Generating other media

AI for Media, Art & Design, Spring 2024

Prof. Perttu Hämäläinen
Aalto University

Contents

General principles

Token-by-token generation using Transformers

- Parti and DALL-E 1 generated images by modeling sequences of both text and image tokens
- In the same way, **anything** can be tokenized for Transformer training
- Problems:
  - Expensive and slow to train
  - Training a large model from scratch requires a lot of data
- Solution: Repurpose pretrained LLMs

Finetuning a pretrained text transformer using custom data

Lu et al. successfully converted a GPT-2 text generator into an image classifier

Pretrained Transformers as Universal Computation Engines

Kevin Lu,[1,2] Aditya Grover,[2,3] Pieter Abbeel,[1] Igor Mordatch[4]

[1] UC Berkeley, [2] Facebook AI Research, [3] UCLA, [4] Google Brain
kzl@fb.com

Abstract

We investigate the capability of a transformer pretrained on natural language to generalize to other modalities with minimal finetuning – in particular, without finetuning of the self-attention and feedforward layers of the residual blocks. We consider such a model, which we call a Frozen Pretrained Transformer (FPT), and study finetuning it on a variety of sequence classification tasks spanning numerical computation, vision, and protein fold prediction. In contrast to prior works which investigate finetuning on the same modality as the pretraining dataset, we show that pretraining on natural language can improve performance and compute efficiency on non-language downstream tasks. Additionally, we perform an analysis of the architecture, comparing the performance of a random initialized transformer to a random LSTM. Combining the two insights, we find language-pretrained transformers can obtain strong performance on a variety of non-language tasks.

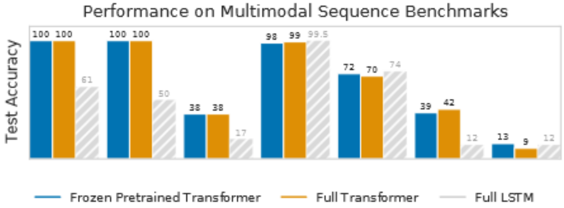


Figure 1: A *frozen* language-pretrained transformer (FPT) – without finetuning the self-attention and feedforward layers – can achieve strong performance compared to a transformer fully trained from scratch on a downstream *modality* on literature benchmarks (Tay et al. 2020; Rao et al. 2019). We show results on diverse classification tasks (see Section 2.1): numerical computation (Bit Memory/XOR, ListOps), image classification (MNIST, CIFAR-10, LRA), and protein fold prediction (Homology). We also show results for a fully-trained from-scratch LSTM as a baseline. Our code is available at: github.com/kzl/universal-computation

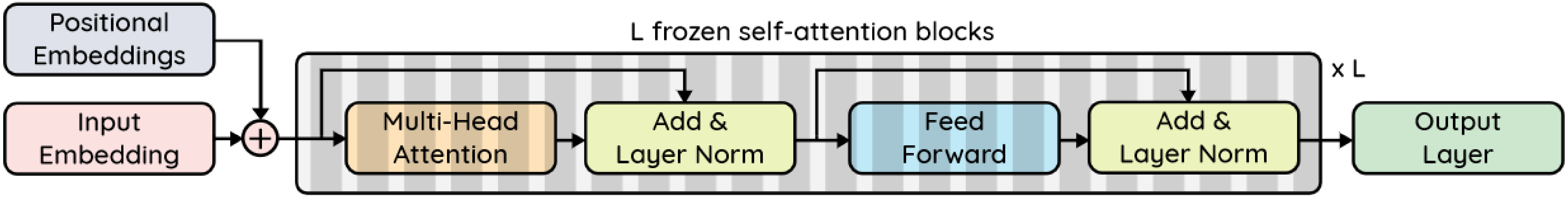
1 Introduction

The transformer architecture (Vaswani et al. 2017) has shown broad successes in deep learning, serving as the backbone of large models for tasks such as modeling natural language (Brown et al. 2020), images (Dosovitskiy et al. 2020), proteins (Jumper et al. 2021), and multimodal tasks comprising of both images and text (Lu et al. 2019; Radford et al. 2021). Inspired by these successes, we seek to explore the generalization capabilities of a transformer in transferring from one modality to another.

Classical approaches to sequence processing used recurrent neural network (RNN) approaches (Rumelhart, Hinton, and Williams 1985; Hochreiter and Schmidhuber 1997). In contrast, transformers utilize self-attention layers to extract features across tokens of a sequence, such as words (Vaswani et al. 2017) or image patches (Dosovitskiy et al. 2020). Furthermore, it has become common practice to train large models on unsupervised objectives before finetuning or evaluating zero-shot generalization on a downstream task. However, the downstream tasks that have been studied are generally restricted to the same modality as the original training set: for example, train GPT (Radford et al. 2018) on a large language corpus, and finetune on a small task-specific dataset. Our goal in this work is to investigate finetuning on modalities distinct from the training modality.

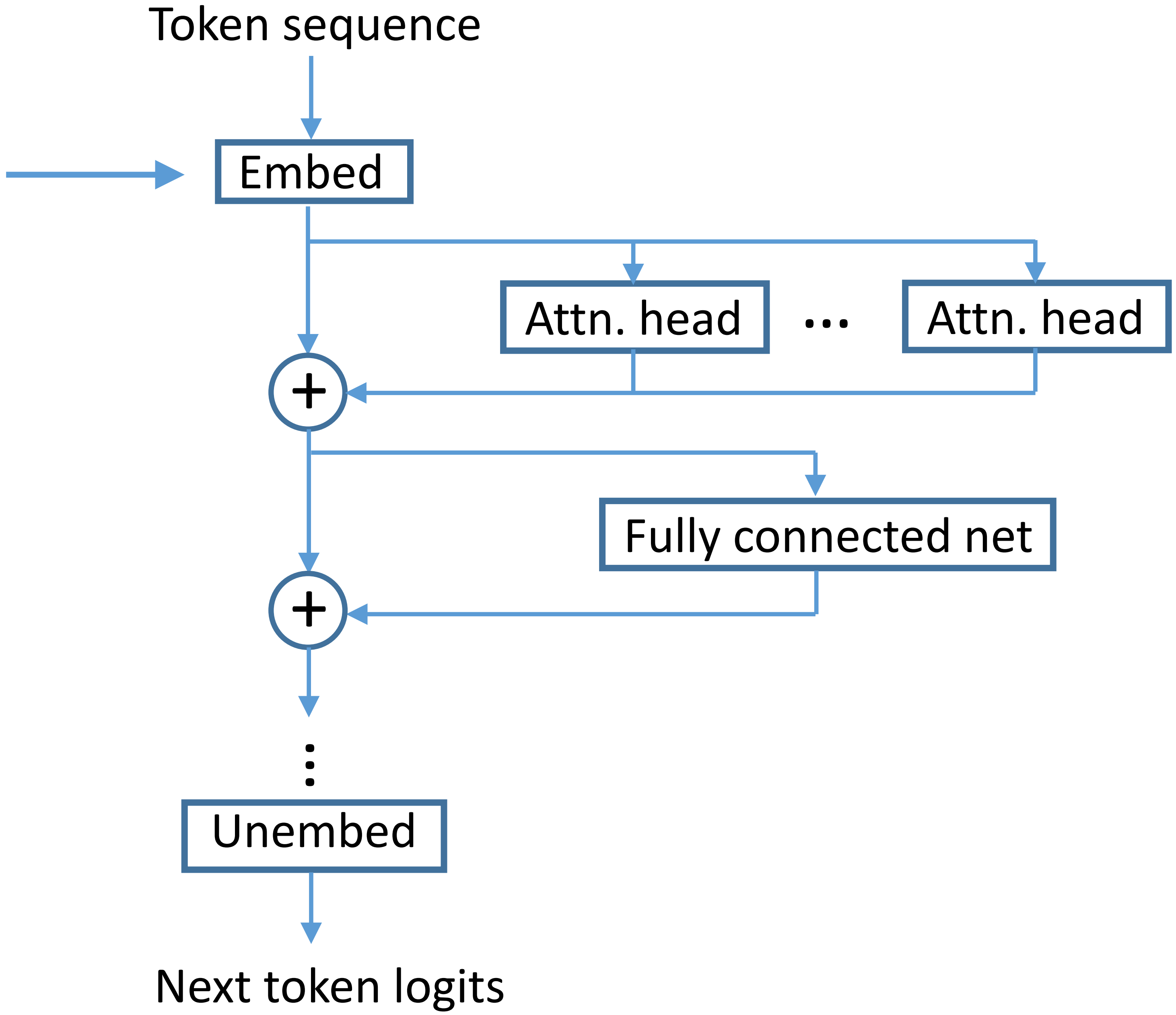
We hypothesize that transformers – namely the self-attention layers – can be pretrained on a data-rich modality (i.e. where data is plentiful, such as a language corpus) and identify feature representations that are useful for *arbitrary* data sequences, enabling downstream transfer to different modalities. In particular, we seek to investigate what pretrained language models (LMs) are capable of in terms of generalizing to other modalities with sequential structure.

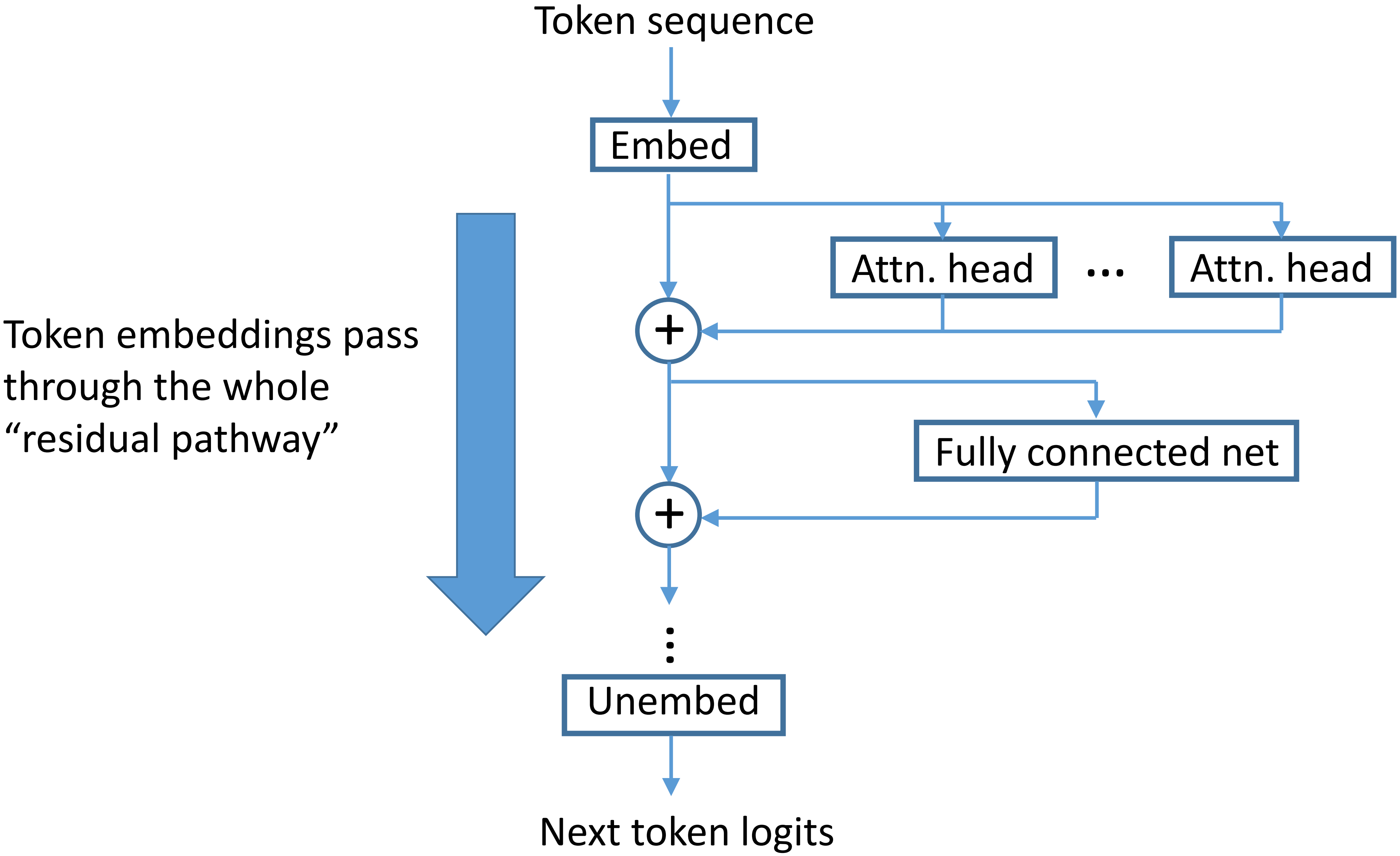
To investigate this hypothesis, we take a transformer model pretrained on natural language data, GPT-2 (Radford et al. 2019), and finetune only the linear input and output layers, as well as the positional embeddings and layer norm parameters. These decisions are made to highlight the parameters already in the language model, and not for performance purposes. We call this model a Frozen Pretrained Transformer (FPT). On a range of tasks across a variety of modalities – including numerical computation, image classification, and protein fold prediction – FPT displays comparable performance to training the entire transformer from



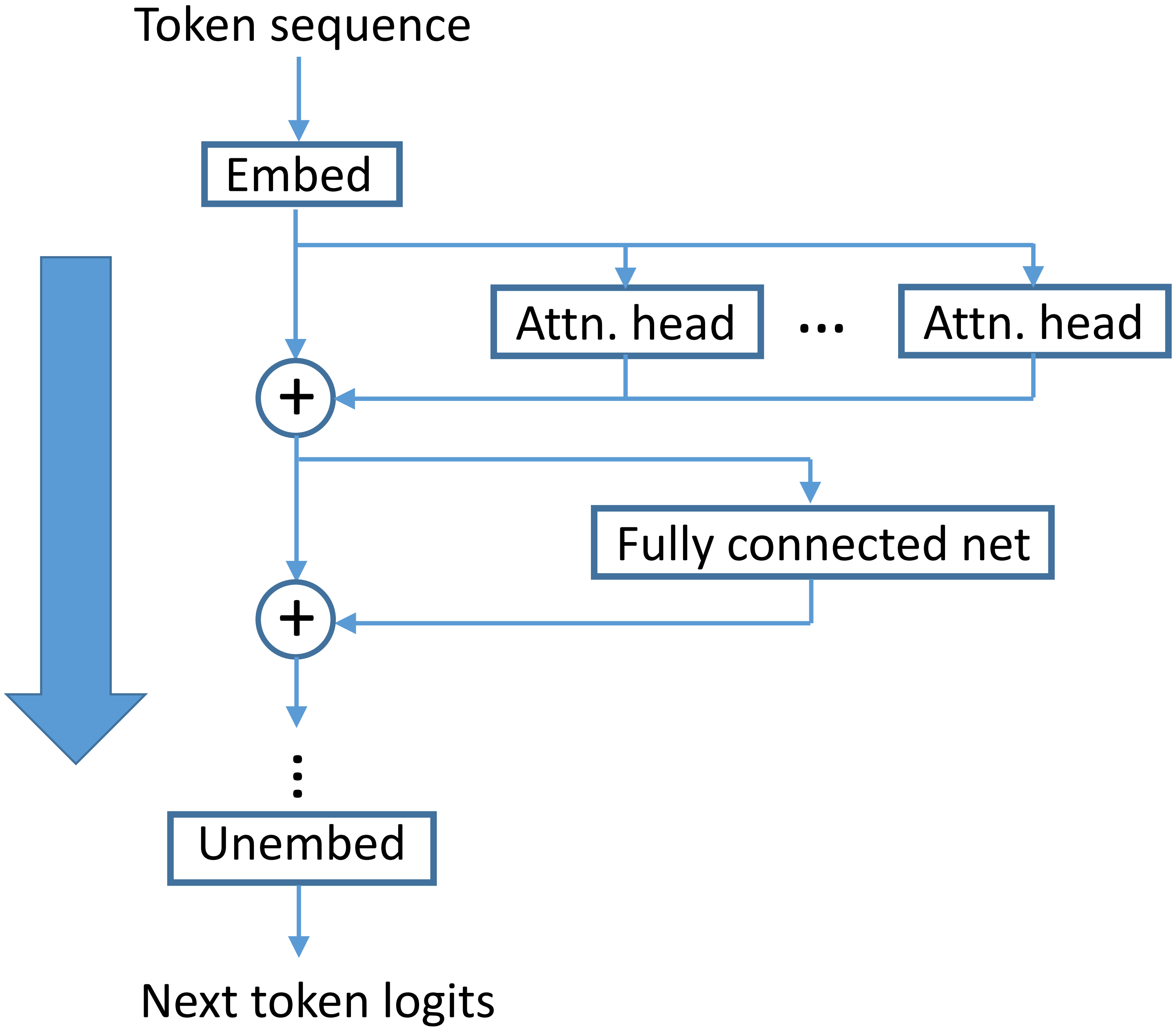
The architecture only retrains the input embedding and output layers, keeping rest of GPT-2 as is.

A look-up table:
An initial embedding vector for each possible token



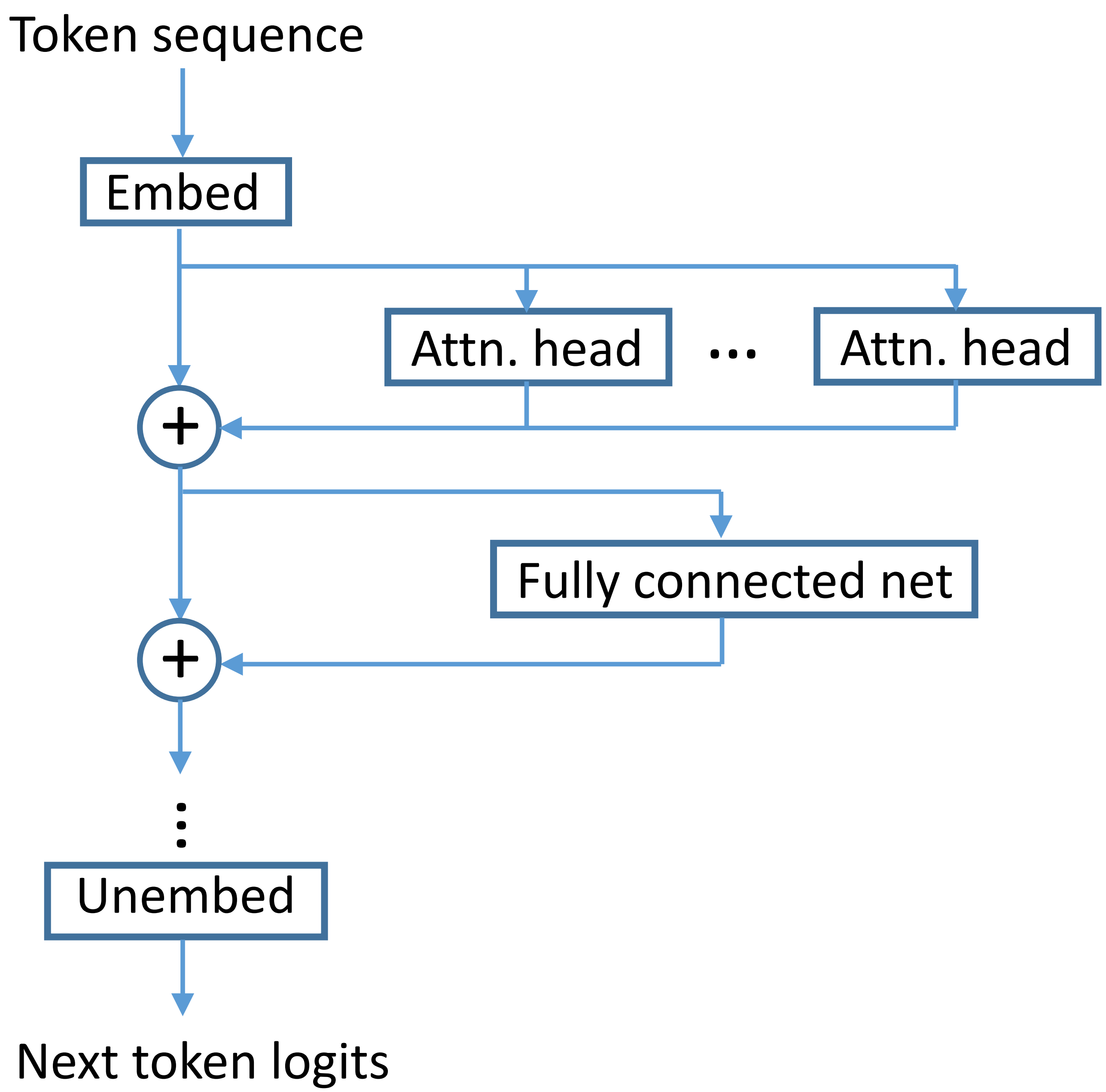

Token sequence
Embed
Attn. head
...
Attn. head
Token embeddings pass
through the whole
“residual pathway”
Fully connected net
Unembed
Next token logits

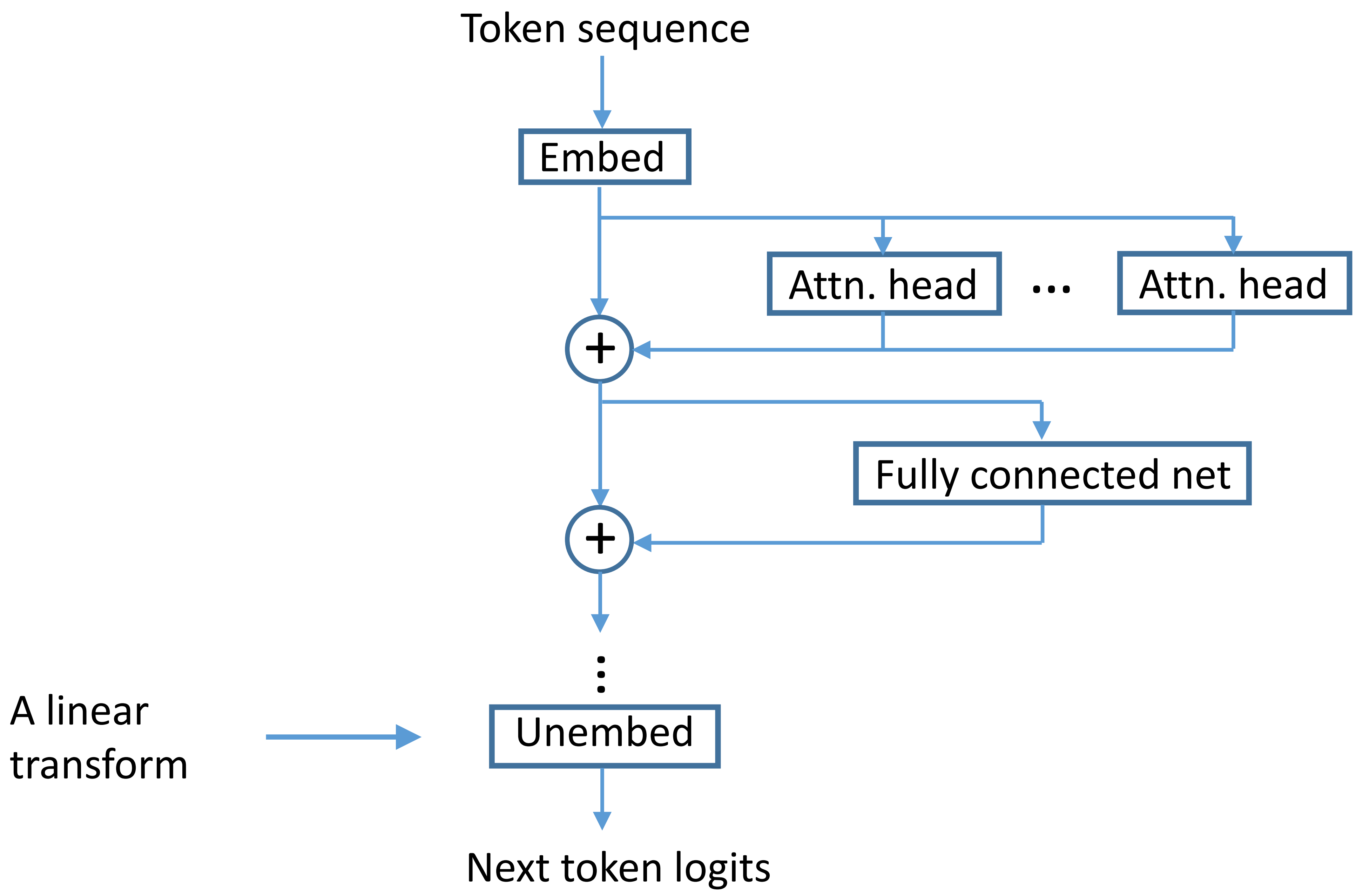
Essentially, the residual pathway works as a working memory, with attention heads reading from it and writing to it



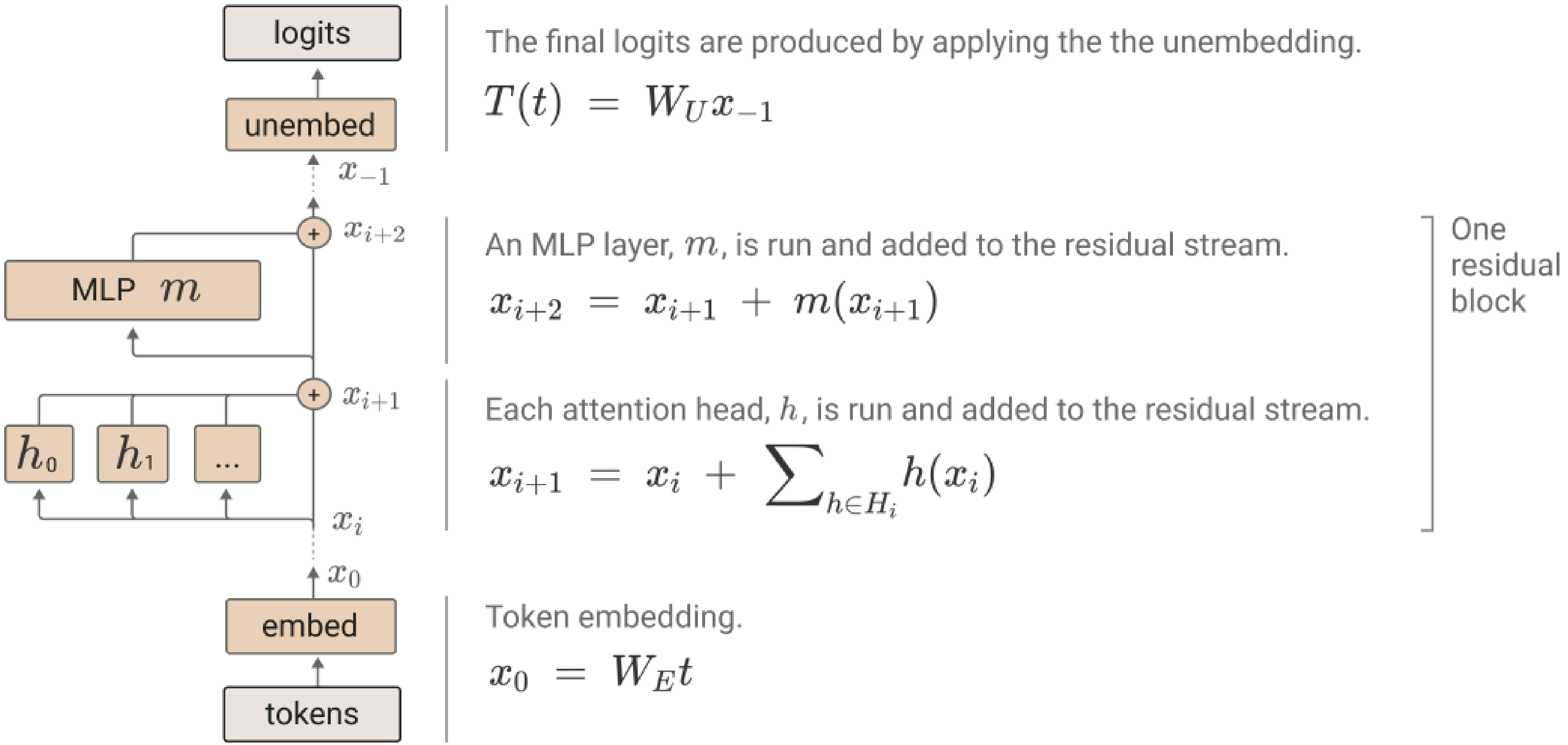
Each token's embedding represents its meaning and relation to other tokens.

The embeddings get gradually modified by each layer, e.g., annotating them with further information




Token sequence
Embed
Attn. head
...
Attn. head
+
Fully connected net
+
A linear transform
Unembed
Next token logits

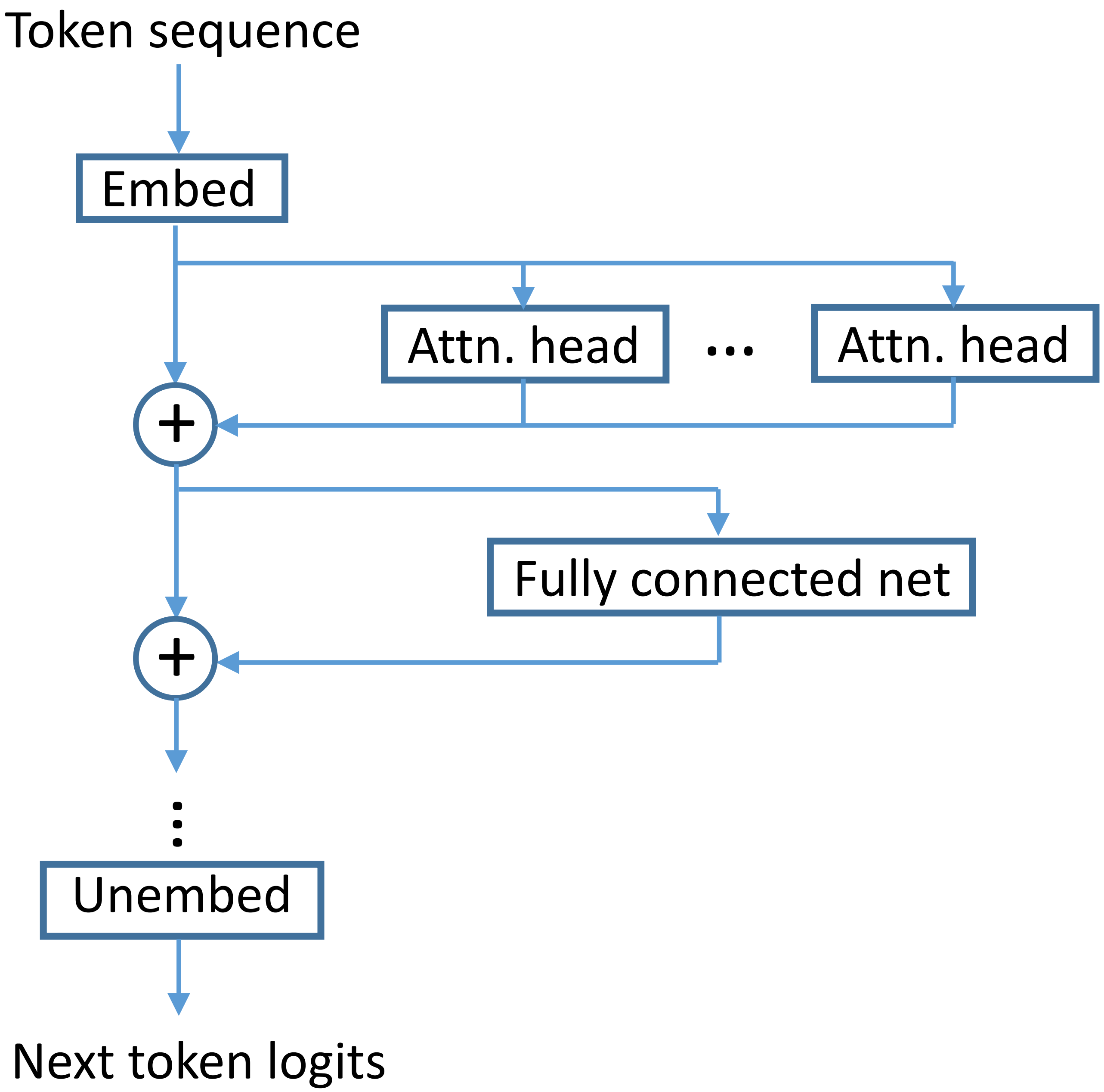
More detail



https://transformer-circuits.pub/2021/framework/index.html

Retraining the embed and unembed layers for a new vocabulary of tokens = repurposing the complex computational machinery learned for text to other data.

Even better results can be obtained by also adding and training LoRA adapters.



How to tokenize?

- An autoencoder can compress/decompress complex data such as image patches into N-dimensional vectors
- VQ-VAE: The vectors can be converted to discrete tokens using vector quantization

Generating Diverse High-Fidelity Images with VQ-VAE-2

Ali Razavi* DeepMind alirazavi@google.com

Aäron van den Oord* DeepMind avdnoord@google.com

Oriol Vinyals DeepMind vinyals@google.com

Abstract

We explore the use of Vector Quantized Variational AutoEncoder (VQ-VAE) models for large scale image generation. To this end, we scale and enhance the autoregressive priors used in VQ-VAE to generate synthetic samples of much higher coherence and fidelity than possible before. We use simple feed-forward encoder and decoder networks, making our model an attractive candidate for applications where the encoding and/or decoding speed is critical. Additionally, VQ-VAE requires sampling an autoregressive model only in the compressed latent space, which is an order of magnitude faster than sampling in the pixel space, especially for large images. We demonstrate that a multi-scale hierarchical organization of VQ-VAE, augmented with powerful priors over the latent codes, is able to generate samples with quality that rivals that of state of the art Generative Adversarial Networks on multifaceted datasets such as ImageNet, while not suffering from GAN's known shortcomings such as mode collapse and lack of diversity.

1 Introduction

Deep generative models have significantly improved in the past few years [5, 27, 25]. This is, in part, thanks to architectural innovations as well as computation advances that allows training them at larger scale in both amount of data and model size. The samples generated from these models are hard to distinguish from real data without close inspection, and their applications range from super resolution [21] to domain editing [44], artistic manipulation [36], or text-to-speech and music generation [25].

Figure 1: Class-conditional 256x256 image samples from a two-level model trained on ImageNet.

We distinguish two main types of generative models: likelihood based models, which include VAEs [16, 31], flow based [9, 30, 10, 17] and autoregressive models [20, 39]; and implicit generative

*Equal contributions.

Residual Vector Quantization (RVQ)

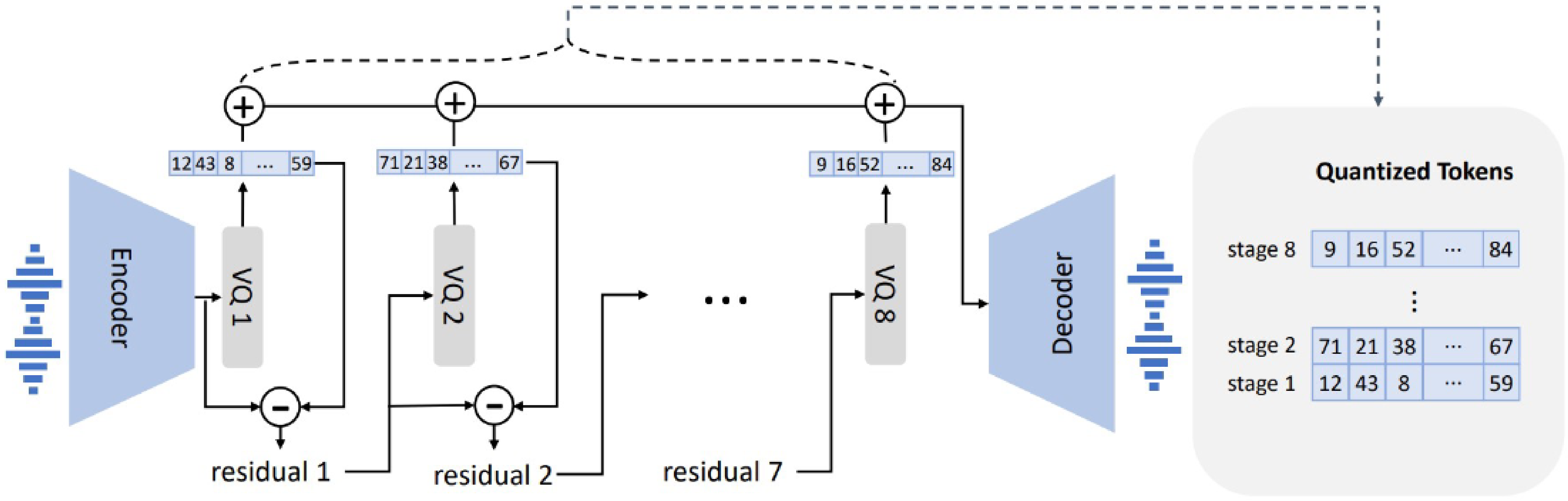


Figure 2: The neural audio codec model revisit. Because RVQ is employed, the first quantizer plays the most important role in reconstruction, and the impact from others gradually decreases.

https://arxiv.org/abs/2301.02111

Recap: A general and powerful recipe

- Tokenize your custom data using an autoencoder and RVQ
    - For simple data, the autoencoder may not be needed
- Finetune an LLM such as Llama or Phi
    - Custom data only: Replace the embedding layer
    - Custom data interleaved with text: Expand the vocabulary and the embedding layer with your RVQ tokens
    - Replace the unembedding layer
    - Add LoRA adapters
- This course: Colab demo provided

Another general recipe

- Diffusion models with a U-Net denoiser are a good default choice if two conditions are met:
  - The data is conveniently represented by real-valued arrays (tensors in ML lingo)
  - You can assume stronger local than global dependencies between the array values (this is what convolutional networks like U-Net are designed for)
- Video is a good example:
  - A video is an array of shape [timesteps,width,height,color channels]
  - Nearby pixels (in both space and time) typically belong to the same object

Multimodal LLMs

Multimodal LLMs

- LLMs where the token vocabulary includes text, image patches, audio segments…
- Motivation:
  - Text is high-level and abstract => can’t produce precise and “grounded” understanding of the physical world
  - There’s only so much text data available
  - => internet video is the next data goldmine, especially tutorials and let’s play videos where a human explains what is happening
- Abilities: chat about images, revise generated images using chat, generate image sequences and video
- How to create: Plug existing image encoders and decoders to an LLM with simple linear or MLP layers that map image embeddings to the LLM’s text embeddings space and back. Train the mapping layers, keep everything else fixed.

Flamingo (2022)

Demonstrated that few-shot prompting (in-context learning) is possible with a mixture of images and words.

I.e., by showing examples, one can instruct the model to perform new tasks involving both images and text.

No image generation yet.

Flamingo: a Visual Language Model for Few-Shot Learning

Jean-Baptiste Alayrac*,‡ Jeff Donahue* Pauline Luc* Antoine Miech*
Iain Barr† Yana Hasson† Karel Lenc† Arthur Mensch† Katie Millican†
Malcolm Reynolds† Roman Ring† Eliza Rutherford† Serkan Cabi Tengda Han
Zhitao Gong Sina Samangooei Marianne Monteiro Jacob Menick
Sebastian Borgeaud Andrew Brock Aida Nematzadeh Sahand Sharifzadeh
Mikolaj Binkowski Ricardo Barreira Oriol Vinyals Andrew Zisserman
Karen Simonyan*,‡

* Equal contributions, ordered alphabetically, † Equal contributions, ordered alphabetically, ‡ Equal senior contributions

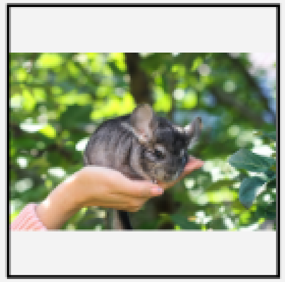
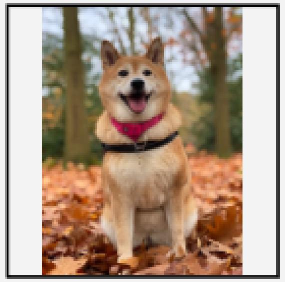
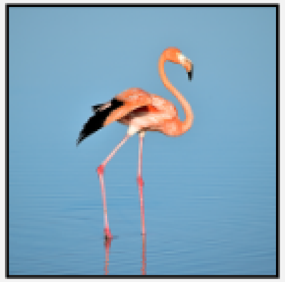
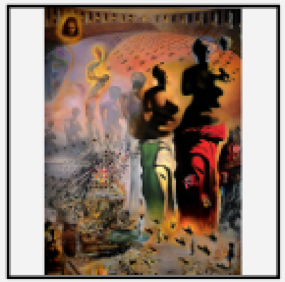
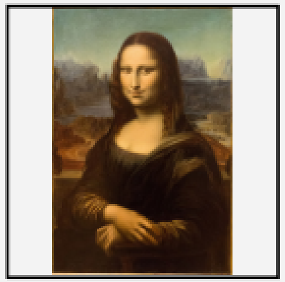
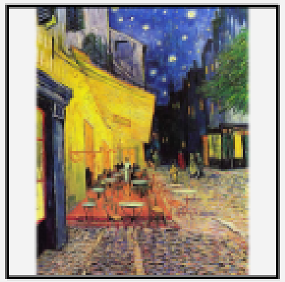
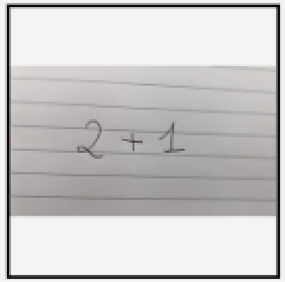
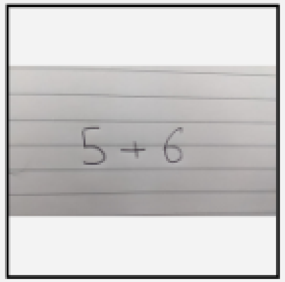
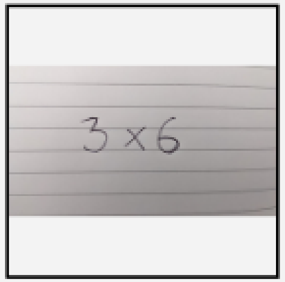
DeepMind

Abstract

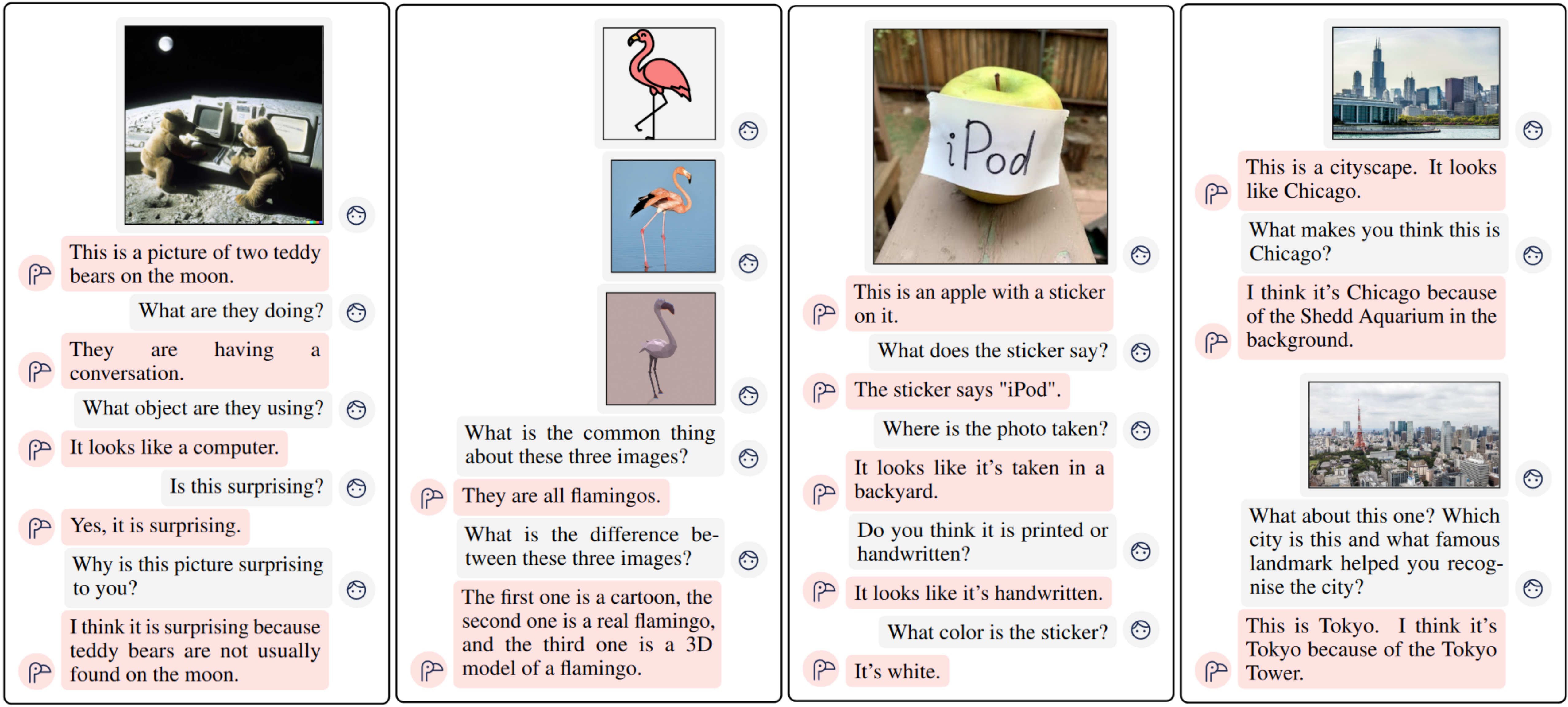
Building models that can be rapidly adapted to novel tasks using only a handful of annotated examples is an open challenge for multimodal machine learning research. We introduce Flamingo, a family of Visual Language Models (VLM) with this ability. We propose key architectural innovations to: (i) bridge powerful pretrained vision-only and language-only models, (ii) handle sequences of arbitrarily interleaved visual and textual data, and (iii) seamlessly ingest images or videos as inputs. Thanks to their flexibility, Flamingo models can be trained on large-scale multimodal web corpora containing arbitrarily interleaved text and images, which is key to endow them with in-context few-shot learning capabilities. We perform a thorough evaluation of our models, exploring and measuring their ability to rapidly adapt to a variety of image and video tasks. These include open-ended tasks such as visual question-answering, where the model is prompted with a question which it has to answer; captioning tasks, which evaluate the ability to describe a scene or an event; and close-ended tasks such as multiple-choice visual question-answering. For tasks lying anywhere on this spectrum, a *single* Flamingo model can achieve a new state of the art with few-shot learning, simply by prompting the model with task-specific examples. On numerous benchmarks, *Flamingo* outperforms models fine-tuned on thousands of times more task-specific data.

Corresponding authors: {jalayrac|jeffdonahue|paulineluc|miech}@deepmind.com
36th Conference on Neural Information Processing Systems (NeurIPS 2022).

https://proceedings.neurips.cc/paper_files/paper/2022/hash/960a172bc7fbf0177ccccbb411a7d800-Abstract-Conference.html

Input Prompt							Completion
	This is a chinchilla. They are mainly found in Chile.		This is a shiba. They are very popular in Japan.		This is	→	**a flamingo. They are found in the Caribbean and South America.**
	What is the title of this painting? Answer: The Hallucinogenic Toreador.		Where is this painting displayed? Answer: Louvres Museum, Paris.		What is the name of the city where this was painted? Answer:	→	**Arles.**
	Output: "Underground"		Output: "Congress"		Output:	→	**"Soulomes"**
	2+1=3		5+6=11			→	**3x6=18**

https://proceedings.neurips.cc/paper_files/paper/2022/hash/960a172bc7fbf0177ccccbb411a7d800-Abstract-Conference.html



https://proceedings.neurips.cc/paper_files/paper/2022/hash/960a172bc7fbf0177ccccbb411a7d800-Abstract-Conference.html

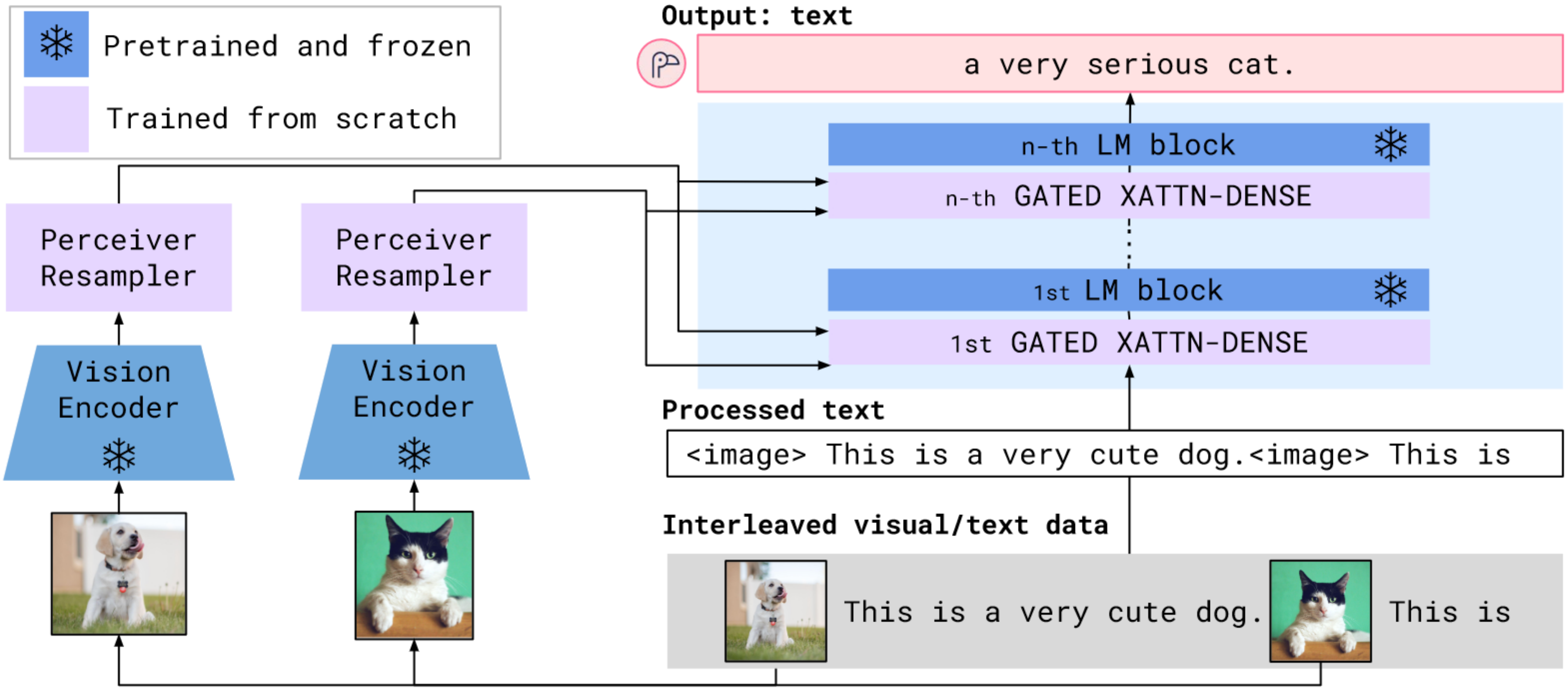


Figure 3: **Flamingo architecture overview.** Flamingo is a family of visual language models (VLMs) that take as input visual data interleaved with text and produce free-form text as output.

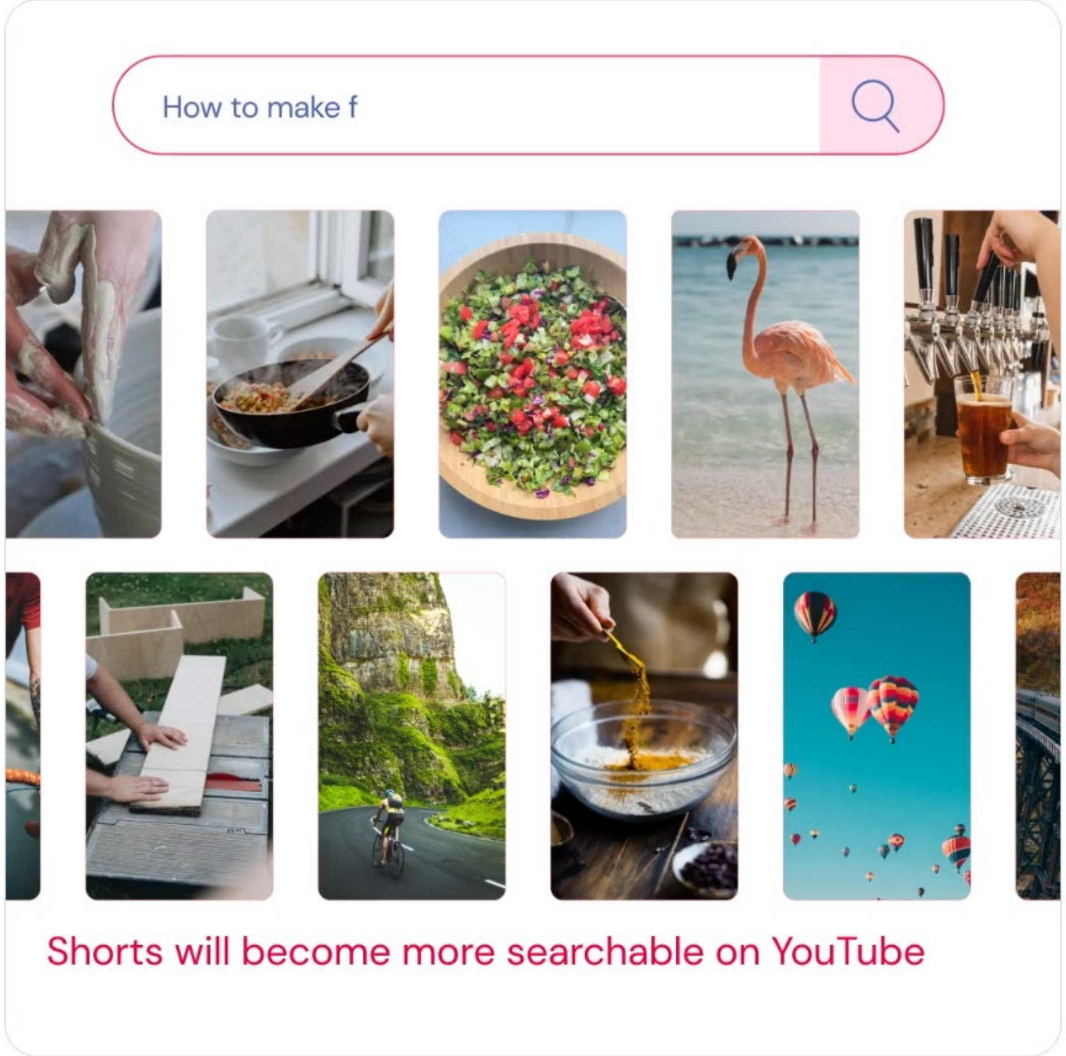
Vision encoder: Normalizer-Free ResNet (NFNet) https://proceedings.mlr.press/v139/brock21a.html
LM blocks: Chinchilla https://arxiv.org/abs/2203.15556

Google DeepMind @GoogleDeepMind · May 24, 2023 ···

Our powerful visual language model Flamingo is changing the way *you* can watch @YouTube Shorts. 🦩

It automatically generates descriptions for hundreds of millions of videos in their metadata, making them more searchable.

Here's how AI is helping creators and viewers. ⬇️



LLaVa (2023)

Like Flamingo, but more simple architecture, better quality, and fully open source.

Combines a CLIP ViT vision encoder with a Vicuna LLM

Trained models available via Huggingface: https://huggingface.co/docs/transformers/main/model_doc/llava

Paper: https://arxiv.org/abs/2304.08485 (orig), https://arxiv.org/abs/2304.08485 (update, shown on the right)

Project page: https://llava-vl.github.io/

Improved Baselines with Visual Instruction Tuning

Haotian Liu[1] Chunyuan Li[2] Yuheng Li[1] Yong Jae Lee[1]
[1]University of Wisconsin–Madison [2]Microsoft Research, Redmond
https://llava-vl.github.io

Abstract

Large multimodal models (LMM) have recently shown encouraging progress with visual instruction tuning. In this note, we show that the fully-connected vision-language cross-modal connector in LLaVA is surprisingly powerful and data-efficient. With simple modifications to LLaVA, namely, using CLIP-ViT-L-336px with an MLP projection and adding academic-task-oriented VQA data with simple response formatting prompts, we establish stronger baselines that achieve state-of-the-art across 11 benchmarks. Our final 13B checkpoint uses merely 1.2M publicly available data, and finishes full training in ∼1 day on a single 8-A100 node. We hope this can make state-of-the-art LMM research more accessible. Code and model will be publicly available.

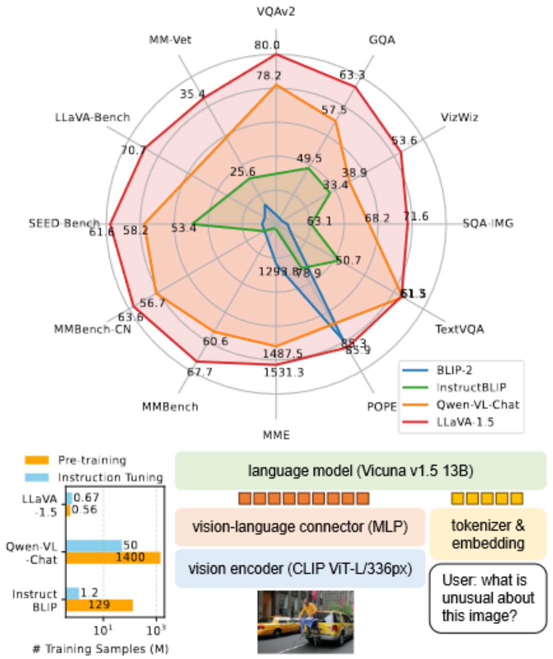


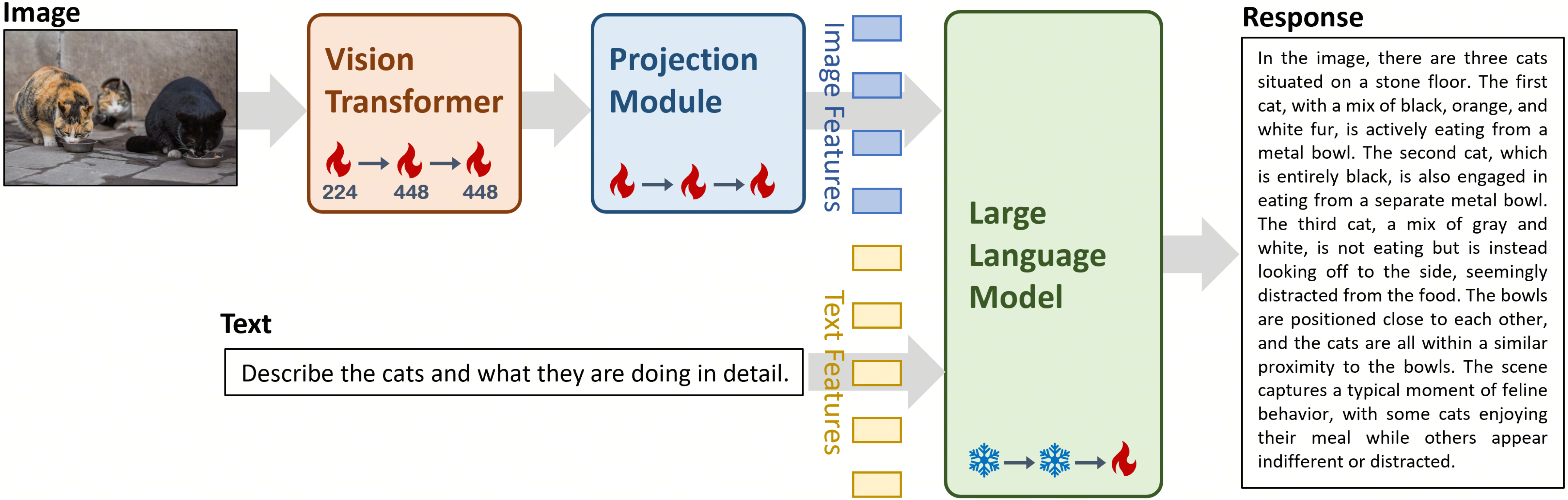
Figure 1. **LLaVA-1.5** achieves SoTA on a broad range of 11 tasks (Top), with high training sample efficiency (Left) and simple modifications to LLaVA (Right): an MLP connector and including academic-task-oriented data with response formatting prompts.

1. Introduction

Large multimodal models (LMMs) have become increasingly popular in the research community, as they are the key building blocks towards general-purpose assistants [1, 22, 35]. Recent studies on LMMs are converging on a central concept known as visual instruction tuning [28]. The results are promising, *e.g.* LLaVA [28] and MiniGPT-4 [49] demonstrate impressive results on natural instruction-following and visual reasoning capabilities. To better understand the capability of LMMs, multiple benchmarks [11, 20, 26, 29, 43] have been proposed. Recent works further demonstrate improved performance by scaling up the pretraining data [2, 9], instruction-following data [9, 21, 45, 46], visual encoders [2], or langauge models [31], respectively. The LLaVA architecture is also leveraged in different downstream tasks and domains, including region-level [6, 44] and pixel-level [19] understanding, biomedical assistants [23], image generation [3], adversarial studies [4, 47].

This note establishes stronger and more feasible baselines built upon the LLaVA framework. We report that two simple improvements, namely, an MLP cross-modal connector and incorporating academic task related data such as VQA, are orthogonal to the framework of LLaVA, and when used with LLaVA, lead to better multimodal understanding capabilities. In contrast to InstructBLIP [9] or Qwen-VL [2], which trains specially designed visual resamplers on hundreds of millions or even billions of image-text paired data, LLaVA uses the simplest architecture design for LMMs and requires only training a simple fully-connected projection layer on merely 600K image-text pairs. Our final model can finish training in ∼1 day on a single 8-A100 machine and achieves state-of-the-art results on a wide range of benchmarks. Moreover, unlike Qwen-VL [2] that includes in-house data in training, LLaVA utilizes only publicly available data. We hope these improved and easily-reproducible baselines will provide a reference for future research in open-source LMM.

Yi-VL (2023)



Multimodal and bilingual (English + Chinese) LLM

LLaVa architecture but with the Yi LLMs instead of Vicuna

https://github.com/01-ai/Yi

MiniGPT-4: A minimalistic open multimodal model

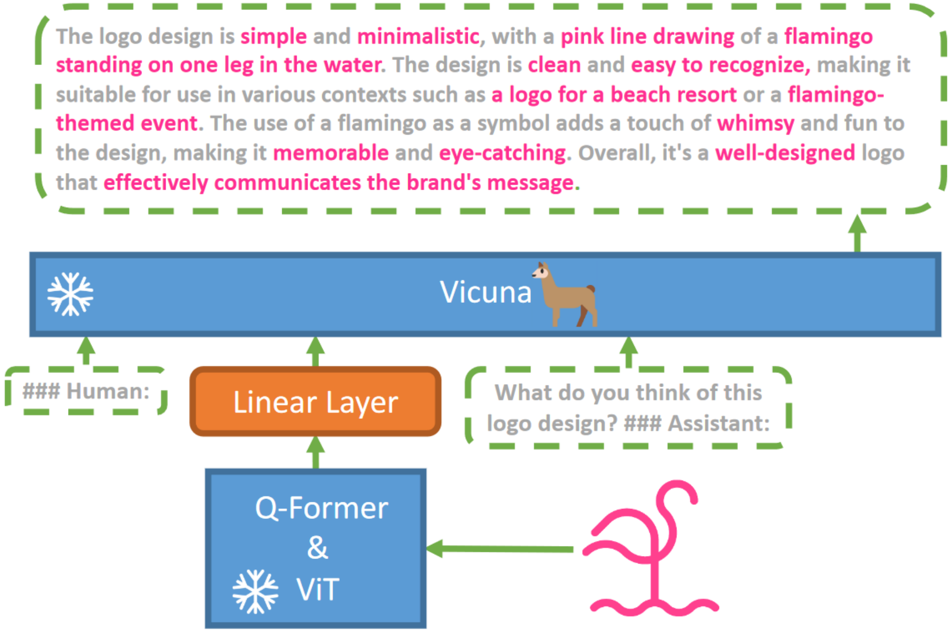


Figure 1: **The architecture of MiniGPT-4.** It consists of a vision encoder with a pretrained ViT and Q-Former, a single linear projection layer, and an advanced Vicuna large language model. MiniGPT-4 only requires training the linear projection layer to align the visual features with the Vicuna.

https://minigpt-4.github.io/

MINIGPT-4: ENHANCING VISION-LANGUAGE UNDERSTANDING WITH ADVANCED LARGE LANGUAGE MODELS

Deyao Zhu*, **Jun Chen***, **Xiaoqian Shen**, **Xiang Li**, **Mohamed Elhoseiny**
King Abdullah University of Science and Technology
{deyao.zhu, jun.chen, xiaoqian.shen, xiang.li.1, mohamed.elhoseiny}@kaust.edu.sa

ABSTRACT

The recent GPT-4 has demonstrated extraordinary multi-modal abilities, such as directly generating websites from handwritten text and identifying humorous elements within images. These features are rarely observed in previous vision-language models. However, the technical details behind GPT-4 continue to remain undisclosed. We believe that the enhanced multi-modal generation capabilities of GPT-4 stem from the utilization of sophisticated large language models (LLM). To examine this phenomenon, we present MiniGPT-4, which aligns a frozen visual encoder with a frozen advanced LLM, Vicuna, using one projection layer. Our work, for the first time, uncovers that properly aligning the visual features with an advanced large language model can possess numerous advanced multi-modal abilities demonstrated by GPT-4, such as detailed image description generation and website creation from hand-drawn drafts. Furthermore, we also observe other emerging capabilities in MiniGPT-4, including writing stories and poems inspired by given images, teaching users how to cook based on food photos, and so on. In our experiment, we found that the model trained on short image caption pairs could produce unnatural language outputs (e.g., repetition and fragmentation). To address this problem, we curate a detailed image description dataset in the second stage to finetune the model, which consequently improves the model's generation reliability and overall usability. Our code, pre-trained model, and collected dataset are available at https://minigpt-4.github.io/.

1 INTRODUCTION https://arxiv.org/abs/2304.10592

In recent years, large language models (LLMs) have experienced rapid advancements (Ouyang et al., 2022; OpenAI, 2022; Brown et al., 2020; Scao et al., 2022a; Touvron et al., 2023; Chowdhery et al., 2022; Hoffmann et al., 2022). With exceptional language understanding capabilities, these models can perform a variety of intricate linguistic tasks in a zero-shot manner. Notably, GPT-4, a large-scale multimodal model, has been recently introduced and demonstrated several impressive capabilities of vision-language understanding and generation (OpenAI, 2023). For example, GPT-4 can produce detailed and accurate image descriptions, explain unusual visual phenomena, and even construct websites based on handwritten text instructions.

Although GPT-4 has exhibited remarkable vision language capabilities, the methods behind its exceptional abilities are still a mystery (OpenAI, 2023). We believe that these impressive skills may stem from the utilization of a more advanced large language model (LLM). LLMs have demonstrated various emergent abilities, as evidenced in GPT-3's few-shot prompting setup (Brown et al., 2020) and the findings of Wei *et al.* (2022) (Wei et al., 2022). Such emergent properties are hard to find in smaller-scale models. It is conjectured that these emergent abilities are also applicable to multi-modal models, which could be the foundation of GPT-4's impressive visual description capabilities.

To substantiate our hypothesis, we present a novel vision-language model named MiniGPT-4. It utilizes an advanced large language model (LLM), Vicuna (Chiang et al., 2023), which is built upon LLaMA (Touvron et al., 2023) and reported to achieve 90% of ChatGPT's quality as per GPT-4's

Emu2 (2023)

Open model like LLaVa but adds an image decoder for image tokens => can generate both images and text

https://baaivision.github.io/emu2/

Generative Multimodal Models are In-Context Learners

Quan Sun[1*] Yufeng Cui[1*] Xiaosong Zhang[1*] Fan Zhang[1*] Qiying Yu[2,1*] Zhengxiong Luo[1]
Yueze Wang[1] Yongming Rao[1] Jingjing Liu[2] Tiejun Huang[1,3] Xinlong Wang[1†]

[1] Beijing Academy of Artificial Intelligence [2] Tsinghua University [3] Peking University

*equal contribution †project lead

code & models: https://github.com/baaivision/Emu

Abstract

The human ability to easily solve multimodal tasks in context (i.e., with only a few demonstrations or simple instructions), is what current multimodal systems have largely struggled to imitate. In this work, we demonstrate that the task-agnostic in-context learning capabilities of large multimodal models can be significantly enhanced by effective scaling-up. We introduce ***Emu2****, a generative multimodal model with 37 billion parameters, trained on large-scale multimodal sequences with a unified autoregressive objective.* ***Emu2*** *exhibits strong multimodal in-context learning abilities, even emerging to solve tasks that require on-the-fly reasoning, such as visual prompting and object-grounded generation. The model sets a new record on multiple multimodal understanding tasks in few-shot settings. When instruction-tuned to follow specific instructions,* ***Emu2*** *further achieves new state-of-the-art on challenging tasks such as question answering benchmarks for large multimodal models and open-ended subject-driven generation. These achievements demonstrate that* ***Emu2*** *can serve as a base model and general-purpose interface for a wide range of multimodal tasks. Code and models are publicly available to facilitate future research.*

1. Introduction

Multimodal tasks [26, 42] encompass anything involving understanding and generation in single or multiple modalities [5, 20, 59], which can be highly diverse and long-tail. Previous multimodal systems largely rely on designing task-specific architecture and collecting a sizable supervised training set, both of which are difficult to scale, particularly when this process needs to be repeated for each new task encountered. By contrast, humans can solve a new task in context, *i.e.*, with only a few demonstrations or simple instructions – a capability that current multimodal models have yet to learn.

Recently, generative pretrained language models have demonstrated strong in-context learning abilities [12, 22, 74]. By training a 37-billion-parameter model **Emu2** and thoroughly evaluating it on diverse multimodal tasks, we demonstrate that a scaled-up multimodal generative pretrained model can harness similar in-context learning abilities and effectively generalize to unseen multimodal tasks. **Emu2** is trained with a unified autoregressive objective: predict-the-next-multimodal-element (either visual embeddings or textual tokens). In this unified generative pretraining process, large-scale multimodal sequences (*e.g.*, text, image-text pairs, and interleaved image-text-video) are used for model training.

We measure **Emu2**'s capabilities of learning from a few examples or instructions on standard multimodal datasets, as well as new tasks unseen in the training set. Specifically, **Emu2** is evaluated under two scenarios: (*a*) *few-shot setting*, where we allow as many examples as possible to fit the context window of the model; and (*b*) *instruction tuning*, where the model is tuned to follow specific instructions.

Emu2 achieves promising results in the few-shot setting on a wide range of vision-language tasks. For example, it demonstrates state-of-the-art few-shot performance on multiple visual question-answering datasets. We observe a performance improvement when the number of examples in context increases. Figure 1 illustrates **Emu2**'s strong multimodal reasoning capabilities for tasks in the wild, *e.g.*, recognition and counting in a specific format. **Emu2** also learns to follow visual prompting in context (*e.g.*, the circles laid on the images in Figure 1), even although it struggles at a smaller scale or at zero shot.

As **Emu2** is inherently equipped to handle interleaved text-image-video at both input and output, it serves as a powerful and versatile base model for diverse multimodal tasks, by following specific task instructions. For example, after instruct tuning with conversational data,

†Correspondence to *wangxinlong@baai.ac.cn*

https://arxiv.org/abs/2312.13286

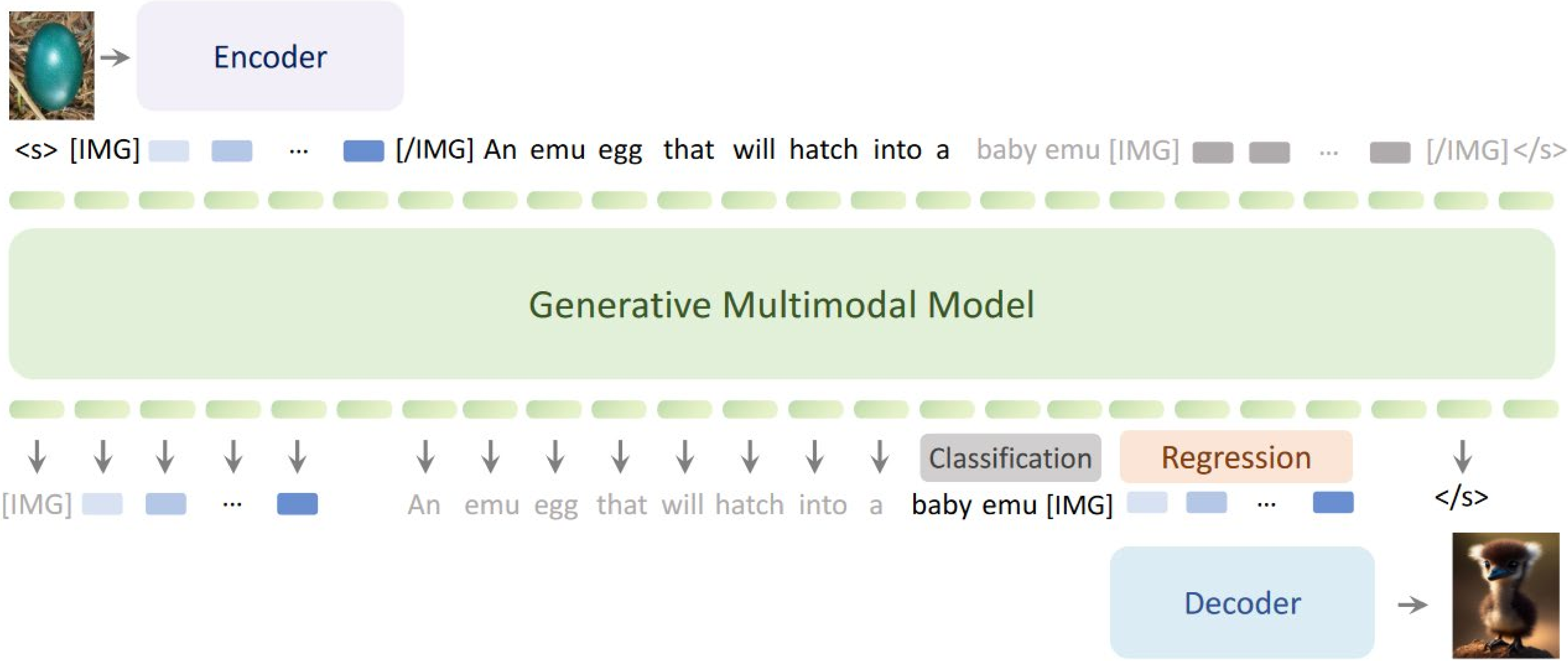
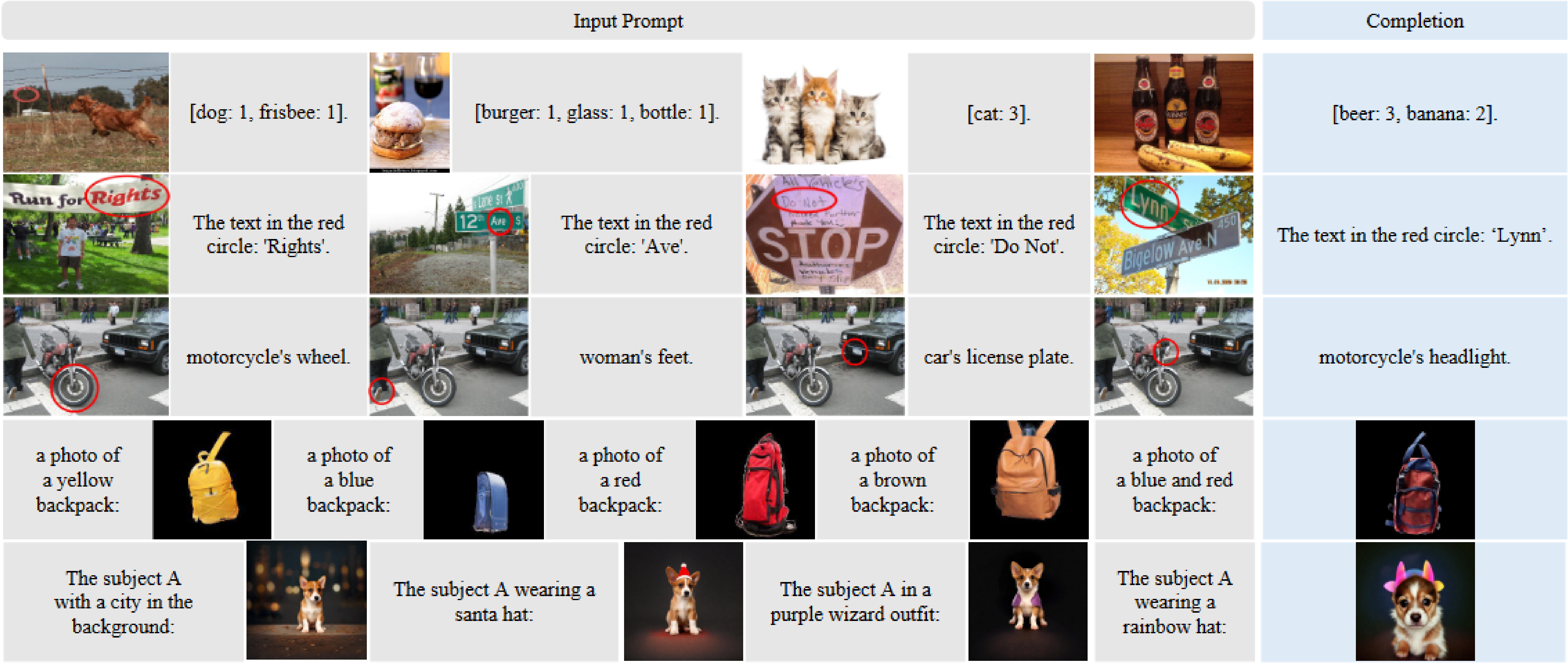
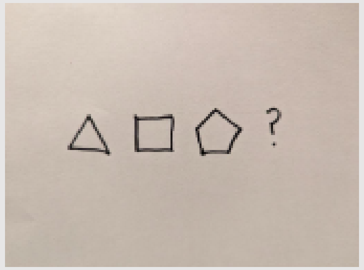


Figure 2. Overview of **Emu2** architecture. **Emu2** learns with a predict-the-next-element objective in multimodality. Each image in the multimodal sequence is tokenized into embeddings via a visual encoder, and then interleaved with text tokens for autoregressive modeling. The regressed visual embeddings will be decoded into an image or a video by a visual decoder.

In-context (few-shot) text & image completion



Language and image understanding



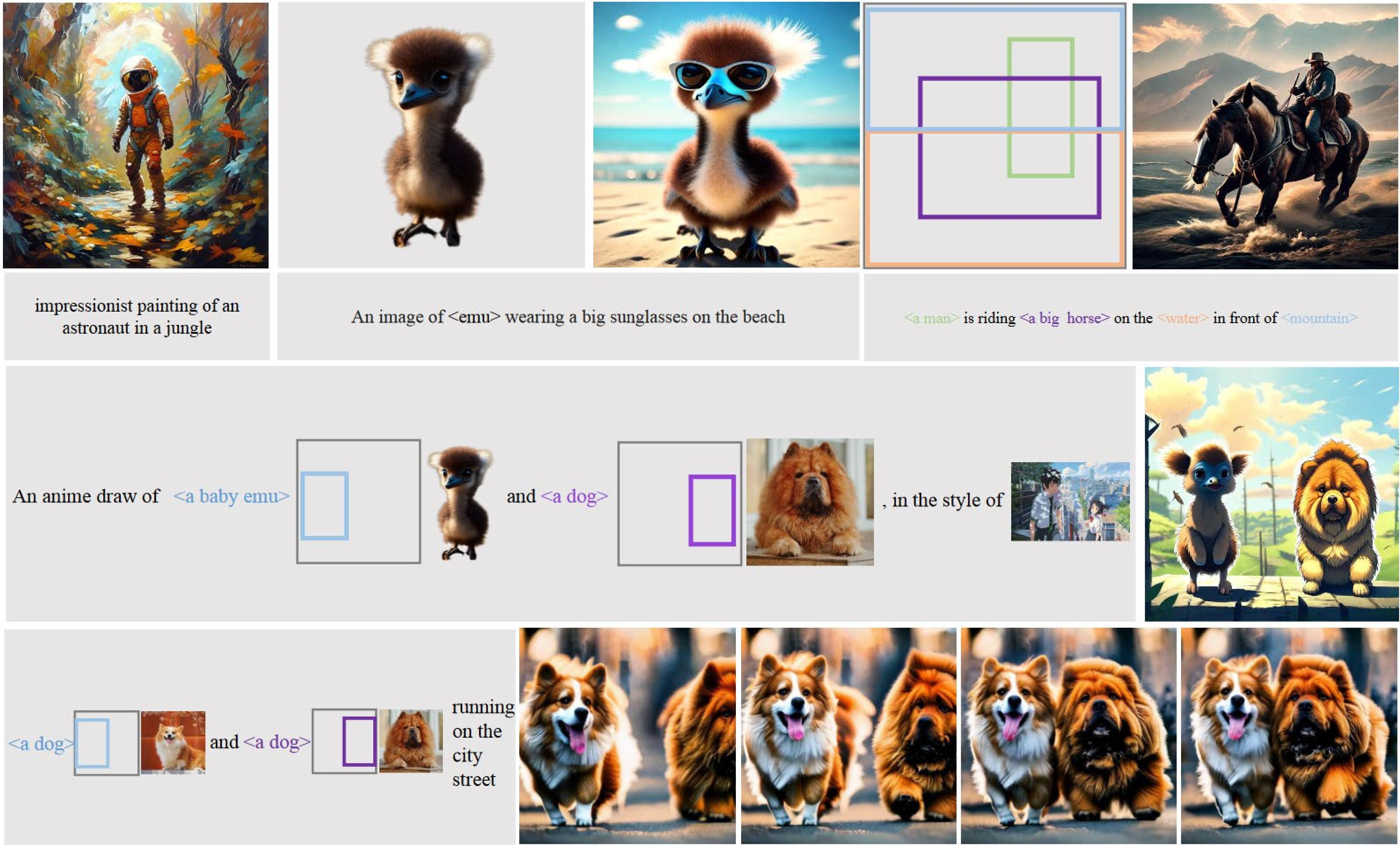
Look at this sequence of three shapes. What shape should come as the fourth shape? Explain your reasoning with detailed descriptions of the first shapes.

The first shape is a triangle, the second shape is a square, and the third shape is a pentagon. To find the next shape in the sequence, we need to identify a pattern. In this case, the pattern is that each shape has one more side than the previous shape. Therefore, the next shape in the sequence should have six sides, which is a **hexagon.**

What is funny about this image?

The funny aspect of this image is that **a squirrel is holding a camera and appears to be taking a picture of a bird.** This is amusing because animals, such as squirrels and birds, do not typically use cameras or engage in human-like activities like photography. The scene is set up in a way that makes it look like the squirrel is intentionally capturing the bird's image, which adds a humorous twist to the scene. Additionally, **the bird is perched on a mushroom**, further enhancing the whimsical and playful nature of the image.

Versatile prompting



GILL

Like Emu2 but can both retrieve and generate images.

https://github.com/kohjingyu/gill

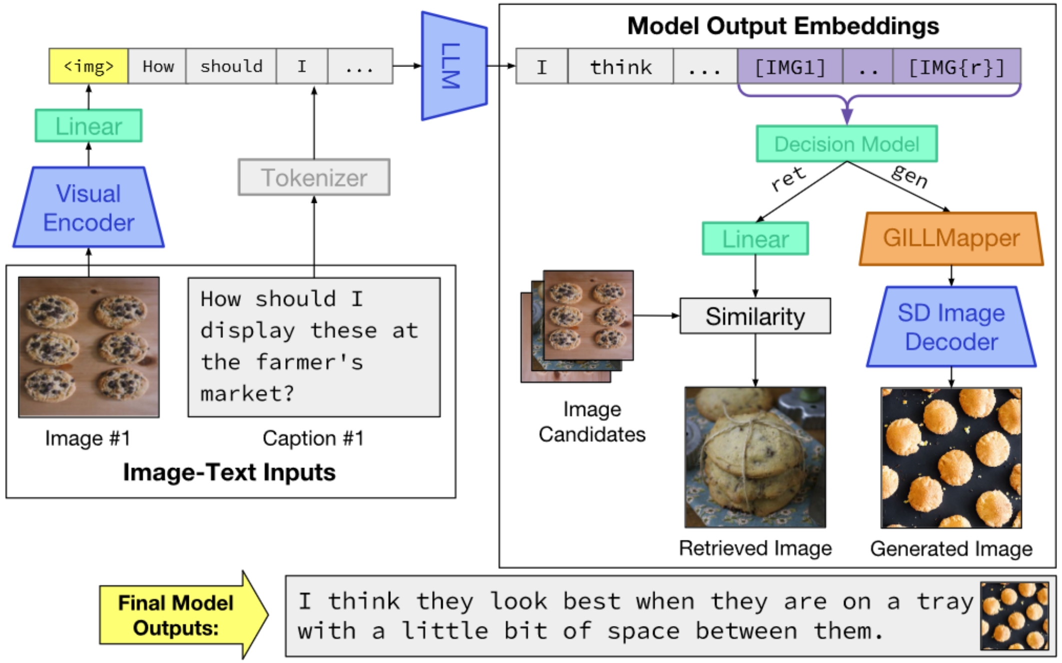


Figure 3: Inference time procedure for GILL. The model takes in image and text inputs, and produces text interleaved with image embeddings. After deciding whether to retrieve or generate for a particular set of tokens, it returns the appropriate image outputs.

Generating Images with Multimodal Language Models

Jing Yu Koh
Carnegie Mellon University
jingyuk@cs.cmu.edu

Daniel Fried
Carnegie Mellon University
dfried@cs.cmu.edu

Ruslan Salakhutdinov
Carnegie Mellon University
rsalakhu@cs.cmu.edu

Abstract

We propose a method to fuse frozen text-only large language models (LLMs) with pre-trained image encoder and decoder models, by mapping between their embedding spaces. Our model demonstrates a wide suite of multimodal capabilities: image retrieval, novel image generation, and multimodal dialogue. Ours is the first approach capable of conditioning on arbitrarily interleaved image and text inputs to generate coherent image (and text) outputs. To achieve strong performance on image generation, we propose an efficient mapping network to ground the LLM to an off-the-shelf text-to-image generation model. This mapping network translates hidden representations of text into the embedding space of the visual models, enabling us to leverage the strong text representations of the LLM for visual outputs. Our approach outperforms baseline generation models on tasks with longer and more complex language. In addition to novel image generation, our model is also capable of image retrieval from a prespecified dataset, and decides whether to retrieve or generate at inference time. This is done with a learnt decision module which conditions on the hidden representations of the LLM. Our model exhibits a wider range of capabilities compared to prior multimodal language models. It can process image-and-text inputs, and produce retrieved images, generated images, and generated text — outperforming non-LLM based generation models across several text-to-image tasks that measure context dependence.

1 Introduction

https://arxiv.org/abs/2305.17216

Autoregressive language models (LMs) and large language models (LLMs) trained on text corpora have shown impressive abilities to efficiently adapt to other modalities. Prior work showcased the effectiveness of grounding text-only LMs to images for vision-and-language tasks [56, 4, 29, 33, 31, 35], to embodied settings for robotics [3, 18], offline reinforcement learning [48], and more. These methods typically keep most of the LLM weights frozen. This allows them to leverage the capabilities that the LLM learns during large scale text-only pretraining, such as the ability to learn from in-context examples [9], more effectively process longer context, and condition on inputs more strongly.

In this work, we tackle the task of extending multimodal language models to generate novel images. Our approach, **G**enerating **I**mages with **L**arge **L**anguage Models (GILL), is capable of processing arbitrarily interleaved image-and-text inputs to generate text, retrieve images, and generate novel images (Fig. 1). Our findings show that it is possible to efficiently map the output embedding space of a frozen text-only LLM to that of a frozen generation model (in this work, Stable Diffusion [49]) despite both models using entirely different text encoders. We achieve this by finetuning a small number of parameters on image-caption pairs [52], in contrast to other methods which require interleaved image-text data [4, 2]. Our approach is computationally efficient and does not require running the image generation model at training time. To achieve strong image generation performance, we propose efficient architectural changes to learn the LLM-to-generation mapping effectively with the GILLMapper module. GILLMapper is a lightweight Transformer [57] conditioned on special

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

GPT-4V

OpenAI's multimodal LLM

"GPT-4 doesn't take videos as input directly, but we can use vision and the new 128K context window to describe the static frames of a whole video at once."

https://cookbook.openai.com/

Gonzalo Espinoza Graham 🏴‍☠️
@geepytee

GPT-4V + TTS = AI Sports narrator 🪄⚽

Passed every frame of a football video to gpt-4-vision-preview, and with some simple prompting asked to generate a narration

No edits, this is as it came out from the model (aka can be SO MUCH BETTER)

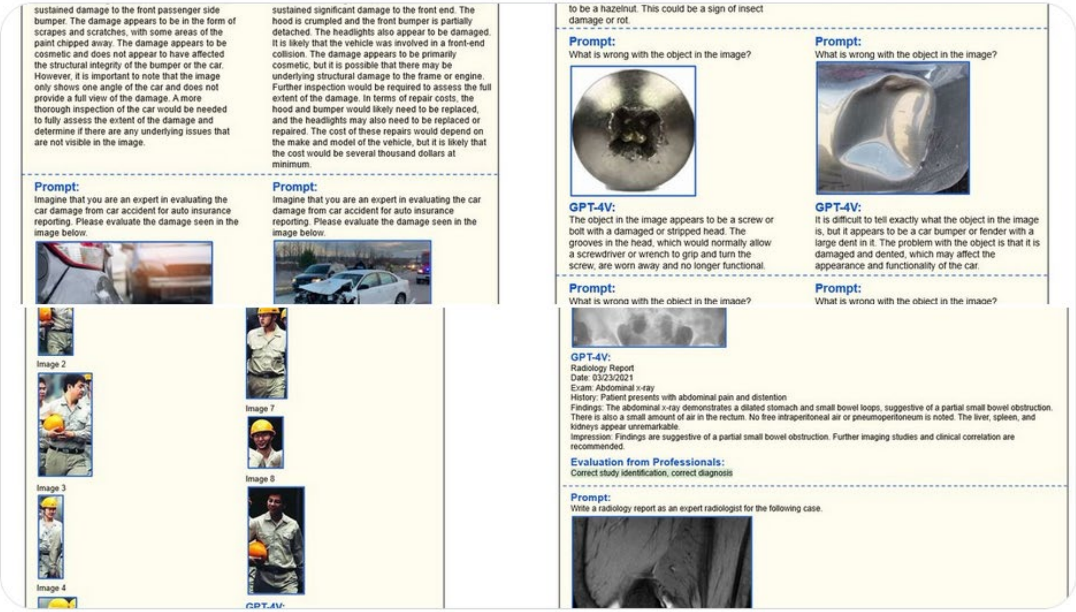
3:45 AM · Nov 7, 2023 · **2.6M** Views

https://x.com/geepytee/status/1721705524176257296?s=20

Ethan Mollick @emollick · Oct 2, 2023

Giving AI vision is going to have some wide-ranging effects on many industries.

It still makes mistakes, but this paper tests a lot of real-world uses for GPT-4V, from auto insurance reporting to safety inspections to industrial engineering to radiology. arxiv.org/abs/2309.17421



22 185 699 178K

https://arxiv.org/abs/2309.17421

Helping GPT-4V see better

Adding number labels to image objects improves GPT-4V's image understanding.

The labels can be autogenerated using an open source image segmentation model.

Set-of-Mark Prompting Unleashes Extraordinary Visual Grounding in GPT-4V

Jianwei Yang[1*‡], Hao Zhang[2*], Feng Li[2*], Xueyan Zou[3*], Chunyuan Li[1], Jianfeng Gao[1]
[1] Microsoft Research, Redmond [2] HKUST [3] University of Wisconsin-Madison
*Core Contributor ‡ Project Lead

Playground: https://som-gpt4v.github.io/

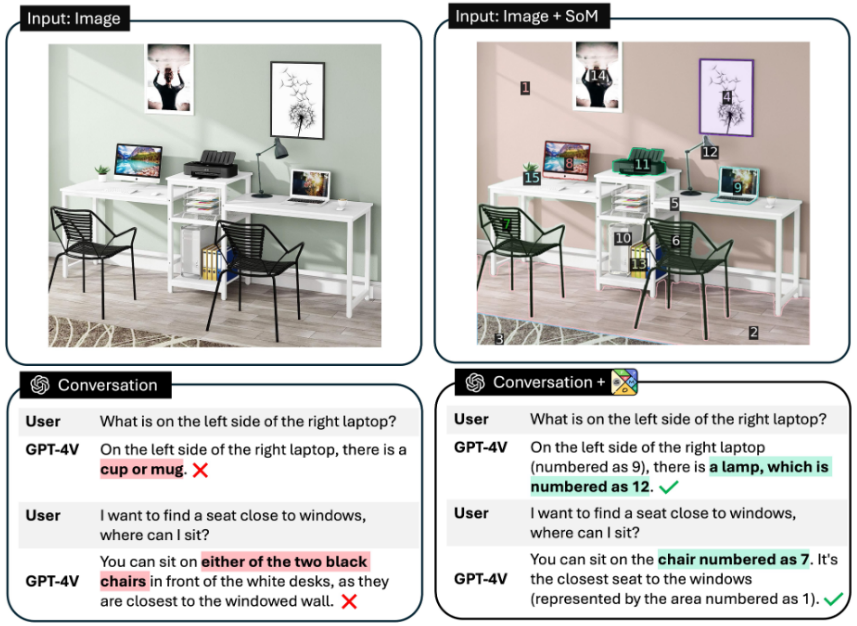


Figure 1: Comparisons of GPT-4V prompting techniques: (left) standard prompting and (right) the proposed *Set-of-Mark Prompting*. Simply overlaying ids on image regions unleashes visual grounding and corrects answers for GPT-4V. *Note that no marks are leaked to user text inputs.*.

Abstract

We present *Set-of-Mark (SoM)*, a new visual prompting method, to unleash the visual grounding abilities of large multimodal models (LMMs), such as GPT-4V. As illustrated in Fig. 1 (right), we employ off-the-shelf interactive segmentation models, such as SEEM/SAM, to partition an image into regions at different levels of granularity, and overlay these regions with a set of marks *e.g.*, alphanumerics, masks, boxes. Using the marked image as input, GPT-4V can answer the questions that require visual grounding. We perform a comprehensive empirical study to validate the effectiveness of SoM on a wide range of fine-grained vision and multimodal tasks. For example, our experiments show that GPT-4V with SoM outperforms the state-of-the-art fully-finetuned referring expression comprehension and segmentation model on RefCOCOg in a zero-shot setting. Code for *SoM* prompting is made public here: https://github.com/microsoft/SoM.

Gemini (Dec 2023)

Google Deepmind's multimodal LLMs

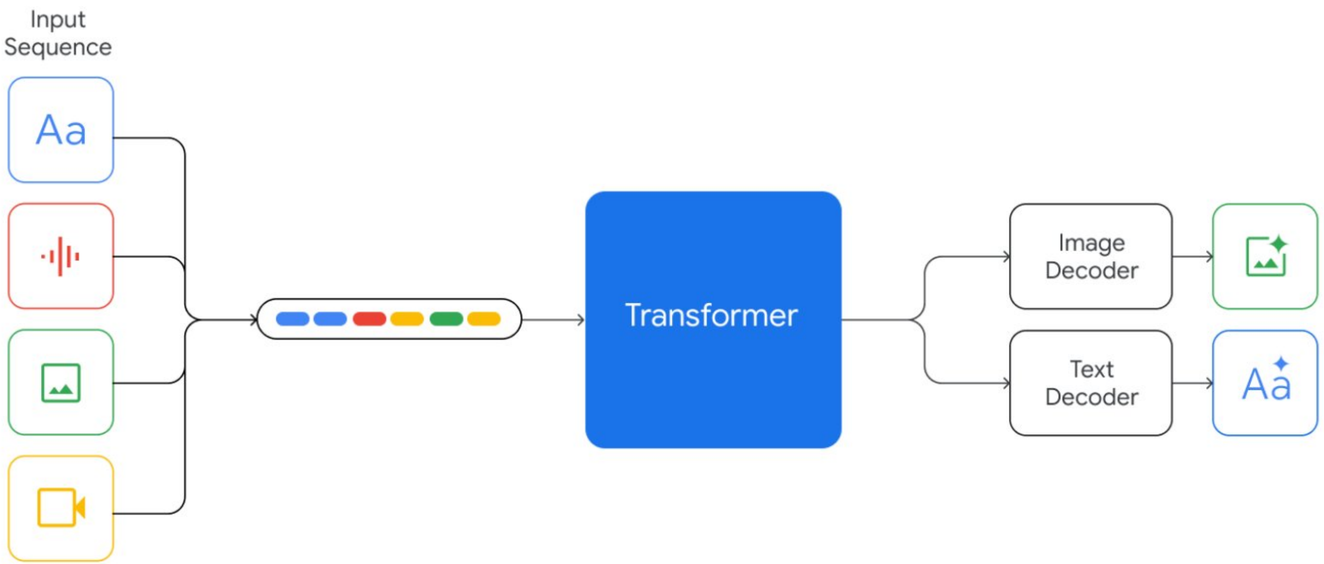


Figure 2 | Gemini supports interleaved sequences of text, image, audio, and video as inputs (illustrated by tokens of different colors in the input sequence). It can output responses with interleaved image and text.

https://www.youtube.com/watch?v=jV1vkHv4zq8

https://storage.googleapis.com/deepmind-media/gemini/gemini_1_report.pdf

Google DeepMind

Gemini: A Family of Highly Capable Multimodal Models

Gemini Team, Google[1]

This report introduces a new family of multimodal models, Gemini, that exhibit remarkable capabilities across image, audio, video, and text understanding. The Gemini family consists of Ultra, Pro, and Nano sizes, suitable for applications ranging from complex reasoning tasks to on-device memory-constrained use-cases. Evaluation on a broad range of benchmarks shows that our most-capable Gemini Ultra model advances the state of the art in 30 of 32 of these benchmarks — notably being the first model to achieve human-expert performance on the well-studied exam benchmark MMLU, and improving the state of the art in every one of the 20 multimodal benchmarks we examined. We believe that the new capabilities of Gemini models in cross-modal reasoning and language understanding will enable a wide variety of use cases and we discuss our approach toward deploying them responsibly to users.

1. Introduction

We present Gemini, a family of highly capable multimodal models developed at Google. We trained Gemini jointly across image, audio, video, and text data for the purpose of building a model with both strong generalist capabilities across modalities alongside cutting-edge understanding and reasoning performance in each respective domain.

Gemini 1.0, our first version, comes in three sizes: Ultra for highly-complex tasks, Pro for enhanced performance and deployability at scale, and Nano for on-device applications. Each size is specifically tailored to address different computational limitations and application requirements. We evaluate the performance of Gemini models on a comprehensive suite of internal and external benchmarks covering a wide range of language, coding, reasoning, and multimodal tasks.

Gemini advances state-of-the-art in large-scale language modeling (Anil et al., 2023; Brown et al., 2020; Chowdhery et al., 2023; Hoffmann et al., 2022; OpenAI, 2023a; Radford et al., 2019; Rae et al., 2021), image understanding (Alayrac et al., 2022; Chen et al., 2022; Dosovitskiy et al., 2020; OpenAI, 2023b; Reed et al., 2022; Yu et al., 2022a), audio processing (Radford et al., 2023; Zhang et al., 2023), and video understanding(Alayrac et al., 2022; Chen et al., 2023). It also builds on the work on sequence models (Sutskever et al., 2014), a long history of work in deep learning based on neural networks (LeCun et al., 2015), and machine learning distributed systems (Barham et al., 2022; Bradbury et al., 2018; Dean et al., 2012) that enable large-scale training.

Our most capable model, Gemini Ultra, achieves new state-of-the-art results in 30 of 32 benchmarks we report on, including 10 of 12 popular text and reasoning benchmarks, 9 of 9 image understanding benchmarks, 6 of 6 video understanding benchmarks, and 5 of 5 speech recognition and speech translation benchmarks. Gemini Ultra is the first model to achieve human-expert performance on MMLU (Hendrycks et al., 2021a) — a prominent benchmark testing knowledge and reasoning via a suite of exams — with a score above 90%. Beyond text, Gemini Ultra makes notable advances on challenging multimodal reasoning tasks. For example, on the recent MMMU benchmark (Yue et al., 2023), that comprises questions about images on multi-discipline tasks requiring college-level subject

[1]See Contributions and Acknowledgments section for full author list. Please send correspondence to gemini-1-report@google.com

Generating videos

Diffusion models dominate

- One can use a GPT to generate video frame-by-frame and patch-by-patch, but limited context length is a bottleneck for longer videos
- U-Net denoisers scale better, but longer videos are still a challenge

How to control?

A text prompt goes only so far.

DragNUWA allows the user to specify control trajectories.

DRAGNUWA: FINE-GRAINED CONTROL IN VIDEO GENERATION BY INTEGRATING TEXT, IMAGE, AND TRAJECTORY

Shengming Yin[1*] **Chenfei Wu**[2*] **Jian Liang**[3] **Jie Shi**[3] **Houqiang Li**[1] **Gong Ming**[2] **Nan Duan**[2†]
[1]University of Science and Technology of China [2]Microsoft Research Asia [3]Peking University
{sheyin@mail.,lihq}@ustc.edu.cn, {chewu,migon,nanduan}@microsoft.com, {j.liang@stu.,jieshi@}pku.edu.cn

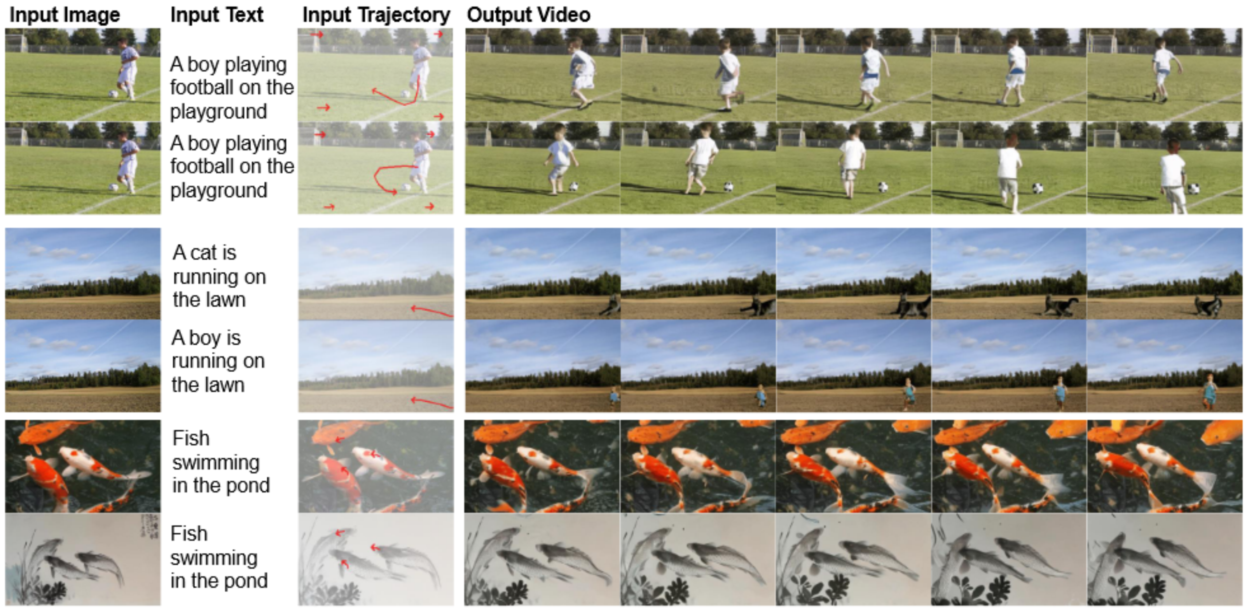
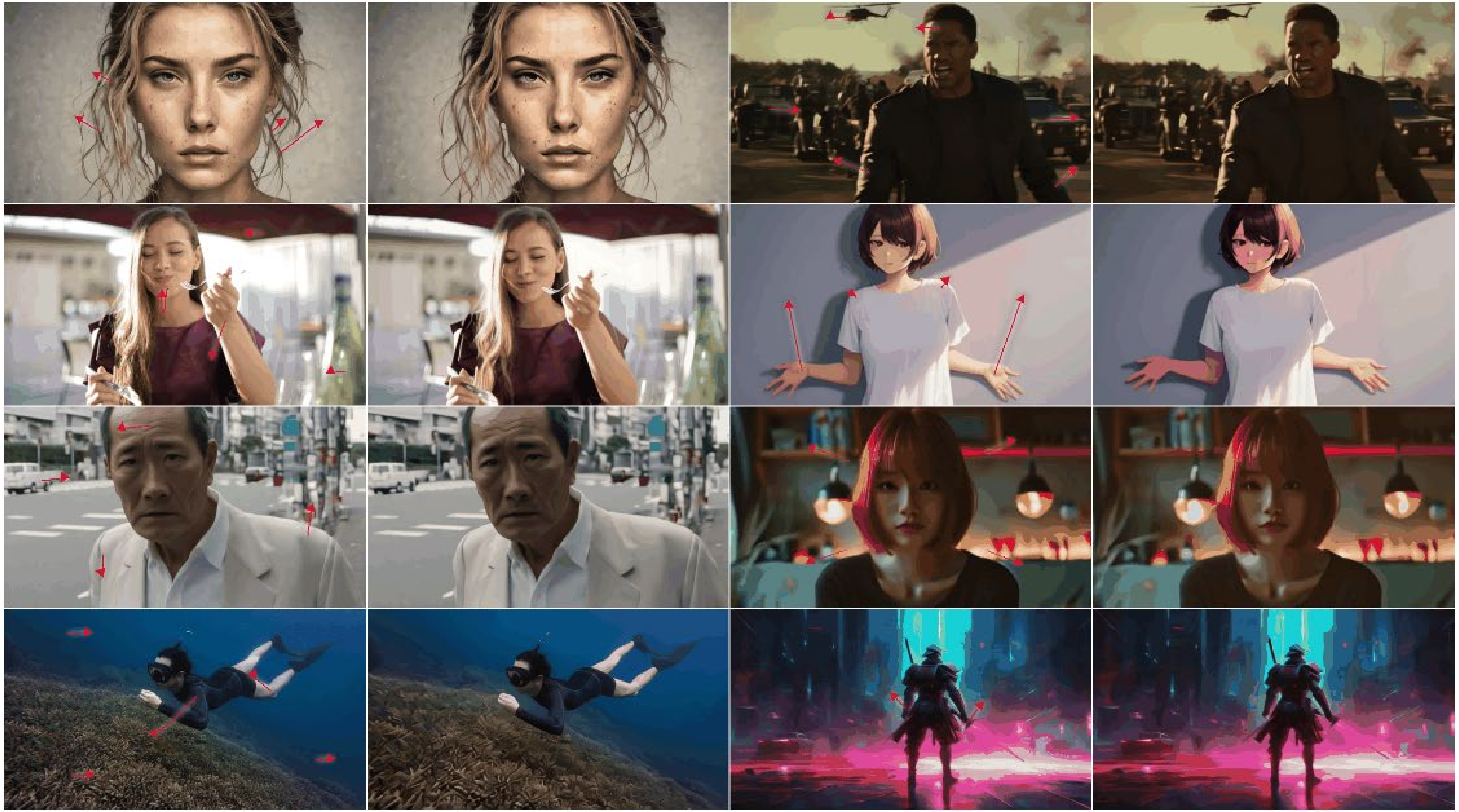


Figure 1: DragNUWA integrates text, image, and trajectory controls to achieve controllable video generation from semantic, spatial, and temporal perspectives. The three groups of examples demonstrate the impact of altering one control while keeping the other two fixed. The first group (Row 1-2) displays the control of complex trajectories, including complex motions (red curved arrows) and camera movements (red rightward arrows). The second group (Row 3-4) illustrates the influence of language control, pairing different text with the same image and trajectory to achieve the effect of introducing new objects in the images. The third group (Row 5-6) demonstrates the impact of image control, showcasing the generation of both real-world and artistic videos.

https://arxiv.org/abs/2308.08089

DragNUWA 1.6 will come soon (Updated on Jan 30, 2024)

- Consistency in human facial representation



https://github.com/ProjectNUWA/DragNUWA

Stable Video Diffusion

Released in November 2023

Image-to-video, i.e., conditioned both using text and a reference image

Stable Video Diffusion: Scaling Latent Video Diffusion Models to Large Datasets

Andreas Blattmann* Tim Dockhorn* Sumith Kulal* Daniel Mendelevitch
Maciej Kilian Dominik Lorenz Yam Levi Zion English Vikram Voleti
Adam Letts Varun Jampani Robin Rombach
Stability AI

"A robot dj is playing the turntables, in heavy raining futuristic tokyo, rooftop, sci-fi, fantasy"

"An exploding cheese house"

"A fat rabbit wearing a purple robe walking through a fantasy landscape"

Figure 1. **Stable Video Diffusion samples.** *Top:* Text-to-Video generation. *Middle:* (Text-to-)Image-to-Video generation. *Bottom:* Multi-view synthesis via Image-to-Video finetuning.

Abstract

We present Stable Video Diffusion — a latent video diffusion model for high-resolution, state-of-the-art text-to-video and image-to-video generation. Recently, latent diffusion models trained for 2D image synthesis have been turned into generative video models by inserting temporal layers and finetuning them on small, high-quality video datasets. However, training methods in the literature vary widely, and the field has yet to agree on a unified strategy for curating video data. In this paper, we identify and evaluate three different stages for successful training of video LDMs: text-to-image pretraining, video pretraining, and high-quality video finetuning. Furthermore, we demonstrate the necessity of a well-curated pretraining dataset for generating high-quality videos and present a systematic curation process to train a strong base model, including captioning and filtering strategies. We then explore the impact of finetuning our base model on high-quality data and train a text-to-video model that is competitive with closed-source video generation. We also show that our base

* Equal contributions.

https://colab.research.google.com/github/mkshing/notebooks/blob/main/stable_video_diffusion_img2vid.ipynb

You reposted

Steve Mills @SteveMills · Nov 24, 2023

SDV (Stable Diffusion Image To Video) Google Colab available here for anyone who wants to play along at home.

colab.research.google.com/github/mkshing...

Generates 3 seconds of video in about 30 seconds using an A100 GPU on Colab+

No control of the actual video in any way at all (yet), but it...
Show more

runway Research ↓ Product ↓ Studios ↓ Customers Pricing Company ↓ LOG IN Sign Up — It's Free

AI Magic Tools

Gen-2

https://runwayml.com/ai-magic-tools/gen-2/

Create videos in any style you can imagine with Text to Video generation. If you can imagine it, you can generate it.

Try Runway for Free >

niceaunties @niceaunties · Dec 23, 2023

'Bear to Bull' (B2B)
A holiday parade of free animals in Auntieverse. Happy Holidays everyone 🙏

A prompt share and conversational collab with @pancakepie360 . Two videos, each made from own interpretation. Details in 🧵

https://x.com/niceaunties/status/1738592581448344038?s=20

https://youtube.com/shorts/UX2uqSIf_2w?si=T333ktMr8vG_SeLU

niceaunties @niceaunties · Dec 23, 2023

Theme: pancake + auntie
Prompt : pancake + auntie
Images: dalle3, @Magnific_AI x auntie
Animation: @runwayml x auntie
Edit: @capcutapp x auntie
Music : @suno_ai_ x auntie
Location: Auntieverse

ChristianF
@ChristianF369

Pushed #gen2 again & made a movie trailer.
#aicinema is finally here!
Every shot made from text prompts, except one iconic shot you all know, done with #gen1
Made possible by @runwayml
@bazluhrmann your movies been a great inspiration! 😍🎞️🙏
Voices: @elevenlabsio
#aianimation

6:20 PM · Apr 27, 2023 · **846.7K** Views

ChristianF @ChristianF369 · Apr 28, 2023
Some stats about used AI Tools:
☑️500+ shots generated with #gen2 beta to get 65 shots that made it into the movie
☑️5000 credits used to generate 3 custom voices with @elevenlabsio, that fitted my taste of timbre and likeness
☑️Initial idea by me, script co-created with #Chatgpt

6 13 113 35K

ChristianF @ChristianF369 · Apr 28, 2023
Some stats about the film edit process:
☑️Music: the most important part for me! Some shots inspired me to find 2 tracks, put them together with the voices first, imagining the film in my head only
☑️Pace, timing, narration - all of that was done with the soundtrack first.
👇

2 6 65 20K

ChristianF @ChristianF369 · Apr 28, 2023
☑️Film editing with about 250-300 shots that I already had, putting them together like a puzzle in several hours
☑️Finetuning prompts for shots that needed improvement, re-generating, also generating new once to tell the story better
☑️Sound FX to enhance athmosphere & story
👇

1 4 59 17K

ChristianF @ChristianF369 · Apr 28, 2023
☑️Final refinements and minor tweaks to the film
☑️Designing titles & Thumbnail with #midjourneyv5 and #photoshop
☑️Share with the world

For those who believe that AI will do everything for you: No!
It can... and will soon.
I`ll always prefer to put my own heart & soul in.🙏

1 3 69 14K

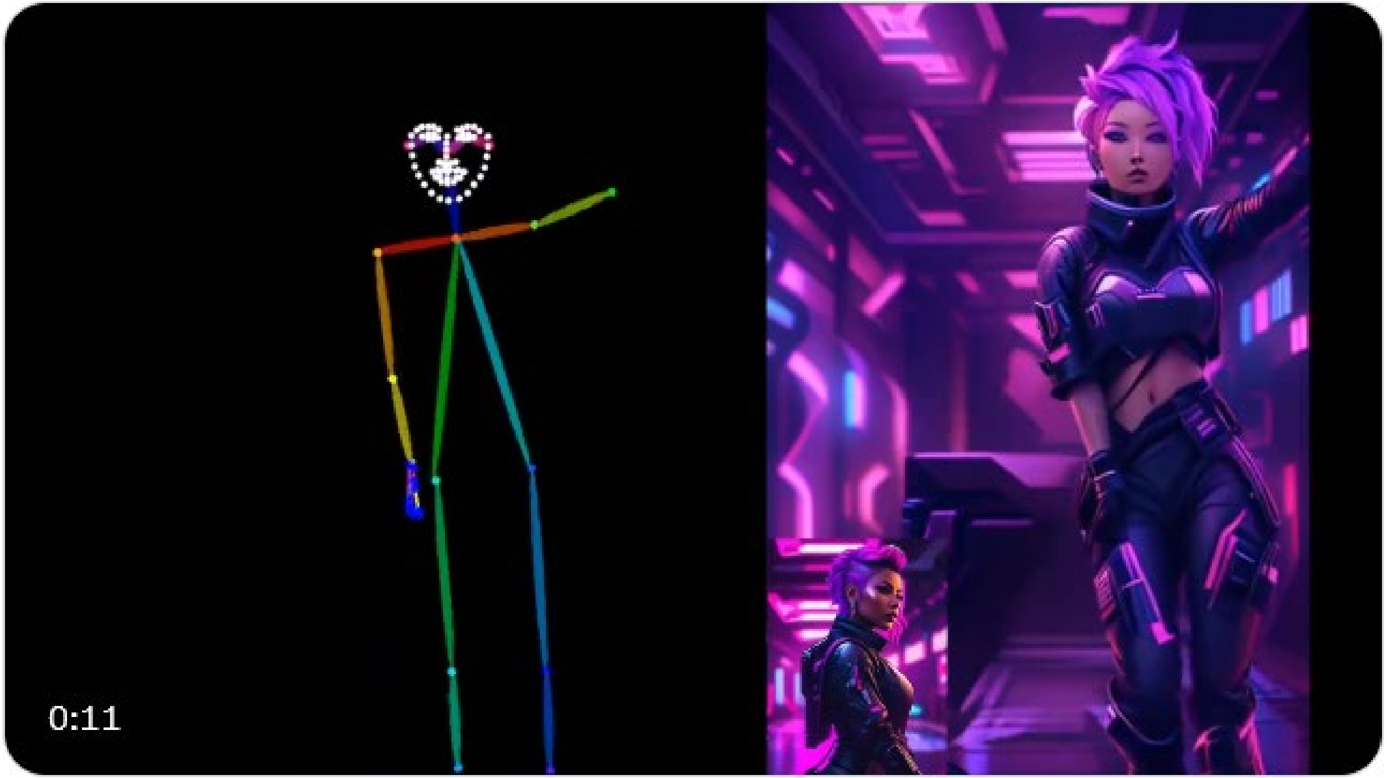
https://twitter.com/ChristianF369/status/1651607149804498946

AK @_akhaliq · Dec 28, 2023 ·

Alibaba releases DreaMoving demo on Hugging Face

A Human Video Generation Framework
based on Diffusion Models

demo: huggingface.co/spaces/jiayong...



23 427 1.7K 180K

https://x.com/_akhaliq/status/1740380744726594003?s=20

Steve Mills @SteveMills · Dec 6, 2023

Imagine a "Channel" in the future that creates an endless stream of 100% new, totally compelling visuals and music on demand.

Today's experiment involved an exploration of @runwayml Gen-2 with AI driven prompt discovery. 100% text to image.

Sound On 🔊
Music from Google...
Show more

https://www.youtube.com/watch?v=ZYXUib51nTc

104 170 708 89K

Steve Mills @SteveMills · 4h

The Channel is Ainfinite.tv Live NOW!

AnimateDiff can add motion capabilities to any finetuned StableDiffusion variant

Demos: https://animatediff.github.io/

Code: https://github.com/guoyww/AnimateDiff

Paper: https://arxiv.org/abs/2307.04725

Also integrated in ComfyUI and Automatic1111

AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning

Click to Play the Animations!

Generated with CivitAI models: ToonYou Lyriel majicMIX Realistic RCNZ Cartoon 3d

Open video models empowering new innovations

ClipDraw-style Bezier curve renderings guided by a video Diffusion model

https://livesketch.github.io/

Breathing Life Into Sketches Using Text-to-Video Priors

Rinon Gal*[,1,2] Yael Vinker*[,1] Yuval Alaluf[1] Amit Bermano[1]
Daniel Cohen-Or[1] Ariel Shamir[3] Gal Chechik[2]

[1]Tel-Aviv University [2]NVIDIA [3]Reichman University

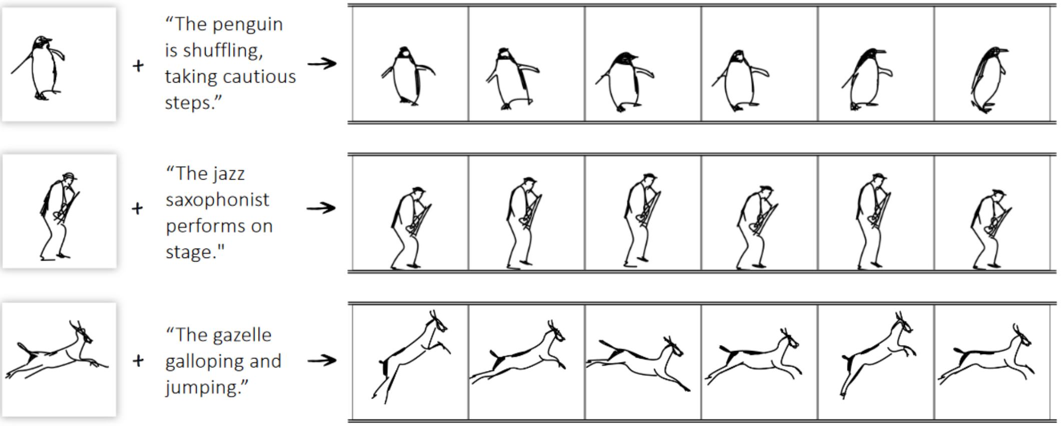


Figure 1. Given a still sketch in vector format and a text prompt describing a desired action, our method automatically animates the drawing with respect to the prompt. Please see the full animations in our project page: https://livesketch.github.io/

Abstract

A sketch is one of the most intuitive and versatile tools humans use to convey their ideas visually. An animated sketch opens another dimension to the expression of ideas and is widely used by designers for a variety of purposes. Animating sketches is a laborious process, requiring extensive experience and professional design skills. In this work, we present a method that automatically adds motion to a single-subject sketch (hence, "breathing life into it"), merely by providing a text prompt indicating the desired motion. The output is a short animation provided in vector representation, which can be easily edited. Our method does not require extensive training, but instead leverages the motion prior of a large pretrained text-to-video diffusion model using a score-distillation loss to guide the placement of strokes. To promote natural and smooth motion and to better preserve the sketch's appearance, we model the learned motion through two components. The first governs small local deformations and the second controls global affine transformations. Surprisingly, we find that even models that struggle to generate sketch videos on their own can still serve as a useful backbone for animating abstract representations.

*Indicates Equal Contribution. Order determined by coin flip.

1. Introduction

Sketches serve as a fundamental and intuitive tool for visual expression and communication [3, 20, 26]. Sketches capture the essence of visual entities with a few strokes, allowing humans to communicate abstract visual ideas. In this paper, we propose a method to "breathe life" into a static sketch by generating semantically meaningful short videos from it. Such animations can be useful for storytelling, illustrations, websites, presentations, and just for fun.

Animating sketches using conventional tools (such as Adobe Animate and Toon Boom) is challenging even for experienced designers [76], requiring specific artistic expertise. Hence, long-standing research efforts in computer graphics sought to develop automatic tools to simplify this process. However, these tools face multiple hurdles, such as a need to identify the semantic component of the sketch, or learning to create motion that appears natural. As such, existing methods commonly rely on user-annotated skeletal key points [17, 74] or user-provided reference motions that align with the sketch semantics [9, 76, 88].

In this work, we propose to bring a given static sketch to life, based on a textual prompt, without the need for any human annotations or explicit reference motions. We do so by leveraging a pretrained text-to-video diffusion model [43]. Several recent works propose using the prior of such mod-

Generating audio

Audio generation

- Early attempts: GANs, RNNs. No conditioning on text.
- 2020: OpenAI Jukebox: First to tokenize audio using vector quantization
- 2022: Transformers work well (LLM + AE + RVQ tokenization). Music is naturally sequential and thus conveniently modeled as a token sequence
- 2023: Diffusion models excel. Instead of CLIP, one uses CLAP (Contrastive Language Audio Pretraining) embeddings for the classifier-free guidance

[Submitted on 16 Nov 2018]

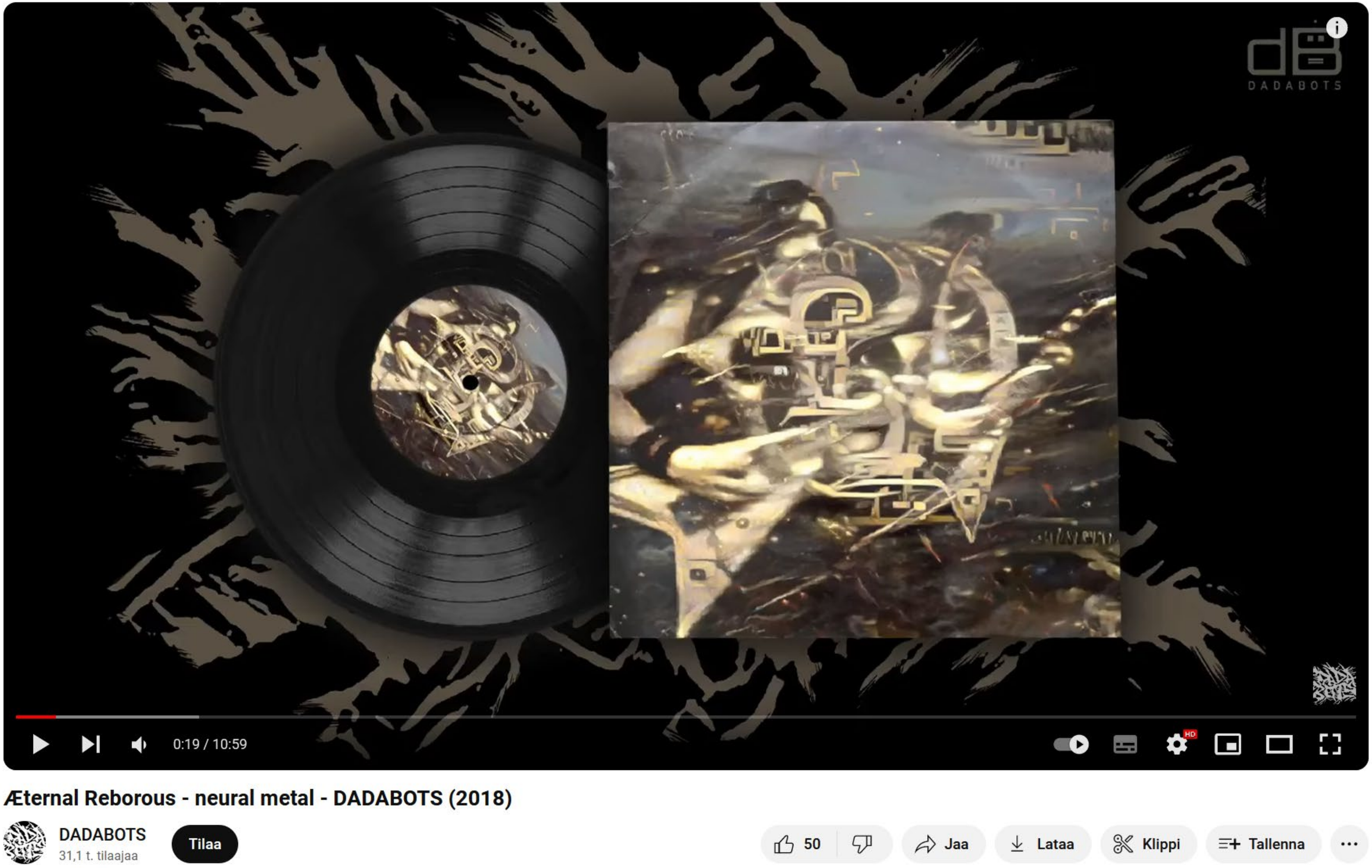
Generating Albums with SampleRNN to Imitate Metal, Rock, and Punk Bands

CJ Carr, Zack Zukowski

This early example of neural synthesis is a proof-of-concept for how machine learning can drive new types of music software. Creating music can be as simple as specifying a set of music influences on which a model trains. We demonstrate a method for generating albums that imitate bands in experimental music genres previously unrealized by traditional synthesis techniques (e.g. additive, subtractive, FM, granular, concatenative). Raw audio is generated autoregressively in the time-domain using an unconditional SampleRNN. We create six albums this way. Artwork and song titles are also generated using materials from the original artists' back catalog as training data. We try a fully-automated method and a human-curated method. We discuss its potential for machine-assisted production.

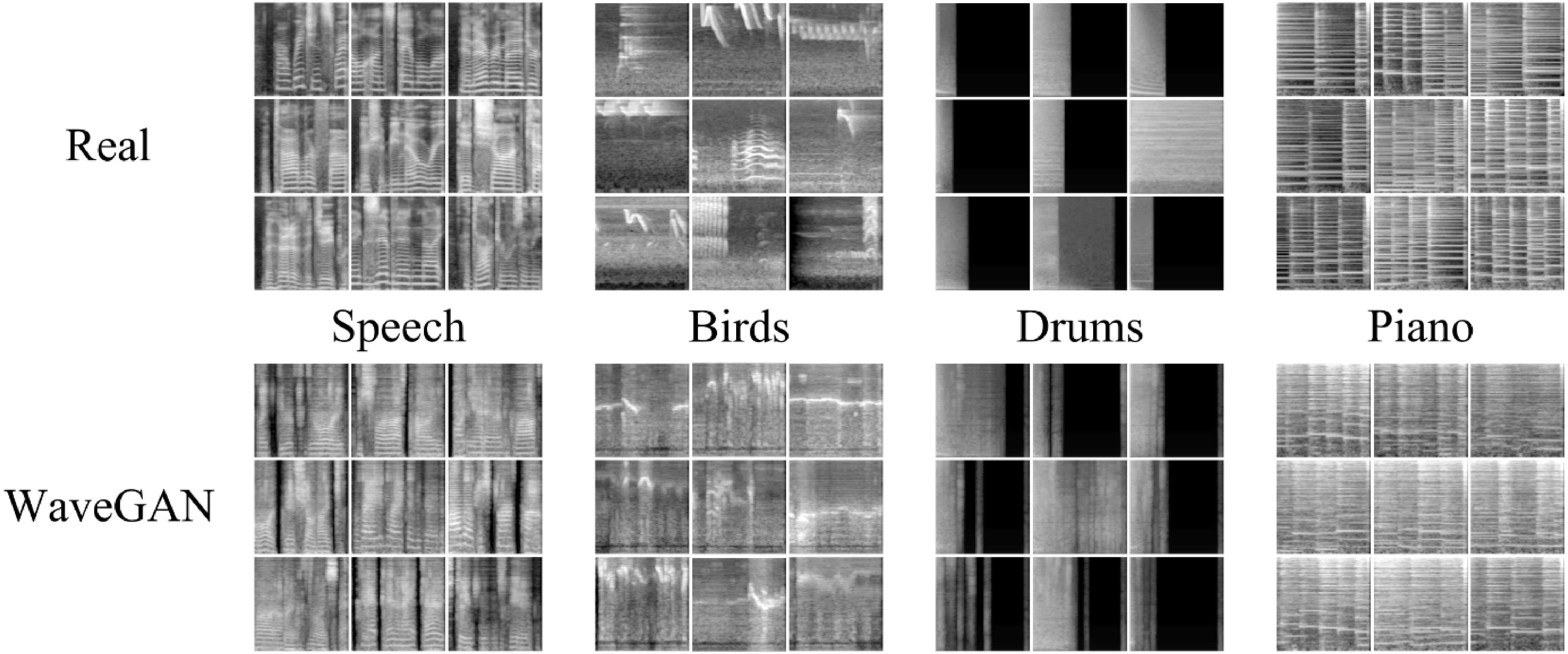
Comments:	3 pages
Subjects:	**Sound (cs.SD)**; Audio and Speech Processing (eess.AS)
Journal reference:	Proceedings of the 6th International Workshop on Musical Metacreation (MUME 2018)
Cite as:	arXiv:1811.06633 **[cs.SD]**
	(or arXiv:1811.06633v1 **[cs.SD]** for this version)

http://dadabots.com



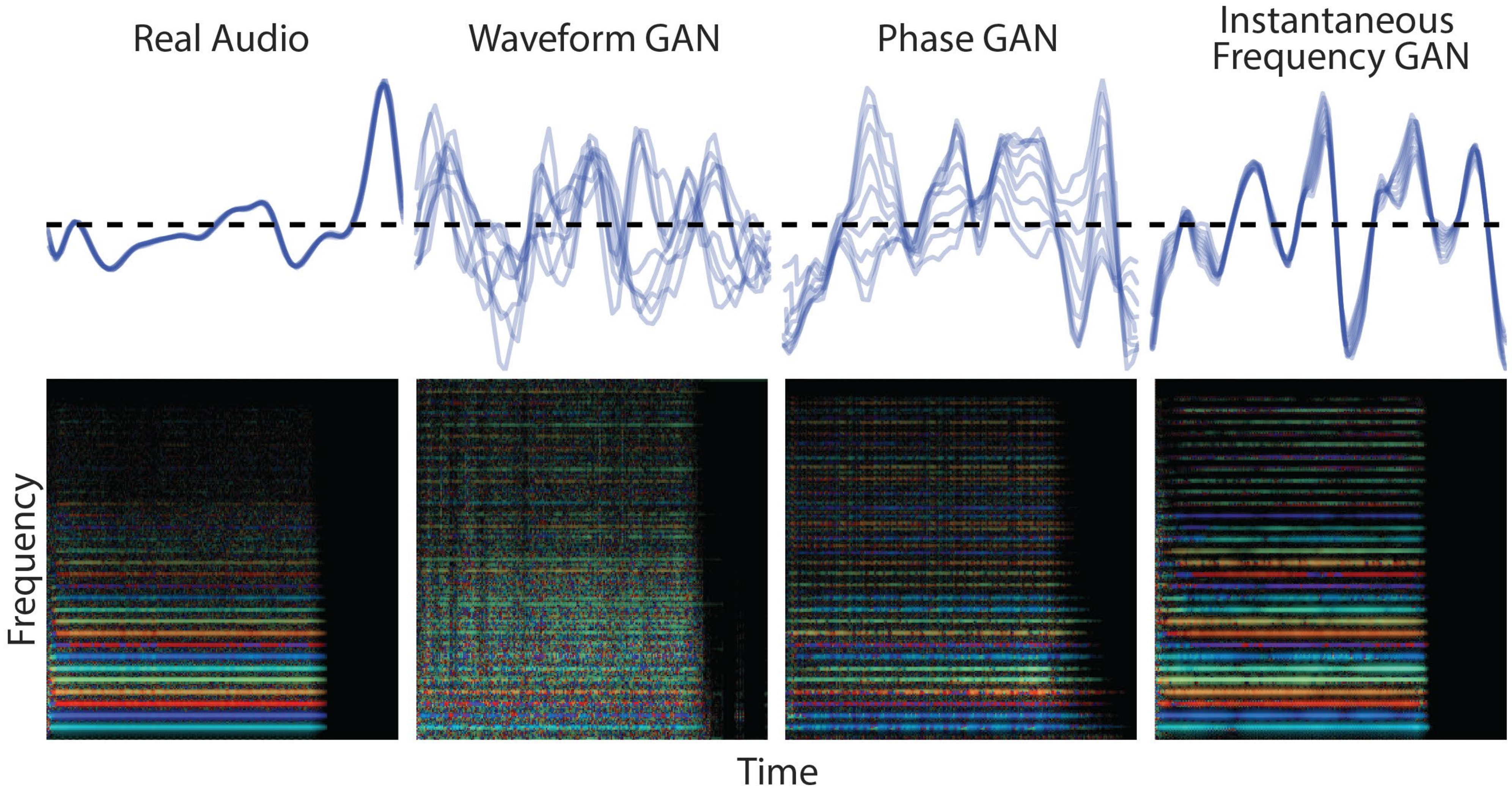
https://www.youtube.com/watch?v=4zyPl0lEjTg

Audio and GANs: WaveGAN



Code: https://github.com/chrisdonahue/wavegan, Interactive demo: https://chrisdonahue.com/wavegan/

GANSynth (2019)



https://magenta.tensorflow.org/gansynth

OpenAI Jukebox (2020)

LM + AE + VQ tokenization, at multiple resolutions. Unnecessarily complex, later RVQ-based approaches work better.

Results mindblowing at the time, but quite noisy due to the VQ imprecision

Paper, code & examples: https://openai.com/research/jukebox

Colab from OpenAI: https://colab.research.google.com/github/openai/jukebox/blob/master/jukebox/Interacting_with_Jukebox.ipynb

Jukebox: A Generative Model for Music

https://arxiv.org/abs/2005.00341

Prafulla Dhariwal [*1] **Heewoo Jun** [*1] **Christine Payne** [*1] **Jong Wook Kim** [1] **Alec Radford** [1] **Ilya Sutskever** [1]

Abstract

We introduce Jukebox, a model that generates music with singing in the raw audio domain. We tackle the long context of raw audio using a multi-scale VQ-VAE to compress it to discrete codes, and modeling those using autoregressive Transformers. We show that the combined model at scale can generate high-fidelity and diverse songs with coherence up to multiple minutes. We can condition on artist and genre to steer the musical and vocal style, and on unaligned lyrics to make the singing more controllable. We are releasing thousands of non cherry-picked samples, along with model weights and code.

1. Introduction

Music is an integral part of human culture, existing from the earliest periods of human civilization and evolving into a wide diversity of forms. It evokes a unique human spirit in its creation, and the question of whether computers can ever capture this creative process has fascinated computer scientists for decades. We have had algorithms generating piano sheet music (Hiller Jr & Isaacson, 1957; Moorer, 1972; Hadjeres et al., 2017; Huang et al., 2017), digital vocoders generating a singer's voice (Bonada & Serra, 2007; Saino et al., 2006; Blaauw & Bonada, 2017) and also synthesizers producing timbres for various musical instruments (Engel et al., 2017; 2019). Each captures a specific aspect of music generation: melody, composition, timbre, and the human voice singing. However, a single system to do it all remains elusive.

The field of generative models has made tremendous progress in the last few years. One of the aims of generative modeling is to capture the salient aspects of the data and to generate new instances indistinguishable from the true data The hypothesis is that by learning to produce the data we can learn the best features of the data[1]. We are surrounded by highly complex distributions in the visual, audio, and text domain, and in recent years we have developed advances in text generation (Radford et al.), speech generation (Xie et al., 2017) and image generation (Brock et al., 2019; Razavi et al., 2019). The rate of progress in this field has been rapid, where only a few years ago we had algorithms producing blurry faces (Kingma & Welling, 2014; Goodfellow et al., 2014) but now we now can generate high-resolution faces indistinguishable from real ones (Zhang et al., 2019b).

Generative models have been applied to the music generation task too. Earlier models generated music symbolically in the form of a pianoroll, which specifies the timing, pitch, velocity, and instrument of each note to be played. (Yang et al., 2017; Dong et al., 2018; Huang et al., 2019a; Payne, 2019; Roberts et al., 2018; Wu et al., 2019). The symbolic approach makes the modeling problem easier by working on the problem in the lower-dimensional space. However, it constrains the music that can be generated to being a specific sequence of notes and a fixed set of instruments to render with. In parallel, researchers have been pursuing the non-symbolic approach, where they try to produce music directly as a piece of audio. This makes the problem more challenging, as the space of raw audio is extremely high dimensional with a high amount of information content to model. There has been some success, with models producing piano pieces either in the raw audio domain (Oord et al., 2016; Mehri et al., 2017; Yamamoto et al., 2020) or in the spectrogram domain (Vasquez & Lewis, 2019). The key bottleneck is that modeling the raw audio directly introduces extremely long-range dependencies, making it computationally challenging to learn the high-level semantics of music. A way to reduce the difficulty is to learn a lower-dimensional encoding of the audio with the goal of losing the less important information but retaining most of the musical information. This approach has demonstrated some success in generating short instrumental pieces restricted to a set of a few instruments (Oord et al., 2017; Dieleman et al., 2018).

In this work, we show that we can use state-of-the-art deep generative models to produce a single system capable of generating diverse high-fidelity music in the raw audio domain, with long-range coherence spanning multiple minutes. Our approach uses a hierarchical VQ-VAE architecture (Razavi

[*]Equal contribution [1]OpenAI, San Francisco. Correspondence to: <jukebox@openai.com>.

[1]Richard Feynmann famously said, "What I cannot create, I do not understand"

Generative AI for audio

AudioCraft: generating high-quality audio and music from text

AudioCraft is a single-stop code base for all your generative audio needs: music, sound effects, and compression after training on raw audio signals. We have released controllable and high-quality models for music and audio generation from text inputs. It represents significant progress in the development of interactive AI systems enabling people to easily and naturally co-create with AI models.

https://ai.meta.com/resources/models-and-libraries/audiocraft/

The models

Open source, single-stage models, built with simplicity in mind

- ↗ **MusicGen paper**
- ↗ **AudioGen paper**
- ↗ **EnCodec paper**
- ↗ **Multi-Band Diffusion paper**

AudioCraft powers our audio compression and generation research and consists of three models: MusicGen, AudioGen, and EnCodec. MusicGen, which was trained with Meta-owned and specifically licensed music, generates music from text-based user inputs, while AudioGen, trained on public sound effects, generates audio from text-based user inputs. EnCodec, typically used foundationally in building MusicGen and AudioGen, is a state-of-the-art, real-time, high-fidelity audio codec that leverages neural networks to compress any kind of audio and reconstruct the original signal with high-fidelity. We further propose a diffusion-based approach to EnCodec to reconstruct the audio from the compressed representation with fewer artifacts.

September 2022: AudioGen

Generates both speech and audio effects.

LLM + AE and RVQ for tokenization

Uses classifier-free guidance for adjusting the quality-diversity tradeoff (had not been done before in LLM-based autoregressive sampling)

https://felixkreuk.github.io/audiogen/

AudioGen: Textually Guided Audio Generation

Felix Kreuk[1], **Gabriel Synnaeve**[1], **Adam Polyak**[1], **Uriel Singer**[1], **Alexandre Défossez**[1], **Jade Copet**[1], **Devi Parikh**[1], **Yaniv Taigman**[1], **Yossi Adi**[1,2]

[1]FAIR Team, Meta AI
[2]The Hebrew University of Jerusalem
felixkreuk@meta.com

https://arxiv.org/abs/2209.15352

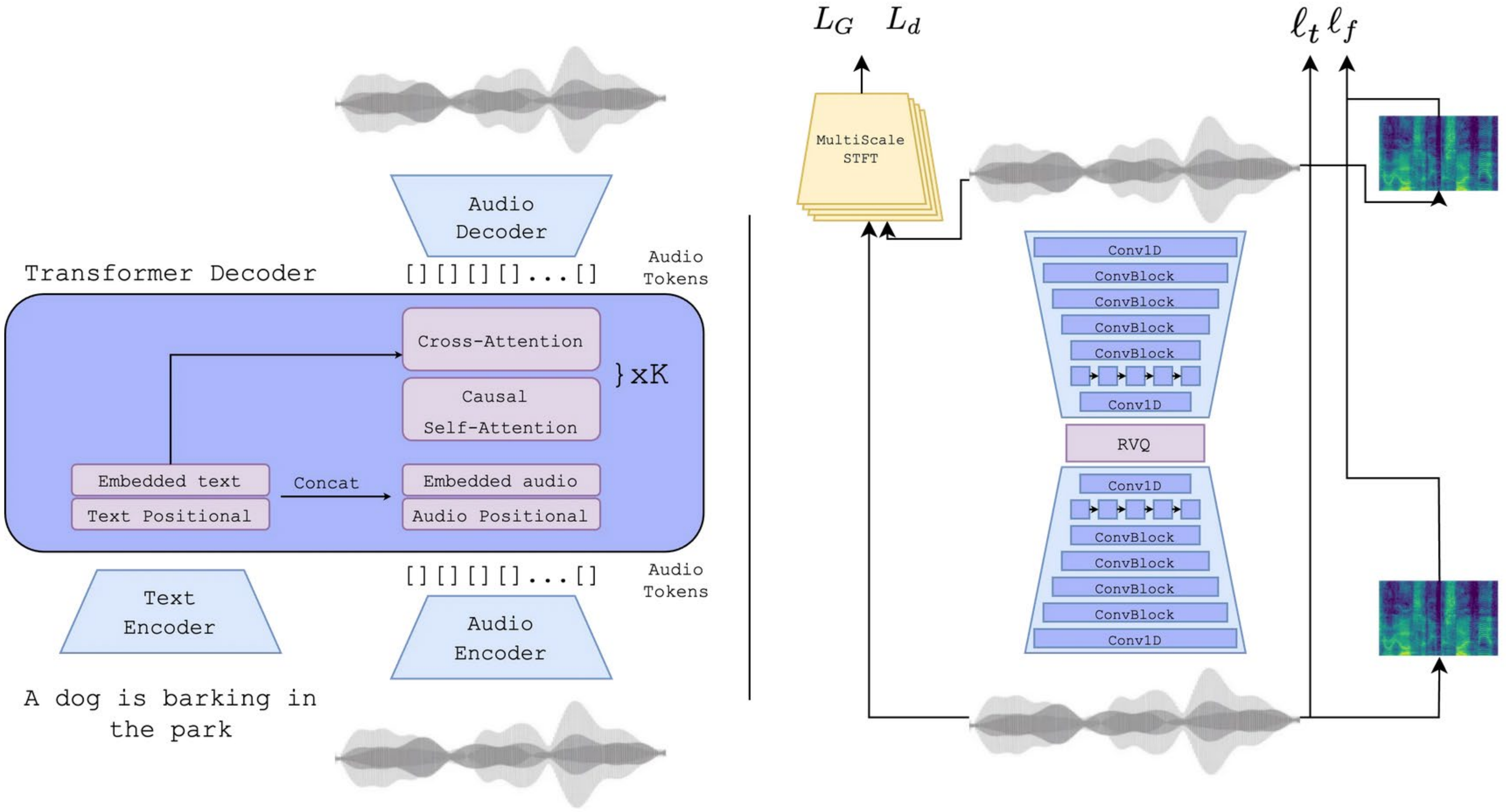
Abstract

We tackle the problem of generating audio samples conditioned on descriptive text captions. In this work, we propose AudioGen, an auto-regressive generative model that generates audio samples conditioned on text inputs. AudioGen operates on a learnt discrete audio representation. The task of text-to-audio generation poses multiple challenges. Due to the way audio travels through a medium, differentiating "objects" can be a difficult task (e.g., separating multiple people simultaneously speaking). This is further complicated by real-world recording conditions (e.g., background noise, reverberation, etc.). Scarce text annotations impose another constraint, limiting the ability to scale models. Finally, modeling high-fidelity audio requires encoding audio at high sampling rate, leading to extremely long sequences. To alleviate the aforementioned challenges we propose an augmentation technique that mixes different audio samples, driving the model to internally learn to separate multiple sources. We curated 10 datasets containing different types of audio and text annotations to handle the scarcity of text-audio data points. For faster inference, we explore the use of multi-stream modeling, allowing the use of shorter sequences while maintaining a similar bitrate and perceptual quality. We apply classifier-free guidance to improve adherence to text. Comparing to the evaluated baselines, AudioGen outperforms over both objective and subjective metrics. Finally, we explore the ability of the proposed method to generate audio continuation conditionally and unconditionally. Samples: https://felixkreuk.github.io/audiogen.

1 Introduction

Neural generative models have challenged the way we create digital content. From generating high-quality images (Karras et al., 2019; Park et al., 2019) and speech (Ren et al., 2021; Oord et al., 2016), through generating long textual spans (Brown et al., 2020; Zhang et al., 2022), to the recently proposed text prompted image generation (Ramesh et al., 2022; Rombach et al., 2022), these models have shown impressive results. This begs the question *what would be the audio equivalent to textually guided generative models?* From generating soundscapes to music or speech, a solution to this problem that is high fidelity, controllable, and diverse in its outputs, would be a useful addition to the modern toolbox of creators of movies, video games, and any virtual environments.

While image generation and audio generation have a lot in common, there are a few key differences. Audio is intrinsically a one dimensional signal and thus has less degrees of freedom to differentiate overlapping "objects" (Capon, 1969; Frost, 1972). Real-world audio inherently has reverberations, which makes the task of differentiating objects from the surrounding environment even harder. Moreover, psychoacoustic and psychovisual properties differ, for instance hearing "resolution" (equal-loudness) is U-shaped in frequencies with a dip at 4kHz and bump at 8kHz (Suzuki et al., 2003). Last but not least, the availability of audio data with textual descriptions is orders of magnitude below that of text-image paired data. This makes generating unseen audio compositions a hard task (e.g. generating an audio equivalent of an image of "an astronaut riding a horse in space").



Audio Decoder
Transformer Decoder
[] [] [] [] . . . []
Audio Tokens
Cross-Attention
} xK
Causal Self-Attention
Embedded text
Text Positional
Concat
Embedded audio
Audio Positional
[] [] [] [] . . . []
Audio Tokens
Text Encoder
Audio Encoder
A dog is barking in the park
L_G L_d
MultiScale STFT
ℓ_t ℓ_f
Conv1D
ConvBlock
ConvBlock
ConvBlock
ConvBlock
Conv1D
RVQ
Conv1D
ConvBlock
ConvBlock
ConvBlock
ConvBlock
Conv1D

Oct 2022: EnCodec

Further development of the AudioGen AE + RVQ tokenizer.

Many later papers use this as a good off-the-shelf audio tokenizer

High Fidelity Neural Audio Compression

Alexandre Défossez* *defossez@meta.com*
Meta AI, FAIR Team, Paris, France

Jade Copet* *jadecopet@meta.com*
Meta AI, FAIR Team, Paris, France

Gabriel Synnaeve† *gab@meta.com*
Meta AI, FAIR Team, Paris, France

Yossi Adi† *adiyoss@meta.com*
Meta AI, FAIR Team, Tel-Aviv, Israel

Abstract

We introduce a state-of-the-art real-time, high-fidelity, audio codec leveraging neural networks. It consists in a streaming encoder-decoder architecture with quantized latent space trained in an end-to-end fashion. We simplify and speed-up the training by using a single multiscale spectrogram adversary that efficiently reduces artifacts and produce high-quality samples. We introduce a novel loss balancer mechanism to stabilize training: the *weight* of a loss now defines the fraction of the overall gradient it should represent, thus decoupling the choice of this hyper-parameter from the typical scale of the loss. Finally, we study how lightweight Transformer models can be used to further compress the obtained representation by up to 40%, while staying faster than real time. We provide a detailed description of the key design choices of the proposed model including: training objective, architectural changes and a study of various perceptual loss functions. We present an extensive subjective evaluation (MUSHRA tests) together with an ablation study for a range of bandwidths and audio domains, including speech, noisy-reverberant speech, and music. Our approach is superior to the baselines methods across all evaluated settings, considering both 24 kHz monophonic and 48 kHz stereophonic audio. Code and models are available at github.com/facebookresearch/encodec.

1 Introduction

Recent studies suggest that streaming audio and video have accounted for the majority of the internet traffic in 2021 (82% according to (Cisco, 2021)). With the internet traffic expected to grow, audio compression is an increasingly important problem. In lossy signal compression we aim at minimizing the bitrate of a sample while also minimizing the amount of distortion according to a given metric, ideally correlated with human perception. Audio codecs typically employ a carefully engineered pipeline combining an encoder and a decoder to remove redundancies in the audio content and yield a compact bitstream. Traditionally, this is achieved by decomposing the input with a signal processing transform and trading off the quality of the components that are less likely to influence perception. Leveraging neural networks as trained transforms via an encoder-decoder mechanism has been explored by Morishima et al. (1990); Rippel et al. (2019); Zeghidour et al. (2021). Our research work is in the continuity of this line of work, with a focus on audio signals.

The problems arising in lossy neural compression models are twofold: first, the model has to represent a wide range of signals, such as not to overfit the training set or produce artifact laden audio outside its comfort zone. We solve this by having a large and diverse training set (described in Section 4.1), as well as discriminator networks (see Section 3.4) that serve as perceptual losses, which we study extensively in Section 4.5.1, Table 2. The other problem is that of compressing efficiently, both in compute time and in size.

*,†Equal contribution.

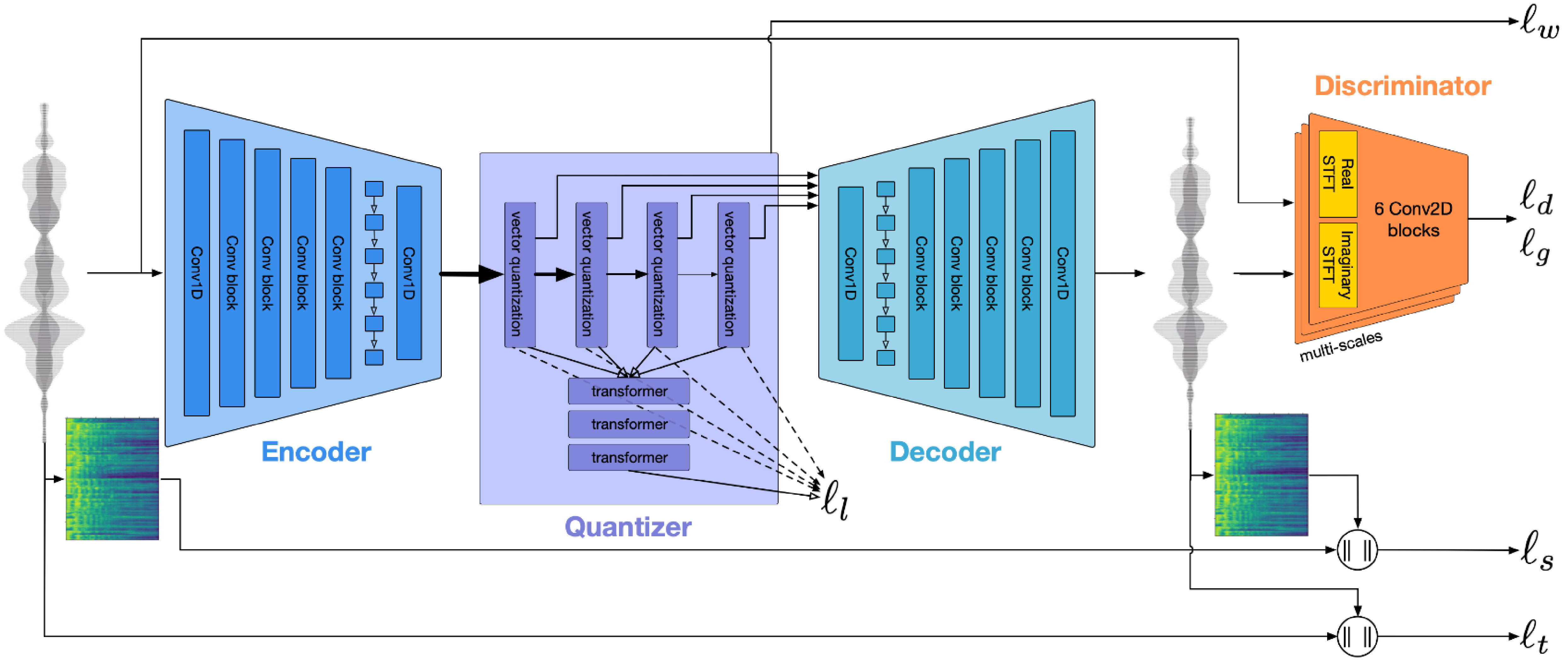


Figure 1: EnCodec : an encoder decoder codec architecture which is trained with reconstruction (ℓ_f and ℓ_t) as well as adversarial losses (ℓ_g for the generator and ℓ_d for the discriminator). The residual vector quantization commitment loss (ℓ_w) applies only to the encoder. Optionally, we train a small Transformer language model for entropy coding over the quantized units with ℓ_l, which reduces bandwidth even further.

January 2023: MusicLM

- Google joins the race with MusicLM
- Basically still the same "LLM trained with text and audio tokens" approach

MusicLM: Generating Music From Text

Andrea Agostinelli [*1] Timo I. Denk [*1]
Zalán Borsos [1] Jesse Engel [1] Mauro Verzetti [1] Antoine Caillon [2] Qingqing Huang [1] Aren Jansen [1]
Adam Roberts [1] Marco Tagliasacchi [1] Matt Sharifi [1] Neil Zeghidour [1] Christian Frank [1]

Abstract

We introduce MusicLM, a model for generating high-fidelity music from text descriptions such as *"a calming violin melody backed by a distorted guitar riff"*. MusicLM casts the process of conditional music generation as a hierarchical sequence-to-sequence modeling task, and it generates music at 24 kHz that remains consistent over several minutes. Our experiments show that MusicLM outperforms previous systems both in audio quality and adherence to the text descriptions. Moreover, we demonstrate that MusicLM can be conditioned on both text and a melody in that it can transform whistled and hummed melodies according to the style described in a text caption. To support future research, we publicly release MusicCaps, a dataset composed of 5.5k music-text pairs, with rich text descriptions provided by human experts.

google-research.github.io/seanet/musiclm/examples

1. Introduction

Conditional neural audio generation covers a wide range of applications, ranging from text-to-speech (Zen et al., 2013; van den Oord et al., 2016) to lyrics-conditioned music generation (Dhariwal et al., 2020) and audio synthesis from MIDI sequences (Hawthorne et al., 2022b). Such tasks are facilitated by a certain level of temporal alignment between the conditioning signal and the corresponding audio output. In contrast, and inspired by progress in text-to-image generation (Ramesh et al., 2021; 2022; Saharia et al., 2022; Yu et al., 2022), recent work has explored generating audio from sequence-wide, high-level captions (Yang et al., 2022; Kreuk et al., 2022) such as *"whistling with wind blowing"*. While generating audio from such coarse captions represents a breakthrough, these models remain limited to simple acoustic scenes, consisting of few acoustic events over a period of seconds. Hence, turning a single text caption into a rich audio sequence with long-term structure and many stems, such as a music clip, remains an open challenge.

AudioLM (Borsos et al., 2022) has recently been proposed as a framework for audio generation. Casting audio synthesis as a language modeling task in a discrete representation space, and leveraging a hierarchy of coarse-to-fine audio discrete units (or *tokens*), AudioLM achieves both high-fidelity and long-term coherence over dozens of seconds. Moreover, by making no assumptions about the content of the audio signal, AudioLM learns to generate realistic audio from audio-only corpora, be it speech or piano music, without any annotation. The ability to model diverse signals suggests that such a system could generate richer outputs if trained on the appropriate data.

Besides the inherent difficulty of synthesizing high-quality and coherent audio, another impeding factor is the scarcity of paired audio-text data. This is in stark contrast with the image domain, where the availability of massive datasets contributed significantly to the remarkable image generation quality that has recently been achieved (Ramesh et al., 2021; 2022; Saharia et al., 2022; Yu et al., 2022). Moreover, creating text descriptions of general audio is considerably harder than describing images. First, it is not straightforward to unambiguously capture with just a few words the salient characteristics of either acoustic scenes (e.g., the sounds heard in a train station or in a forest) or music (e.g., the melody, the rhythm, the timbre of vocals and the many instruments used in accompaniment). Second, audio is structured along a temporal dimension which makes sequence-wide captions a much weaker level of annotation than an image caption.

In this work, we introduce MusicLM, a model for generating high-fidelity music from text descriptions. MusicLM leverages AudioLM's multi-stage autoregressive modeling as the generative component, while extending it to incorporate text conditioning. To address the main challenge of paired data scarcity, we rely on MuLan (Huang et al., 2022), a joint music-text model that is trained to project music and its corresponding text description to representations close to each other in an embedding space. This shared embedding space eliminates the need for captions at training time alto-

[*]Equal contribution [1]Google Research [2]IRCAM - Sorbonne Université (work done while interning at Google). Correspondence to: Christian Frank <chfrank@google.com>.

https://google-research.github.io/seanet/musiclm/examples/

MusicLM: Generating Music From Text

| **paper** | **dataset** |

Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, Christian Frank
Google Research

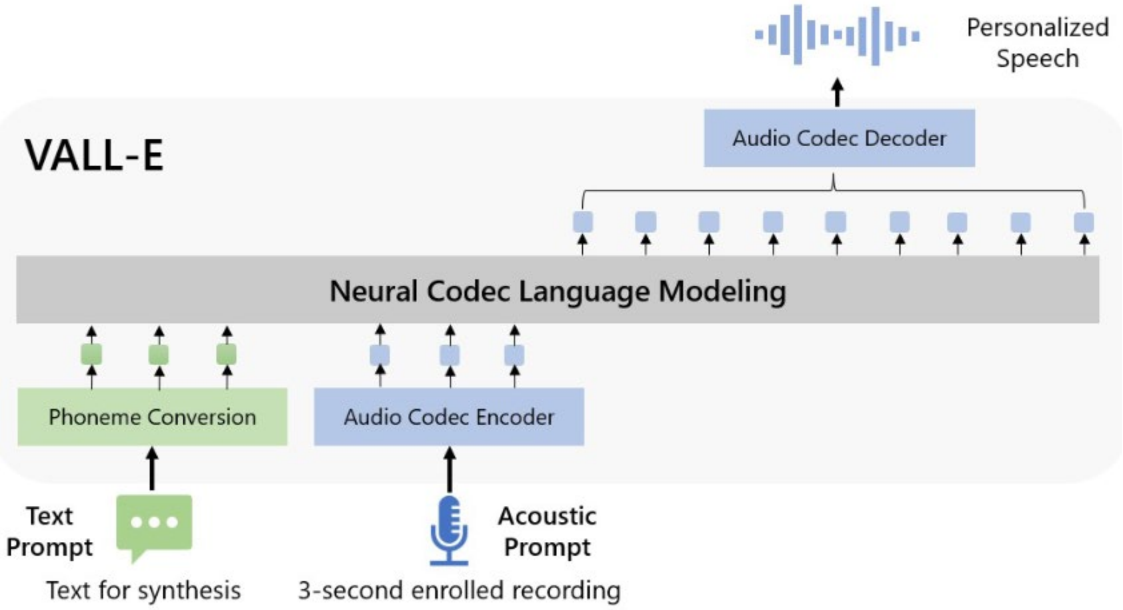
Abstract We introduce MusicLM, a model generating high-fidelity music from text descriptions such as "*a calming violin melody backed by a distorted guitar riff*". MusicLM casts the process of conditional music generation as a hierarchical sequence-to-sequence modeling task, and it generates music at 24 kHz that remains consistent over several minutes. Our experiments show that MusicLM outperforms previous systems both in audio quality and adherence to the text description. Moreover, we demonstrate that MusicLM can be conditioned on both text and a melody in that it can transform whistled and hummed melodies according to the style described in a text caption. To support future research, we publicly release MusicCaps, a dataset composed of 5.5k music-text pairs, with rich text descriptions provided by human experts.

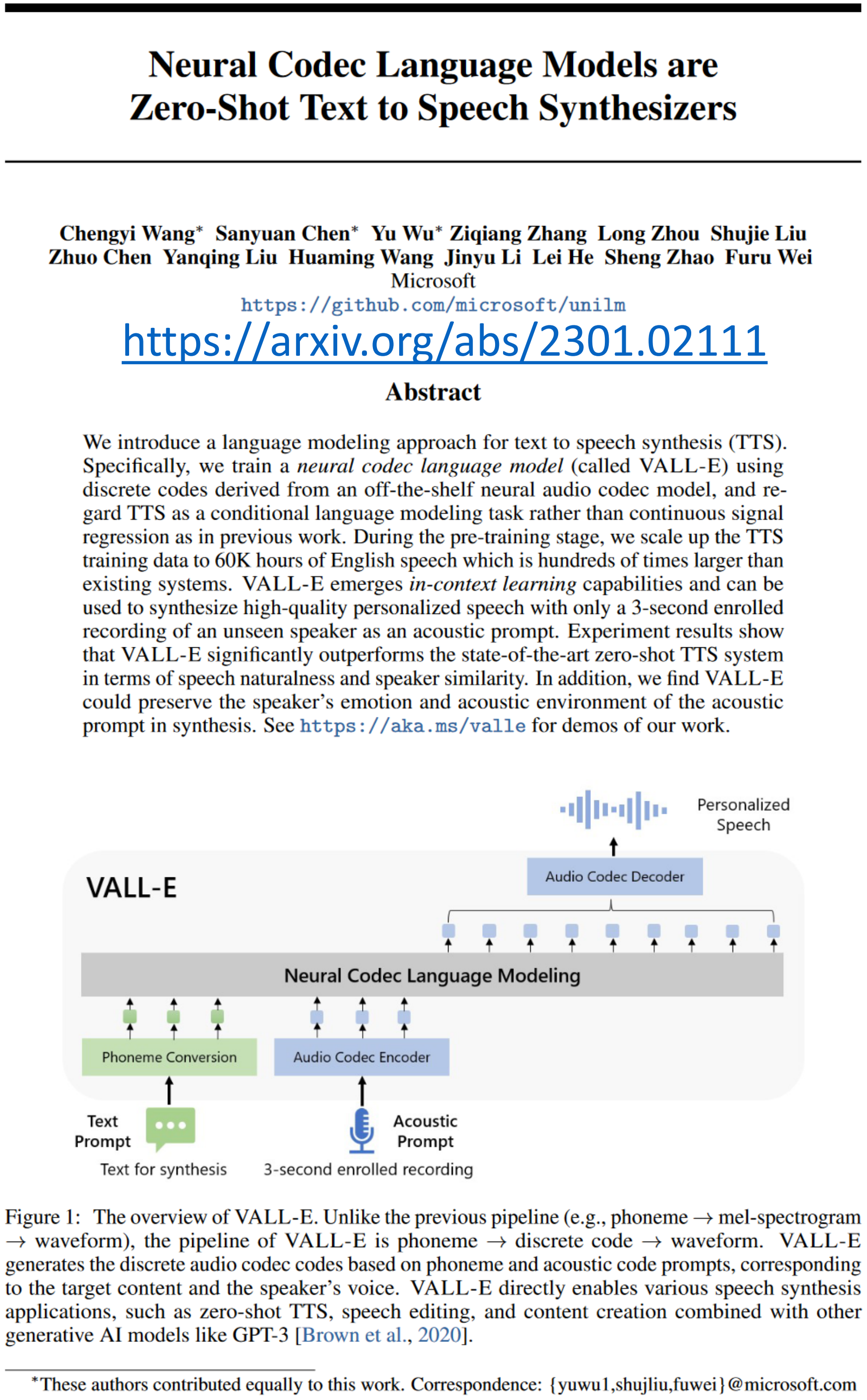
Audio Generation From Rich Captions

Caption	Generated audio
The main soundtrack of an arcade game. It is fast-paced and upbeat, with a catchy electric guitar riff. The music is repetitive and easy to remember, but with unexpected sounds, like cymbal crashes or drum rolls.	0:02 / 0:30
A fusion of reggaeton and electronic dance music, with a spacey, otherworldly sound. Induces the experience of being lost in space, and the music would be designed to evoke a sense of wonder and awe, while being danceable.	0:00 / 0:30

January 2023: VALL-E

Microsoft's take on the DALL-E 1 but for audio (speech), i.e., LLM + AE + RVQ





Neural Codec Language Models are Zero-Shot Text to Speech Synthesizers

Chengyi Wang* Sanyuan Chen* Yu Wu* Ziqiang Zhang Long Zhou Shujie Liu Zhuo Chen Yanqing Liu Huaming Wang Jinyu Li Lei He Sheng Zhao Furu Wei
Microsoft
https://github.com/microsoft/unilm

https://arxiv.org/abs/2301.02111

Abstract

We introduce a language modeling approach for text to speech synthesis (TTS). Specifically, we train a *neural codec language model* (called VALL-E) using discrete codes derived from an off-the-shelf neural audio codec model, and regard TTS as a conditional language modeling task rather than continuous signal regression as in previous work. During the pre-training stage, we scale up the TTS training data to 60K hours of English speech which is hundreds of times larger than existing systems. VALL-E emerges *in-context learning* capabilities and can be used to synthesize high-quality personalized speech with only a 3-second enrolled recording of an unseen speaker as an acoustic prompt. Experiment results show that VALL-E significantly outperforms the state-of-the-art zero-shot TTS system in terms of speech naturalness and speaker similarity. In addition, we find VALL-E could preserve the speaker's emotion and acoustic environment of the acoustic prompt in synthesis. See https://aka.ms/valle for demos of our work.

Figure 1: The overview of VALL-E. Unlike the previous pipeline (e.g., phoneme → mel-spectrogram → waveform), the pipeline of VALL-E is phoneme → discrete code → waveform. VALL-E generates the discrete audio codec codes based on phoneme and acoustic code prompts, corresponding to the target content and the speaker's voice. VALL-E directly enables various speech synthesis applications, such as zero-shot TTS, speech editing, and content creation combined with other generative AI models like GPT-3 [Brown et al., 2020].

*These authors contributed equally to this work. Correspondence: {yuwu1,shujliu,fuwei}@microsoft.com

June 2023: MusicGen

Like AudioGen, but optimized for music

Demo: https://huggingface.co/spaces/facebook/MusicGen

Simple and Controllable Music Generation

Jade Copet♠♢ Felix Kreuk♠♢ Itai Gat Tal Remez David Kant
Gabriel Synnaeve ♢ Yossi Adi*♢ Alexandre Défossez ♢
♠: equal contributions, ♢: core team
Meta AI
{jadecopet, felixkreuk, adiyoss}@meta.com

Abstract

We tackle the task of conditional music generation. We introduce MusicGen, a single Language Model (LM) that operates over several streams of compressed discrete music representation, i.e., tokens. Unlike prior work, MusicGen is comprised of a single-stage transformer LM together with efficient token interleaving patterns, which eliminates the need for cascading several models, e.g., hierarchically or upsampling. Following this approach, we demonstrate how MusicGen can generate high-quality samples, both mono and stereo, while being conditioned on textual description or melodic features, allowing better controls over the generated output. We conduct extensive empirical evaluation, considering both automatic and human studies, showing the proposed approach is superior to the evaluated baselines on a standard text-to-music benchmark. Through ablation studies, we shed light over the importance of each of the components comprising MusicGen. Music samples, code, and models are available at github.com/facebookresearch/audiocraft.

1 Introduction

Text-to-music is the task of generating musical pieces given text descriptions, e.g., "90s rock song with a guitar riff". Generating music is a challenging task as it requires modeling long range sequences. Unlike speech, music requires the use of the full frequency spectrum [Müller, 2015]. That means sampling the signal at a higher rate, i.e., the standard sampling rates of music recordings are 44.1 kHz or 48 kHz vs. 16 kHz for speech. Moreover, music contains harmonies and melodies from different instruments, which create complex structures. Human listeners are highly sensitive to disharmony [Fedorenko et al., 2012, Norman-Haignere et al., 2019], hence generating music does not leave a lot of room for making melodic errors. Lastly, the ability to control the generation process in a diverse set of methods, e.g., key, instruments, melody, genre, etc. is essential for music creators.

Recent advances in self-supervised audio representation learning [Balestriero et al., 2023], sequential modeling [Touvron et al., 2023], and audio synthesis [Tan et al., 2021] provide the conditions to develop such models. To make audio modeling more tractable, recent studies proposed representing audio signals as multiple streams of discrete tokens representing the same signal [Défossez et al., 2022]. This allows both high-quality audio generation and effective audio modeling. However, this comes at the cost of jointly modeling several parallel dependent streams.

Kharitonov et al. [2022], Kreuk et al. [2022] proposed modeling multi-streams of speech tokens in parallel following a delay approach, i.e., introduce offsets between the different streams. Agostinelli et al. [2023] proposed representing musical segments using multiple sequences of discrete tokens at different granularity and model them using a hierarchy of autoregressive models. In parallel, Donahue et al. [2023] follows a similar approach but for the task of singing to accompaniment generation. Recently, Wang et al. [2023] proposed tackling this problem in two stages: (i) modeling the first

*Yossi Adi is Affiliated with both The Hebrew University of Jerusalem & MetaAI.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

July 2023: Vampnet

Training with masked/unknown tokens enables capabilities such as inpainting and looping with variations.

https://arxiv.org/abs/2307.04686

VAMPNET: MUSIC GENERATION VIA MASKED ACOUSTIC TOKEN MODELING

Hugo Flores García[1,2] **Prem Seetharaman**[1] **Rithesh Kumar**[1] **Bryan Pardo**[2]

[1] Descript Inc.
[2] Northwestern University
hugofg@u.northwestern.edu

ABSTRACT

We introduce VampNet, a masked acoustic token modeling approach to music synthesis, compression, inpainting, and variation. We use a variable masking schedule during training which allows us to sample coherent music from the model by applying a variety of masking approaches (called prompts) during inference. VampNet is non-autoregressive, leveraging a bidirectional transformer architecture that attends to all tokens in a forward pass. With just 36 sampling passes, VampNet can generate coherent high-fidelity musical waveforms. We show that by prompting VampNet in various ways, we can apply it to tasks like music compression, inpainting, outpainting, continuation, and looping with variation (vamping). Appropriately prompted, VampNet is capable of maintaining style, genre, instrumentation, and other high-level aspects of the music. This flexible prompting capability makes VampNet a powerful music co-creation tool. Code [3] and audio samples [4] are available online.

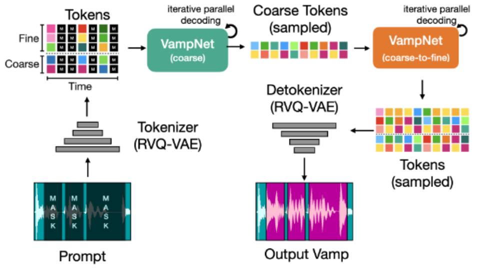


Figure 1. VampNet overview. We first convert audio into a sequence of discrete tokens using an audio tokenizer. Tokens are masked, and then passed to a masked generative model, which predicts values for masked tokens via an efficient iterative parallel decoding sampling procedure at two levels. We then decode the result back to audio.

1. INTRODUCTION

In recent years, advances in discrete acoustic token modeling have resulted in significant leaps in autoregressive generation of speech [1, 2] and music [3]. Meanwhile, approaches that use non-autoregressive parallel iterative decoding have been developed for efficient image synthesis [4, 5]. Parallel iterative decoding promises to allow faster inference than autoregressive methods and is more suited to tasks like infill, which require conditioning on both past and future sequence elements.

In this work, we combine parallel iterative decoding with acoustic token modeling, and apply them to music audio synthesis. To the best of our knowledge, ours is the first [1] extension of parallel iterative decoding to neural audio music generation. Our model, called VampNet, can be flexibly applied to a variety of applications via token-based prompting. We show that we can guide VampNet's generation with selectively masked music token sequences, asking it to fill in the blanks. The outputs of this procedure can range from a high-quality audio compression technique to variations on the original input music that match the original input music in terms of style, genre, beat and instrumentation, while varying specifics of timbre and rhythm.

Unlike auto-regressive music models [2, 3], which can only perform music continuations – using some prefix audio as a prompt, and having the model generate music that could plausibly come after it – our approach allows the prompts to be placed anywhere. We explore a variety of prompt designs, including periodic, compression, and musically informed ones (e.g. masking on the beat). We find that our model responds well to prompts to make loops and variations, thus the name VampNet [2]. We make our code open source [3] and highly encourage readers to listen to our audio samples [4].

[1] While our work was under peer review, Google released SoundStorm [6], which leverages a similar parallel iterative decoding approach to ours.

Attribution: H. Flores García, P. Seetharaman, R. Kumar, and B. Pardo, "VampNet: Music Generation via Masked Acoustic Token Modeling", in *Proc. of the 24th Int. Society for Music Information Retrieval Conf.*, Milan, Italy, 2023.

[2] To vamp is to repeat a short passage of music with variation.
[3] https://github.com/hugofloresgarcia/vampnet
[4] audio samples: https://tinyurl.com/bdfj7rdx

Singing voice conversion (SVC)

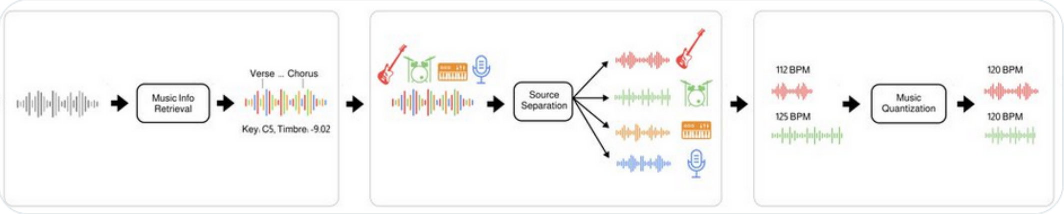
https://github.com/svc-develop-team/so-vits-svc

dadabots
@dadabots
https://twitter.com/dadabots/status/1659074568487567360

Just trained SVC singing voice conversion model on my 1hr throat bass dataset.

samim @samim

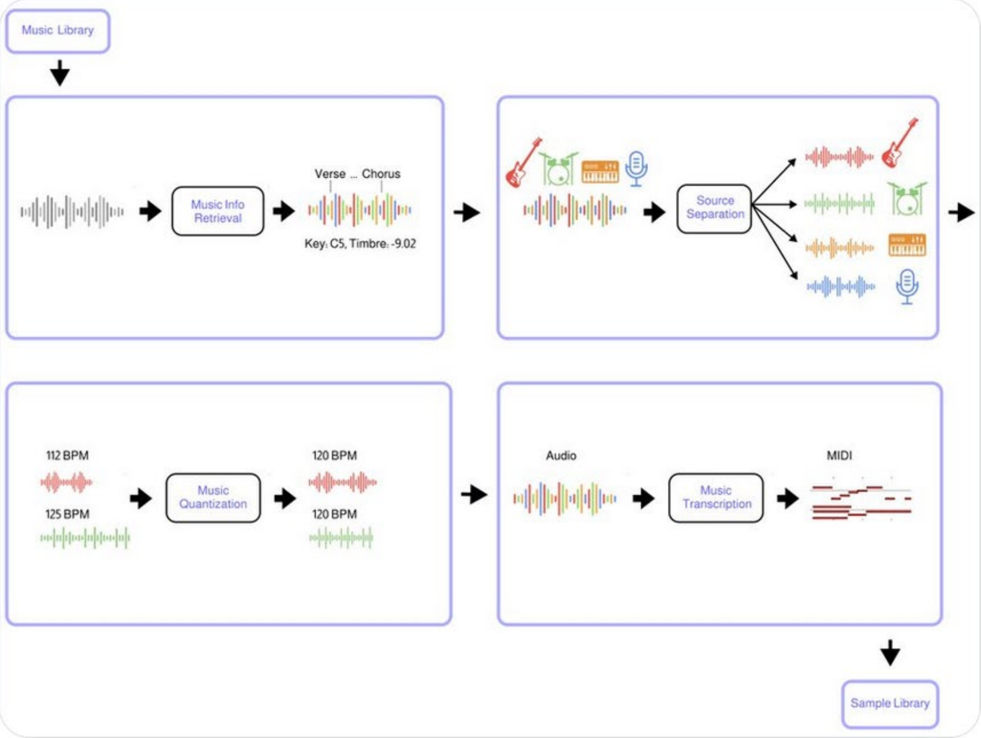
Introducing Polymath: The open-source tool that converts any music-library into a sample-library with machine learning. It separates songs into stems, quantizes to same BPM, detects key and much more. A game-changing workflow for music producers & DJs: github.com/samim23/polyma...



3:17 PM · Feb 7, 2023 · **261.2K** Views

samim @samim · Feb 21, 2023

Introducing Polymath v0.2: Now with "Auto Music Transcription" that uses machine learning to generate MIDI files from any audio file (incl of extracted stems). 2) Docker setup, making the tool easier to use 3) Many improvements by open source contributors. github.com/samim23/polyma...



https://github.com/samim23/polymath

MAGNeT (2024)

High-quality text-to-music Transformer.

Fast because it generates multiple tokens in parallel, instead of token-by-token autoregressive sampling.

Code and pretrained models:

https://github.com/facebookresearch/audiocraft/blob/main/docs/MAGNET.md

MASKED AUDIO GENERATION USING A SINGLE NON-AUTOREGRESSIVE TRANSFORMER

*Alon Ziv[1,3], Itai Gat[1], Gael Le Lan[1], Tal Remez[1], Felix Kreuk[1], Alexandre Défossez[2]
Jade Copet[1], Gabriel Synnaeve[1], Yossi Adi[1,3]

[1]FAIR Team, Meta
[2]Kyutai
[3]The Hebrew University of Jerusalem
alonzi@cs.huji.ac.il

ABSTRACT

We introduce MAGNET, a masked generative sequence modeling method that operates directly over several streams of audio tokens. Unlike prior work, MAGNET is comprised of a single-stage, non-autoregressive transformer. During training, we predict spans of masked tokens obtained from a masking scheduler, while during inference we gradually construct the output sequence using several decoding steps. To further enhance the quality of the generated audio, we introduce a novel rescoring method in which, we leverage an external pre-trained model to rescore and rank predictions from MAGNET, which will be then used for later decoding steps. Lastly, we explore a hybrid version of MAGNET, in which we fuse between autoregressive and non-autoregressive models to generate the first few seconds in an autoregressive manner while the rest of the sequence is being decoded in parallel. We demonstrate the efficiency of MAGNET for the task of text-to-music and text-to-audio generation and conduct an extensive empirical evaluation, considering both objective metrics and human studies. The proposed approach is comparable to the evaluated baselines, while being significantly faster (x7 faster than the autoregressive baseline). Through ablation studies and analysis, we shed light on the importance of each of the components comprising MAGNET, together with pointing to the trade-offs between autoregressive and non-autoregressive modeling, considering latency, throughput, and generation quality. Samples are available on our demo page https://pages.cs.huji.ac.il/adiyoss-lab/MAGNeT

1 INTRODUCTION

https://arxiv.org/pdf/2401.04577.pdf

Recent developments in self-supervised representation learning (Hsu et al., 2021; Défossez et al., 2022), sequence modeling (Touvron et al., 2023; Rozière et al., 2023), and audio synthesis (Lee et al., 2022; Polyak et al., 2021) allow a great leap in performance when considering high quality conditional audio generation. The prominent approach, in recent years, is to represent the audio signals as a compressed representation, either discrete or continuous, and apply a generative model on top of it (Lakhotia et al., 2021; Kharitonov et al., 2022; Borsos et al., 2023a; Kreuk et al., 2022a; Copet et al., 2023; Lam et al., 2023; Agostinelli et al., 2023; Gat et al., 2023; Sheffer & Adi, 2023; Maimon & Adi, 2022; Schneider et al., 2023; Huang et al., 2023b; Liu et al., 2023a; Li et al., 2023; Liu et al., 2023b). Recently, Défossez et al. (2022); Zeghidour et al. (2021) proposed to apply a VQ-VAE directly on the raw waveform using residual vector quantization to obtain a multi-stream discrete representation of the audio signal. Later on, Kreuk et al. (2022a); Wang et al. (2023); Zhang et al. (2023); Copet et al. (2023); Kreuk et al. (2022b) presented a conditional language modeling on such audio signals representations. In parallel, Schneider et al. (2023); Huang et al. (2023b); Liu et al. (2023a) proposed training a conditional diffusion-based generative model operating on learned continuous representations of the audio signal obtained from a pre-trained auto-encoder model.

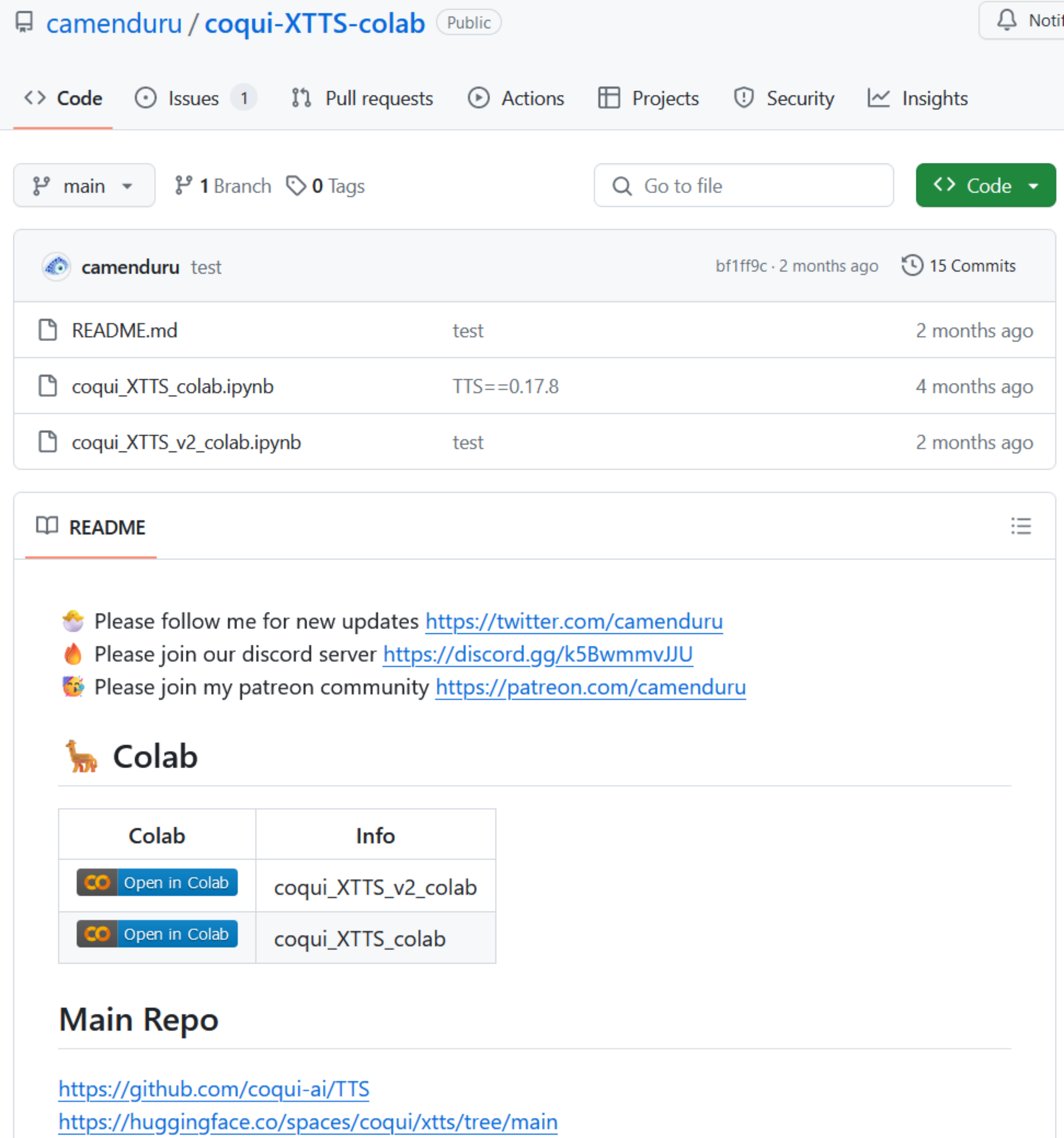
*Work was done as part of Alon's internship at FAIR.

High-quality text-to-speech in Colab

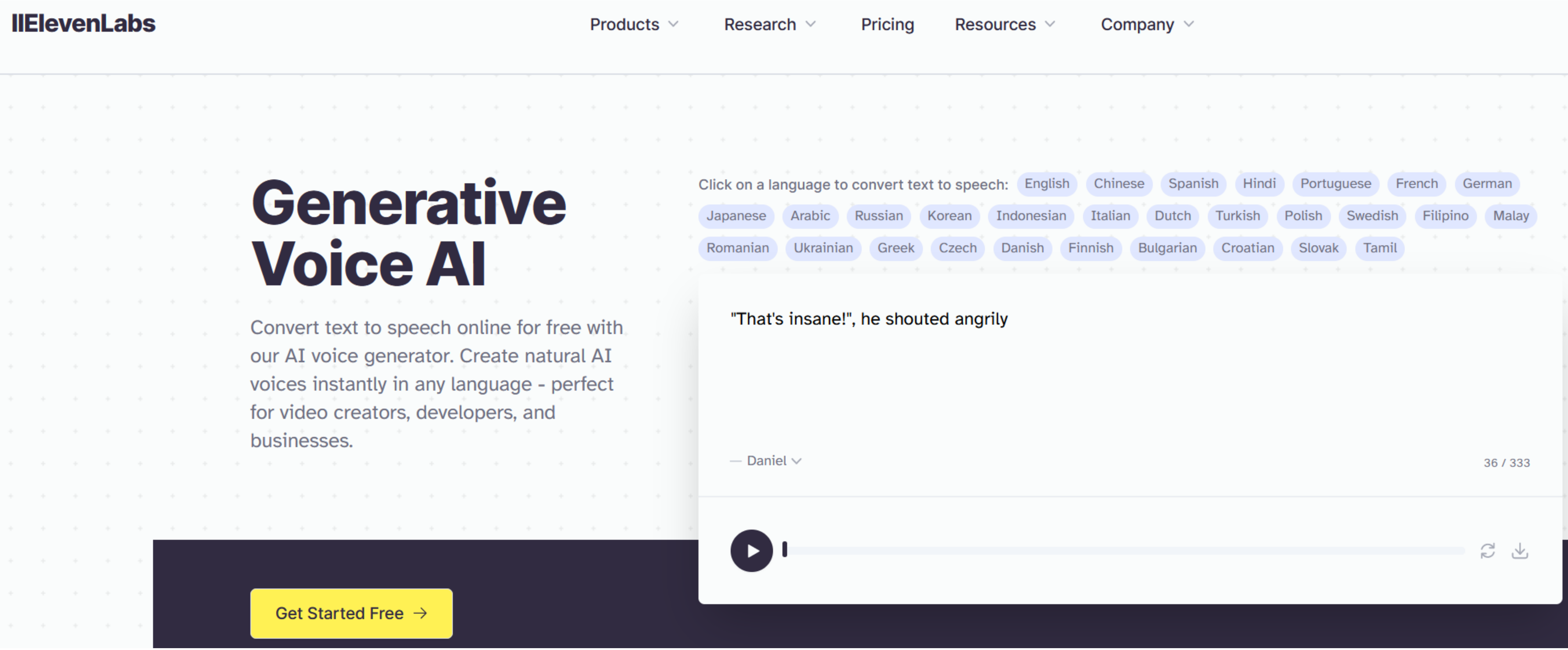
Supports voice cloning

https://github.com/camenduru/coqui-XTTS-colab

For a recent discussion on alternatives, see: https://www.reddit.com/r/MachineLearning/comments/195cxim/d_what_is_the_best_texttospeech_tool_currently/



Commercial services



https://elevenlabs.io/

Another text-to-speech service with high-quality emotional voices.

Their advertising is totally tone-deaf, though…

Stable Audio

Try it out for free

Log in

Create music with AI.

Trance, Ibiza, Beach, Sun, 4 AM, Progressive, Synthesizer, 909, Dramatic Chords, Choir, Euphoric, Nostalgic, Dynamic, Flowing

https://www.stableaudio.com/

Trance, Ibiza, Beach, Sun, 4 AM, Progressive,...

Stable Audio

AI music creation

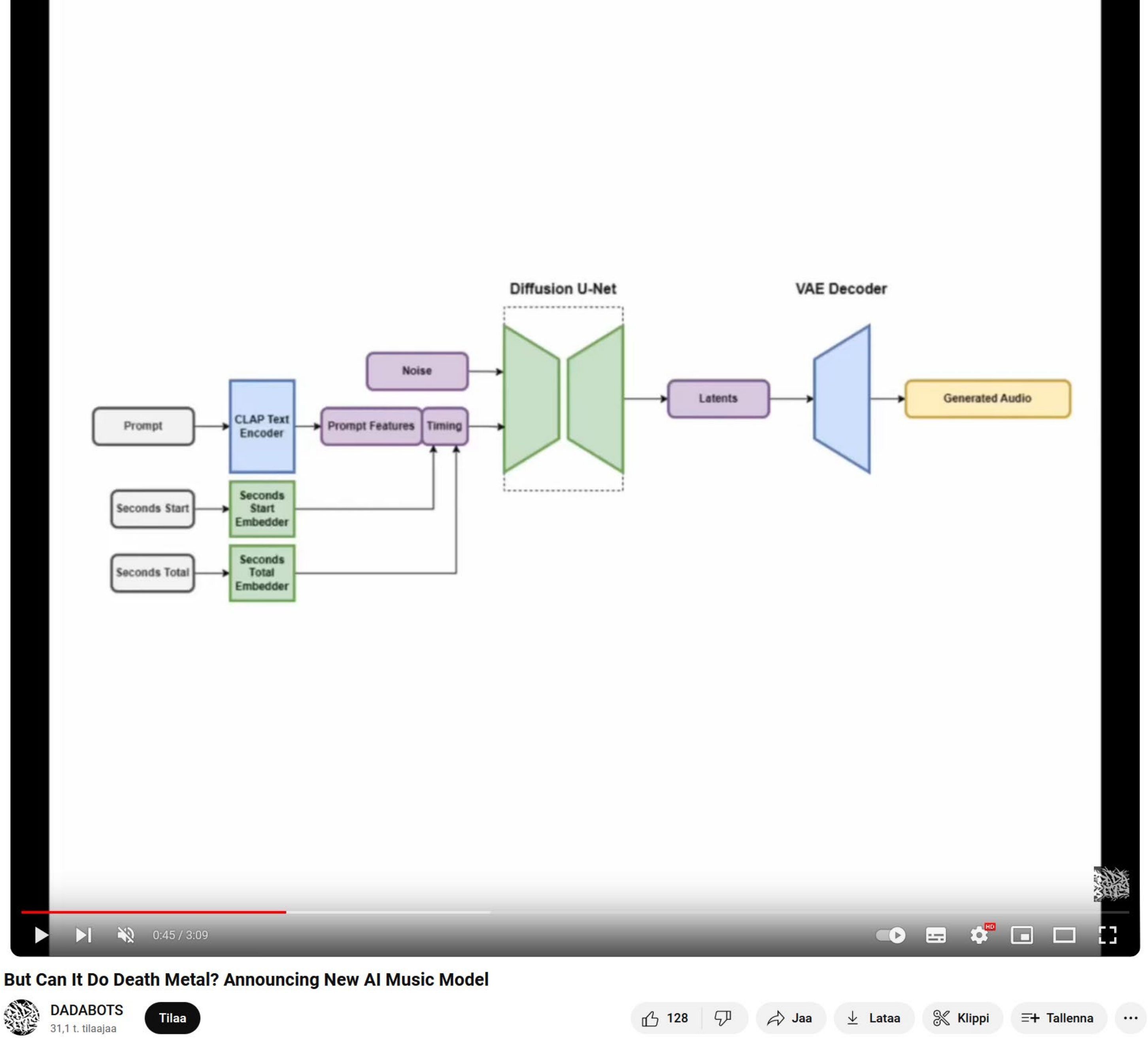
Generate high-quality audio that you can use commercially.
Get started for free.

Try it out

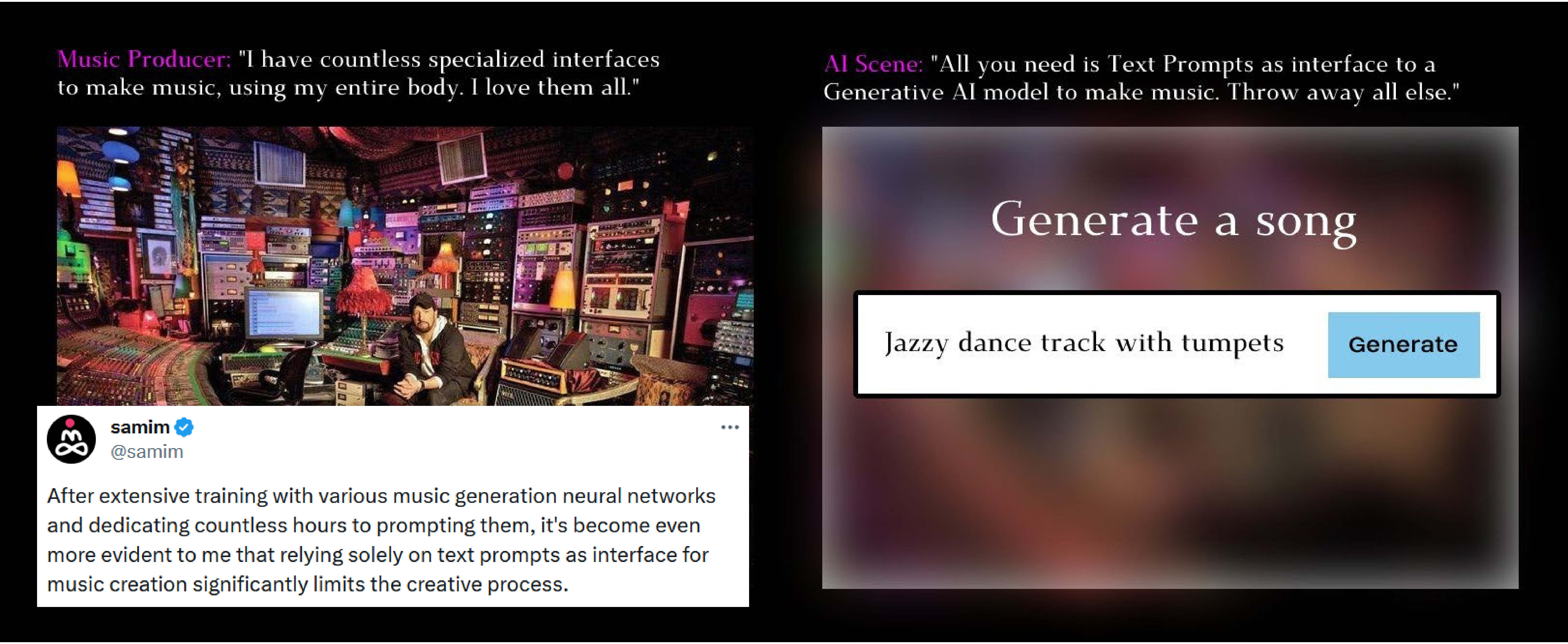
0:00

1:35

https://www.youtube.com/watch?v=6MVysjDl_0E



Valid critique



https://twitter.com/samim/status/1651439941627310080

How to control?

Musicfy focuses on voice-based control: https://musicfy.lol/

dadabots @dadabots · Nov 21, 2023

I've learned many instruments, but voice is the most immediately expressive. In this sick video, Ummet Ozcan uses Musicfy's couple dozen (RVC?) instrument models.

Lately I've been beatboxing into our prompt models that generate full bands/subgenres. Can't wait to share more!!

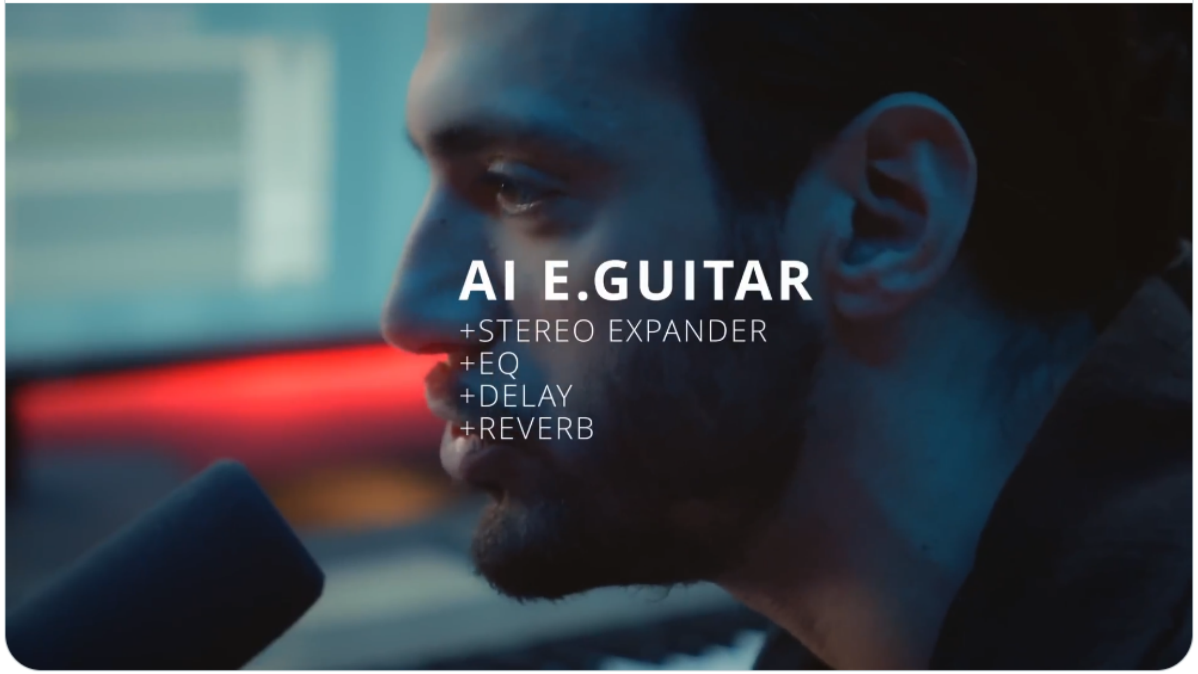
Arib 🇺🇸 🇵🇰 @aribk24 · Nov 20, 2023

voice to music

we just launched a feature that allows you to sing and turn your notes into any instrument you want

...

Show more



Emvoice

Melody and text control

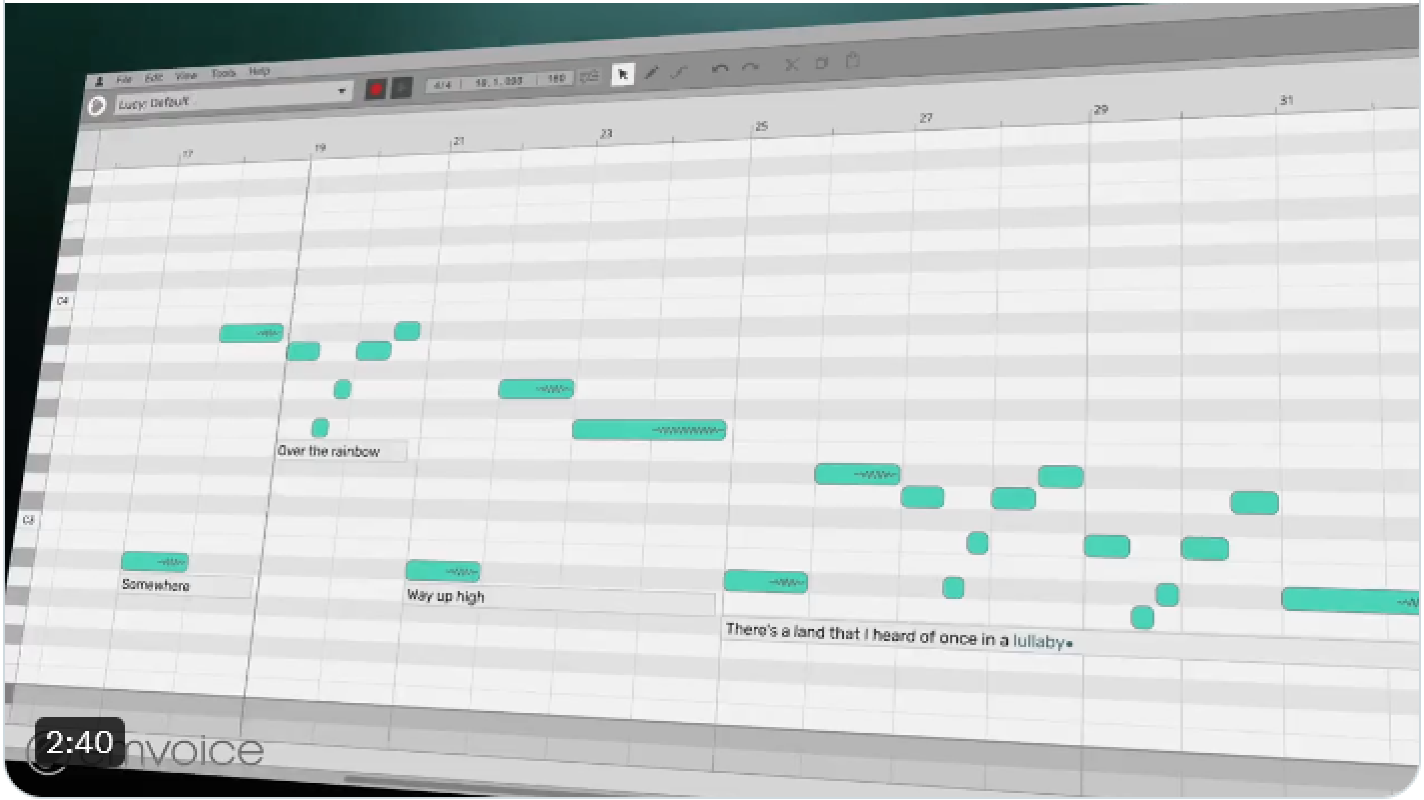
https://emvoiceapp.com/

AaltoMediaAI
@aaltomediaai

Emvoice lets you input both lyrics and melody and choose between 3 AI singing voices. Might be good for initial exploration of ideas if not real production.

Emvoice @emvoiceapp · Sep 15, 2023

The legendary @MaartenVorwerk used Lucy as the lead vocalist in Jeckyll & Hyde's new single, "Over The Rainbow." Take a listen!



Generating 3D animation

Use case: Interactive character control

- Train a model (LLM, Diffusion, Motion Matching) on some existing mocap dataset
- Use the model to sample the next pose given the current movement state and a goal(s)

Use case: Sparse keyframing

- Given just a few poses, determine a full movement sequence
- Analogous to *inpainting* in image synthesis

Use case: Generate new data

- Useful for both traditional game animation systems and training interactive character control and
- Text-to-motion: Can generate variations of training data, but offers no control of details
- AI-based motion capture from video: Good control of details, limited only by motion capture actors

Common tools

Neural nets for labeling of body parts from video.

Landmark paper: OpenPose (Cao et al., CVPR 2017)

For a lightweight version that runs in real-time on mobile devices, see:

https://arxiv.org/abs/2006.10204

Off-the-shelf solution:
https://developers.google.com/mediapipe/solutions/vision/pose_landmarker

Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields *

Zhe Cao Tomas Simon Shih-En Wei Yaser Sheikh
The Robotics Institute, Carnegie Mellon University
{zhecao,shihenw}@cmu.edu {tsimon,yaser}@cs.cmu.edu

https://arxiv.org/abs/1611.08050

Abstract

We present an approach to efficiently detect the 2D pose of multiple people in an image. The approach uses a nonparametric representation, which we refer to as Part Affinity Fields (PAFs), to learn to associate body parts with individuals in the image. The architecture encodes global context, allowing a greedy bottom-up parsing step that maintains high accuracy while achieving realtime performance, irrespective of the number of people in the image. The architecture is designed to jointly learn part locations and their association via two branches of the same sequential prediction process. Our method placed first in the inaugural COCO 2016 keypoints challenge, and significantly exceeds the previous state-of-the-art result on the MPII Multi-Person benchmark, both in performance and efficiency.

Figure 1. **Top:** Multi-person pose estimation. Body parts belonging to the same person are linked. **Bottom left:** Part Affinity Fields (PAFs) corresponding to the limb connecting right elbow and right wrist. The color encodes orientation. **Bottom right:** A zoomed in view of the predicted PAFs. At each pixel in the field, a 2D vector encodes the position and orientation of the limbs.

1. Introduction

Human 2D pose estimation—the problem of localizing anatomical keypoints or "parts"—has largely focused on finding body parts of *individuals* [8, 4, 3, 21, 33, 13, 25, 31, 6, 24]. Inferring the pose of multiple people in images, especially socially engaged individuals, presents a unique set of challenges. First, each image may contain an unknown number of people that can occur at any position or scale. Second, interactions between people induce complex spatial interference, due to contact, occlusion, and limb articulations, making association of parts difficult. Third, runtime complexity tends to grow with the number of people in the image, making realtime performance a challenge.

A common approach [23, 9, 27, 12, 19] is to employ a person detector and perform single-person pose estimation for each detection. These top-down approaches directly leverage existing techniques for single-person pose estimation [17, 31, 18, 28, 29, 7, 30, 5, 6, 20], but suffer from early commitment: if the person detector fails–as it is prone to do when people are in close proximity–there is no recourse to recovery. Furthermore, the runtime of these top-down approaches is proportional to the number of people: for each detection, a single-person pose estimator is run, and the more people there are, the greater the computational cost. In contrast, bottom-up approaches are attractive as they offer robustness to early commitment and have the potential to decouple runtime complexity from the number of people in the image. Yet, bottom-up approaches do not directly use global contextual cues from other body parts and other people. In practice, previous bottom-up methods [22, 11] do not retain the gains in efficiency as the final parse requires costly global inference. For example, the seminal work of Pishchulin et al. [22] proposed a bottom-up approach that jointly labeled part detection candidates and associated them to individual people. However, solving the integer linear programming problem over a fully connected graph is an NP-hard problem and the average processing time is on the order of hours. Insafutdinov et al. [11] built on [22] with stronger part detectors based on ResNet [10] and image-dependent pairwise scores, and vastly improved the runtime, but the method still takes several minutes per image, with a limit on the number of part proposals. The pairwise representations used in [11], are difficult to regress precisely and thus a separate logistic regression is required.

*Video result: https://youtu.be/pW6nZXeW1GM

Common tools

Deformable body models that can be fitted to images.

SMPL-X (2019) is currently a common one

Figure 4: Qualitative results of SMPL-X for the in-the-wild images of the LSP dataset [33]. A strong holistic model like SMPL-X results in *natural* and *expressive* reconstruction of bodies, hands and faces. Gray color depicts the gender-specific model for confident gender detections. Blue is the gender-neutral model that is used when the gender classifier is uncertain.

Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

Georgios Pavlakos[*1,2], Vasileios Choutas[*1], Nima Ghorbani[1], Timo Bolkart[1], Ahmed A. A. Osman[1], Dimitrios Tzionas[1], and Michael J. Black[1]

[1]MPI for Intelligent Systems, Tübingen, DE , [2] University of Pennsylvania, PA, USA
{gpavlakos, vchoutas, nghorbani, tbolkart, aosman, dtzionas, black}@tuebingen.mpg.de

https://smpl-x.is.tue.mpg.de/

Abstract

To facilitate the analysis of human actions, interactions and emotions, we compute a 3D model of human body pose, hand pose, and facial expression from a single monocular image. To achieve this, we use thousands of 3D scans to train a new, unified, 3D model of the human body, SMPL-X, that extends SMPL with fully articulated hands and an expressive face. Learning to regress the parameters of SMPL-X directly from images is challenging without paired images and 3D ground truth. Consequently, we follow the approach of SMPLify, which estimates 2D features and then optimizes model parameters to fit the features. We improve on SMPLify in several significant ways: (1) we detect 2D features corresponding to the face, hands, and feet and fit the full SMPL-X model to these; (2) we train a new neural network pose prior using a large MoCap dataset; (3) we define a new interpenetration penalty that is both fast and accurate; (4) we automatically detect gender and the appropriate body models (male, female, or neutral); (5) our PyTorch implementation achieves a speedup of more than 8× over Chumpy. We use the new method, SMPLify-X, to fit SMPL-X to both controlled images and images in the wild. We evaluate 3D accuracy on a new curated dataset comprising 100 images with pseudo ground-truth. This is a step towards automatic expressive human capture from monocular RGB data. The models, code, and data are available for research purposes at `https://smpl-x.is.tue.mpg.de`.

1. Introduction

Humans are often a central element in images and videos. Understanding their posture, the social cues they communicate, and their interactions with the world is critical for holistic scene understanding. Recent methods have shown rapid progress on estimating the major body joints, hand joints and facial features in 2D [15, 31, 70]. Our interactions with the world, however, are fundamentally 3D and recent work has also made progress on the 3D estimation

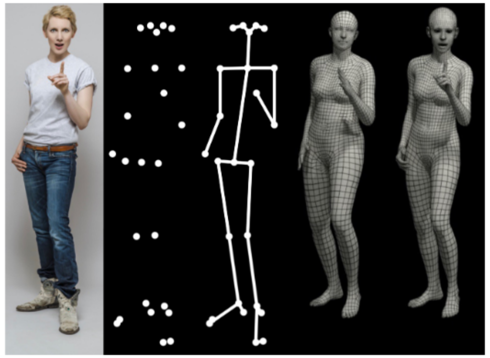


Figure 1: Communication and gesture rely on the *body* pose, *hand* pose, and *facial* expression, all *together*. The major joints of the body are not sufficient to represent this and current 3D models are not expressive enough. In contrast to prior work, our approach estimates a more detailed and expressive 3D model from a single image. From left to right: RGB image, major joints, skeleton, SMPL (female), SMPL-X (female). The hands and face in SMPL-X enable more *holistic* and *expressive* body capture.

of the major joints and rough 3D pose directly from single images [10, 37, 59, 62].

To understand human behavior, however, we have to capture more than the major joints of the body – we need the full 3D surface of the body, hands and the face. There is no system that can do this today due to several major challenges including the lack of appropriate 3D models and rich 3D training data. Figure 1 illustrates the problem. The interpretation of expressive and communicative images is difficult using only sparse 2D information or 3D representations that lack hand and face detail. To address this problem, we need two things. First, we need a 3D model of the body that is able to represent the complexity of human faces, hands, and body pose. Second, we need a method to extract such a model from a single image.

Advances in neural networks and large datasets of manually labeled images have resulted in rapid progress in 2D human "pose" estimation. By "pose", the field often means

[*] equal contribution

Common tools

Differentiable renderers

Enable fast gradient-based fitting of rendered shapes to images

E.g., https://github.com/NVlabs/nvdiffrast

Learning to Regress Bodies from Images using Differentiable Semantic Rendering

Sai Kumar Dwivedi[1] Nikos Athanasiou[1] Muhammed Kocabas[1,2] Michael J. Black[1]
[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany [2]ETH Zurich
{sdwivedi, nathanasiou, mkocabas, black}@tue.mpg.de

https://arxiv.org/abs/2110.03480

Figure 1: **Differentiable Semantic Rendering (DSR)**. A state-of-the-art approach [14] (purple) fails to estimate accurate 3D pose and shape for in-the-wild scenarios. We address this by exploiting the clothing semantics of the human body. Our approach, DSR, (blue) captures more accurate 3D pose and shape compared to previous work.

Abstract

Learning to regress 3D human body shape and pose (e.g. SMPL parameters) from monocular images typically exploits losses on 2D keypoints, silhouettes, and/or part-segmentation when 3D training data is not available. Such losses, however, are limited because 2D keypoints do not supervise body shape and segmentations of people in clothing do not match projected minimally-clothed SMPL shapes. To exploit richer image information about clothed people, we introduce higher-level semantic information about clothing to penalize clothed and non-clothed regions of the human body differently. To do so, we train a body regressor using a novel "Differentiable Semantic Rendering (DSR)" loss. For Minimally-Clothed (MC) regions, we define the DSR-MC loss, which encourages a tight match between a rendered SMPL body and the minimally-clothed regions of the image. For clothed regions, we define the DSR-C loss to encourage the rendered SMPL body to be inside the clothing mask. To ensure end-to-end differentiable training, we learn a semantic clothing prior for SMPL vertices from thousands of clothed human scans. We perform extensive qualitative and quantitative experiments to evaluate the role of clothing semantics on the accuracy of 3D human pose and shape estimation. We outperform all previous state-of-the-art methods on 3DPW and Human3.6M and obtain on par results on MPI-INF-3DHP. Code and trained models are available for research at https://dsr.is.tue.mpg.de/.

1. Introduction

Estimating 3D human pose and shape from in-the-wild images has received great research interest [5, 14, 15, 18, 20, 30, 34, 54] because of its varied applications in animation, games, and the fashion industry. One aspect that makes this problem challenging is the difficulty of obtaining accurate 3D ground-truth annotations, as they require either specialized –mostly indoors– MoCap systems or careful calibration and setup of IMU sensors [46]. Such data would facilitate training robust regressors paving the way for estimating human-scene interaction with greater granularity.

Given the lack of in-the-wild 3D ground-truth, the vast majority of previous methods focus on 2D keypoints [5, 14] with some learned 3D priors. Even though sparse 2D keypoints give useful constrained, relying only on these leads to unrealistic poses because of depth ambiguities and occlu-

Common tools

Differentiable physics simulation

Can allow more efficient optimizing of physics simulation results to match images or other measurements

https://openaccess.thecvf.com/content/CVPR2022/papers/Gartner_Differentiable_Dynamics_for_Articulated_3D_Human_Motion_Reconstruction_CVPR_2022_paper.pdf

Differentiable Dynamics for Articulated 3d Human Motion Reconstruction

Erik Gärtner[1,2] Mykhaylo Andriluka[1] Erwin Coumans[1] Cristian Sminchisescu[1]

[1]Google Research, [2]Lund University

erik.gartner@math.lth.se

{mykhayloa,erwincoumans,sminchisescu}@google.com

Abstract

We introduce DiffPhy, a differentiable physics-based model for articulated 3d human motion reconstruction from video. Applications of physics-based reasoning in human motion analysis have so far been limited, both by the complexity of constructing adequate physical models of articulated human motion, and by the formidable challenges of performing stable and efficient inference with physics in the loop. We jointly address such modeling and inference challenges by proposing an approach that combines a physically plausible body representation with anatomical joint limits, a differentiable physics simulator, and optimization techniques that ensure good performance and robustness to suboptimal local optima. In contrast to several recent methods [39, 42, 55], our approach readily supports full-body contact including interactions with objects in the scene. Most importantly, our model connects end-to-end with images, thus supporting direct gradient-based physics optimization by means of image-based loss functions. We validate the model by demonstrating that it can accurately reconstruct physically plausible 3d human motion from monocular video, both on public benchmarks with available 3d ground-truth, and on videos from the internet.

1. Introduction

We seek to contribute to the development of physics-based methodology as one of the building blocks in constructing accurate and robust 3d visual human sensing systems. Incorporating the laws of physics into the visual reasoning process is appealing as it promotes the plausibility of estimated motion and facilitates more efficient use of training examples [9]. We focus on articulated human motion as an epitome of a real-world prediction task that is both well studied and challenging. Existing state-of-the-art approaches demonstrate relatively high accuracy in terms of joint position estimation metrics [23, 24, 54, 62]. However, predictions can sometimes be physically implausible, even for simple motions such as walking and running. For instance, estimates can include unreasonably abrupt transitions in world space, or artifacts such as foot skating or non-equilibrium states [39,42]. Many methods are typically trained on large motion capture datasets and encounter difficulties when tested on motions not well represented in those training sets. Arguably, imposing some form of physics-based generally valid prior on the articulated motion estimates should greatly improve the plausibility of results.

However, physics-based reasoning comes at the cost of substantial modeling and inference complexity. Typically, physics-based articulated estimation methods rely on rigid body dynamics (RBD) [10, 44], a formulation that introduces many auxiliary variables corresponding to forces acting at the body joints at each time step. Moreover, physical contact results in non-smooth effects where small changes to model parameters might result in substantially different motions. Therefore inferring physics variables given the inherent uncertainty in monocular video, and under contact discontinuities, becomes significantly difficult, algorithmically and computationally. Despite such challenges, a number of recent methods successfully apply physics-based constraints for articulated human motion estimation [2, 39, 42, 59]. One possibility to cope with modeling complexity, explored in recent work, is to simplify the physics and model contacts only between the body and the feet [39, 42, 55]. Others use auxiliary external forces applied at the body to compensate for modeling error [42, 59].

In this paper, we aim to broaden the methodology for physics-based articulated human motion estimation. Specifically, we demonstrate that we can successfully leverage recent progress in differentiable simulation [17, 19, 52] in order to incorporate physics-based constraints into the articulated 3d human motion reconstruction. Our approach, *DiffPhy*, relies on gradient-based optimization, connects end-to-end with images, and does not require simplifying assumptions on contacts or the introduction of external non-physical residual forces.

Common tools

Large-scale motion capture datasets.

The AMASS meta-dataset is currently the largest and best one. https://openaccess.thecvf.com/content_ICCV_2019/html/Mahmood_AMASS_Archive_of_Motion_Capture_As_Surface_Shapes_ICCV_2019_paper.html

Ubisoft LaForge LAFAN1 is a high-quality dataset for game animation research https://github.com/ubisoft/ubisoft-laforge-animation-dataset

AMASS: Archive of Motion Capture as Surface Shapes

Naureen Mahmood[1] Nima Ghorbani[2] Nikolaus F. Troje[3]
Gerard Pons-Moll[4] Michael J. Black[2]
[1]Meshcapade [2]MPI for Intelligent Systems [3]York University [4]MPI for Informatics
nmahmood@meshcapade.com, {nghorbani, black}@tue.mpg.de
troje@yorku.ca, gpons@mpi-inf.mpg.de

Abstract

Large datasets are the cornerstone of recent advances in computer vision using deep learning. In contrast, existing human motion capture (mocap) datasets are small and the motions limited, hampering progress on learning models of human motion. While there are many different datasets available, they each use a different parameterization of the body, making it difficult to integrate them into a single meta dataset. To address this, we introduce AMASS, a large and varied database of human motion that unifies 15 different optical marker-based mocap datasets by representing them within a common framework and parameterization. We achieve this using a new method, MoSh++, that converts mocap data into realistic 3D human meshes represented by a rigged body model. Here we use SMPL [26], which is widely used and provides a standard skeletal representation as well as a fully rigged surface mesh. The method works for arbitrary markersets, while recovering soft-tissue dynamics and realistic hand motion. We evaluate MoSh++ and tune its hyperparameters using a new dataset of 4D body scans that are jointly recorded with marker-based mocap. The consistent representation of AMASS makes it readily useful for animation, visualization, and generating training data for deep learning. Our dataset is significantly richer than previous human motion collections, having more than 40 hours of motion data, spanning over 300 subjects, more than 11000 motions, and is available for research at https://amass.is.tue.mpg.de/.

Figure 1: We unify a large corpus of archival marker-based optical human mocap datasets by representing them within a common framework and parameterization. A sampling of shapes and poses from a few datasets in AMASS is shown, from left to right: CMU [9], MPI-HDM05 [30, 31], MPI-Pose Limits [3], KIT [27], BMLrub [42], TCD [21] and ACCAD [34] datasets. The input is sparse markers and the output is SMPL body models.

1. Introduction

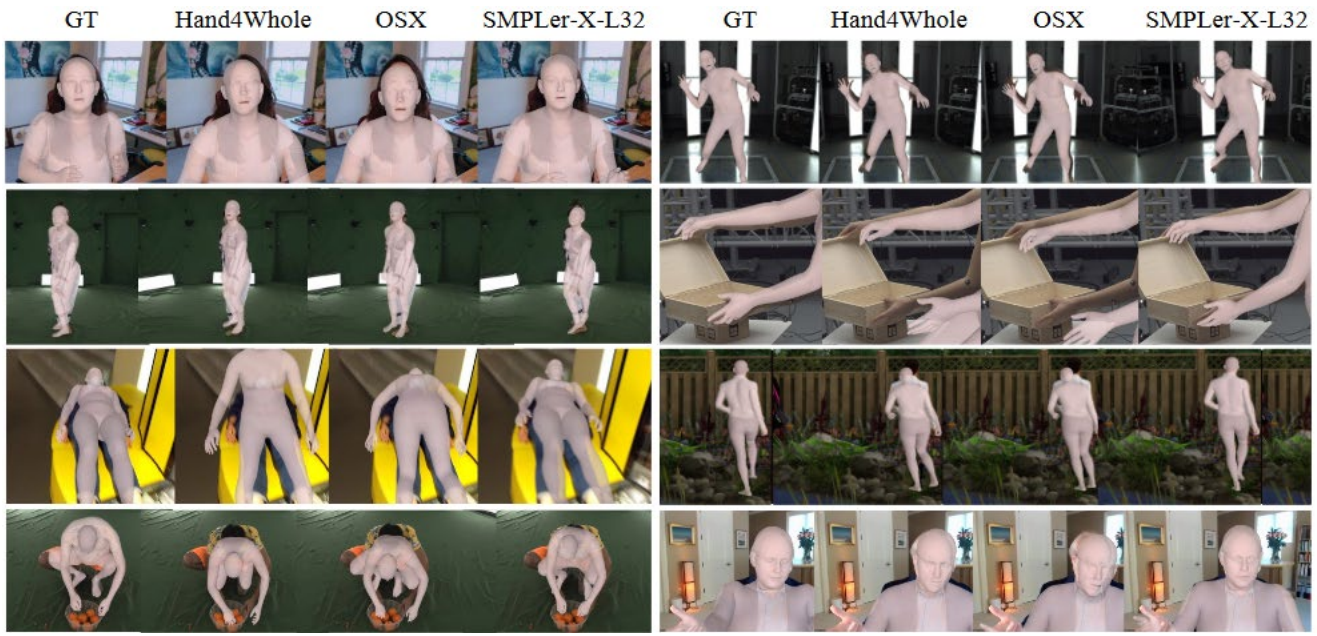
This paper addresses two interrelated goals. First, we develop a method to accurately recover the shape and pose of a person in motion from standard motion capture (mocap) marker data. This enables the second goal, which is to create the largest publicly available database of human motions that can enable machine learning for applications in animation and computer vision. While there have been attempts in both these directions, existing mocap databases are insufficient in terms of size and complexity to exploit the full power of existing deep learning tools. There are many different mocap datasets available, but pulling them together into a coherent formulation is challenging due to the use of widely varying markersets and laboratory-specific procedures [16]. We achieve this by extending MoSh [25] in several important ways, enabling us to collect a large and varied dataset of human motions in a consistent format (Fig. 1).

MoSh employs a generative model of the body, learned from a large number of 3D body scans, to compute the full 3D body shape and pose from a sparse set of motion capture markers. The results are realistic, but the method has several important limitations, which make it inappropriate for our task. First, MoSh relies on a formulation of the *SCAPE body model* [8], which is not compatible with

Common tools

Neural networks that directly map an image to body pose parameters

Now superseding or augmenting earlier iterative optimization approaches (much faster)



SMPLer-X: Scaling Up Expressive Human Pose and Shape Estimation

Zhongang Cai*[1,2,3], Wanqi Yin*[2,4], Ailing Zeng[5], Chen Wei[2], Qingping Sun[2], Yanjun Wang[2], Hui En Pang[1,2], Haiyi Mei[2], Mingyuan Zhang[1], Lei Zhang[5], Chen Change Loy[1], Lei Yang†[2,3], Ziwei Liu†[1]

[1] S-Lab, Nanyang Technological University, [2] SenseTime Research, [3] Shanghai AI Laboratory, [4] The University of Tokyo, [5] International Digital Economy Academy (IDEA)

Abstract

Expressive human pose and shape estimation (EHPS) unifies body, hands, and face motion capture with numerous applications. Despite encouraging progress, current state-of-the-art methods still depend largely on a confined set of training datasets. In this work, we investigate scaling up EHPS towards the first *generalist* foundation model (dubbed **SMPLer-X**), with up to ViT-Huge as the backbone and training with up to 4.5M instances from diverse data sources. With big data and the large model, SMPLer-X exhibits strong performance across diverse test benchmarks and excellent transferability to even unseen environments. *1) For the data scaling*, we perform a systematic investigation on 32 EHPS datasets, including a wide range of scenarios that a model trained on any single dataset cannot handle. More importantly, capitalizing on insights obtained from the extensive benchmarking process, we optimize our training scheme and select datasets that lead to a significant leap in EHPS capabilities. *2) For the model scaling,* we take advantage of vision transformers to study the scaling law of model sizes in EHPS. Moreover, our finetuning strategy turn SMPLer-X into *specialist* models, allowing them to achieve further performance boosts. Notably, our foundation model SMPLer-X consistently delivers state-of-the-art results on seven benchmarks such as AGORA (107.2 *mm* NMVE), UBody (57.4 *mm* PVE), EgoBody (63.6 *mm* PVE), and EHF (62.3 *mm* PVE without finetuning). [2]

1 Introduction

https://arxiv.org/abs/2309.17448

The recent progress in expressive human pose and shape estimation (EHPS) from monocular images or videos offers transformative applications for the animation, gaming, and fashion industries. This task typically employs parametric human models (*e.g.*, SMPL-X [51]) to adeptly represent the highly complicated human body, face, and hands. In recent years, a large number of diverse datasets have entered the field [5, 8, 8, 63, 68, 39, 3, 14, 16, 16, 64, 9], providing the community new opportunities to study various aspects such as capture environment, pose distribution, body visibility, and camera views. Yet, the state-of-the-art methods remain tethered to a limited selection of these datasets, creating a bottleneck in performance across varied scenarios and hindering the ability to generalize to unseen situations.

Our mission in this study is to explore existing data resources comprehensively, providing key insights crucial for establishing robust, universally applicable models for EHPS. Accordingly, we establish the first systematic benchmark for EHPS, utilizing 32 datasets and evaluating their performance

*Equal contributions. †Co-corresponding authors.
[2]Homepage: `https://caizhongang.github.io/projects/SMPLer-X/`.

37th Conference on Neural Information Processing Systems (NeurIPS 2023) Track on Datasets and Benchmarks.

Common tools

RGB-D sensors (depth cameras) in addition to regular cameras

E.g., Azure Kinect https://azure.microsoft.com/en-us/products/kinect-dk

HuMoR: 3D Human Motion Model for Robust Pose Estimation

Davis Rempe[1] Tolga Birdal[1] Aaron Hertzmann[2] Jimei Yang[2]
Srinath Sridhar[3] Leonidas J. Guibas[1]
[1]Stanford University [2]Adobe Research [3]Brown University

Figure 1: *Overview.* HuMoR is a 3D **Hu**man **Mo**tion model for **R**obust estimation of temporal pose formulated as a conditional variational autoencoder. (Left) The proposed approach can operate on many input modalities and is designed to handle partial and noisy observations. (Middle/Right) A test-time optimization fits 3D motion and shape to an input sequence using HuMoR as a prior; additional outputs include the ground and person-ground contacts (colored as ground plane and contacts).

Abstract

*We introduce HuMoR: a 3D **Hu**man **Mo**tion Model for **R**obust Estimation of temporal pose and shape. Though substantial progress has been made in estimating 3D human motion and shape from dynamic observations, recovering plausible pose sequences in the presence of noise and occlusions remains a challenge. For this purpose, we propose an expressive generative model in the form of a conditional variational autoencoder, which learns a distribution of the change in pose at each step of a motion sequence. Furthermore, we introduce a flexible optimization-based approach that leverages HuMoR as a motion prior to robustly estimate plausible pose and shape from ambiguous observations. Through extensive evaluations, we demonstrate that our model generalizes to diverse motions and body shapes after training on a large motion capture dataset, and enables motion reconstruction from multiple input modalities including 3D keypoints and RGB(-D) videos. See the project page at* geometry.stanford.edu/projects/humor.

1. Introduction

As humans, we are constantly moving in, interacting with, and manipulating the world around us. Thus, applications such as action recognition [79, 80] or holistic dynamic indoor scene understanding [15] require accurate perception of 3D human pose, shape, motion, contacts, and interaction. Extensive previous work has focused on estimating 2D or 3D human pose [13, 52, 53], shape [57, 26, 67], and motion [37] from videos. These are challenging problems due to the large space of articulations, body shape, and appearance variations. Even the best methods struggle to accurately capture a wide variety of motions from varying input modalities, producing noisy or overly-smoothed motions (especially at ground contact, *i.e.*, footskate), and struggle with occlusions (*e.g.*, walking behind a couch as in Fig. 1).

We focus on the problem of building a robust human motion model that can address these challenges. To date, most motion models directly represent sequences of likely poses — *e.g.*, in PCA space [55, 77, 70] or via future-predicting autoregressive processes [75, 76, 61]. However, purely pose-based predictions either make modeling environment interactions and generalization beyond training poses difficult, or quickly diverge from the space of realistic motions. On the other hand, explicit physical dynamics models [63, 43, 69, 62, 12, 11] are resource intensive and require knowledge of unobservable physical quantities. While generative models potentially offer the required flexibility, building an *expressive*, *generalizable* and *robust* model for *realistic* 3D human motions remains an open problem.

To address this, we introduce a learned, autoregressive, generative model that captures the *dynamics* of 3D human

Generating new animation data

Both diffusion and LLMs can generate animation

- LLMs: Body poses (root translation + rotation, joint rotations) can be tokenized with RVQ
- Diffusion + Unet: Body poses over time can also be represented as real-valued arrays

MotionGPT: Text-to-motion

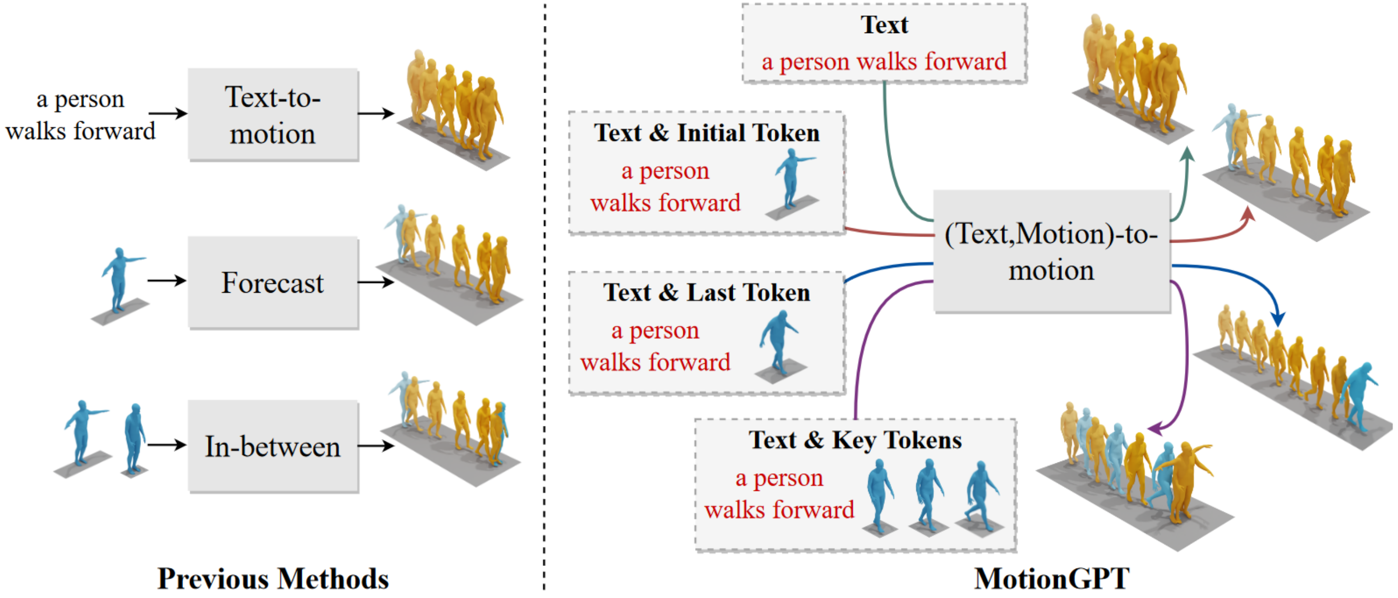


Figure 1: This work proposes a novel human motion generation method via fine-tuned LLMs, named **MotionGPT**. Compared with previous methods, MotionGPT has the unique ability to accept multiple control conditions and solve various motion generation tasks using a unified model.

MotionGPT: Finetuned LLMs are General-Purpose Motion Generators

Yaqi Zhang[1,2], Di Huang[4], Bin Liu[1,2*], Shixiang Tang[4], Yan Lu[4], Lu Chen[5], Lei Bai[3], Qi Chu[1,2], Nenghai Yu[1,2], Wanli Ouyang[3]

[1]University of Science and Technology of China

[2]CAS Key Laboratory of Electromagnetic Space Information

[3]Shanghai AI Laboratory [4]The University of Sydney [5]Zhejiang University

Abstract

Generating realistic human motion from given action descriptions has experienced significant advancements because of the emerging requirement of digital humans. While recent works have achieved impressive results in generating motion directly from textual action descriptions, they often support only a single modality of the control signal, which limits their application in the real digital human industry. This paper presents a **Motion** **G**eneral-**P**urpose genera**T**or (MotionGPT) that can use multimodal control signals, *e.g.*, text and single-frame poses, for generating consecutive human motions by treating multimodal signals as special input tokens in large language models (LLMs). Specifically, we first quantize multimodal control signals into discrete codes and then formulate them in a unified prompt instruction to ask the LLMs to generate the motion answer. Our MotionGPT demonstrates a unified human motion generation model with multimodal control signals by tuning a mere 0.4% of LLM parameters. To the best of our knowledge, MotionGPT is the first method to generate human motion by multimodal control signals, which we hope can shed light on this new direction. Codes shall be released upon acceptance. Visit our webpage at `https://qiqiapink.github.io/MotionGPT/`.

1 Introduction

https://arxiv.org/abs//2306.10900

Human motion is pivotal in various applications such as video gaming, filmmaking, and virtual reality. Recent advancements in AI [41; 48; 38; 40; 39; 30; 25] have paved the way for novel approaches to motion creation, enabling various control conditions including textual descriptions, music pieces, and human poses. However, one significant shortcoming of existing works [32; 50; 43; 31; 52] is that they only target a single type of control condition, greatly limiting their applications in the real world, *e.g.*, unable to generate motion sequences conditioned on text descriptions and several keyframe human poses. To facilitate such applications, it is important to develop a unified human motion generation framework that can efficiently utilize multiple control signals simultaneously.

This paper proposes a novel and more unified framework for text-motion generation. The framework facilitates the generation of human motions using multiple control conditions, formulated as $output\_motion = f(text, task, input\_motion)$. Newly added inputs $task$ and $input\_motion$ represent the task and given motion prompts, respectively. Here, $task$ indicates the specific task the model should adapt to, while $input\_motion$ provides the keyframe poses corresponding to the given task. This framework is a departure from traditional text-motion generation models as the introduction of $input\_motion$ enables more precise control. For example, given an $input\_motion$ and set the $task$ as "generate motion given init poses", the model should compensate for the subsequent frames

*Corresponding author

MDM

Text-to-motion similar to MotionGPT, but using diffusion models

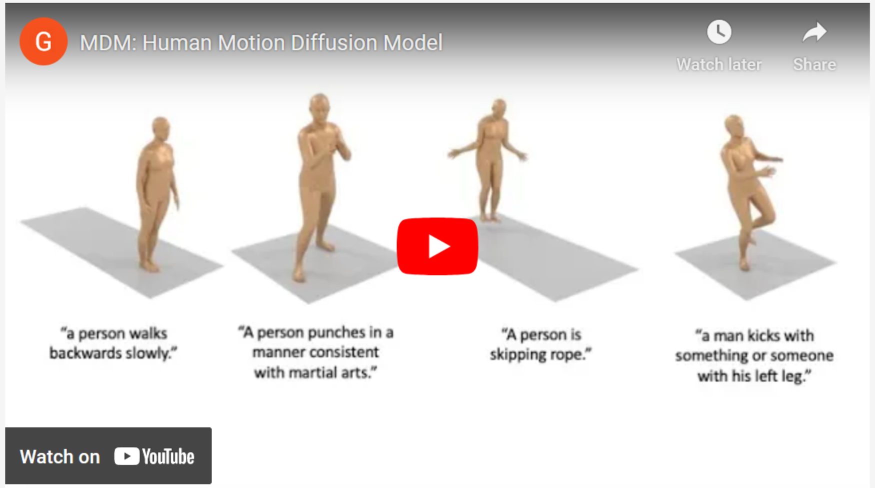
https://guytevet.github.io/mdm-page/

MDM: Human Motion Diffusion Model

ICLR2023 (Top-25%)

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, Amit H. Bermano
Tel Aviv University, Israel

arXiv | Code | Demo



Abstract

Natural and expressive human motion generation is the holy grail of computer animation. It is a challenging task, due to the diversity of possible motion, human perceptual sensitivity to it, and the difficulty of accurately describing it. Therefore, current generative solutions are either low-quality or limited in expressiveness. Diffusion models, which have already shown remarkable generative capabilities in other domains, are promising candidates for human motion due to their many-to-many nature, but they tend to be resource hungry and hard to control. In this paper, we introduce Motion Diffusion Model (MDM), a carefully adapted classifier-free diffusion-based generative model for the human motion domain. MDM is transformer-based, combining insights from motion generation literature. A notable design-choice is the prediction of the sample, rather than the noise, in each diffusion step. This facilitates the use of established geometric losses on the locations and velocities of the motion, such as the foot contact loss. As we demonstrate, MDM is a generic approach, enabling different modes of conditioning, and different generation tasks. We show that our model is trained with lightweight resources and yet achieves state-of-the-art results on leading benchmarks for text-to-motion and action-to-motion.

EMDM

Faster variant of MDM

EMDM: Efficient Motion Diffusion Model for Fast, High-Quality Motion Generation

Wenyang Zhou[1] Zhiyang Dou[2,5] Zeyu Cao[1] Zhouyingcheng Liao[2] Jingbo Wang[3]
Wenjia Wang[2] Yuan Liu[2] Taku Komura[2] Wenping Wang[4] Lingjie Liu[5]
[1]University of Cambridge [2]University of Hong Kong [3]Shanghai AI Laboratory
[4]Texas A&M University [5]University of Pennsylvania
https://frank-zy-dou.github.io/projects/EMDM/index.html

Abstract

We introduce Efficient Motion Diffusion Model (EMDM) for fast and high-quality human motion generation. Although previous motion diffusion models have shown impressive results, they struggle to achieve fast generation while maintaining high-quality human motions. Motion latent diffusion has been proposed for efficient motion generation. However, effectively learning a latent space can be non-trivial in such a two-stage manner. Meanwhile, accelerating motion sampling by increasing the step size, e.g., DDIM, typically leads to a decline in motion quality due to the inapproximation of complex data distributions when naively increasing the step size. In this paper, we propose EMDM that allows for much fewer sample steps for fast motion generation by modeling the complex denoising distribution during multiple sampling steps. Specifically, we develop a Conditional Denoising Diffusion GAN to capture multimodal data distributions conditioned on both control signals, i.e., textual description and denoising time step. By modeling the complex data distribution, a larger sampling step size and fewer steps are achieved during motion synthesis, significantly accelerating the generation process. To effectively capture the human dynamics and reduce undesired artifacts, we employ motion geometric loss during network training, which improves the motion quality and training efficiency. As a result, EMDM achieves a remarkable speedup at the generation stage while maintaining high-quality motion generation in terms of fidelity and diversity.

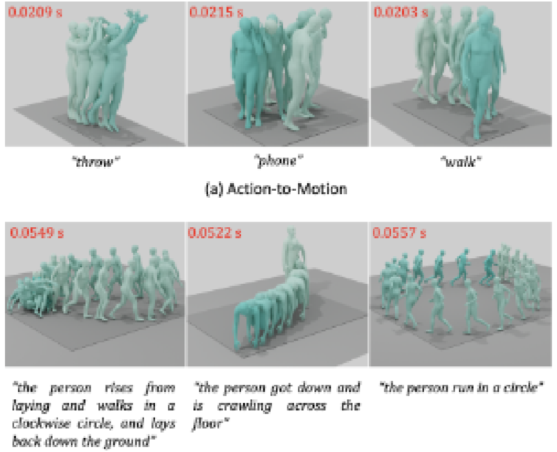


Figure 1. EMDM efficiently produces high-quality human motion aligned with input conditions in a short runtime. The average running time for EMDM in action-to-motion and text-to-motion tasks is 0.02s and 0.05s per sequence, respectively. For reference, the corresponding times for MDM [55] are 2.5s and 12.3s. We deepen the color of the character w.r.t. the time step of the sequence.

1. Introduction

Tremendous efforts have been made for conditional human motion generation with different modalities, including action labels [8, 11, 24, 36, 61], textual descriptions [1, 7, 12, 13, 18, 21, 37, 55, 60], and audio [2, 23, 25, 27], just to name a few. The diffusion model [16, 40, 51] has been at the forefront of these advancements [7, 20, 49, 55, 65], due to its promise in effectively capturing the target distribution of diverse body motions. However, these models struggle to achieve fast motion generation while maintaining high motion quality. For instance, MDM [55] produces high-quality motion while it takes around 12 seconds for a sequence given a textual description. Such low efficiency limits the effectiveness of previous works in various real-world applications, such as online motion synthesis.

To enhance motion generation efficiency, recent approaches, such as MLD [7], propose motion latent diffusion. This involves firstly learning a latent space of body motion and subsequently conducting latent diffusion. However, such a two-stage approach relies on effectively embedding the motion in the first stage—an endeavour that can be a non-trivial effort but is critical for the subsequent latent diffusion model. Given that the standard number of

Sparse keyframing

Robust Motion In-betweening

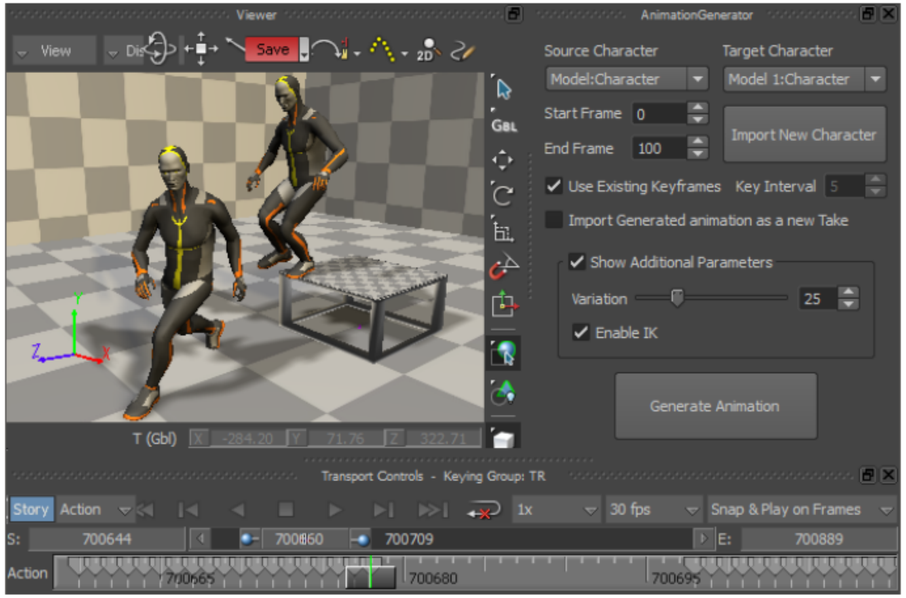


Fig. 5. **Generating animations inside MotionBuilder.** On the left is a scene where the last seed frame and target keyframe are visible. On the right is our user interface for the plugin that allows, among other things, to specify the level of scheduled target noise for the next generation through the *variation* parameter, and to use the network's contact predictions to apply IK. On the bottom is the timeline where the gap of missing motion is visible.

Robust Motion In-betweening

FÉLIX G. HARVEY, Polytechnique Montreal, Canada, Mila, Canada, and Ubisoft Montreal, Canada
MIKE YURICK, Ubisoft Montreal, Canada
DEREK NOWROUZEZAHRAI, McGill University, Canada and Mila, Canada
CHRISTOPHER PAL, CIFAR AI Chair, Canada, Polytechnique Montreal, Canada, Mila, Canada, and Element AI, Canada

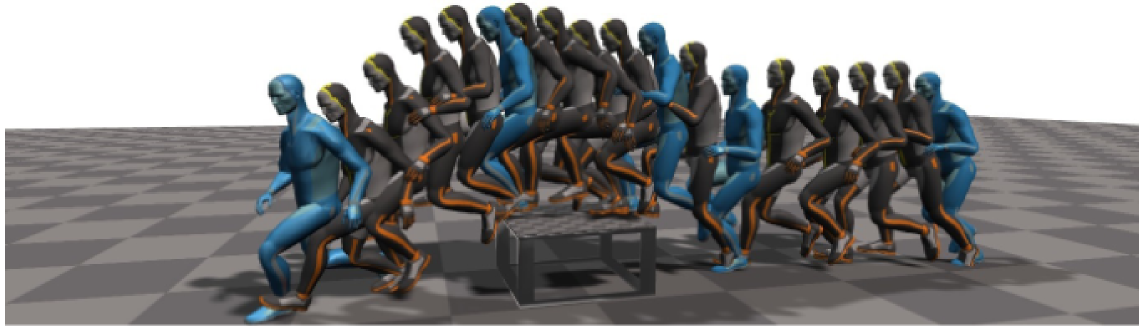


Fig. 1. Transitions automatically generated by our system between target keyframes (in blue). For clarity, only one in four generated frames is shown. Our tool allows for generating transitions of variable lengths and for sampling different variations of motion given fixed keyframes.

In this work we present a novel, robust transition generation technique that can serve as a new tool for 3D animators, based on adversarial recurrent neural networks. The system synthesises high-quality motions that use temporally-sparse keyframes as animation constraints. This is reminiscent of the job of *in-betweening* in traditional animation pipelines, in which an animator draws motion frames between provided keyframes. We first show that a state-of-the-art motion prediction model cannot be easily converted into a robust transition generator when only adding conditioning information about future keyframes. To solve this problem, we then propose two novel additive embedding modifiers that are applied at each timestep to latent representations encoded inside the network's architecture. One modifier is a *time-to-arrival embedding* that allows variations of the transition length with a single model. The other is a *scheduled target noise* vector that allows the system to be robust to target distortions and to sample different transitions given fixed keyframes. To qualitatively evaluate our method, we present a custom MotionBuilder plugin that uses our trained model to perform in-betweening in production scenarios. To quantitatively evaluate performance on transitions and generalizations to longer time horizons, we present well-defined in-betweening benchmarks on a subset of the widely used Human3.6M dataset and on LaFAN1, a novel high quality motion capture dataset that is more appropriate for transition generation. We are releasing this new dataset along with this work, with accompanying code for reproducing our baseline results.

CCS Concepts: • **Computing methodologies** → **Motion capture**; *Neural networks*.

Additional Key Words and Phrases: animation, locomotion, transition generation, in-betweening, deep learning, LSTM

ACM Reference Format:
Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. 2020. Robust Motion In-betweening. *ACM Trans. Graph.* 39, 4, Article 60 (July 2020), 12 pages. https://doi.org/10.1145/3386569.3392480

1 INTRODUCTION

Human motion is inherently complex and stochastic for long-term horizons. This is why Motion Capture (MOCAP) technologies still often surpass generative modeling or traditional animation techniques for 3D characters with many degrees of freedom. However, in modern video games, the number of motion clips needed to properly animate a complex character with rich behaviors is often very large and manually authoring animation sequences with keyframes or using a MOCAP pipeline are highly time-consuming processes. Some methods to improve curve fitting between keyframes [Ciccone et al. 2019] or to accelerate the MOCAP workflow [Holden 2018] have been proposed to improve these processes. On another

Authors' addresses: Félix G. Harvey, Polytechnique Montreal, 2500 Chemin de la Polytechique, Montreal, QC, H3T 1J4, Canada, Mila, 6666 St-Urbain Street, #200, Montreal, QC, H2S 3H1, Canada, Ubisoft Montreal, 5505 Boul Saint-Laurent, #2000, Montreal, QC, H2T 1S6, Canada, felix.gingras-harvey@polymtl.ca; Mike Yurick, Ubisoft Montreal, 5505 Boul Saint-Laurent, #2000, Montreal, QC, H2T 1S6, Canada, mike.yurick@ubisoft.com; Derek Nowrouzezahrai, McGill University, 3480 University St, Montreal, QC, H3A 0E9, Canada, Mila, 6666 St-Urbain Street, #200, Montreal, QC, H2S 3H1, Canada, derek@cim.mcgill.ca; Christopher Pal, CIFAR AI Chair, 661 University Ave., Suite 505, Toronto, ON, M5G 1M1, Canada, Polytechnique Montreal, 2500 Chemin de la Polytechique, Montreal, QC, H3T 1J4, Canada, Mila, 6666 St-Urbain Street, #200, Montreal, QC, H2S 3H1, Canada, Element AI, 6650 St-Urbain Street, #500, Montreal, QC, H2S 3G9, Canada.

0730-0301/2020/7-ART60 $15.00
https://doi.org/10.1145/3386569.3392480

MDM

MDM was also tested in inbetweening—as demonstrated by StableDiffusion, diffusion models can be modified to support inpainting/inbetweening

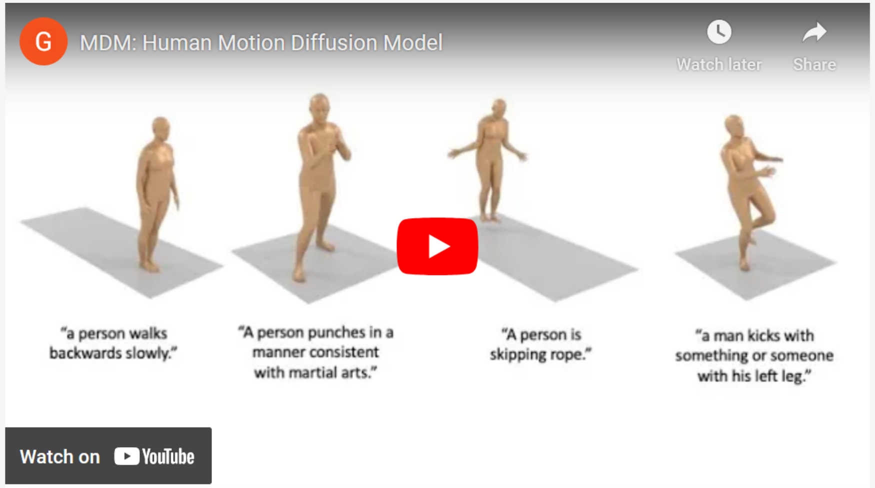
https://guytevet.github.io/mdm-page/

MDM: Human Motion Diffusion Model

ICLR2023 (Top-25%)

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, Amit H. Bermano
Tel Aviv University, Israel

arXiv | Code | Demo



Abstract

Natural and expressive human motion generation is the holy grail of computer animation. It is a challenging task, due to the diversity of possible motion, human perceptual sensitivity to it, and the difficulty of accurately describing it. Therefore, current generative solutions are either low-quality or limited in expressiveness. Diffusion models, which have already shown remarkable generative capabilities in other domains, are promising candidates for human motion due to their many-to-many nature, but they tend to be resource hungry and hard to control. In this paper, we introduce Motion Diffusion Model (MDM), a carefully adapted classifier-free diffusion-based generative model for the human motion domain. MDM is transformer-based, combining insights from motion generation literature. A notable design-choice is the prediction of the sample, rather than the noise, in each diffusion step. This facilitates the use of established geometric losses on the locations and velocities of the motion, such as the foot contact loss. As we demonstrate, MDM is a generic approach, enabling different modes of conditioning, and different generation tasks. We show that our model is trained with lightweight resources and yet achieves state-of-the-art results on leading benchmarks for text-to-motion and action-to-motion.

Interactive character control

Motion Graphs (2002)

Finds possible transition points between multiple animations

The resulting graph can be traversed to synthesize new animation combinations.

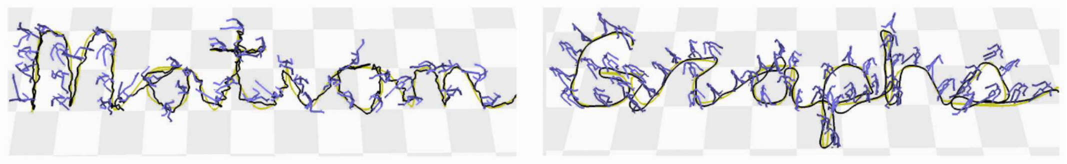
Main limitation: delayed reaction to user input

Motion Graphs

Lucas Kovar
University of Wisconsin-Madison

Michael Gleicher*
University of Wisconsin-Madison

Frédéric Pighin†
University of Southern California
Institute for Creative Technologies



Abstract

In this paper we present a novel method for creating realistic, controllable motion. Given a corpus of motion capture data, we automatically construct a directed graph called a *motion graph* that encapsulates connections among the database. The motion graph consists both of pieces of original motion and automatically generated transitions. Motion can be generated simply by building walks on the graph. We present a general framework for extracting particular graph walks that meet a user's specifications. We then show how this framework can be applied to the specific problem of generating different styles of locomotion along arbitrary paths.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: motion synthesis, motion capture, animation with constraints

1 Introduction

Realistic human motion is an important part of media like video games and movies. More lifelike characters make for more immersive environments and more believable special effects. At the same time, realistic animation of human motion is a challenging task, as people have proven to be adept at discerning the subtleties of human movement and identifying inaccuracies.

One common solution to this problem is motion capture. However, while motion capture is a reliable way of acquiring realistic human motion, by itself it is a technique for *reproducing* motion. Motion capture data has proven to be difficult to modify, and editing techniques are reliable only for small changes to a motion. This limits the utility of motion capture — if the data on hand isn't sufficiently similar to what is desired, then often there is little that can be done other than acquire more data, a time-consuming and expensive process. This in particular is a problem for applications that require motion to be synthesized dynamically, such as interactive environments.

Our goal is to retain the realism of motion capture while also giving a user the ability to control and direct a character. For example, we would like to be able to ask a character to walk around a room without worrying about having a piece of motion data that contains the correct number of steps and travels in the right directions. We also need to be able to direct characters who can perform multiple actions, rather than those who are only capable of walking around.

This paper presents a method for synthesizing streams of motions based on a corpus of captured movement while preserving the quality of the original data. Given a set of motion capture data, we compile a structure called a *motion graph* that encodes how the captured clips may be re-assembled in different ways. The motion graph is a directed graph wherein edges contain either pieces of original motion data or automatically generated transitions. The nodes then serve as choice points where these small bits of motion join seamlessly. Because our methods automatically detect and create transitions between motions, users needn't capture motions specifically designed to connect to one another. If desired, the user can tune the high-level structure of the motion graph to produce desired degrees of connectivity among different parts.

Motion graphs transform the motion synthesis problem into one of selecting sequences of nodes, or *graph walks*. By drawing upon algorithms from graph theory and AI planning, we can extract graph walks that satisfy certain properties, thereby giving us control over the synthesized motions.

To demonstrate the potential of our approach, we introduce a simple example. We were donated 78.5 seconds of motion capture, or about 2400 frames of animation, of a performer randomly walking around with both sharp and smooth turns. Since the motion was donated, we did not carefully plan out each movement, as the literature suggests is critical to successful application of motion capture data [Washburn 2001]. From this data we constructed a motion graph and used an algorithm described later in this paper to extract motions that travelled along paths sketched on the ground. Characteristic movements of the original data like sharp turns were automatically used when appropriate, as seen in Figure 1.

It is possible to place additional constraints on the desired motion. For example, we noticed that part of the motion had the character sneaking around. By labelling these frames as special, we were able to specify that at certain points along the path the character must only use sneaking movements, and at other parts of the motion it must use normal walking motions, as is also shown in Figure 1.

*e-mail:{kovar,gleicher}@cs.wisc.edu
†e-mail:pighin@ict.usc.edu

Motion Fields (2010)

Solves the latency issue.

Mathematically unnecessarily complex

Motion Fields for Interactive Character Locomotion

Yongjoon Lee[1,2*] Kevin Wampler[1†] Gilbert Bernstein[1] Jovan Popović[1,3] Zoran Popović[1]
[1]University of Washington [2]Bungie [3]Adobe Systems

Abstract

We propose a novel representation of motion data and control that enables characters with both highly agile responses to user input and natural handling of arbitrary external disturbances. The representation organizes motion data as samples in a high dimensional generalization of a vector field we call a 'motion field'. Our runtime motion synthesis mechanism freely 'flows' in the motion field and is capable of creating novel and natural motions that are highly-responsive to the real time user input, and generally not explicitly specified in the data.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: animation, motion representation, data-driven animation

1 Introduction

Human motion is a highly-varied and continuous phenomenon: it quickly adapts to different tasks, responds to external disturbances, and in general is capable of continuing locomotion from almost any initial state. As video games increasingly demand that characters move and behave in realistic ways, it is important to bring these properties of natural human motion into the virtual world. Unfortunately this is easier said than done. For instance, despite many advances in character animation techniques, creating highly agile and realistic interactive locomotion controllers remains a common but difficult task.

We propose a new motion representation for interactive character animation, termed a *motion field* which provides two key abilities: the ability for a user to control the character in real time and the ability to operate in the fully-continuous configuration space of the character. Although there exist techniques which allow one or the other of these abilities, it is the combination of the two which allows for highly agile controllers which can respond to user commands in a short amount of time.

More specifically, a motion field is a mapping which associates each possible configuration of a character with a *set* of motions describing how the character is able to move from their current state. In order to generate an animation we select a single motion from this set, follow it for a single frame, and repeat from the character's resulting state. The motion of the character thus 'flows' through the state space according to the integration process, similar to a particle flowing through a force field. However, instead of a single fixed flow, a motion field allows multiple possible motions at each frame. By using reinforcement learning to choose between these possibilities at runtime the direction of the flow can be altered, allowing the character to respond optimally to user commands.

Because motion fields allow a range of actions at every frame, a character can immediately respond to new user commands rather than waiting for pre-determined transition points as in a motion graph. This allows motion field-based controllers to be significantly more agile than their graph-based counterparts. By further altering this flow with other external methods, such as inverse kinematics or physical simulation, we also can directly integrate these techniques into the motion synthesis and control process. Furthermore, since our approach requires very little structure in the motion capture data that it uses, minimal effort is needed to generate a new controller. The primary contribution of this work lies in the combining of a continuous state representation with an optimal control framework. We find that this approach provides many advantages for character animation.

2 Related Work

In the past ten years, the bag-of-clips data structures such as motions graphs have emerged as primary sources of realistic character controllers [Lee et al. 2002; Arikan and Forsyth 2002; Kovar et al. 2002]. These structures are inherently discrete with coarse transitioning abilities that provide great computational advantages. Unfortunately, this discretization also obscures continuous properties of motion. First, it is difficult to create graphs which allow very quick responses to changes of direction or unexpected disturbances since a change to the motion can only happen when a new edge is reached [Treuille et al. 2007; McCann and Pollard 2007]. Second, because the motions are restricted to the clips which constitute the graph it is difficult to couple these methods to physical simulators and other techniques which perturb the state away from states representable by the graph. More generally, it is very hard to use a graph-based controller when the character starts from an arbitrary state configuration [Zordan et al. 2005].

Although a number of methods have been proposed to alleviate some of the representational weaknesses of pure graph-based controllers, including parameterized motion graphs [Shin and Oh 2006; Heck and Gleicher 2007], increasing the numbers of possible transitions [Arikan et al. 2005; Yin et al. 2005; Zhao and Safonova 2008] and splicing rag doll dynamics in the graph structure [Zordan et al. 2005], the fundamental issue remains: unless the representation prescribes motion at every continuous state in a way that is controllable in real time, the movement of characters will remain restricted. Hence, even when the method anticipates some user inputs [McCann and Pollard 2007], the character may react too slowly, or transition too abruptly because there is no shorter path in the graph. Similarly, when methods anticipate some types of upper-body pushes [Yin et al. 2005; Arikan et al. 2005], the character may not react at all to hand pulls or lower-body pushes.

Another group of methods use nonparametric models to learn the dynamics of character motion in a fully continuous space [Wang et al. 2008; Ye and Liu 2010; Chai and Hodgins 2005]. These techniques are generally able to synthesize starting from any initial

*e-mail: yjlee@bungie.com
†e-mail: wampler@cs.washington.edu

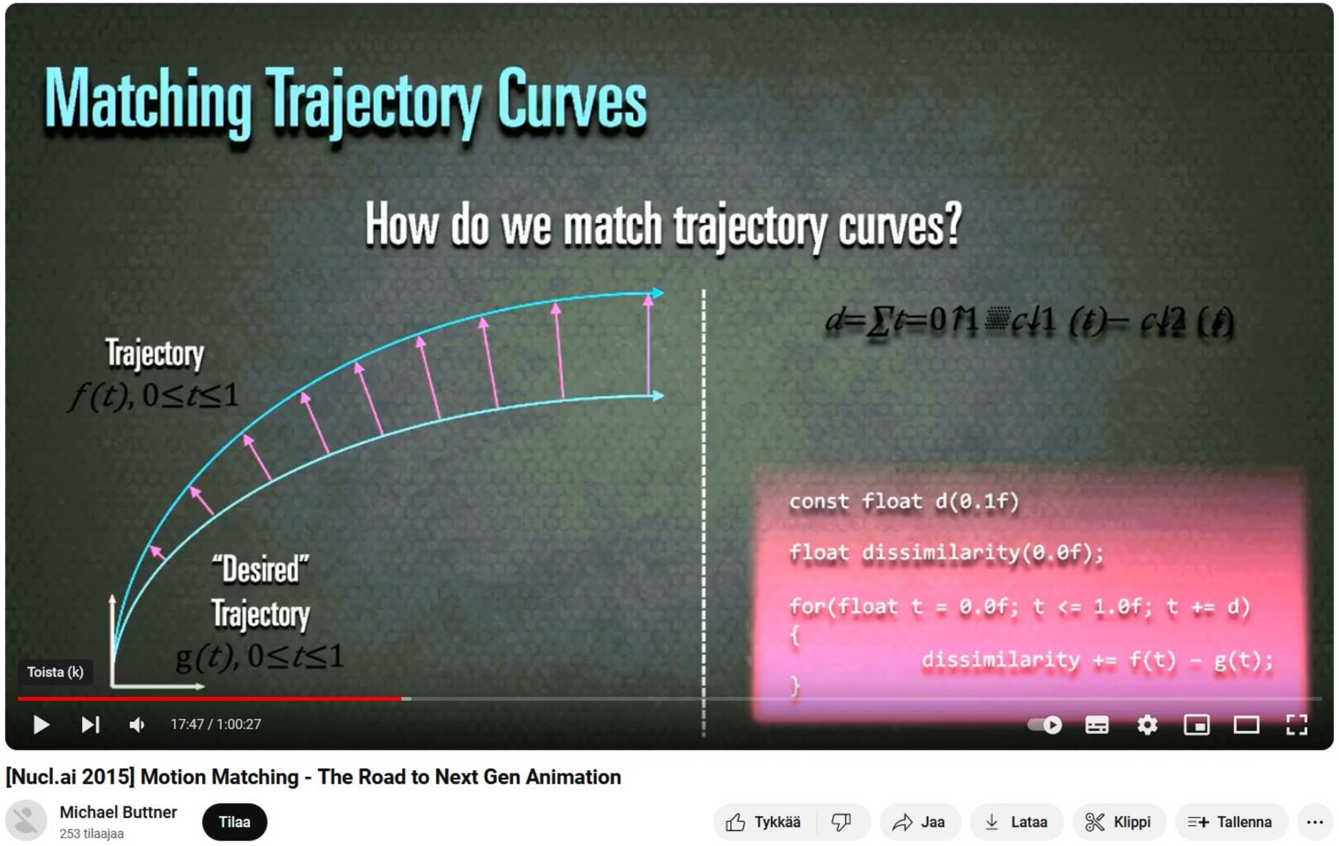
ACM Reference Format
Lee, Y., Wampler, K., Bernstein, G., Popović, J., Popović, Z. 2010. Motion Fields for Interactive Character Animation. *ACM Trans. Graph. 29*, 6, Article 138 (December 2010), 8 pages. DOI = 10.1145/1866158.1866160 http://doi.acm.org/10.1145/1866158.1866160.

Motion matching (2015)

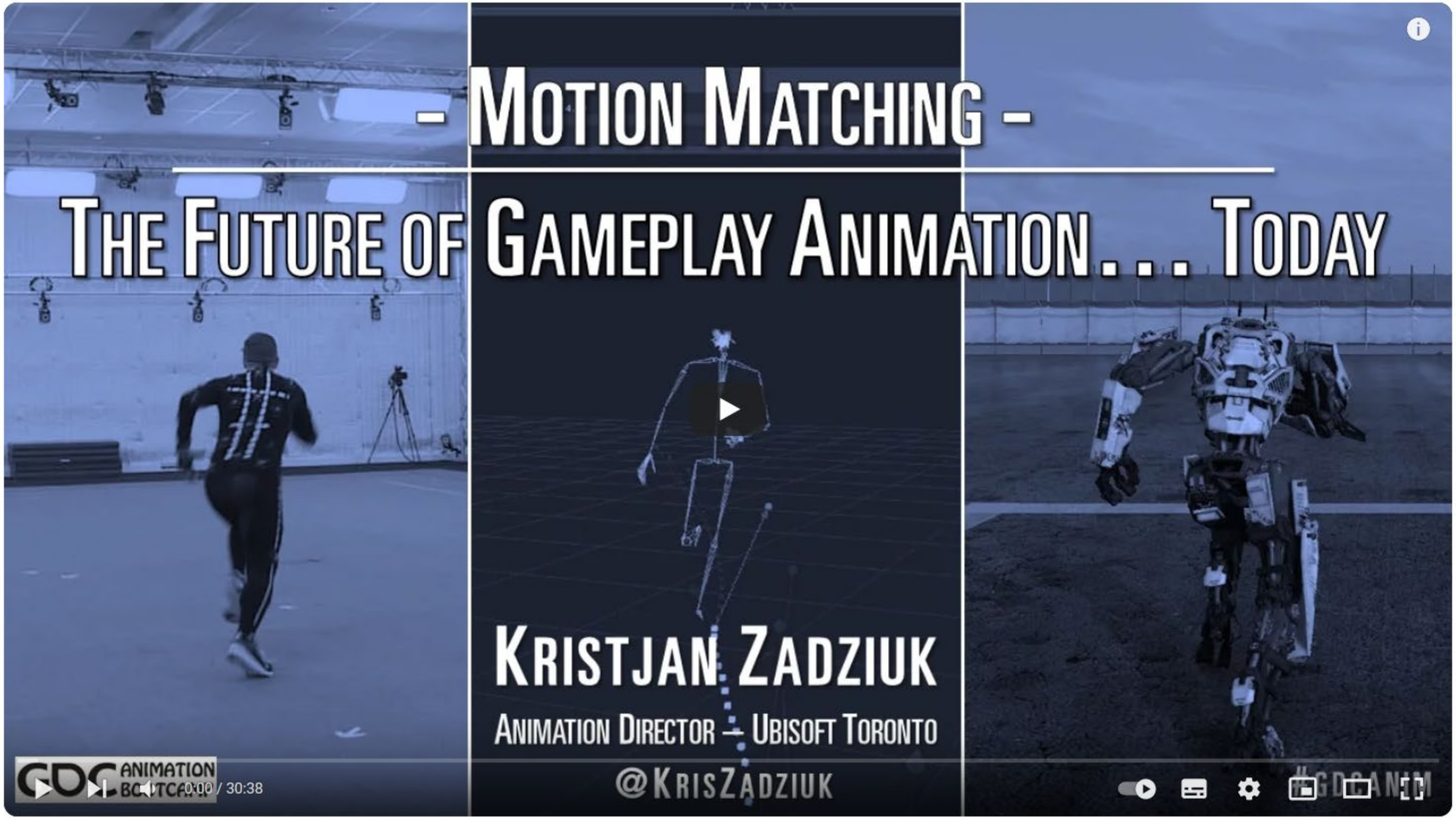
Simplification of Motion Fields, presented in a Nucl.ai 2015 talk

Creates and interactive and responsive controller from a pool of motion data, no manually constructed animation graph needed

Now used in many modern games



https://www.youtube.com/watch?v=z_wpgHFSWss&t=658s



GDC 2016 - Motion Matching, The Future of Games Animation... Today

Kristjan Zadziuk
4,75 t. tilaajaa
Tilaa

2,3 t. Jaa Lataa Klippi Tallenna ...

128 t. katselukertaa 7 vuotta sitten
Game animation has come into its own in recent years, forcing animators to wield a unique blend of art, design and technical prowess.

From GDC 2016's Animation Bootcamp ...lisää

https://www.youtube.com/watch?v=KSTn3ePDt50

How does it work?

- Preprocess: Loop over all animation data frames, precompute goal features (e.g., position and orientation of the character in the future at 0.25s, 0.5s, 1s)
- Interactive control: In every frame:
    1. Calculate goal features based on user input
    2. Loop over all data, find the frame that is the closest match based on the current pose and the goal features
    3. Start blending towards the animation starting from the found frame

With the right blend time (typically, 100ms or 200ms), the “always be blending” approach works surprisingly well

Tweaks to the base algorithm

- Maximize continuity by not blending in every frame:
  - Blend only at every N:th frame
  - Blend only if the current animation deviates too much from the goal
  - Correct the current animation’s facing direction if it only slightly deviates from the goal
- Prune and prioritize the goal features
  - For Honor: the tip of the sword is what matters the most, user’s attention focused on it
- Split the data into subsets and change the available subsets depending on game state

Take 'Control' of Animation

https://youtu.be/JH69g7yA7QM?si=yl-qwhWxeY-oSauq

GDC
502 t. tilaajaa

Tilattu | 554 | Jaa | Lataa | Klippi | Tallenna

22 t. katselukertaa 2 vuotta sitten

In this 2021 Animation Summit session, Remedy Entertainment Plc's Ilkka Kuusela and Ville Ruusutie share how they built their own technology for 'Control' from scratch, after their animation middleware was suddenly taken off the market.

Learned Motion Matching (2020)

Same algorithm but with neural networks to speed up handling of large datasets.

Learned Motion Matching

DANIEL HOLDEN, Ubisoft La Forge, Ubisoft, Canada
OUSSAMA KANOUN, Ubisoft La Forge, Ubisoft, Canada
MAKSYM PEREPICHKA, Concordia University, Canada
TIBERIU POPA, Concordia University, Canada

Fig. 1. Our method applied to various situations including navigating rough terrain, interaction with other characters, and using scene props.

In this paper we present a learned alternative to the Motion Matching algorithm which retains the positive properties of Motion Matching but additionally achieves the scalability of neural-network-based generative models. Although neural-network-based generative models for character animation are capable of learning expressive, compact controllers from vast amounts of animation data, methods such as Motion Matching still remain a popular choice in the games industry due to their flexibility, predictability, low preprocessing time, and visual quality - all properties which can sometimes be difficult to achieve with neural-network-based methods. Yet, unlike neural networks, the memory usage of such methods generally scales linearly with the amount of data used, resulting in a constant trade-off between the diversity of animation which can be produced and real world production budgets. In this work we combine the benefits of both approaches and, by breaking down the Motion Matching algorithm into its individual steps, show how learned, scalable alternatives can be used to replace each operation in turn. Our final model has no need to store animation data or additional matching meta-data in memory, meaning it scales as well as existing generative models. At the same time, we preserve the behavior of Motion Matching, retaining the quality, control, and quick iteration time which are so important in the industry.

CCS Concepts: • **Computing methodologies** → **Motion capture**.

Authors' addresses: Daniel Holden, Ubisoft La Forge, Ubisoft, 5505 St Laurent Blvd, Montreal, QC, H2T 1S6, Canada, daniel.holden@ubisoft.com; Oussama Kanoun, Ubisoft La Forge, Ubisoft, 5505 St Laurent Blvd, Montreal, QC, H2T 1S6, Canada, oussama.kanoun@ubisoft.com; Maksym Perepichka, maksym@perepichka.com, Concordia University, 1493 Saint-Catherine Street West, Montreal, Quebec, Canada, H3G 2W1; Tiberiu Popa, tiberiu.popa@concordia.ca, Concordia University, 1493 Saint-Catherine Street West, Montreal, Quebec, Canada, H3G 2W1.

0730-0301/2020/7-ART53 $15.00
https://doi.org/10.1145/3386569.3392440

Additional Key Words and Phrases: Motion Matching, Generative Models, Neural Networks, Character Animation, Animation,

ACM Reference Format:
Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa. 2020. Learned Motion Matching. *ACM Trans. Graph.* 39, 4, Article 53 (July 2020) 13 pages. https://doi.org/10.1145/3386569.3392440

1 INTRODUCTION

In interactive applications such as video games, demand for larger, more immersive and dynamic worlds has steadily made it more difficult to produce characters which can respond realistically and naturally in the exponentially growing number of different situations that are presented to them. Meanwhile, the amount of data required has also slowly grown, and AAA video games now often contain tens of thousands of unique animations that all must be triggered in the correct context [Holden 2018].

Introduced by Clavet and Büttner [2015], *Motion Matching* is a method of searching a large database of animations for the animation which best fits the given context. This method has quickly been adopted by many studios due to its simplicity, flexibility, controllability, and the quality of the motion it produces [Büttner 2019; Clavet 2016; Harrower 2018; Hussain 2019; Zinno 2019]. Rather than specifying the fine-grained animation logic via a state-graph, Motion Matching allows animators to specify the properties of the animation which should be produced, and the best fitting match is selected automatically via a nearest neighbor search. When combined with large amounts of data, Motion Matching proves a simple and effective way of dealing with the vast number of possible transitions and interactions that are required by a modern AAA video game. Additionally, since Motion Matching plays back the animation data stored in the database as-is, with only simple blending and post-processing such as inverse kinematics applied, quality is generally preserved, animators retain a level of control, and the behaviour can be tracked and debugged with appropriate tools. Finally, since it has minimal training/pre-processing time, adjustments can often

Motion Matching available for both Unity and Unreal

Unity: https://jlpm22.github.io/motionmatching-docs/

Unreal: Unreal Engine 5: Motion Matching (youtube.com), https://drive.google.com/file/d/1fzb-qz3CLT26PuYsJ31br-m49iT0ySPE/view

VR avatar control

Low-cost consumer VR hardware (Meta Quest etc) only tracks the head and hands.

Machine learning can be used to reconstruct the rest of the body based on the partial tracking data

https://dulucas.github.io/agrol/

Motion Matching can also work (2018 Master's thesis by Jukka Huiskonen, an Aalto game student): https://aaltodoc.aalto.fi/items/f871aba9-e661-4e21-afae-6bd48b574d3a

Avatars Grow Legs: Generating Smooth Human Motion from Sparse Tracking Inputs with Diffusion Model

https://arxiv.org/abs/2304.08577

Yuming Du* Robin Kips Albert Pumarola Sebastian Starke Ali Thabet Artsiom Sanakoyeu
Meta AI

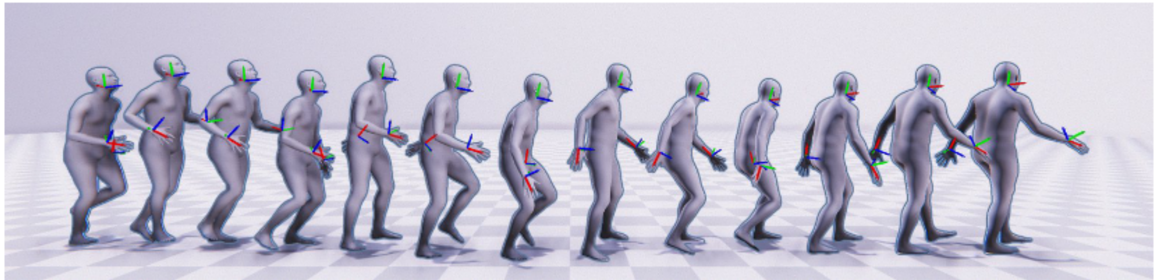


Figure 1. Full body motion synthesis based on HMD and hand controllers input. We show synthesis results of the proposed AGRoL method. RGB axes illustrate the orientation of the head and hands which serves as the input to to our model.

Abstract

With the recent surge in popularity of AR/VR applications, realistic and accurate control of 3D full-body avatars has become a highly demanded feature. A particular challenge is that only a sparse tracking signal is available from standalone HMDs (Head Mounted Devices), often limited to tracking the user's head and wrists. While this signal is resourceful for reconstructing the upper body motion, the lower body is not tracked and must be synthesized from the limited information provided by the upper body joints. In this paper, we present AGRoL, a novel conditional diffusion model specifically designed to track full bodies given sparse upper-body tracking signals. Our model is based on a simple multi-layer perceptron (MLP) architecture and a novel conditioning scheme for motion data. It can predict accurate and smooth full-body motion, particularly the challenging lower body movement. Unlike common diffusion architectures, our compact architecture can run in real-time, making it suitable for online body-tracking applications. We train and evaluate our model on AMASS motion capture dataset, and demonstrate that our approach outperforms state-of-the-art methods in generated motion accuracy and smoothness. We further justify our design choices through extensive experiments and ablation studies.

*Work done during an internship at Meta AI.
Code is available at github.com/facebookresearch/AGRoL.

1. Introduction

Humans are the primary actors in AR/VR applications. As such, being able to track full-body movement is in high demand for these applications. Common approaches are able to accurately track upper bodies only [25, 59]. Moving to full-body tracking unlocks engaging experiences where users can interact with the virtual environment with an increased sense of presence.

However, in the typical AR/VR setting there is no strong tracking signal for the entire human body – only the head and hands are usually tracked by means of Inertial Measurement Unit (IMU) sensors embedded in Head Mounted Displays (HMD) and hand controllers. Some works suggest adding additional IMUs to track the lower body joints [22, 25], those additions come at higher costs and the expense of the user's comfort [24, 27]. In an ideal setting, we want to enable high-fidelity full-body tracking using the standard three inputs (head and hands) provided by most HMDs.

Given the position and orientation information of the head and both hands, predicting full-body pose, especially the lower body, is inherently an underconstrained problem. To address this challenge, different methods rely on generative models such as normalizing flows [46] and Variational Autoencoders (VAE) [11] to synthesize lower body motions. In the realm of generative models, diffusion models have recently shown impressive results in image and video generation [21, 40, 49], especially for conditional generation. This inspires us to employ the diffusion model to generate the fully-body poses conditioned on the sparse track-

Motion capture from video

Why is it hard?

- A single view provides limited information due to occlusions
- Physical interactions hard to infer precisely => sliding and floating characters
- PhysCap (2020) was the first one to produce production-quality results without obvious artefacts

PhysCap: Physically Plausible Monocular 3D Motion Capture in Real Time https://dl.acm.org/doi/abs/10.1145/3414685.3417877

SOSHI SHIMADA, Max Planck Institute for Informatics, Saarland Informatics Campus
VLADISLAV GOLYANIK, Max Planck Institute for Informatics, Saarland Informatics Campus
WEIPENG XU, Facebook Reality Labs
CHRISTIAN THEOBALT, Max Planck Institute for Informatics, Saarland Informatics Campus

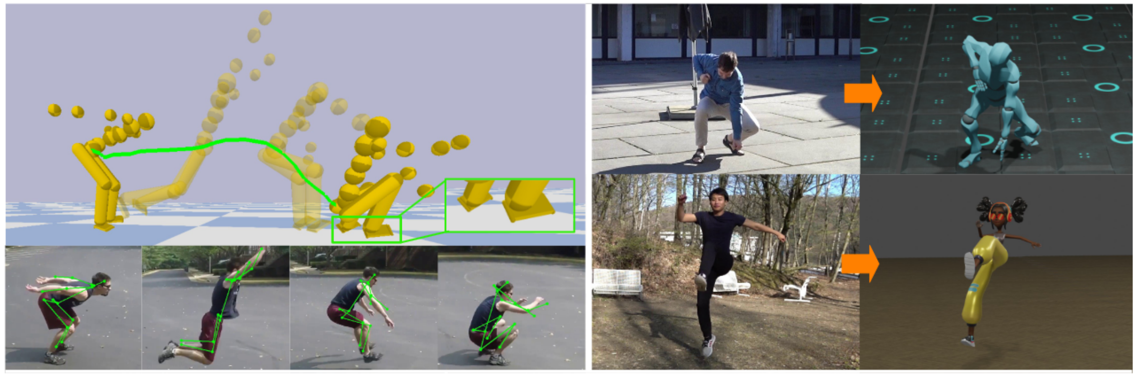


Fig. 1. *PhysCap* captures global 3D human motion in a physically plausible way from monocular videos in real time, automatically and without the use of markers. (Left:) Video of a *standing long jump* [Peng et al. 2018] and our 3D reconstructions. Thanks to its formulation on the basis of physics-based dynamics, our algorithm recovers challenging 3D human motion observed in 2D while significantly mitigating artefacts such as foot sliding, foot-floor penetration, unnatural body leaning and jitter along the depth channel that troubled earlier monocular pose estimation methods. (Right:) Since the output of *PhysCap* is environment-aware and the returned root position is global, it is directly suitable for virtual character animation, without any further post-processing. The 3D characters are taken from [Adobe 2020]. See our supplementary video for further results and visualisations.

Marker-less 3D human motion capture from a single colour camera has seen significant progress. However, it is a very challenging and severely ill-posed problem. In consequence, even the most accurate state-of-the-art approaches have significant limitations. Purely kinematic formulations on the basis of individual joints or skeletons, and the frequent frame-wise reconstruction in state-of-the-art methods greatly limit 3D accuracy and temporal stability compared to multi-view or marker-based motion capture. Further, captured 3D poses are often physically incorrect and biomechanically implausible, or exhibit implausible environment interactions (floor penetration, foot skating, unnatural body leaning and strong shifting in depth), which is problematic for any use case in computer graphics.

This work was funded by the ERC Consolidator Grant 4DRepLy (770784).

Authors' addresses: Soshi Shimada, Max Planck Institute for Informatics, Saarland Informatics Campus , sshimada@mpi-inf.mpg.de; Vladislav Golyanik, Max Planck Institute for Informatics, Saarland Informatics Campus, golyanik@mpi-inf.mpg.de; Weipeng Xu, Facebook Reality Labs, xuweipeng@fb.com; Christian Theobalt, Max Planck Institute for Informatics, Saarland Informatics Campus , theobalt@mpi-inf.mpg.de.

0730-0301/2020/12-ART235
https://doi.org/10.1145/3414685.3417877

We, therefore, present *PhysCap*, the first algorithm for physically plausible, real-time and marker-less human 3D motion capture with a single colour camera at 25 fps. Our algorithm first captures 3D human poses purely kinematically. To this end, a CNN infers 2D and 3D joint positions, and subsequently, an inverse kinematics step finds space-time coherent joint angles and global 3D pose. Next, these kinematic reconstructions are used as constraints in a real-time physics-based pose optimiser that accounts for environment constraints (*e.g.*, collision handling and floor placement), gravity, and biophysical plausibility of human postures. Our approach employs a combination of ground reaction force and residual force for plausible root control, and uses a trained neural network to detect foot contact events in images. Our method captures physically plausible and temporally stable global 3D human motion, without physically implausible postures, floor penetrations or foot skating, from video in real time and in general scenes. *PhysCap* achieves state-of-the-art accuracy on established pose benchmarks, and we propose new metrics to demonstrate the improved physical plausibility and temporal stability.

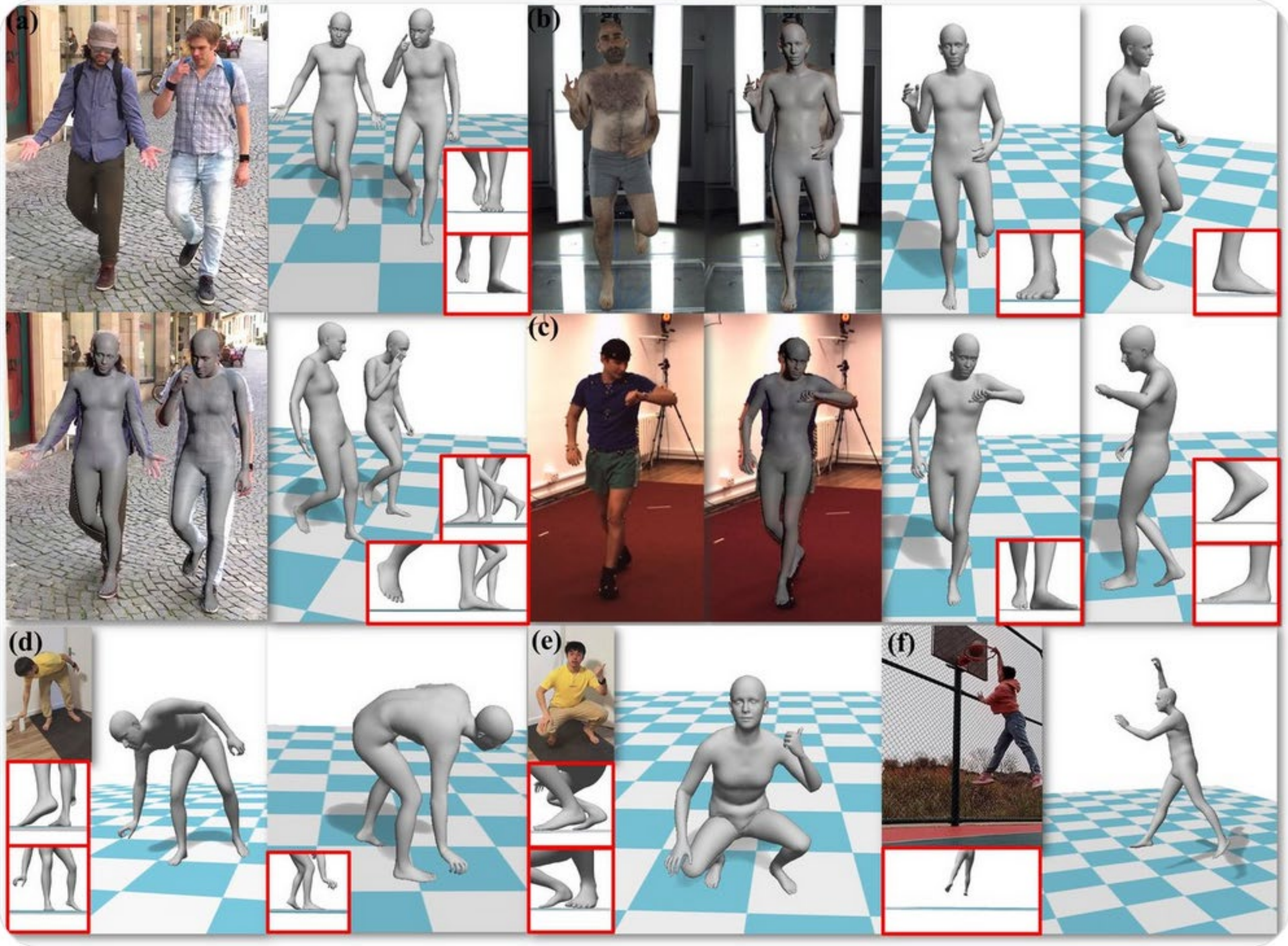
CCS Concepts: • **Computing methodologies** → **Computer graphics**; **Motion capture**.

Additional Key Words and Phrases: Monocular Motion Capture, Physics-Based Constraints, Real Time, Human Body, Global 3D

ACM Reference Format:
Soshi Shimada, Vladislav Golyanik, Weipeng Xu, and Christian Theobalt. 2020. PhysCap: Physically Plausible Monocular 3D Motion Capture in Real

AaltoMediaAI @aaltomediaai · Jul 4, 2023

An interesting neural motion capture approach that only needs a single monocular video and improves the tracking of ground contacts arxiv.org/abs/2307.01200



https://arxiv.org/abs/2307.01200

https://github.com/zju3dv/EasyMocap

Internet video

This part is the basic code for fitting SMPL[1] with 2D keypoints estimation[4][5] and CNN initialization[6].

Internet video with a mirror

CVPR21 mirror quickstart

The raw video is from Youtube.

EasyMocap is an open-source toolbox for **markerless human motion capture** and **novel view synthesis** from RGB videos. In this project, we provide a lot of motion capture demos in different settings.

python 98.8% Stars 3.1k

News

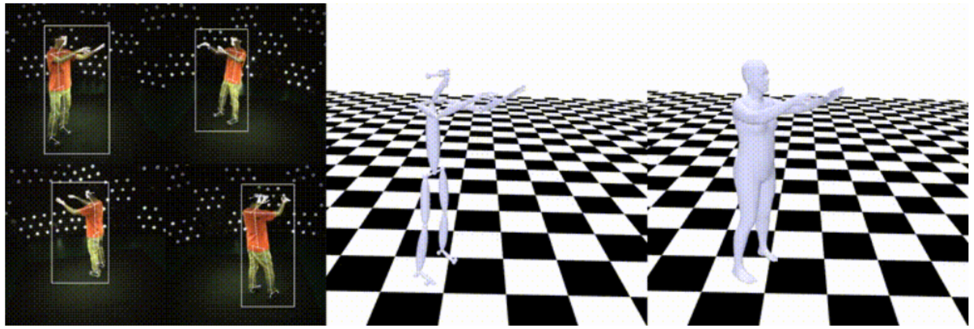
- 🎉 Our SIGGRAPH 2022 Novel View Synthesis of Human Interactions From Sparse Multi-view Videos is released! Check the documentation.
- 🎉 EasyMocap v0.2 is released! We support motion capture from Internet videos. Please check the Quick Start for more details.

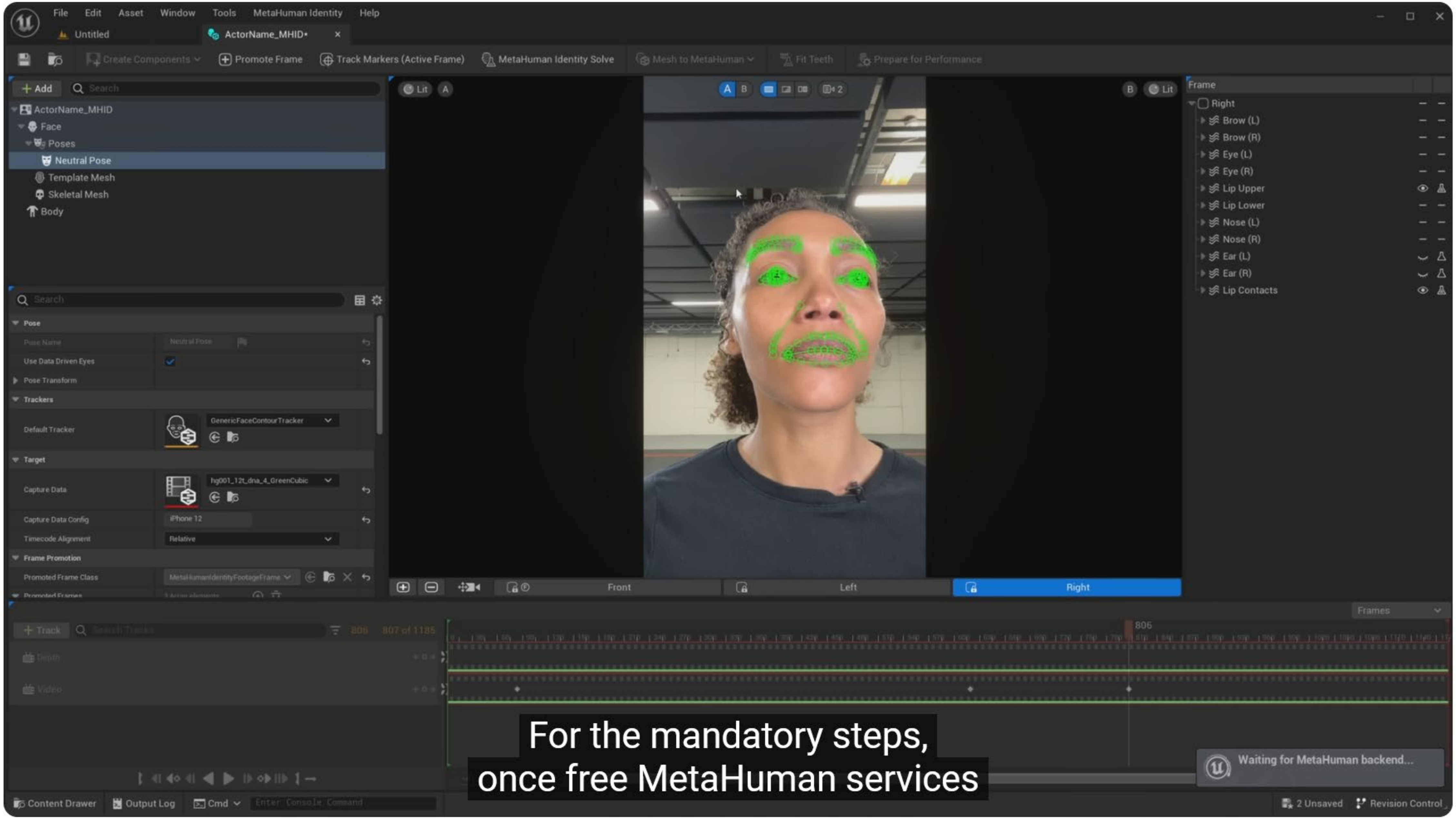
Core features

Multiple views of a single person

quickstart Open in Colab

This is the basic code for fitting SMPL[1]/SMPL+H[2]/SMPL-X[3]/MANO[2] model to capture body+hand+face poses from multiple views.





How to Use MetaHuman Animator in Unreal Engine

https://www.youtube.com/watch?v=WWLF-a68-CE

Unreal Engine
1,09 milj. tilaajaa
Tilaa
4,6 t.
Jaa
Lataa
Klippi
Tallenna

DeepMotion Animate3D

	Freemium	Starter	Innovator
	Free no credit card required	$9 per month, paid annually	$17 per month, paid annually
Plan Info	Get Started	Buy Now	Buy Now
Animation Credits	60 credits / month FREE	180 credits / month 5¢ / credit	480 credits / month 3.5¢ / credit
Single Video Length	20 seconds / clip	20 seconds / clip	30 seconds / clip
Output Formats	FBX, BVH, GLB, MP4, JPG, PNG, GIF	FBX, BVH, GLB, MP4, JPG, PNG, GIF	FBX, BVH, GLB, MP4, JPG, PNG, GIF
Commercial License	✗	✓	✓

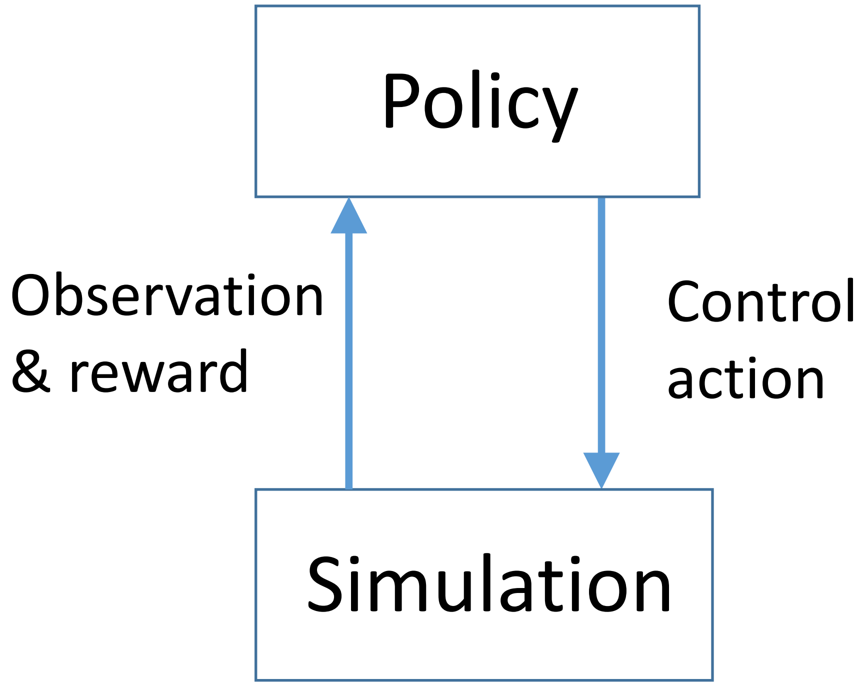
Commercial AI service for mocap from a single video

https://www.deepmotion.com/

Physically simulated characters

Physically simulated characters

- The examples above are *kinematic* => problems like feet sliding on the ground when they should not
- For best realism and interactivity, one would like to generate control commands for physics simulation
- Hard: Play QWOP to see why—even small errors can make a character lose balance
- Deep Reinforcement Learning (DRL) is now the default way to train *control policies* for simulated characters
- DRL initially explores actions randomly and then starts to repeat actions that yielded desired results (high rewards)



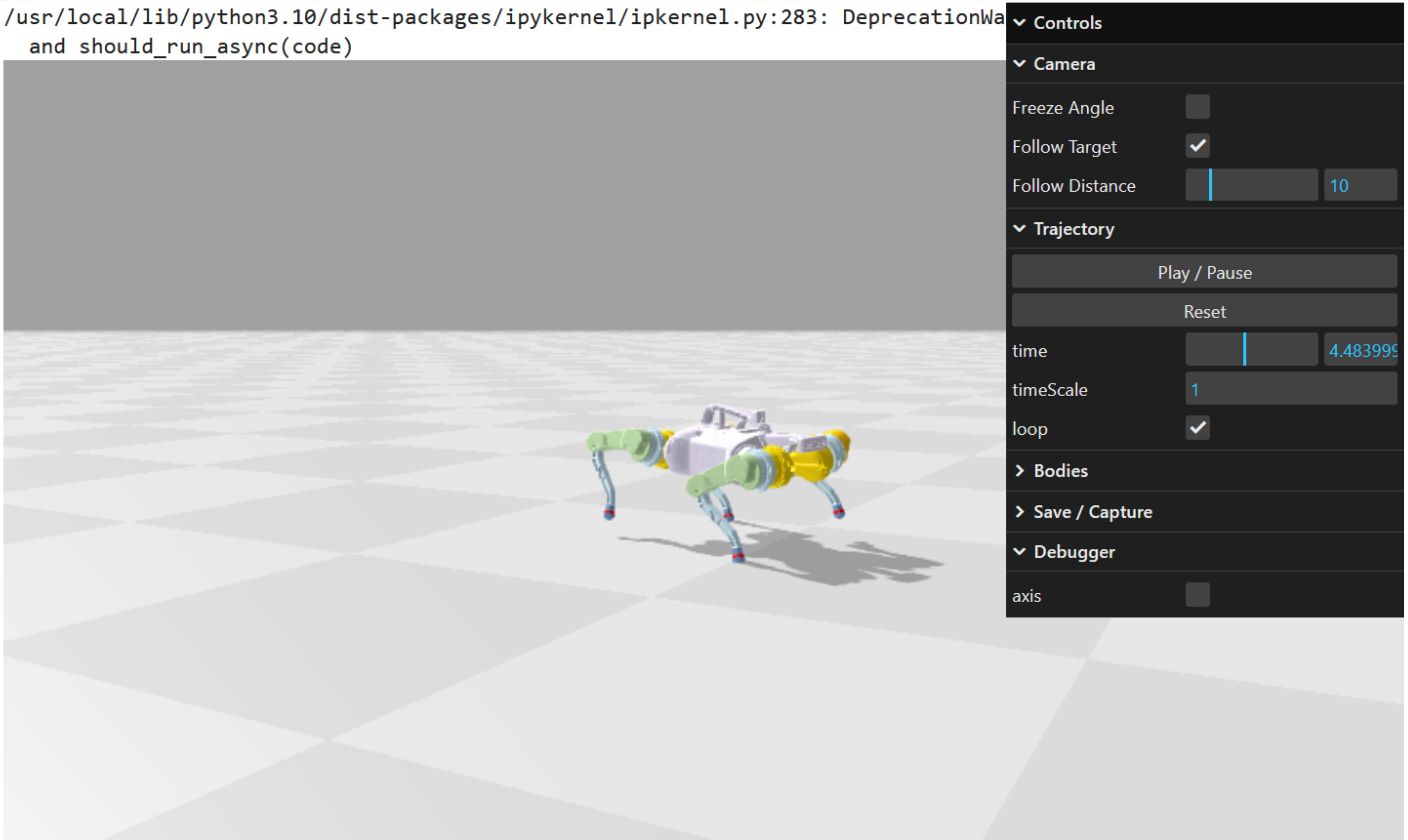
Deep Reinforcement Learning

- Very compute-heavy: training a walking humanoid can take 100M simulation steps.
- Fortunately, also highly parallelizable => best to run 1000's of simulations parallel on GPU
- GPU physics simulators: MuJoCo XLA, NVIDIA Isaac
- Colab tutorial: https://colab.research.google.com/github/google-deepmind/mujoco/blob/main/mjx/tutorial.ipynb

Humanoid walking policy trained using the tutorial in about 20 minutes

```
HTML(html.render(eval_env.brax_sys, rollout))
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWa
  and should_run_async(code)
```



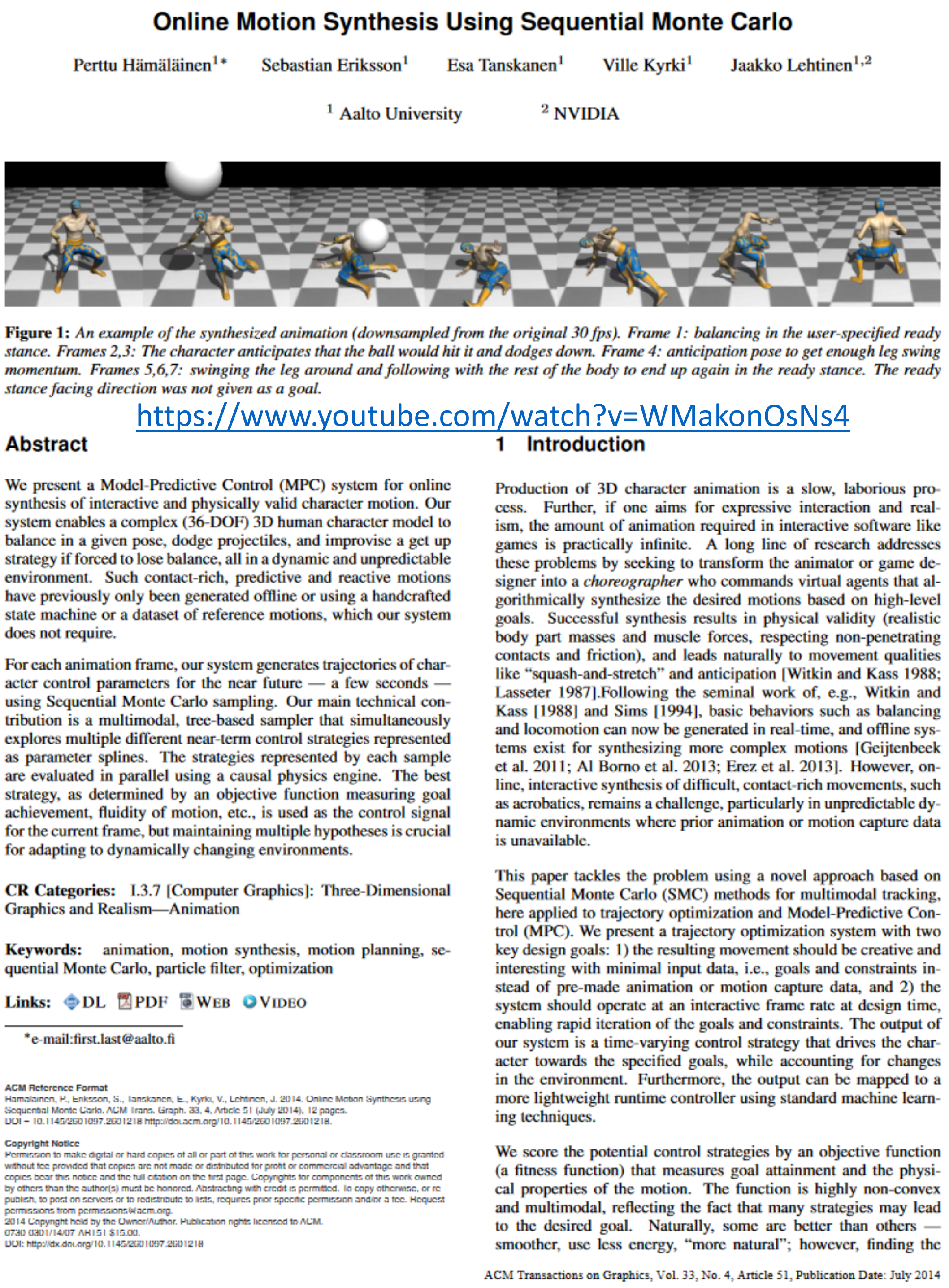
Sampling-based trajectory optimization

Prior to DRL, used to control simulated characters.

For each timestep, sample and simulate multiple candidate trajectories forward and pick the best one

A modern MuJoCo implementation: https://arxiv.org/abs/2212.00541

Gradient-based: https://arxiv.org/abs/2212.00541



Online Motion Synthesis Using Sequential Monte Carlo

Perttu Hämäläinen[1]* Sebastian Eriksson[1] Esa Tanskanen[1] Ville Kyrki[1] Jaakko Lehtinen[1,2]

[1] Aalto University [2] NVIDIA

Figure 1: *An example of the synthesized animation (downsampled from the original 30 fps). Frame 1: balancing in the user-specified ready stance. Frames 2,3: The character anticipates that the ball would hit it and dodges down. Frame 4: anticipation pose to get enough leg swing momentum. Frames 5,6,7: swinging the leg around and following with the rest of the body to end up again in the ready stance. The ready stance facing direction was not given as a goal.*

https://www.youtube.com/watch?v=WMakonOsNs4

Abstract

We present a Model-Predictive Control (MPC) system for online synthesis of interactive and physically valid character motion. Our system enables a complex (36-DOF) 3D human character model to balance in a given pose, dodge projectiles, and improve a get up strategy if forced to lose balance, all in a dynamic and unpredictable environment. Such contact-rich, predictive and reactive motions have previously only been generated offline or using a handcrafted state machine or a dataset of reference motions, which our system does not require.

For each animation frame, our system generates trajectories of character control parameters for the near future — a few seconds — using Sequential Monte Carlo sampling. Our main technical contribution is a multimodal, tree-based sampler that simultaneously explores multiple different near-term control strategies represented as parameter splines. The strategies represented by each sample are evaluated in parallel using a causal physics engine. The best strategy, as determined by an objective function measuring goal achievement, fluidity of motion, etc., is used as the control signal for the current frame, but maintaining multiple hypotheses is crucial for adapting to dynamically changing environments.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: animation, motion synthesis, motion planning, sequential Monte Carlo, particle filter, optimization

Links: DL PDF WEB VIDEO

*e-mail:first.last@aalto.fi

ACM Reference Format
Hamalainen, P., Eriksson, S., Tanskanen, E., Kyrki, V., Lehtinen, J. 2014. Online Motion Synthesis using Sequential Monte Carlo. ACM Trans. Graph. 33, 4, Article 51 (July 2014), 12 pages.
DOI = 10.1145/2601097.2601218 http://doi.acm.org/10.1145/2601097.2601218.

0730-0301/14/07-ART51 $15.00.
DOI: http://dx.doi.org/10.1145/2601097.2601218

1 Introduction

Production of 3D character animation is a slow, laborious process. Further, if one aims for expressive interaction and realism, the amount of animation required in interactive software like games is practically infinite. A long line of research addresses these problems by seeking to transform the animator or game designer into a *choreographer* who commands virtual agents that algorithmically synthesize the desired motions based on high-level goals. Successful synthesis results in physical validity (realistic body part masses and muscle forces, respecting non-penetrating contacts and friction), and leads naturally to movement qualities like "squash-and-stretch" and anticipation [Witkin and Kass 1988; Lasseter 1987].Following the seminal work of, e.g., Witkin and Kass [1988] and Sims [1994], basic behaviors such as balancing and locomotion can now be generated in real-time, and offline systems exist for synthesizing more complex motions [Geijtenbeek et al. 2011; Al Borno et al. 2013; Erez et al. 2013]. However, online, interactive synthesis of difficult, contact-rich movements, such as acrobatics, remains a challenge, particularly in unpredictable dynamic environments where prior animation or motion capture data is unavailable.

This paper tackles the problem using a novel approach based on Sequential Monte Carlo (SMC) methods for multimodal tracking, here applied to trajectory optimization and Model-Predictive Control (MPC). We present a trajectory optimization system with two key design goals: 1) the resulting movement should be creative and interesting with minimal input data, i.e., goals and constraints instead of pre-made animation or motion capture data, and 2) the system should operate at an interactive frame rate at design time, enabling rapid iteration of the goals and constraints. The output of our system is a time-varying control strategy that drives the character towards the specified goals, while accounting for changes in the environment. Furthermore, the output can be mapped to a more lightweight runtime controller using standard machine learning techniques.

We score the potential control strategies by an objective function (a fitness function) that measures goal attainment and the physical properties of the motion. The function is highly non-convex and multimodal, reflecting the fact that many strategies may lead to the desired goal. Naturally, some are better than others — smoother, use less energy, "more natural"; however, finding the

DeepMimic (2018)

Demonstrated that tracking complex human movements is possible using DRL

Policy trained separately for each movement, up to a few days per clip (GPU simulators not yet available)

Reward based on difference of simulated movement and reference mocap data

DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills

XUE BIN PENG, University of California, Berkeley
PIETER ABBEEL, University of California, Berkeley
SERGEY LEVINE, University of California, Berkeley
MICHIEL VAN DE PANNE, University of British Columbia

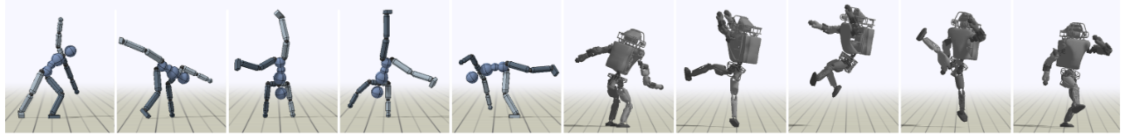


Fig. 1. Highly dynamic skills learned by imitating reference motion capture clips using our method, executed by physically simulated characters. **Left:** Humanoid character performing a cartwheel. **Right:** Simulated Atlas robot performing a spinkick.

A longstanding goal in character animation is to combine data-driven specification of behavior with a system that can execute a similar behavior in a physical simulation, thus enabling realistic responses to perturbations and environmental variation. We show that well-known reinforcement learning (RL) methods can be adapted to learn robust control policies capable of imitating a broad range of example motion clips, while also learning complex recoveries, adapting to changes in morphology, and accomplishing user-specified goals. Our method handles keyframed motions, highly-dynamic actions such as motion-captured flips and spins, and retargeted motions. By combining a motion-imitation objective with a task objective, we can train characters that react intelligently in interactive settings, e.g., by walking in a desired direction or throwing a ball at a user-specified target. This approach thus combines the convenience and motion quality of using motion clips to define the desired style and appearance, with the flexibility and generality afforded by RL methods and physics-based animation. We further explore a number of methods for integrating multiple clips into the learning process to develop multi-skilled agents capable of performing a rich repertoire of diverse skills. We demonstrate results using multiple characters (human, Atlas robot, bipedal dinosaur, dragon) and a large variety of skills, including locomotion, acrobatics, and martial arts.

CCS Concepts: • **Computing methodologies** → **Animation**; *Physical simulation*; *Control methods*; *Reinforcement learning*;

Additional Key Words and Phrases: physics-based character animation, motion control, reinforcement learning

ACM Reference Format:
Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. 2018. DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills. *ACM Trans. Graph.* 37, 4, Article 143 (August 2018), 18 pages. https://doi.org/10.1145/3197517.3201311

Authors' addresses: Xue Bin Peng, University of California, Berkeley; Pieter Abbeel, University of California, Berkeley; Sergey Levine, University of California, Berkeley; Michiel van de Panne, University of British Columbia.

0730-0301/2018/8-ART143 $15.00
https://doi.org/10.1145/3197517.3201311

1 INTRODUCTION

Physics-based simulation of passive phenomena, such as cloth and fluids, has become nearly ubiquitous in industry. However, the adoption of physically simulated characters has been more modest. Modeling the motion of humans and animals remains a challenging problem, and currently, few methods exist that can simulate the diversity of behaviors exhibited in the real world. Among the enduring challenges in this domain are generalization and directability. Methods that rely on manually designed controllers have produced compelling results, but their ability to generalize to new skills and new situations is limited by the availability of human insight. Though humans are adept at performing a wide range of skills themselves, it can be difficult to articulate the internal strategies that underly such proficiency, and more challenging still to encode them into a controller. Directability is another obstacle that has impeded the adoption of simulated characters. Authoring motions for simulated characters remains notoriously difficult, and current interfaces still cannot provide users with an effective means of eliciting the desired behaviours from simulated characters.

Reinforcement learning (RL) provides a promising approach for motion synthesis, whereby an agent learns to perform various skills through trial-and-error, thus reducing the need for human insight. While deep reinforcement learning has been demonstrated to produce a range of complex behaviors in prior work [Duan et al. 2016; Heess et al. 2016; Schulman et al. 2015b], the quality of the generated motions has thus far lagged well behind state-of-the-art kinematic methods or manually designed controllers. In particular, controllers trained with deep RL exhibit severe (and sometimes humorous) artifacts, such as extraneous upper body motion, peculiar gaits, and unrealistic posture [Heess et al. 2017].[1] A natural direction to improve the quality of learned controllers is to incorporate motion capture or hand-authored animation data. In prior work, such systems have typically been designed by layering a physics-based tracking controller on top of a kinematic animation system [Da Silva et al. 2008; Lee et al. 2010a]. This type of approach is challenging because the kinematic animation system must produce reference motions that

[1] See, for example, https://youtu.be/hx_bgoTF7bs

SFV (2018)

Like DeepMimic, but reward based on 2D pose estimation from video (OpenPose).

OpenPose paper: https://openaccess.thecvf.com/content_cvpr_2017/html/Cao_Realtime_Multi-Person_2D_CVPR_2017_paper.html

SFV: Reinforcement Learning of Physical Skills from Videos

XUE BIN PENG, University of California, Berkeley
ANGJOO KANAZAWA, University of California, Berkeley
JITENDRA MALIK, University of California, Berkeley
PIETER ABBEEL, University of California, Berkeley
SERGEY LEVINE, University of California, Berkeley

Fig. 1. Simulated characters performing highly dynamic skills learned by imitating video clips of human demonstrations. **Left:** Humanoid performing cartwheel B on irregular terrain. **Right:** Backflip A retargeted to a simulated Atlas robot.

Data-driven character animation based on motion capture can produce highly naturalistic behaviors and, when combined with physics simulation, can provide for natural procedural responses to physical perturbations, environmental changes, and morphological discrepancies. Motion capture remains the most popular source of motion data, but collecting mocap data typically requires heavily instrumented environments and actors. In this paper, we propose a method that enables physically simulated characters to learn skills from videos (SFV). Our approach, based on deep pose estimation and deep reinforcement learning, allows data-driven animation to leverage the abundance of publicly available video clips from the web, such as those from YouTube. This has the potential to enable fast and easy design of character controllers simply by querying for video recordings of the desired behavior. The resulting controllers are robust to perturbations, can be adapted to new settings, can perform basic object interactions, and can be retargeted to new morphologies via reinforcement learning. We further demonstrate that our method can predict potential human motions from still images, by forward simulation of learned controllers initialized from the observed pose. Our framework is able to learn a broad range of dynamic skills, including locomotion, acrobatics, and martial arts. (Video[1])

[1]https://xbpeng.github.io/projects/SFV/index.html

Authors' addresses: Xue Bin Peng, University of California, Berkeley; Angjoo Kanazawa, University of California, Berkeley; Jitendra Malik, University of California, Berkeley; Pieter Abbeel, University of California, Berkeley; Sergey Levine, University of California, Berkeley.

0730-0301/2018/11-ART178 $15.00
https://doi.org/10.1145/3272127.3275014

CCS Concepts: • **Computing methodologies** → **Animation**; *Physical simulation*; *Control methods*; *Reinforcement learning*; *Tracking*;

Additional Key Words and Phrases: physics-based character animation, computer vision, video imitation, reinforcement learning, motion reconstruction

ACM Reference Format:
Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. 2018. SFV: Reinforcement Learning of Physical Skills from Videos. *ACM Trans. Graph.* 37, 6, Article 178 (November 2018), 17 pages. https://doi.org/10.1145/3272127.3275014

1 INTRODUCTION

Data-driven methods have been a cornerstone of character animation for decades, with motion-capture being one of the most popular sources of motion data. Mocap data is a staple for kinematic methods, and is also widely used in physics-based character animation. Imitation of mocap clips has been shown to be an effective approach for developing controllers for simulated characters, yielding some of the most diverse and naturalistic behaviors. However, the acquisition of mocap data can pose major hurdles for practitioners, often requiring heavily instrumented environments and actors. The infrastructure required to procure such data can be prohibitive, and some activities remain exceedingly difficult to motion capture, such as large-scale outdoor sports. A more abundant and flexible source of motion data is monocular video. A staggering 300 hours of video is uploaded to YouTube every minute [Aslam 2018]. Searching and querying video sources on the web can quickly yield a large number of clips for any desired activity or behavior. However, it is a daunting challenge to extract the necessary motion information from monocular video frames, and the quality of the motions generated by previous methods still falls well behind the best mocap-based animation systems [Vondrak et al. 2012].

A general recipe for interactive physics-based characters (2019)

1. Generate a kinematic target trajectory based on current pose and goals.

2. Use a DeepMimic-like trajectory-follower policy to convert the target trajectory into control actions

DReCon: Data-Driven Responsive Control of Physics-Based Characters

https://dl.acm.org/doi/abs/10.1145/3355089.3356536

KEVIN BERGAMIN, McGill University, Canada
SIMON CLAVET, Ubisoft La Forge, Canada
DANIEL HOLDEN, Ubisoft La Forge, Canada
JAMES RICHARD FORBES, McGill University, Canada

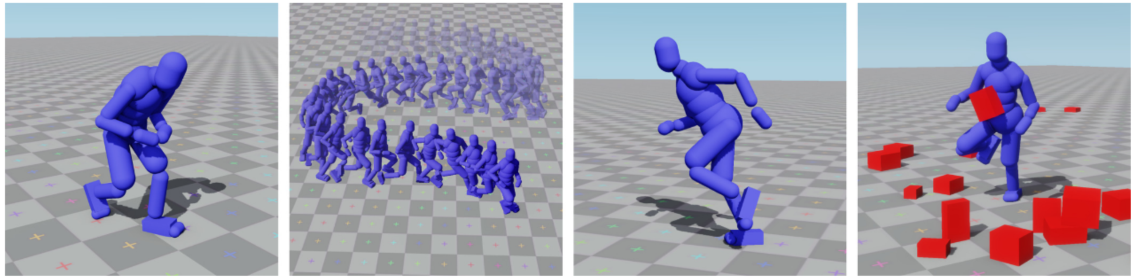


Fig. 1. A physically simulated character demonstrating user-controlled motion in real-time. The motion is interactively generated based on the content of a 10 minute motion database, in order to responsively meet user input requirements while maintaining a natural style and high robustness to perturbations.

Interactive control of self-balancing, physically simulated humanoids is a long standing problem in the field of real-time character animation. While physical simulation guarantees realistic interactions in the virtual world, simulated characters can appear unnatural if they perform unusual movements in order to maintain balance. Therefore, obtaining a high level of responsiveness to user control, runtime performance, and diversity has often been overlooked in exchange for motion quality. Recent work in the field of deep reinforcement learning has shown that training physically simulated characters to follow motion capture clips can yield high quality tracking results. We propose a two-step approach for building responsive simulated character controllers from unstructured motion capture data. First, meaningful features from the data such as movement direction, heading direction, speed, and locomotion style, are interactively specified and drive a kinematic character controller implemented using motion matching. Second, reinforcement learning is used to train a simulated character controller that is general enough to track the entire distribution of motion that can be generated by the kinematic controller. Our design emphasizes responsiveness to user input, visual quality, and low runtime cost for application in video-games.

CCS Concepts: • **Computing methodologies** → **Animation**; *Control methods*; *Reinforcement learning*; *Physical simulation*;

Authors' addresses: Kevin Bergamin, McGill University, 845 Sherbrooke Street West, Montreal, QC, H3A 0G4, Canada, kevin.bergamin@mail.mcgill.ca; Simon Clavet, Ubisoft La Forge, 5505 St Laurent Blvd, Montreal, QC, H2T 1S6, Canada, simon.clavet@ubisoft.com; Daniel Holden, Ubisoft La Forge, 5505 St Laurent Blvd, Montreal, QC, H2T 1S6, Canada, daniel.holden@ubisoft.com; James Richard Forbes, McGill University, 845 Sherbrooke Street West, Montreal, QC, H3A 0G4, Canada, james.richard.forbes@mcgill.ca.

0730-0301/2019/11-ART206 $15.00
https://doi.org/10.1145/3355089.3356536

Additional Key Words and Phrases: physically based animation, reinforcement learning, motion capture, real-time graphics

ACM Reference Format:
Kevin Bergamin, Simon Clavet, Daniel Holden, and James Richard Forbes. 2019. DReCon: Data-Driven Responsive Control of Physics-Based Characters. *ACM Trans. Graph.* 38, 6, Article 206 (November 2019), 11 pages. https://doi.org/10.1145/3355089.3356536

1 INTRODUCTION

In many interactive applications, such as video-games, physical simulation has been adopted for the modelling of a number of complex phenomenon including dynamics of cloth, fluids, and rigid bodies. However, physical simulation has seen little adoption for use in character animation, and is most often applied in limited scope such as for secondary motion, or to produce passive ragdolls that are enabled only for uncontrolled tumbling.

Historically, the core challenge of implementing physically based characters has been in producing a self-balancing controller that is robust, can produce realistic movements, and is adaptable enough to maintain control in all possible situations that may be presented. Recently, research with reinforcement learning (RL) based methods has shown promise in overcoming some of these problems, and it is now possible to design physically based character controllers that can effectively follow motion capture data with a high degree of quality [Liu and Hodgins 2018; Peng et al. 2018]. Yet, such controllers lack responsiveness when steered by user input, cannot scale easily to a large diverse set of motions, and can be prohibitively expensive, making them unsuitable for the kind of real-time character control required in video-games.

In this paper, we propose a new approach to physically based character control that allows for a high degree of responsiveness, while retaining the natural visual qualities of human motion. Our

AMP (2021)

Training method for interactive DRL controllers that use reference data for style, tasks specified by a reward function.

Human-like movement ensured using an “adversarial motion prior” learned from a mocap dataset.

Only 30min or a few hours of training using a GPU-based physics simulator

Balancing emergence and precise control.

AMP: Adversarial Motion Priors for Stylized Physics-Based Character Control

XUE BIN PENG*, University of California, Berkeley, USA
ZE MA*, Shanghai Jiao Tong University, China
PIETER ABBEEL, University of California, Berkeley, USA
SERGEY LEVINE, University of California, Berkeley, USA
ANGJOO KANAZAWA, University of California, Berkeley, USA

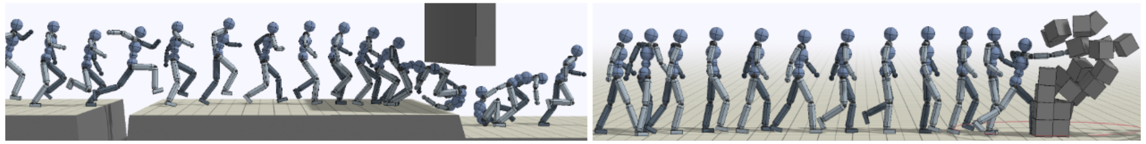


Fig. 1. Our framework enables physically simulated character to solve challenging tasks while adopting stylistic behaviors specified by unstructured motion data. **Left:** A character learns to traverse an obstacles course using a variety of locomotion skills. **Right:** A character learns to walk to and punch a target.

Synthesizing graceful and life-like behaviors for physically simulated characters has been a fundamental challenge in computer animation. Data-driven methods that leverage motion tracking are a prominent class of techniques for producing high fidelity motions for a wide range of behaviors. However, the effectiveness of these tracking-based methods often hinges on carefully designed objective functions, and when applied to large and diverse motion datasets, these methods require significant additional machinery to select the appropriate motion for the character to track in a given scenario. In this work, we propose to obviate the need to manually design imitation objectives and mechanisms for motion selection by utilizing a fully automated approach based on adversarial imitation learning. High-level task objectives that the character should perform can be specified by relatively simple reward functions, while the low-level style of the character's behaviors can be specified by a dataset of unstructured motion clips, without any explicit clip selection or sequencing. For example, a character traversing an obstacle course might utilize a task-reward that only considers forward progress, while the dataset contains clips of relevant behaviors such as running, jumping, and rolling. These motion clips are used to train an adversarial motion prior, which specifies style-rewards for training the character through reinforcement learning (RL). The adversarial RL procedure automatically selects which motion to perform, dynamically interpolating and generalizing from the dataset. Our system produces high-quality motions that are comparable to those achieved by state-of-the-art tracking-based techniques, while also being able to easily accommodate large datasets of unstructured motion clips. Composition of disparate skills emerges automatically from the motion prior, without requiring a high-level motion planner or other task-specific annotations of the motion clips. We demonstrate the effectiveness of our framework on a diverse cast of complex simulated characters and a challenging suite of motor control tasks.

CCS Concepts: • **Computing methodologies** → **Procedural animation**; **Adversarial learning**; *Control methods*.

Additional Key Words and Phrases: Wireless sensor networks, media access control, multi-channel, radio interference, time synchronization

ACM Reference Format:
Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. 2021. AMP: Adversarial Motion Priors for Stylized Physics-Based Character Control. *ACM Trans. Graph.* 40, 4, Article 144 (August 2021), 20 pages. https://doi.org/10.1145/3450626.3459670

1 INTRODUCTION

Synthesizing natural and life-like motions for virtual characters is a crucial element for breathing life into immersive experiences, such as films and games. The demand for realistic motions becomes even more apparent for VR applications, where users are provided with rich modalities through which to interact with virtual agents. Developing control strategies that are able to replicate the properties of naturalistic behaviors is also of interest for robotic systems, as natural motions implicitly encode important properties, such as safety and energy efficiency, which are vital for effective operation of robots in the real world. While examples of natural motions are commonplace, identifying the underlying characteristics that constitute these behaviors is nonetheless challenging, and more difficult still to replicate in a controller.

So what are the characteristics that constitute natural and life-like behaviors? Devising quantitative metrics of the *naturalness* of motions has been a fundamental challenge for optimization-based

*Joint first authors.

Authors' addresses: Xue Bin Peng, University of California, Berkeley, 2121 Berkeley Way, Berkeley, CA, 94704, USA, xbpeng@berkeley.edu; Ze Ma, Shanghai Jiao Tong University, 800 Dongchuan Rd, Shanghai, 200240, China, maze1234556@sjtu.edu.cn; Pieter Abbeel, University of California, Berkeley, 2121 Berkeley Way, Berkeley, CA, 94704, USA, pabbeel@cs.berkeley.edu; Sergey Levine, University of California, Berkeley, 2121 Berkeley Way, Berkeley, CA, 94704, USA, svlevine@eecs.berkeley.edu; Angjoo Kanazawa, University of California, Berkeley, 2121 Berkeley Way, Berkeley, CA, 94704, USA, kanazawa@eecs.berkeley.edu.

0730-0301/2021/8-ART144 $15.00
https://doi.org/10.1145/3450626.3459670

Policy learning with differentiable simulation

With a differentiable simulator, one might think it's simple to just backpropagate and optimize the control actions or policy.

Turns out it's not, but recent papers offer solutions.

ACCELERATED POLICY LEARNING WITH PARALLEL DIFFERENTIABLE SIMULATION

Jie Xu[1,2], Viktor Makoviychuk[1], Yashraj Narang[1], Fabio Ramos[1,3], Wojciech Matusik[2], Animesh Garg[1,4], Miles Macklin[1]

[1]NVIDIA [2]Massachusetts Institute of Technology [3]University of Sydney [4]University of Toronto

https://arxiv.org/abs/2204.07137

ABSTRACT

Deep reinforcement learning can generate complex control policies, but requires large amounts of training data to work effectively. Recent work has attempted to address this issue by leveraging differentiable simulators. However, inherent problems such as local minima and exploding/vanishing numerical gradients prevent these methods from being generally applied to control tasks with complex contact-rich dynamics, such as humanoid locomotion in classical RL benchmarks. In this work we present a high-performance differentiable simulator and a new policy learning algorithm (SHAC) that can effectively leverage simulation gradients, even in the presence of non-smoothness. Our learning algorithm alleviates problems with local minima through a smooth critic function, avoids vanishing/exploding gradients through a truncated learning window, and allows many physical environments to be run in parallel. We evaluate our method on classical RL control tasks, and show substantial improvements in sample efficiency and wall-clock time over state-of-the-art RL and differentiable simulation-based algorithms. In addition, we demonstrate the scalability of our method by applying it to the challenging high-dimensional problem of muscle-actuated locomotion with a large action space, achieving a greater than 17× reduction in training time over the best-performing established RL algorithm. More visual results are provided at: https://short-horizon-actor-critic.github.io/.

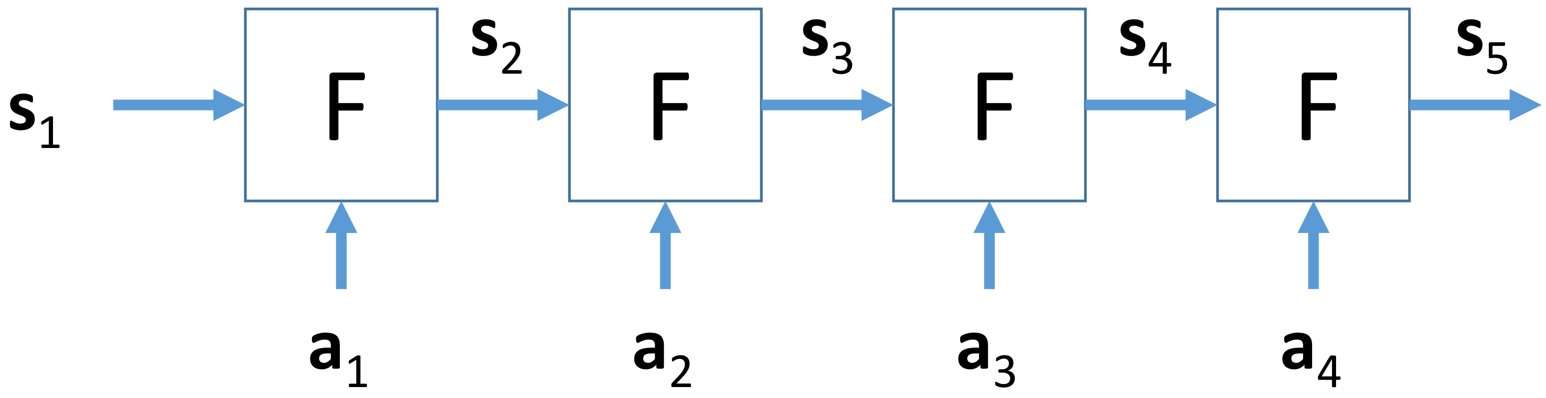
1 INTRODUCTION

Learning control policies is an important task in robotics and computer animation. Among various policy learning techniques, reinforcement learning (RL) has been a particularly successful tool to learn policies for systems ranging from robots (*e.g.*, Cheetah, Shadow Hand) (Hwangbo et al., 2019; Andrychowicz et al., 2020) to complex animation characters (*e.g.*, muscle-actuated humanoids) (Lee et al., 2019) using only high-level reward definitions. Despite this success, RL requires large amounts of training data to approximate the policy gradient, making learning expensive and time-consuming, especially for high-dimensional problems (Figure 1, Right). The recent development of differentiable simulators opens up new possibilities for accelerating the learning and optimization of control policies. A differentiable simulator may provide accurate first-order gradients of the task performance reward with respect to the control inputs. Such additional information potentially allows the use of efficient gradient-based methods to optimize policies. As recently Freeman et al. (2021) show, however, despite the availability of differentiable simulators, it has not yet been convincingly demonstrated that they can effectively accelerate policy learning in complex high-dimensional and contact-rich tasks, such as some traditional RL benchmarks. There are several reasons for this:

1. Local minima may cause gradient-based optimization methods to stall.
2. Numerical gradients may vanish/explode along the backward path for long trajectories.
3. Discontinuous optimization landscapes can occur during policy failures/early termination.

Because of these challenges, previous work has been limited to the optimization of open-loop control policies with short task horizons (Hu et al., 2019; Huang et al., 2021), or the optimization of policies for relatively simple tasks (*e.g.*, contact-free environments) (Mora et al., 2021; Du et al., 2021). In this work, we explore the question: *Can differentiable simulation accelerate policy learning in tasks with continuous closed-loop control and complex contact-rich dynamics?*

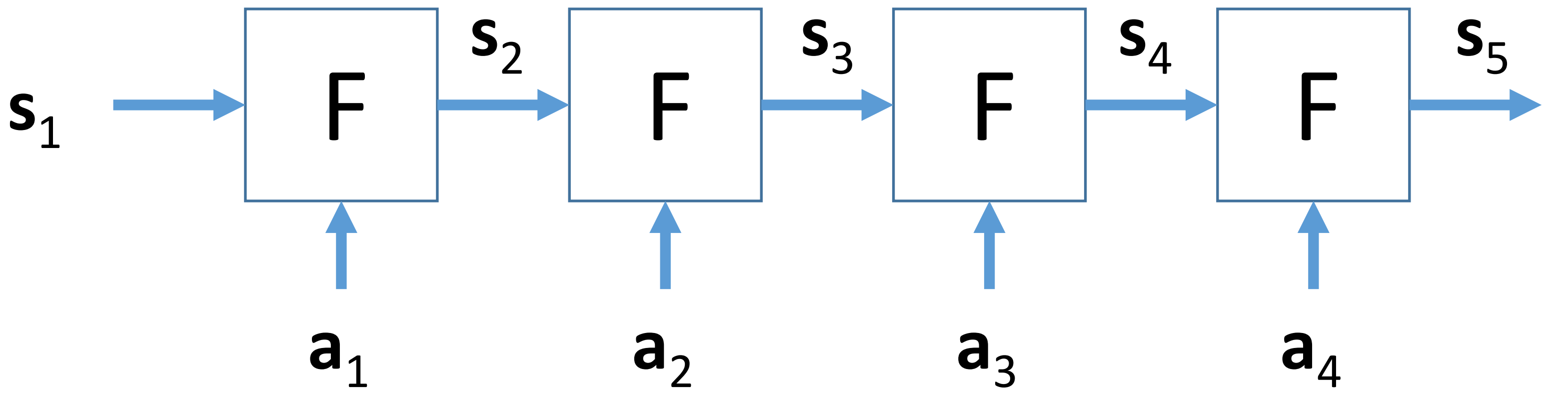
Planning through backpropagation



F is a simulator or learned forward model that produces the next simulation state when given the current state and action.

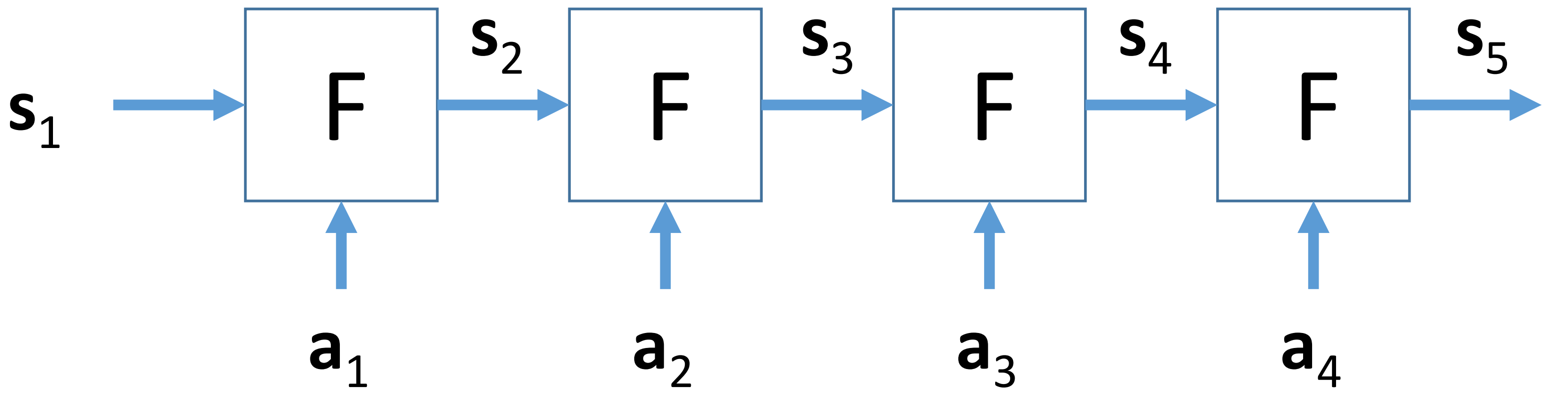
Planning through backpropagation

Objective: Maximize $\sum_i r(\mathbf{a}_i, \mathbf{s}_i)$



Planning through backpropagation

Objective: Maximize $\sum_i r(\mathbf{a}_i, \mathbf{s}_i)$



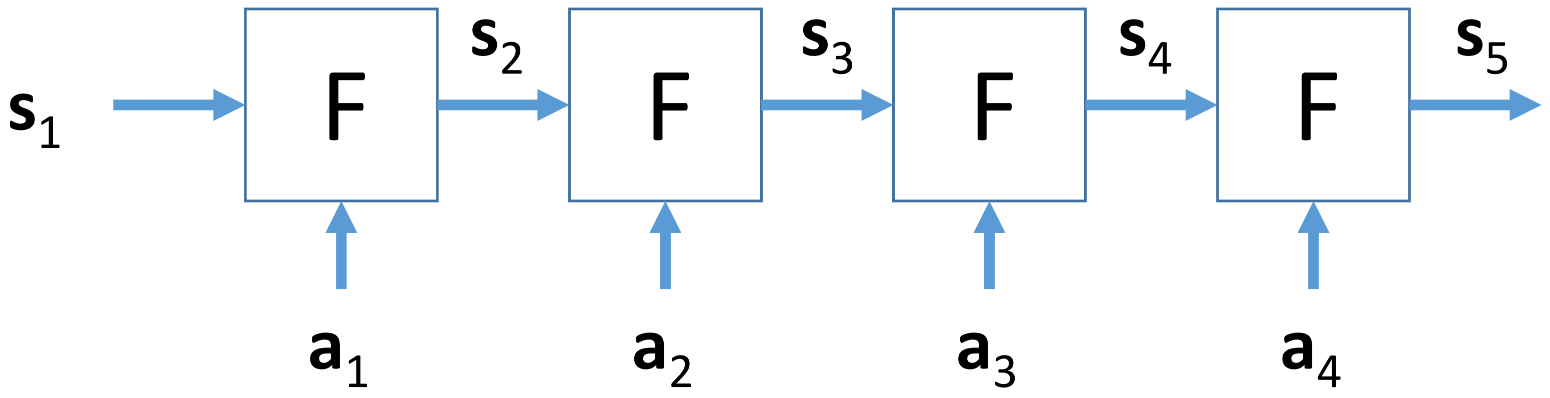
Gradient-based optimization of actions is trivial in PyTorch etc., if both **F** and $r(\mathbf{a}_i, \mathbf{s}_i)$ are differentiable

Policy optimization

Objective: $$\underset{\theta}{\operatorname{argmax}} \quad \mathbb{E}_{\mathbf{s}_0 \sim p(\mathbf{s}_0),\, \mathbf{a}_t = \pi_\theta(\mathbf{s}_t)} \left[\sum_{t=0}^{T} \gamma^t r(\mathbf{a}_t, \mathbf{s}_t) \right]$$

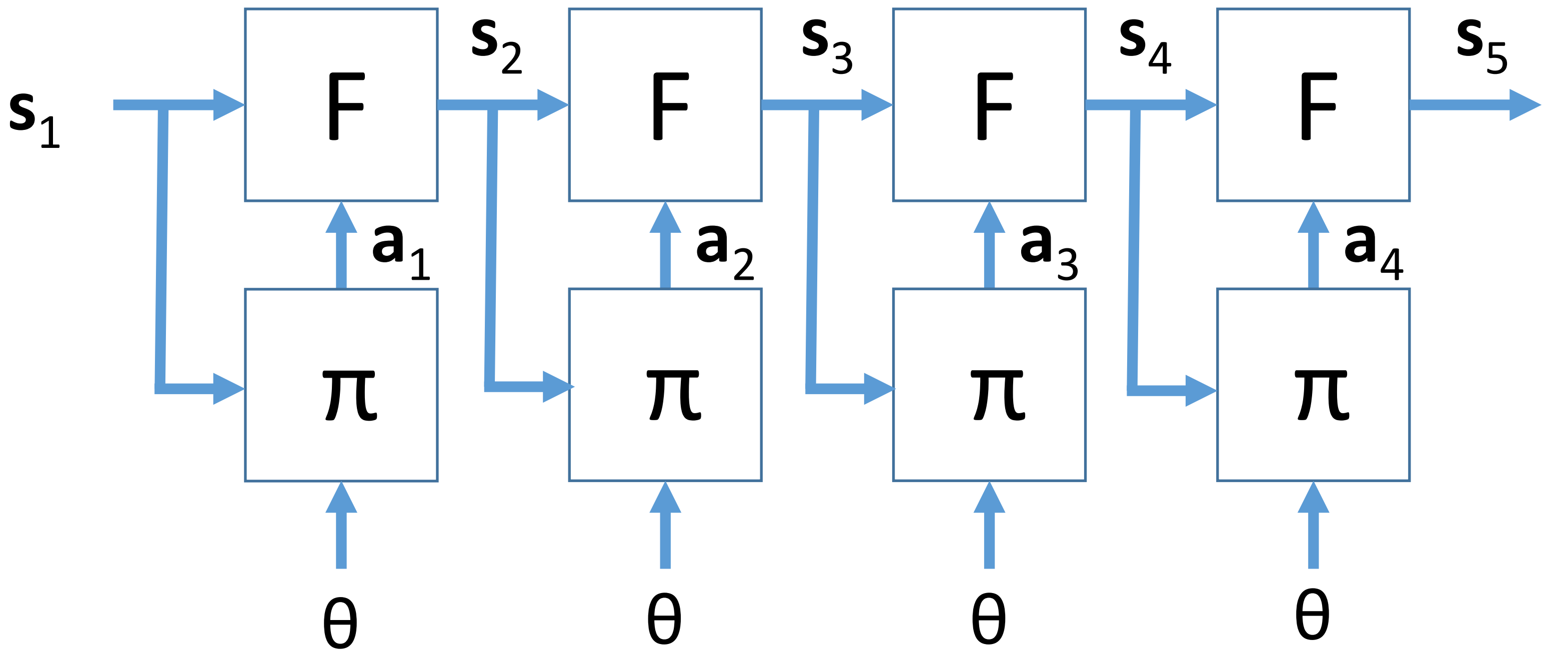
Policy optimization

Objective: $$\underset{\theta}{\operatorname{argmax}} \; \mathbb{E}_{\mathbf{s}_0 \sim p(\mathbf{s}_0),\, \mathbf{a}_t = \pi_\theta(\mathbf{s}_t)} \left[\sum_{t=0}^{T} \gamma^t r(\mathbf{a}_t, \mathbf{s}_t) \right]$$



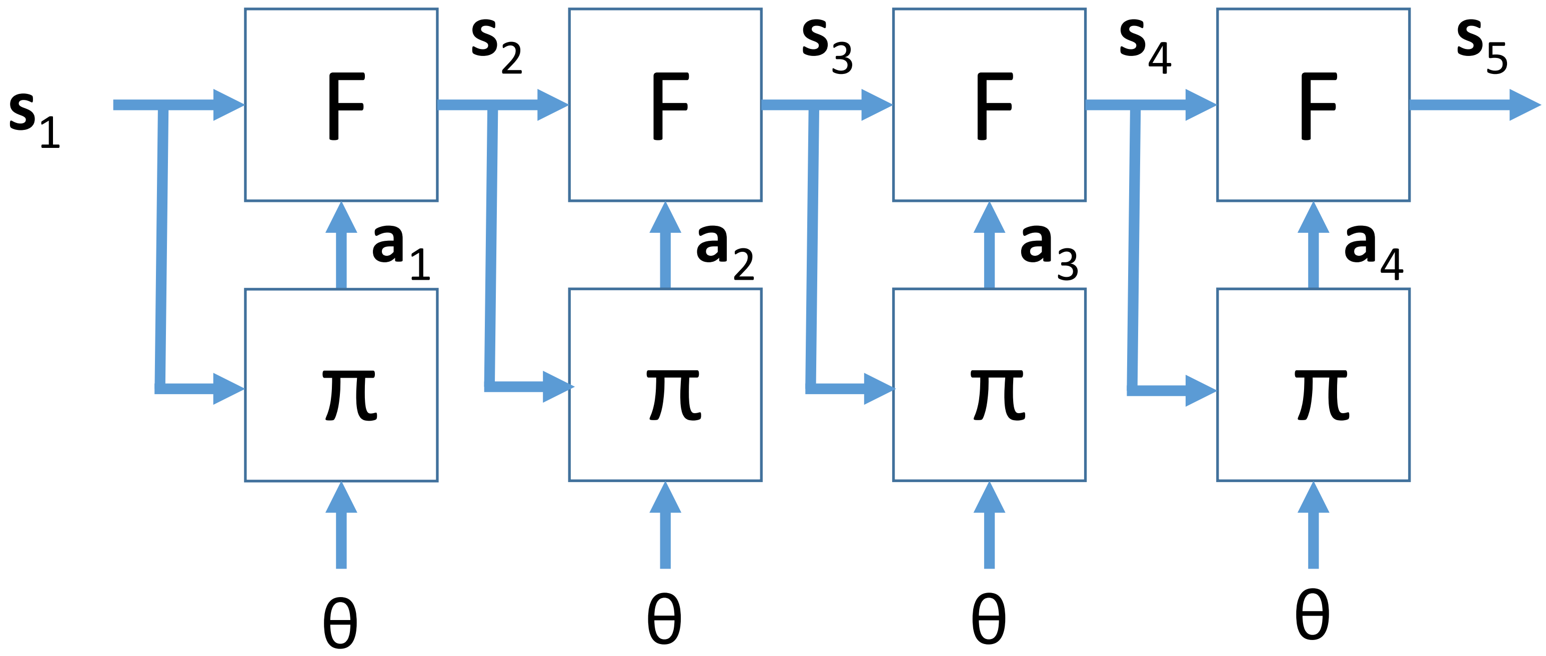
Policy optimization

Objective: $\underset{\theta}{\operatorname{argmax}} \; \mathbb{E}_{\mathbf{s}_0 \sim p(\mathbf{s}_0),\, \mathbf{a}_t = \pi_\theta(\mathbf{s}_t)} \left[\sum_{t=0}^{T} \gamma^t r(\mathbf{a}_t, \mathbf{s}_t) \right]$



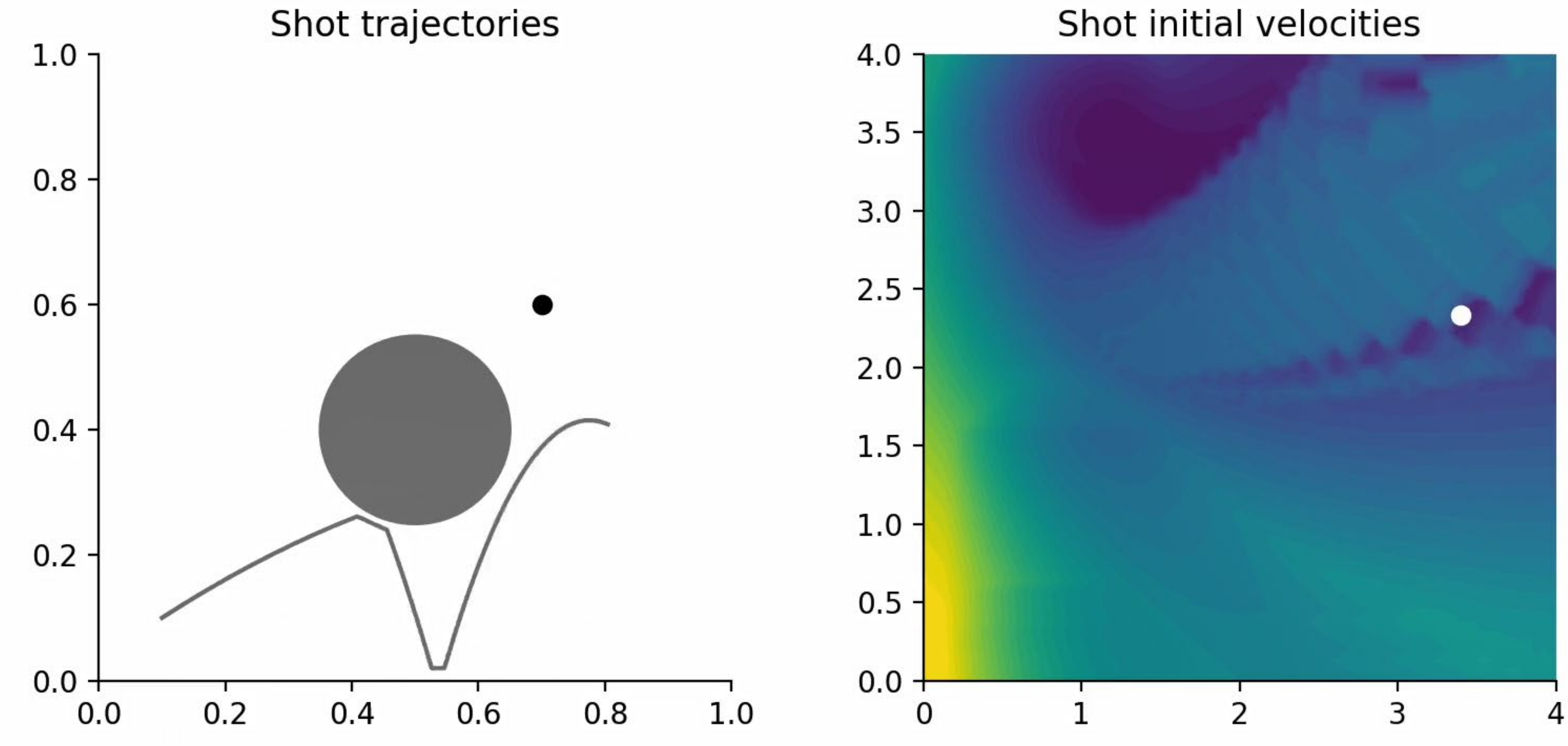
Policy optimization

Objective: $$\arg\max_{\theta} \; \mathbb{E}_{\mathbf{s}_0 \sim p(\mathbf{s}_0),\, \mathbf{a}_t = \pi_\theta(\mathbf{s}_t)} \left[\sum_{t=0}^{T} \gamma^t r(\mathbf{a}_t, \mathbf{s}_t) \right]$$



Replace expectation with minibatch average, backprop to optimize θ

Problem: Local optima



Directly optimizing initial shot vx,vy using Adam (possible because of differentiable dynamics) gets stuck in local optimum

SHAC (2022) optimizes a smoothed approximation of the true landscape

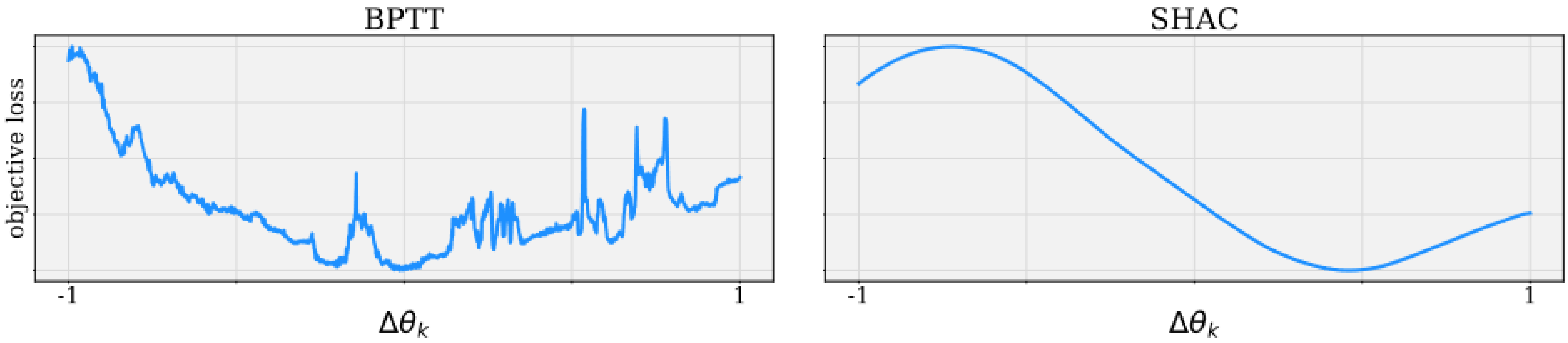


Figure 2: **Landscape comparison between BPTT and SHAC.** We select one single weight from a policy and change its value by $\Delta\theta_k \in [-1, 1]$ to plot the task loss landscapes of BPTT and SHAC w.r.t. one policy parameter. The task horizon is $H = 1000$ for BPTT, and the short horizon length for our method is $h = 32$. As we can see, longer optimization horizons lead to noisy loss landscape that are difficult to optimize, and the landscape of our method can be regarded as a smooth approximation of the real landscape.

https://arxiv.org/abs/2204.07137

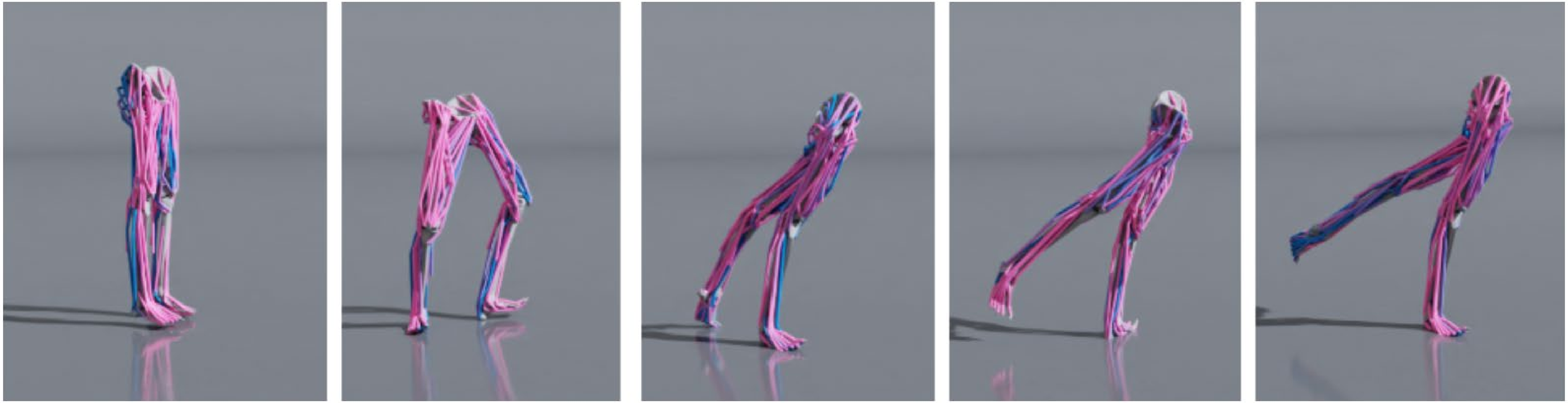


Figure 5: **Humanoid MTU**: A sequence of frames from a learned running gait. The muscle unit color indicates the activation level at the current state.

https://arxiv.org/abs/2204.07137

SuperTrack (2021)

A learned neural network forward model provides a smooth differentiable approximation of a non-differentiable simulator.

https://dl.acm.org/doi/abs/10.1145/3478513.3480527

SuperTrack: Motion Tracking for Physically Simulated Characters using Supervised Learning

LEVI FUSSELL, Ubisoft La Forge and The University of Edinburgh
KEVIN BERGAMIN, Ubisoft La Forge, Ubisoft, Canada
DANIEL HOLDEN, Ubisoft La Forge, Ubisoft, Canada

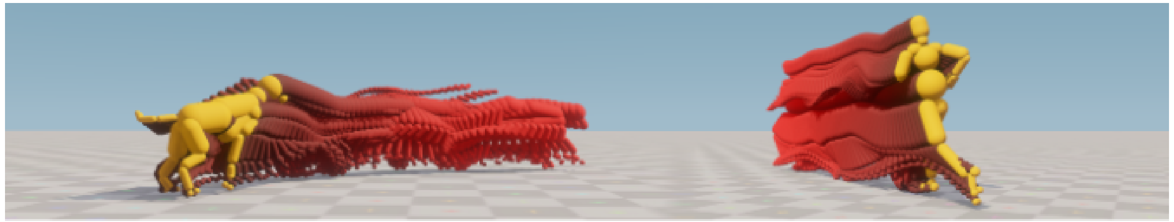


Fig. 1. Visualization of the predictions of the world model, which is used by our method as an approximate differentiable physics simulator.

In this paper we show how the task of motion tracking for physically simulated characters can be solved using supervised learning and optimizing a policy directly via back-propagation. To achieve this we make use of a world model trained to approximate a specific subset of the environment's transition function, effectively acting as a differentiable physics simulator through which the policy can be optimized to minimize the tracking error. Compared to popular model-free methods of physically simulated character control which primarily make use of Proximal Policy Optimization (PPO) we find direct optimization of the policy via our approach consistently achieves a higher quality of control in a shorter training time, with a reduced sensitivity to the rate of experience gathering, dataset size, and distribution.

CCS Concepts: • **Computing methodologies → Animation.**

Additional Key Words and Phrases: Motion Tracking, Motion Imitation, Imitation Learning, Reinforcement Learning, Character Animation, Motion Capture

ACM Reference Format:
Levi Fussell, Kevin Bergamin, and Daniel Holden. 2021. SuperTrack: Motion Tracking for Physically Simulated Characters using Supervised Learning. *ACM Trans. Graph.* 40, 6, Article 1 (December 2021), 13 pages. https://doi.org/10.1145/3478513.3480527

Authors' addresses: Levi Fussell, Ubisoft La Forge and The University of Edinburgh, 10 Crichton Street, Edinburgh, EH8 9AB, UK, levi.fussell@ed.ac.uk; Kevin Bergamin, Ubisoft La Forge, Ubisoft, 5505 St Laurent Blvd, Montreal, QC, H2T 1S6, Canada, kevin.bergamin@ubisoft.com; Daniel Holden, Ubisoft La Forge, Ubisoft, 5505 St Laurent Blvd, Montreal, QC, H2T 1S6, Canada, daniel.holden@ubisoft.com.

0730-0301/2021/12-ART1 $15.00
https://doi.org/10.1145/3478513.3480527

1 INTRODUCTION

In recent years Reinforcement Learning (RL) has been shown to consistently achieve state-of-the-art results on a large number of physically simulated character control tasks - and the ability to track high quality motion capture data with a physically simulated character is now well within the reach of both animation researchers and practitioners. Many of these works make use of Proximal Policy Optimization (PPO), a popular, on-policy Reinforcement Learning algorithm that has been shown to work effectively on a vast range of different problems and applications. However, like many Reinforcement Learning algorithms, PPO is notoriously sample inefficient, sensitive to hyper-parameter settings, class or data imbalance, gym design, reward and feature design, and even the random seed [Henderson et al. 2017]. This makes wide adoption of such methods in areas like video games difficult, as the unpredictability of the algorithm's success can make it a risky choice in a production setting.

In this work we show how the most common task in physically simulated character control - motion tracking - can instead be formulated using only supervised learning such that a control policy can be optimized for directly using back-propagation of a tracking loss. This can be interpreted in two ways: as a form of model-based learning, but also as a supervised learning problem where two models are learned simultaneously - one to predict the dynamics of the world, and another which attempts to produce the optimal actions which minimize the tracking losses. We find our approach achieves a better quality of animation within a shorter training time than PPO, with a reduced sensitivity to the rate of experience gathering and data imbalance, which we show on a large variety of challenging tracking tasks.

Our method relies on four key observations: first, that the reward or loss function for the motion tracking task is generally a simple, differentiable function of the kinematic and simulated character states. Second, that the transition function of these two individual character states is partially separable; generally the kinematic character's behaviour does not change because of the simulated

DiffMimic (2023)

https://arxiv.org/abs/2304.03274

https://github.com/jiawei-ren/diffmimic

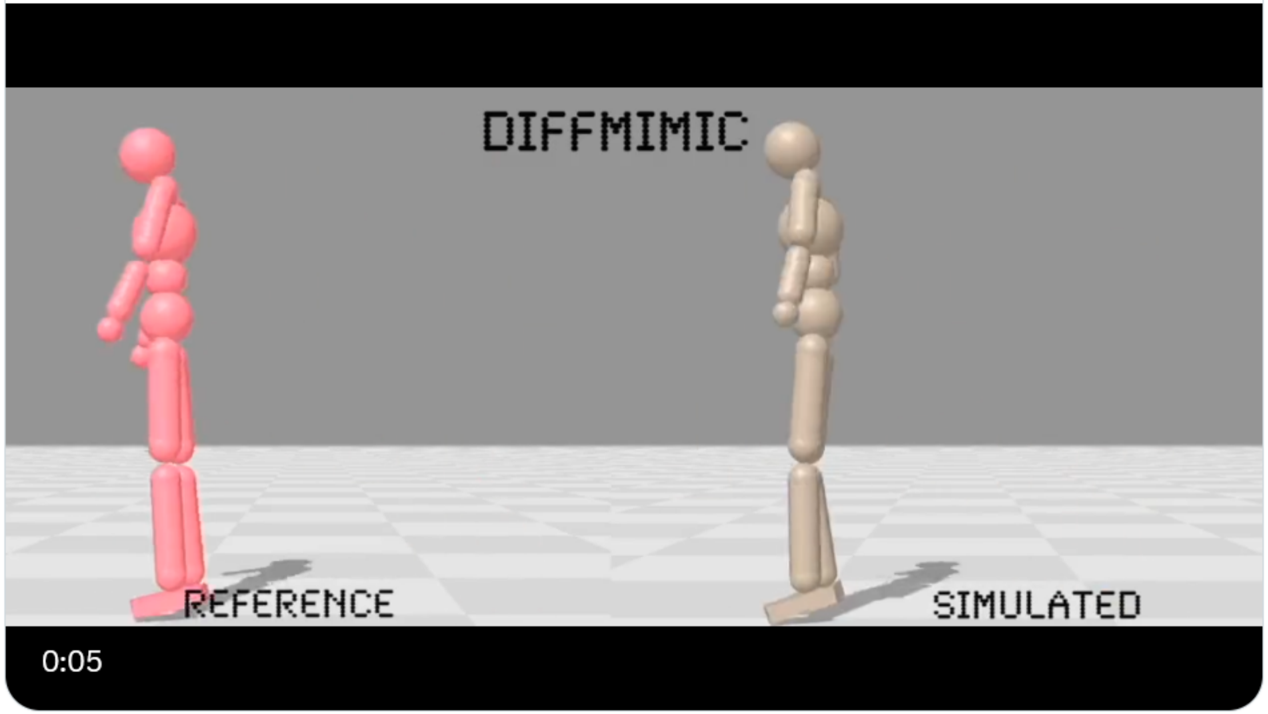
AaltoMediaAI
@aaltomediaai

Apparently, a simple variant of teacher forcing (force only if necessary) can solve the difficulty of direct policy optimization using backprop and differentiable physics. Learning a backflip in 5 minutes is not bad at all.

AK @_akhaliq · Apr 7, 2023

DiffMimic: Efficient Motion Mimicking with Differentiable Physics

abs: arxiv.org/abs/2304.03274
github: github.com/jiawei-ren/dif...



Physically controlled VR avatars (2023)

Trains a DRL policy to produce full body movements based on the VR controller and headset movements.

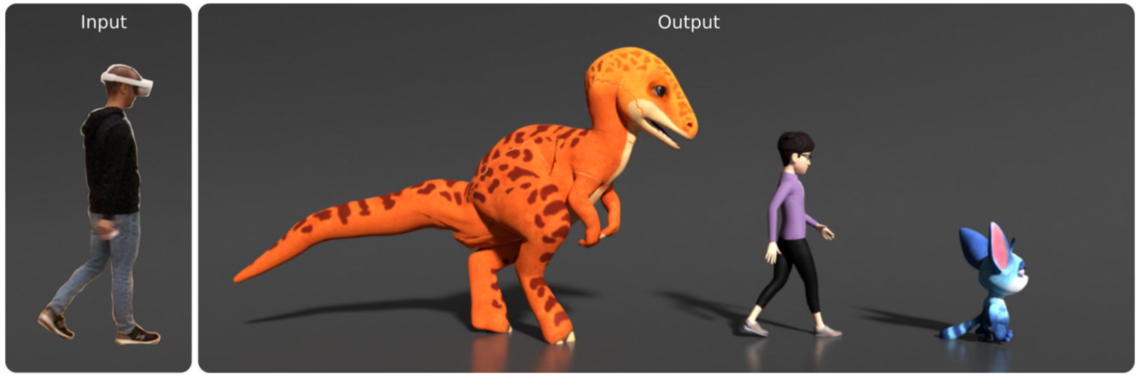
Also works for different avatar characters, e.g., different leg movements emerge for shorter legs

https://www.cs.ubc.ca/~dreda/retargeting.html

Physics-based Motion Retargeting from Sparse Inputs

In 22nd ACM SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation (SCA 2023)

DANIELE REDA, University of British Columbia, Canada
JUNGDAM WON, Seoul National University, South Korea
YUTING YE, Reality Labs Research, Meta, United States
MICHIEL VAN DE PANNE, University of British Columbia, Canada
ALEXANDER WINKLER, Reality Labs Research, Meta, United States



Paper: ArXiv **Video:** Youtube

Avatars are important to create interactive and immersive experiences in virtual worlds. One challenge in animating these characters to mimic a user's motion is that commercial AR/VR products consist only of a headset and controllers, providing very limited sensor data of the user's pose. Another challenge is that an avatar might have a different skeleton structure than a human and the mapping between them is unclear. In this work we address both of these challenges. We introduce a method to retarget motions in real-time from sparse human sensor data to characters of various morphologies. Our method uses reinforcement learning to train a policy to control characters in a physics simulator. We only require human motion capture data for training, without relying on artist-generated animations for each avatar. This allows us to use large motion capture datasets to train general policies that can track unseen users from real and sparse data in real-time. We demonstrate the feasibility of our approach on three characters with different skeleton structure: a dinosaur, a mouse-like creature and a human. We show that the avatar poses often match the user surprisingly well, despite having no sensor information of the lower body available. We discuss and ablate the critical components in our framework, specifically the kinematic retargeting step, the imitation, contact and action reward as well as our asymmetric actor-critic observations. We further explore the robustness of our method in a variety of settings including unbalancing, dancing and sports motions.

PULSE and PHC+: Foundation models for physics-based animation (2023)

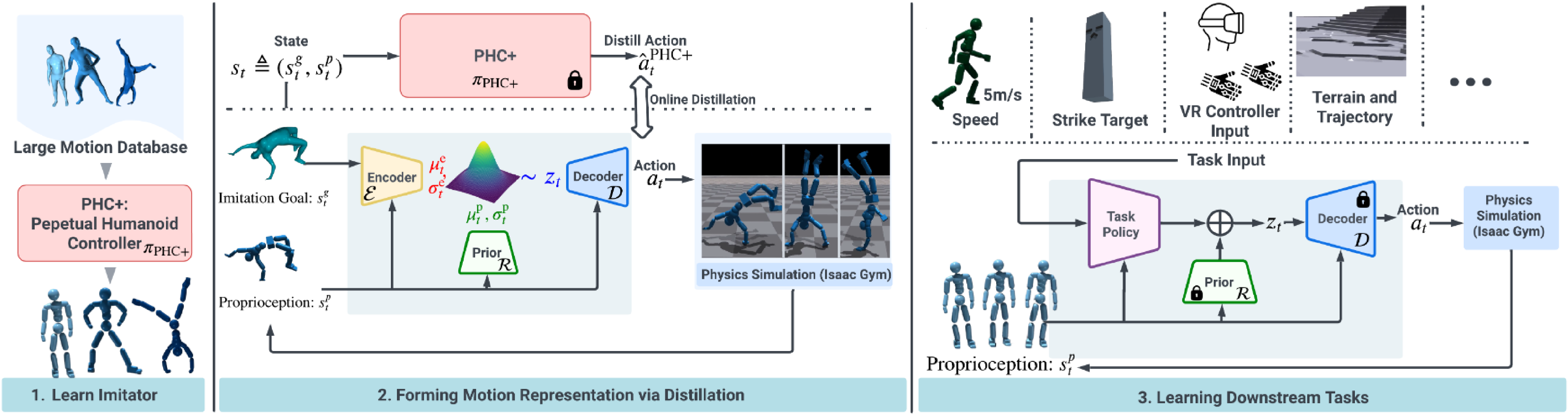


Figure 2: We form our latent space by directly distilling from a pretrained motion imitator that can imitate all of the motion sequences from a large-scale dataset. A variational information bottleneck is used to model the distribution of motor skills conditioned on proprioception. After training the latent space model, the decoder $\mathcal{D}$ and prior $\mathcal{R}$ are frozen and used for downsteam tasks.

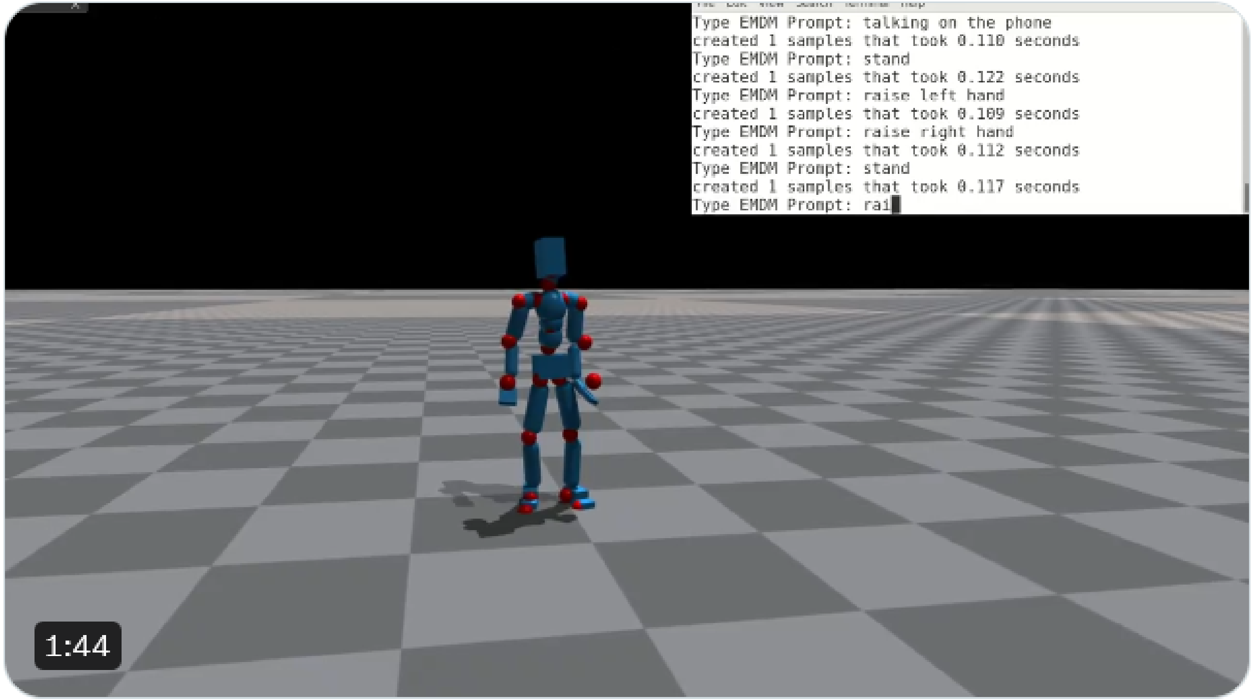
https://www.zhengyiluo.com/PULSE/, https://arxiv.org/abs/2310.04582

Zhengyi (Zen) Luo @zhengyiluo · Dec 29, 2023 · ···

You can now ask your simulated humanoid to perform actions, in REAL-TIME 👇🏻

Powered by the amazing EMDM (@frankzydou, @Alex_wangjingbo, etal) and PHC.

EMDM: frank-zy-dou.github.io/projects/EMDM/...
PHC: github.com/ZhengyiLuo/Per...
Simulation: Isaac Gym



https://x.com/zhengyiluo/status/1740506602615214231?s=20

PhysDiff (2023)

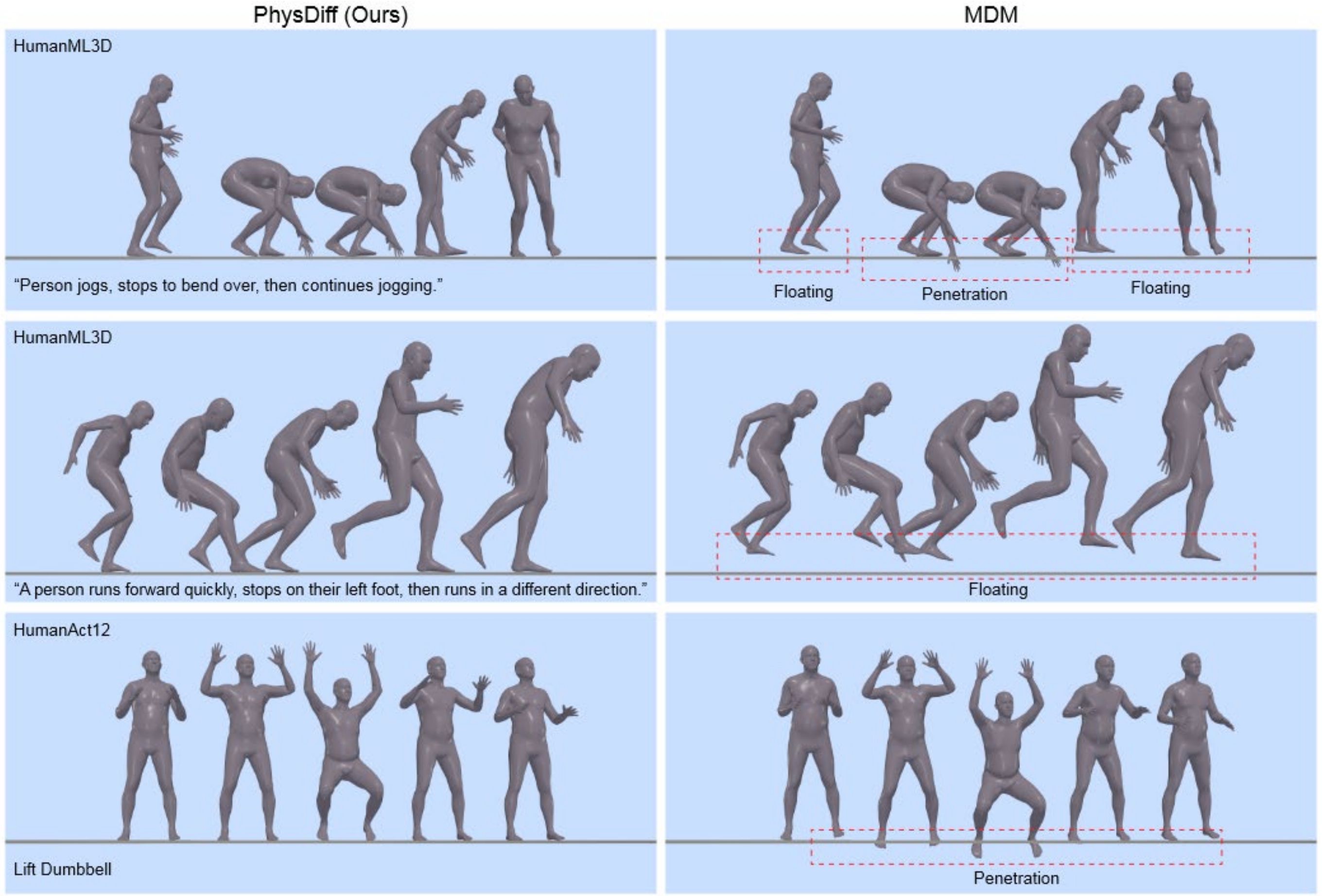


Figure 4. Visual comparison of PhysDiff against the SOTA, MDM [79], on HumanML3D, HumanAct12, and UESTC. PhysDiff reduces physical artifacts such as floating and penetration significantly. Please refer to the project page for more qualitative comparison.

PhysDiff: Physics-Guided Human Motion Diffusion Model

Ye Yuan Jiaming Song Umar Iqbal Arash Vahdat Jan Kautz
NVIDIA
https://nvlabs.github.io/PhysDiff

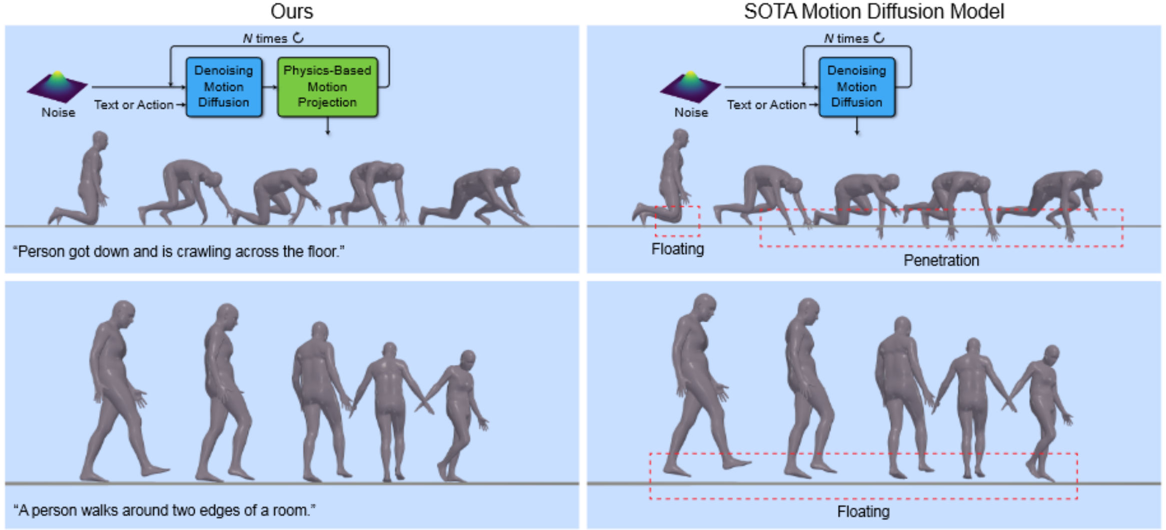


Figure 1. Our PhysDiff model generates physically-plausible motions using a physics-based motion projection in the diffusion process, eliminating artifacts such as floating, ground penetration, and foot sliding, often observed with state-of-the-art models.

Abstract

Denoising diffusion models hold great promise for generating diverse and realistic human motions. However, existing motion diffusion models largely disregard the laws of physics in the diffusion process and often generate physically-implausible motions with pronounced artifacts such as floating, foot sliding, and ground penetration. This seriously impacts the quality of generated motions and limits their real-world application. To address this issue, we present a novel physics-guided motion diffusion model (PhysDiff), which incorporates physical constraints into the diffusion process. Specifically, we propose a physics-based motion projection module that uses motion imitation in a physics simulator to project the denoised motion of a diffusion step to a physically-plausible motion. The projected motion is further used in the next diffusion step to guide the denoising diffusion process. Intuitively, the use of physics in our model iteratively pulls the motion toward a physically-plausible space, which cannot be achieved by simple post-processing. Experiments on large-scale human motion datasets show that our approach achieves state-of-the-art motion quality and improves physical plausibility drastically (>78% for all datasets).

https://nvlabs.github.io/PhysDiff/

1. Introduction

Deep learning-based human motion generation is an important task with numerous applications in animation, gaming, and virtual reality. In common settings such as text-to-motion synthesis, we need to learn a conditional generative model that can capture the multi-modal distribution of human motions. The distribution can be highly complex due to the high variety of human motions and the intricate interaction between human body parts. Denoising diffusion models [75, 25, 76] are a class of generative models that are especially suited for this task due to their strong ability to model complex distributions, which has been demonstrated extensively in the image generation domain [68, 63, 67, 13]. These models have exhibited strong mode coverage often indicated by high test likelihood [77, 33, 81]. They also

Generating game levels

Sokoban GPT

Sokoban levels encoded row-by-row as text strings—no need to build a custom tokenizer and learn new embeddings

https://arxiv.org/abs/2302.05817

Level Generation Through Large Language Models

Graham Todd
gdrtodd@nyu.edu
New York University Tandon
Brooklyn, New York, USA

Sam Earle
se2161@nyu.edu
New York University Tandon
Brooklyn, New York, USA

Muhammad Umair Nasir
umairnasir1@students.wits.ac.za
University of the Witwatersrand
Johannesburg, South Africa

Michael Cerny Green
mike.green@nyu.edu
New York University Tandon
Brooklyn, New York, USA

Julian Togelius
julian@togelius.com
New York University Tandon
Brooklyn, New York, USA

Abstract

Large Language Models (LLMs) are powerful tools, capable of leveraging their training on natural language to write stories, generate code, and answer questions. But can they generate functional video game levels? Game levels, with their complex functional constraints and spatial relationships in more than one dimension, are very different from the kinds of data an LLM typically sees during training. Datasets of game levels are also hard to come by, potentially taxing the abilities of these data-hungry models. We investigate the use of LLMs to generate levels for the game *Sokoban*, finding that LLMs are indeed capable of doing so, and that their performance scales dramatically with dataset size. We also perform preliminary experiments on controlling LLM level generators and discuss promising areas for future work.

Keywords: procedural content generation, sokoban, language models, transformers

ACM Reference Format:
Graham Todd, Sam Earle, Muhammad Umair Nasir, Michael Cerny Green, and Julian Togelius. 2023. Level Generation Through Large Language Models. In *Foundations of Digital Games 2023 (FDG 2023), April 12–14, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3582437.3587211

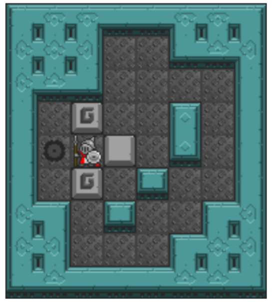


Figure 1. A level for the puzzle game *Sokoban* generated by GPT-3, visualized with the Griddly tileset [2]

1 Introduction

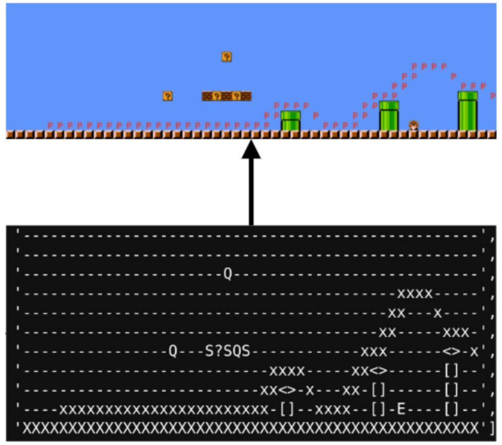
In recent years, attention-based large language models (LLMs) have taken the world by storm, demonstrating surprisingly high performance on a variety of natural language tasks. With the right tuning, LLMs have been shown to generate coherent text in a number of styles, produce working snippets of computer code, and even respond naturalistically to human questions and conversation. While the *architectures* underlying these models have been leveraged for tasks outside the realm of standard text generation, from music [7] to reinforcement learning [4], comparatively less effort has been spent on analyzing the capacity of the LLMs themselves to produce non-lingusitic artifacts while still leveraging their vast amounts of training data. In this paper, we investigate the ability of LLMs to generate video game levels and the extent to which truths about these models taken from natural language processing apply to this new domain. We also conduct preliminary experiments on the capacity to control the levels generated by LLMs using simple data augmentation and prompting.

Despite their impressive performance, there are reasons to doubt that LLMs would be well suited to the task of level generation. The first is representational. For context, the last few years have seen a steady increase in the use of machine learning to generate novel game content, including game levels. This procedural content generation through machine learning (PCGML) has made use of a variety of methods, including cellular automata, Markov models, convolutional neural networks, and generative adversarial networks. While dissimilar in function, these methods are nonetheless unified in that they tend to represent game levels spatially, as

FDG 2023, April 12–14, 2023, Lisbon, Portugal

ACM ISBN 978-1-4503-9855-8/23/04...$15.00
https://doi.org/10.1145/3582437.3587211

MarioGPT

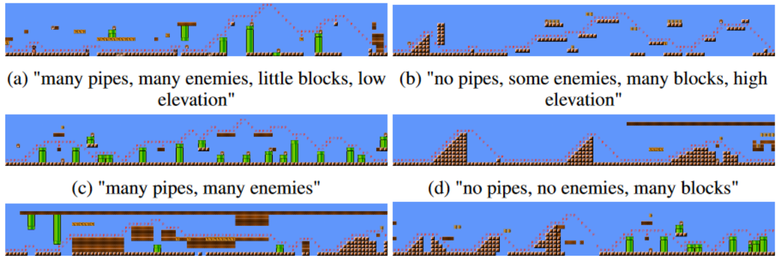


Finetuned a GPT-2 with some extra layers. Seems overly complex, project probably started before better open LLMs were released.

Now, probably better to LoRA-finetune a Llama model without any extra tweaks.

MarioGPT: Open-Ended Text2Level Generation through Large Language Models

Shyam Sudhakaran[1,2], **Miguel González-Duque***[1], **Matthias Freiberger***[1],
Claire Glanois[1], **Elias Najarro**[1], **Sebastian Risi**[1,2]
[1]IT University of Copenhagen, [2]modl.ai, Copenhagen
shyamsnair@protonmail.com, sebr@itu.dk



(a) "many pipes, many enemies, little blocks, low elevation"
(b) "no pipes, some enemies, many blocks, high elevation"
(c) "many pipes, many enemies"
(d) "no pipes, no enemies, many blocks"
(e) Prompt that does not exist in the dataset: "many pipes, no enemies, many blocks"
(f) Failure case: "Many pipes, no enemies, some blocks"

Figure 1: MarioGPT is able to successfully generate levels that follow the text prompt (**a–e**). Failure cases rarely happen: for example in (**f**) the model manages to generate many pipes and some blocks, but it still generates enemies even though it was prompted with "no enemies".

Abstract

Procedural Content Generation (PCG) is a technique to generate complex and diverse environments in an automated way. However, while generating content with PCG methods is often straightforward, generating meaningful content that reflects specific intentions and constraints remains challenging. Furthermore, many PCG algorithms lack the ability to generate content in an open-ended manner. Recently, Large Language Models (LLMs) have shown to be incredibly effective in many diverse domains. These trained LLMs can be fine-tuned, re-using information and accelerating training for new tasks. Here, we introduce MarioGPT, a fine-tuned GPT2 model trained to generate tile-based game levels, in our case Super Mario Bros levels. MarioGPT can not only generate diverse levels, but can be text-prompted for controllable level generation, addressing one of the key challenges of current PCG techniques. As far as we know, MarioGPT is the first text-to-level model and combined with novelty search it enables the generation of diverse levels with varying play-style dynamics (i.e. player paths) and the open-ended discovery of an increasingly diverse range of content. Code available at https://github.com/shyamsn97/mario-gpt.

1 Introduction

https://arxiv.org/abs/2302.05981

Procedural Content Generation (PCG) refers to techniques that can automatically create game content, such as levels, maps, or characters [36]. Some of the benefits of PCG are an increase in the replayability of a game and reduced production costs.

*These authors contributed equally to this work.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Human-in-the loop finetuning reduces the need for data

Iterative finetuning of GPT-3 with only 60 initial examples

https://arxiv.org/abs/2305.18243

Practical PCG Through Large Language Models

[α]Muhammad U Nasir and [β]Julian Togelius
[α]University of the Witwatersrand, South Africa, umairnasir1@students.wits.ac.za
[β]New York University, USA

***Abstract*—Large Language Models (LLMs) have proven to be useful tools in various domains outside of the field of their inception, which was natural language processing. In this study, we provide practical directions on how to use LLMs to generate 2D-game rooms for an under-development game, named `Metavoidal`. Our technique can harness the power of `GPT-3` by Human-in-the-loop fine-tuning which allows our method to create 37% Playable-Novel levels from as scarce data as only 60 hand-designed rooms under a scenario of the non-trivial game, with respect to (Procedural Content Generation) PCG, that has a good amount of local and global constraints.**

***Index Terms*—Procedural Content Generation, Large Language Models**

I. Introduction

There are many ways of generating game levels. While most games featuring online PCG that are actually shipped rely on domain-specific heuristic solutions, methods explored by experimenters include evolutionary computation, constraint satisfaction, grammar expansion, and fractals [1], [2]. Rogue-like games in particular often feature relatively ambitious PCG methods [3]. Over the last decade, machine learning has turned out to be fruitfully applicable to essentially everything under the sun. This includes level generation. Researchers have explored the ways machine learning in general can be applied to generating levels and other types of game content [4], as well as deep learning in particular [5]. PCG itself holds utmost importance in many important research fields, like Open-ended Learning [6] or continual learning [7].

It's 2023, and the new new thing that can be applied to everything under the sun is Large Language Models (LLMs), such as image generation [8] and neural architecture search [9]. While originally developed for natural language processing, LLMs have proven effective for anything that can be expressed as sequences of tokens, including images. The versatility of LLMs go beyond what would normally be considered text completion, as they are capable of performing many tasks that would seem to require cognitive efforts from humans. Could LLMs also be useful for generating game content? Game levels, like everything else that passes through a computer, are after all just strings.

Two recent studies examine this. In one of them, GPT-2 and GPT-3 were finetuned to generate Sokoban levels. The generated levels were good and novel but, particularly for GPT-2, the dataset requirements were excessive [10]. Another study showed that special-purpose LLM architecture produced good levels for the classic platformer Super Mario Bros [11].

In this paper we explore the possibility of using LLMs to generate levels for a game under active development, where only a limited number of levels are available, forcing us to find a data-efficient method. Furthermore, these levels are relatively large and have a nontrivial number of constraints. Our approach is to encode the constraints into the prompt and fine-tune GPT-3. To efficiently use the limited data available without overfitting we use several types of data augmentation as well as a form of bootstrapping, where novel high-quality levels are added back into the dataset.

II. Metavoidal and Room Generation Setup

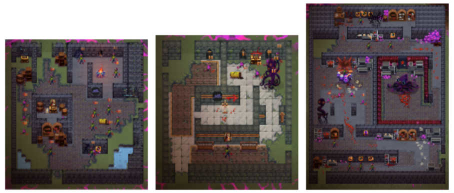


Fig. 1: Levels of different sizes created by the developers with all assets on them.

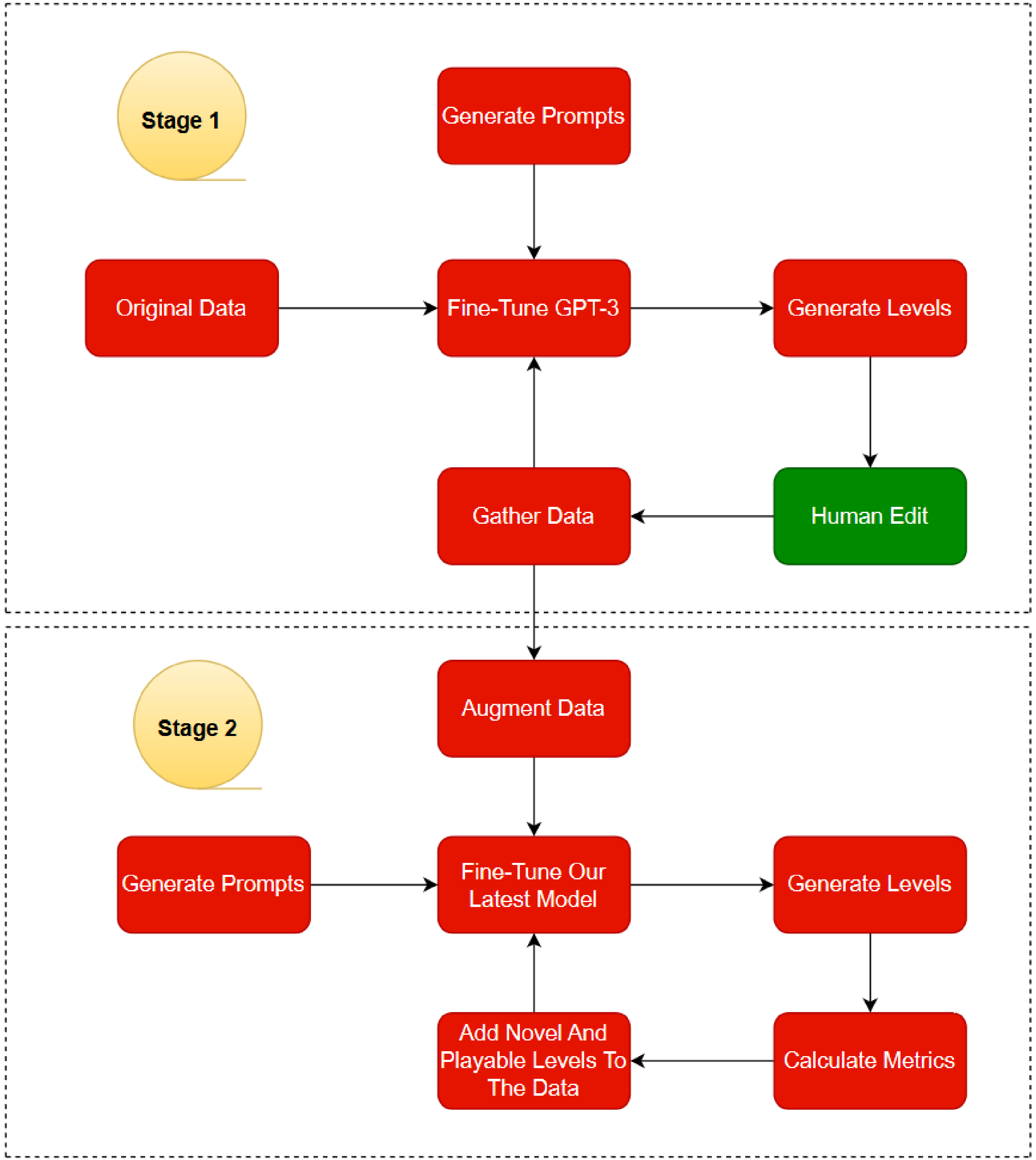
A. Gameplay

Metavoidal[1] is a roguelite brawler game, being developed by *Yellow Lab Games*[2] that features a metal band trying to hire new drummers. The metal band turns out to be full of eldritch monsters sacrificing drummers to gain more power. You play as a drummer in a church where the auditions are happening. You are trying to escape as you are a bad drummer and hordes of monsters are trying to sacrifice you. You will have drumsticks as your weapon. You can find power-ups like music disks to fight enemies. Your goal is to escape the church.

The layout of the game includes 3 levels. Each level with many rooms connected to hallways. There are some tunnel-like connections. Rooms are essentially the main areas where assets are discovered to progress in the game. Figure 1 shows rooms in the game developed by a 2D game artist. There are many tiles to be considered while generating. There are three types of tiles that make patterns: wood, marble and moss. There are two types of walls: marble and moss marble wall. There is

[1]https://yellowlabgames.itch.io/metavoidal
[2]https://yellowlab.games/

979-8-3503-2277-4/23/$31.00 ©2023 IEEE


Stage 1
Generate Prompts
Original Data
Fine-Tune GPT-3
Generate Levels
Gather Data
Human Edit
Stage 2
Augment Data
Generate Prompts
Fine-Tune Our Latest Model
Generate Levels
Add Novel And Playable Levels To The Data
Calculate Metrics

Generating 3D models

Problems

- LLMs: 3D geometry can be represented as text, e.g., XML
- Problem: 3D model data is complex, including both geometry and materials.
  - Not enough training data that's easily available
  - Hard to find a single data representation that works
  - Algorithms often use NeRFs or Gaussian Splatting as an intermediate representation
- Easy to generate geometry or materials. Hard to generate a combination of both.
- Current systems are either low quality or focus on specific types of models. No high-quality and general generators exist for game-ready assets.

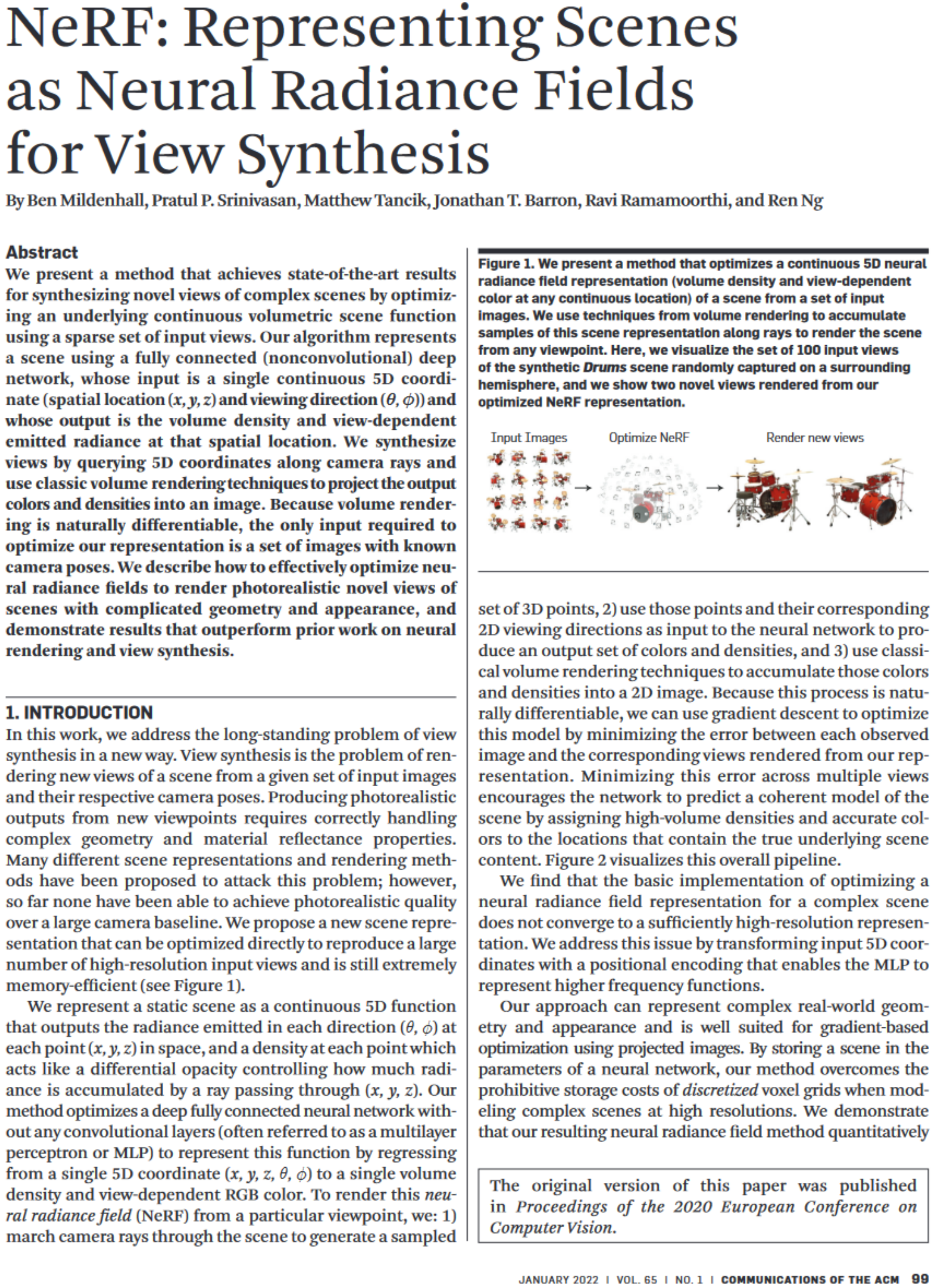
NeRFs (2020)

A neural network can learn compact representation of 3D shapes and materials from reference images

Radiance = *MLP*(x,y,z,viewing direction)

Volumetric rendering techniques can be used to synthesize novel views

https://dl.acm.org/doi/pdf/10.1145/3503250



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis

By Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng

Abstract

We present a method that achieves state-of-the-art results for synthesizing novel views of complex scenes by optimizing an underlying continuous volumetric scene function using a sparse set of input views. Our algorithm represents a scene using a fully connected (nonconvolutional) deep network, whose input is a single continuous 5D coordinate (spatial location (x, y, z) and viewing direction (θ, ϕ)) and whose output is the volume density and view-dependent emitted radiance at that spatial location. We synthesize views by querying 5D coordinates along camera rays and use classic volume rendering techniques to project the output colors and densities into an image. Because volume rendering is naturally differentiable, the only input required to optimize our representation is a set of images with known camera poses. We describe how to effectively optimize neural radiance fields to render photorealistic novel views of scenes with complicated geometry and appearance, and demonstrate results that outperform prior work on neural rendering and view synthesis.

1. INTRODUCTION

In this work, we address the long-standing problem of view synthesis in a new way. View synthesis is the problem of rendering new views of a scene from a given set of input images and their respective camera poses. Producing photorealistic outputs from new viewpoints requires correctly handling complex geometry and material reflectance properties. Many different scene representations and rendering methods have been proposed to attack this problem; however, so far none have been able to achieve photorealistic quality over a large camera baseline. We propose a new scene representation that can be optimized directly to reproduce a large number of high-resolution input views and is still extremely memory-efficient (see Figure 1).

We represent a static scene as a continuous 5D function that outputs the radiance emitted in each direction (θ, ϕ) at each point (x, y, z) in space, and a density at each point which acts like a differential opacity controlling how much radiance is accumulated by a ray passing through (x, y, z). Our method optimizes a deep fully connected neural network without any convolutional layers (often referred to as a multilayer perceptron or MLP) to represent this function by regressing from a single 5D coordinate (x, y, z, θ, ϕ) to a single volume density and view-dependent RGB color. To render this *neural radiance field* (NeRF) from a particular viewpoint, we: 1) march camera rays through the scene to generate a sampled set of 3D points, 2) use those points and their corresponding 2D viewing directions as input to the neural network to produce an output set of colors and densities, and 3) use classical volume rendering techniques to accumulate those colors and densities into a 2D image. Because this process is naturally differentiable, we can use gradient descent to optimize this model by minimizing the error between each observed image and the corresponding views rendered from our representation. Minimizing this error across multiple views encourages the network to predict a coherent model of the scene by assigning high-volume densities and accurate colors to the locations that contain the true underlying scene content. Figure 2 visualizes this overall pipeline.

We find that the basic implementation of optimizing a neural radiance field representation for a complex scene does not converge to a sufficiently high-resolution representation. We address this issue by transforming input 5D coordinates with a positional encoding that enables the MLP to represent higher frequency functions.

Our approach can represent complex real-world geometry and appearance and is well suited for gradient-based optimization using projected images. By storing a scene in the parameters of a neural network, our method overcomes the prohibitive storage costs of *discretized* voxel grids when modeling complex scenes at high resolutions. We demonstrate that our resulting neural radiance field method quantitatively

Figure 1. We present a method that optimizes a continuous 5D neural radiance field representation (volume density and view-dependent color at any continuous location) of a scene from a set of input images. We use techniques from volume rendering to accumulate samples of this scene representation along rays to render the scene from any viewpoint. Here, we visualize the set of 100 input views of the synthetic *Drums* scene randomly captured on a surrounding hemisphere, and we show two novel views rendered from our optimized NeRF representation.

Input Images → Optimize NeRF → Render new views

The original version of this paper was published in *Proceedings of the 2020 European Conference on Computer Vision.*

Gaussian Splatting takes over (2023)

No neural network needed.

Shapes rendered as point clouds of soft round or elongated particles

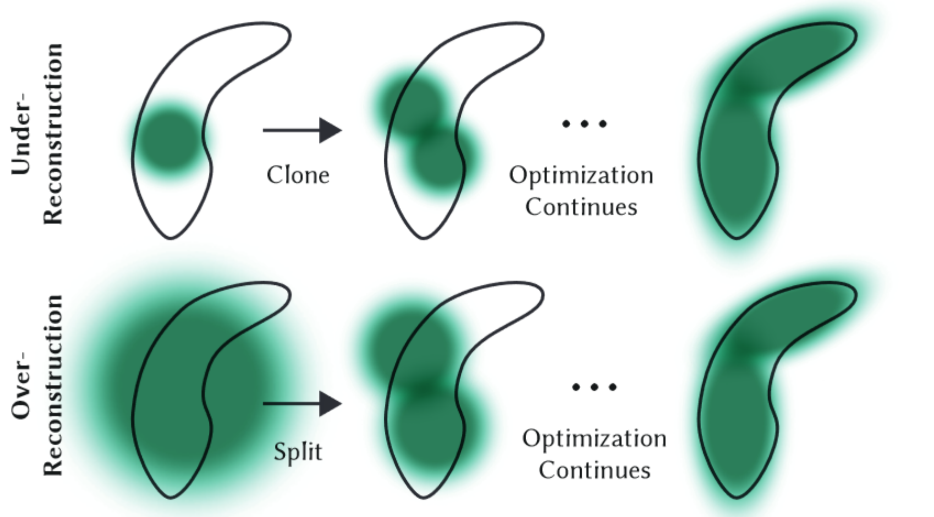


Fig. 4. Our adaptive Gaussian densification scheme. *Top row (under-reconstruction)*: When small-scale geometry (black outline) is insufficiently covered, we clone the respective Gaussian. *Bottom row (over-reconstruction)*: If small-scale geometry is represented by one large splat, we split it in two.

3D Gaussian Splatting for Real-Time Radiance Field Rendering

BERNHARD KERBL*, Inria, Université Côte d'Azur, France
GEORGIOS KOPANAS*, Inria, Université Côte d'Azur, France
THOMAS LEIMKÜHLER, Max-Planck-Institut für Informatik, Germany
GEORGE DRETTAKIS, Inria, Université Côte d'Azur, France

https://dl.acm.org/doi/pdf/10.1145/3592433

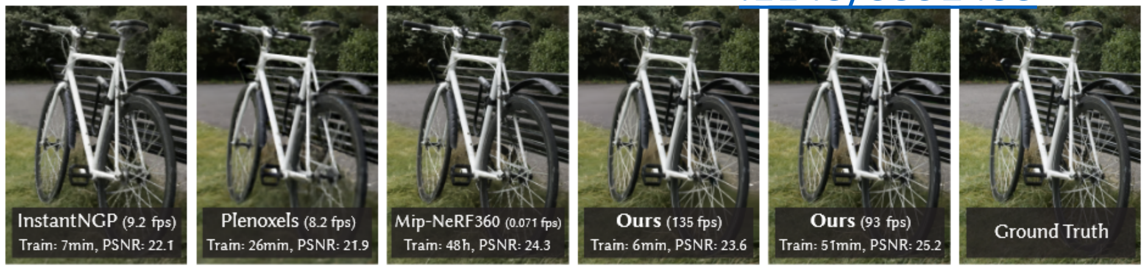


Fig. 1. Our method achieves real-time rendering of radiance fields with quality that equals the previous method with the best quality [Barron et al. 2022], while only requiring optimization times competitive with the fastest previous methods [Fridovich-Keil and Yu et al. 2022; Müller et al. 2022]. Key to this performance is a novel 3D Gaussian scene representation coupled with a real-time differentiable renderer, which offers significant speedup to both scene optimization and novel view synthesis. Note that for comparable training times to InstantNGP [Müller et al. 2022], we achieve similar quality to theirs; while this is the maximum quality they reach, by training for 51min we achieve state-of-the-art quality, even slightly better than Mip-NeRF360 [Barron et al. 2022].

Radiance Field methods have recently revolutionized novel-view synthesis of scenes captured with multiple photos or videos. However, achieving high visual quality still requires neural networks that are costly to train and render, while recent faster methods inevitably trade off speed for quality. For unbounded and complete scenes (rather than isolated objects) and 1080p resolution rendering, no current method can achieve real-time display rates. We introduce three key elements that allow us to achieve state-of-the-art visual quality while maintaining competitive training times and importantly allow high-quality real-time (≥ 30 fps) novel-view synthesis at 1080p resolution. First, starting from sparse points produced during camera calibration, we represent the scene with 3D Gaussians that preserve desirable properties of continuous volumetric radiance fields for scene optimization while avoiding unnecessary computation in empty space; Second, we perform interleaved optimization/density control of the 3D Gaussians, notably optimizing anisotropic covariance to achieve an accurate representation of the scene; Third, we develop a fast visibility-aware rendering algorithm that supports anisotropic splatting and both accelerates training and allows real-time rendering. We demonstrate state-of-the-art visual quality and real-time rendering on several established datasets.

CCS Concepts: • **Computing methodologies** → **Rendering**; **Point-based models**; **Rasterization**; **Machine learning approaches**.

*Both authors contributed equally to the paper.

Authors' addresses: Bernhard Kerbl, bernhard.kerbl@inria.fr, Inria, Université Côte d'Azur, France; Georgios Kopanas, georgios.kopanas@inria.fr, Inria, Université Côte d'Azur, France; Thomas Leimkühler, thomas.leimkuehler@mpi-inf.mpg.de, Max-Planck-Institut für Informatik, Germany; George Drettakis, george.drettakis@inria.fr, Inria, Université Côte d'Azur, France.

0730-0301/2023/8-ART139 $15.00
https://doi.org/10.1145/3592433

Additional Key Words and Phrases: novel view synthesis, radiance fields, 3D gaussians, real-time rendering

ACM Reference Format:
Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.* 42, 4, Article 139 (August 2023), 14 pages. https://doi.org/10.1145/3592433

1 INTRODUCTION

Meshes and points are the most common 3D scene representations because they are explicit and are a good fit for fast GPU/CUDA-based rasterization. In contrast, recent Neural Radiance Field (NeRF) methods build on continuous scene representations, typically optimizing a Multi-Layer Perceptron (MLP) using volumetric ray-marching for novel-view synthesis of captured scenes. Similarly, the most efficient radiance field solutions to date build on continuous representations by interpolating values stored in, e.g., voxel [Fridovich-Keil and Yu et al. 2022] or hash [Müller et al. 2022] grids or points [Xu et al. 2022]. While the continuous nature of these methods helps optimization, the stochastic sampling required for rendering is costly and can result in noise. We introduce a new approach that combines the best of both worlds: our 3D Gaussian representation allows optimization with state-of-the-art (SOTA) visual quality and competitive training times, while our tile-based splatting solution ensures real-time rendering at SOTA quality for 1080p resolution on several previously published datasets [Barron et al. 2022; Hedman et al. 2018; Knapitsch et al. 2017] (see Fig. 1).

Our goal is to allow real-time rendering for scenes captured with multiple photos, and create the representations with optimization times as fast as the most efficient previous methods for typical real scenes. Recent methods achieve fast training [Fridovich-Keil

Better datasets are emerging

10M+ 3D objects with text captions

Every Objaverse object is Blender-compatible

https://objaverse.allenai.org/

Objaverse: A Universe of Annotated 3D Objects

Matt Deitke[†ψ], Dustin Schwenk[†], Jordi Salvador[†], Luca Weihs[†], Oscar Michel[†]
Eli VanderBilt[†], Ludwig Schmidt[ψ], Kiana Ehsani[†], Aniruddha Kembhavi[†ψ], Ali Farhadi[ψ]
[†]PRIOR @ Allen Institute for AI, [ψ]University of Washington, Seattle
objaverse.allenai.org

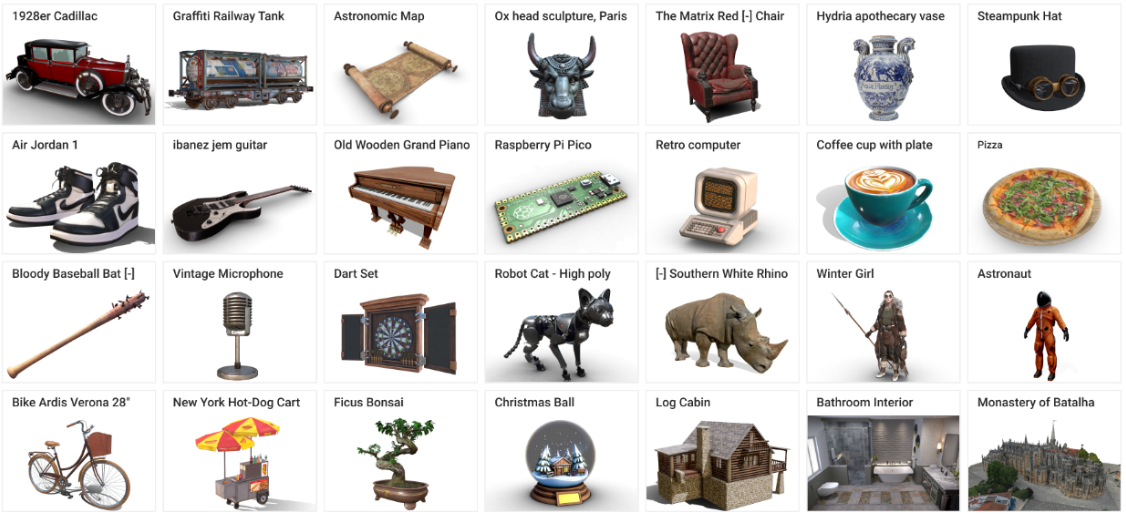


Figure 1. Example instances from our large-scale 3D asset dataset OBJAVERSE. OBJAVERSE 3D assets are semantically diverse, high-quality, and paired with natural-language descriptions.

Abstract

Massive data corpora like WebText, Wikipedia, Conceptual Captions, WebImageText, and LAION have propelled recent dramatic progress in AI. Large neural models trained on such datasets produce impressive results and top many of today's benchmarks. A notable omission within this family of large-scale datasets is 3D data. Despite considerable interest and potential applications in 3D vision, datasets of high-fidelity 3D models continue to be mid-sized with limited diversity of object categories. Addressing this gap, we present Objaverse 1.0, a large dataset of objects with 800K+ (and growing) 3D models with descriptive captions, tags, and animations. Objaverse improves upon present day 3D repositories in terms of scale, number of categories, and in the visual diversity of instances within a category. We demonstrate the large potential of Objaverse via four diverse applications: training generative 3D models, improving tail category segmentation on the LVIS benchmark, training open-vocabulary object-navigation models for Embodied AI, and creating a new benchmark for robustness analysis of vision models. Objaverse can open new directions for research and enable new applications across the field of AI.

1. Introduction

Massive datasets have enabled and driven rapid progress in AI. Language corpora on the web led to large language models like GPT-3 [4]; paired image and text datasets like Conceptual Captions [68] led to vision-and-language pretrained models like VilBERT [45]; YouTube video datasets led to video capable models like Merlot-Reserve [87]; and massive multimodal datasets like WebImageText [70] and LAION [66, 67] led to models like CLIP [60] and StableDiffusion [64]. These leaps in dataset scale and diversity were triggered by moving from manually curated datasets to harnessing the power of the web and its creative content.

In contrast to the datasets described above, the size of

Correspondence to <mattd@allenai.org>.

MeshGPT

Finetunes GPT-2 for triangle-by-triangle mesh generation

https://arxiv.org/abs/2311.15475

MeshGPT: Generating Triangle Meshes with Decoder-Only Transformers

Yawar Siddiqui[1] Antonio Alliegro[2] Alexey Artemov[1]
Tatiana Tommasi[2] Daniele Sirigatti[3] Vladislav Rosov[3] Angela Dai[1] Matthias Nießner[1]
Technical University of Munich[1] Politecnico di Torino[2] AUDI AG[3]

Figure 1. Our method creates triangle meshes by autoregressively sampling from a transformer model that has been trained to produce tokens from a learned geometric vocabulary. These tokens can then be decoded into the faces of a triangle mesh. Our method generates clean, coherent, and compact meshes, characterized by sharp edges and high fidelity.

Abstract

We introduce MeshGPT, a new approach for generating triangle meshes that reflects the compactness typical of artist-created meshes, in contrast to dense triangle meshes extracted by iso-surfacing methods from neural fields. Inspired by recent advances in powerful large language models, we adopt a sequence-based approach to autoregressively generate triangle meshes as sequences of triangles. We first learn a vocabulary of latent quantized embeddings, using graph convolutions, which inform these embeddings of the local mesh geometry and topology. These embeddings are sequenced and decoded into triangles by a decoder, ensuring that they can effectively reconstruct the mesh. A transformer is then trained on this learned vocabulary to predict the index of the next embedding given previous embeddings. Once trained, our model can be autoregressively sampled to generate new triangle meshes, directly generating compact meshes with sharp edges, more closely imitating the efficient triangulation patterns of human-crafted meshes. MeshGPT demonstrates a notable improvement over state of the art mesh generation methods, with a 9% increase in shape coverage and a 30-point enhancement in FID scores across various categories.

1. Introduction

Triangle meshes are the main representation for 3D geometry in computer graphics. They are the predominant representation for 3D assets used in video games, movies, and virtual reality interfaces. Compared to alternative 3D shape representations such as point clouds or voxels, meshes provide a more coherent surface representation; they are more controllable, easier to manipulate, more compact, and fit directly into modern rendering pipelines, attaining high visual quality with far fewer primitives. In this paper, we tackle the task of automated generation of triangle meshes, streamlining the process of crafting 3D assets.

Recently, 3D vision research has seen great interest in generative 3D models using representations such as voxels [3, 62], point clouds [37, 67, 68], and neural fields [14, 19, 31, 35, 41]. However, these representations must then be converted into meshes through a post-process for use in downstream applications, for instance by iso-surfacing with Marching Cubes [36]. Unfortunately, this results in dense, over-tessellated meshes that often exhibit oversmoothing and bumpy artifacts from the iso-surfacing, as shown in Figure 2. In contrast, artist-modeled 3D meshes are compact in representation, while maintaining sharp details with much fewer triangles.

Thus, we propose MeshGPT[1] to generate a mesh representation directly, as a set of triangles. Inspired by powerful recent advances in generative models for language, we adopt a direct sequence generation approach to synthesize triangle meshes as sequences of triangles. Following text generation paradigms, we first learn a vocabulary of triangles. Triangles are encoded into latent quantized embeddings through an encoder. To encourage learned trian-

[1] nihalsid.github.io/mesh-gpt

DreamFusion (2022)

https://dreamfusion3d.github.io/

Using a pretrained Diffusion model to guide NeRF optimization.

Very slow.

DreamFusion: Text-to-3D using 2D Diffusion

Ben Poole
Google Research

Ajay Jain
UC Berkeley

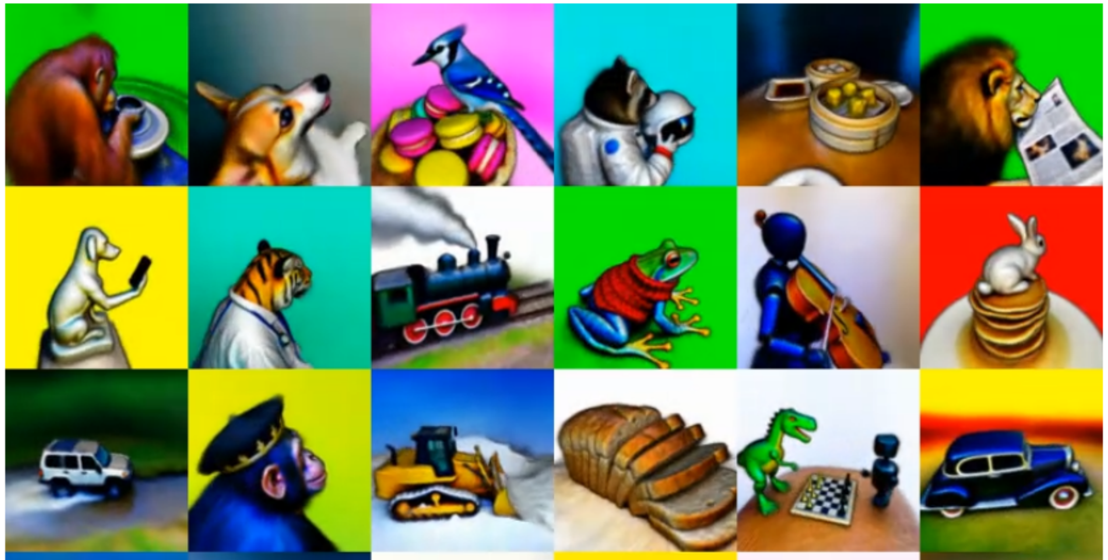
Jonathan T. Barron
Google Research

Ben Mildenhall
Google Research

Paper | Project | Gallery

Abstract

Recent breakthroughs in text-to-image synthesis have been driven by diffusion models trained on billions of image-text pairs. Adapting this approach to 3D synthesis would require large-scale datasets of labeled 3D assets and efficient architectures for denoising 3D data, neither of which currently exist. In this work, we circumvent these limitations by using a pretrained 2D text-to-image diffusion model to perform text-to-3D synthesis. We introduce a loss based on probability density distillation that enables the use of a 2D diffusion model as a prior for optimization of a parametric image generator. Using this loss in a DeepDream-like procedure, we optimize a randomly-initialized 3D model (a Neural Radiance Field, or NeRF) via gradient descent such that its 2D renderings from random angles achieve a low loss. The resulting 3D model of the given text can be viewed from any angle, relit by arbitrary illumination, or composited into any 3D environment. Our approach requires no 3D training data and no modifications to the image diffusion model, demonstrating the effectiveness of pretrained image diffusion models as priors.



ProlificDreamer (2023)

https://ml.cs.tsinghua.edu.cn/prolificdreamer/

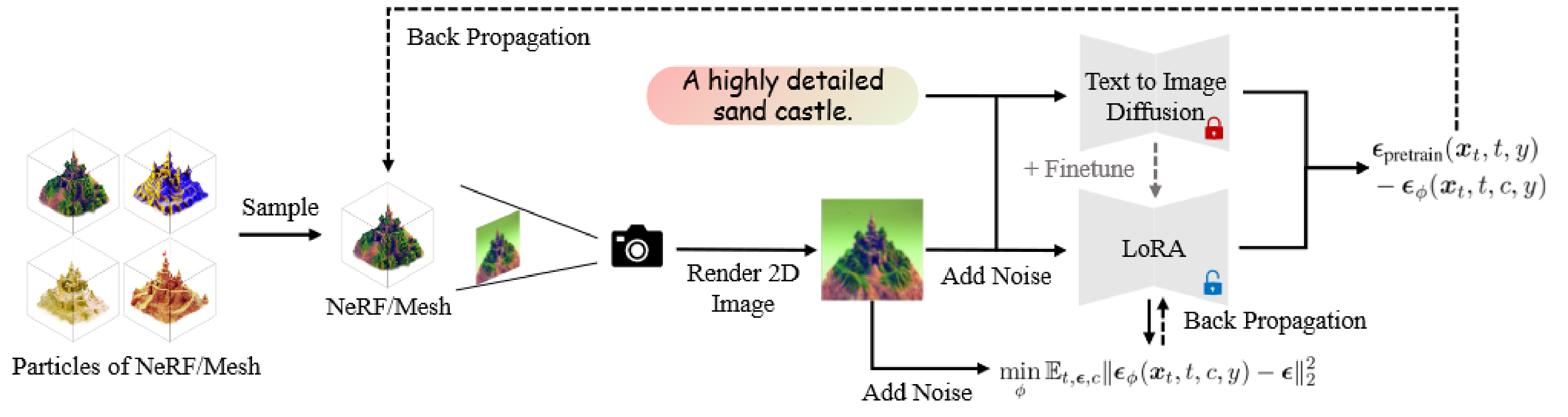
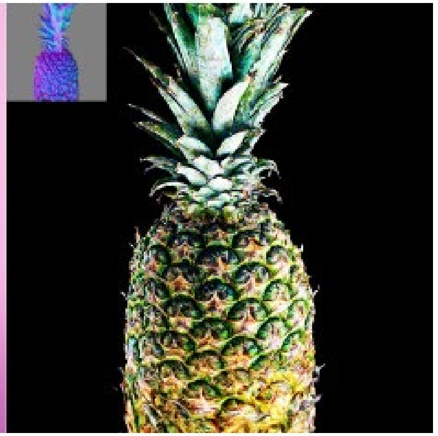


Figure 2: Overview of VSD. The 3D representation is differentiably rendered at a random pose c. The rendered image is sent to the pretrained diffusion and the score of the variational distribution (estimated by LoRA) to compute the gradient of VSD. LoRA is also updated on the rendered image.

Michelangelo style statue of dog reading news on a cellphone.



A pineapple.

A model of a house in Tudor style.

A tarantula, highly detailed.

LucidDreamer (2023)

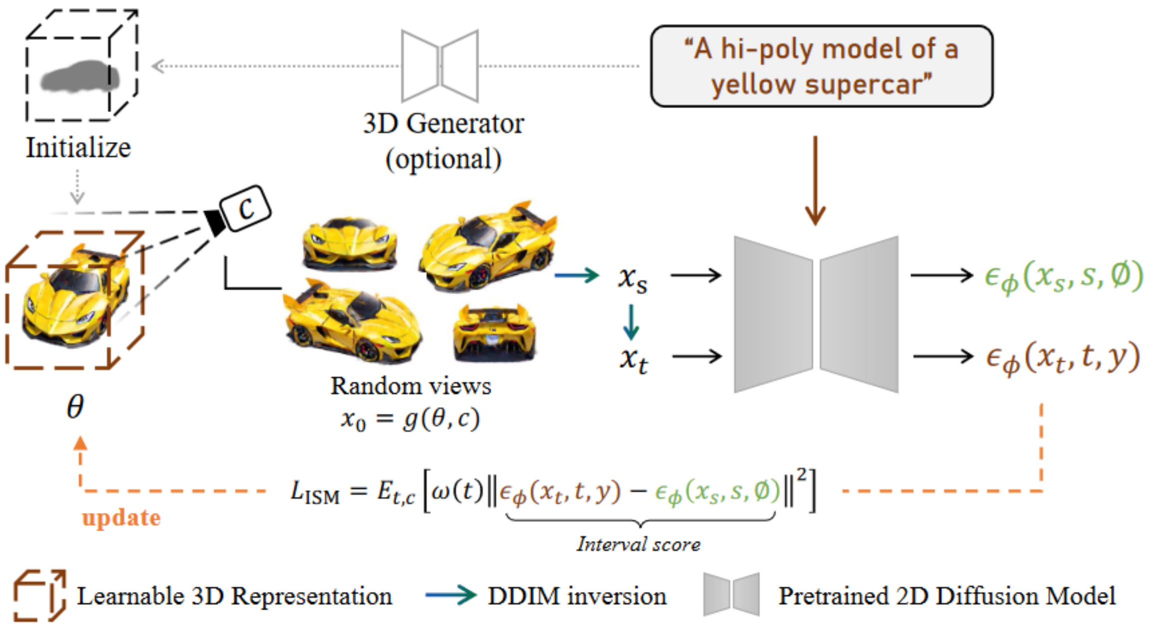


Figure 3. **An overview of LucidDreamer.** In our paper, we first initialize the 3D representation (i.e. Gaussian Splatting [20]) θ via the pretrained text-to-3D generator [33] with prompt y. Incorporate with pretrained 2D DDPM, we disturb random views $\boldsymbol{x}_0 = \boldsymbol{g}(\theta, c)$ to unconditional noisy latent trajectories $\{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_s, \boldsymbol{x}_t\}$ via DDIM inversion [42]. Then, we update θ with the *interval score*. Please refer to Sec. 3.2 for details.

LucidDreamer: Towards High-Fidelity Text-to-3D Generation via Interval Score Matching

Yixun Liang*[1] Xin Yang*[1,2] Jiantao Lin[1] Haodong Li[1] Xiaogang Xu[3,4] Yingcong Chen**[1,2]
[1] HKUST (GZ) [2] HKUST [3] Zhejiang Lab [4] Zhejiang University
yliang982@connect.hkust-gz.edu.cn xin.yang@connect.ust.hk jlin695@hkust-gz.edu.cn
hli736@connect.hkust-gz.edu.cn xgxu@zhejianglab.com yingcongchen@ust.hk

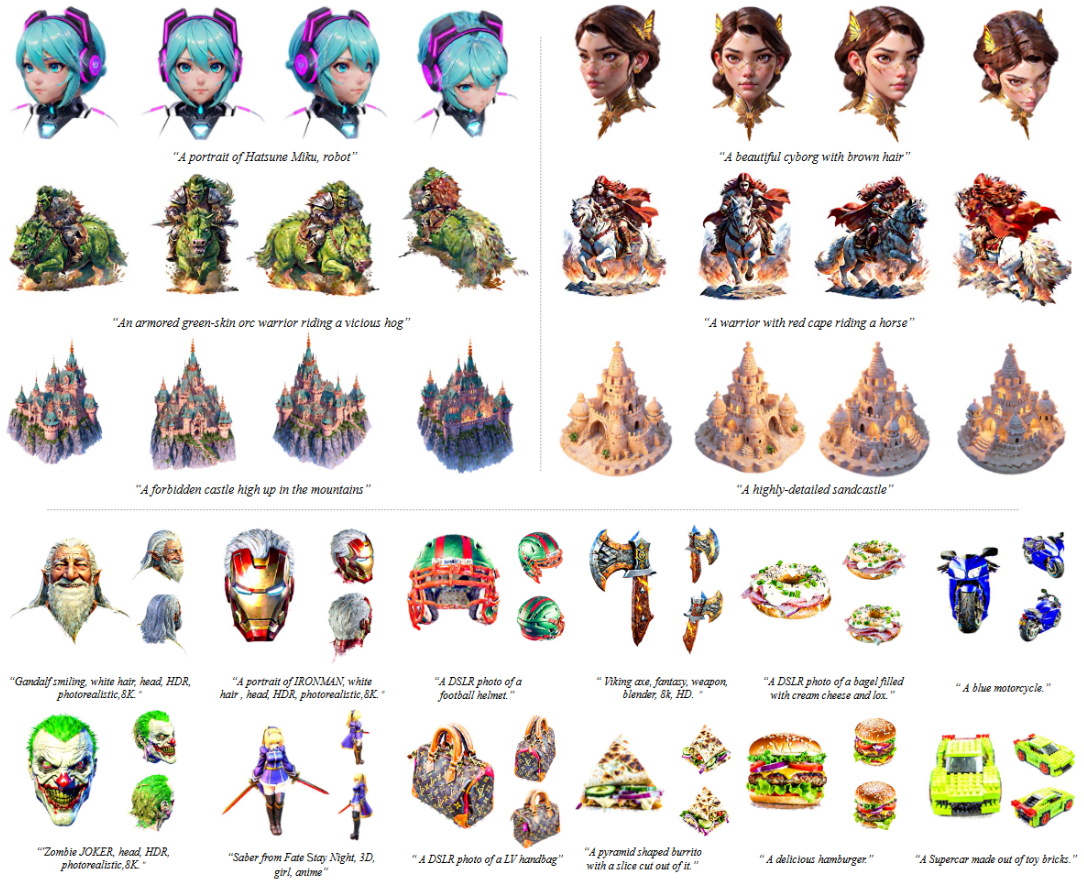


Figure 1. **Examples of text-to-3D content creations with our framework.** We present a text-to-3D generation framework, named the *LucidDreamer*, to distill high-fidelity textures and shapes from pretrained 2D diffusion models (detailed shows on Sec. 4) with a novel **Interval Score Matching** objective and an *Advanced 3D distillation pipeline*. Together, we achieve superior 3D generation results with photorealistic quality in a short training time. Please zoom in for details.

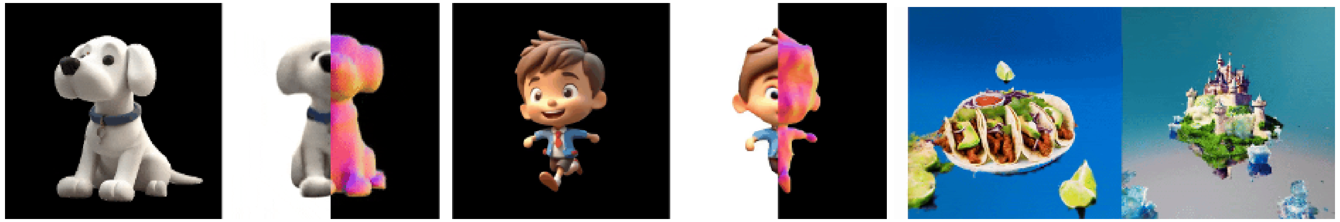
** Corresponding author.
*The first two authors contributed equally to this work.
* *Conceptualization*: **Yixun Liang**: 60%, **Xin Yang**: 40%,
Methodology: **Xin Yang**: 60%, **Yixun Liang**: 40%.

threestudio

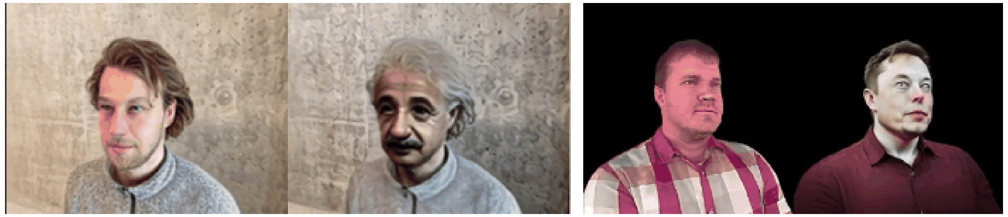
https://github.com/threestudio-project/threestudio

threestudio is a unified framework for 3D content creation from text prompts, single images, and few-shot images, by lifting 2D text-to-image generation models.

ProlificDreamer DreamFusion Magic3D SJC LatentNeRF Fantasia3D TextMesh



Zero-1–to-3 Magic123 HiFA



Instruct-NeRF2NeRF Control4D (static)

👆 Results obtained from methods implemented by threestudio 👆

| ProlificDreamer | DreamFusion | Magic3D | SJC | Latent-NeRF | Fantasia3D | TextMesh |

| Zero-1-to-3 | Magic123 | HiFA |

| InstructNeRF2NeRF | Control4D |

Generating rigged and skinned characters

Output: SMPL deformation parameters and textures.

https://dancasas.github.io/projects/SMPLitex/index.html

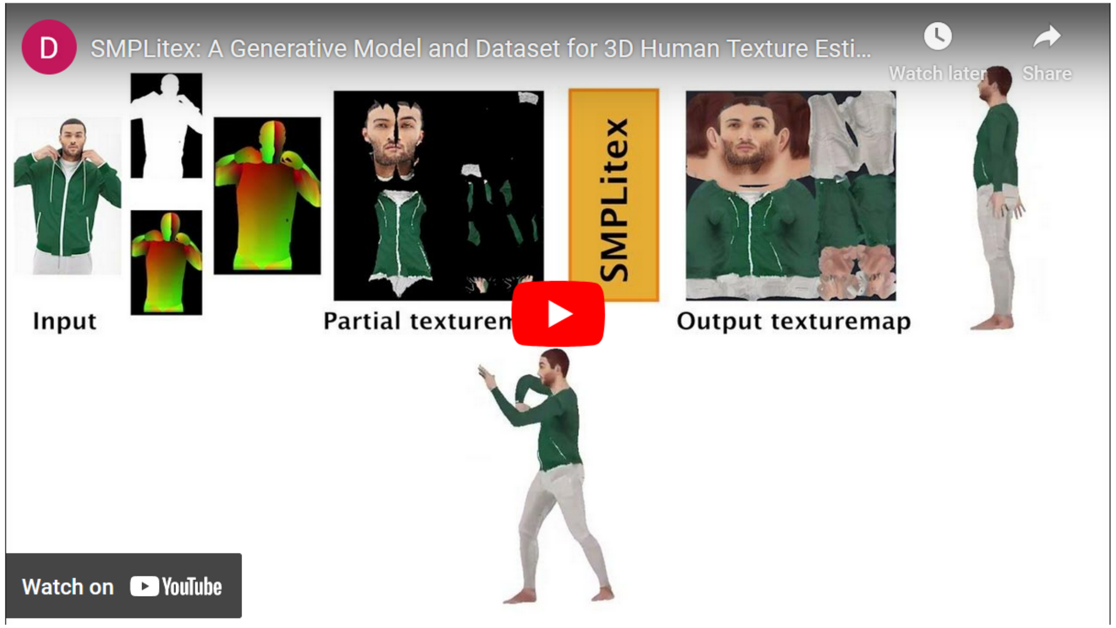
How to use SMPL in Unity and Blender:
https://files.is.tue.mpg.de/nmahmood/smpl_website/How-to_SMPLinUnity.pdf

https://ps.is.mpg.de/code/smpl-x-for-blender-and-unity

SMPLitex: A Generative Model and Dataset for 3D Human Texture Estimation from Single Image

Dan Casas and Marc Comino-Trinidad

British Machine Vision Conference (BMVC), 2023



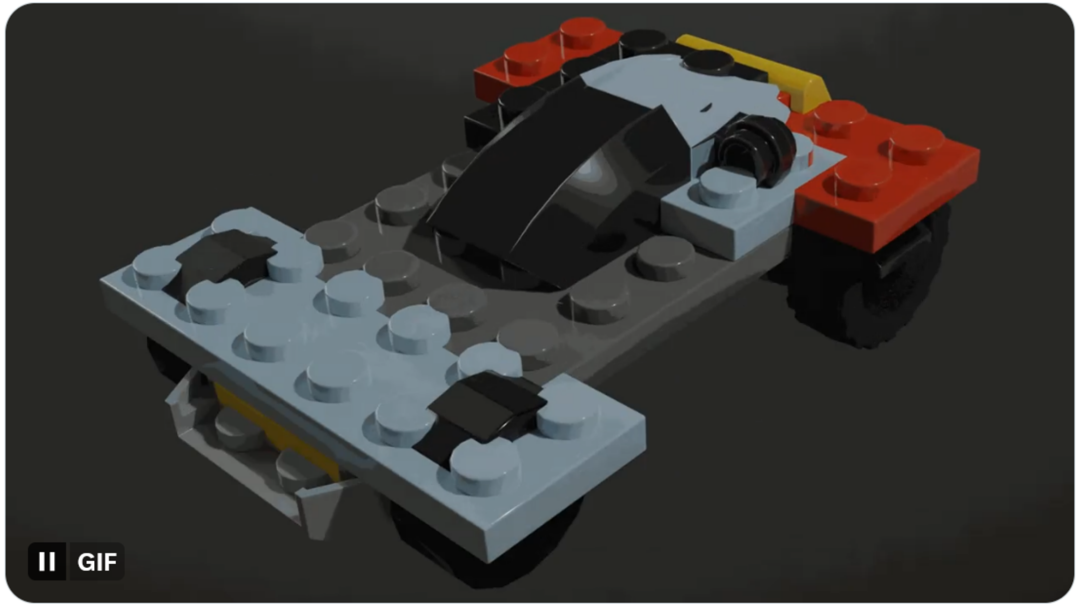
Abstract

We propose SMPLitex, a method for estimating and manipulating the complete 3D appearance of humans captured from a single image. SMPLitex builds upon the recently proposed generative models for 2D images, and extends their use to the 3D domain through pixel-to-surface correspondences computed on the input image. To this end, we first train a generative model for complete 3D human appearance, and then fit it into the input image by conditioning the generative model to the visible parts of subject. Furthermore, we propose a new dataset of high-quality human textures built by sampling SMPLitex conditioned on subject descriptions and images. We quantitatively and qualitatively evaluate our method in 3 publicly available datasets, demonstrating that SMPLitex significantly outperforms existing methods for human texture estimation while allowing for a wider variety of tasks such as editing, synthesis, and manipulation..

Generating Lego designs

Matthew Siper
@MatthewSiper

In "Controllable Path of Destruction", we extend the Path of Destruction (PoD) algorithm to be controllable along various spatial metrics in 2D and 3D. We show the efficiency of the PoD algorithm via its ability to generate 3D lego cars from a dataset of only 15 examples.



3:39 AM · Jul 3, 2023 · **17.3K** Views

Controllable Path of Destruction

Matthew Siper, *Game Innovation Lab*, *New York University*, New York, USA, ms12010@nyu.edu
Sam Earle, *Game Innovation Lab*, *New York University*, New York, USA, sam.earle@nyu.edu
Zehua Jiang, *Game Innovation Lab*, *New York University*, New York, USA, zehua.jiang@nyu.edu
Ahmed Khalifa, *Institute of Digital Games*, *University of Malta*, Msida, Malta, ahmed@akhalifa.com
Julian Togelius, *Game Innovation Lab*, *New York University*, New York, USA, julian@togelius.com

https://arxiv.org/abs/2305.18553

***Abstract*—Path of Destruction (PoD) is a self-supervised method for learning iterative generators. The core idea is to produce a training set by destroying a set of artifacts, and for each destructive step create a training instance based on the corresponding repair action. A generator trained on this dataset can then generate new artifacts by repairing from arbitrary states. The PoD method is very data-efficient in terms of original training examples and well-suited to functional artifacts composed of categorical data, such as game levels and discrete 3D structures. In this paper, we extend the Path of Destruction method to allow designer control over aspects of the generated artifacts. Controllability is introduced by adding conditional inputs to the state-action pairs that make up the repair trajectories. We test the controllable PoD method in a 2D dungeon setting, as well as in the domain of small 3D Lego cars.**

***Index Terms*—Procedural Content Generation, Supervised Learning, Repair Function, Controllability, Data Augmentation**

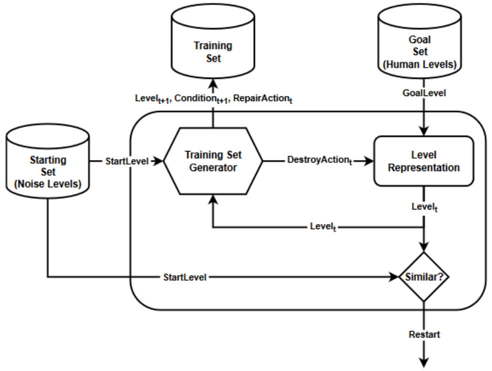


Fig. 1: System diagram of the path of destruction data generation loop including the new condition signal.

I. Introduction

Self-supervised learning in various guises has enabled dramatic advances in generative AI over the last decade. Generative Adversarial Networks [1] and diffusion models [2] have enabled the creation of high-quality images in a variety of styles, and transformers underlie large language models [3] which are poised to impact any field of human activity which involves text processing. These deep learning architectures are generally tied to particular representation formats: in particular, transformers operate on sequences of tokens and GANs and diffusion models work with matrices of real numbers which are interpreted as RGB values. But what these methods are above all is data-hungry. A GAN or diffusion model is trained on at least thousands, often millions, of images [4], and LLMs are typically trained on terabytes of text [5].

This leaves a large space of creative domains which are not naturally expressed as matrices of real numbers or sequences of tokens underserved by current self-supervised learning methods. Even more importantly, for most domains comparatively little data is available to train on, rendering existing self-supervised methods largely ineffective.

Path of Destruction is a recently developed self-supervised method for learning generators of structured content. The method is very data-effective. Although it was developed separately from diffusion models, it has conceptual similarities. The basic idea is to iteratively destroy the target content, one small change at a time, and create a dataset of the changes. In the generated dataset, each instance has part of the destroyed state as an input and the associated repair action (reverse of the destruction action) as the target. By rolling out multiple paths of destruction, an arbitrarily large set of repair actions can be created even when starting with a small initial set of artifacts. Standard supervised learning can then be used to train a generator that generates novel artifacts by "repairing" from random starting states.

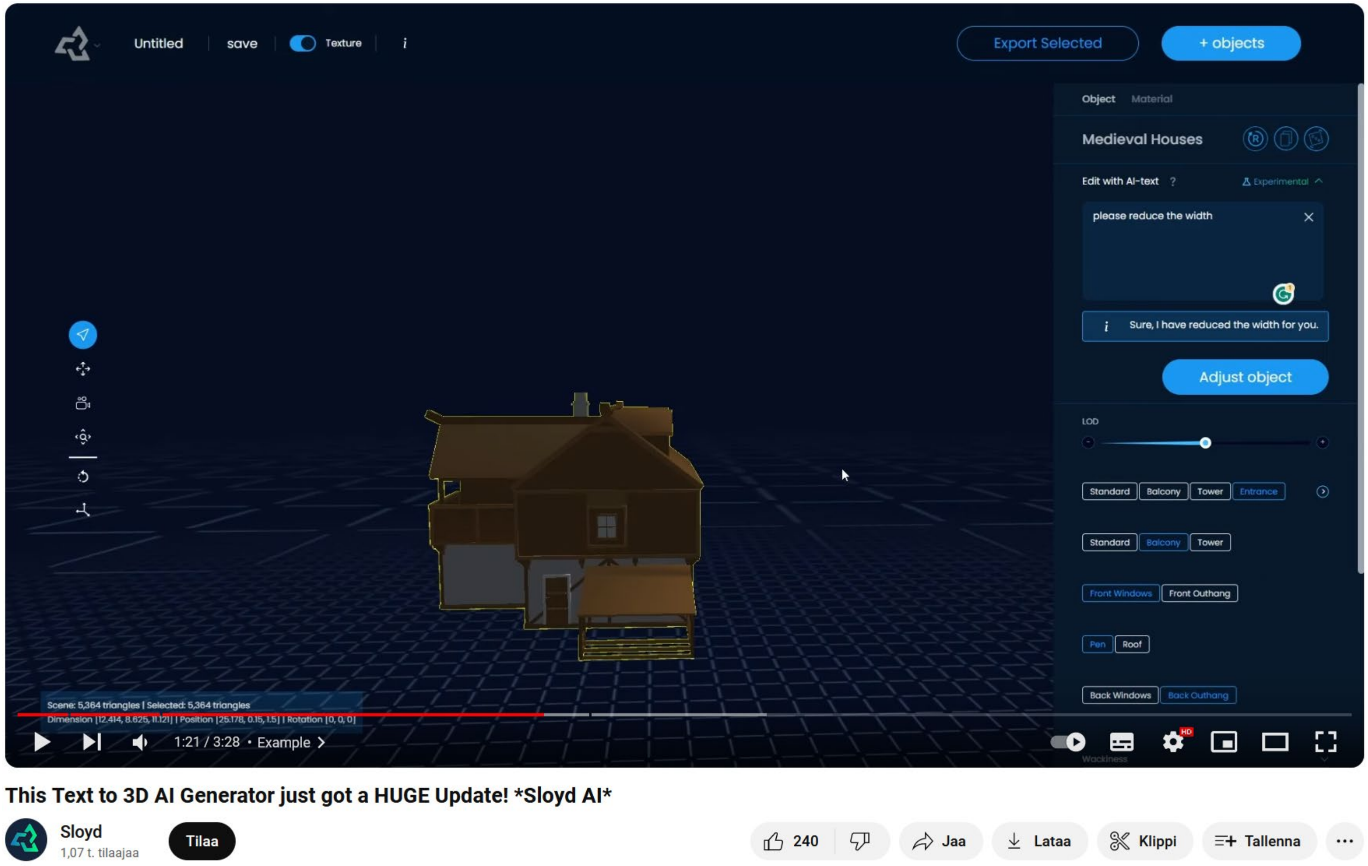
The original Path of Destruction method [6] could generate novel, playable levels for several 2D games based on training on as little as 5 source levels, although performance improved with more levels to train on. However, that algorithm provided no way for the user to influence the design of the generated content beyond curating the training set. In this paper, we describe and evaluate a new version of the Path of Destruction algorithm which allows a user to control aspects of the output. This opens up new avenues for using Path of Destruction as a design tool in interactive applications [7].

The basic idea of Controllable Path of Destruction is to include conditional inputs in the generator. These conditional inputs are also part of the training data that is generated by destroying the artifact and are based on the properties of the artifact that was originally destroyed in the process. For example, if the controllable aspect of the generator is the number of tiles of type X, then the dataset will contain a

Sloyd.ai

Instead of one generic model, offers a parametric custom model for each object type.

High quality, low diversity.



https://www.youtube.com/watch?v=RN0Cafij3rI

SceneTex: High-Quality Texture Synthesis for Indoor Scenes via Diffusion Priors

Dave Zhenyu Chen[1], Haoxuan Li[1], Hsin-Ying Lee[2], Sergey Tulyakov[2], Matthias Nießner[1],

[1]Technical University of Munich, [2]Snap Research

Paper | arXiv | Video | Code

https://daveredrum.github.io/SceneTex/

SCENETEX generates high-quality textures for 3D indoor scenes from the given text prompts. At its core, SceneTex proposes a multiresolution texture field to implicitly encode the mesh appearance. We optimize the target texture via a score-distillation-based objective function in respective RGB renderings. To further secure the style consistency across views, we introduce a cross-attention decoder to predict the RGB values by cross-attending to the pre-sampled reference locations in each instance. Our method enables various and accurate texture synthesis for 3D-FRONT scenes, demonstrating significant improvements in visual quality and prompt fidelity over the prior texture generation methods.

TextureDreamer (2024)

Takes in a few reference images, generates textures for any 3D shape.

No code or models available (yet)

https://arxiv.org/abs/2401.09416

TextureDreamer: Image-guided Texture Synthesis through Geometry-aware Diffusion

Yu-Ying Yeh[13] Jia-Bin Huang[23] Changil Kim[3] Lei Xiao[3] Thu Nguyen-Phuoc[3] Numair Khan[3]
Cheng Zhang[3] Manmohan Chandraker[1] Carl S Marshall[3] Zhao Dong[3] Zhengqin Li[3]

[1]University of California, San Diego [2]University of Maryland, College Park [3]Meta

Figure 1. **Texture transfer from sparse images.** Given a small number of images and a target mesh, our method synthesizes geometry-aware texture that looks similar to the input appearances for diverse objects.

Abstract

We present TextureDreamer, a novel image-guided texture synthesis method to transfer relightable textures from a small number of input images (3 to 5) to target 3D shapes across arbitrary categories. Texture creation is a pivotal challenge in vision and graphics. Industrial companies hire experienced artists to manually craft textures for 3D assets. Classical methods require densely sampled views and accurately aligned geometry, while learning-based methods are confined to category-specific shapes within the dataset. In contrast, TextureDreamer can transfer highly detailed, intricate textures from real-world environments to arbitrary objects with only a few casually captured images, potentially significantly democratizing texture creation. Our core idea, personalized geometry-aware score distillation (PGSD), draws inspiration from recent advancements in diffuse models, including personalized modeling for texture information extraction, variational score distillation for detailed appearance synthesis, and explicit geometry guidance with ControlNet. Our integration and several essential modifications substantially improve the texture quality. Experiments on real images spanning different categories show that TextureDreamer can successfully transfer highly realistic, semantic meaningful texture to arbitrary objects, surpassing the visual quality of previous state-of-the-art. Project page: https://texturedreamer.github.io

Real-world production pipelines

The reality

- Text-to-X almost never gives exactly what you want
- Control interfaces are still emerging
- Most AI tools are highly dedicated
- Typical: inpaint to fix mistakes, chain multiple AI tools, e.g., DALL-E 3 image to Stable Video Diffusion
- ComfyUI emerging as a powerful experimentation platform

Jussi Kemppainen: Generative AI in Game Development | Games Now!

https://www.youtube.com/watch?v=q8eMFjujitY

StableDiffusion and ControlNet integrated in a painting app

[https://x.com/dreamwieber/status/172754882274490792 6?s=20](https://x.com/dreamwieber/status/1727548822744907926?s=20)

https://apps.apple.com/us/app/sagebrush-ai-painter/id6472820126

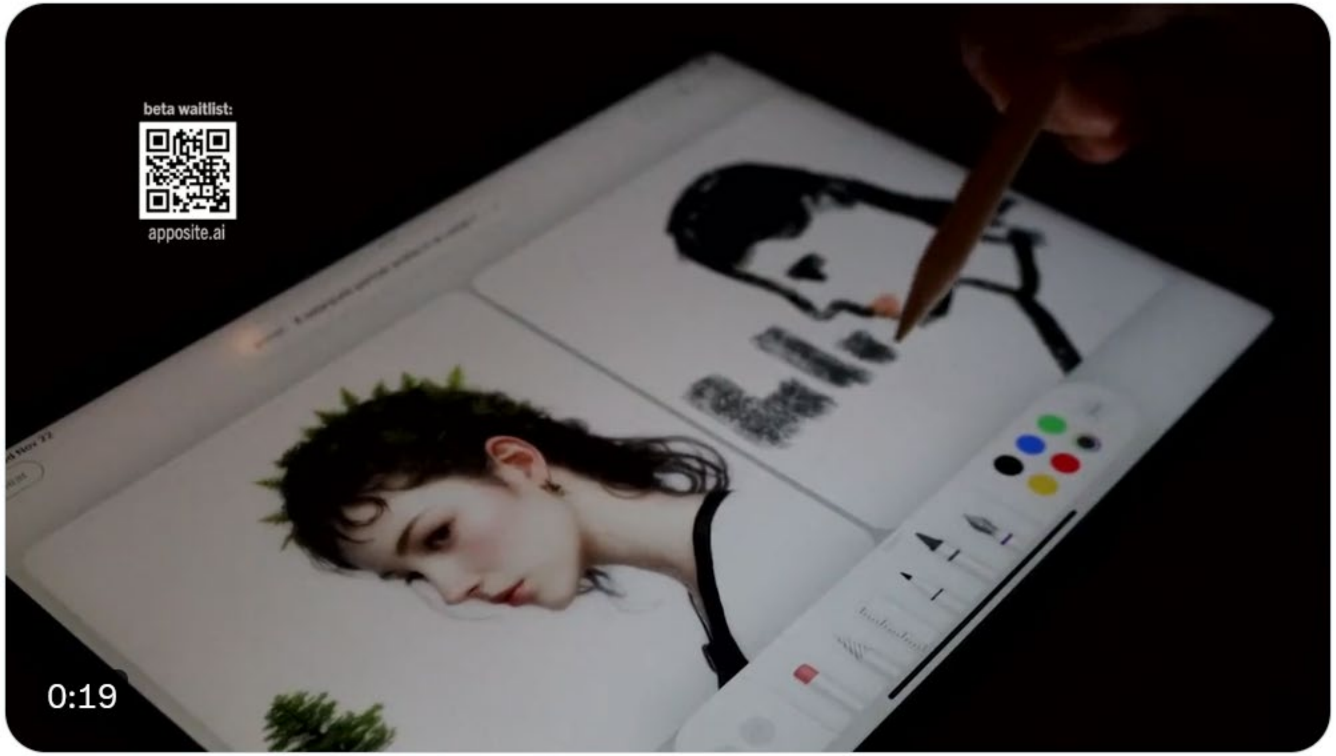
You can also create something similar using ComfyUI and a screen capture node

Gregory Wieber @dreamwieber · Nov 23, 2023

My new iPad app that lets you paint with #ai just got approved for TestFlight 🚀

So much interest, thank you! I'll be rolling out to close friends first, and scaling up over time.

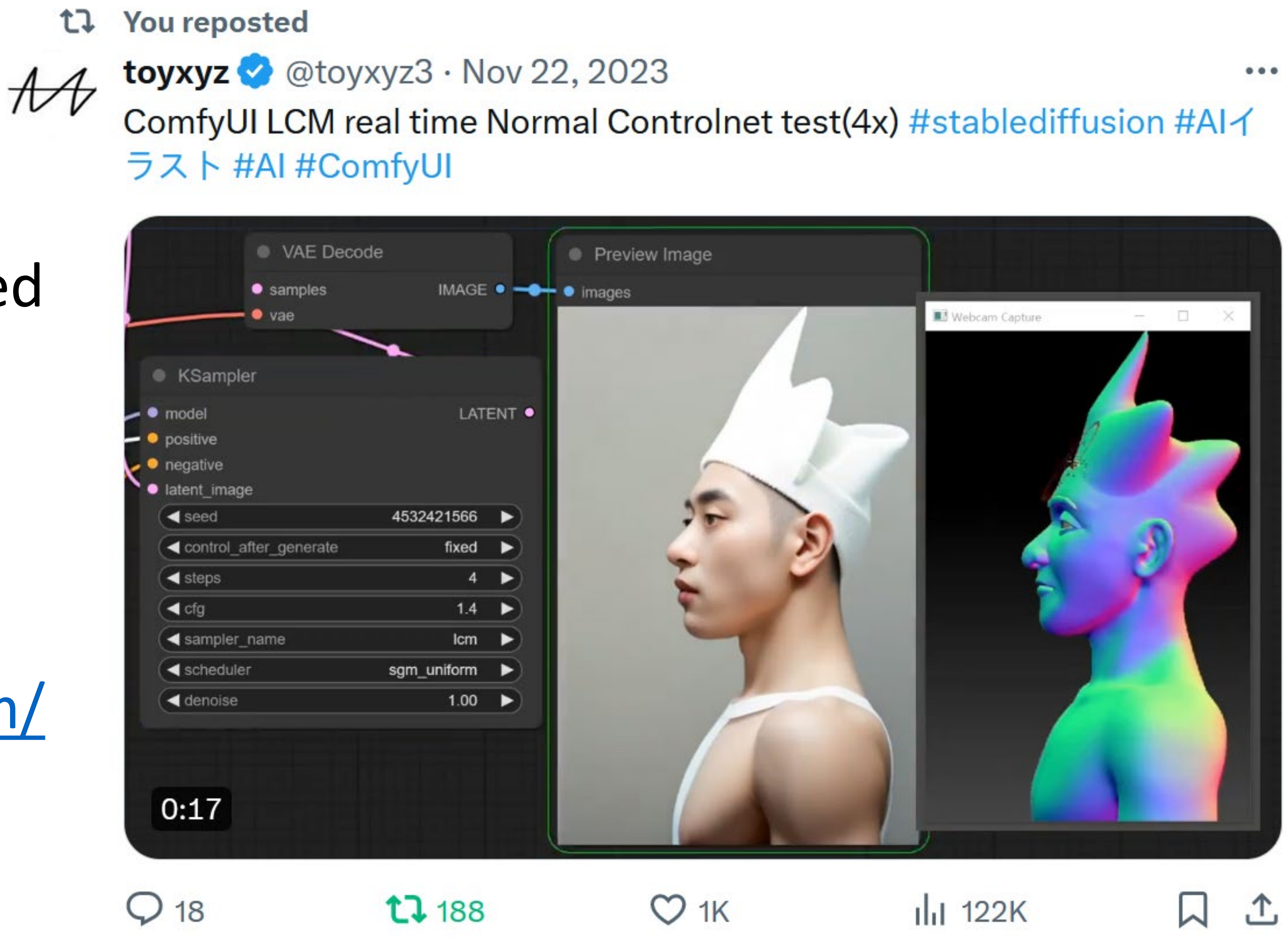
Sign up for the waitlist, and don't forget to follow here! ✨



Same in ComfyUI using a webcam capture node to feed live 3D modeling as ControlNet input

How to replicate:

https://comfyworkflows.com/workflows/62906180-21b5-48ba-a131-44e1eae2e915



https://x.com/toyxyz3/status/1727343696692277367?s=20

Comfy Workflows

Share art/workflow

Search for workflows, node types, art, creators, models, etc.

- Home
- Workflows
- Images
- Videos
- Creators
- Featured
- Trending
- Following
- Guides
- ComfyUI extension
- ComfyUI Cloud
- Profile
- Give Feedback

Share, discover, & run ComfyUI workflows

Explore thousands of workflows created by the community.

https://comfyworkflows.com/

Latest workflows

Browse →

Simple workflow for beginners with controlnet, lora and img2img.

+ Follow Creator

created a day ago

View workflow

img2img | controlnet | lora | 27 nodes | 213 | 40

Simple workflow for beginners with Lora & Img2img

+ Follow Creator

created a day ago

View workflow

img2img | lora | 13 nodes | 67 | 14

NewRealityXL ❗ All-In-One Photographic v2.1 | Basic Workflow | Barren Wardo

+ Follow Creator

created 3 days ago

View workflow

8 nodes | 248 | 58

Posted by u/tarkansarim 28 days ago

655

Cosmic Horror - AnimateDiff - ComfyUI

Workflow Included

https://www.reddit.com/r/StableDiffusion/comments/1925ipt/cosmic_horror_animatediff_comfyui/

Midjourney + Stable Video Diffusion + Topaz.

“The water simulation could technically be achieved with Runway or Pika as alternatives but Topaz was used for frame interpolation which is why the water seems to be so smooth”

r/midjourney • 1 mo. ago
AuralTuneo

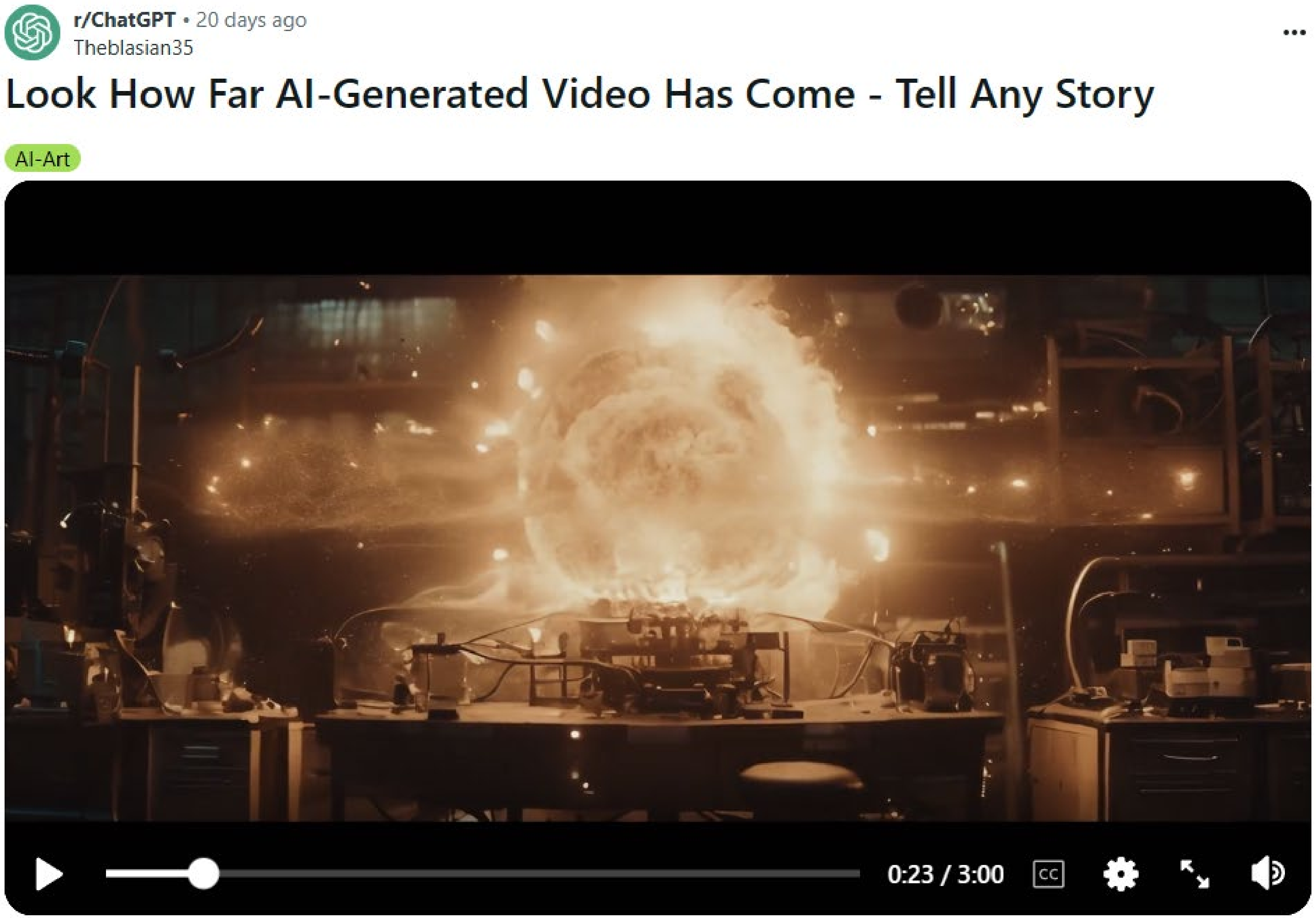
Pushing the limits of AI video

Showcase

https://www.reddit.com/r/midjourney/comments/18qkhaf/pushing_the_limits_of_ai_video/

“Midjourney to Magnific. Then Runway and Pika to animate. topaz to upscale the video. Edited in premiere. Did a lot of the voices with Eleven Labs.”

YouTube 4K version: https://youtu.be/WClsHNQHcKQ?si=udLkaRoe16_H4Fo3



Look How Far AI-Generated Video Has Come - Tell Any Story : r/ChatGPT (reddit.com)

Speech-to-text + LLM + text-to-speech

Closing the “natural conversation loop” has been hard because all the machine learning components have some latency.

Client-cloud communication adds to that.

Now, it looks like NVIDIA has been able to push the latency low enough.

https://www.youtube.com/watch?v=UamcBgWz0i8

I spoke to an Nvidia AI-powered NPC about his ramen and his responses were frighteningly good

By Jacob Ridley published January 09, 2024

Ramen, chit chat and AI-powered people.

COMMENTS

I just had a conversation with an in-game NPC that could easily appear scripted. Except it wasn't, at all. I asked a question, the NPC answered, and all thanks to Nvidia's ACE technology and Convai.

You might have caught sight of Nvidia ACE in action during the company's Special Address stream. It's essentially a technology that allows in-game NPCs to react and respond to players in real-time, with voiced dialogue and animations. Nvidia's been showing off the same tech demo called Kairos, which takes place inside a ramen restaurant in a cyberpunk world, since ACE was announced back at Computex. Over at CES 2024, I got to try it out for myself.

Javi Lopez ⛩️
@javilopen

Subscribe

Midjourney, DALL•E 3 and GPT-4 have opened a world of endless possibilities.

I just coded "Angry Pumpkins 🎃" (any resemblance is purely coincidental 😂) using GPT-4 for all the coding and Midjourney / DALLE for the graphics.

Here are the prompts and the process I followed:

4:38 PM · Oct 31, 2023 · **5.2M** Views

https://x.com/javilopen/status/1719363262179938401?s=20

Javi Lopez ⛩️
@javilopen

Subscribe

🤖 Programming (GPT-4)

🔗 Full source code here: bestaiprompts.art/angry-pumpkins...

Although the game is just 600 lines of which I haven't written ANY, this was the most challenging part. As you can see, I got into adding many details like different particle effects, different types of objects, etc. And to this day, we're still not at a point where GPT-4 can generate an entire game with just a prompt. But I have no doubt that in the future we'll be able to create triple AAA video games just by asking for it.

Anyway, back to the present, the TRICK is to request things from GPT-4 iteratively. Actually, very similar to how a person would program it: Starting with a simple functional base and iterate, expand, and improve the code from there.

Let's see some tricks and prompts I used:

👉 Start with something simple

- "Can we now create a simple game using matter.js and p5.js in the style of "Angry Birds"? Just launch a ball with angle and force using the mouse and hit some stacked boxes with 2D physics.

👉 And from there, keep asking for more and more things. And every

@levelsio
@levelsio

✨ I made my first video game with ChatGPT:

1) ChatGPT generates a text-based adventure game with DALL-E 3 generating images for it

2) Every time you play the game is different because it generates the story and images live

3) The images from DALL-E are sent to @runwayML which turns images into video

4) The text is sent to @elevenlabs which turns the text adventure into a pirate narrator voice

5) It's merged into a video

6) Interactive buttons are overlayed

The game is called:

🐒🏝️🇳🇱The Secret of Monkey Island: Amsterdam (unofficial)

And you can play it here:

monkeyislandamsterdam.com

(video + TTS + buttons doesn't work auto yet, for now manual but text + img works, I'm building an interface for it now)

9:49 PM · Nov 16, 2023 · **2.7M** Views