

Generative Künstliche Intelligenz für Video und Audio Technologien

Digitale AV Technik, MIB 5

Übersicht

- Was ist generative KI?
- Welche Arten von generativen Modellen gibt es?
 - Generative Adversarial Networks (GANs)
 - Autoencoder
 - Stable Diffusion
- Bezug zu Encoding und Decoding

Generative Adversarial Networks (GANs)

- Ein maschinelles Lernmodell mit zwei konkurrierenden Netzwerken:
 - **Generator**
 - **Diskriminator**
- Entwickelt von [Ian Goodfellow et al. \(NeurIPS2014\)](#)
- Anwendung: Bildgenerierung, Stiltransfer, Datenaugmentation, u.v.m.

Grundidee: Generator vs. Diskriminator

1. Generator

- Ziel: Realistisch aussehende Daten erzeugen.
- Erzeugt zufällige Daten zu Beginn.
- Lernt, echte Daten zu imitieren.

2. Diskriminator

- Ziel: Zwischen echten und generierten Daten unterscheiden.
- Prüft die Arbeit des Generators.
- Lernt, echte Daten von generierten zu unterscheiden.

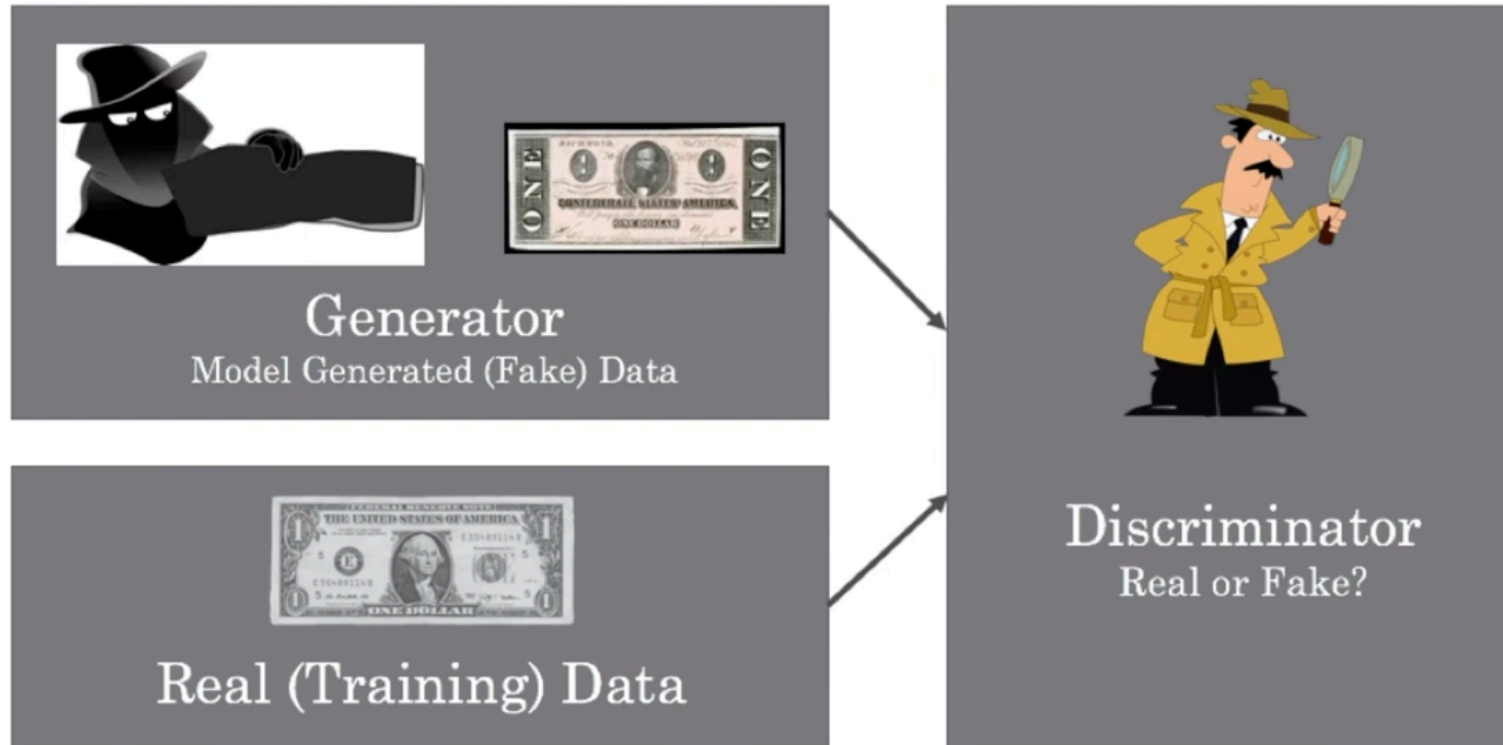
Spiel zwischen Generator und Diskriminator

- **Generator:** Täuscht den Diskriminator mit besseren Fälschungen.
- **Diskriminator:** Lernt, die Täuschungen zu erkennen.

Ergebnis: Beide Netzwerke verbessern sich stetig.

Ziel: Der Generator erzeugt Daten, die ununterscheidbar von echten Daten sind.

Beispiel mit Illustration



Quelle: [Gentle Intro to Generative Adversarial Networks - Part 1 \(GANs\)](#) by WelcomeAIOverlords (YouTube)

Analogie: Künstler vs. Kritiker 🎨🔍

- Der **Generator** ist ein Künstler:
 - Versucht, Kunstwerke zu erstellen, die wie echte aussehen.
- Der **Diskriminator** ist ein Kunstkritiker:
 - Erkennt echte Kunstwerke und entlarvt Fälschungen.

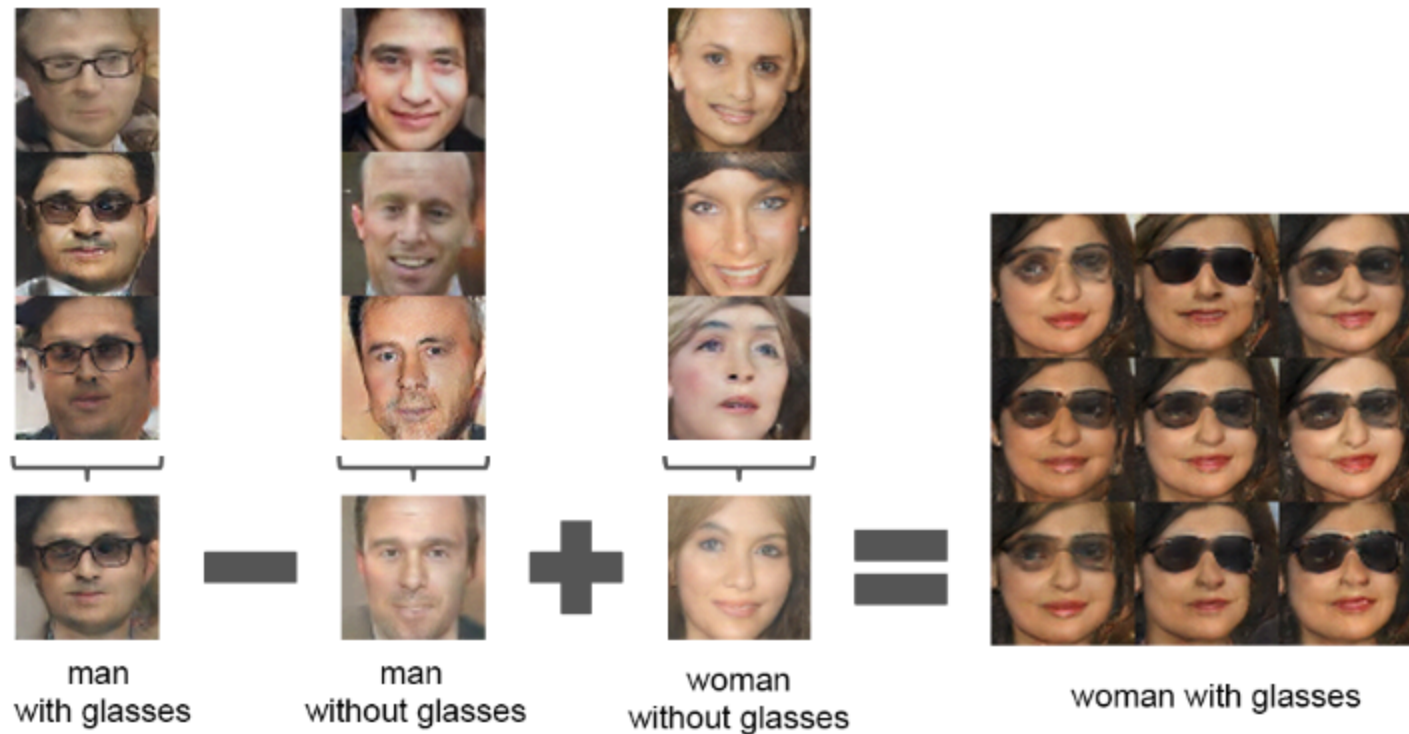
Training:

- Der Künstler wird immer besser, bis seine Werke nicht mehr von echten zu unterscheiden sind.

Ende des Trainings

- Der Diskriminator kann nur noch zufällig zwischen echten und generierten Daten unterscheiden (50 % Genauigkeit).
- Die vom Generator erzeugten Daten sind nahezu perfekt.
- Es kann sehr lange dauern und es hängt stark von den Trainingsdaten ab.

Arithmetik



- Quelle: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, *Alec Radford, Luke Metz, Soumith Chintala*, ICLR 2016 [pdf](#)

Varianten

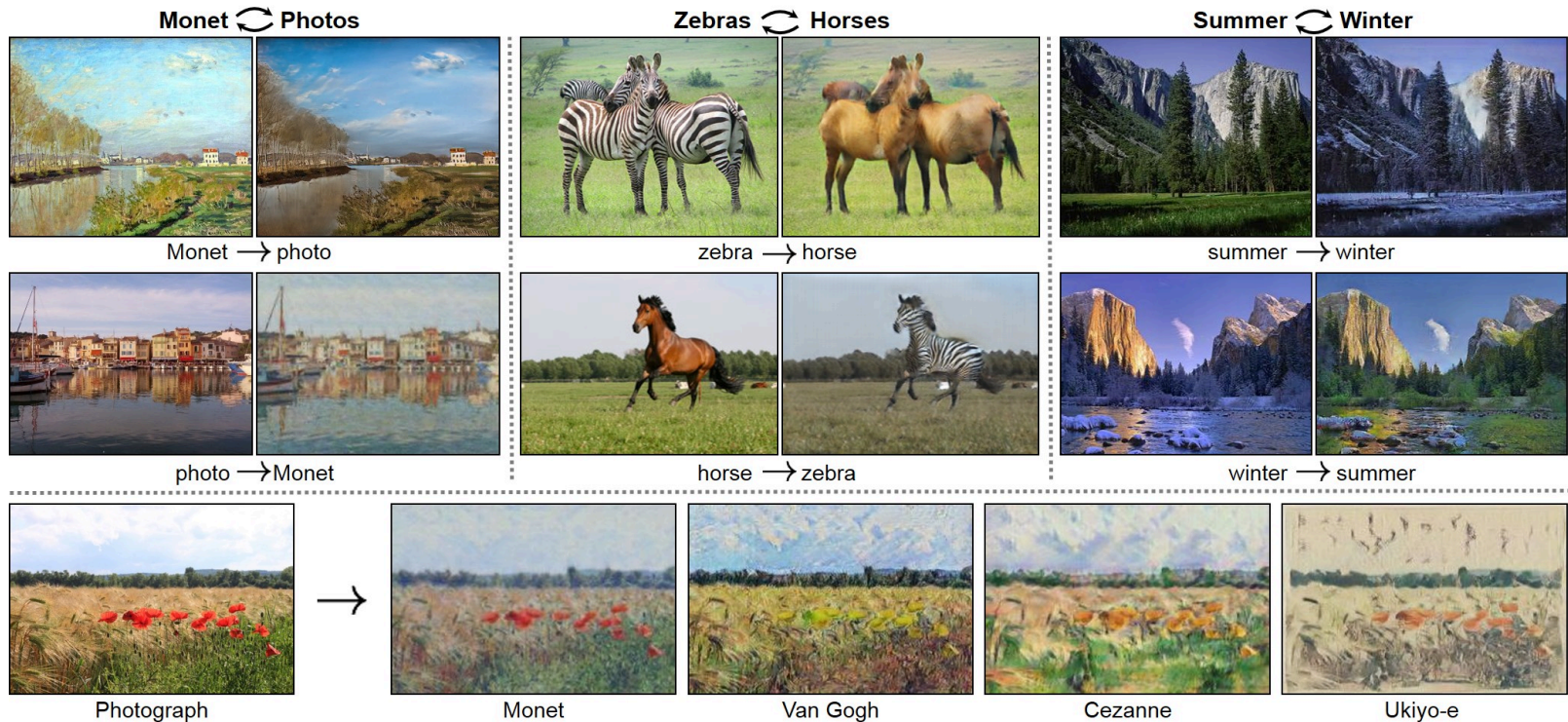
- StyleGAN, StyleGAN2 Demo
- CycleGAN
- 3D GANs
- Audio GAN

StyleGAN



A Style-Based Generator Architecture for Generative Adversarial Networks, *Tero Karras (NVIDIA), Samuli Laine (NVIDIA), Timo Aila (NVIDIA)*, CVPR 2019

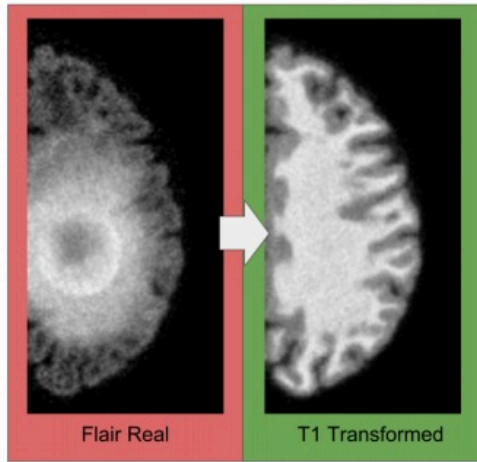
CycleGAN



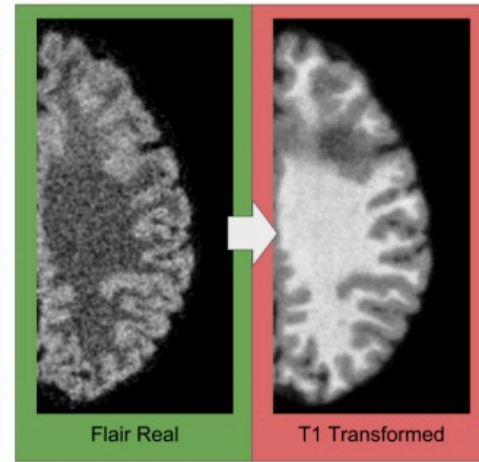
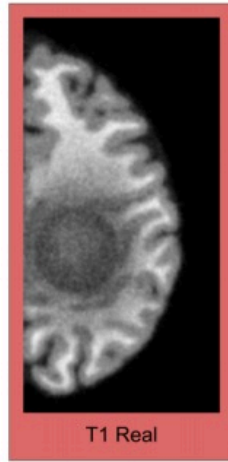
Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks,

Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros, ICCV 2017

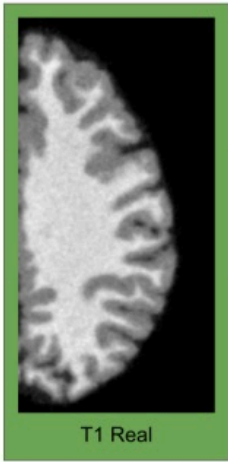
Ein Hinweis zu CycleGAN



(a) A translation removing tumors



(b) A translation adding tumors



"CycleGAN should only be used with great care and calibration in domains where critical decisions are to be taken based on its output."

Quelle: [CycleGAN Projektseite](#)

3D GANs

- Idee: Bilder aus mehreren Perspektiven des selben Objekts erzeugen und dann 3D Rekonstruktion anwenden
- Stand der Technik 2021 mit [Video](#)
- Nachteil: Training extrem aufwendig

3D GANs (Voxel-basiert)

Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling

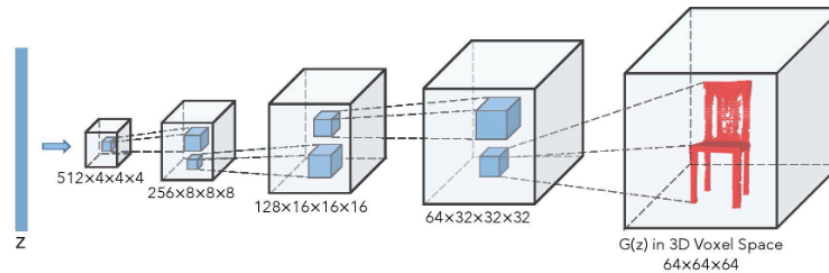


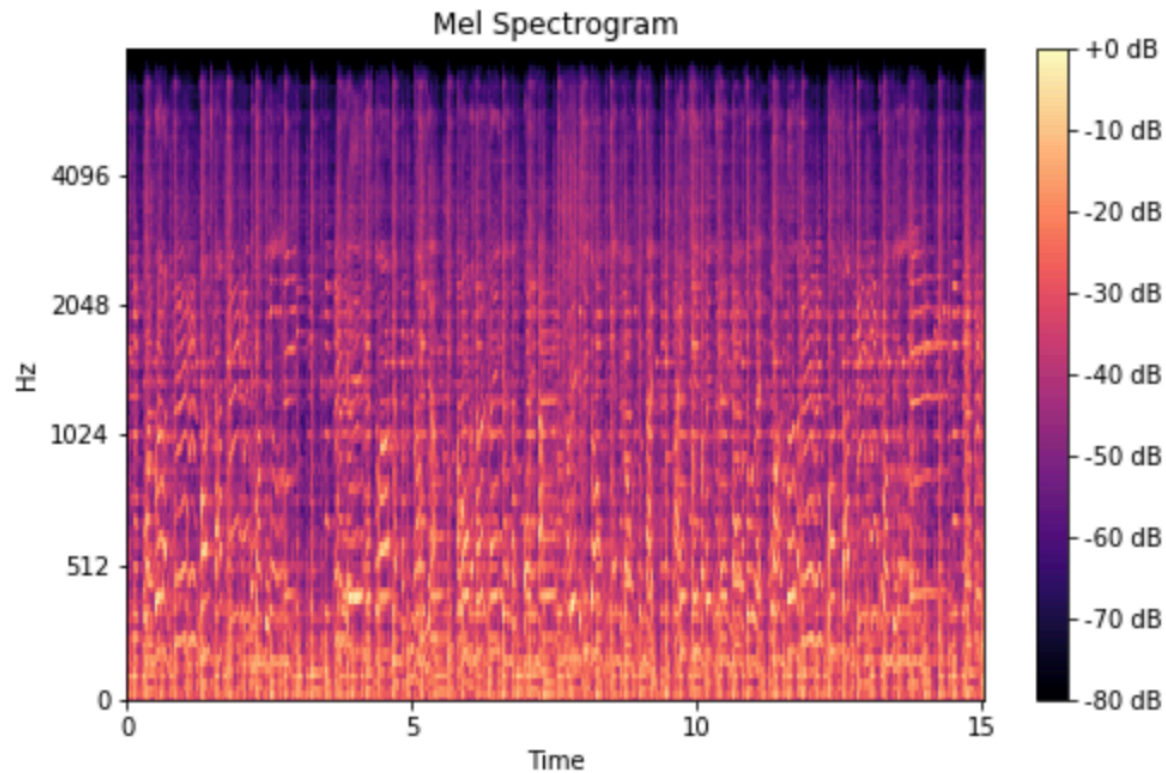
Figure 1: The generator of 3D Generative Adversarial Networks (3D-GAN)



Figure 2: Shapes synthesized by 3D-GAN

Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling, Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T. Freeman, and Joshua B. Tenenbaum, NeurIPS 2016

Audio GANs (früherer Ansatz)



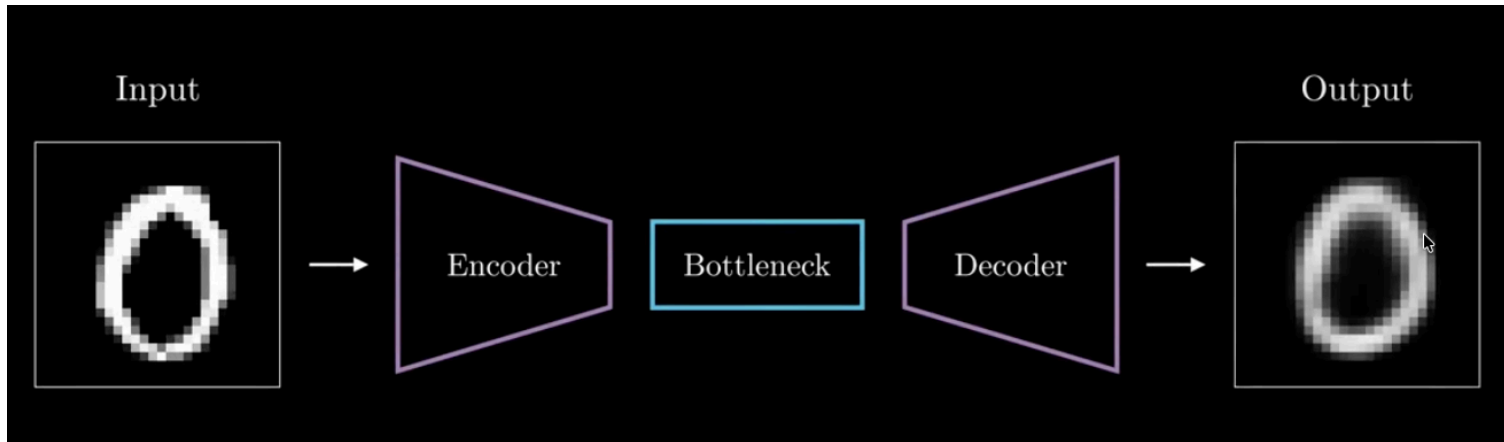
Quelle: [Understanding the Mel Spectrogram by Leland Roberts](#)

AudioGANs (Beispiele)

Stand der Technik 2021 mit Hörbeispielen:

[Unofficial Parallel WaveGAN implementation demo](#)

Autoencoder (AE)



Quelle: [Autoencoders | Deep Learning Animated by Deepia](#)
(Thibaut Modrzyk)

Neurales Netzwerk, das Eingabedaten in eine niedrigdimensionale Darstellung (Encoding) komprimiert und dann die Originaldaten aus dieser Darstellung rekonstruiert (Decoding).

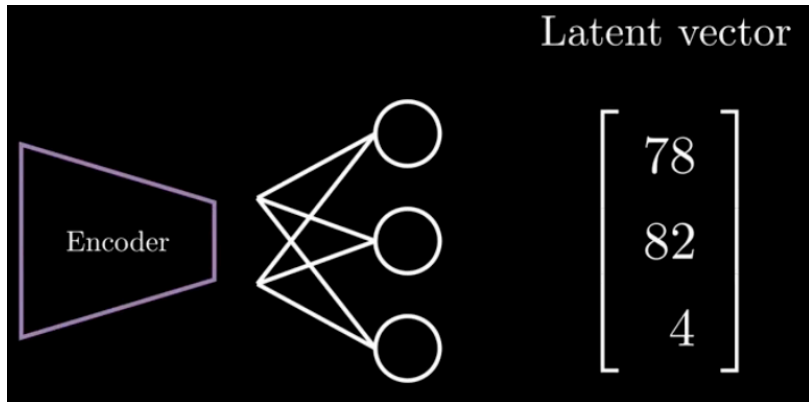
Autoencoder (AE): Aufbau



Quelle: [Autoencoders | Deep Learning Animated by Deepia](#)
(Thibaut Modrzyk)

Besteht aus einem Encoder- und einem Decoder-Netzwerk.
Dazwischen liegt der sogenannte "Bottleneck", der die
komprimierte Repräsentation enthält.

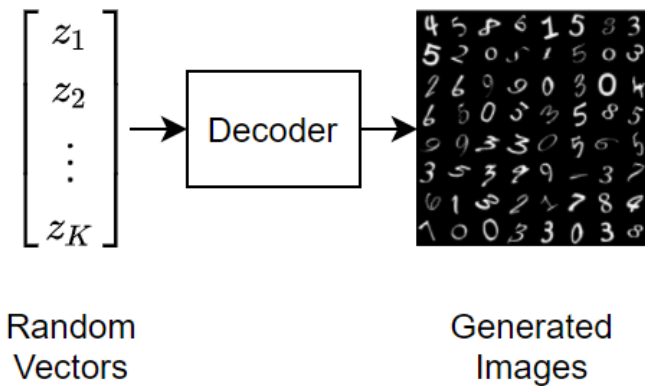
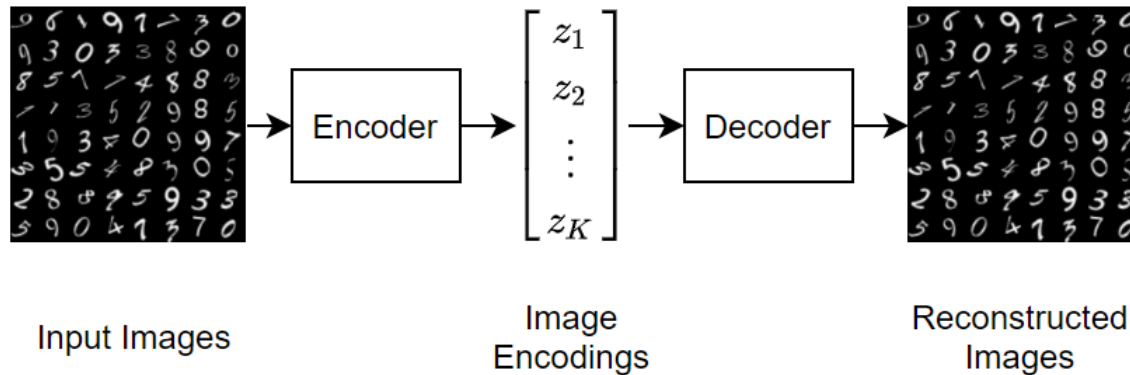
Autoencoder (AE): Latent Space



Quelle: [Autoencoders | Deep Learning Animated by Deepia](#)
(Thibaut Modrzyk)

Im Bottleneck befindet sich der "Latent Space", eine niedrigdimensionale Darstellung der Eingabedaten. Diese Repräsentation kann für verschiedene Aufgaben genutzt werden, z.B. Datenkompression oder Generierung neuer Daten.

Bildgenerierung mit (Variational) Autoencoder



Quelle: Train Variational Autoencoder (VAE) to Generate Images
- MATLAB & Simulink

Stable Diffusion: Ein neuer Ansatz

- **Diffusionsmodell:**
 - Bilder werden **schrittweise aus Rauschen rekonstruiert.**
 - Training: Das Modell lernt, den **Rauschprozess umzukehren.**
- Erfunden in München: **High-Resolution Image Synthesis with Latent Diffusion Models**, *Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, Björn Ommer*, CVPR 2022 ([Details](#))

Stable Diffusion im Detail

Die folgenden Bilder stammen aus dem Artikel:

[The Illustrated Stable Diffusion by Jay Alammar](#)

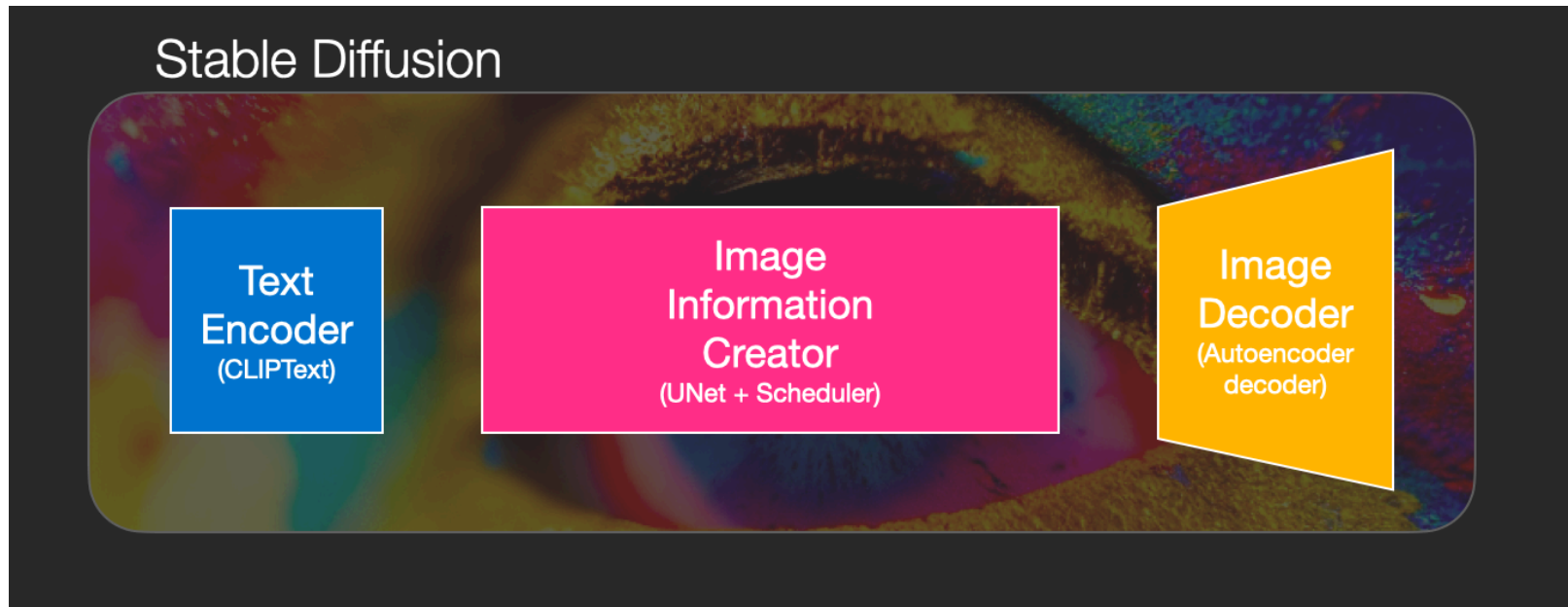
Zum besseren Verständnis zuhause durchgehen und/oder das Video dazu anschauen.

SD: Aufbau



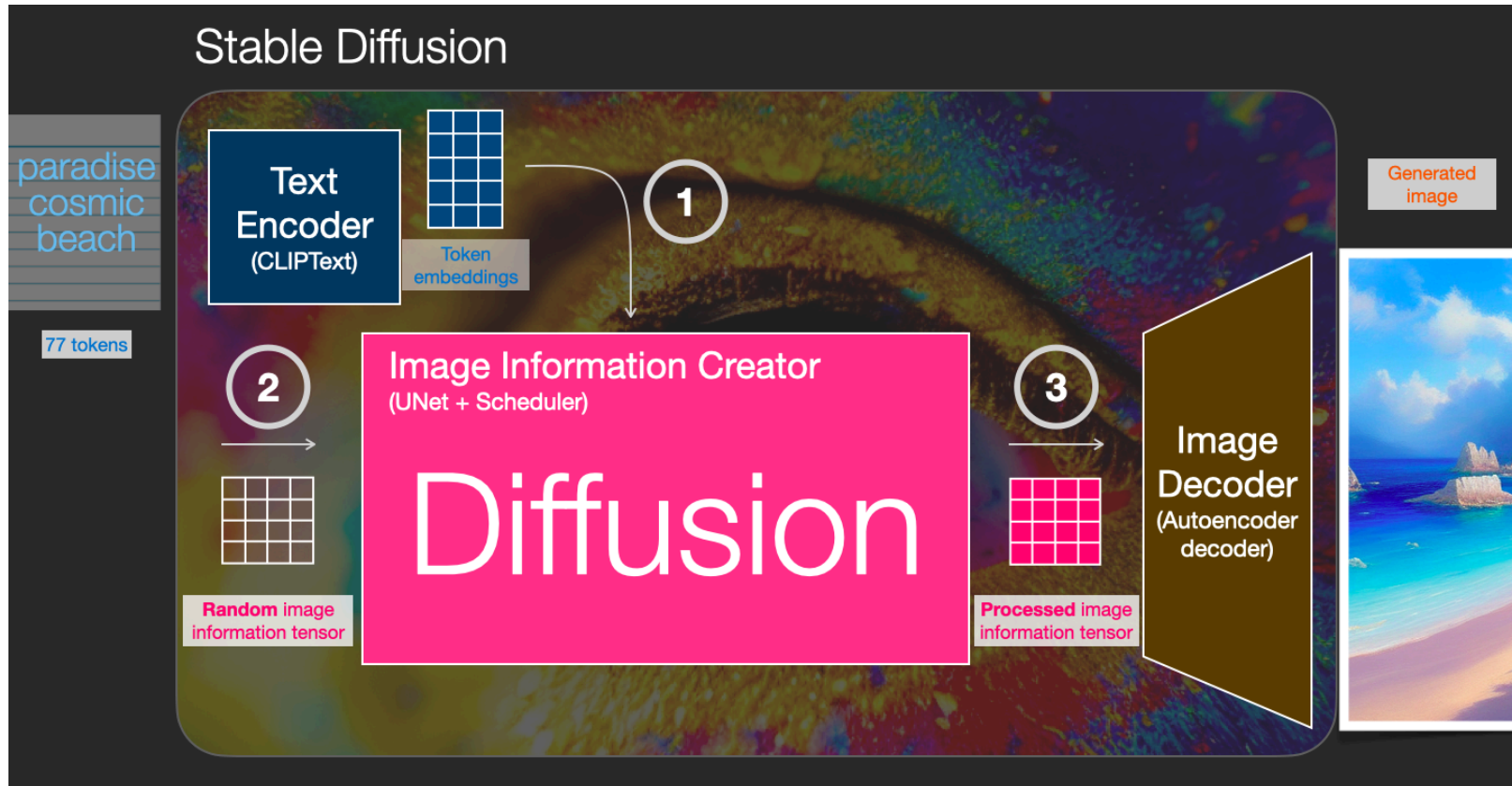
Quelle: [The Illustrated Stable Diffusion by Jay Alammar](#)

SD: Architektur



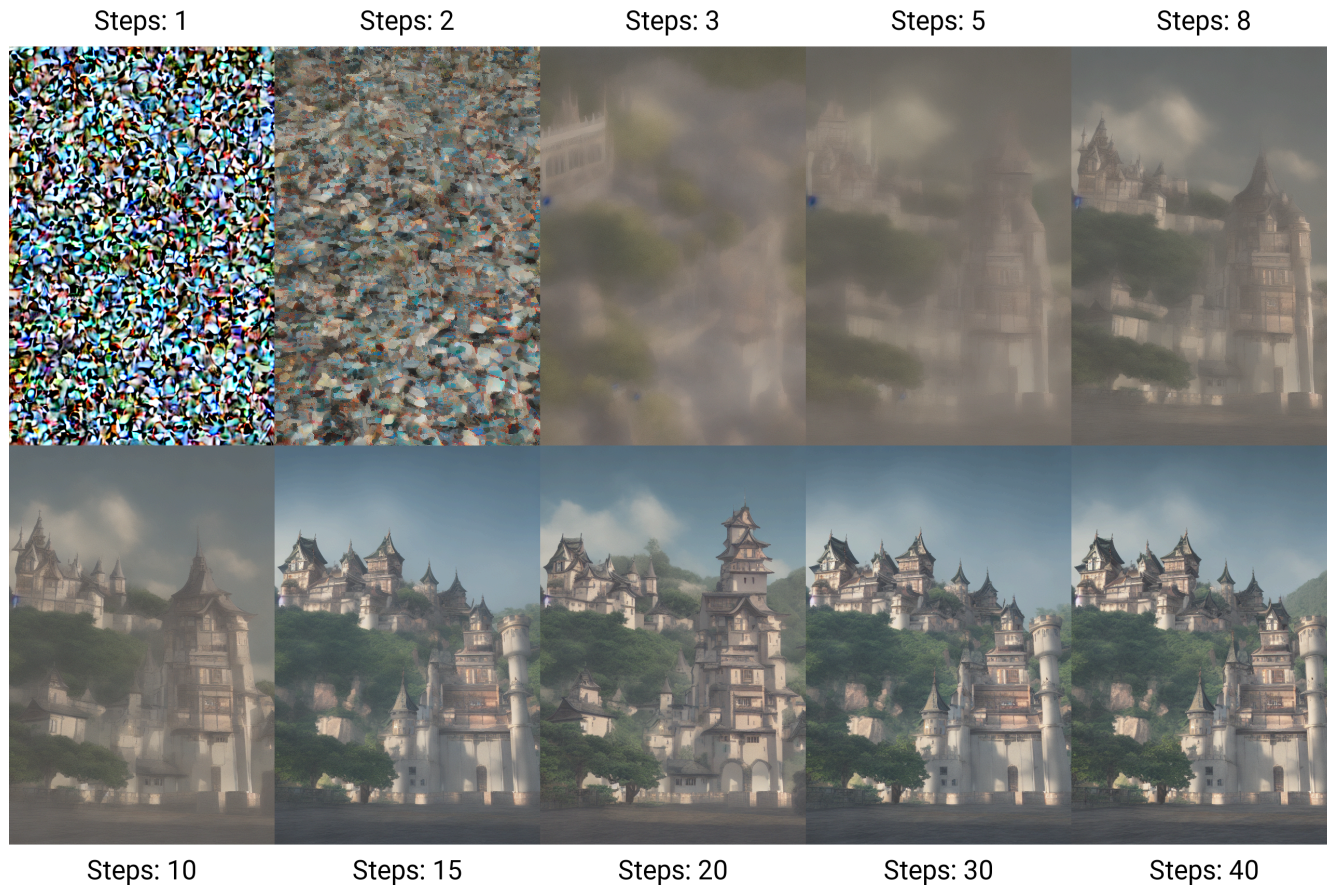
Quelle: [The Illustrated Stable Diffusion by Jay Alammar](#)

SD: Diffusionsprozess



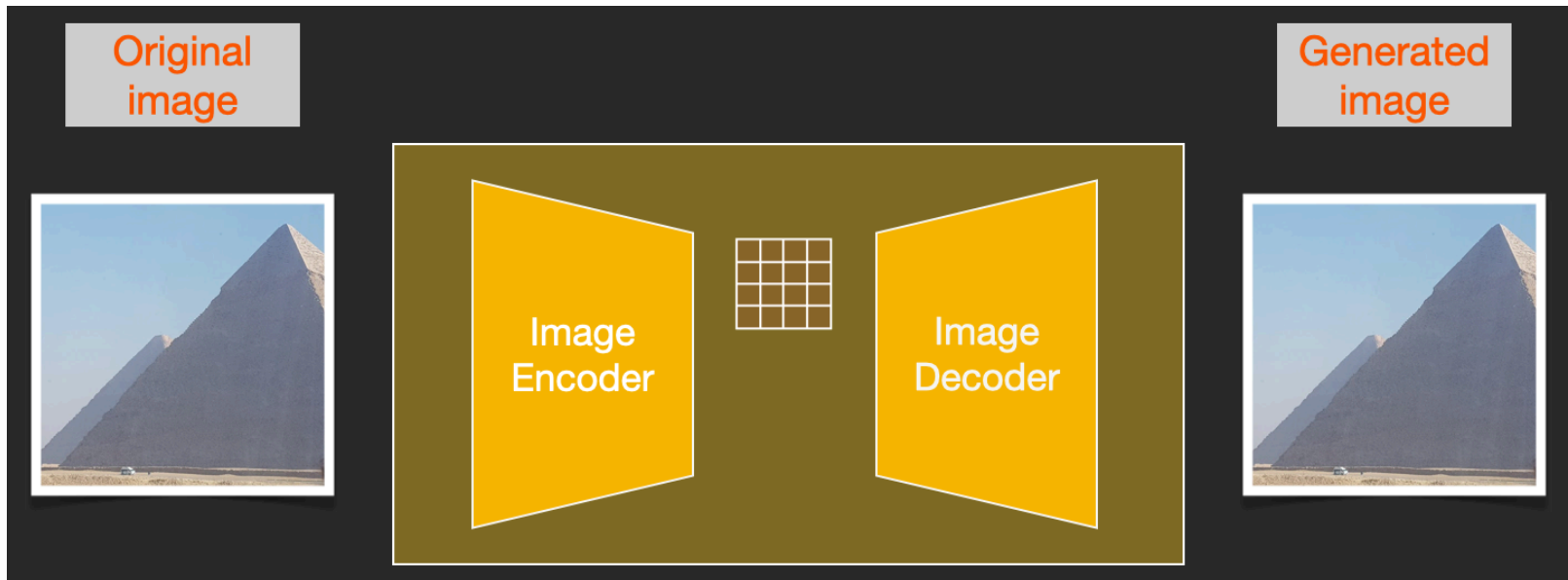
Quelle: [The Illustrated Stable Diffusion by Jay Alammar](#)

SD: Diffusion: Rauschprozess umkehren



Quelle: [Benlisquare](#) - Own work, CC BY-SA 4.0

SD: Einsatz von Autoencodern



Quelle: [The Illustrated Stable Diffusion by Jay Alammar](#)

Vorteile von Stable Diffusion

- **Stabilität:** Robusteres Training im Vergleich zu GANs.
- **Effizienz:** Reduzierte Ressourcenanforderungen durch optimierte Architektur.
- **Kontrollierbarkeit:** Ermöglicht gezielte Generierung (z. B. Text-zu-Bild).

Stable Diffusion (v1.5 from 2022)



Prompt: "frog with melon hat"

Stable Diffusion 1.5 at huggingface

Flow matching statt Diffusion

- Neuere Modelle (2024/2025) verwenden **Flow Matching** anstelle von Diffusion.
- Anstatt beim Training Rauschen schrittweise hinzuzufügen, wird ein **direkter Pfad** zwischen Rauschen und Daten gelernt – es wird also **kein Rauschprozess** simuliert, sondern ein **Fluss** zwischen den beiden Zuständen modelliert.
- Gute, genauere Erklärung in [Flow-Matching vs Diffusion Models explained side by side](#) by AI Coffee Break with Letitia.

Stable Diffusion (v3.5 from 2024)



Prompt: "frog with melon hat"

StabilityAI Stable Diffusion 3.5 large

FLUX.1 [dev] (2025)



Prompt: "frog with melon hat"

Black Forest Labs (Gegründet von den ex-StabilityAI
Mitarbeitern Rombach, Blattmann und Esser)

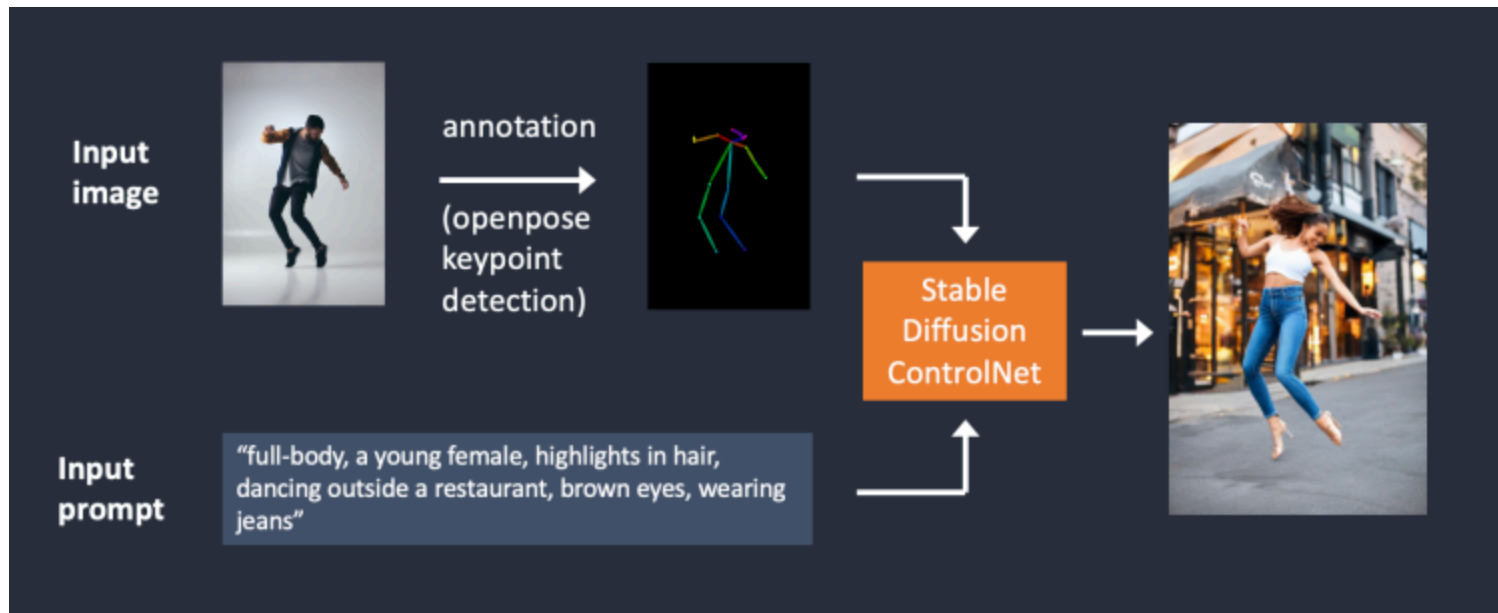
Flux.2 [max] (2026)



Prompt: "frog

with melon hat"

Stable Diffusion steuern (ControlNet)



Bildquelle: [ControlNet: A Complete Guide, Updated July 7, 2024,](#)
By Andrew

Text zu 3D

- Stand der Technik 2021 mit [Video](#)
- Stand der Technik 2024: [Sora von OpenAI](#), [veo 2 von Google](#)
- Stand der Technik 2025: [Sora2 von OpenAI](#), [veo3 von Google](#)

GANs vs. Stable Diffusion

- **Beide sind generative Modelle**, aber mit unterschiedlichen Ansätzen.
- Ziel: Realistische Daten, wie Bilder oder Texte, erzeugen.
- Unterschiedliche Prinzipien:
 - GANs: Wettbewerb zwischen Generator und Diskriminator.
 - Stable Diffusion: Iterative Umkehr eines Rauschprozesses.
- Beide verwenden Autoencoder-Konzepte (GANs lassen sich mit Autoencodern kombinieren).

The future: Frame generation



NEW
DLSS Multi Frame Generation
Multiplies performance by generating multiple frames

[NVIDIA DLSS4, video](#)