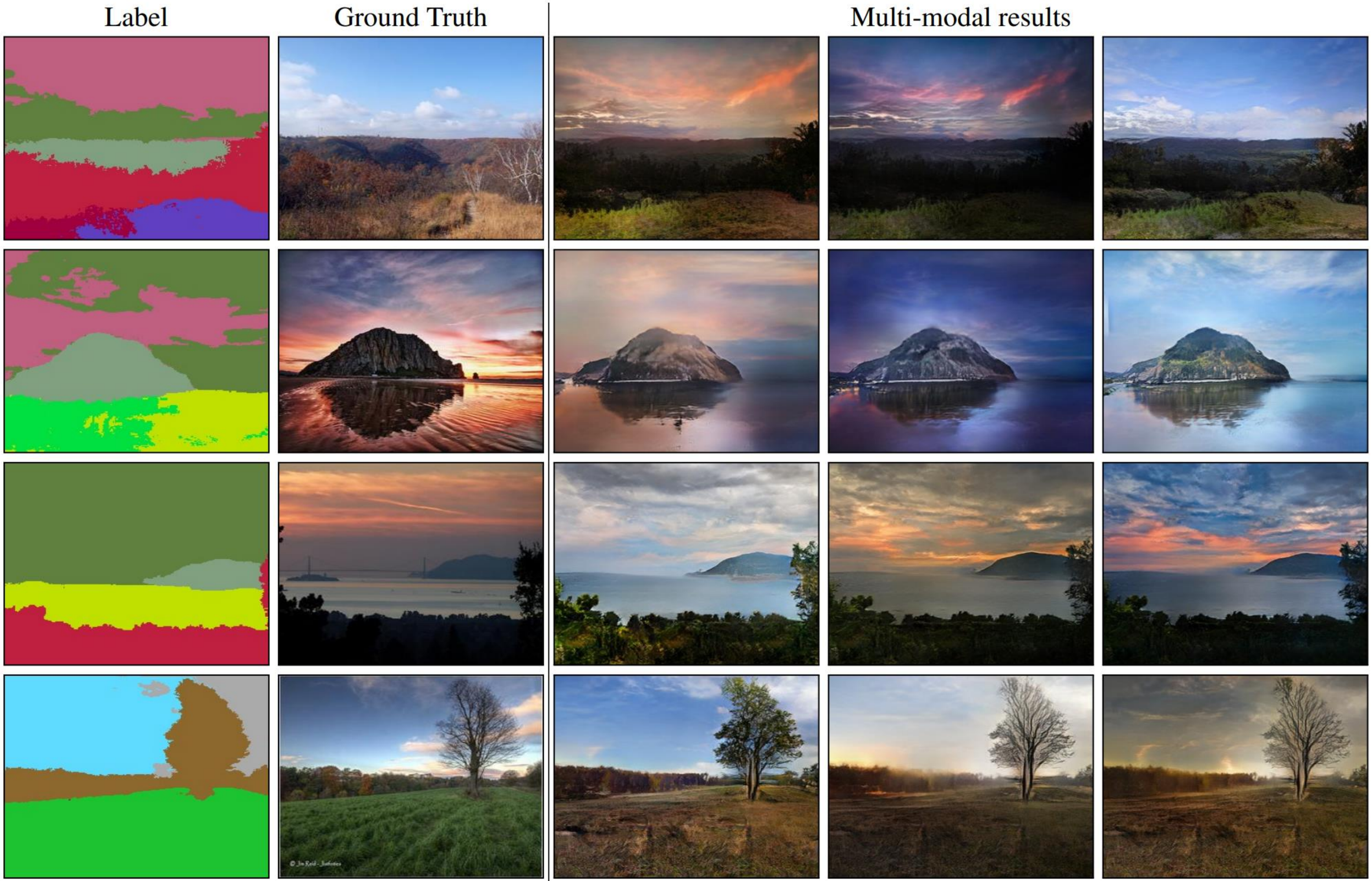


Style Control Learning via a Variational Learning Framework

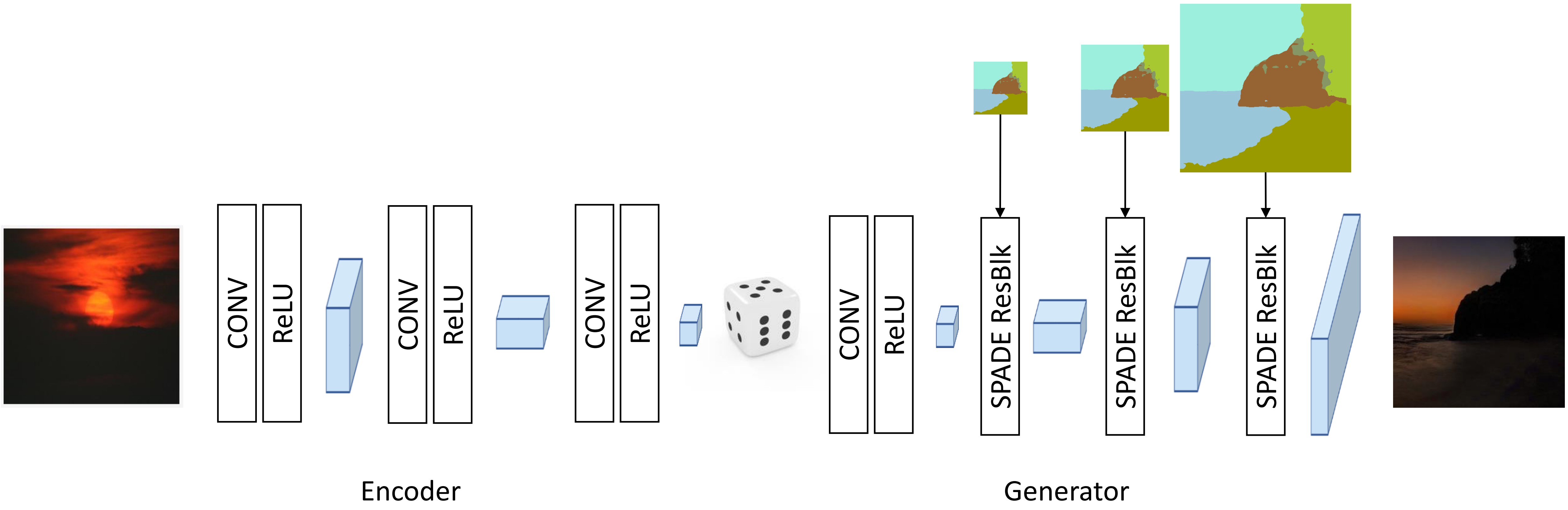


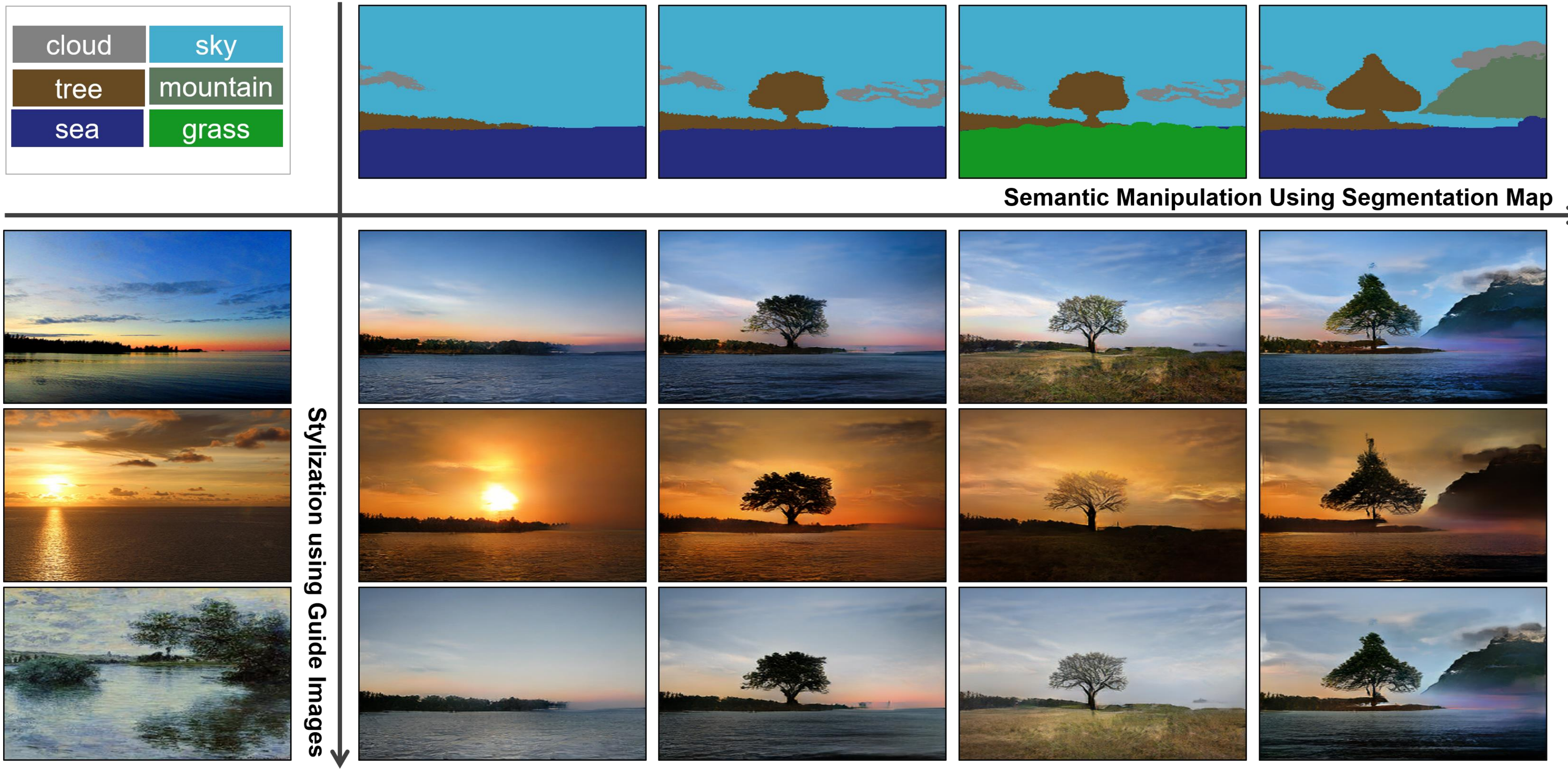
Style Control Learning via a Variational Learning Framework




Label
Ground Truth
Multi-modal results

In the test time, we can then use different style images to control the global color tone of the output image.

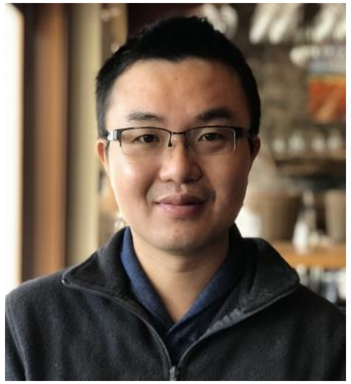



cloud
sky
tree
mountain
sea
grass
Semantic Manipulation Using Segmentation Map
Stylization using Guide Images

Conclusion

- Segmentation to Image Synthesis Task
- SPADE: Spatially Adaptive Denormalization
- Joint Style and Layout Control
- CVPR2019
- Online demo link: http://nvidia-research-mingyuliu.com/gaugan
- SPADE code: https://github.com/nvlabs/spade/
- Paper: https://arxiv.org/abs/1903.07291

Taesung Park
NVIDIA, UC Berkeley



Ming-Yu Liu
NVIDIA

Ting-Chun Wang
NVIDIA

Jun-Yan Zhu
NVIDIA, MIT