PAT 464/564 (Winter 2026)

Generative AI for Music & Audio Creation

Lecture 13: Generative Adversarial Nets

Instructor: Hao-Wen Dong

UNIVERSITY OF MICHIGAN

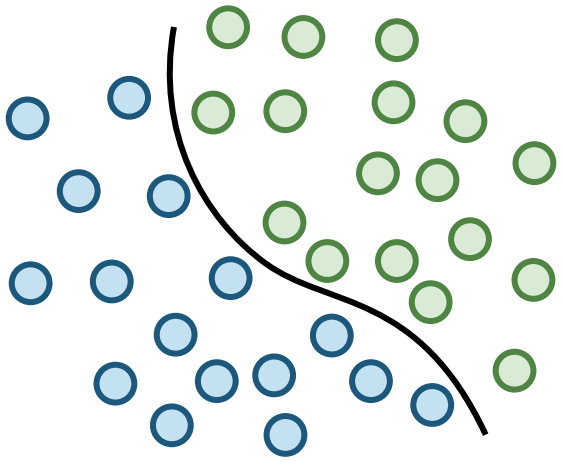
Representative Types of Deep Generative Models

- **Deep autoregressive models**
  - Recurrent neural network (RNN)
  - Long short-term memory (LSTM)
  - Transformer model
- **Deep latent variable models**
  - Variational autoencoder (VAE)
  - Generative adversarial network (GAN) **Today's topic!**
  - Diffusion model
  - Flow-based model
- *And many others…*

Deep Latent Variable Models

Discriminative vs Generative Models

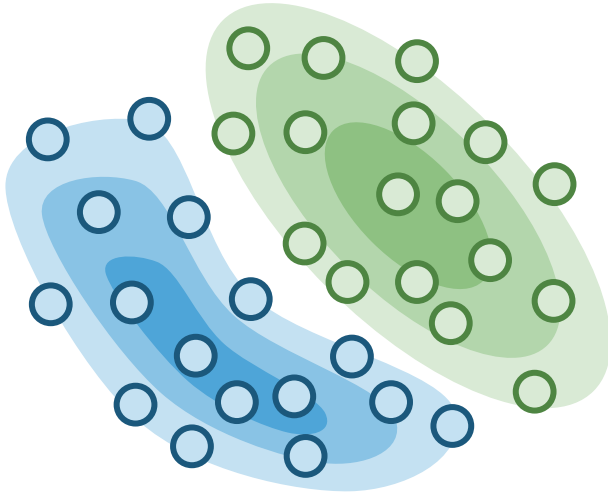
Discriminative



Discriminative models learn the decision boundary

$P(y|x)$

Generative

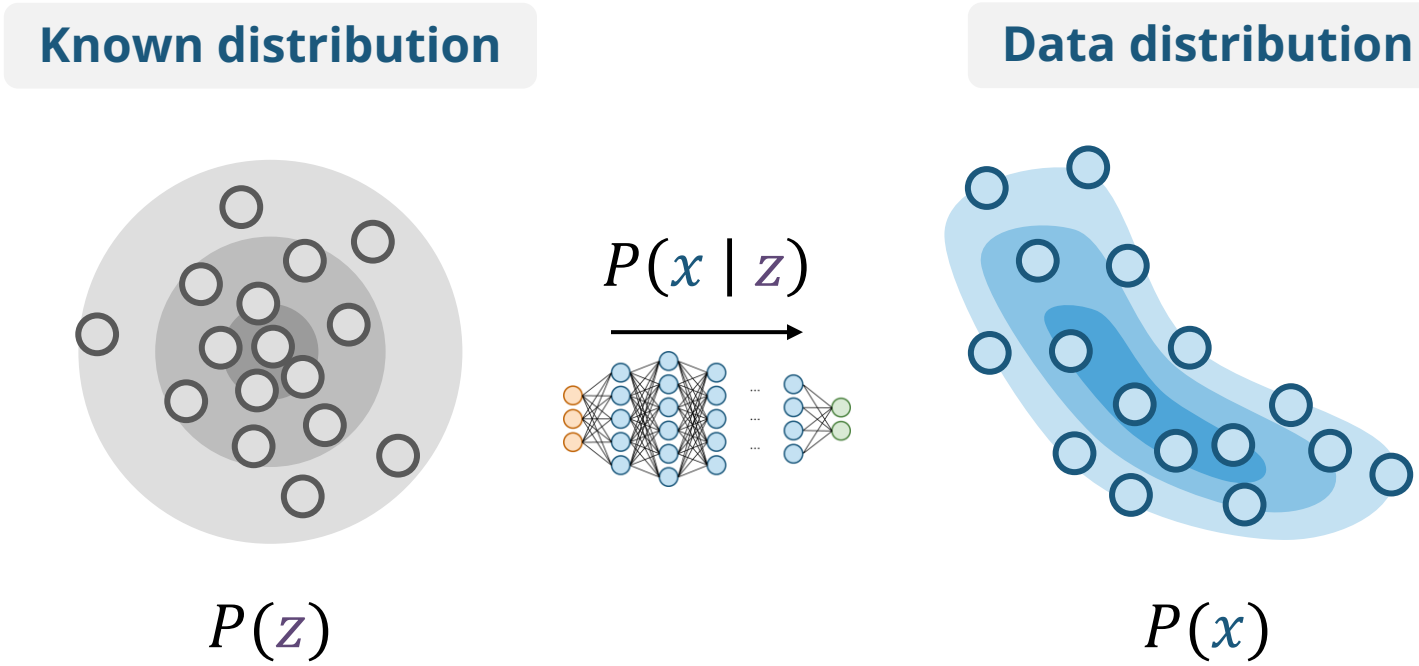


Generative models learn the underlying distribution

$P(x)$ or $P(x|y)$

Deep Latent Variable Models

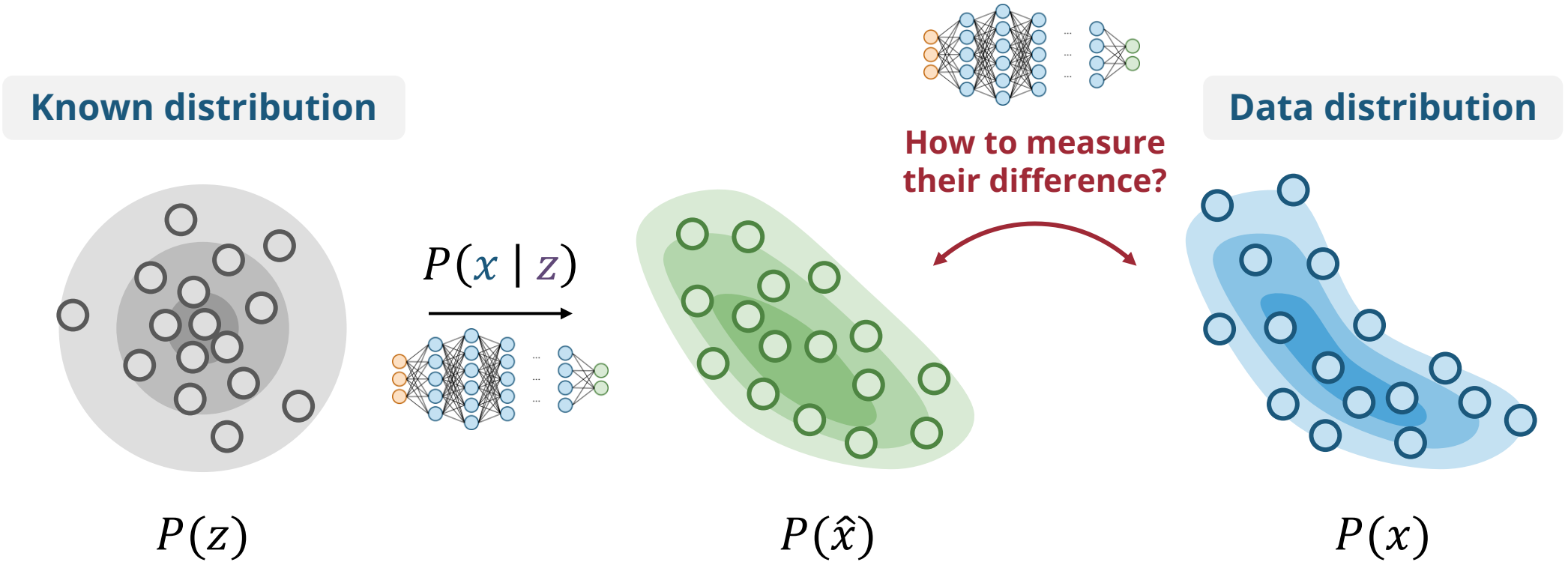
- **Intuition**: Learn to map a known distribution to the data distribution



$$P(x) = P(z)\ P(x \mid z)$$

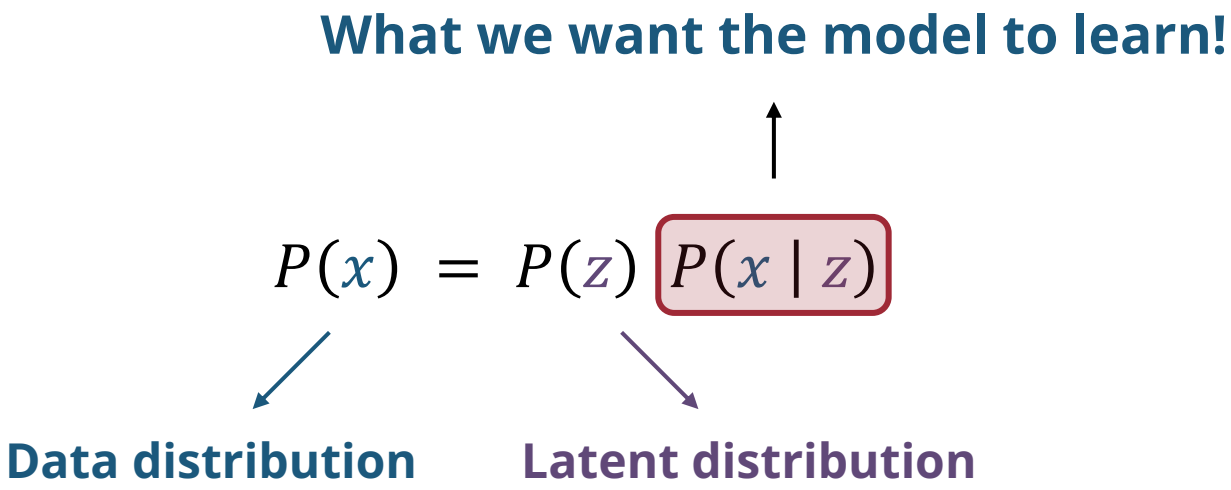
Deep Latent Variable Models

- **Intuition**: Learn to map a known distribution to the data distribution



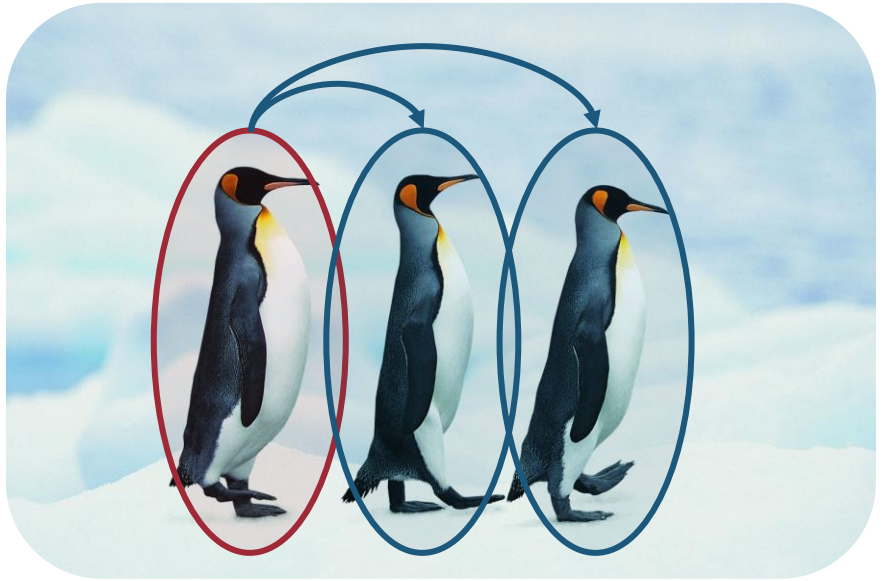
Deep Latent Variable Models

- **Intuition**: Learn to map a known distribution to the data distribution

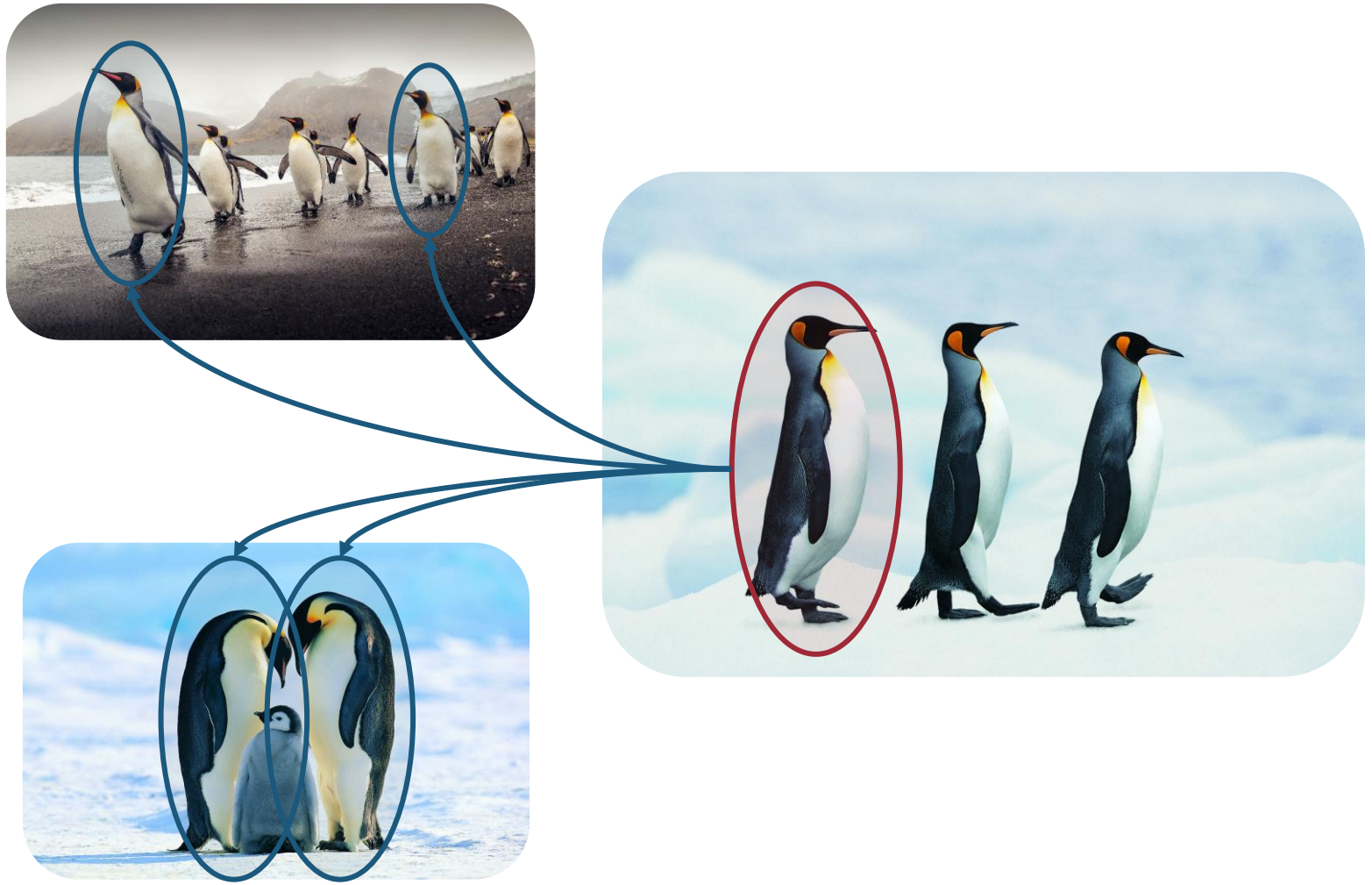


Convolutional Neural Networks (CNNs)

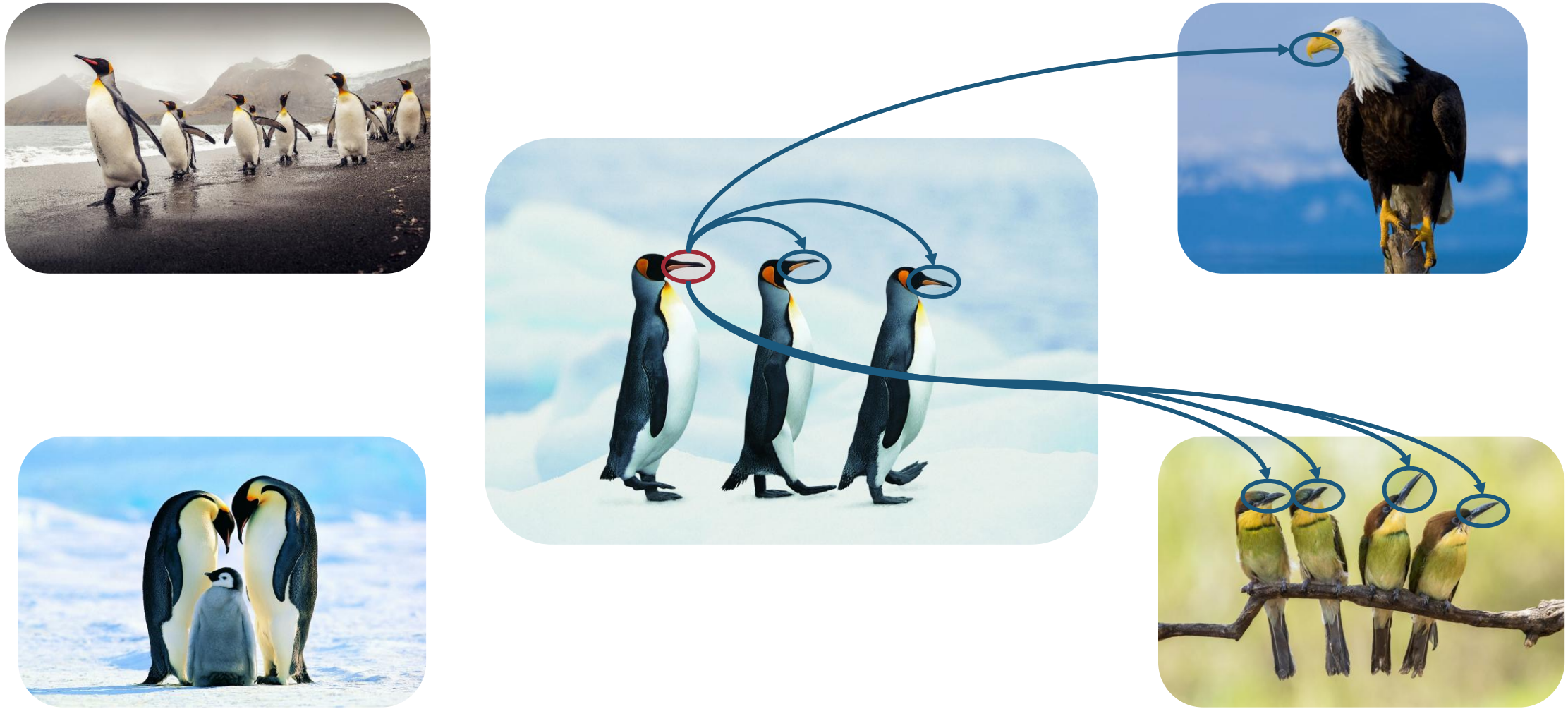
Reusable Pattern Detectors



Reusable Pattern Detectors

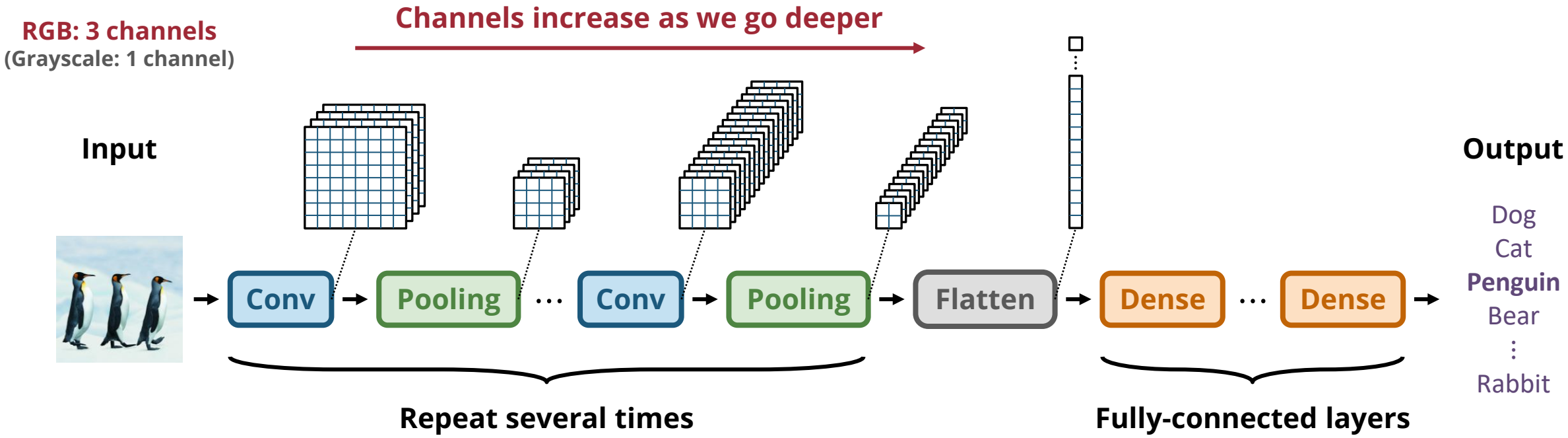


Reusable Pattern Detectors

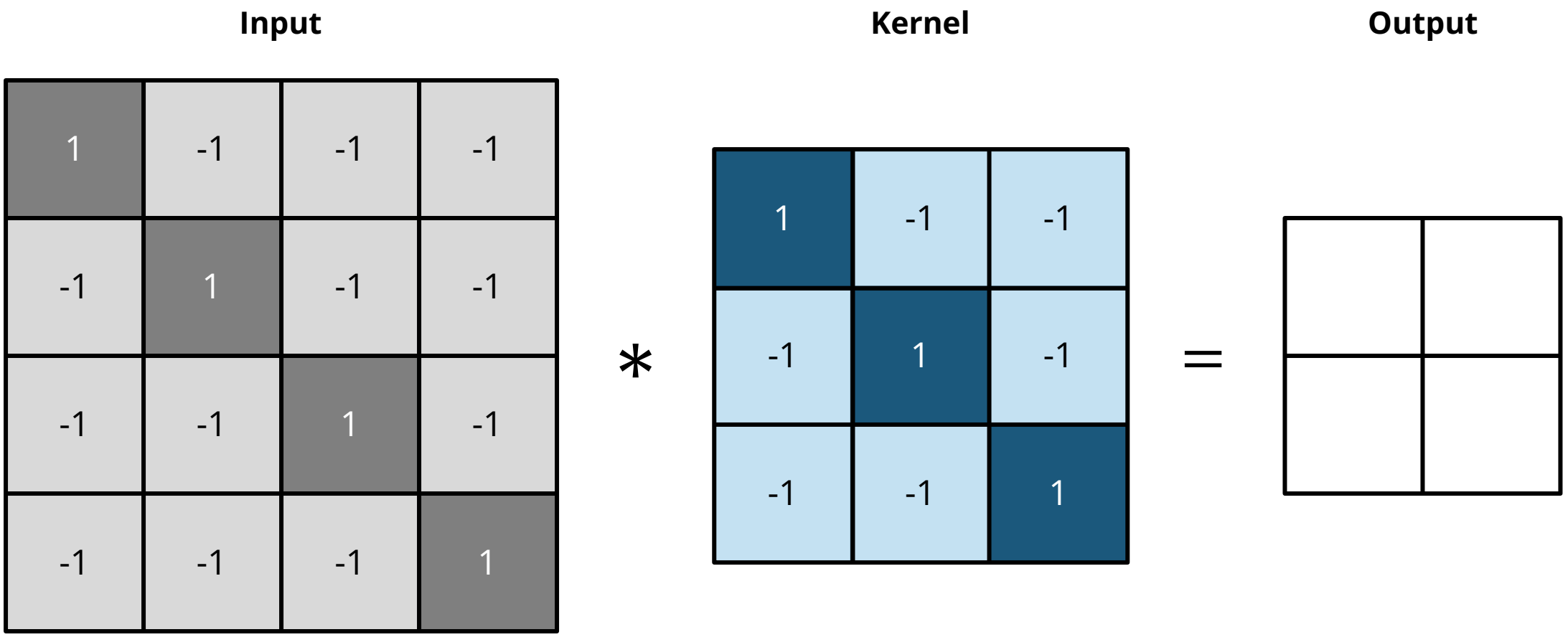


Reusable Pattern Detectors

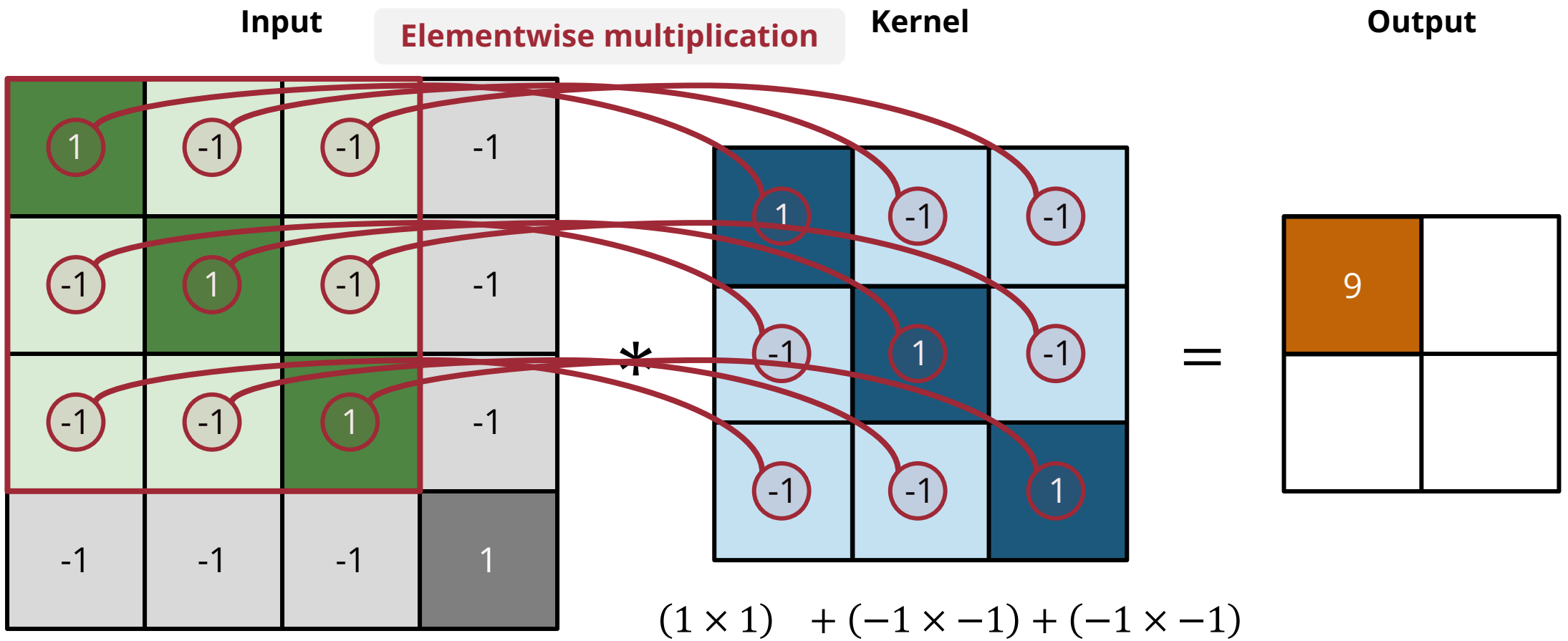
Convolutional Neural Network (CNNs)



2D Convolution

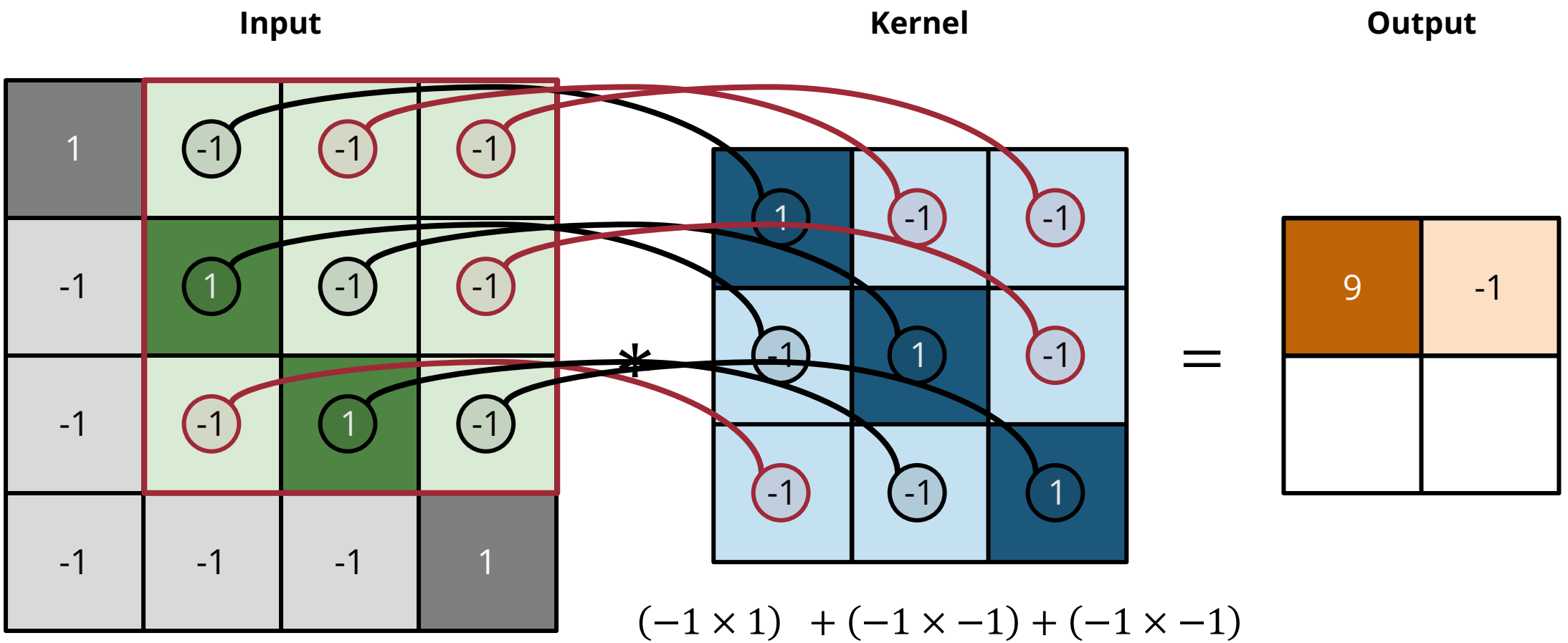


2D Convolution



$$(1 \times 1) + (-1 \times -1) + (-1 \times -1)$$
$$+(-1 \times -1) + (1 \times 1) + (-1 \times -1) = \mathbf{9}$$
$$+(-1 \times -1) + (-1 \times -1) + (1 \times 1)$$

2D Convolution



2D Convolution

Input

1	-1	-1	-1
-1	1	-1	-1
-1	-1	1	-1
-1	-1	-1	1

$*$

Kernel

1	-1	-1
-1	1	-1
-1	-1	1

$=$

Output

9	-1
-1	

$$(-1 \times 1) + (1 \times -1) + (-1 \times -1)$$
$$+(-1 \times -1) + (-1 \times 1) + (1 \times -1) = -1$$
$$+(-1 \times -1) + (-1 \times -1) + (-1 \times 1)$$

2D Convolution

Input

1	-1	-1	-1
-1	1	-1	-1
-1	-1	1	-1
-1	-1	-1	1

$*$

Kernel

1	-1	-1
-1	1	-1
-1	-1	1

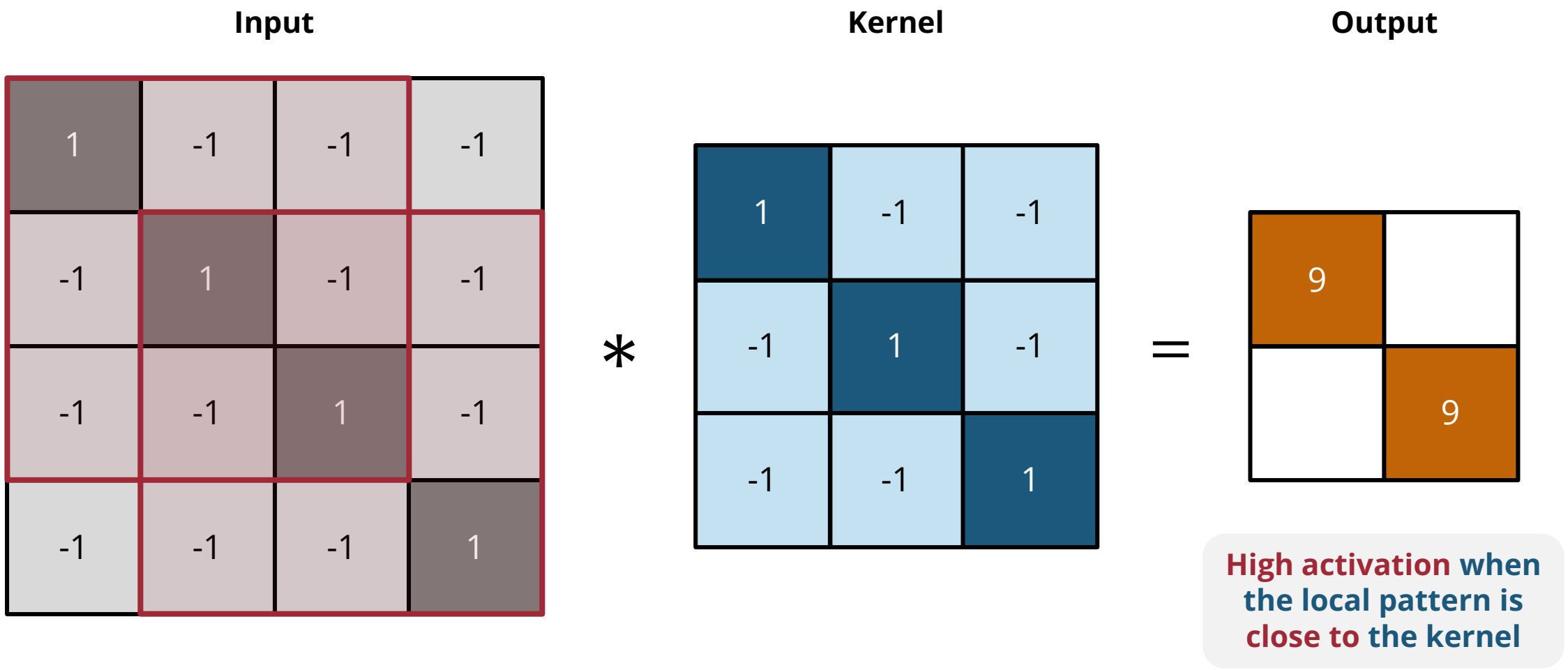
$=$

Output

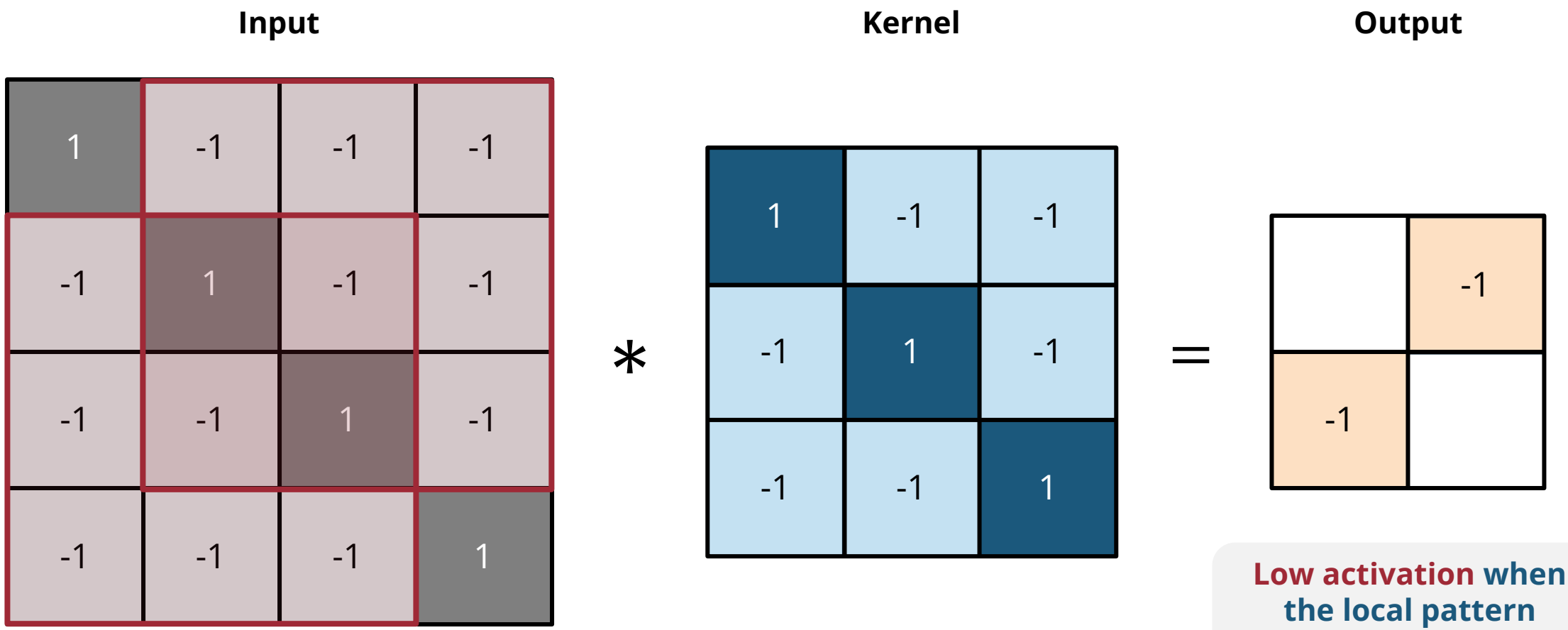
9	-1
-1	9

$$(1 \times 1) + (-1 \times -1) + (-1 \times -1)$$
$$+(-1 \times -1) + (1 \times 1) + (-1 \times -1) = \mathbf{9}$$
$$+(-1 \times -1) + (-1 \times -1) + (1 \times 1)$$

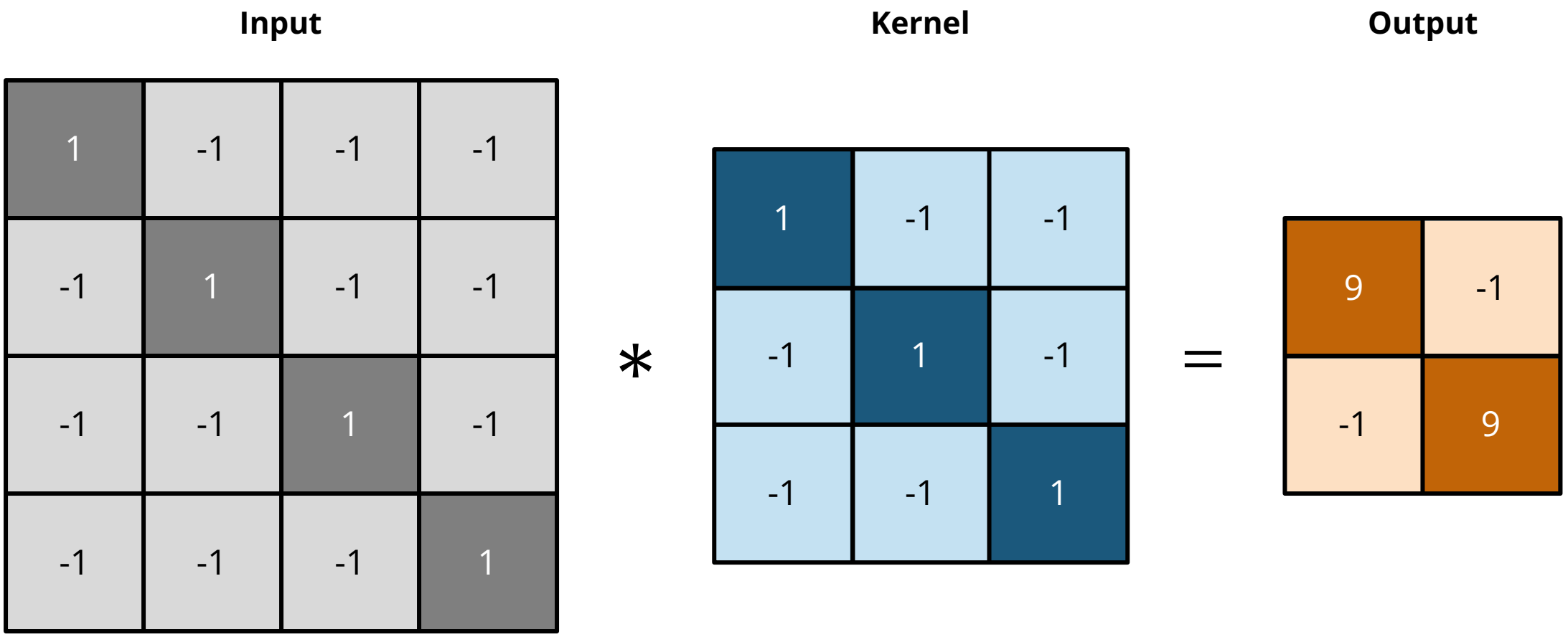
2D Convolution



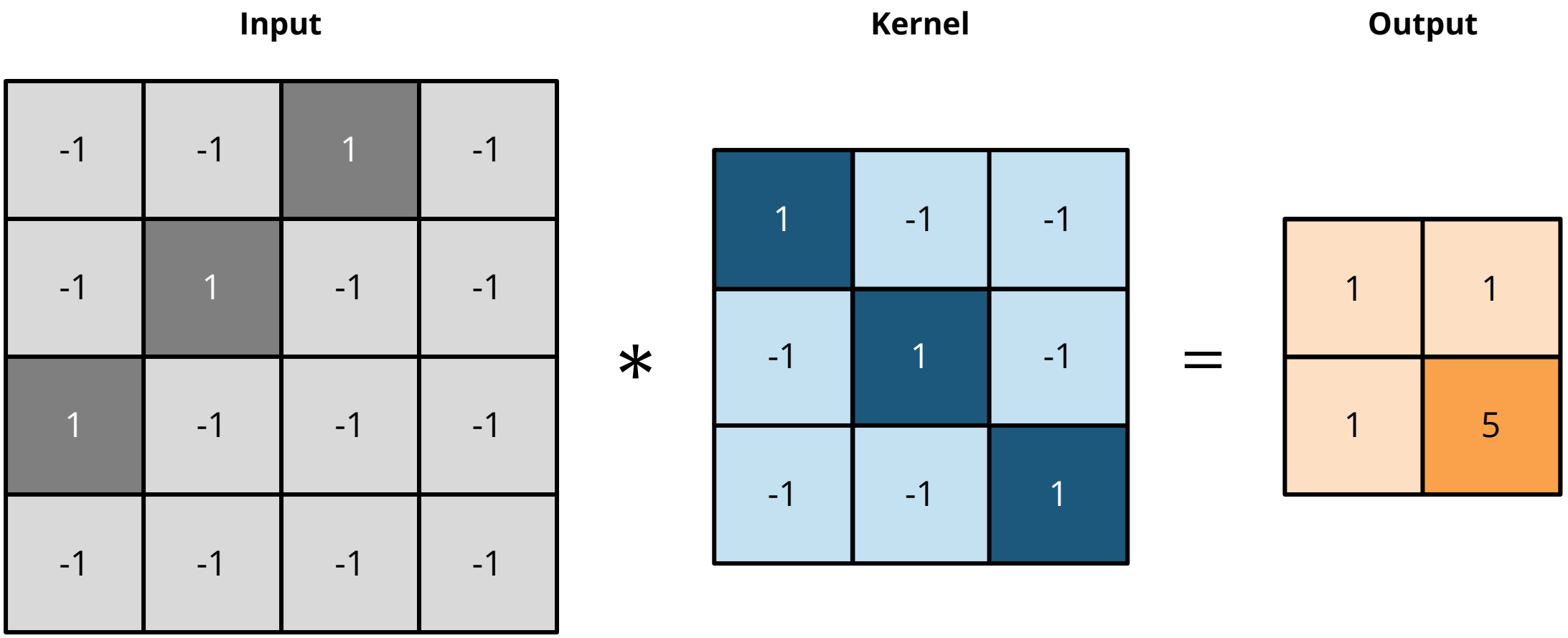
2D Convolution



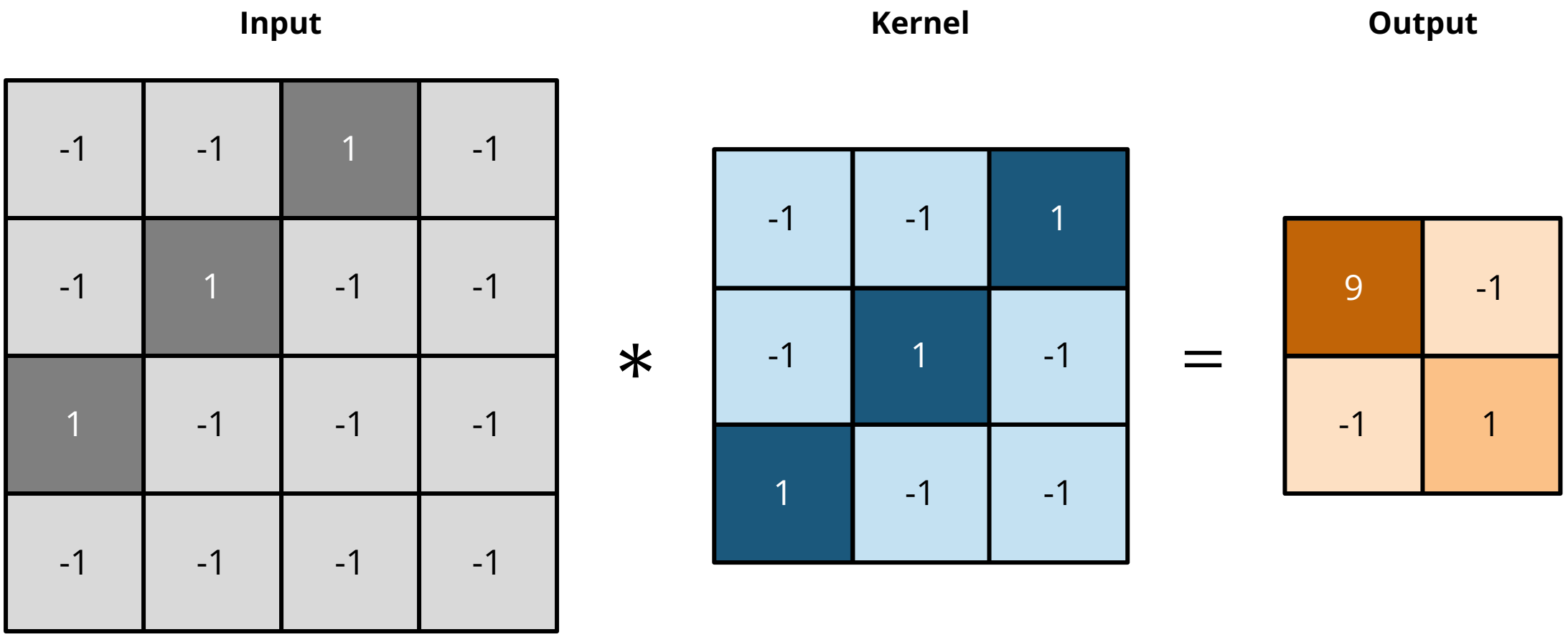
2D Convolution



2D Convolution

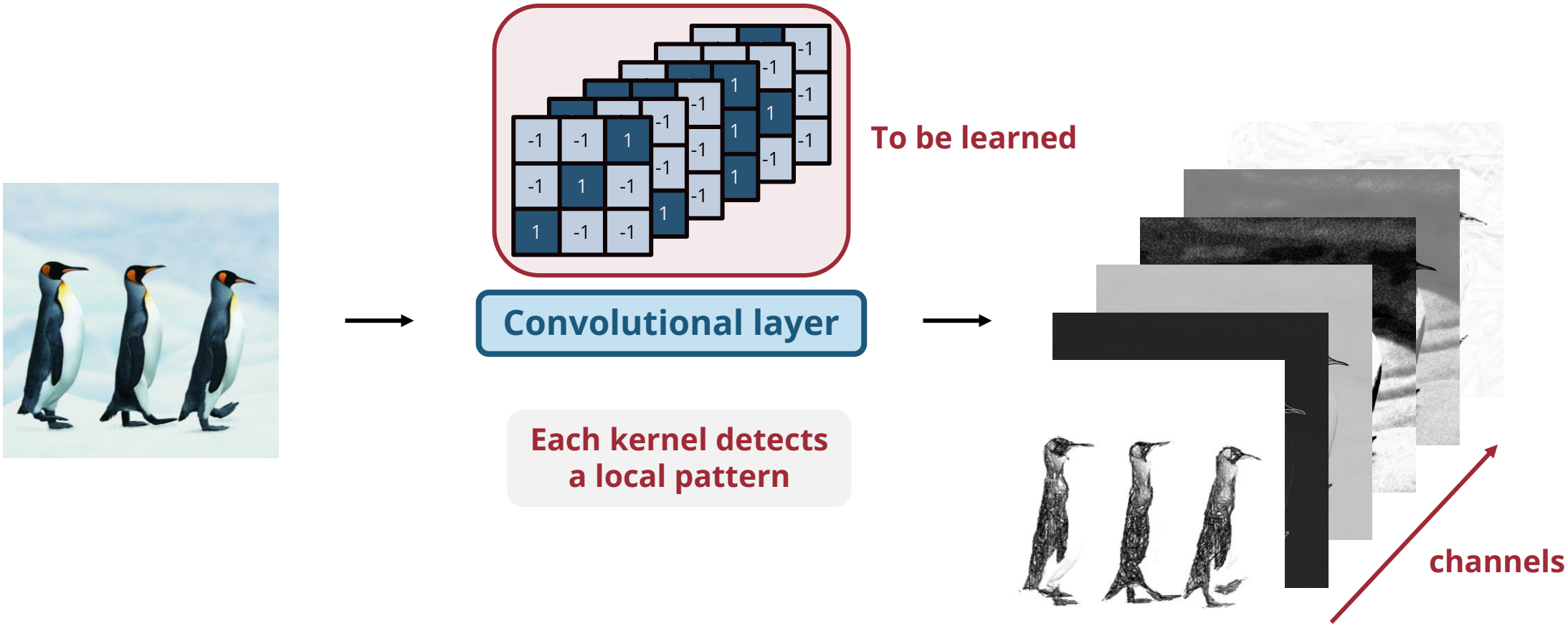


2D Convolution

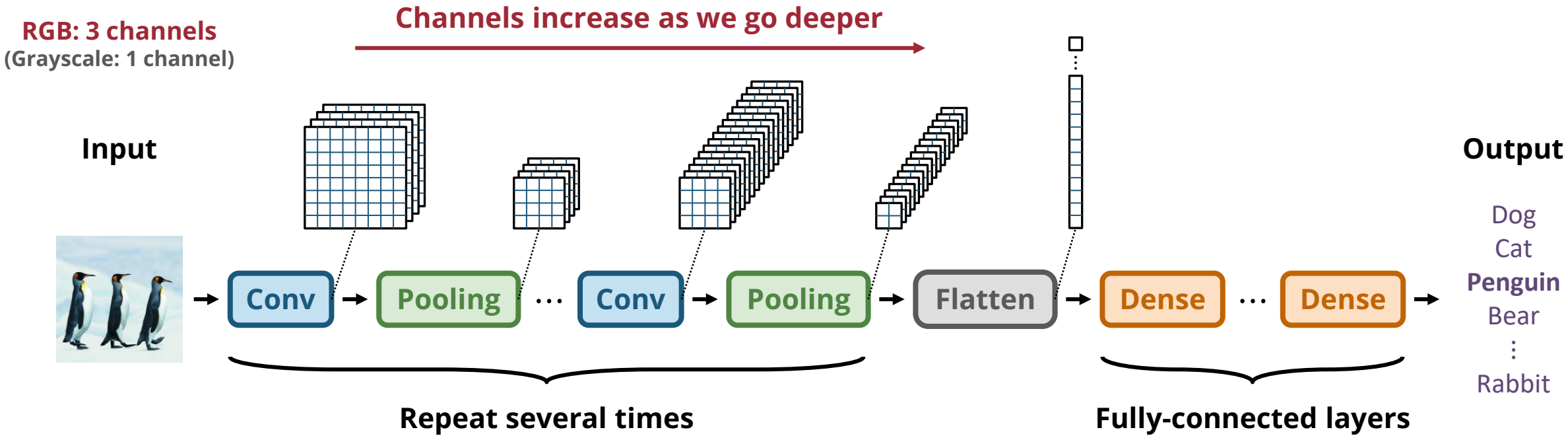


Convolutional Layer

- A convolutional layer consists of many **learnable kernels** (channels)



Convolutional Neural Network (CNNs)



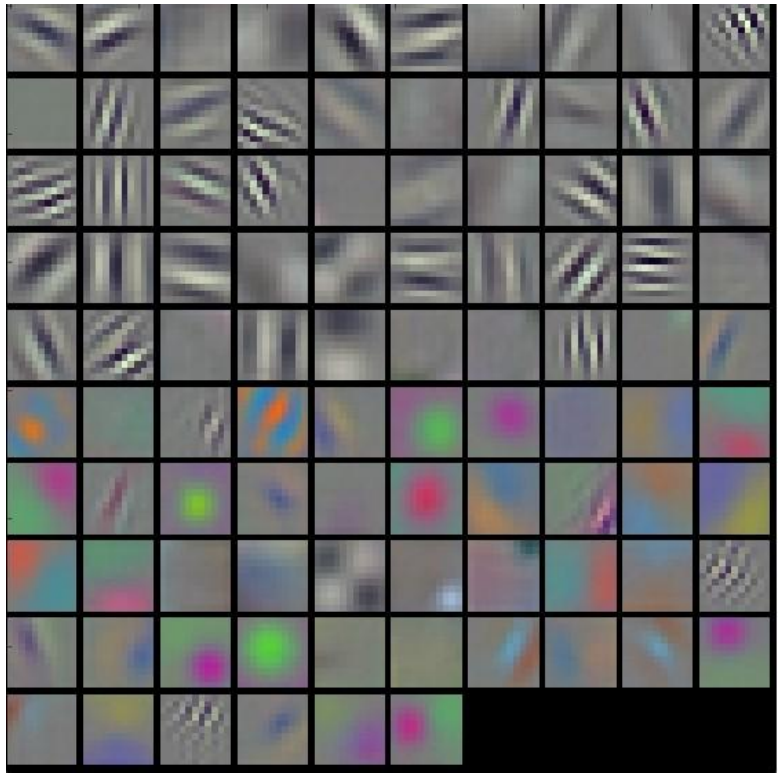
What does a CNN Learn?

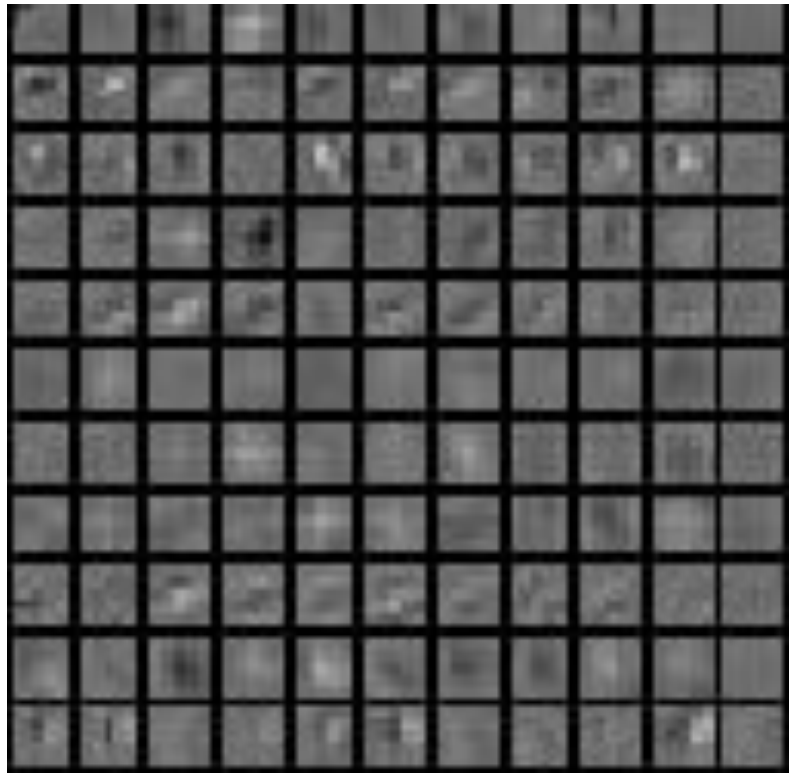
Learned CNN Kernels in a Trained AlexNet

1st convolutional layer

2nd convolutional layer

11x11 kernels



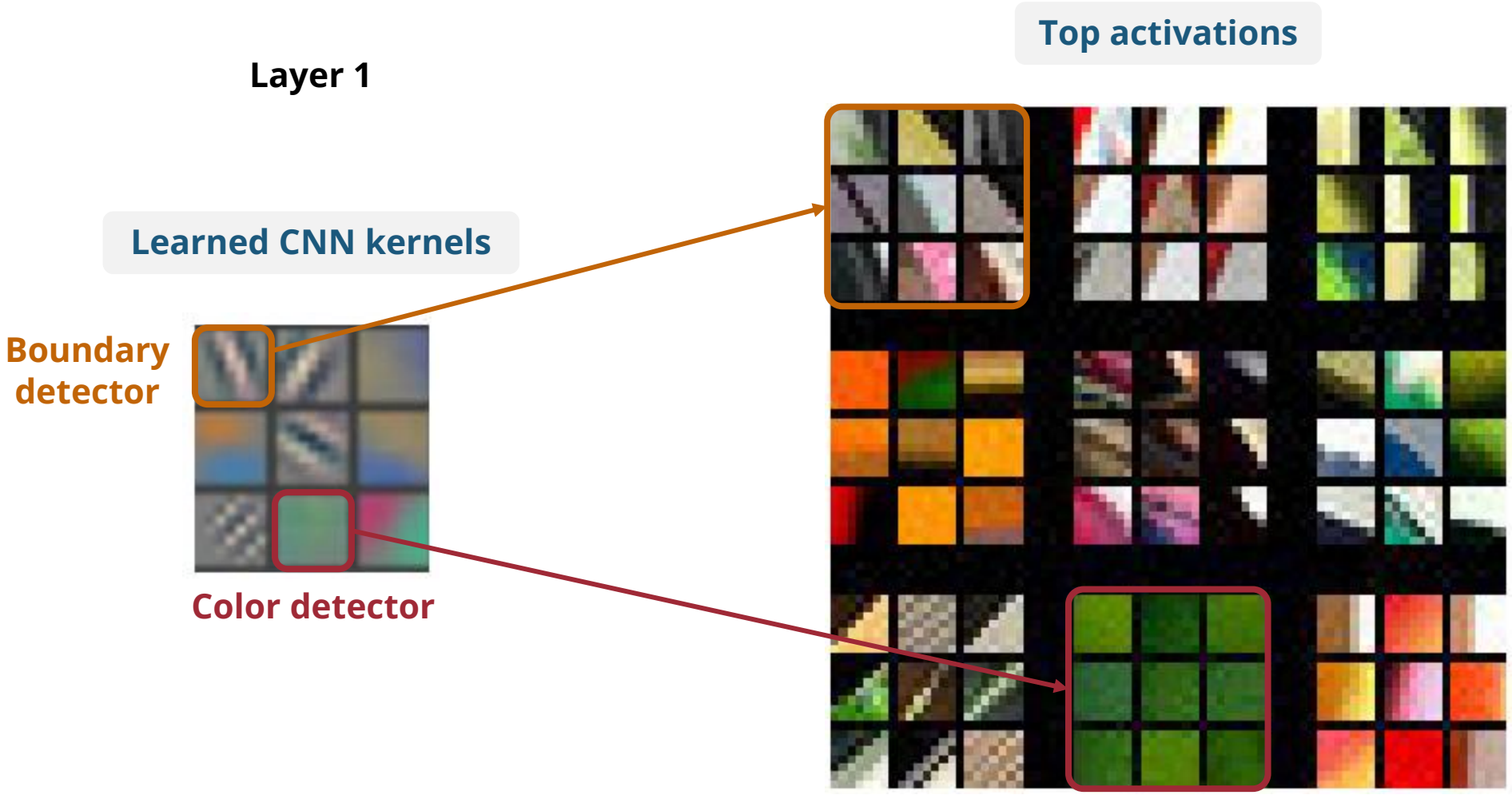


5x5 kernels

(Source: cs231n.github.io)

cs231n.github.io/understanding-cnn

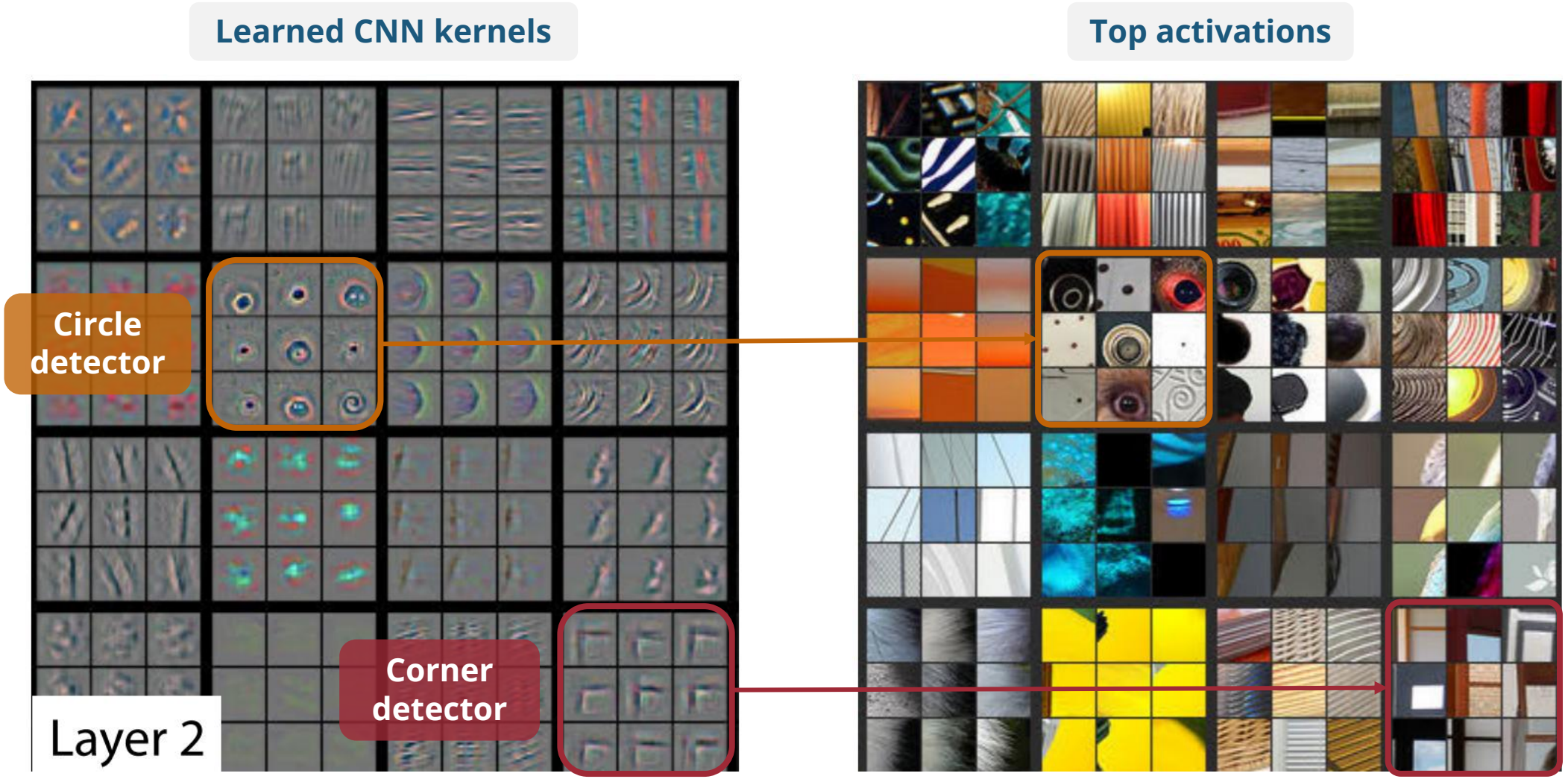
Learned CNN Kernels in a Trained AlexNet



(Source: Zeiler et al., 2014)

Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

Learned CNN Kernels in a Trained AlexNet



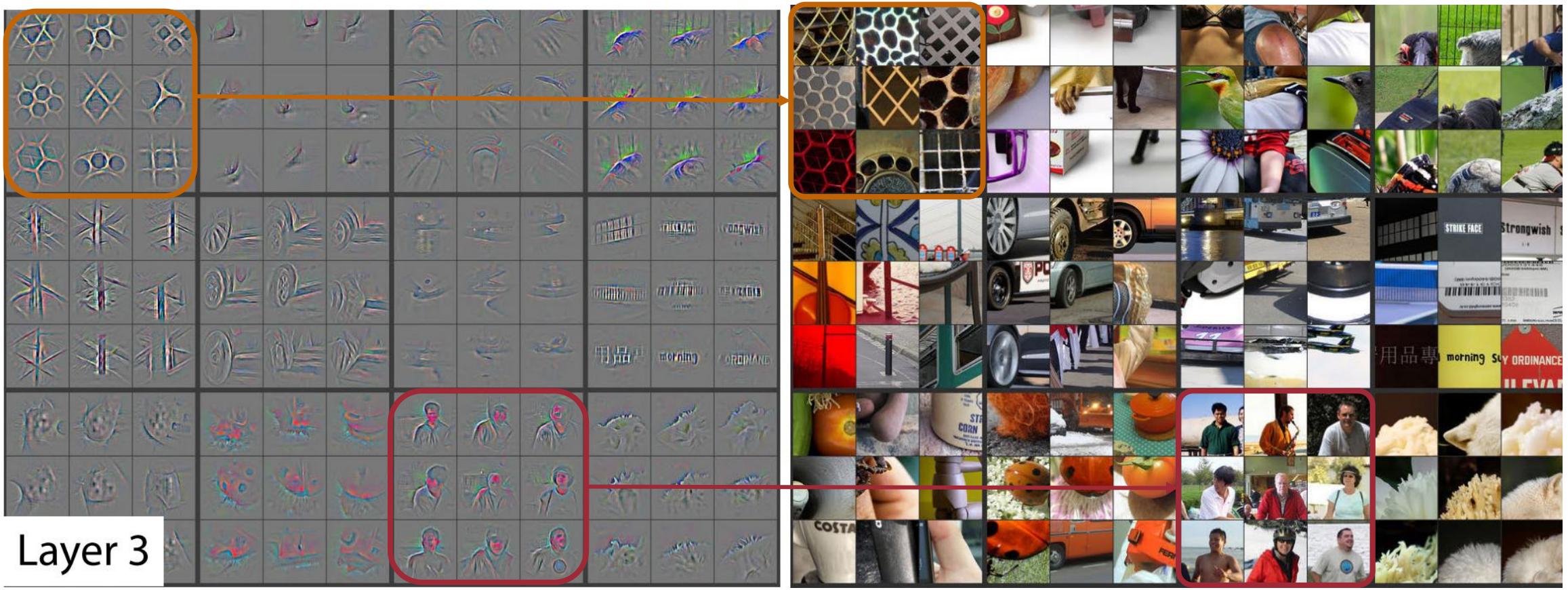
(Source: Zeiler et al., 2014)

Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

Learned CNN Kernels in a Trained AlexNet

Learned CNN kernels | **Top activations**

Grid detector



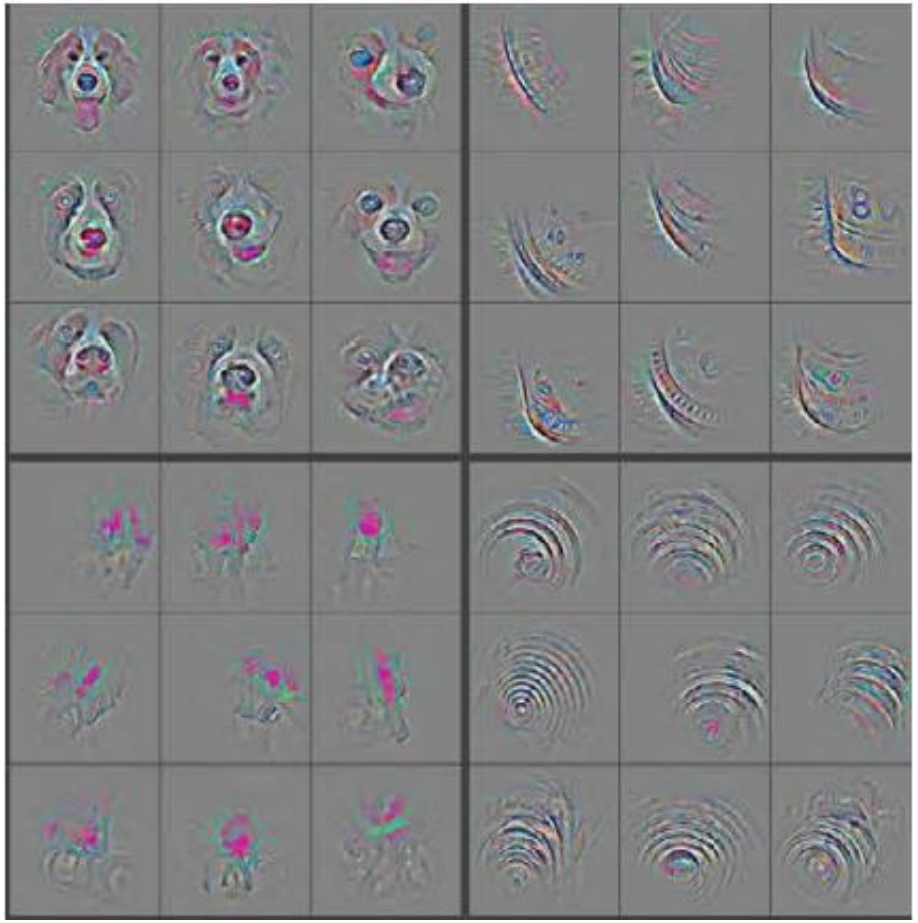
Human detector

(Source: Zeiler et al., 2014)

Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

Learned CNN Kernels in a Trained AlexNet

Learned CNN kernels



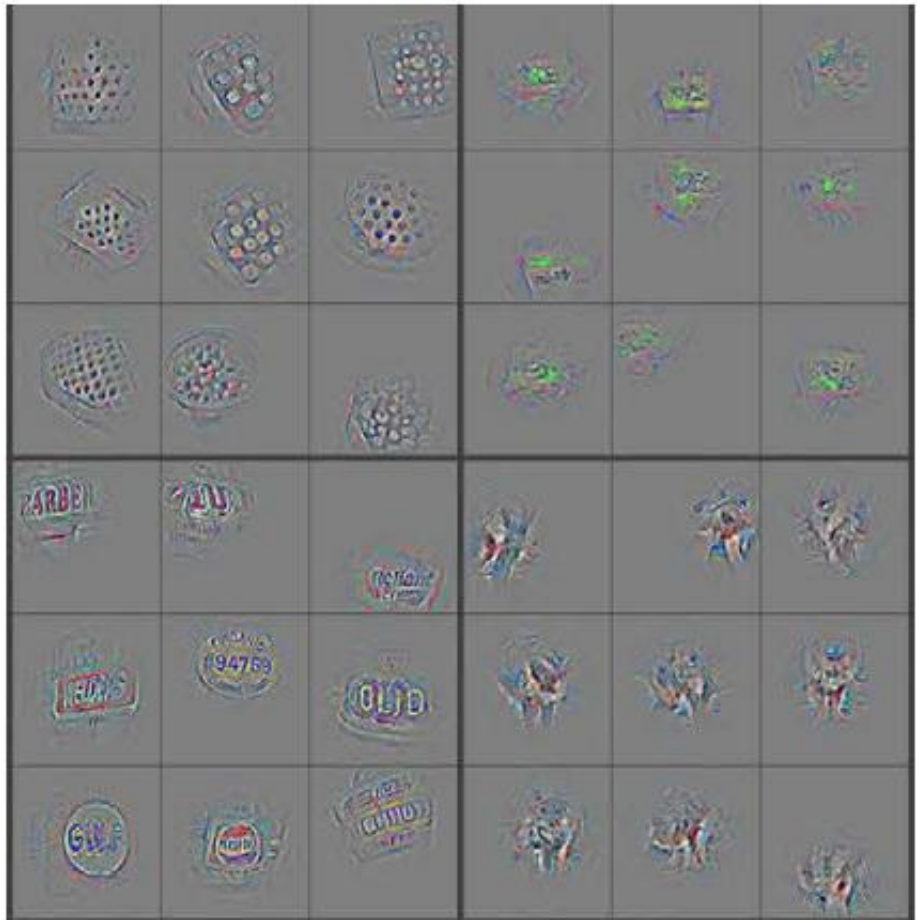
Top activations

(Source: Zeiler et al., 2014)

Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

Learned CNN Kernels in a Trained AlexNet

Learned CNN kernels



Top activations

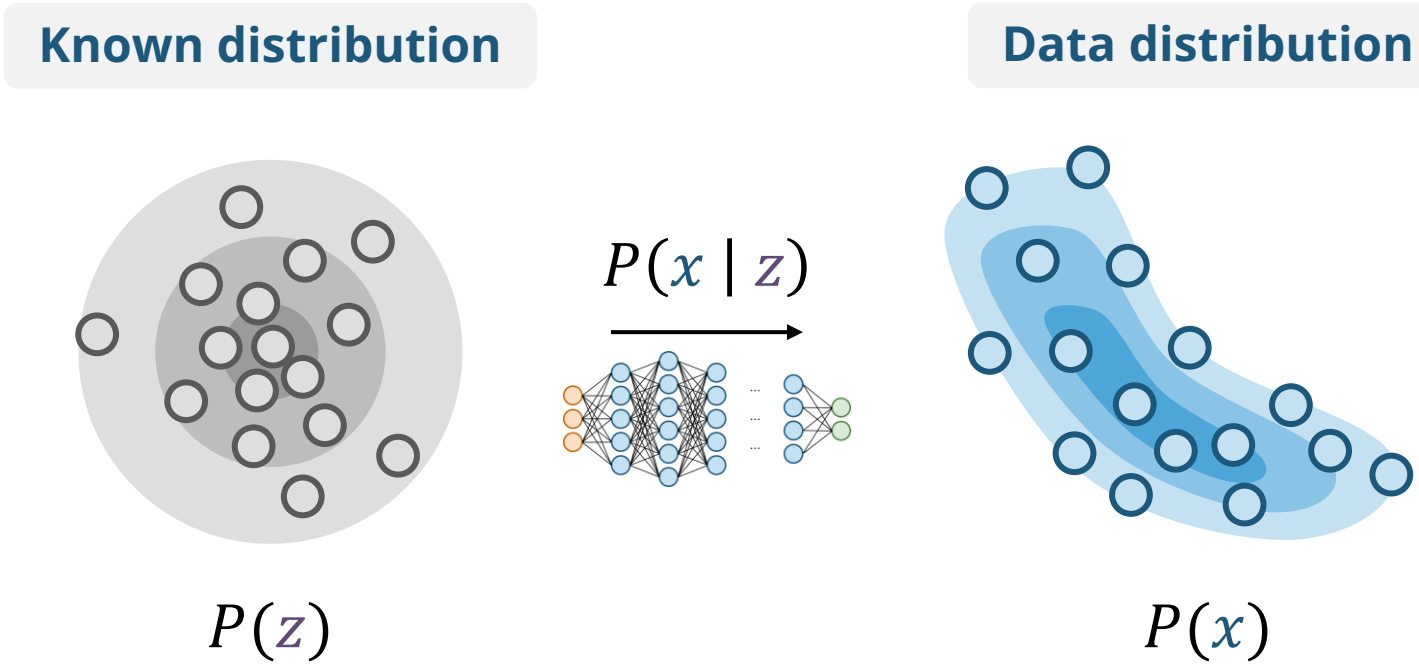
(Source: Zeiler et al., 2014)

Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

Generative Adversarial Net (GAN)

Deep Latent Variable Models

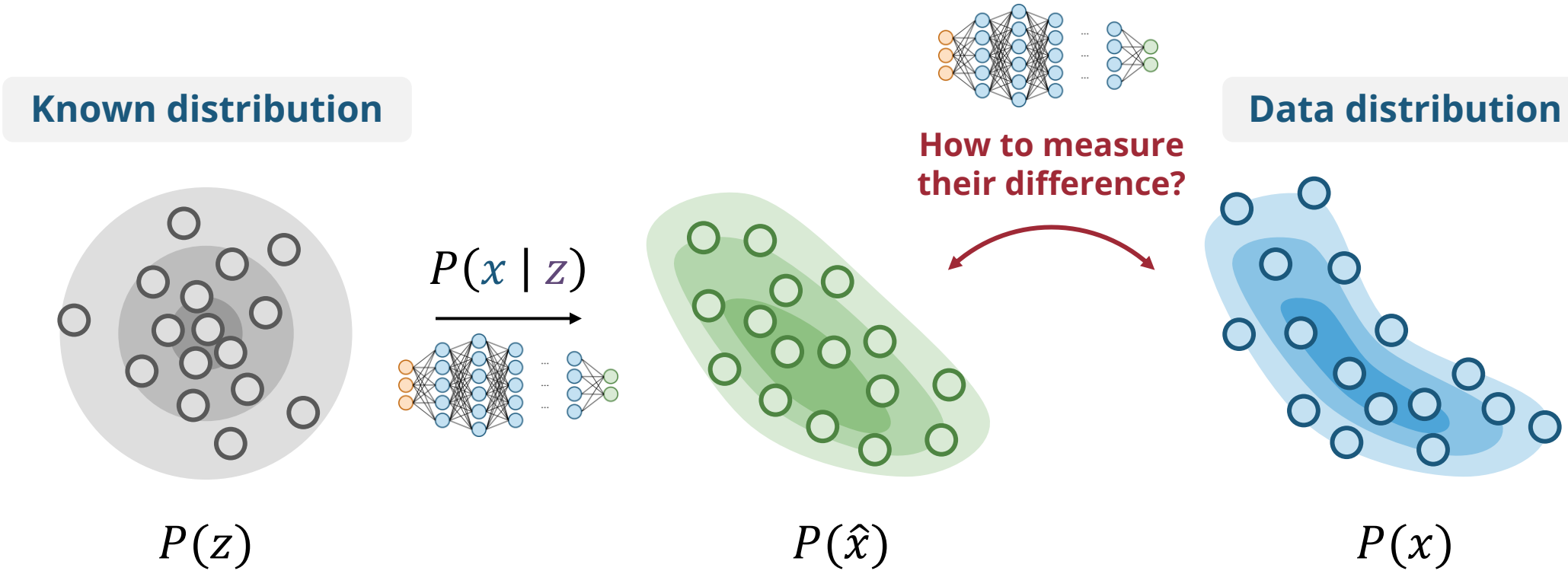
- **Intuition**: Learn to map a known distribution to the data distribution



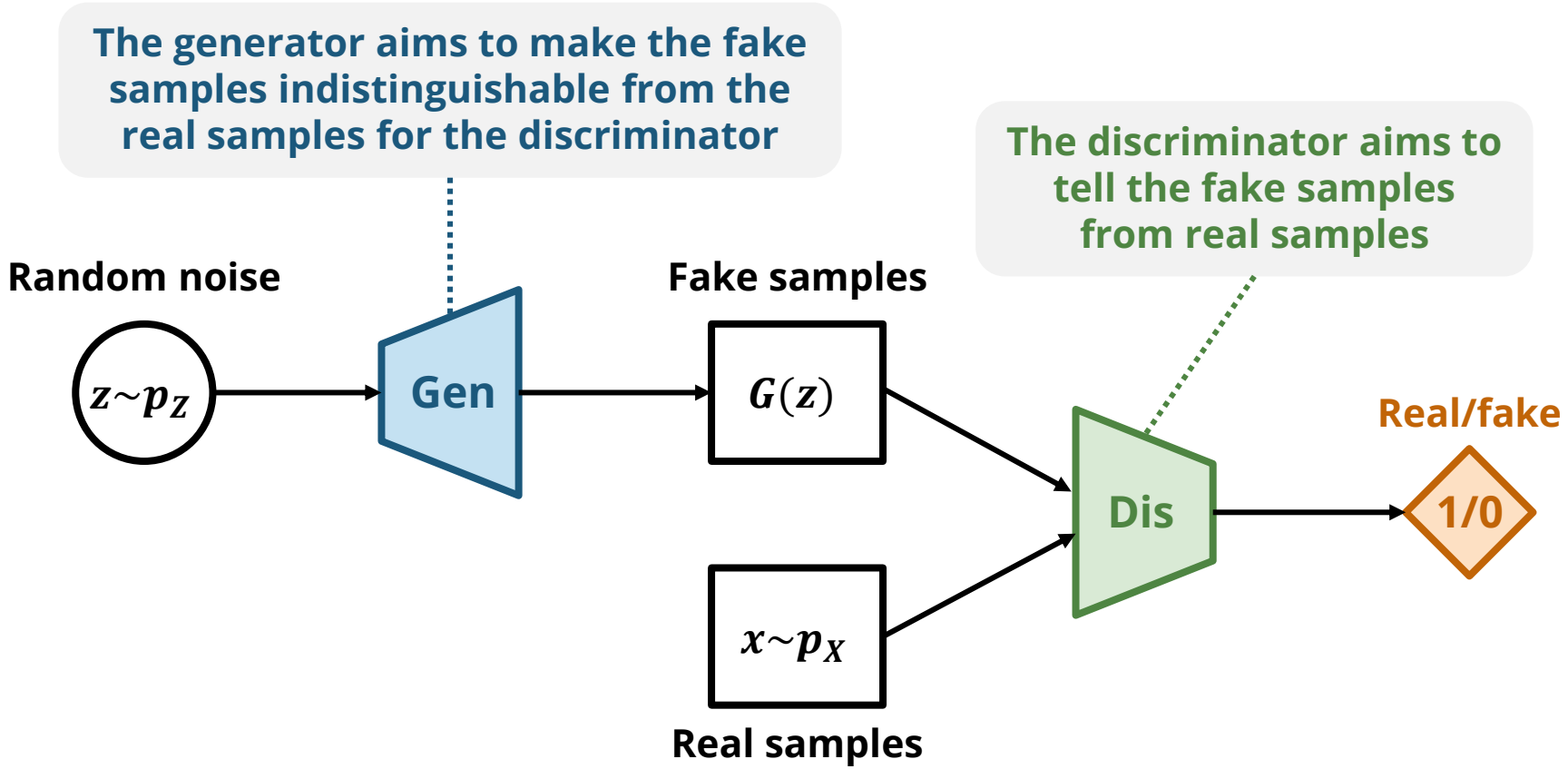
$$P(x) = P(z)\ P(x \mid z)$$

Deep Latent Variable Models

- **Intuition**: Learn to map a known distribution to the data distribution

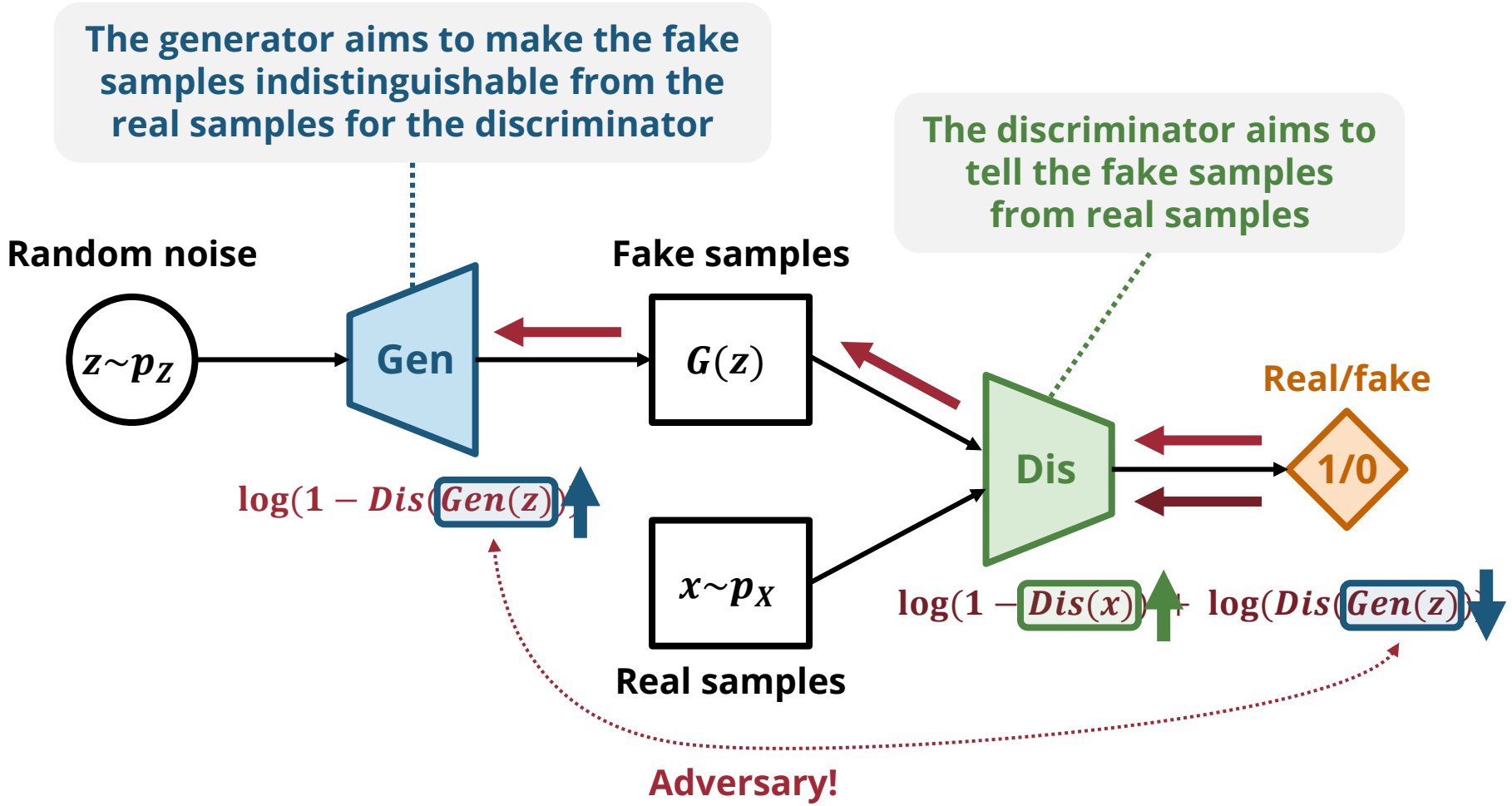


Generative Adversarial Nets (GANs) (Goodfellow et al., 2014)



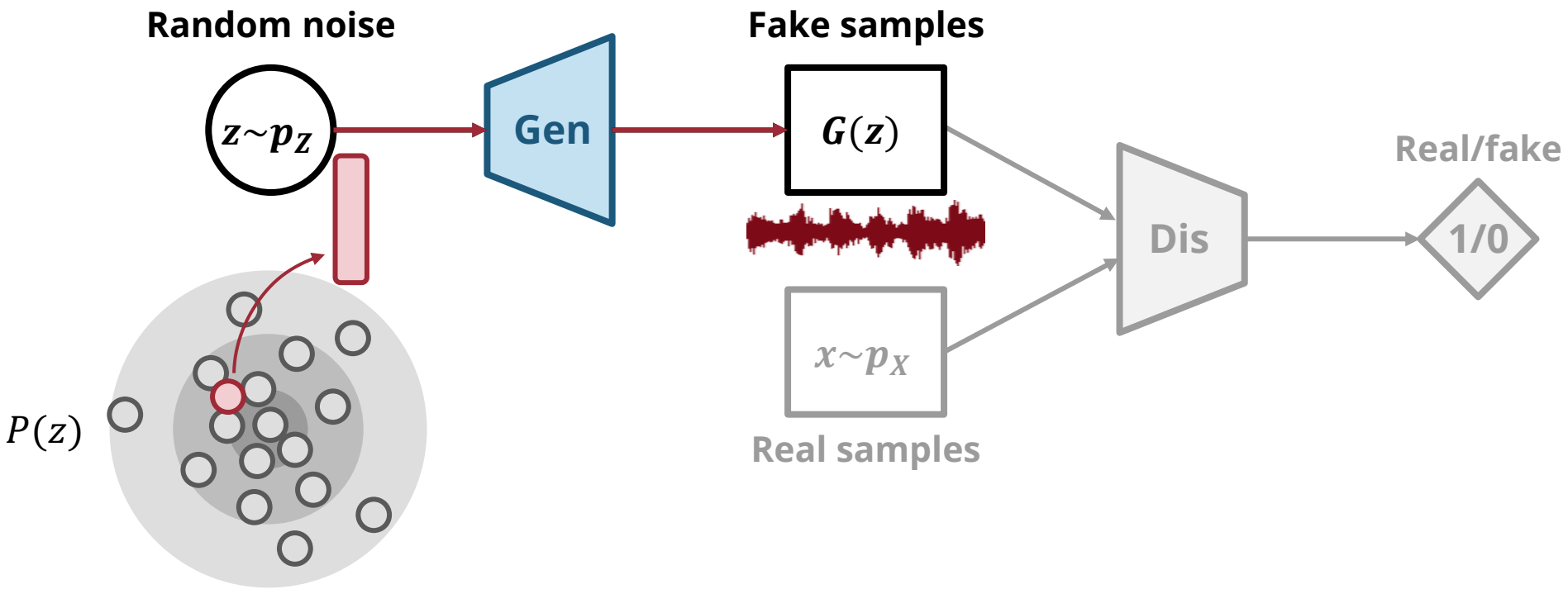
Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Networks," *NeurIPS*, 2014.

Generative Adversarial Nets (GANs): Training



Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Networks," *NeurIPS*, 2014.

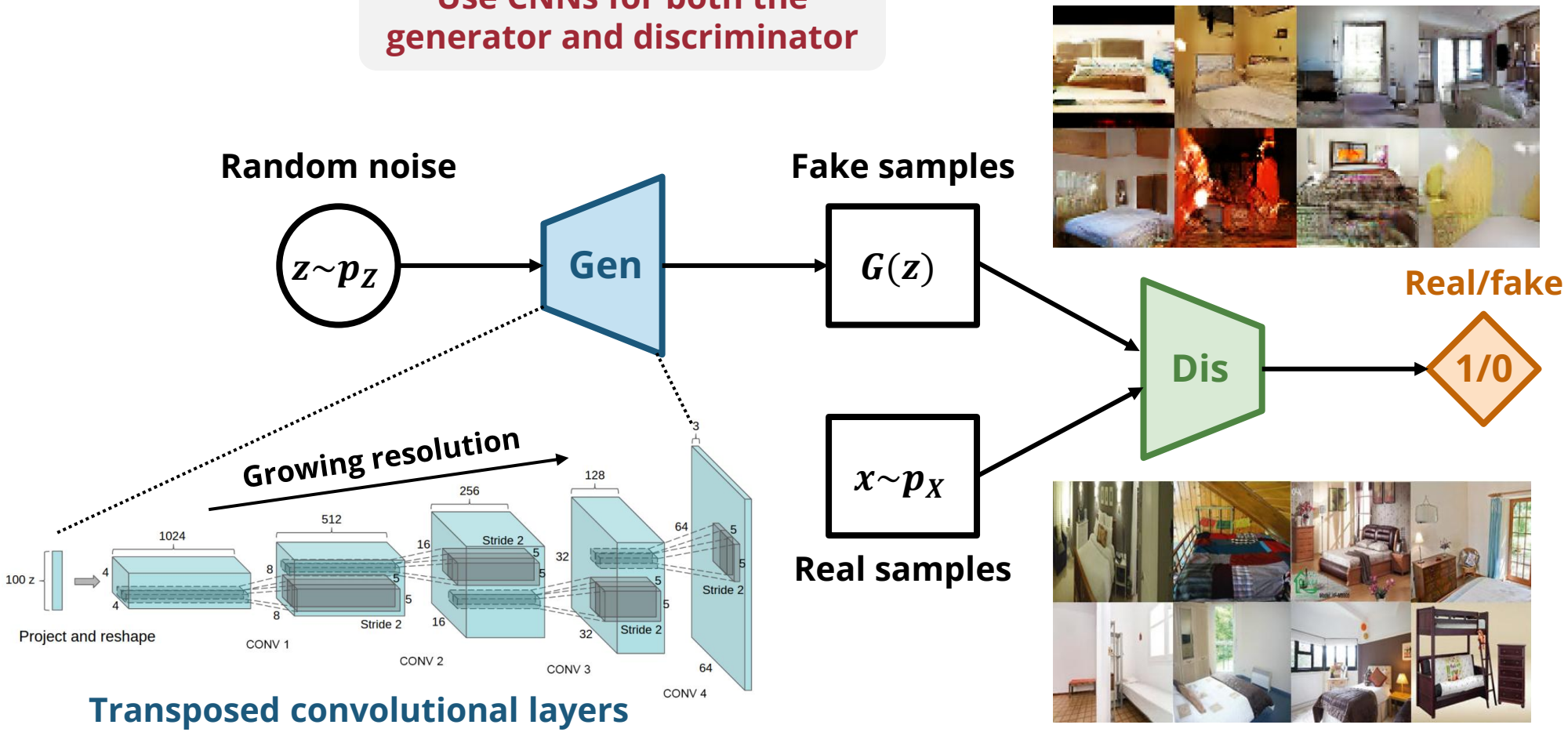
Generative Adversarial Nets (GANs): Generation



Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Networks," *NeurIPS*, 2014.

Deep Convolutional GANs (DCGANs) (Radford et al., 2014)

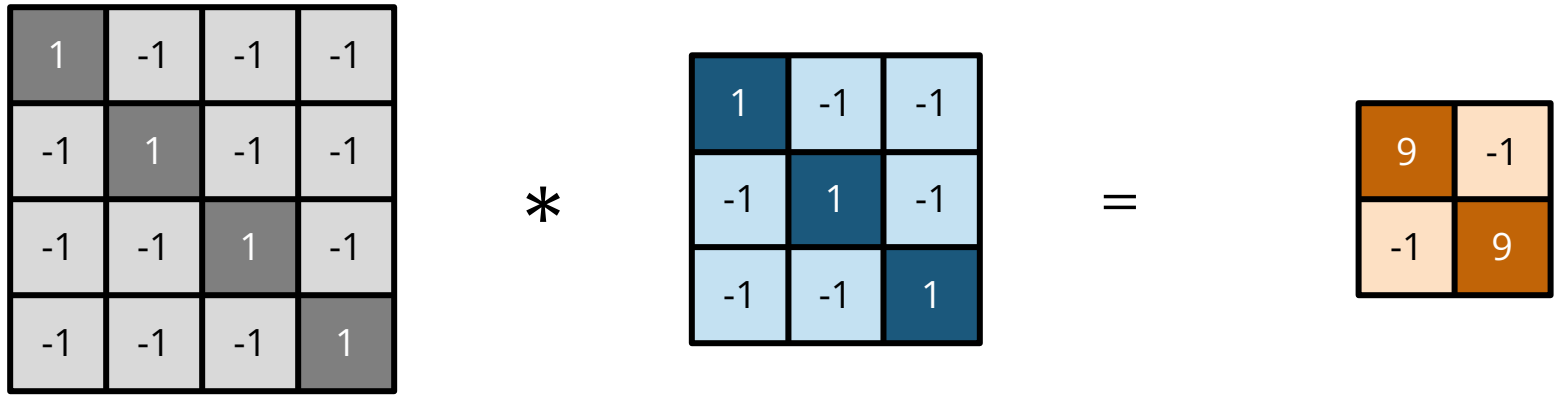
Use CNNs for both the generator and discriminator



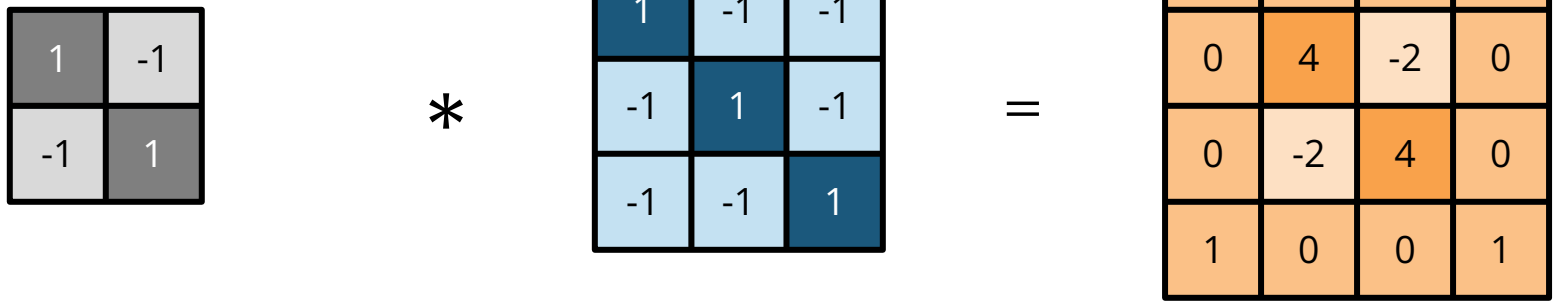
Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *ICLR*, 2016.

Transposed Convolution

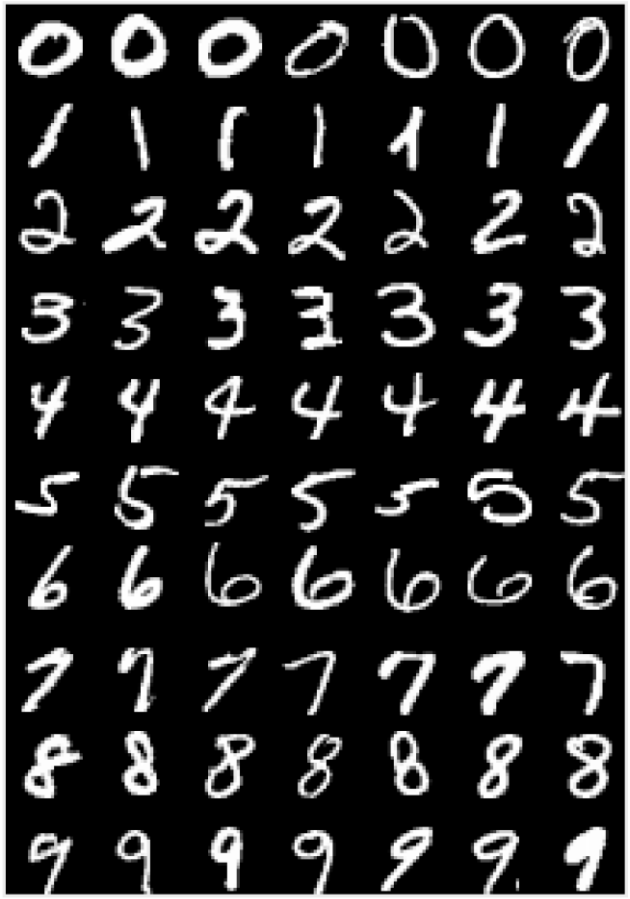
Convolution



Transposed convolution

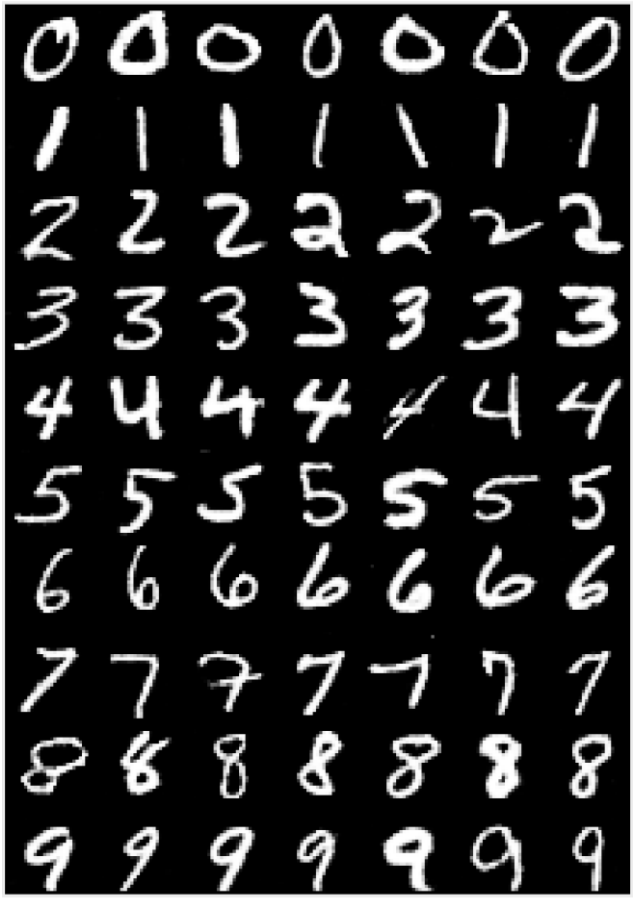


Deep Convolutional GAN: Examples



Groundtruth MNIST

GAN

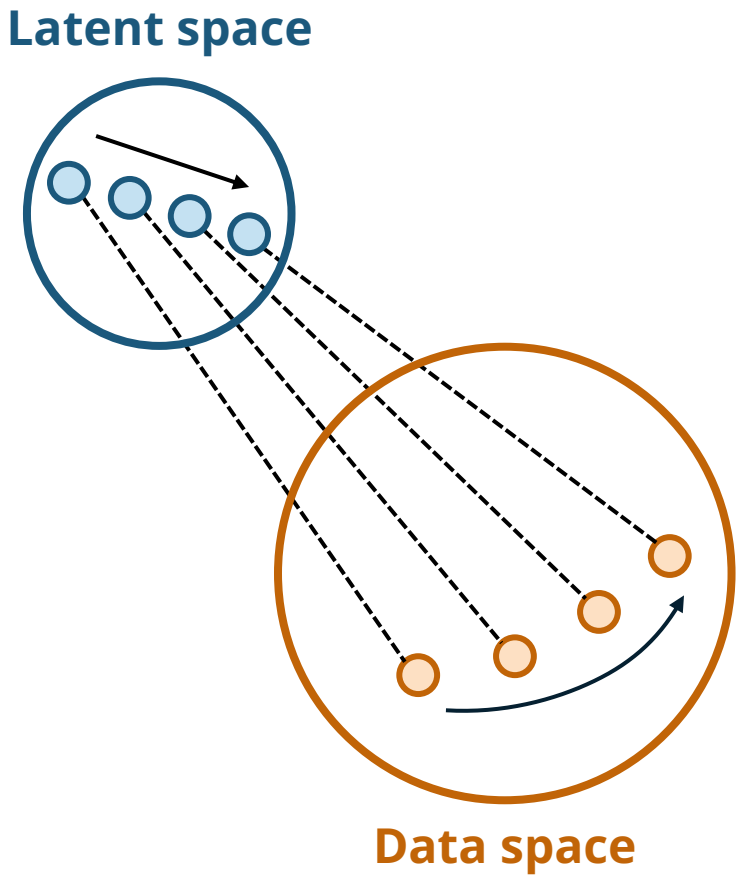


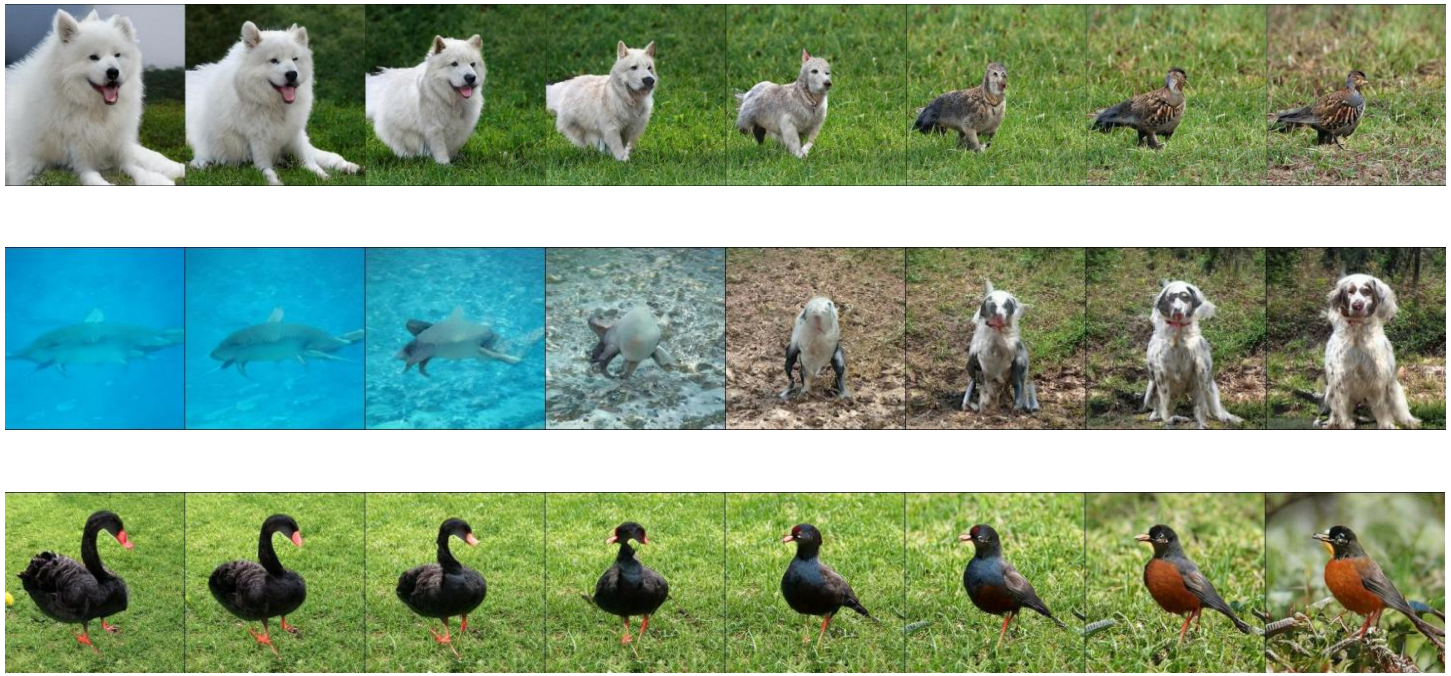
DCGAN (ours)

(Source: Radford et al., 2016)

Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *ICLR*, 2016.

Latent Space Interpolation of a GAN





(Source: Brock et al., 2019)

Andrew Brock, Jeff Donahue, and Karen Simonyan, "Large Scale GAN Training for High Fidelity Natural Image Synthesis," *ICLR*, 2019.

GigaGAN: Scaling up GANs


A portrait of a human growing colorful flowers from her hair. Hyperrealistic oil painting. Intricate details.

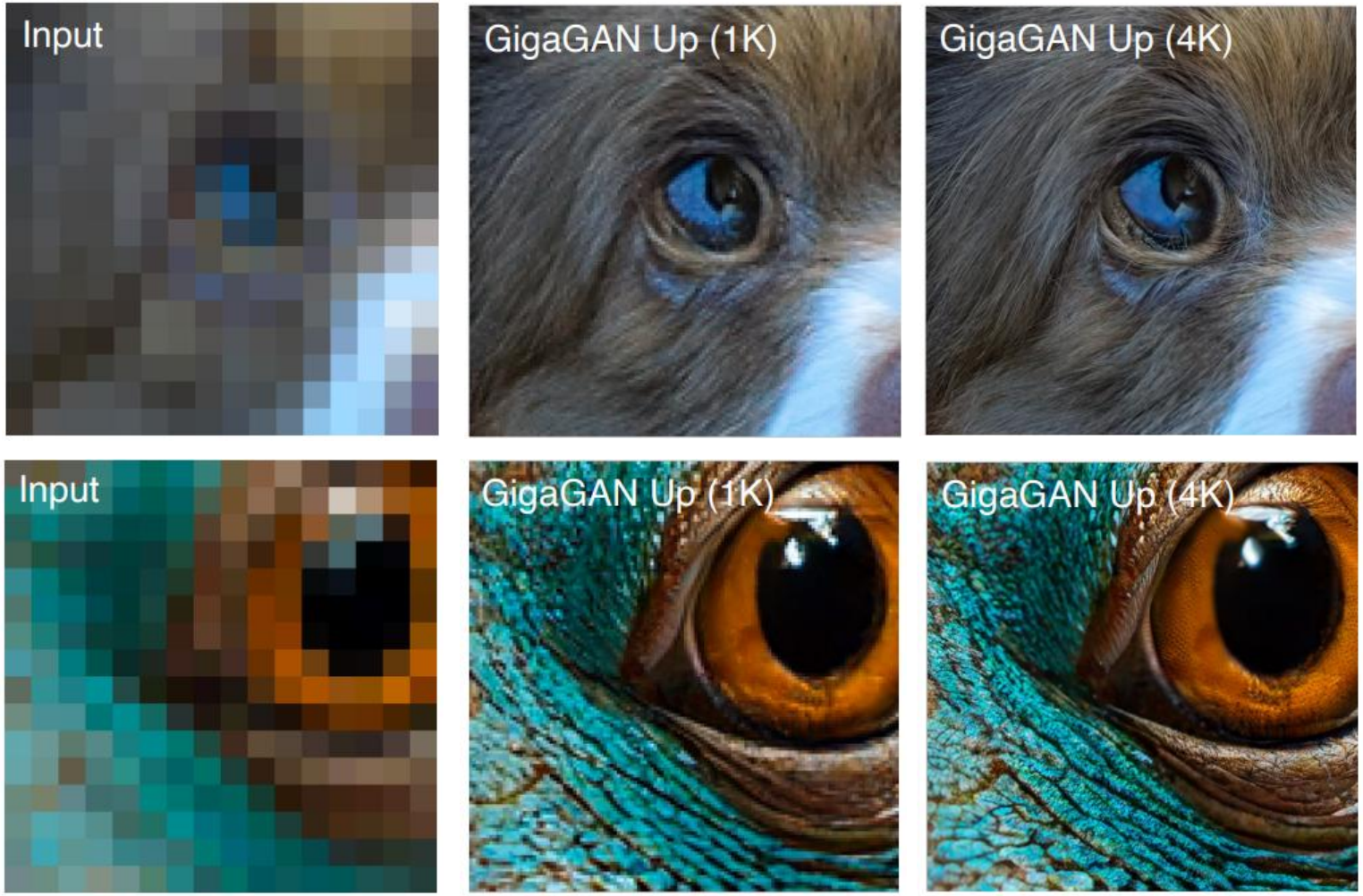

A golden luxury motorcycle parked at the King's palace. 35mm f/4.5.


a cute magical flying maltipoo at light speed, fantasy concept art, bokeh, wide sky

(Source: Kang et al., 2023)

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park, "Scaling up GANs for Text-to-Image Synthesis," *CVPR*, 2023.

GigaGAN for Image Super-resolution

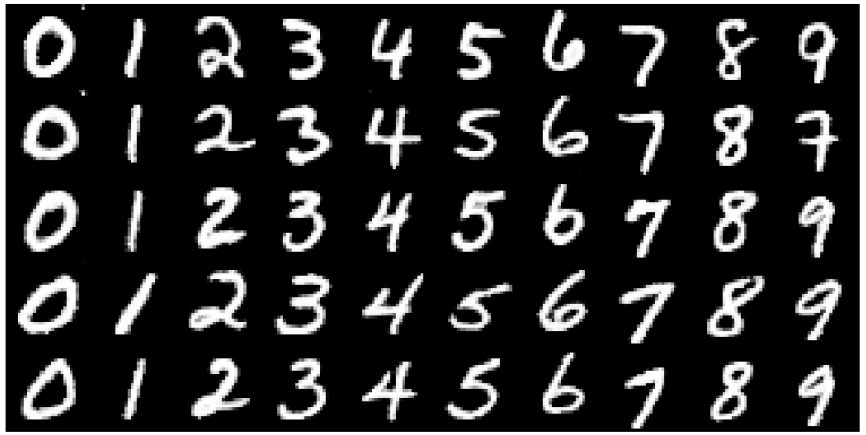


(Source: Kang et al., 2023)

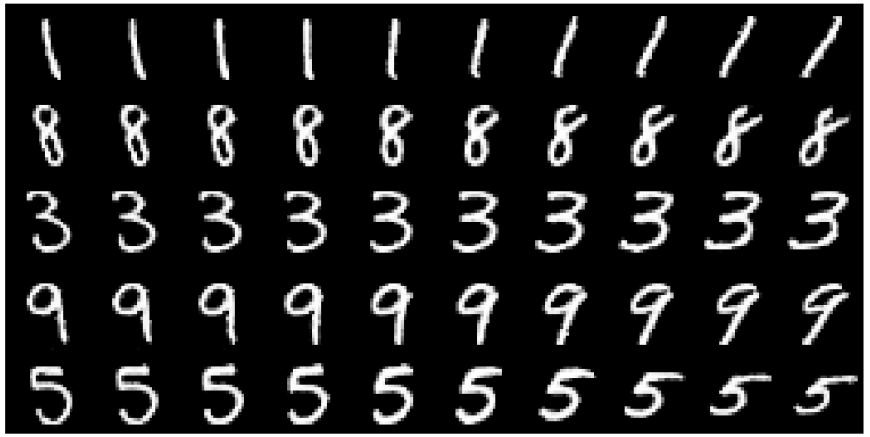
Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park, "Scaling up GANs for Text-to-Image Synthesis," *CVPR*, 2023.

Controlling the Latent Variables

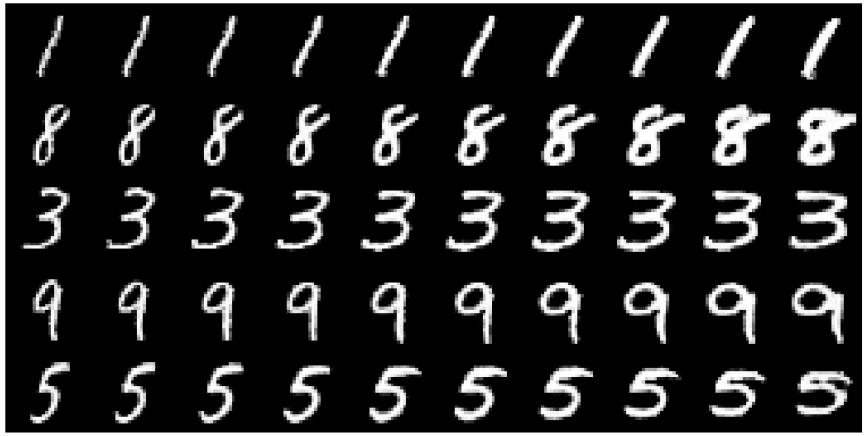
Digit type



Rotation



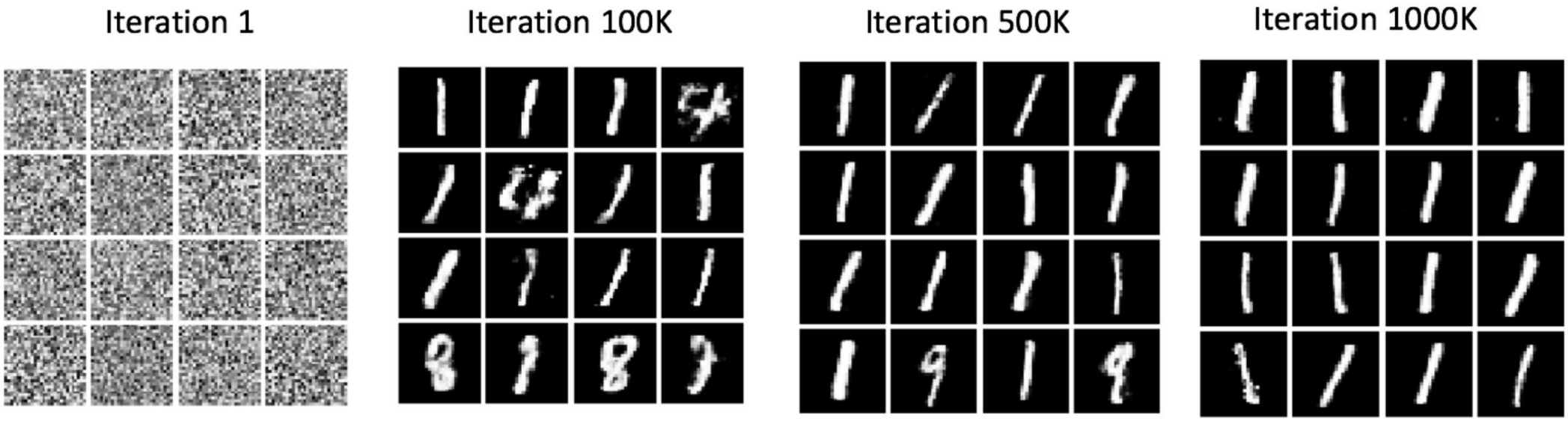
Width



(Source: Chen et al., 2016)

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel, "InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets," *NuerIPS*, 2016.

Mode Collapse



(Source: Mi et al., 2018)

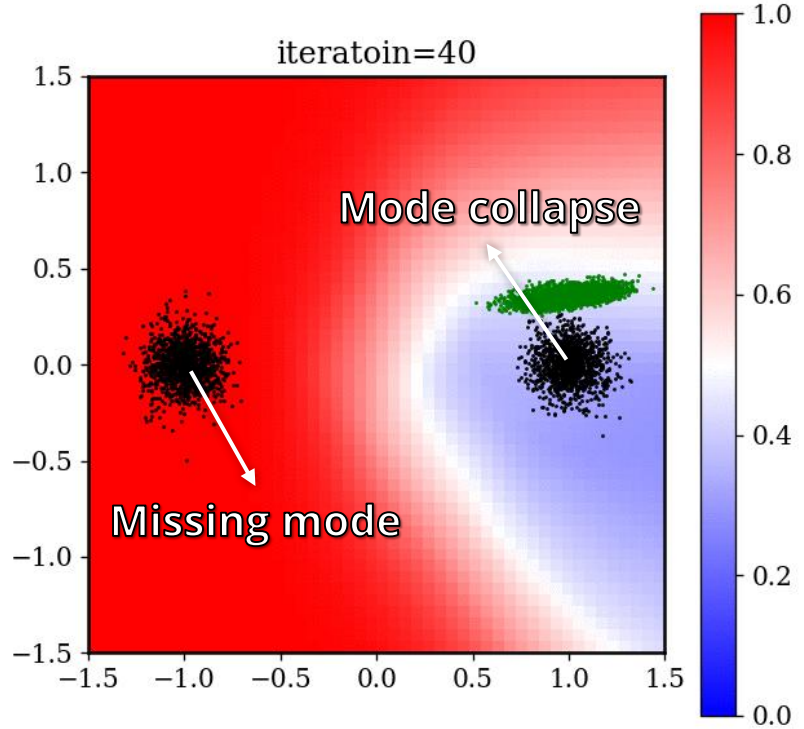
Cover only 1 out of the 10 modes

Lu Mi, Macheng Shen, and Jingzhao Zhang, "A Probe Towards Understanding GAN and VAE Models," *arXiv preprint arXiv:1812.05676*, 2018.

Problems of Unregularized GANs

- **Key**—discriminator provides generator with gradients as a **guidance for improvement**
  - Discriminator has an easier job than the generator
  - Discriminator tends to provide large gradients
  - Results in unstable training of the generator
- Common failure cases
  - **Mode collapse**
  - **Missing modes**

A steep color change indicates a large gradient



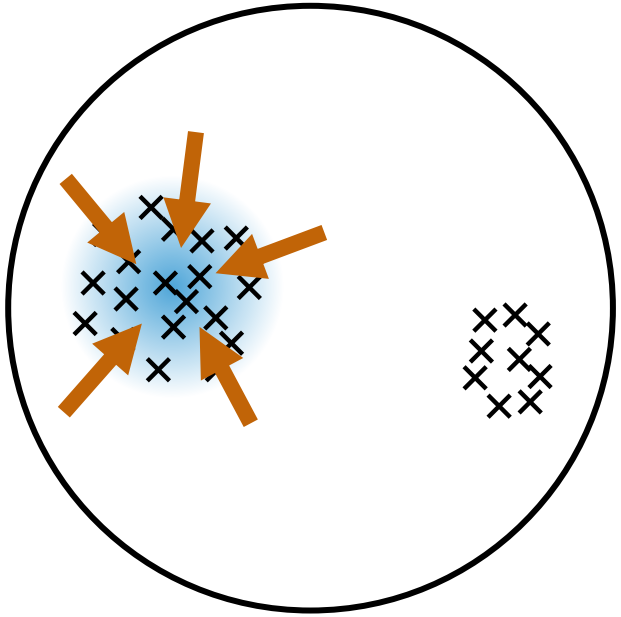
(Colors show the discriminator outputs)

Regularizing GANs

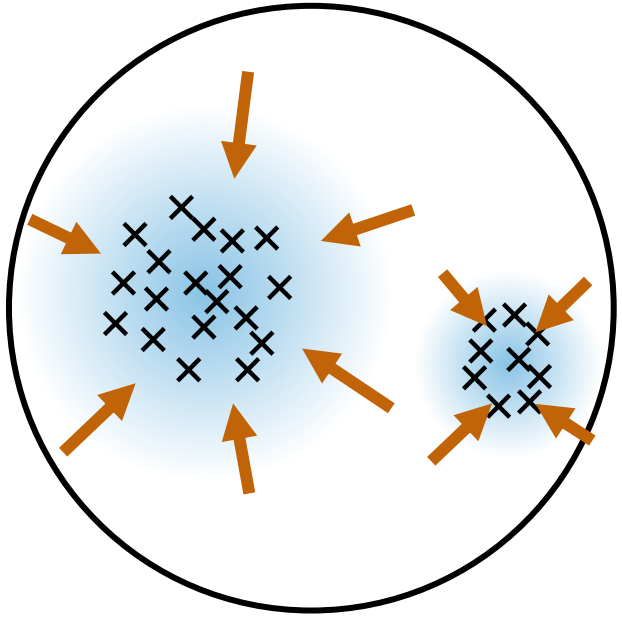
Advantages of gradient regularization

- **Provide a smoother guidance to the generator**
- **Alleviate mode collapse and missing modes issues**

Unregularized



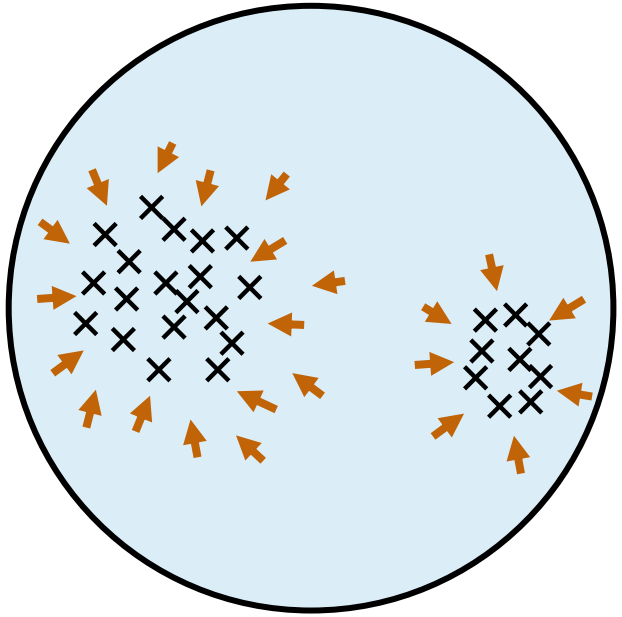
Locally regularized



Gradient clipping [1]

Gradient penalties [2,3]

Globally regularized

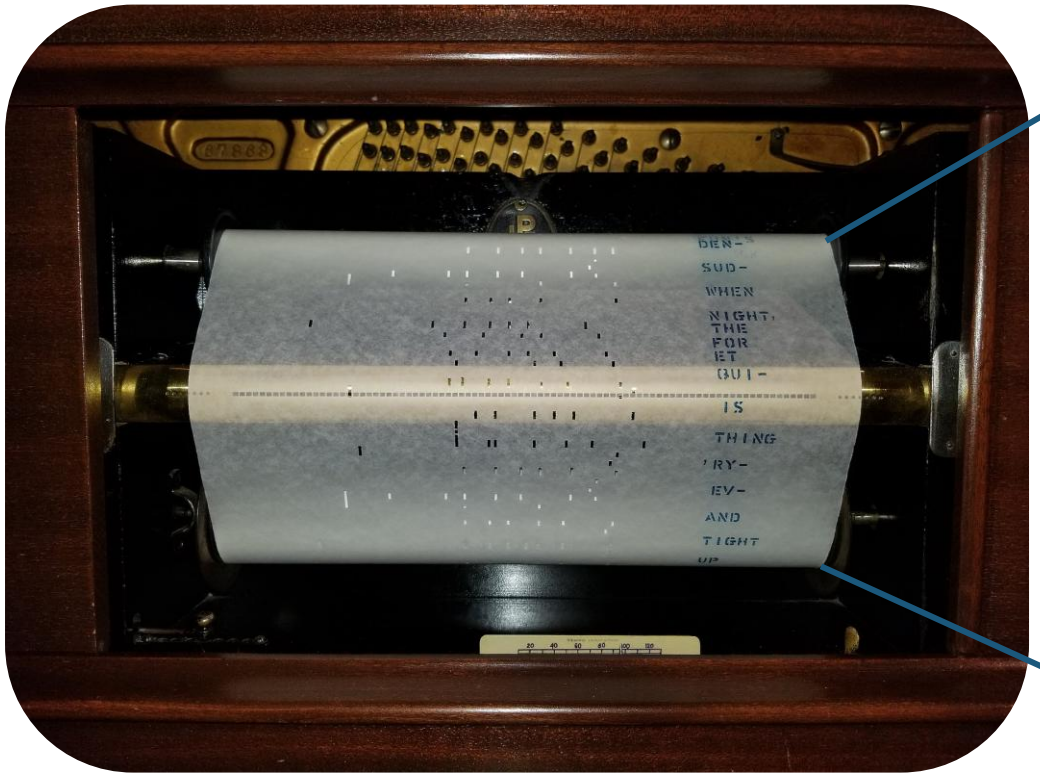


Spectral normalization [4]

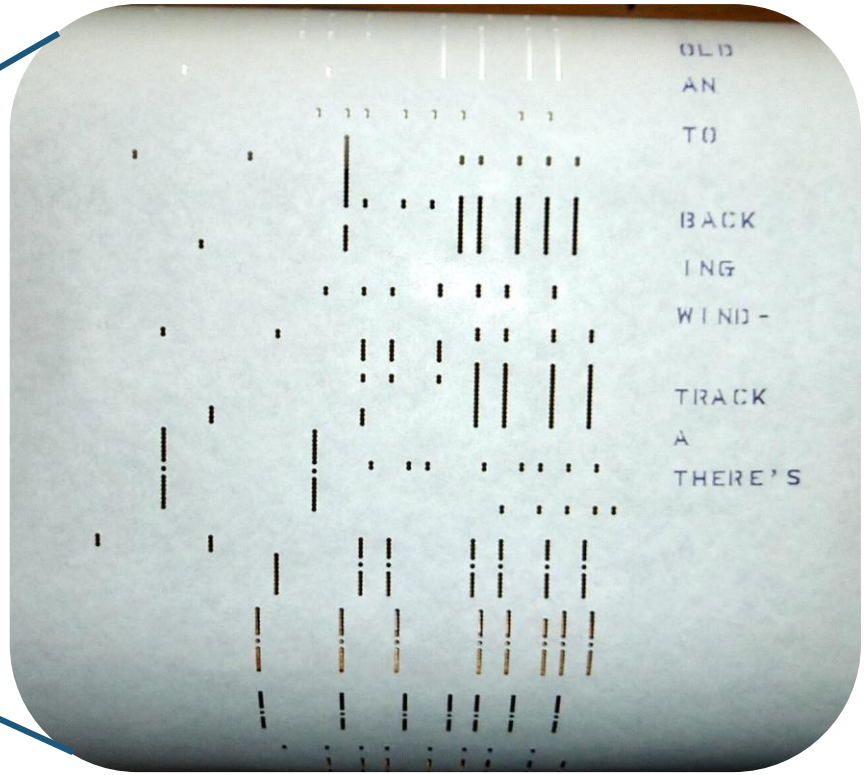
[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "Wasserstein Generative Adversarial Networks," *ICML,* 2017.
[2] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville, "Improved Training of Wasserstein GANs," *NeurIPS,* 2017.
[3] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira, "On Convergence and Stability of GANs," *arXiv preprint arXiv:1705.07215,* 2017.
[4] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida, "Spectral Normalization for Generative Adversarial Networks," *ICLR,* 2018.

Piano Roll Representation

Piano Rolls



(Source: Draconichiaro)

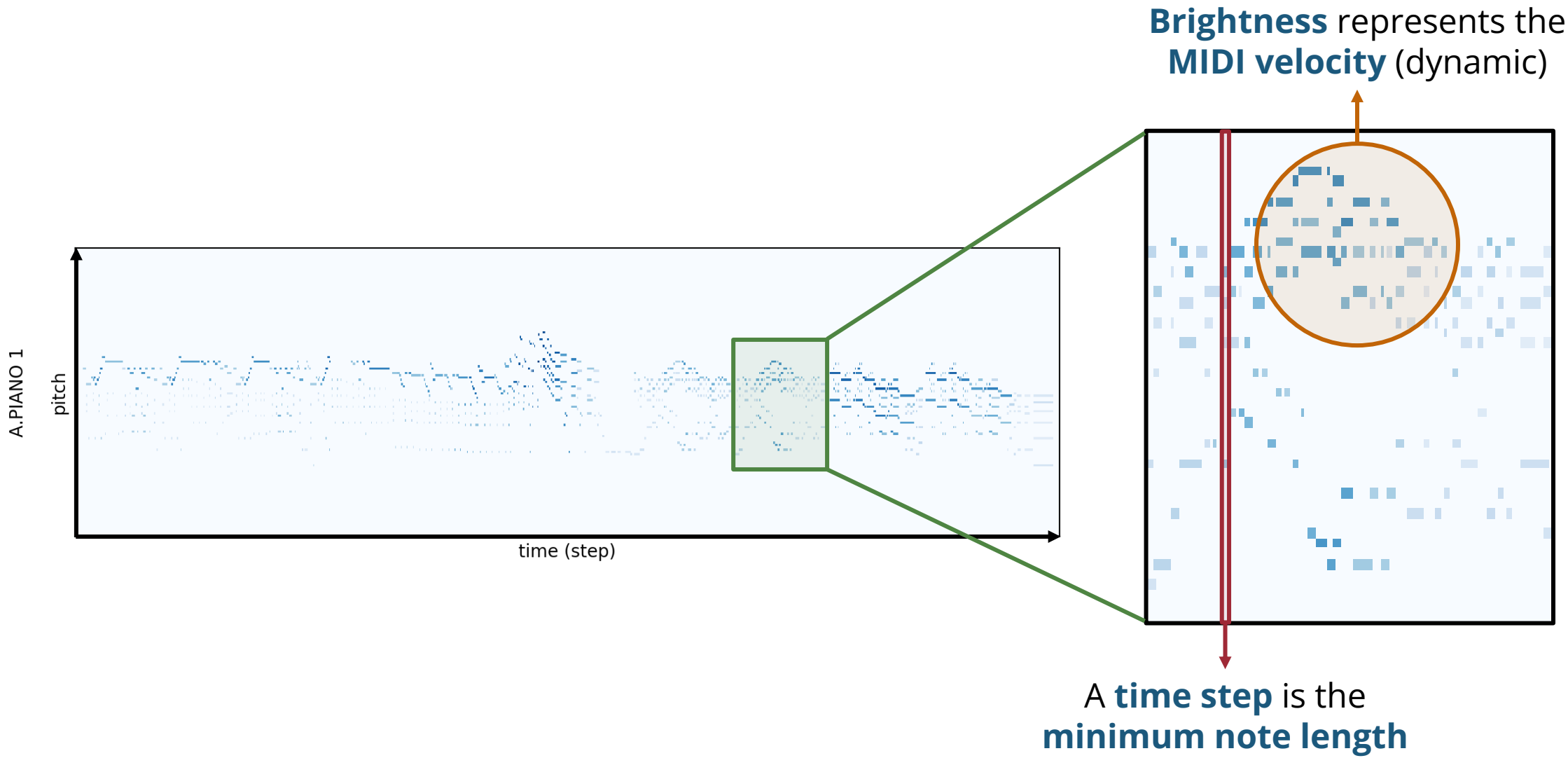


(Source: Tangerineduel)

Player Pianos

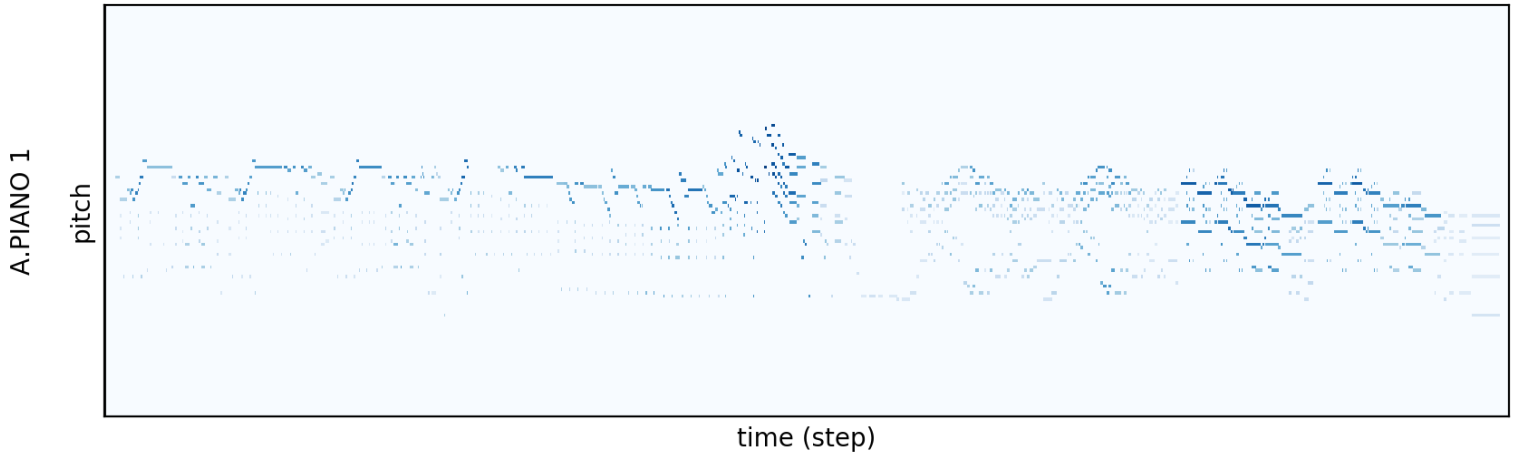
youtu.be/07krQ661fok

Piano Roll Representation

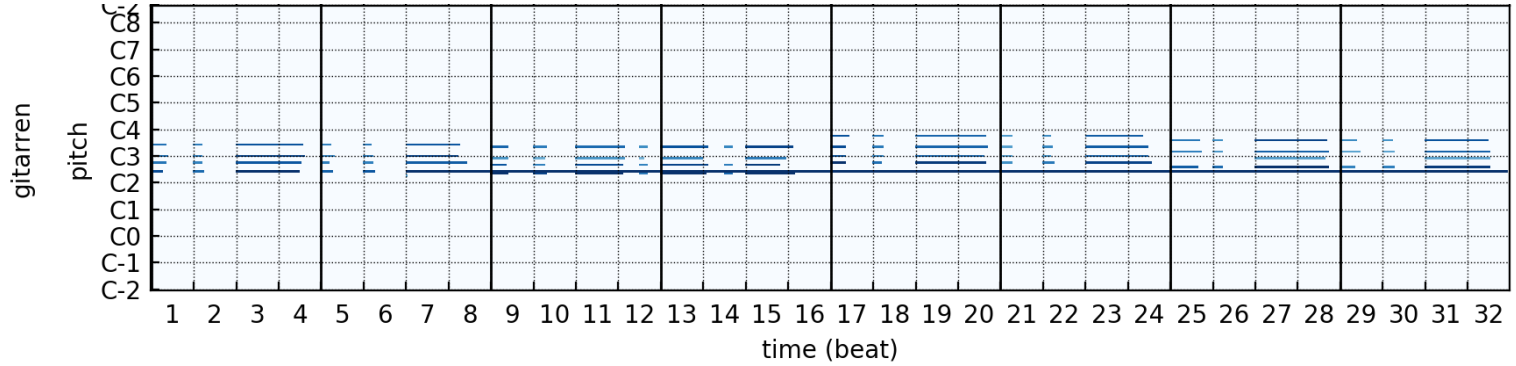


Piano Roll Representation

With expressive timing

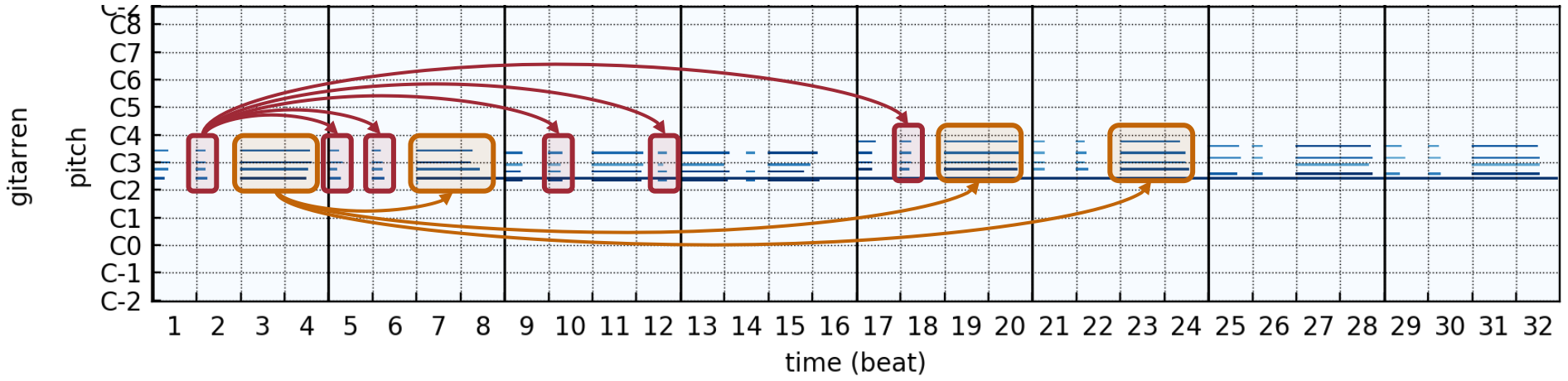


Without expressive timing



Reusable Pattern Detectors

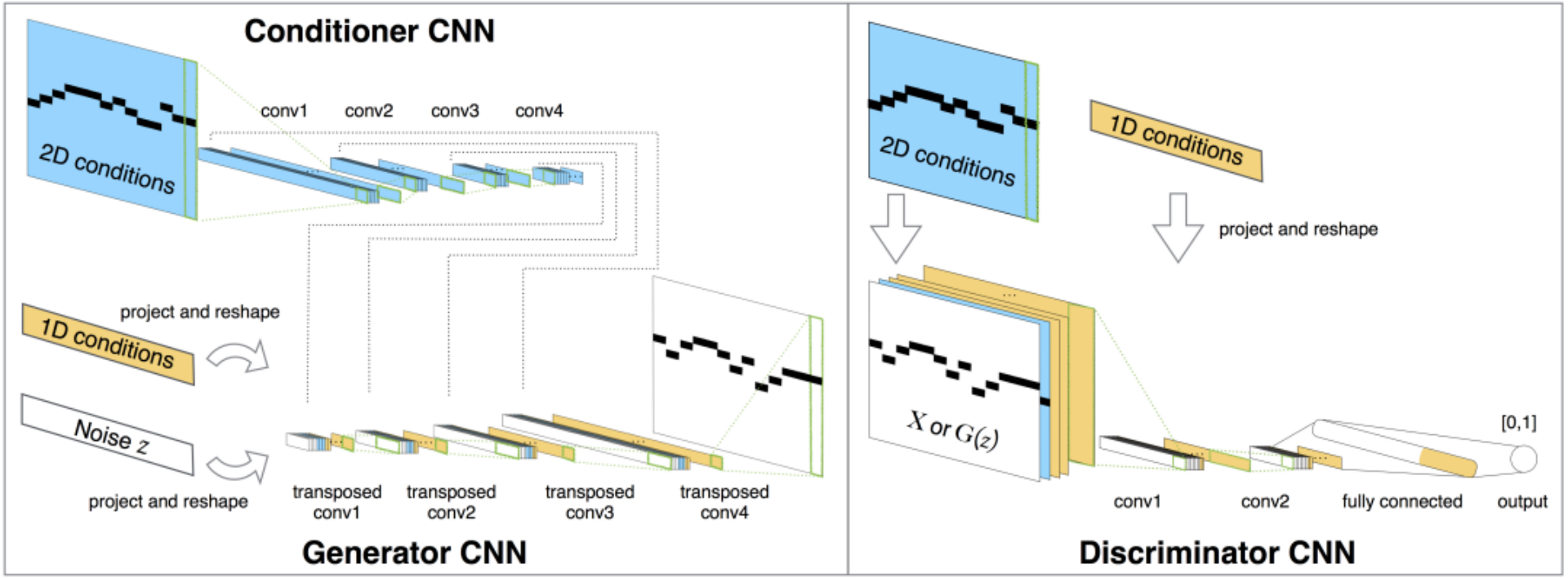
Why Piano Rolls?



Many musical patterns like melodies, chords, scales and arpeggios are translational invariant in the temporal and pitch axes

Generating Music using GANs

MidiNet (Yang et al., 2017)

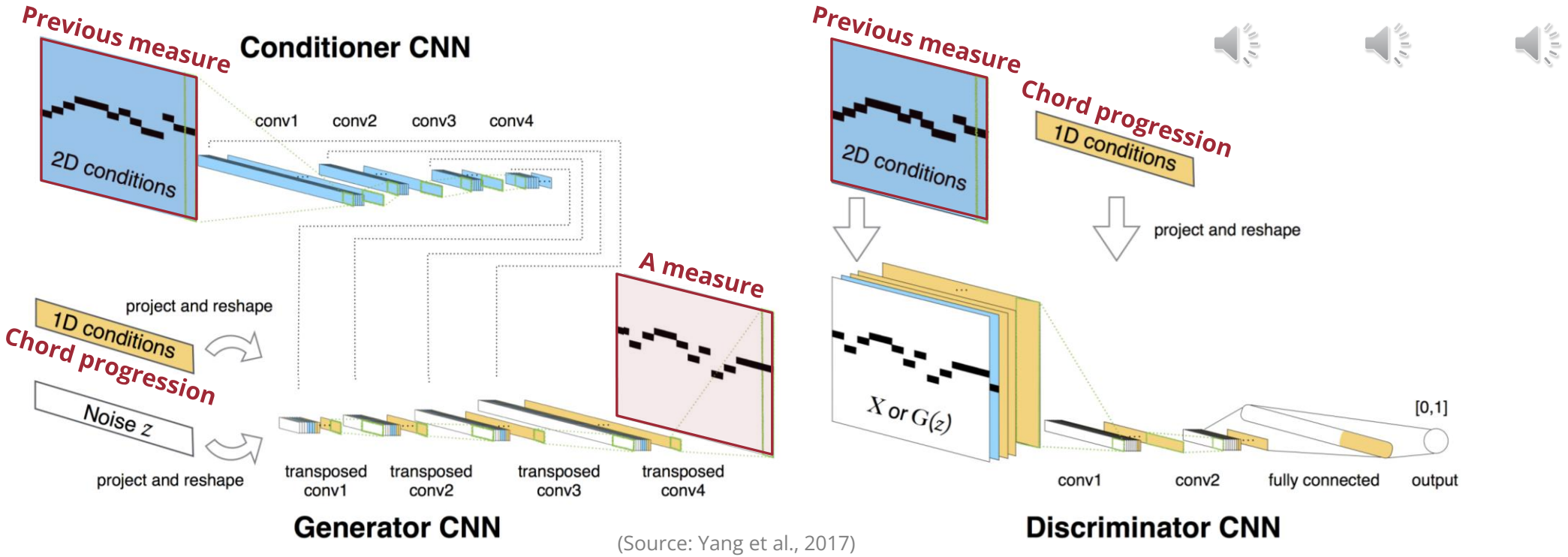


(Source: Yang et al., 2017)

Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang, "MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation," *ISMIR*, 2017.

MidiNet (Yang et al., 2017)

Examples of generated music

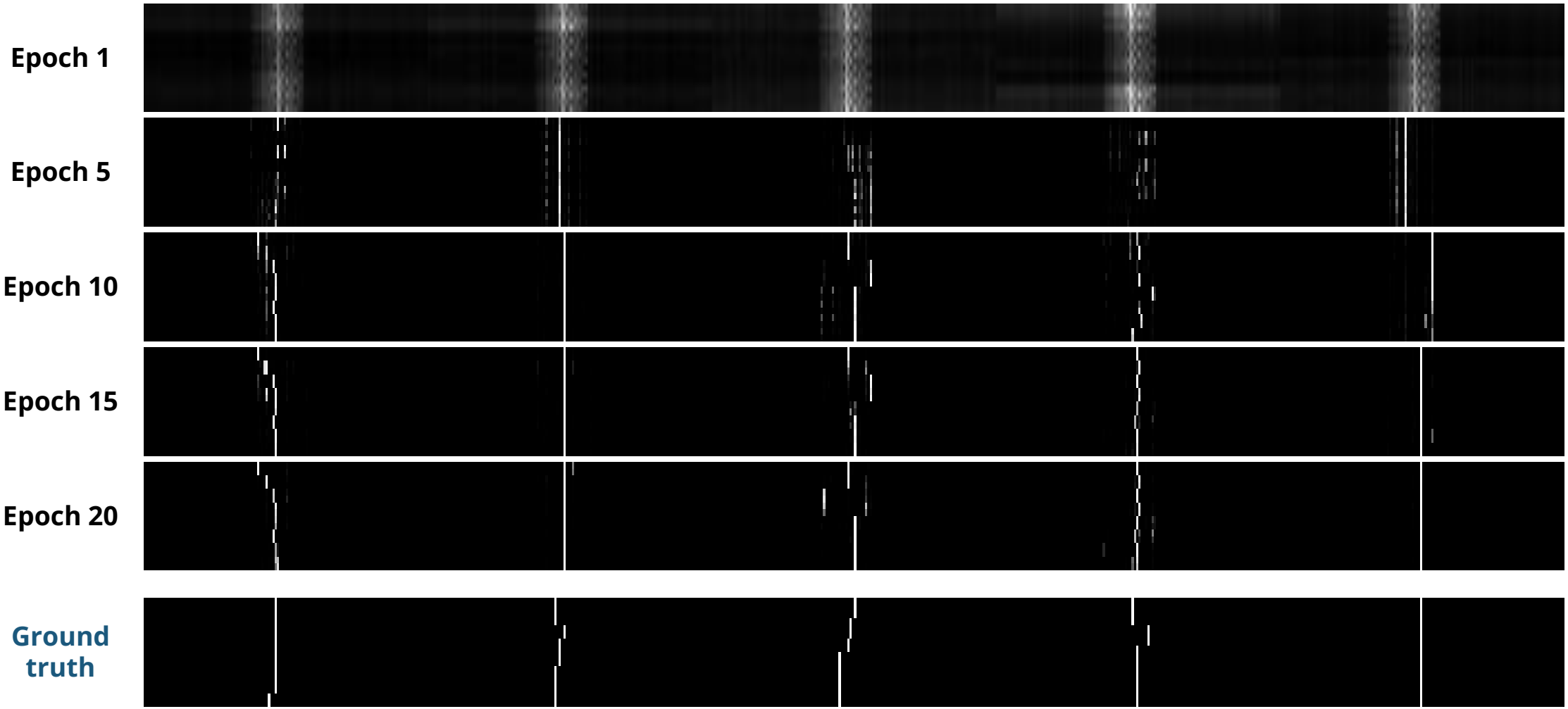


(Source: Yang et al., 2017)

Generate the next measure given the previous one (measure by measure)

Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang, "MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation," *ISMIR*, 2017.

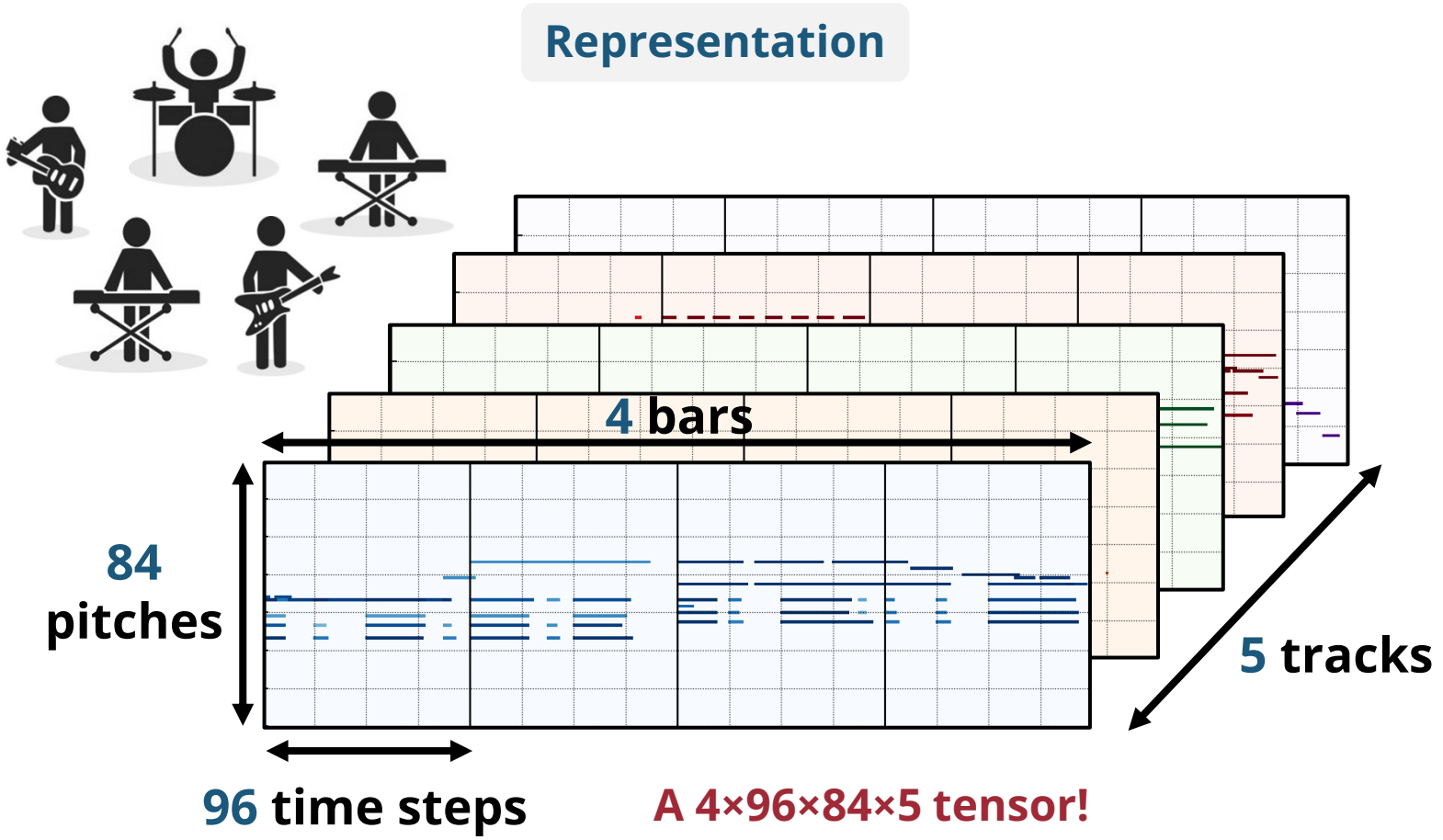
MidiNet (Yang et al., 2017)

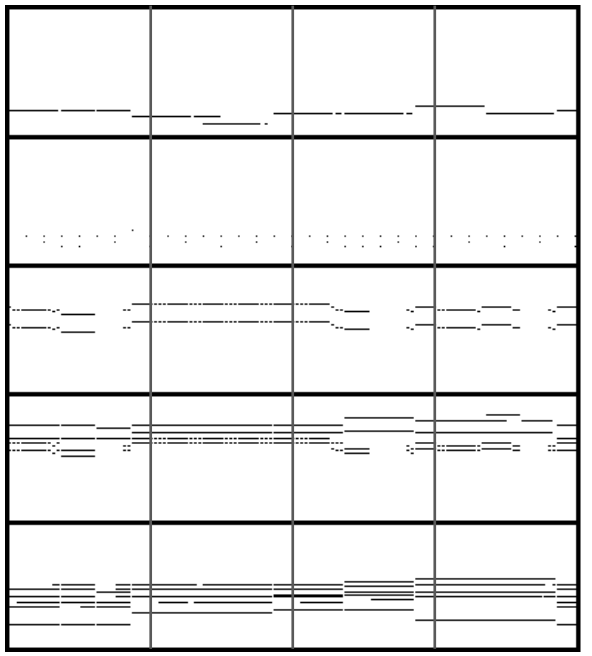


(Source: Yang et al., 2017)

Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang, "MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation," *ISMIR*, 2017.

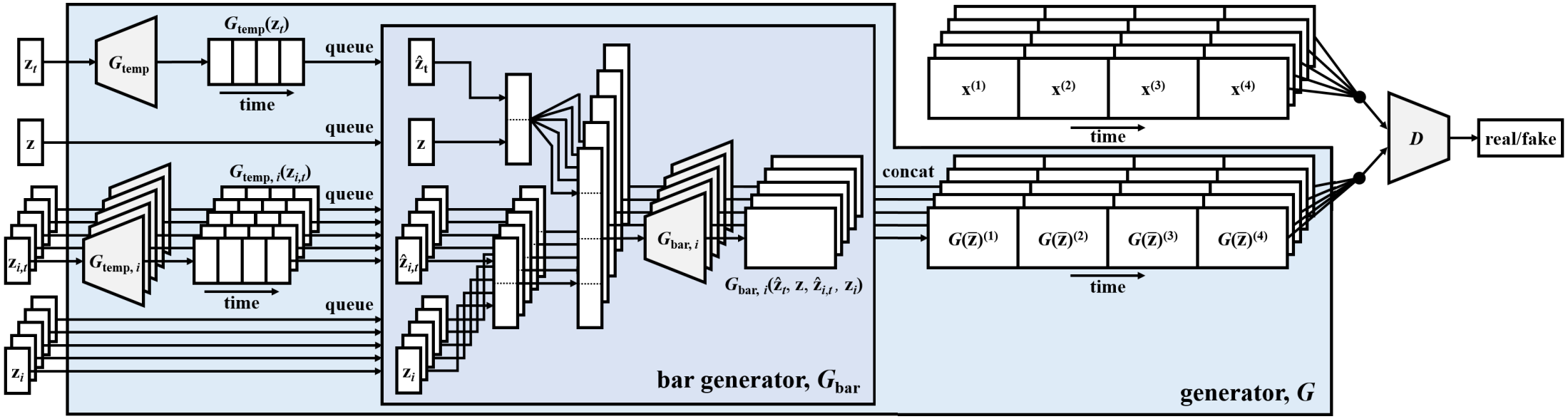
MuseGAN (Dong et al., 2018)





Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," *AAAI*, 2018.

MuseGAN (Dong et al., 2018)

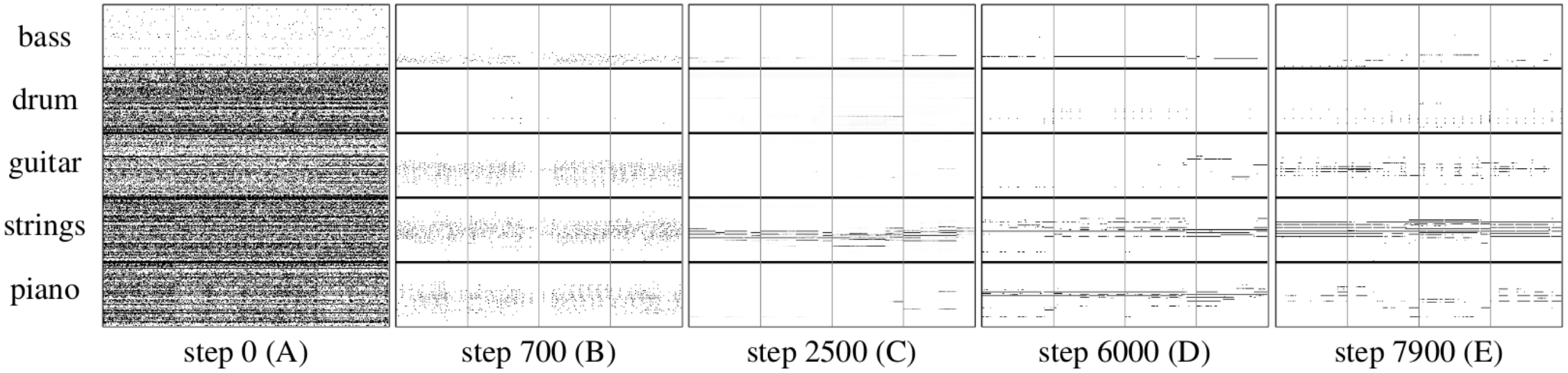


(Source: Dong et al., 2018)

Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," *AAAI*, 2018.

MuseGAN (Dong et al., 2018)

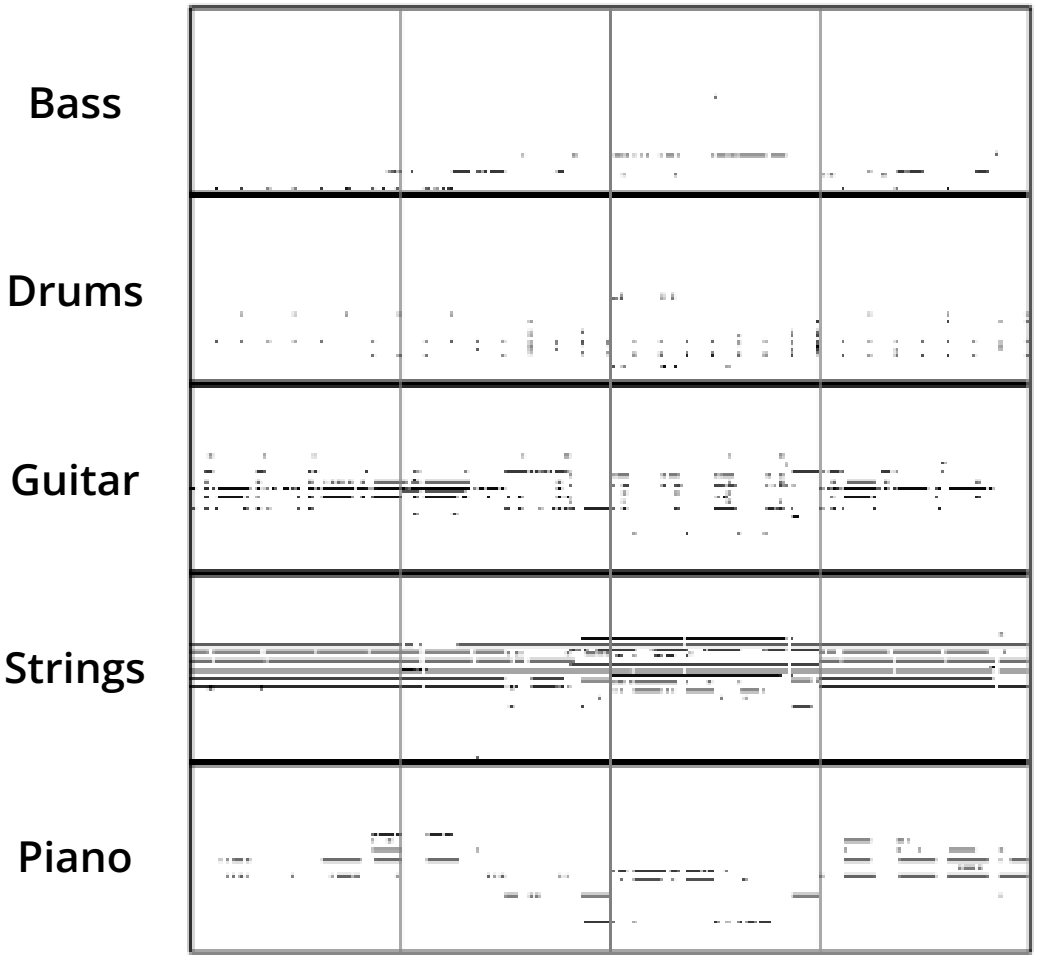
The generator improves over time



(Source: Dong et al., 2018)

Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," *AAAI*, 2018.

MuseGAN (Dong et al., 2018)

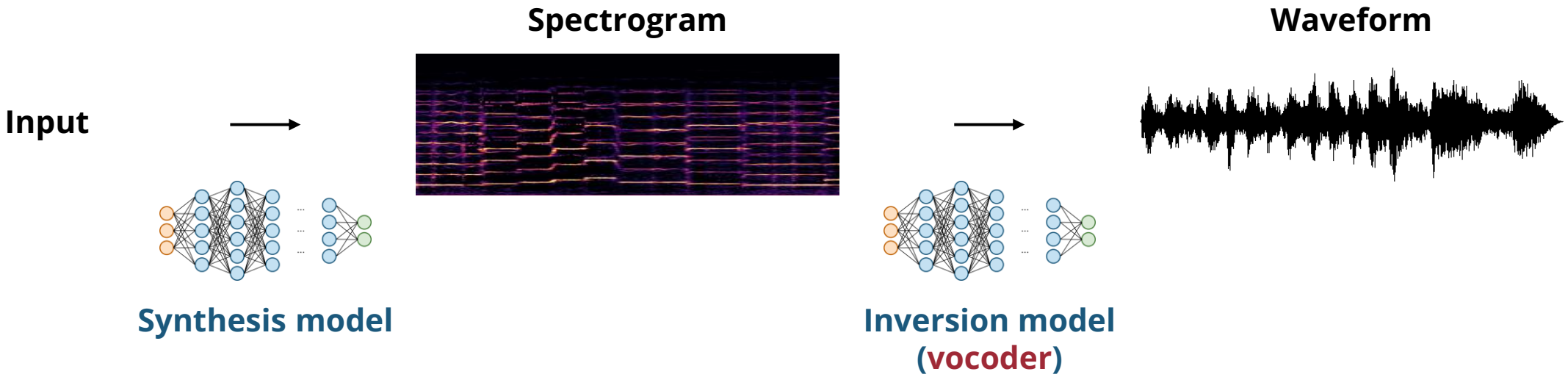


Examples of generated music

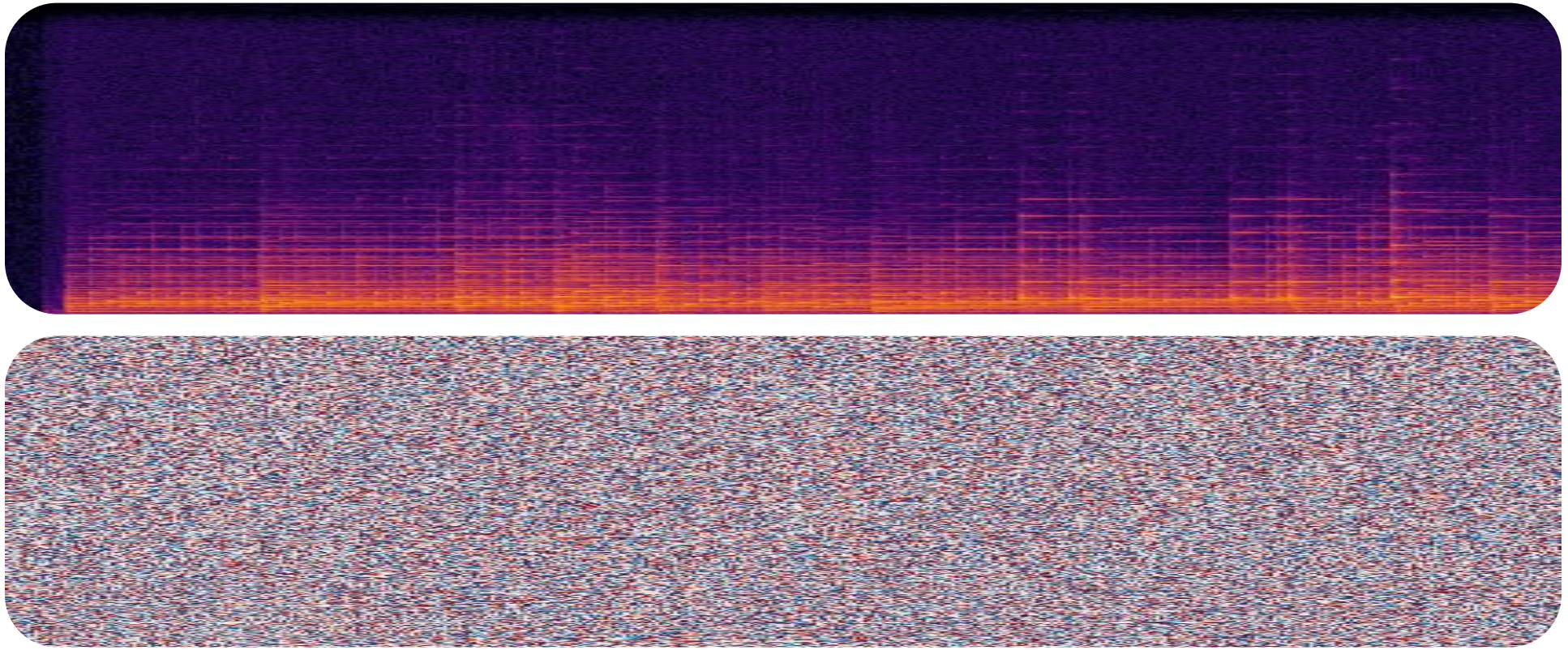
Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," *AAAI*, 2018.

Generative Adversarial Nets for Audio

Frequency-domain Audio Synthesis



Importance of the Phase Information

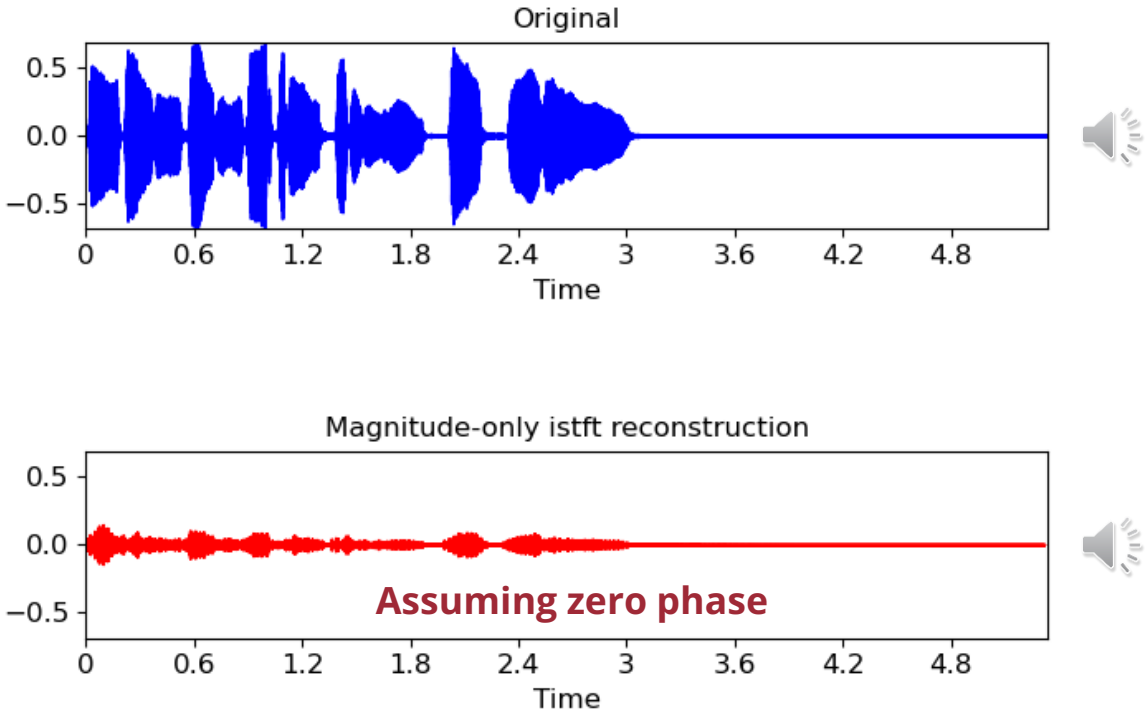


(Source: Dieleman et al., 2020)

Real phase **Random phase**

Sander Dieleman, "Generating music in the waveform domain," *Sander Dieleman's Blog*, March 24, 2020.

Inverse STFT without Phase Information



(Source: librosa documentation)

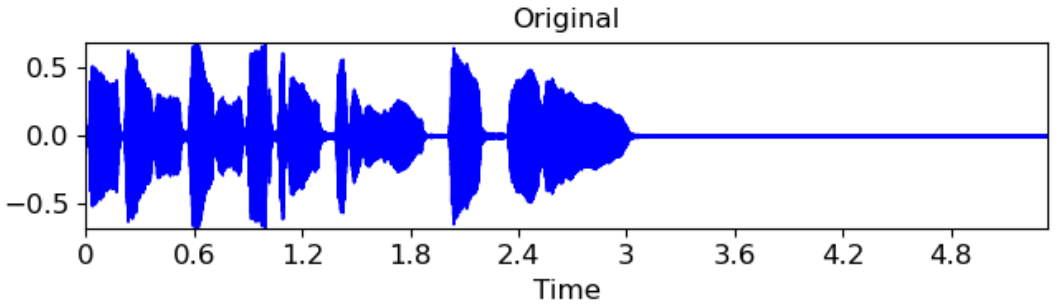
Complex-valued STFT matrix

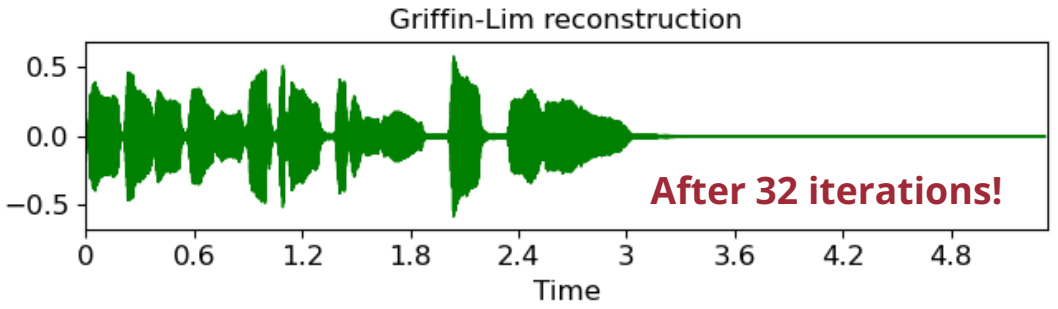
$$\text{ISTFT}(M) = \arg\min_{y} (M - \text{STFT}(y))^2$$

Find the signal y that minimize the MSE between the input and STFT(y)

librosa.org/doc/main/generated/librosa.griffinlim.html

Griffin-Lim Algorithm (Griffin & Lim, 1984)





(Source: librosa documentation)

Given a magnitude-only STFT matrix

↓

Randomly initialize the phase

↓

$$y' = \arg\min_{y} (M - \mathrm{STFT}(y))^2$$

Find the signal y that minimize the MSE between the input and STFT(y)

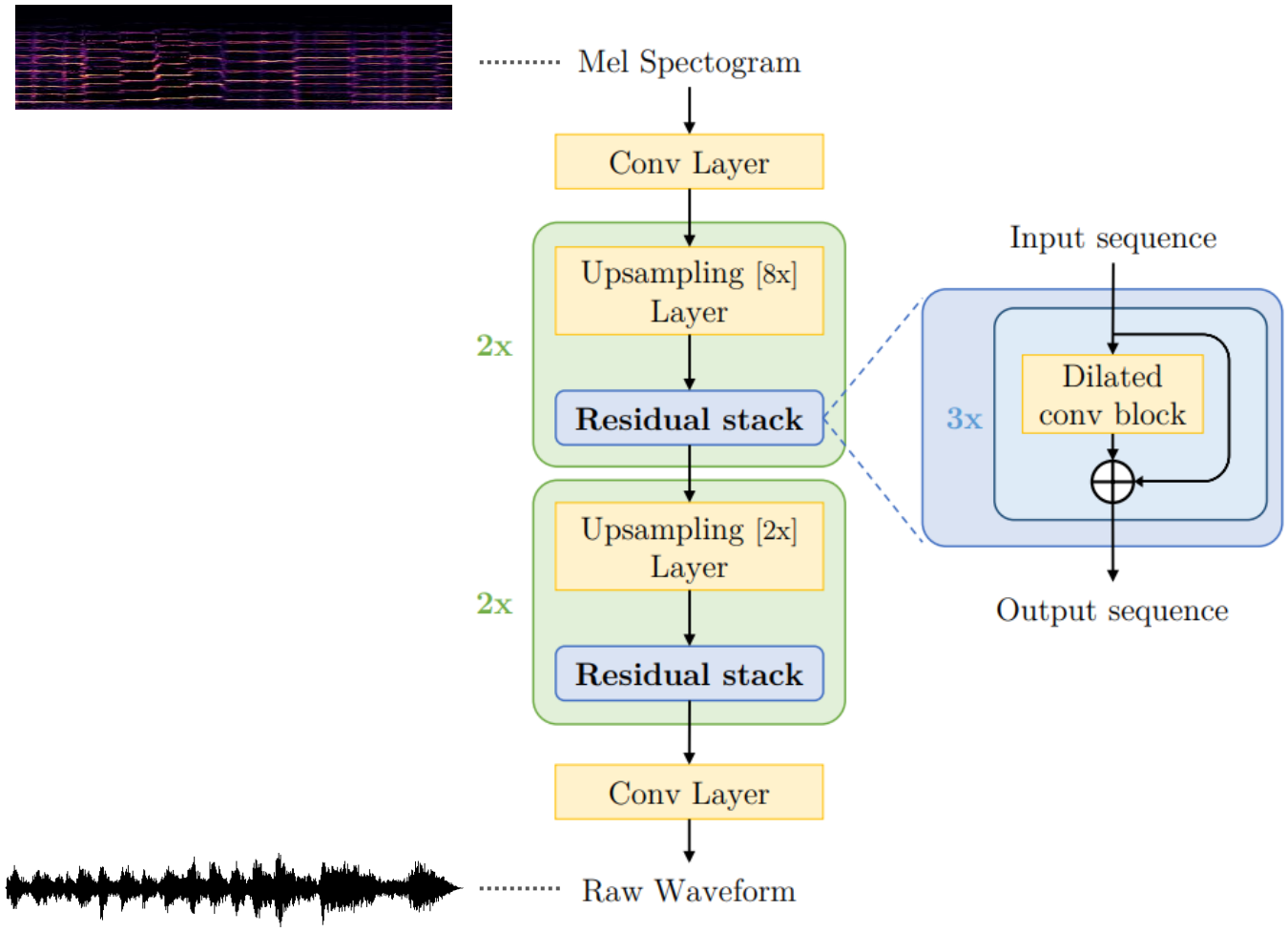
↓

$$M' = \mathrm{STFT}(y')$$

Find the STFT of the signal y

Daniel Griffin and Jae Lim, "Signal estimation from modified short-time fourier transform," *IEEE TASSP*, 1984.
librosa.org/doc/main/generated/librosa.griffinlim.html

MelGAN (Kumar et al., 2019)

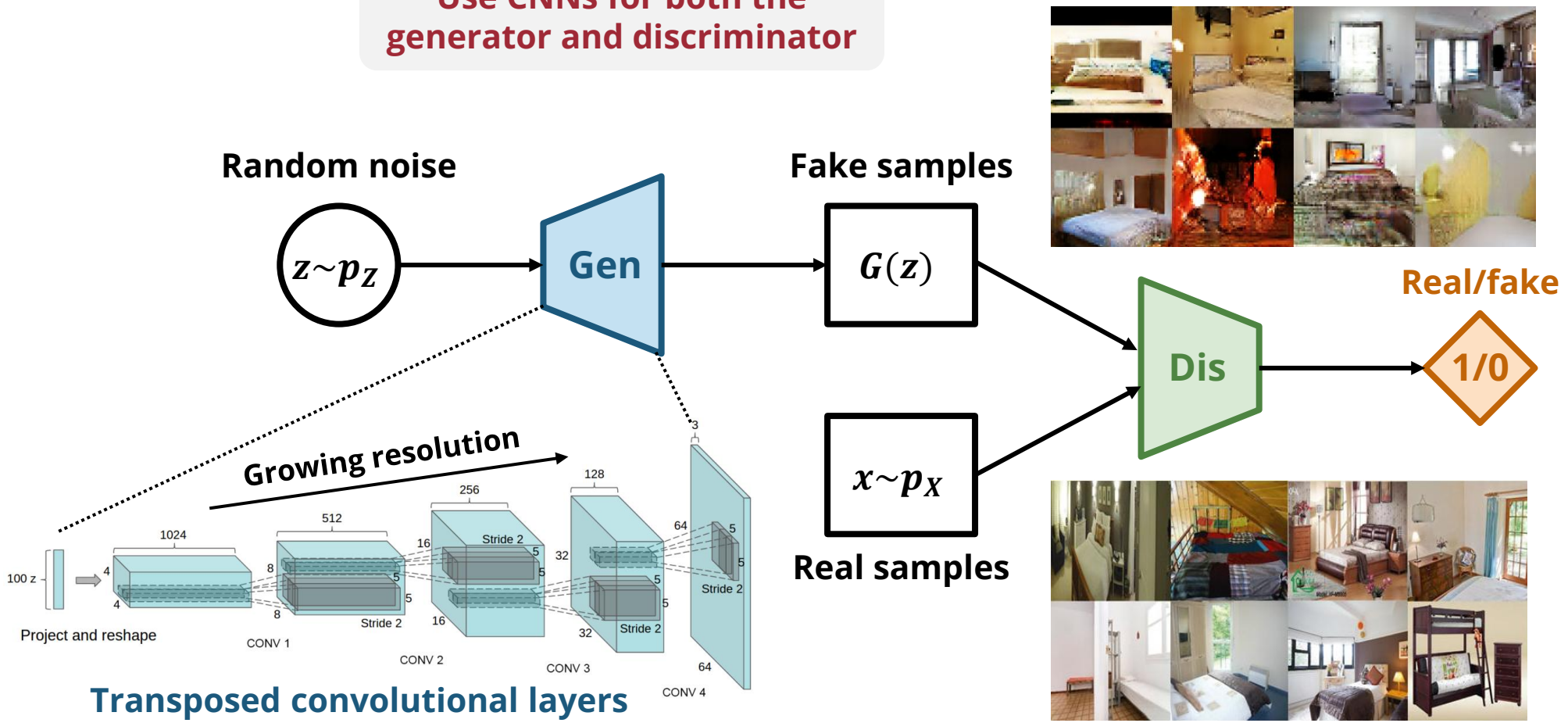


(Source: Kumar et al., 2019)

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville, "MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis," *NeurIPS*, 2019.

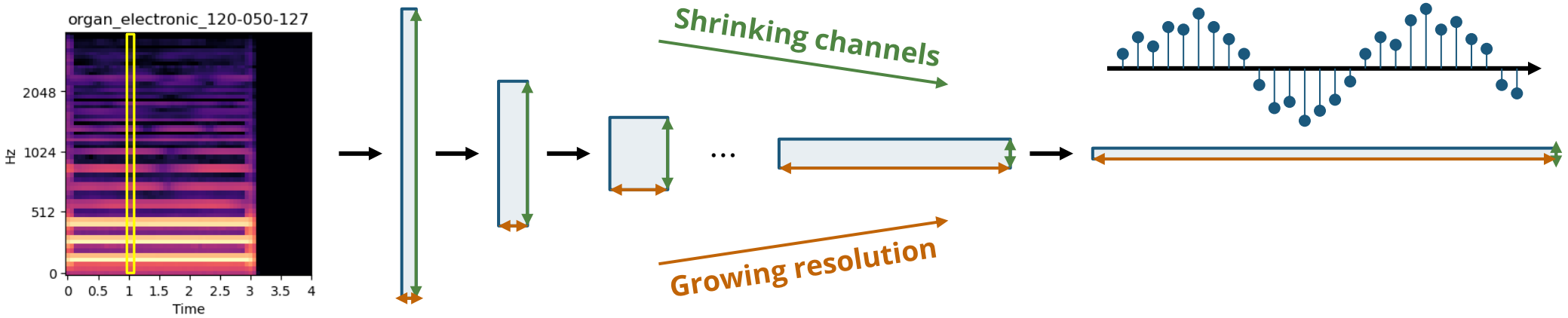
Deep Convolutional GANs (DCGANs) (Radford et al., 2014)

Use CNNs for both the generator and discriminator

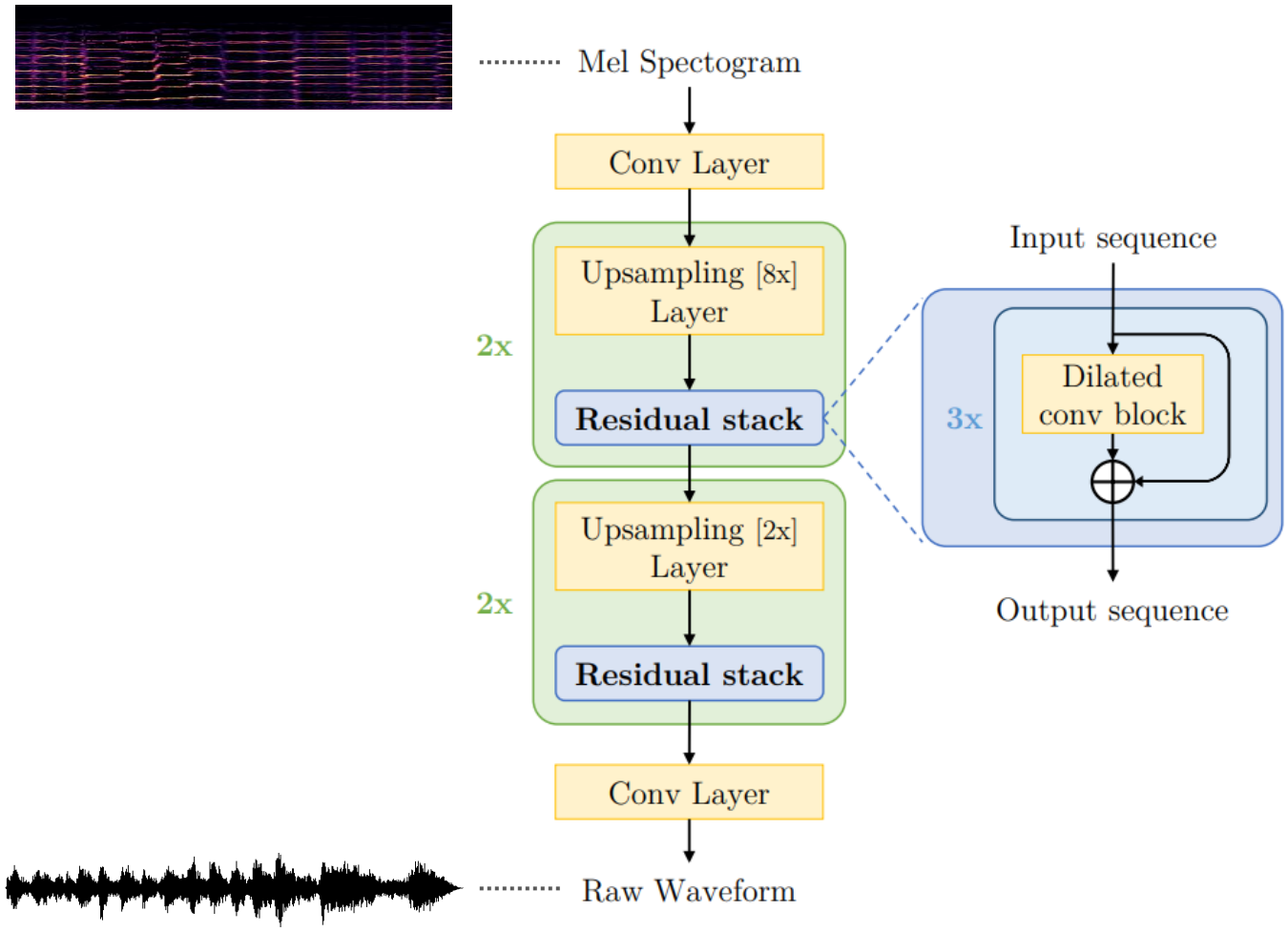


Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *ICLR*, 2016.

Upsampling for Vocoders



MelGAN (Kumar et al., 2019)

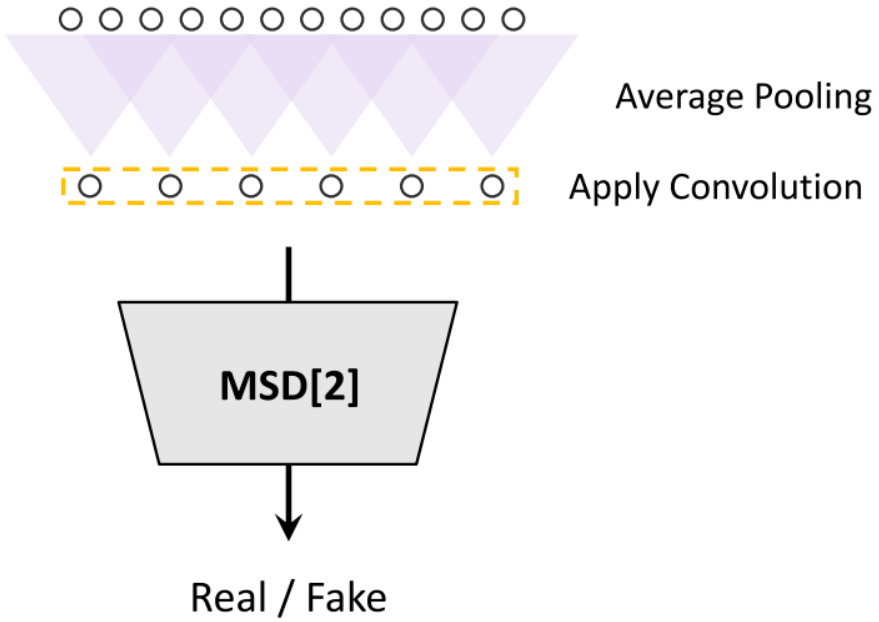


(Source: Kumar et al., 2019)

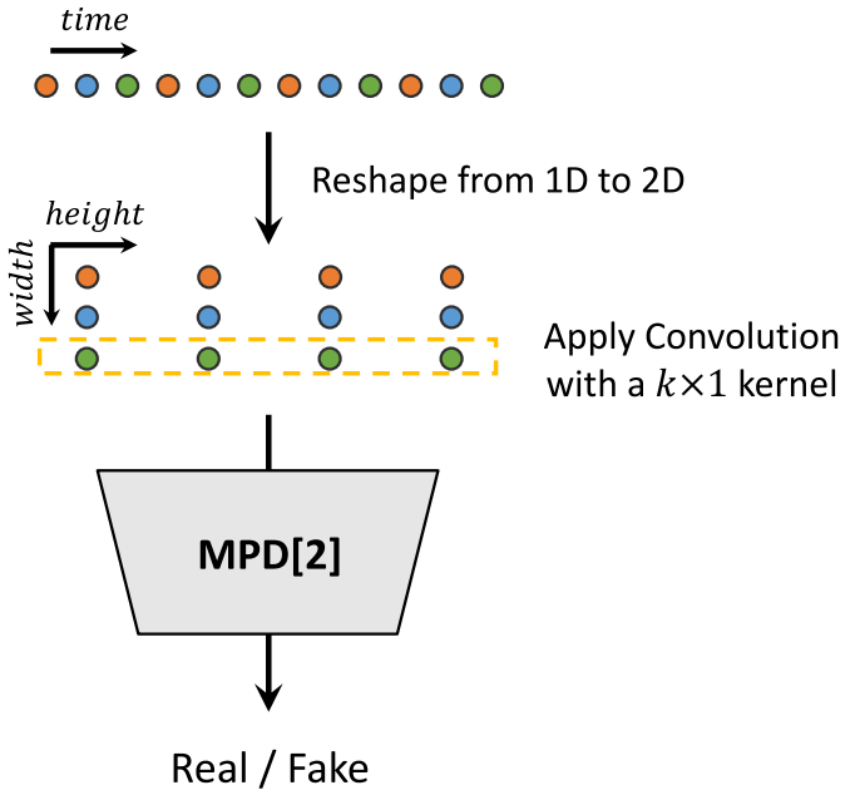
Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville, "MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis," *NeurIPS*, 2019.

MelGAN (Kumar et al., 2019)

Multi-scale discriminator



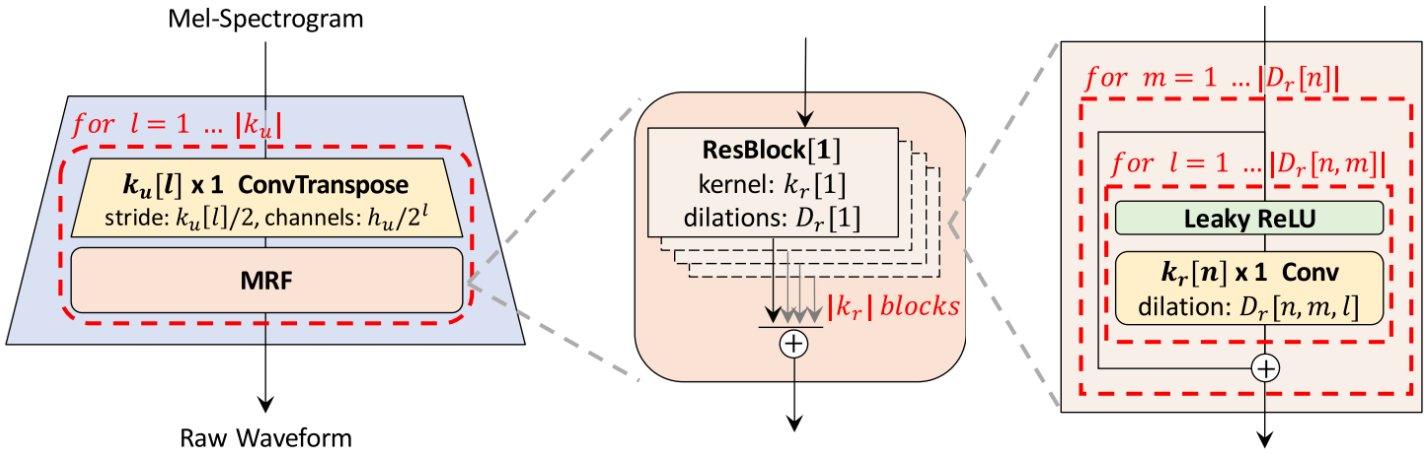
Multi-period discriminator

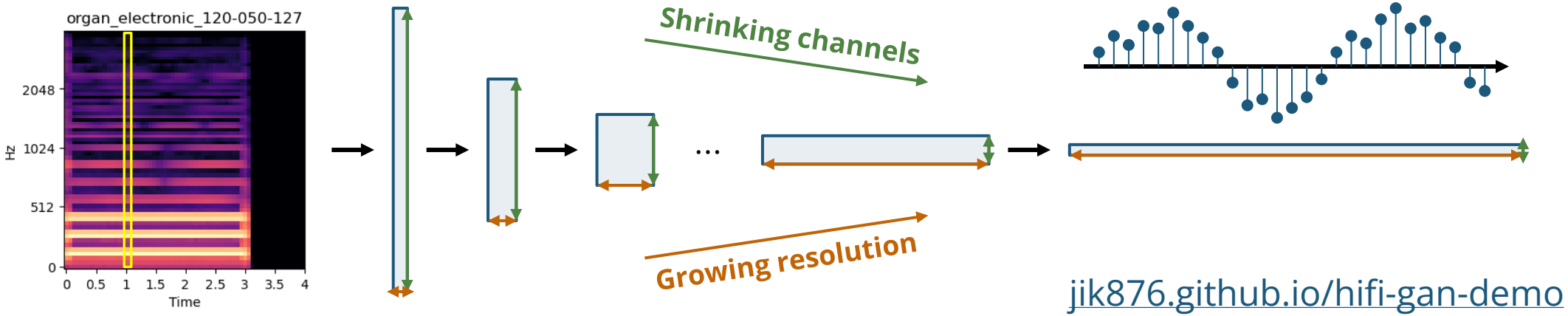


(Source: Kong et al., 2020)

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville, "MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis," *NeurIPS*, 2019.
Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis," *NeurIPS*, 2020.

Hifi-GAN (Kong et al., 2020)





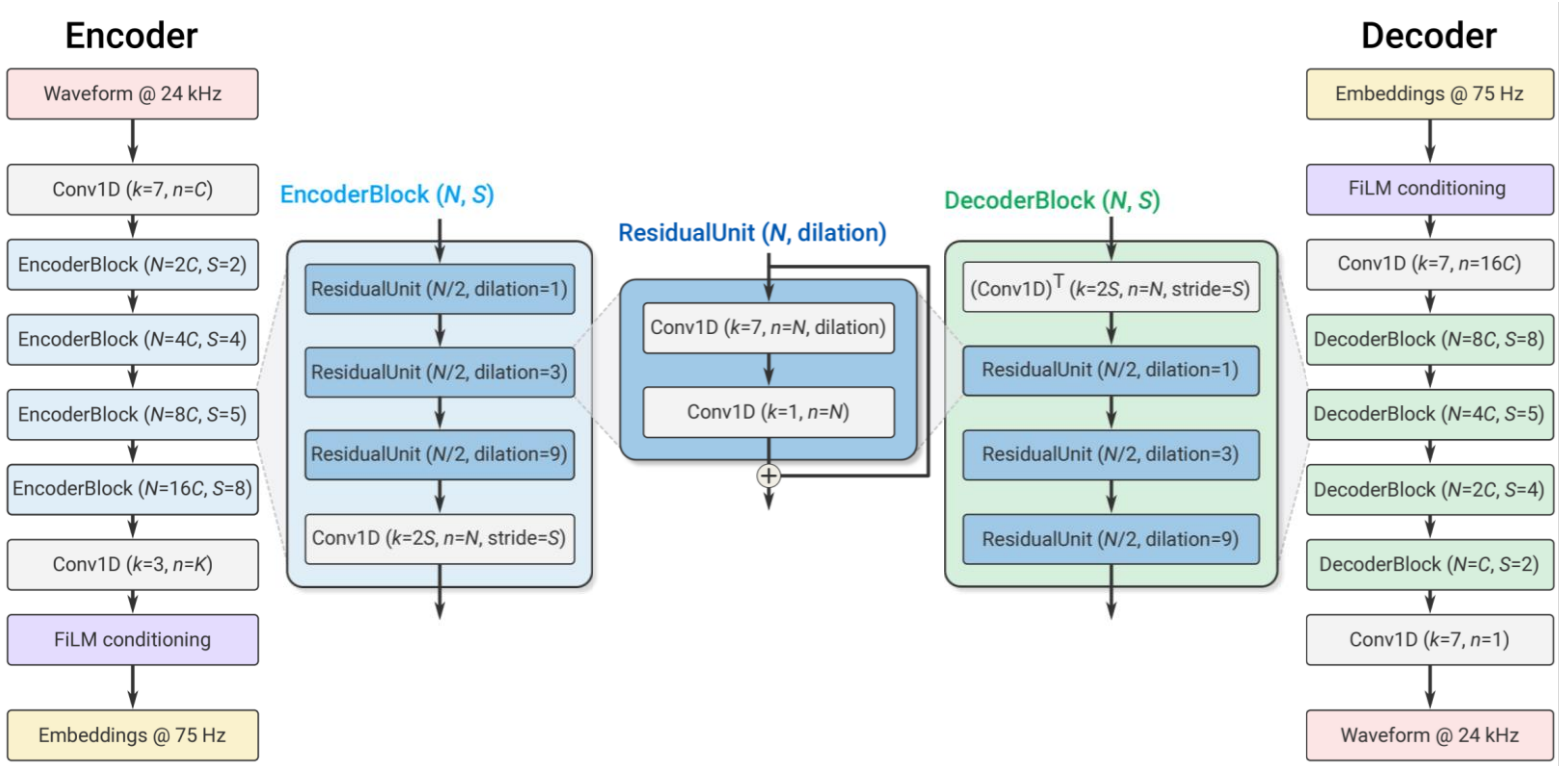
jik876.github.io/hifi-gan-demo

(Source: Kong et al., 2020)

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis," *NeurIPS*, 2020.

SoundStream (Zeghidour et al., 2021)

- **Fully-convolutional autoencoder** for audio

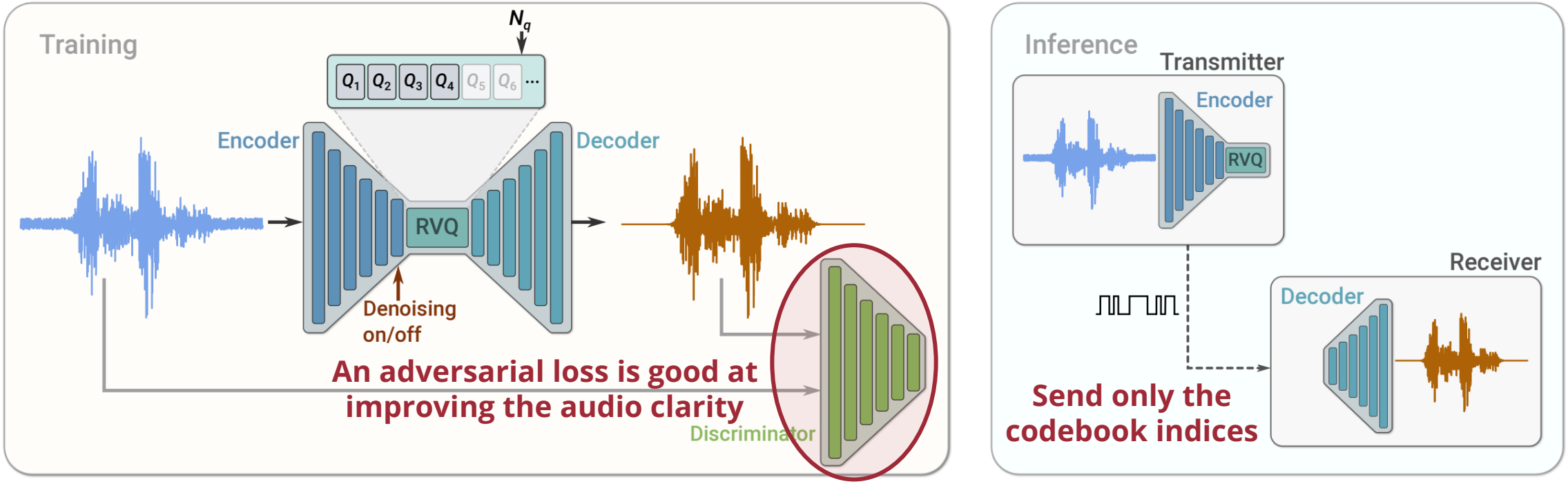


(Source: Zeghidour et al., 2021)

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi, "SoundStream: An End-to-End Neural Audio Codec," *TASLP*, 2021.

SoundStream (Zeghidour et al., 2021)

- **Fully-convolutional autoencoder** for audio



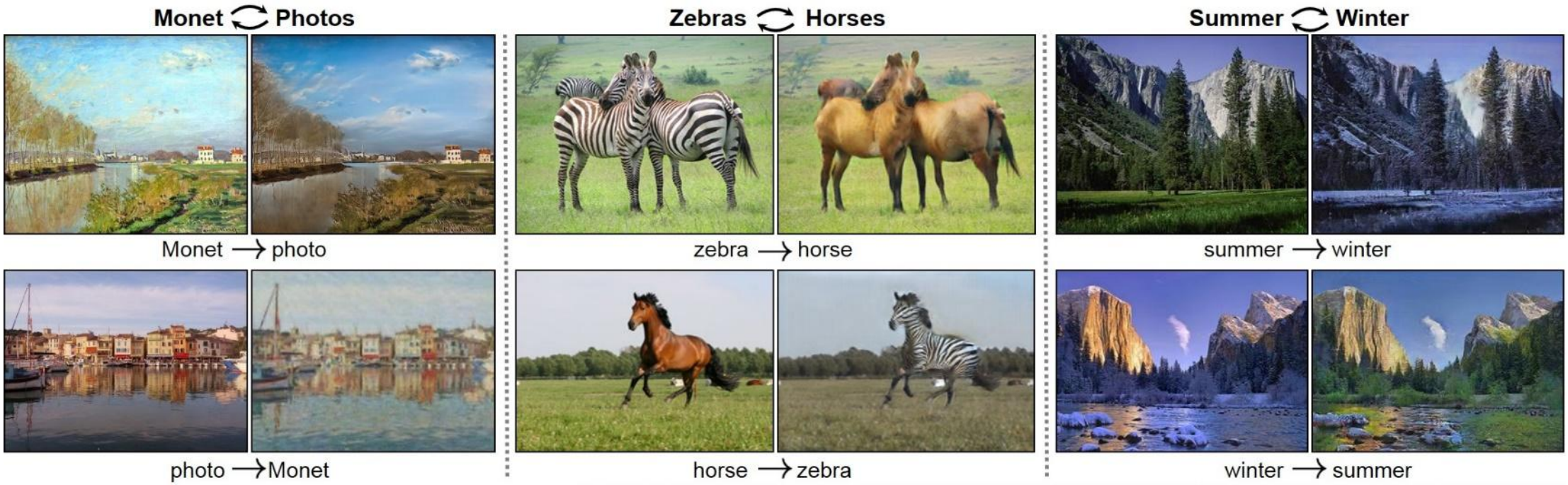
(Source: Zeghidour et al., 2021)

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi, "SoundStream: An End-to-End Neural Audio Codec," *TASLP*, 2021.

CycleGAN

CycleGAN: Examples

No paired data needed!



(Source: Zhu et al., 2017)

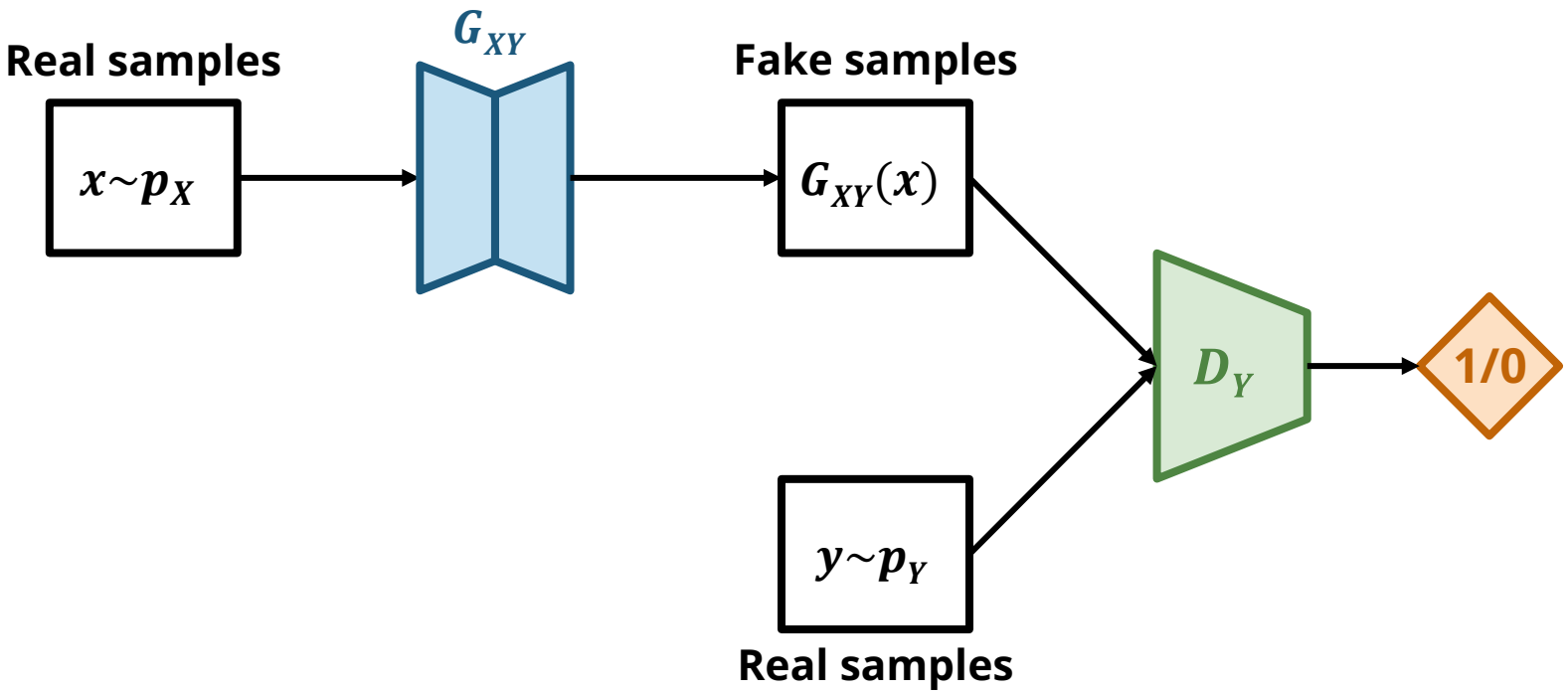
For example, we only need a collection of photos and a collection of Monet's paintings

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

CycleGAN: Goal

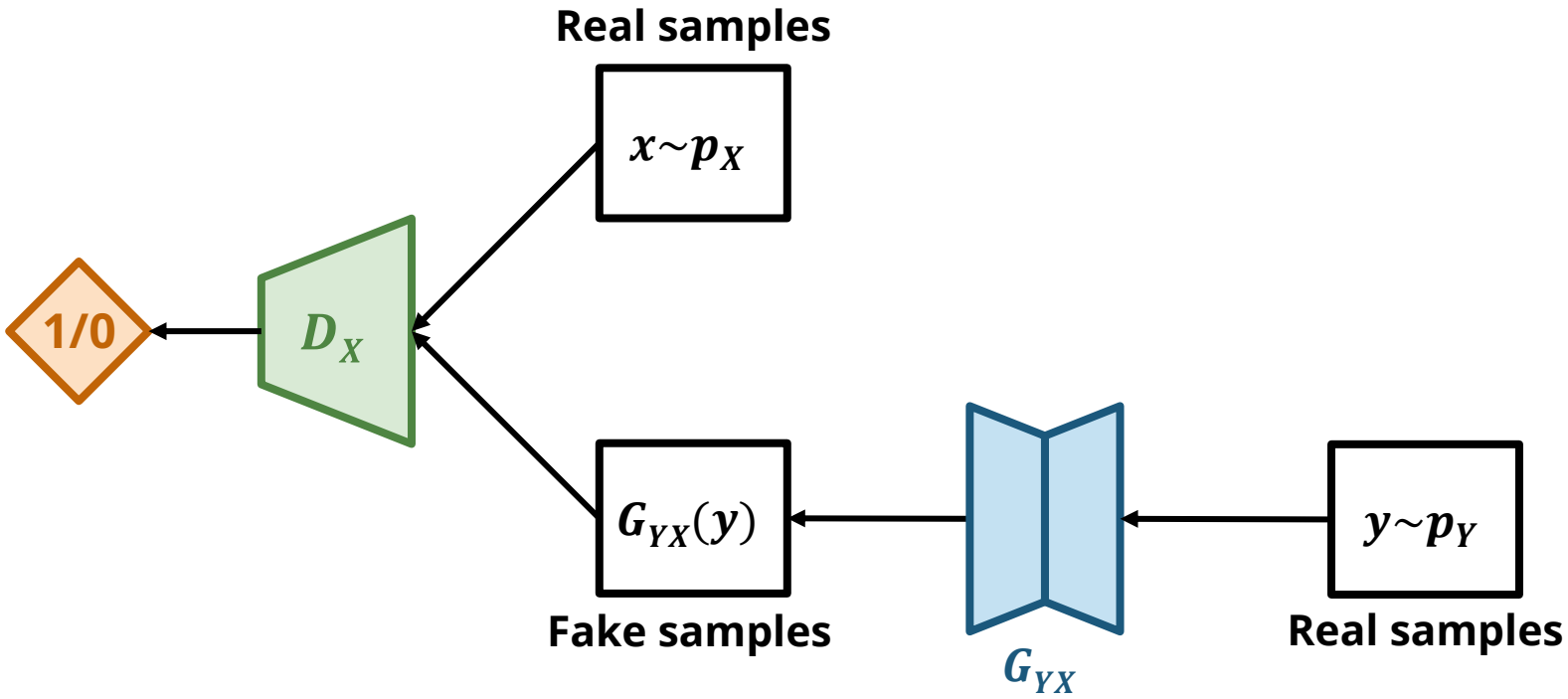
How can we learn the mapping without any paired data?

Cycle-consistent GAN (CycleGAN)



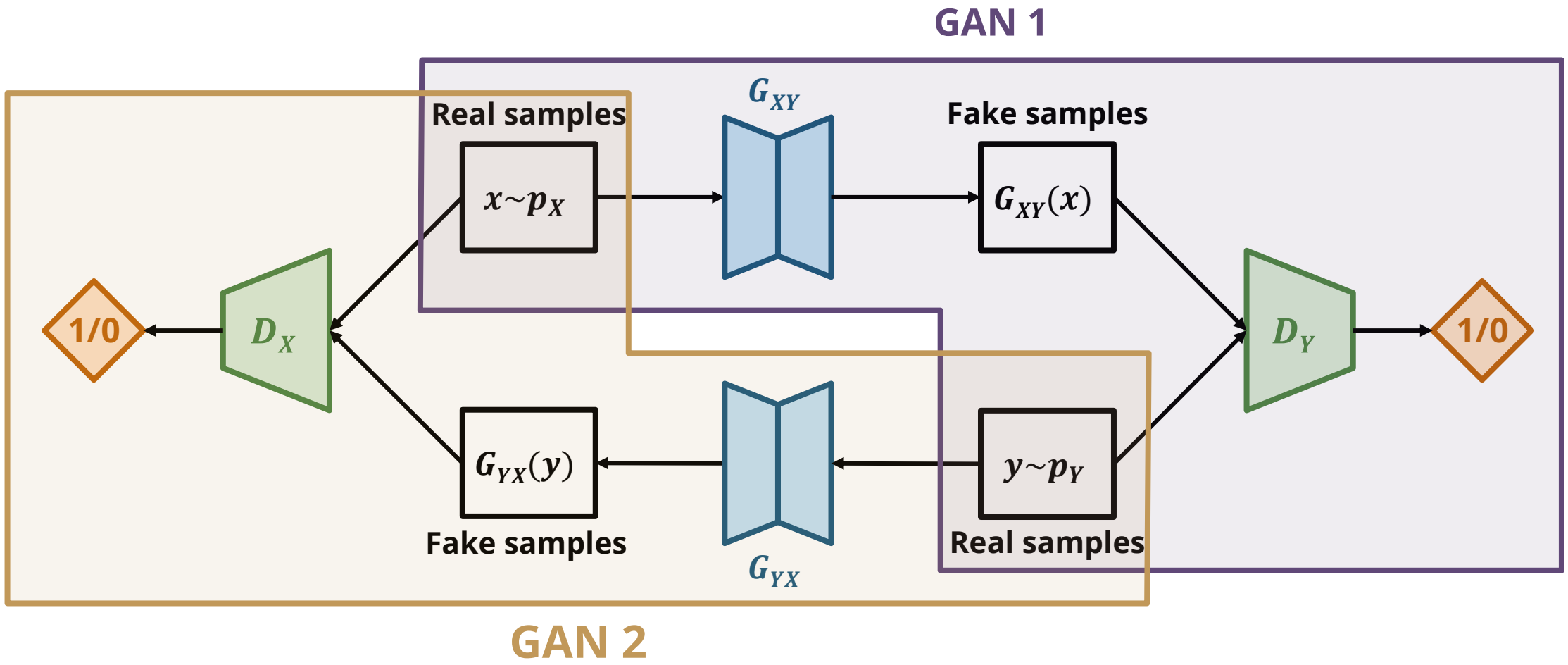
Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

Cycle-consistent GAN (CycleGAN)



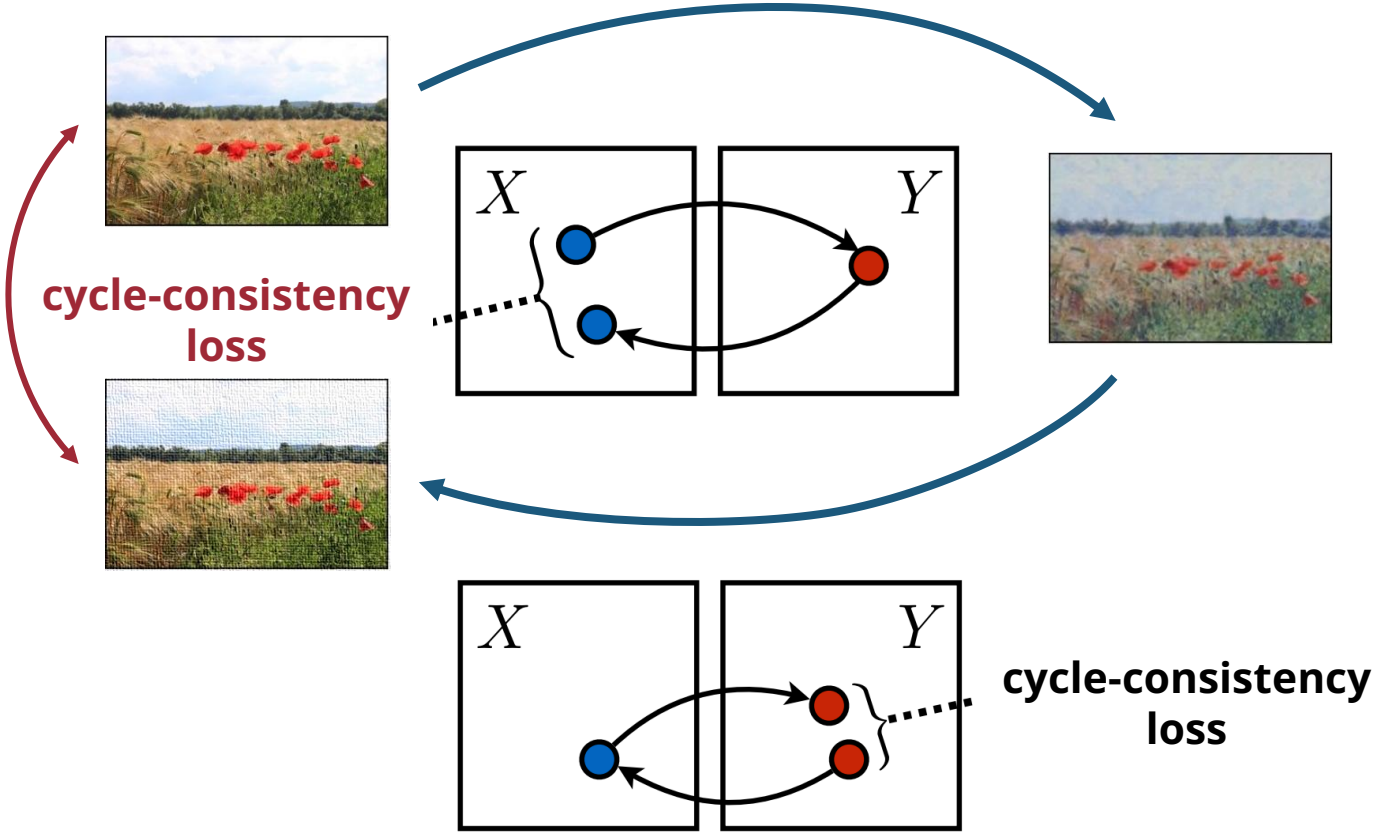
Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

Cycle-consistent GAN (CycleGAN)



Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

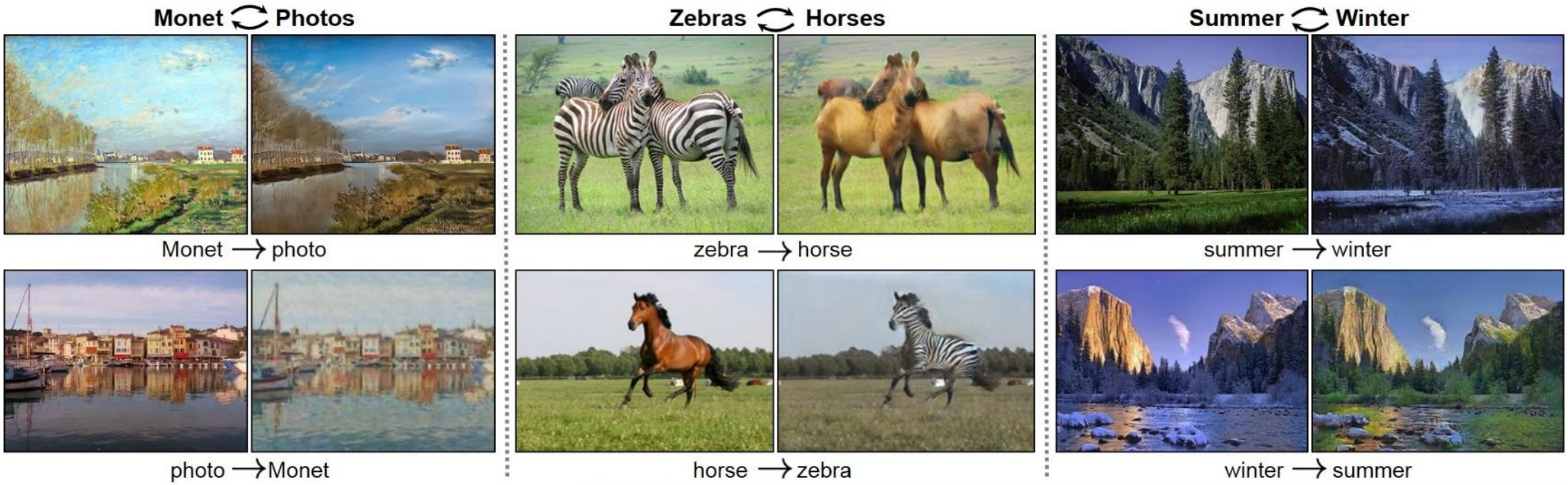
Cycle-consistency Loss



We only need unpaired samples in two domains

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

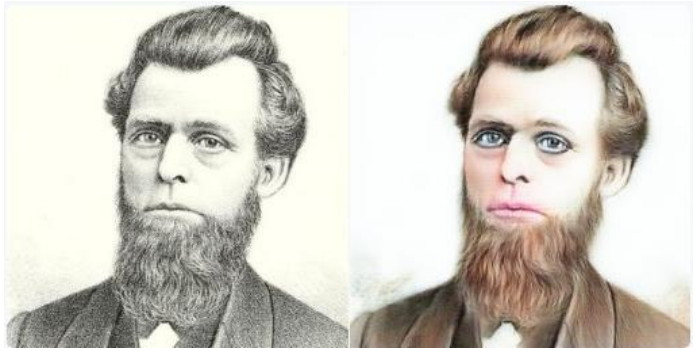
CycleGAN: Examples



(Source: Zhu et al., 2017)

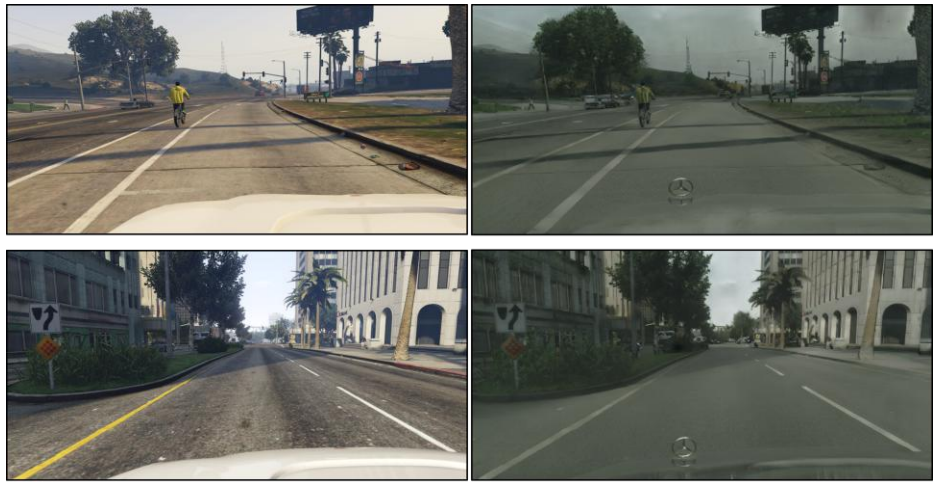
Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

CycleGAN: Examples



B&W → Color

Old city plan → Map → Satellite



GTA screenshot → Cityscape

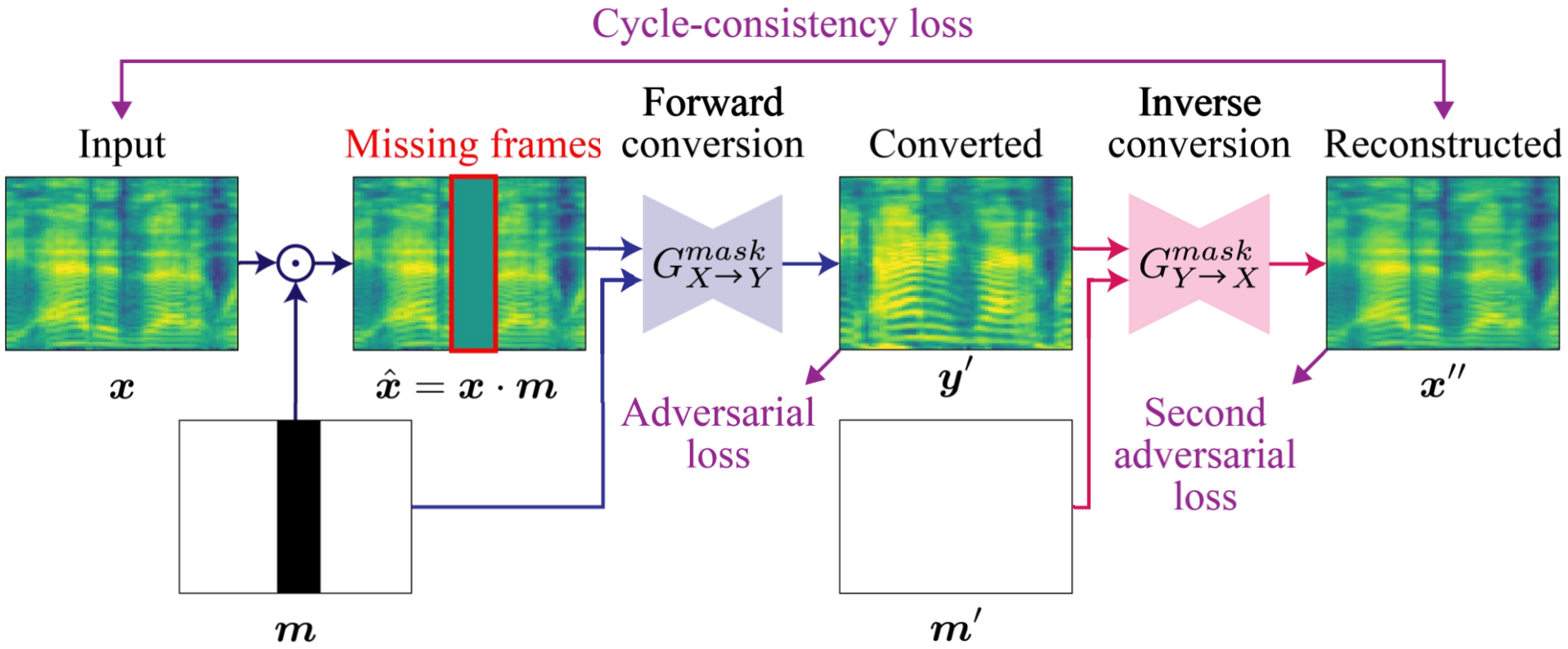
Dog → Cat

Cat → Dog

(Source: junyanz.github.io)

junyanz.github.io/CycleGAN

CycleGAN for Voice Conversion (Kaneko et al., 2021)



(Source: Kaneko et al., 2017)

Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "MaskCycleGAN-VC: Learning Non-parallel Voice Conversion with Filling in Frames," *ICASSP*, 2021.

CycleGAN for Voice Conversion (Kaneko et al., 2021)

	Source	Target	Converted	Ground truth
Female → male				
Male → female				

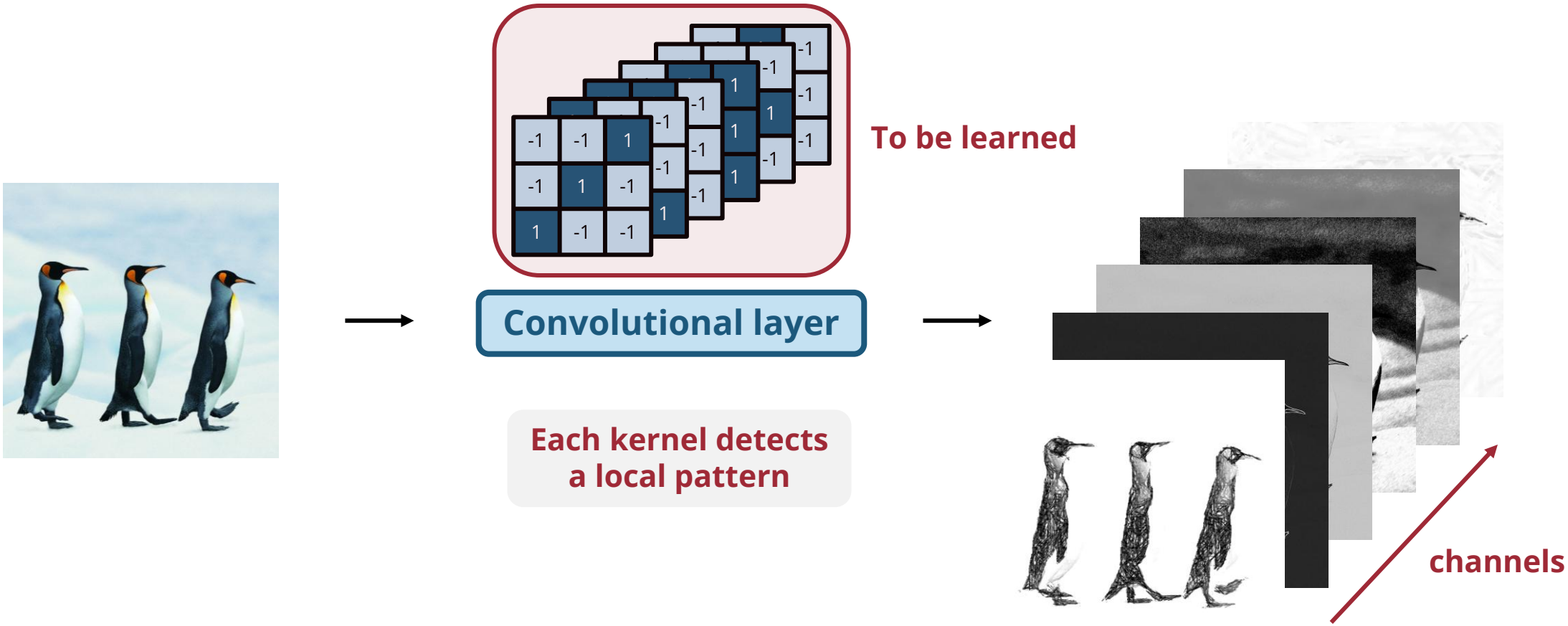
kecl.ntt.co.jp/people/kaneko.takuhiro/projects/maskcyclegan-vc/index.html

Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "MaskCycleGAN-VC: Learning Non-parallel Voice Conversion with Filling in Frames," *ICASSP*, 2021.

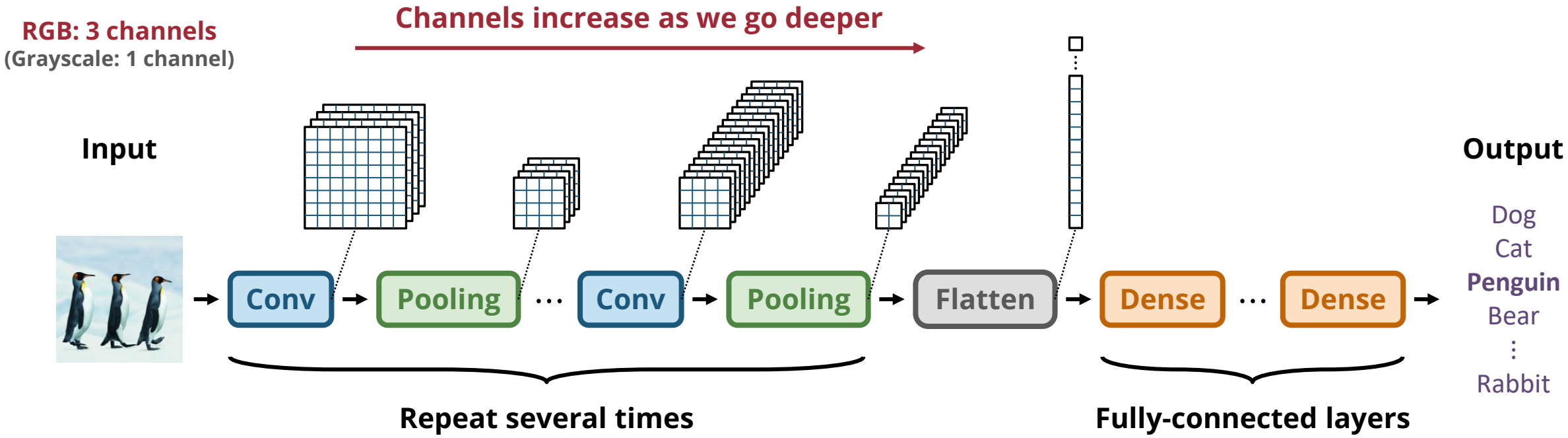
Recap

Convolutional Layer

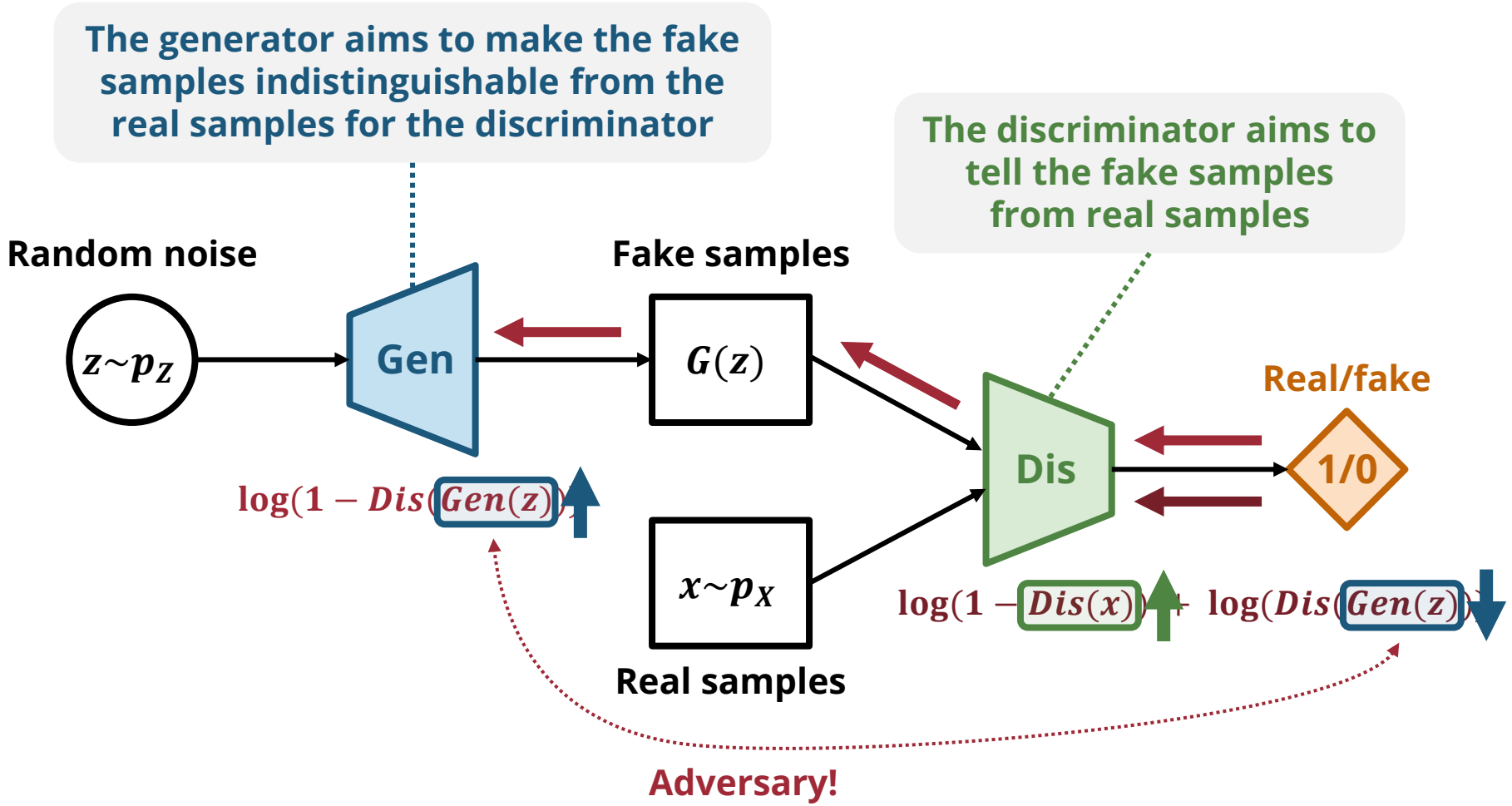
- A convolutional layer consists of many **learnable kernels** (channels)



Convolutional Neural Network (CNNs)

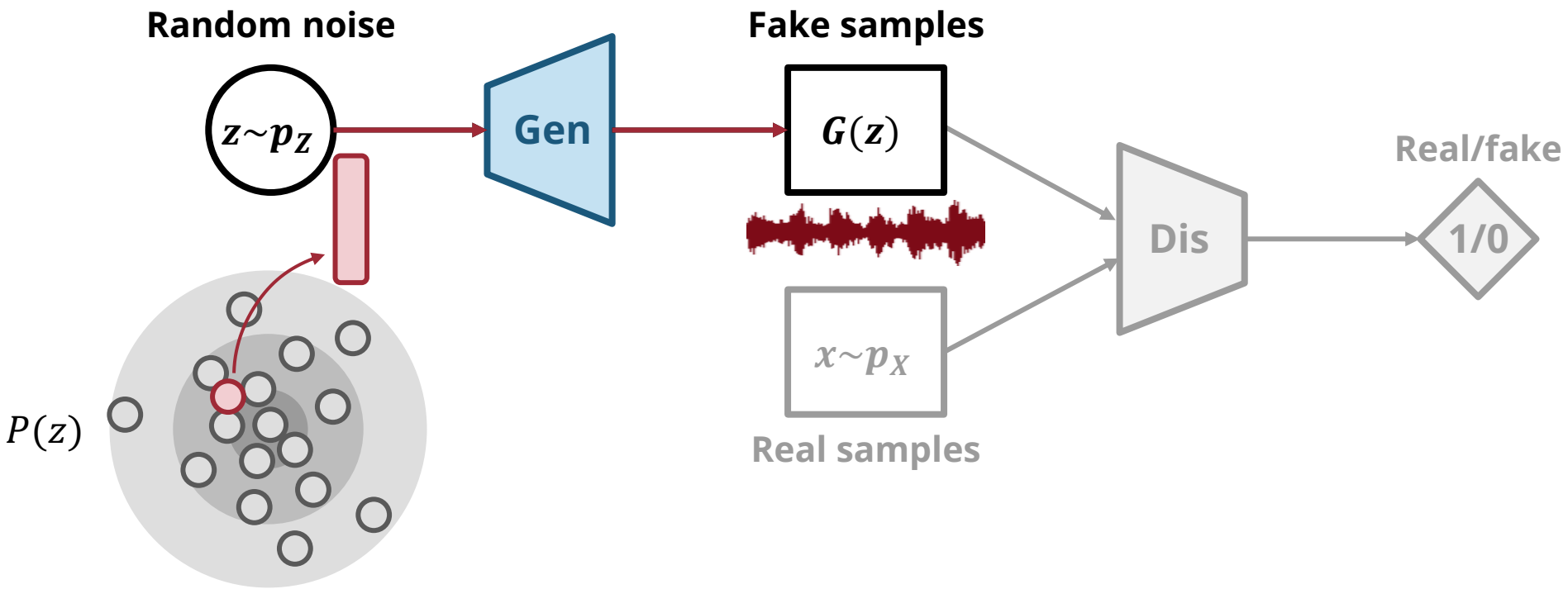


Generative Adversarial Nets (GANs): Training



Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Networks," *NeurIPS*, 2014.

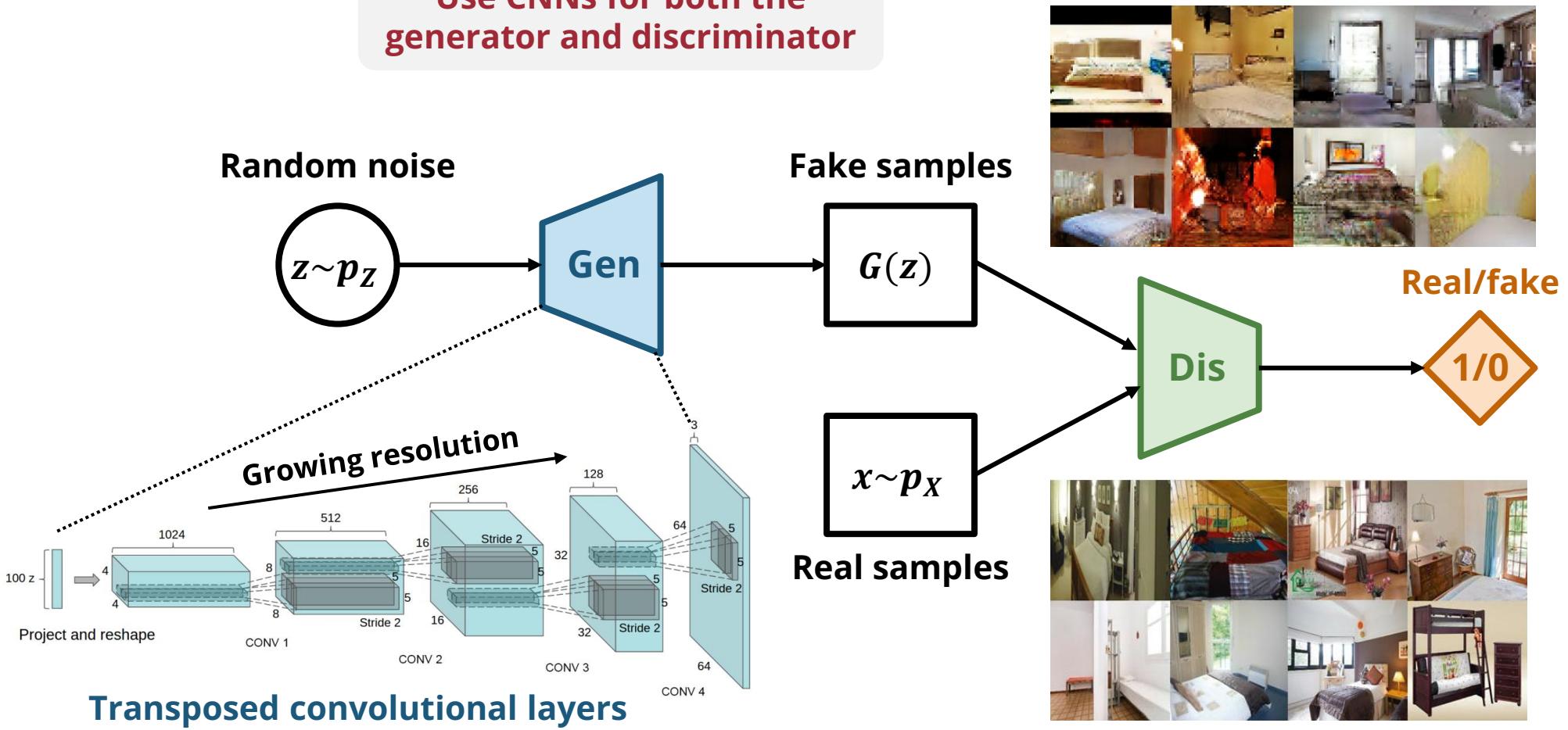
Generative Adversarial Nets (GANs): Generation



Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Networks," *NeurIPS*, 2014.

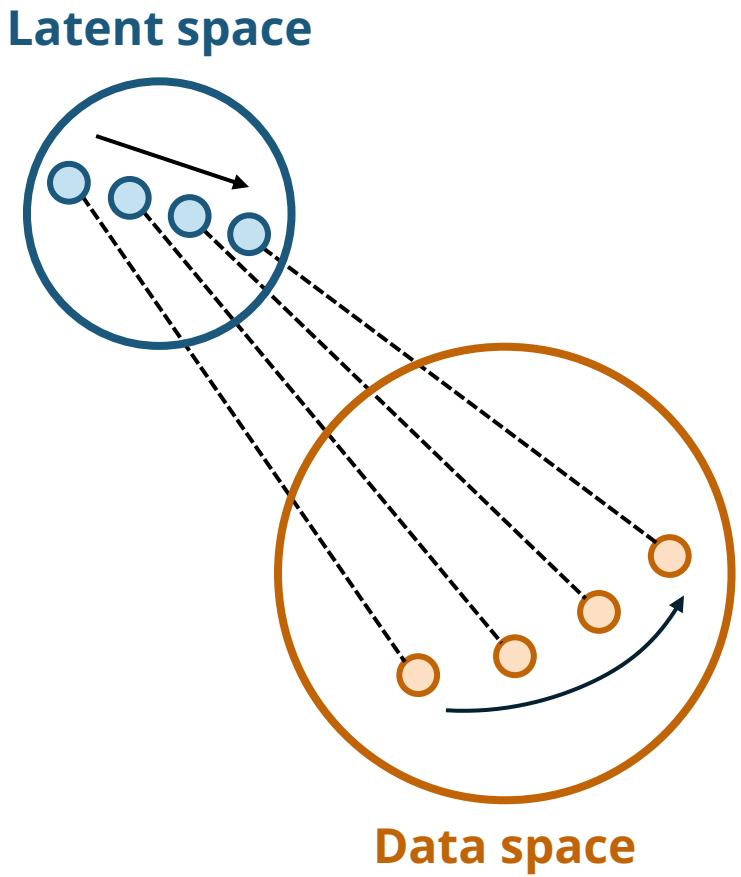
Deep Convolutional GANs (DCGANs) (Radford et al., 2014)

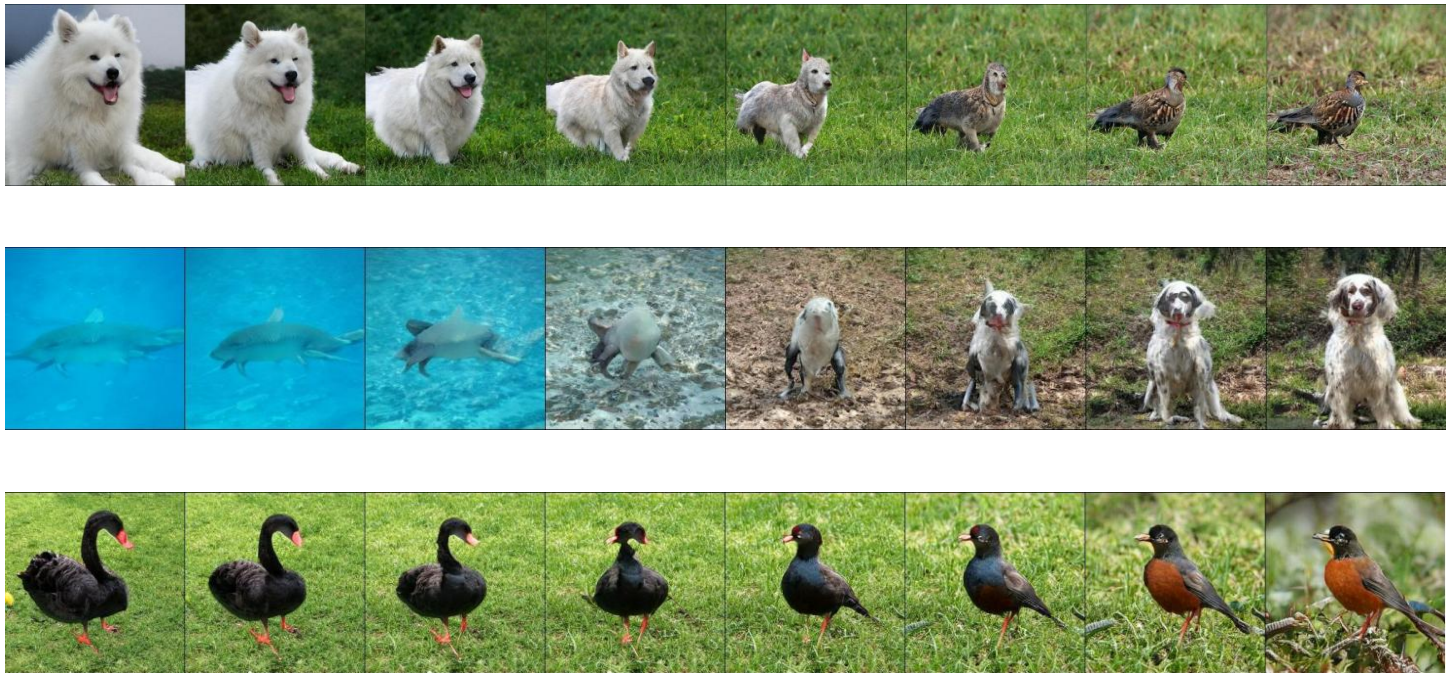
Use CNNs for both the generator and discriminator



Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *ICLR*, 2016.

Latent Space Interpolation of a GAN

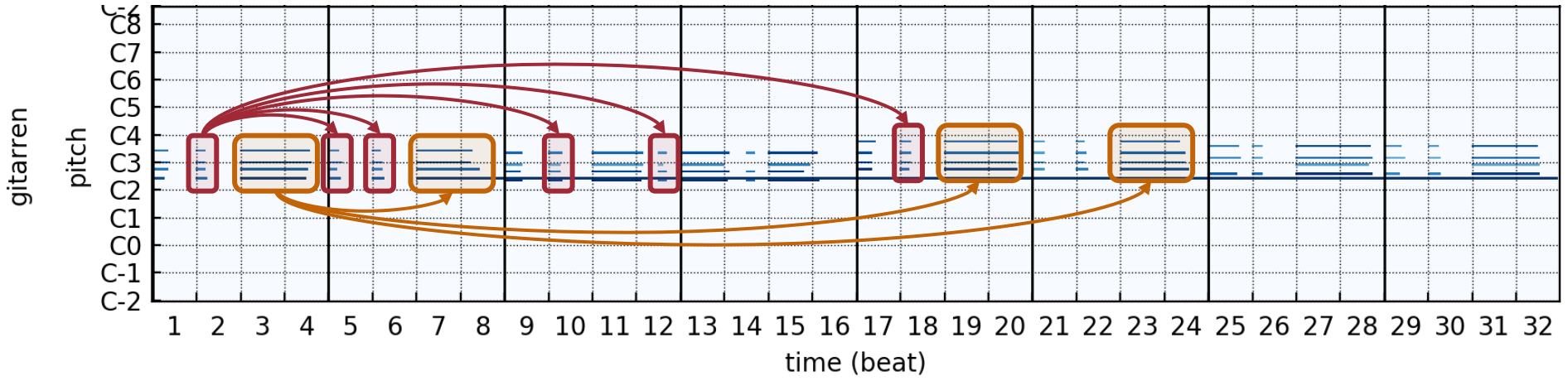




(Source: Brock et al., 2019)

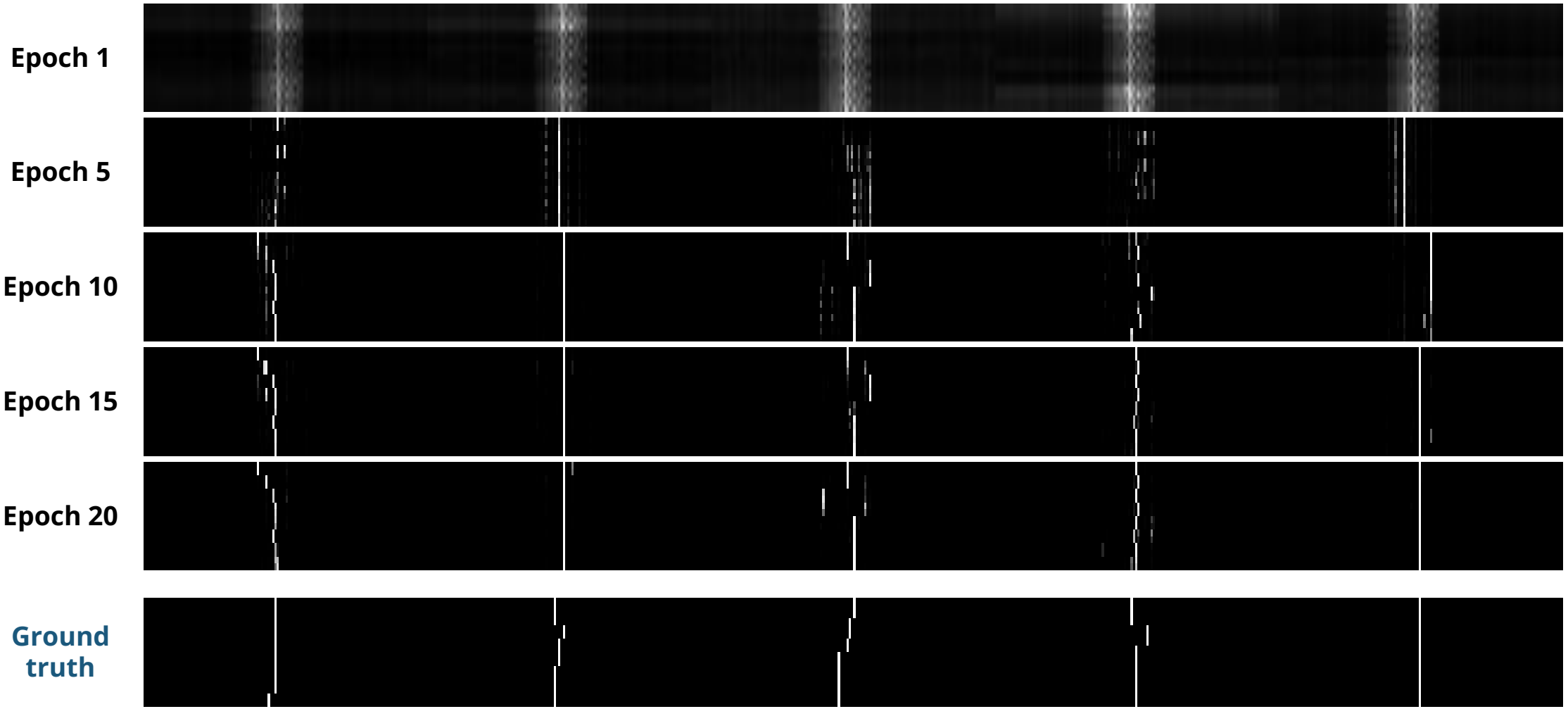
Andrew Brock, Jeff Donahue, and Karen Simonyan, "Large Scale GAN Training for High Fidelity Natural Image Synthesis," *ICLR*, 2019.

Why Piano Rolls?



Many musical patterns like melodies, chords, scales and arpeggios are translational invariant in the temporal and pitch axes

MidiNet (Yang et al., 2017)

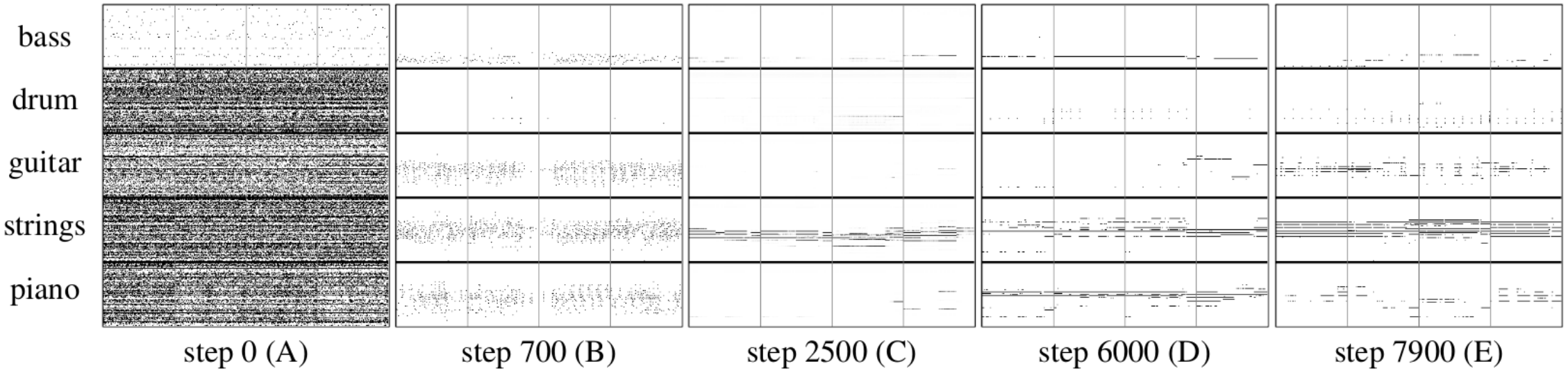


(Source: Yang et al., 2017)

Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang, "MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation," *ISMIR*, 2017.

MuseGAN (Dong et al., 2018)

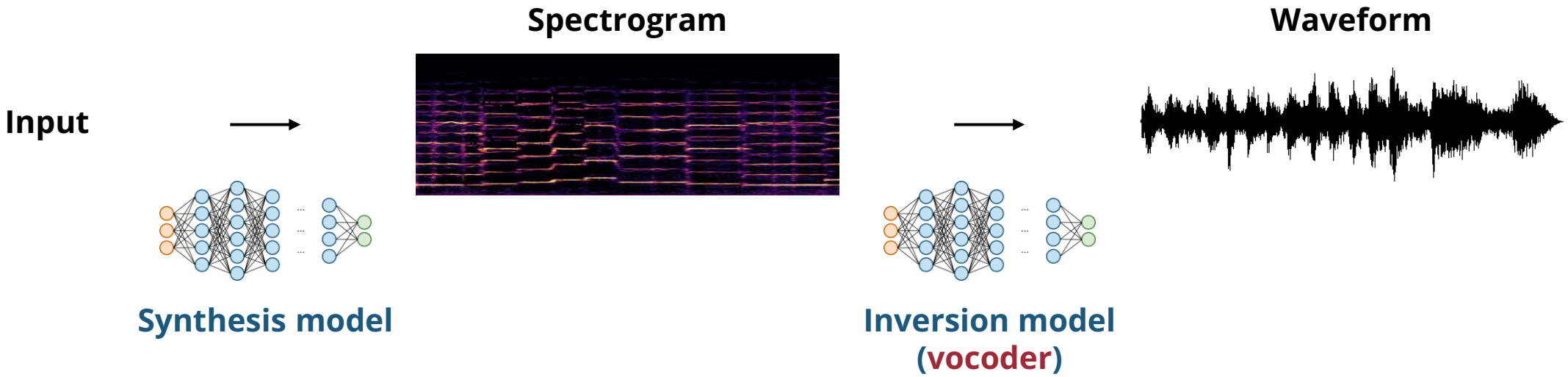
The generator improves over time



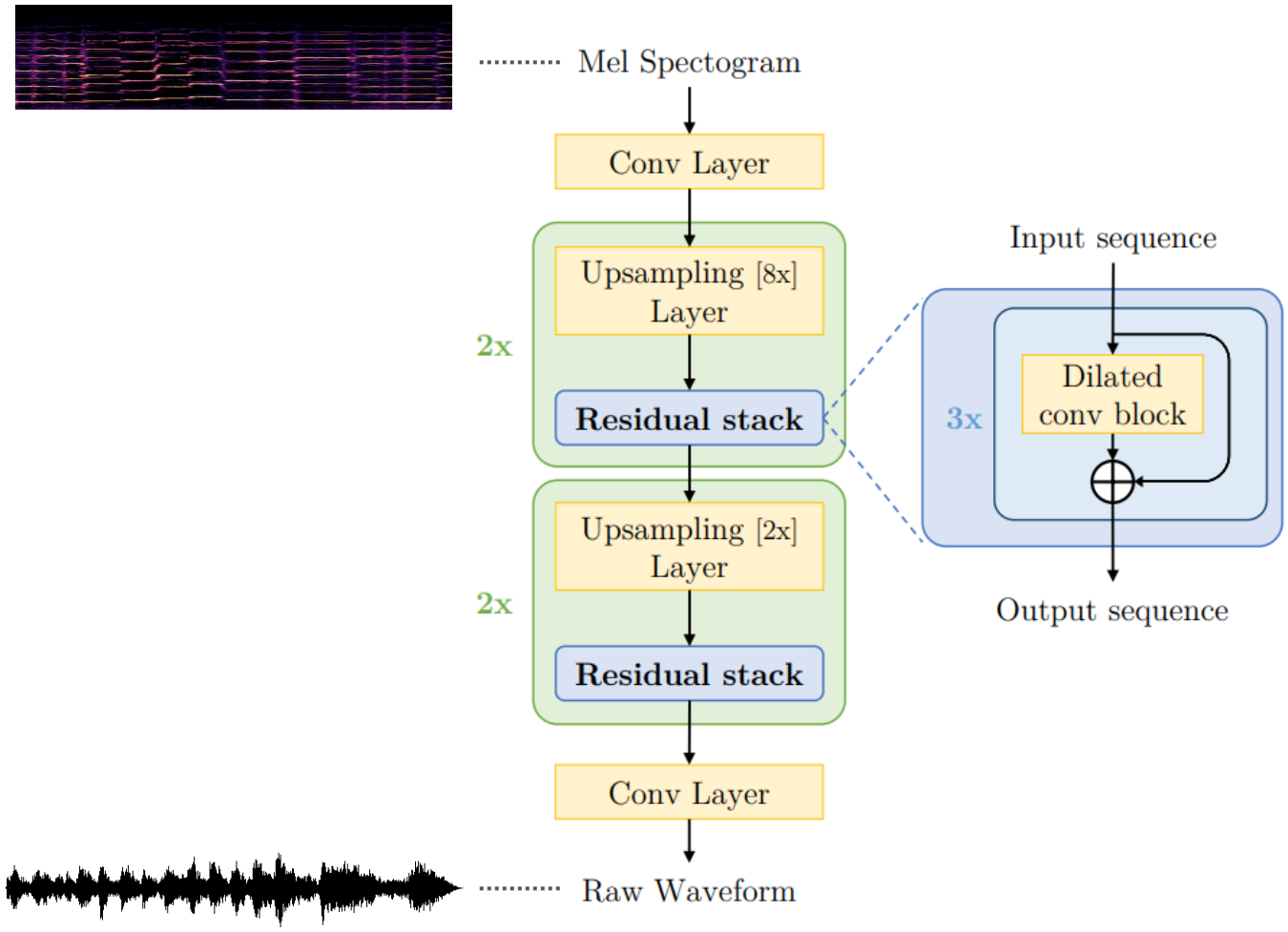
(Source: Dong et al., 2018)

Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," *AAAI*, 2018.

Frequency-domain Audio Synthesis



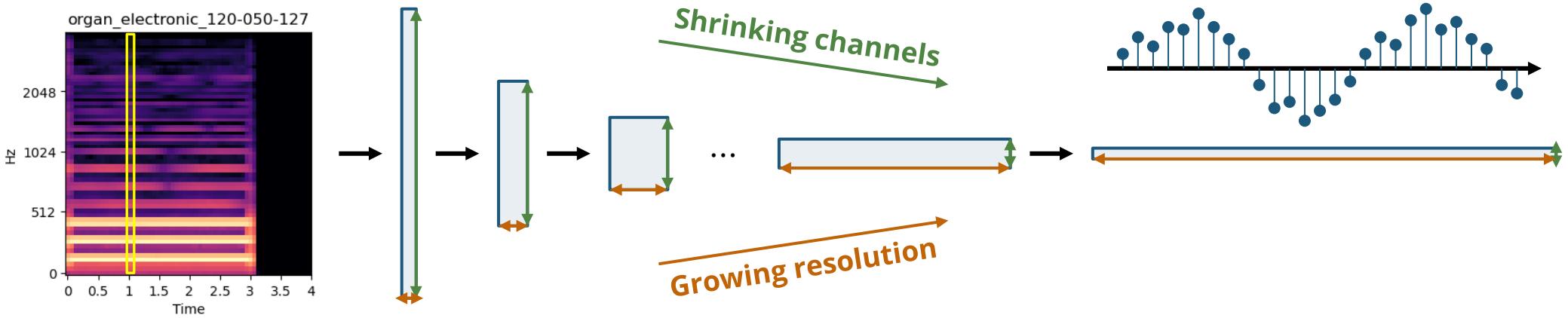
MelGAN (Kumar et al., 2019)



(Source: Kumar et al., 2019)

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville, "MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis," *NeurIPS*, 2019.

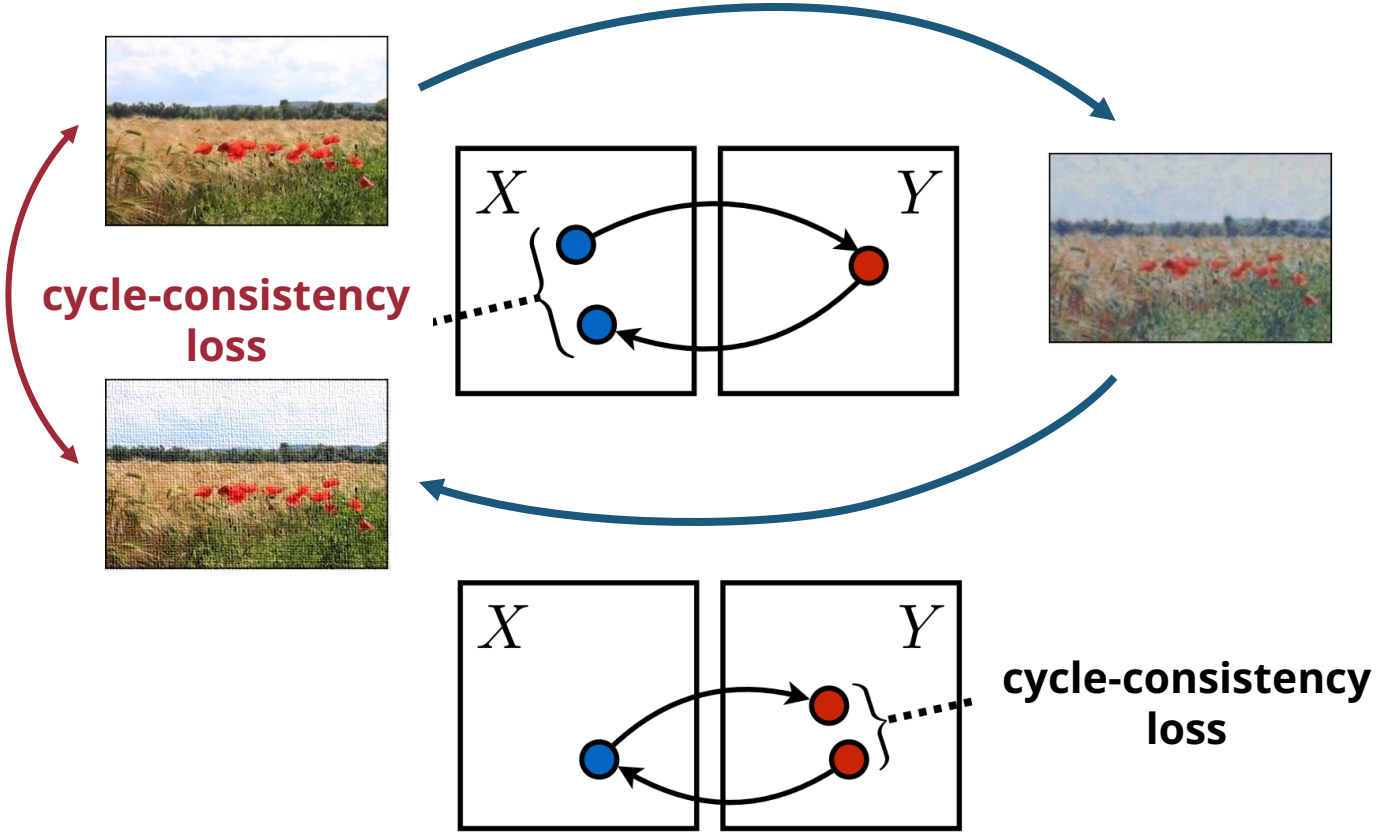
Upsampling for Vocoders



CycleGAN: Goal

How can we learn the mapping without any paired data?

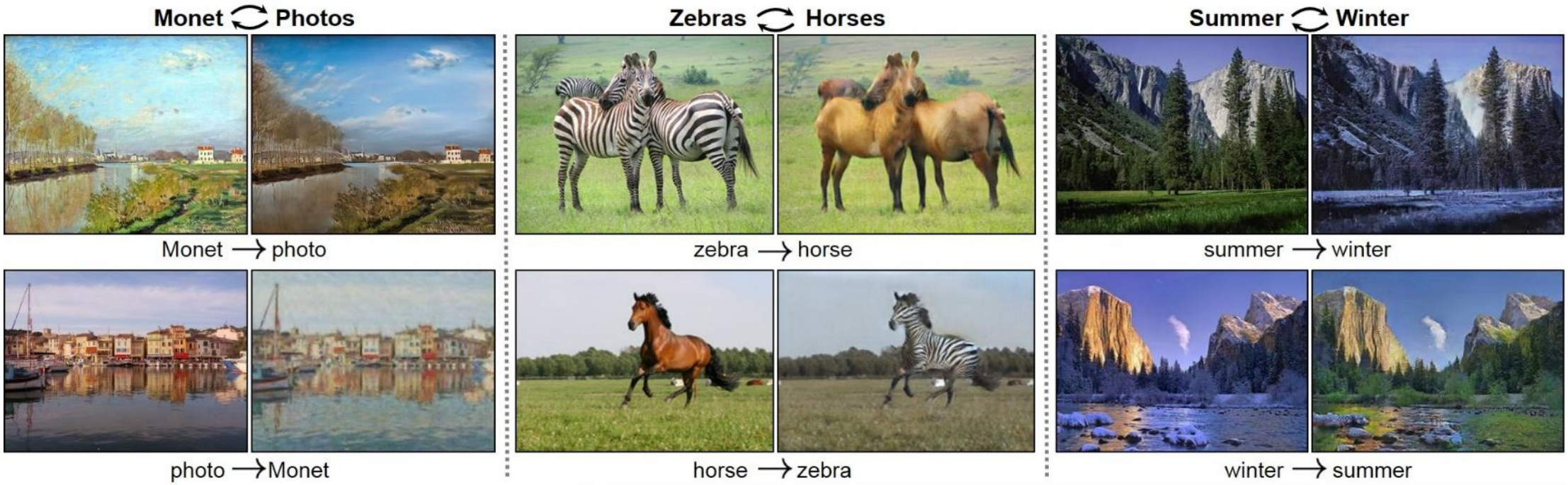
Cycle-consistency Loss



We only need unpaired samples in two domains

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

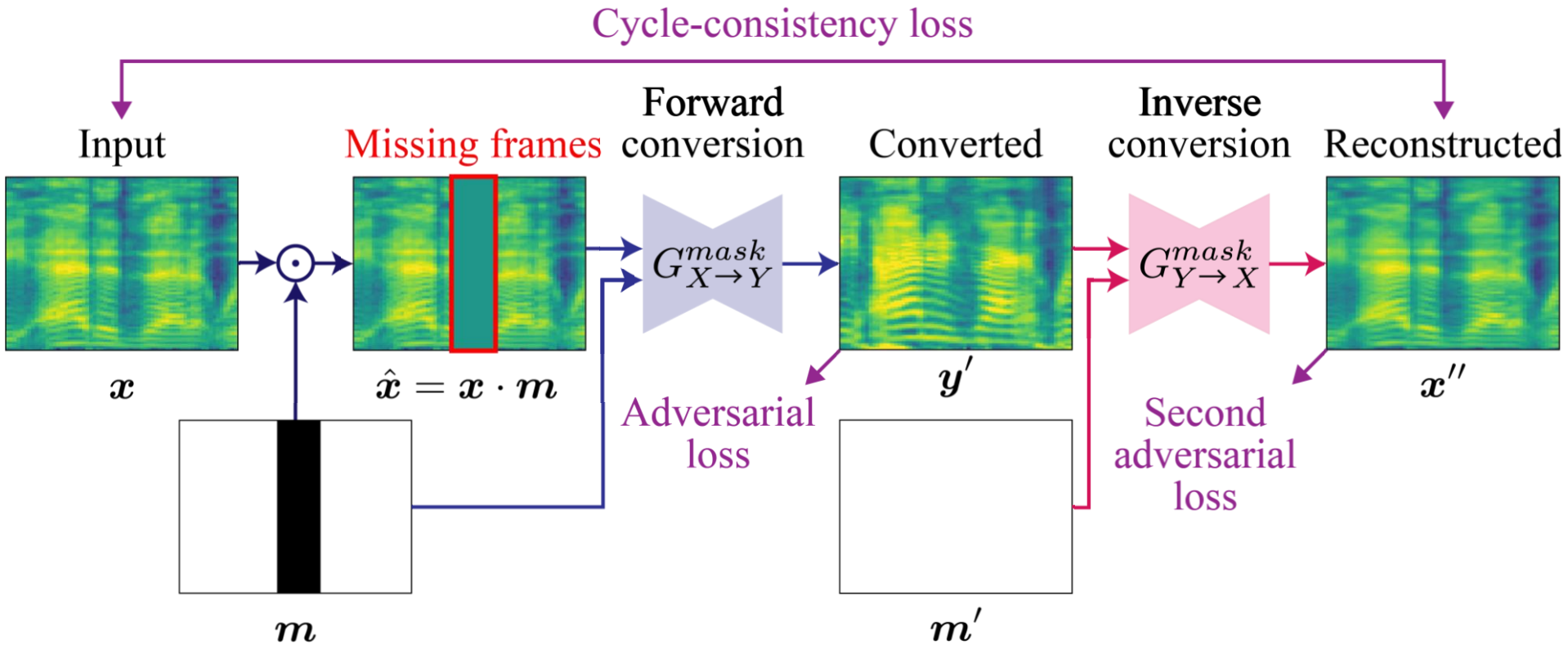
CycleGAN: Examples



(Source: Zhu et al., 2017)

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *ICCV*, 2017.

CycleGAN for Voice Conversion (Kaneko et al., 2021)

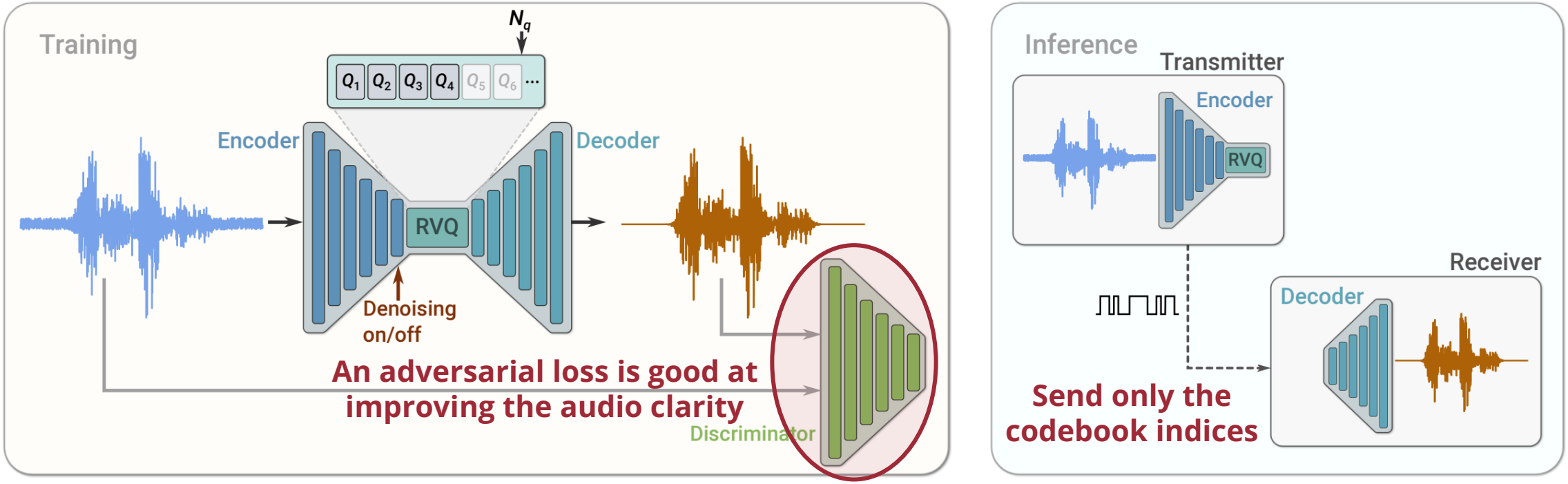


(Source: Kaneko et al., 2017)

Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "MaskCycleGAN-VC: Learning Non-parallel Voice Conversion with Filling in Frames," *ICASSP*, 2021.

SoundStream (Zeghidour et al., 2021)

- **Fully-convolutional autoencoder** for audio



(Source: Zeghidour et al., 2021)

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi, "SoundStream: An End-to-End Neural Audio Codec," *TASLP*, 2021.

Next Lecture

Diffusion Models

(Source: Ho et al., 2020)