
GAIA-1: A Generative World Model for Autonomous Driving

Anthony Hu* **Lloyd Russell*** **Hudson Yeo*** **Zak Murez** **George Fedoseev**

Alex Kendall

Jamie Shotton

Gianluca Corrado

Wayve
research@wayve.ai
* equal contributions

Abstract

Autonomous driving promises transformative improvements to transportation, but building systems capable of safely navigating the unstructured complexity of real-world scenarios remains challenging. A critical problem lies in effectively predicting the various potential outcomes that may emerge in response to the vehicle’s actions as the world evolves.

To address this challenge, we introduce GAIA-1 (‘Generative AI for Autonomy’), a generative world model that leverages video, text, and action inputs to generate realistic driving scenarios while offering fine-grained control over ego-vehicle behavior and scene features. Our approach casts world modeling as an unsupervised sequence modeling problem by mapping the inputs to discrete tokens, and predicting the next token in the sequence. Emerging properties from our model include learning high-level structures and scene dynamics, contextual awareness, generalization, and understanding of geometry. The power of GAIA-1’s learned representation that captures expectations of future events, combined with its ability to generate realistic samples, provides new possibilities for innovation in the field of autonomy, enabling enhanced and accelerated training of autonomous driving technology.

1 Introduction

Predicting future events is a fundamental and critical aspect of autonomous systems. Accurate future prediction enables autonomous vehicles to anticipate and plan their actions, enhancing safety and efficiency on the road. To achieve this, the development of a robust model of the world is imperative [1] and huge efforts have been made in the past to build such predictive world models for autonomous driving [2, 3, 4, 5, 6]. A world model [7, 8] learns a structured representation and understanding of the environment that can be leveraged for making informed decisions when driving.

However, current approaches have had significant limitations. World models have been successfully applied to control tasks in both simulation [9, 10, 11, 12, 13] and to real-world robotics tasks [14, 15]. These methods often rely on labeled data, which is challenging to obtain at scale, and models that work on simulated data may not fully capture the complexities of real-world scenarios. Furthermore, due to their low-dimensional representations, these models may struggle to generate highly realistic samples of future events, posing challenges in achieving a high level of fidelity in predictions for complex real-world applications such as autonomous driving.

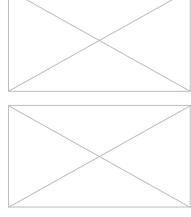
MODE	CONTEXT	CONDITIONING	GENERATED FRAMES
Video rollout			
Action-conditioned rollout		Action: Speed: -- Curvature: LEFT	
Text-conditioned rollout		Text: "The traffic light is green"	
Text-conditioned generation		Text: "It is snowing" Text: "It is night"	
Text and action conditioned generation		Text: "We are 15 meters behind a bus" Action: Speed: ACCELERATE Curvature: RIGHT	
Unconditional generation			

Figure 1: GAIA-1 multimodal video generation. GAIA-1 can generate videos by performing future rollouts starting from a video prompt. These future rollouts can be further conditioned on actions to influence particular behaviors of the ego-vehicle (e.g. steer left), or on text to drive a change in some aspects of the scene (change the color of the traffic light). For speed and curvature we condition the model by passing the sequence of future speed and / or curvature values. Our model can also generate realistic videos from text prompts, or by simply drawing samples from its prior distribution (fully unconditional generation).

Meanwhile, progress in generative image and video generation has harnessed the power of self-supervised learning to learn from large quantities of real-world data to generate remarkably realistic video samples [16, 17, 18]. Yet, a significant challenge persists in this domain: the difficulty of learning a representation that captures the expected future events. While such generative models excel at generating visually convincing content, they may fall short in learning representations of the evolving world dynamics that are crucial for precise future predictions and robust decision-making in complex scenarios.

In this work we introduce GAIA-1, a method designed with the goal of maintaining the benefits of both world models and generative video generation. It combines the scalability and realism of generative video models with the ability of world models to learn meaningful representations of the evolution into the future. GAIA-1 works as follows. First, we partition the model into two components: the world model and the video diffusion decoder. The world model reasons about the scene’s high-level components and dynamics, while the diffusion model takes on the responsibility of translating latent representations back into high-quality videos with realistic detail.

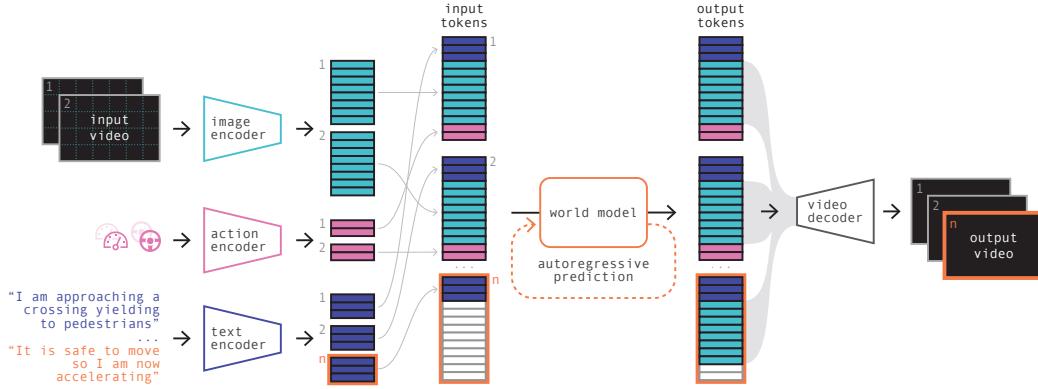


Figure 2: Architecture of GAIA-1. First, we encode information from all input modalities (video, text, action) into a common representation: images, text and actions are encoded as a sequence of tokens. The world model is an autoregressive transformer that predicts the next image token conditioned on past image, text, and action tokens. Finally, the video decoder maps the predicted image tokens back to the pixel space, at a higher temporal resolution.

For the world model, we use vector-quantized representations of video frames to discretize each frame, transforming them into a sequence of tokens. Subsequently, we reframe the challenge of predicting the future into predicting the next token in the sequence [10, 19]. This approach has been widely employed in recent years to train large language models [20, 21, 22, 23], and it is recognized for its effectiveness in enhancing model performance through the scaling of model size and data. It is possible to generate samples within the latent space of the world model through autoregressive generation.

The second component is a multi-task video diffusion decoder that is able to perform high-resolution video rendering as well as temporal upsampling to generate smooth videos from the information autoregressively generated by the world model. Similarly to large language models, video diffusion models have demonstrated a clear correlation between scale of training and overall performance, making both components of GAIA-1 suitable for effective compound scaling.

GAIA-1 is designed to be multimodal, allowing video, text and action to be used as prompts to generate diverse and realistic driving scenarios, as demonstrated in Figure 1. By training it on a large corpus of real-world UK urban driving data, GAIA-1 learns to understand and disentangle important concepts such as static and dynamic elements, including cars, buses, pedestrians, cyclists, road layouts, buildings, and even traffic lights. Further, it provides fine-grained control over both ego-vehicle behavior and other scene features through action and language conditioning.

GAIA-1 demonstrates the ability to manifest the generative rules of the real world. Emerging properties such as learning high-level structures, generalization, creativity, and contextual awareness indicate that the model can comprehend and reproduce the rules and behaviors of the world. Moreover, GAIA-1 exhibits understanding of 3D geometry, for example, by effectively capturing the intricate interplay of pitch and roll induced by road irregularities such as speed bumps. It showcases reactive behaviors of other agents demonstrating the ability to understand causality in decision making of road users. Surprisingly, it shows the capability to successfully extrapolate beyond the training data, for example to driving outside of the boundaries of the road. See Section 7 for a comprehensive list of examples.

The power of GAIA-1’s learned representations to predict future events, paired with control over both ego-vehicle dynamics and scene elements, is an exciting advance that paves the way for improving embodied intelligence and providing synthetic data to accelerate training and validation. World models, such as GAIA-1, are the basis for the ability to predict what might happen next, which is fundamentally important for decision-making in autonomous driving.

2 Model

In this section we describe the model architecture of the trainable components of GAIA-1. The general architecture is presented in Figure 2.

2.1 Encoding Video, Text and Action

GAIA-1 can leverage three different input modalities (video, text, action), which are encoded into a shared d -dimensional space.

Image tokens. Each image frame of a video is represented as discrete tokens. To achieve this, we use a pre-trained image tokenizer for discretization (for details about the pre-training see Section 2.2). Formally, let us consider a sequence of T images $(\mathbf{x}_1, \dots, \mathbf{x}_T)$, where each image \mathbf{x}_t in this sequence is discretized into $n = 576$ discrete tokens using the pre-trained image tokenizer. We obtain a sequence denoted as $(\mathbf{z}_1, \dots, \mathbf{z}_T)$, where each $\mathbf{z}_t = (z_{t,1}, \dots, z_{t,n}) \in \mathbb{R}^n$ corresponds to $n = \frac{H}{D} \times \frac{W}{D}$ discrete tokens. Here, H and W represent the height and width of the input image, while D denotes the downsampling factor of the image tokenizer. These discrete tokens are then mapped to a d -dimensional space via an embedding layer that is trained alongside the world model.

Text tokens. At each time step t , we incorporate information from both text and action. Textual input is encoded using the pre-trained T5-large model [24], resulting in $m = 32$ text tokens per time step. These tokens are mapped to a d -dimensional space through a linear layer that is trained in conjunction with the world model. This process yields a text representation denoted as $\mathbf{c}_t = (\mathbf{c}_{t,1}, \dots, \mathbf{c}_{t,m}) \in \mathbb{R}^{m \times d}$.

Action tokens. For actions, we consider $l = 2$ scalar values (representing speed and curvature). Each scalar is independently mapped to the d -dimensional space via a linear layer that is trained with the world model. Consequently, the action at time step t is represented as $\mathbf{a}_t = (\mathbf{a}_{t,1}, \dots, \mathbf{a}_{t,l}) \in \mathbb{R}^{l \times d}$.

For each time step, the input tokens are interleaved in the following order: text - image - action. The final input of the world model is therefore $(\mathbf{c}_1, \mathbf{z}_1, \mathbf{a}_1, \dots, \mathbf{c}_T, \mathbf{z}_T, \mathbf{a}_T)$. To encode the position of the input tokens, we use a factorized spatio-temporal positional embedding. 1) A learnable temporal embedding is shared across all the tokens of a given time step, i.e. there are T temporal embeddings. 2) A learnable spatial embedding indicates the position of a token within a time step, i.e. there are $m + n + l = 610$ spatial embeddings (m text tokens, n image tokens, and l action tokens) of dimension $d = 4096$.

2.2 Image Tokenizer

When modeling discrete input data with a sequence model, there is a trade-off between the sequence length and the vocabulary size. The sequence length refers to the number of discrete tokens that are needed to describe the data. The vocabulary size corresponds to the number of possible values a single token can take. For language, there are two obvious choices for tokens: characters and words. When using character-level tokens, the input data has a longer sequence length, and each individual token belongs to a smaller vocabulary, but conveys little meaning. When using word-level tokens, the input data has a shorter sequence length, and each token contains a lot of semantics but the vocabulary is extremely large. Most language models [25, 26, 24, 21, 27, 22] use byte-pair encoding (or equivalent) as a trade-off between character-level and word-level tokenization.

Likewise for video, we would like to reduce the sequence length of the input, while possibly making the vocabulary larger, but with tokens that are more semantically meaningful than raw pixels. We do this with a discrete image autoencoder [28]. There are two objectives we would like to achieve in this first stage:

1. Compress the information from raw pixels to make the sequence modeling problem tractable. Images contain a lot of redundant and noisy information. We would like to reduce the sequence length needed to describe the input data.
2. Guide the compression towards meaningful representations, such as semantics, instead of high-frequency signals. The resulting input space for the world model will be simpler to

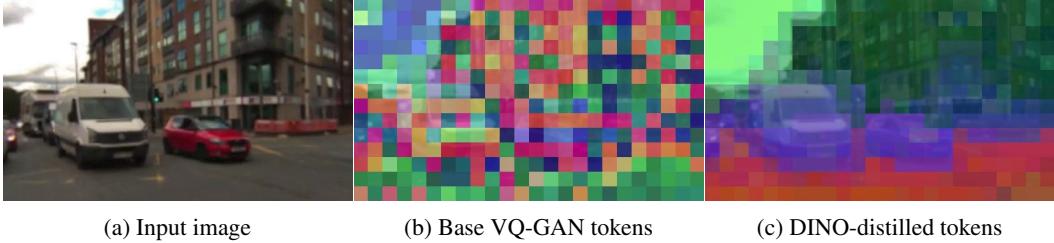


Figure 3: Increasing semantic content of image tokens through DINO distillation. Visualization shows the top 3 PCA components of token embeddings mapped to RGB values. DINO-distilled tokens corresponding to a semantic class (e.g. vehicle, road, or sky) have similar embeddings.

compose with, and less dominated by high-frequency signals that can considerably slow down the learning process.

We reduce the sequence length of the input data by downsampling each input image by a factor $D = 16$ in both height and width. Each image \mathbf{x}_t of size $H \times W$ is described by $n = \frac{H}{D} \times \frac{W}{D}$ tokens with a vocabulary size K . Inspired by [29], we guide the compression towards meaningful representations by regressing to the latent features of a pre-trained DINO model [30], a self-supervised image model that is known to contain semantic information. See Figure 3 for a qualitative example.

The discrete autoencoder is a fully convolutional 2D U-Net [31]. The encoder E_θ quantizes the image features using nearest neighbor look-up from a learnable embedding table [28], resulting in image tokens $\mathbf{z}_t = E_\theta(\mathbf{x}_t)$. Note that the decoder is only used to train the image autoencoder, solely the discrete encoder E_θ is part of the final GAIA-1 model. Due to the decoder being trained on single images it lacks temporal consistency when decoding to a video. For this reason we also train a video decoder that is described in Section 2.4.

The training losses for the image autoencoder are the following:

- **Image reconstruction loss.** The image reconstruction loss is a weighted sum of L_1 , L_2 , perceptual loss $L_{\text{perceptual}}$ [32], and GAN loss L_{GAN} [33].
- **Quantization loss.** To update the embedding vectors, we use the embedding loss and the commitment loss from [28]. We adopted the linear projection of the embedding and L_2 normalization from [34] as we found this helped increase vocabulary usage.
- **Inductive bias loss.** The quantized image features are encouraged to match the image features of a pre-trained DINO [30] model with a cosine similarity loss. Distilling the information from DINO into the learned tokens is important as it allows them to benefit from the inductive biases of this model.

2.3 World Model

As described in Section 2.1 the input of the world model is $(\mathbf{c}_1, \mathbf{z}_1, \mathbf{a}_1, \dots, \mathbf{c}_T, \mathbf{z}_T, \mathbf{a}_T)$. The world model is an autoregressive transformer network that models the sequence input. Its training objective is to predict the next image token in the sequence conditioned on all past tokens, using causal masking in the attention matrix of the transformer blocks [35].

$$L_{\text{world model}} = - \sum_{t=1}^T \sum_{i=1}^n \log p(z_{t,i} | \mathbf{z}_{<t}, z_{t,j < i}, \mathbf{c}_{\leq t}, \mathbf{a}_{<t}) \quad (1)$$

We randomly dropout conditioning tokens during training so that the world model can do (i) unconditional generation, (ii) action-conditioned generation, and (iii) text-conditioned generation.

To further reduce the sequence length of our world model we temporally subsample videos from 25Hz to 6.25Hz. This allows the world model to reason over longer periods without leading to intractable sequence lengths. To recover video predictions at full frame rate we perform temporal super-resolution using the video decoder described in Section 2.4.

2.4 Video Decoder

Following the recent advances in image [36, 37] and video generation [16, 18] we use denoising video diffusion models for the GAIA-1 decoder. A naive approach of independently decoding each frame-tokens to pixel space results in a temporally inconsistent video output. Modeling the problem as denoising a sequence of frames during the diffusion process, where the model can access information across time, greatly improves temporal consistency of the output video.

We follow [38] and use a 3D U-Net with factorized spatial and temporal attention layers. During training, our video diffusion model is conditioned on the image tokens obtained by discretizing input images with the pre-trained image tokenizer E_θ . During inference, the diffusion model is conditioned on the predicted image tokens from the world model.

We train a single model jointly on both image and video generation tasks. Training on videos teaches the decoder to be temporally consistent, while training on images is crucial for the quality of individual frames [16] as it teaches the model to extract information from conditioning image tokens. We disable temporal layers when training on images.

To train our video diffusion decoder for multiple inference tasks we take inspiration from [17] where we can perform multiple tasks by masking certain frames or the conditioning image tokens. We choose to train a single video diffusion model for all tasks as it has been shown that multi-task training improves performance on individual tasks [17]. The tasks include image generation, video generation, autoregressive decoding, and video interpolation. Each task is sampled equally. For example, for the autoregressive generation task, we provide previously generated past frames as context and conditioning image tokens for frames we want to predict. We include both forward and backward autoregressive tasks. See Figure 4 for examples of each task. We also apply a conditioning dropout by randomly masking out each conditioning image token with probability $p = 0.15$ as it helps the model generalize beyond relying on tokens for information and improves temporal consistency.

The video decoder is trained on the noise prediction objective. More specifically, we use the v-parameterization as proposed in [39] because it avoided unnatural color shifts and maintained long-term consistency as similarly found in [16]. In practice, we use a weighted average of L_1 and L_2 losses. The video decoder loss L_{video} is:

$$L_{\text{video}} = \mathbb{E}_{\epsilon, t'} \left[\|\epsilon_\theta(\mathbf{x}^{t'}, t', \mathbf{z}, \mathbf{m}) - \epsilon\|_2^2 \right] \quad (2)$$

where:

- ϵ_θ is the denoising video model.
- ϵ is the denoising target, which uses the v-parameterization.
- $t' \sim U(0, 1)$ is the sampled discrete diffusion time.
- $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_{T'})$ is a video sequence of length T' .
- $\mathbf{x}^{t'} = \alpha_{t'} \mathbf{x} + \sigma_{t'} \epsilon$ represents the noised video, with $\alpha_{t'}$ and $\sigma_{t'}$ functions of t' that define the noise schedule.
- $\mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_{T'}) = E_\theta(\mathbf{x})$ is the sequence of conditioning image tokens.
- $\mathbf{m} = (\mathbf{m}_1, \dots, \mathbf{m}_{T'})$ is a sequence of image masks as specified by the training task (see Figure 4).

3 Data

Our training dataset consists of 4,700 hours at 25Hz of proprietary driving data collected in London, UK between 2019 and 2023. This corresponds to approximately 420M unique images. During training we balance over a customizable set of features to control the distribution of data (Figure 5). We achieve this by sampling individual data points with weighting inversely proportional to the (binned and precomputed) empirical distribution of a given feature. For a given example we take the joint probability across all features to balance and stochastically decide whether to include or discard that example. We can control the strength of balancing by raising the sampling weight to an exponent, where an exponent of 0 would result in the empirical distribution (no balancing) and an exponent of 1

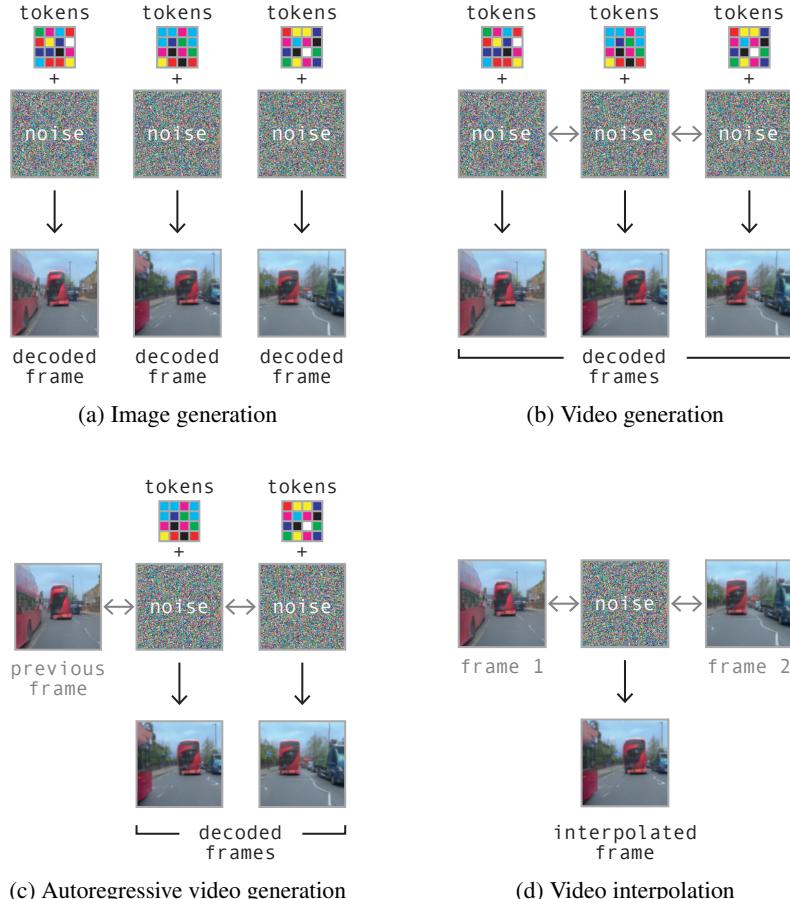


Figure 4: Video decoder training tasks. Each task is defined by masking ground truth images and context tokens. We pass noise as input for frames we want to predict. Tokens are provided for predicted frames except for video interpolation task where the diffusion process is guided solely by image context.

would result in a uniformly balanced distribution. We used an exponent of 0.5 for all features as a compromise between final balancing achieved and the severity of discarding samples for training efficiency.

For the tokenizer we balanced over (latitude, longitude, weather category) to account for geography and visually distinct weather conditions ensuring our tokenizer can adequately represent a diverse range of scenes.

For the world model and the video diffusion model we balanced over (latitude, longitude, weather category, steering behavior category, speed behavior category), additionally considering speed and steering behaviors to ensure the dynamics of different behaviors are captured and sufficiently modeled by the world model and the temporal decoder.

Our validation dataset contains 400 hours of driving data from runs not included in the training set. The runs selected for validation are those that pass through predetermined geofences as well as a selection of randomly selected runs. We further split our validation set into strict geofences in order to analyze only those samples strictly within the validation geofence (i.e., roads never seen during training) and another geofence around our main data collection routes (i.e., roads seen during training) as a way to monitor overfitting and generalization.

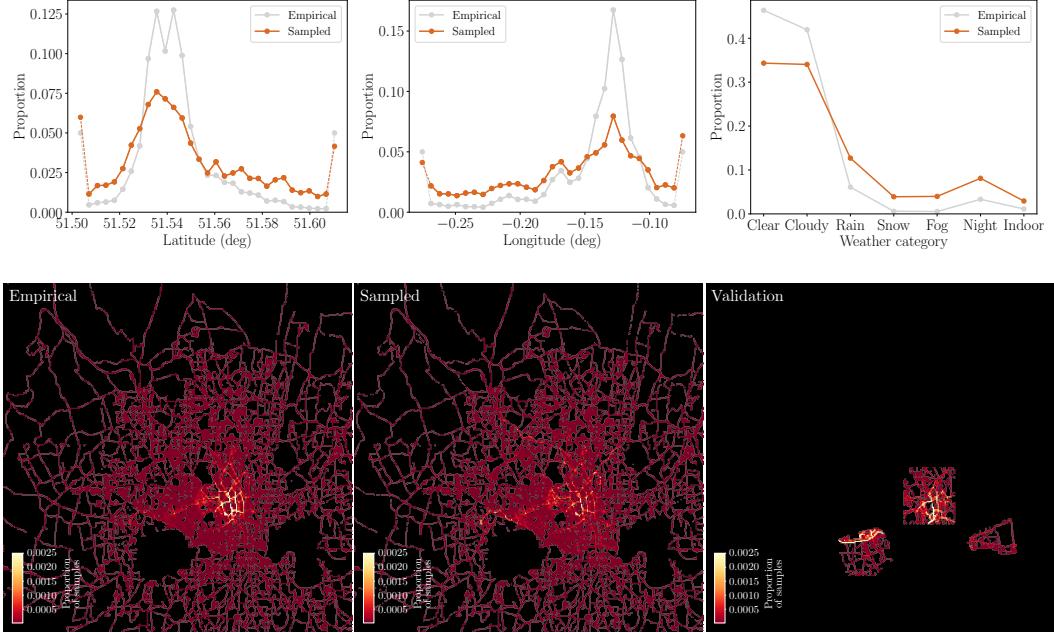


Figure 5: Data sampling. The top row shows the empirical distribution and the sampled distribution for three features we selected to balance over during training: latitude, longitude and weather condition. Dashed lines indicate data outside of the range we balanced over. The bottom row shows the geographical heatmap of sampled latitude and longitude coordinates for the whole training set, the sampled training set and the geofenced validation set.

4 Training Procedure

In this section, we describe how the three trainable components of GAIA-1 were optimized. We provide details of hyperparameter configurations, hardware used and training times.

4.1 Image Tokenizer

The image tokenizer (0.3B parameters) was trained on images of resolution $H \times W = 288 \times 512$ (9/16 ratio). The spatial downsampling of the encoder is $D = 16$, therefore each image is encoded as $n = 18 \times 32 = 576$ discrete tokens with a vocabulary size $K = 8192$. The bit compression is $\frac{288 \times 512 \times 3 \times 8}{18 \times 32 \times 13} \approx 470$.

The discrete autoencoder was optimised with AdamW [40] and a learning rate of 1×10^{-4} , weight decay 0.01, beta coefficients (0.5, 0.9). The loss weights are $\lambda_{L_1} = 0.2$, $\lambda_{L_2} = 2.0$, $\lambda_{L_{\text{perceptual}}} = 0.1$, $\lambda_{L_{\text{GAN}}} = 1.0$, $\lambda_{L_{\text{codebook}}} = 1.0$, $\lambda_{L_{\text{DINO}}} = 0.1$.

The model was trained for 200k steps in 4 days with a batch size equal to 160, split across 32 A100 80GB GPUs. We used 5k of linear warm-up and 10k of cosine decay to a final learning rate of 1×10^{-5} .

4.2 World Model

The world model (6.5B parameters) was trained on video sequences of size $T = 26$ at 6.25 Hz, which correspond to 4s-long videos. The text was encoded as $m = 32$ text tokens per time step, and the action as $l = 2$ tokens. The total sequence length of the world model is therefore $T \times (m + n + l) = 15860$.

The world model was optimized with AdamW and a learning rate of 1×10^{-4} , weight decay 0.1, beta coefficients (0.9, 0.95), norm gradient clipping 1.0. Training examples were either unconditioned,

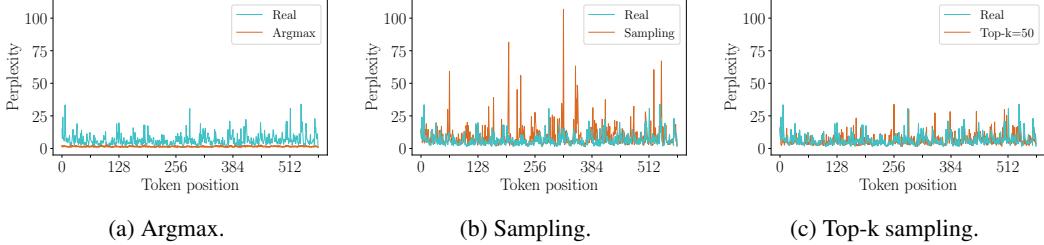


Figure 6: Perplexity of the world model as a function of the position of the generated token. We consider the $n = 576$ tokens of a single image frame. We compare the perplexity of the tokens from a real image, to those generated with the following strategies: argmax, sampling, or top-k. In (a), by inspecting the perplexity of a real image we notice it oscillates between low and high values, meaning there is a good range of diversity in those tokens. In contrast, if we look at the argmax strategy, we notice the perplexity only takes extremely low values (no diversity, manifesting in the predicted frames to repeat themselves). Conversely in (b), if we sample from the entire distribution, the perplexity of some tokens can take extremely high values, due to sampling from the unreliable tail. In (c), we observe that top-k=50 sampling produces tokens that have a similar perplexity distribution to real tokens.

action-conditioned, or text conditioned. The ratios of these respective conditioning modes were 20%/40%/40%.

The model was trained for 100k steps in 15 days, with 2.5k of linear warm-up and 97.5k of cosine decay reducing the learning rate by a factor of 10 over the course of training. The batch size was 128 split across 64 A100 80GB GPUs. We used the FlashAttention v2 implementation [41] in the transformer module, as it offered significant advantages in terms of both memory utilization and inference speed. To optimize distributed training, we used the Deepspeed ZeRO-2 training strategy [42] with activation checkpointing.

4.3 Video Decoder

The video decoder (2.6B) was trained on sequences of $T' = 7$ images of resolution $H \times W = 288 \times 512$ sampled from the dataset at either 6.25 Hz, 12.5 Hz or 25 Hz. The training tasks (Figure 4) were sampled with equal probability. We used a cosine β -noise schedule [43].

The video decoder was optimized with AdamW and a learning rate of 5×10^{-5} , weight decay 0.01, beta coefficients (0.9, 0.99), norm gradient clipping 1.0. The model was trained for 300k steps in 15 days, with 2.5k of linear warm-up and 5k of cosine decay to a final learning rate of 1×10^{-6} . We used a weighted average of L_1 and L_2 losses with weights $\lambda_{L_1} = 0.1$ and $\lambda_{L_2} = 1.0$. The batch size was 64 split across 32 A100 80GB GPUs. We used an exponential moving average for the parameters with a decay of 0.999. The training strategy was also Deepspeed ZeRO-2 with activation checkpointing.

5 Inference

In this section, we describe in more detail the inference procedure of the world model and the video decoder.

5.1 World Model

Sampling. The world model autoregressively predicts the next image token, conditioned on previous text, image and action tokens. Given the past tokens we perform n forward steps to generate one new image frame. At each step we must sample a token from the predicted logits to select the next token in our sequence. Empirically we observed that maximization-based sampling (i.e. argmax) generates futures that get stuck in a repetitive loop, similarly to language models [44]. Conversely, if we simply sample from the logits, the selected token can come from the unreliable tail of the probability distribution, which throws the model out-of-distribution, see Figure 6.



Figure 7: Classifier-free guidance.

To encourage diversity as well as realism we employ top-k sampling to sample the next image token from the top-k most likely choices. The chosen value of k is a function of the number of tokens that constitute an image frame as well as the pre-learnt codebook (vocabulary) size.

Our world model can be used to roll out possible futures given starting context as well as generating futures from scratch without any starting context. For long video generation, if the length of the video exceeds the context length of the world model, we employ a sliding window.

Text-conditioning. The video prediction can be prompted, and thus directed, with text. At training time, we condition our video sequences with text coming from either online narration or offline metadata sources. Because these text sources are imperfect, to improve the alignment between generated futures and the text prompt, we employ classifier-free guidance [45, 46] at inference time. The effect of guidance is to increase text-image alignment by decreasing the diversity of possible samples. More precisely, for each next token to predict, we compute logits conditioned on text as well as logits with no conditioning (unconditioned). At inference, we can then amplify the differences between the unconditioned and the text-conditioned logits with a scale factor to give the final logits used for sampling.

$$l_{\text{final}} = (1 + t)l_{\text{conditioned}} - tl_{\text{unconditioned}} \quad (3)$$

By substituting the unconditioned logits with those conditioned on another text prompt, we can perform “negative” prompting [47]. Pushing the logits away from the negative prompt and towards the positive one encourages the future tokens to include the “positive” prompt features while removing the “negative” ones.

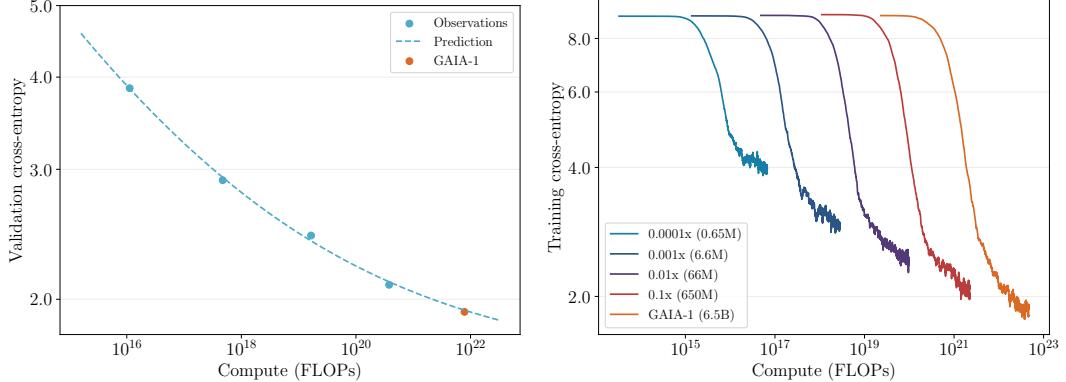
We found it was important to schedule the scale factor used for guidance over tokens as well as frames. Scheduling over tokens allows some to be sampled with high guidance (hence adhering strongly to the prompt) and others to be sampled with low guidance (hence increasing sample diversity). Scheduling over frames allows for controlling the transition from earlier frames as well as mitigating compounding guidance over subsequent consecutive frames. In Figure 7 we show an example guidance schedule over twelve frames. Typically we used a schedule that sampled tokens with linearly decreasing guidance over tokens and we lowered the guidance over future frames with a cosine decay, with or without an initial plateau. We note that guidance scale and schedule are hyperparameters to be tuned to particular use cases.

5.2 Video Decoder

To decode a sequence of generated tokens from the world model, we use the following video decoding method:

1. Decode the first $T' = 7$ frames, conditioned on the corresponding T' image tokens.
2. Autoregressively decode the next $T' - 2$ frames, using 2 past overlapping frames as image context, and the following $T' - 2$ image tokens.
3. Repeat the autoregressive process until the N frames have been generated at 6.25 Hz.
4. Temporally upsample the N frames from 6.25 Hz to 12.5 Hz
5. Temporally upsample the $2N - 1$ frames from 12.5 Hz to 25.0 Hz

We use the DDIM sampler [48] with 50 diffusion steps. During autoregressive decoding, we see a trade-off between reflecting token information content in the generated video and temporal



(a) The final performance of the GAIA-1 world model could be predicted with smaller models trained with less than $20\times$ the compute.
(b) Training loss curves for world models up to 10,000x smaller. We used an exponential moving average to smooth the training loss curves.

Figure 8: Validation and training cross-entropy of the world model.

consistency. To balance between these two objectives, we calculate a weighted average of the two tasks [18].

$$\tilde{\epsilon}_\theta(\mathbf{x}^{t'}, t', \mathbf{z}, \mathbf{m}) = w \cdot \epsilon_\theta^\pi(\mathbf{x}^{t'}, t', \mathbf{z}, \mathbf{m}) + (1 - w) \cdot \epsilon_\theta(\mathbf{x}^{t'}, t', \mathbf{z}, \mathbf{m}) \quad (4)$$

where function $\epsilon_\theta^\pi(\mathbf{x}^{t'}, t', \mathbf{z}, \mathbf{m})$ denoises each frame individually as images and function $\epsilon_\theta(\mathbf{x}^{t'}, t', \mathbf{z}, \mathbf{m})$ denoises the sequence of frames jointly as a video. In practice, we simply switch on and off the temporal layers. We apply this weighted average randomly for each diffusion step with probability $p = 0.25$ and weight $w = 0.5$.

While exploring different inference approaches for video decoding we found that decoding video frames autoregressively backwards starting from the end of the sequence led to more stable objects and less flickering on the horizon. In our overall video decoding method, we thus decode the last T' frames and autoregressively decodes the remaining frames backward from there.

6 Scaling

The formulation of the world modeling task in GAIA-1 shares a commonality with the approach frequently used in large language models (LLMs). In both instances, the task is streamlined to focus on predicting the next token. Although this approach is adapted for world modeling in GAIA-1 rather than the traditional language tasks seen in LLMs, it is intriguing to observe that scaling laws [49, 21, 27], analogous to those observed in LLMs, are also applicable to GAIA-1. This suggests the broader applicability of scaling principles in modern AI models across diverse domains, including autonomous driving.

To explore scaling laws with GAIA-1, we predicted the final performance of the world model using models trained with less than $20\times$ the compute. We evaluated those models on a held-out geofenced validation set by measuring cross-entropy. A power-law of the form $f(x) = c + (x/a)^b$ was then fitted to the data points. In Figure 8a we can see that the final cross-entropy of GAIA-1 could be predicted with high accuracy.

The models used to fit the power-law ranged from 10,000x to 10x smaller models in terms of parameters (0.65M to 650M), as visualized in Figure 8b. Similarly to [49], the compute was estimated as a function of the parameter count. If we denote by C the compute and by N the parameter count (excluding embedding layers), the number of floating point operations for a forward-backward pass of a single token is given by $C = 6N$. To obtain the total amount of compute, this value is multiplied by the number of training tokens.

It is worth noting that our extrapolation leads us to the conclusion that there is substantial potential for further improvement through the expansion of both data and computational resources.



Figure 9: Images generated by GAIA-1, highlighting the diversity of the generated driving scenes.

7 Capabilities and Emerging Properties

In this section we showcase the capabilities and emerging properties of GAIA-1 through a series of qualitative examples. The comprehensive list of video examples can be found [here](#). Figure 9 shows the variety of scenarios that can be generated by our model. As evidenced by the examples presented in the rest of this section, GAIA-1 exhibits a level of understanding and summarization of the generative rules of the world through the following emergent properties:

1. Learning high-level structures and scene dynamics: it generates coherent scenes with objects positioned in plausible locations and exhibiting realistic object interactions, such as traffic lights, rules of the road, give ways, etc. This suggests that the model is not just memorizing statistical patterns but is understanding the underlying rules that govern the arrangement and behavior of objects in the world (see Section 7.1).
2. Generalization and creativity: it can generate novel and diverse videos that go beyond specific instances in the training set. It can produce unique combinations of objects, movements, and scenes that were not explicitly present in the training data, demonstrating remarkable extrapolation capabilities. This demonstrates a certain level of generalization and creativity, which suggests an understanding of the underlying generative rules that govern video sequences (see Section 7.2).
3. Contextual awareness: GAIA-1 can capture contextual information and generate videos that reflect this understanding. For example, it can generate coherent actions and responses in videos based on the initial conditions or the context provided. Moreover, GAIA-1 exhibits the understanding of 3D geometry, effectively capturing the intricate interplay of pitch and roll induced by road irregularities (e.g. speed bumps). This contextual awareness suggests that the models are not merely reproducing statistical patterns but are actively processing and summarizing the given information to generate appropriate video sequences (see Section 7.3).

7.1 Generation of Long Driving Scenarios

GAIA-1 can generate stable long videos (minutes) entirely from imagination (Figure 10). In order to do this, the model leverages its learned implicit prior distribution of the world to generate fully-imagined realistic driving scenarios, with complex road layouts, buildings, cars, pedestrians, and more. This is a demonstration that GAIA-1 understands the rules that underpin the world we inhabit and its structures and dynamics.

7.2 Generation of Multiple Plausible Futures

GAIA-1 has the ability to generate a variety of distinct future scenarios based on a single initial prompt. When presented with a brief video as context, it can generate numerous plausible and diverse outcomes by repeatedly sampling. GAIA-1 accurately models multiple potential future scenarios in response to the video prompt while maintaining consistency with the initial conditions observed in the video. As seen in Figure 11, the world model can reason about (i) dynamic interactions with road users (e.g. giving way or not giving way), (ii) multimodal ego-behaviors (e.g. going straight or turning at a roundabout), and (iii) multimodal dynamic scene (e.g. variable traffic density and types of road users such as pedestrians, cyclists, motorcyclists, vehicles) and static scene (e.g. road layout, buildings, vegetation).

7.3 Fine-Grained Control of the Ego-Vehicle Behavior and Driving Scenes

GAIA-1 can generate videos from text prompts only, completely imagining the scene. To demonstrate this we showcase how we can generate driving scenarios from text prompts that guide the model towards specific weather or lighting conditions in Figure 12.

Next, we present compelling examples where the model exhibits fine-grained control over the vehicle dynamics in the video. By leveraging this control, we can prompt the model to generate videos depicting scenarios that lie outside the bounds of the training data. This shows that GAIA-1 is able to disentangle the ego-vehicle dynamics from the surrounding environment and effectively generalize to unfamiliar scenarios. It provides explicit ability to reason about the impact of our actions on the

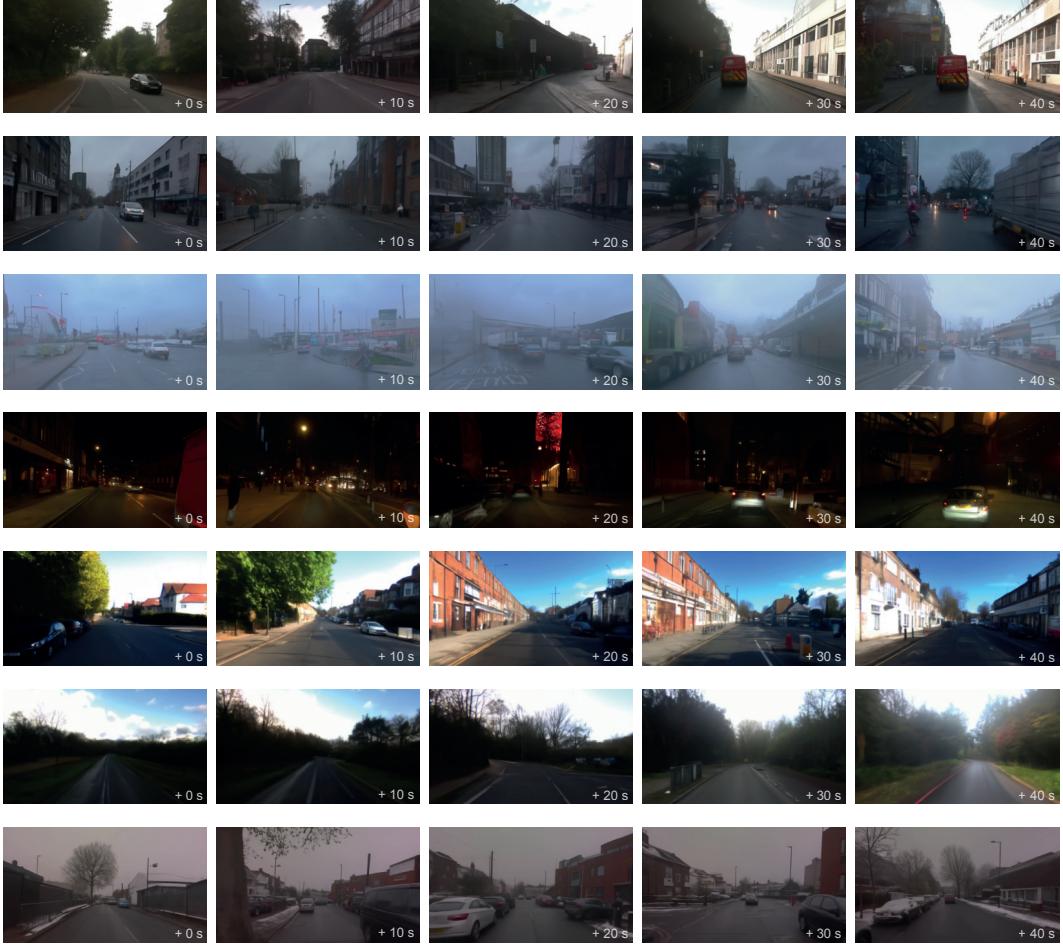


Figure 10: Long, diverse driving scenarios generated entirely from imagination by the world model.

environment (safety), it allows richer understanding of dynamic scenes (intelligence), it unlocks model-based policy learning (planning in the world model), and it enables exploration in closed-loop (by considering the world model as a neural simulator). To showcase this, we make GAIA-1 generate futures where the ego-vehicle steers left or right, deviating from its lane (Figure 13). GAIA-1 would never have seen these incorrect behaviors in the expert driving dataset used to train it, indicating that it can extrapolate driving concepts previously unseen in the training data. We also see realistic reactions of other agents to the ego-vehicle’s controlled behavior.

Finally we demonstrate the ability of GAIA-1 to leverage both text and action to fully imagine a driving scenario. In this particular case we prompt the model to generate a bus in front of the ego-vehicle and then we force its actions to overtake the bus (see Figure 1).

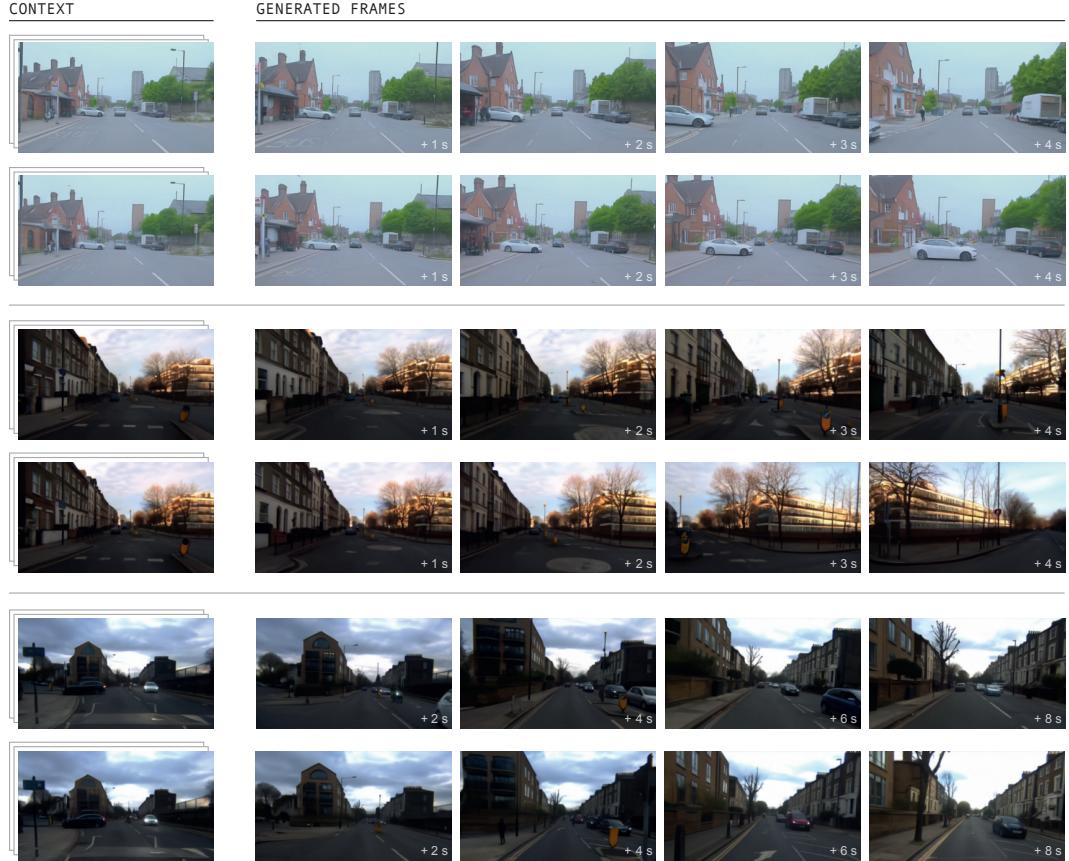
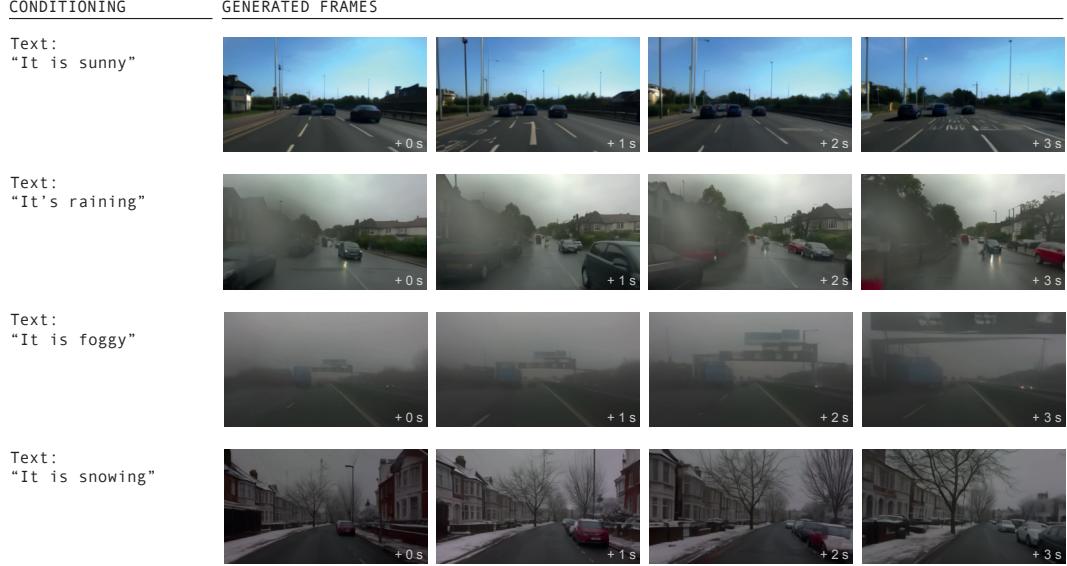
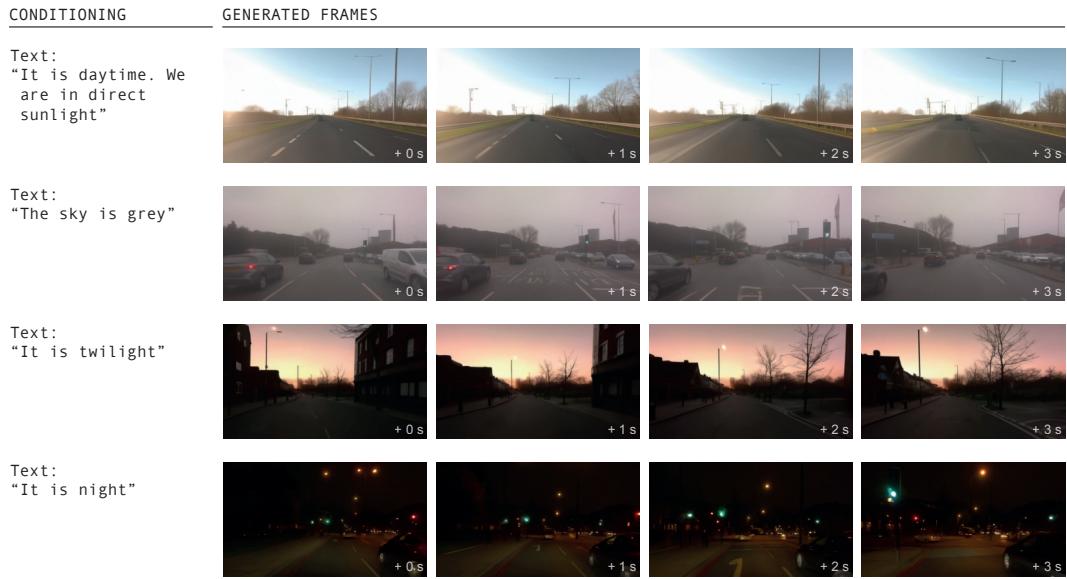


Figure 11: Examples of multiple plausible futures predicted by the world model from a given video context. 1) We observe a complex giving way interaction between the white vehicle and the ego-vehicle. In the first future, the white vehicle reverses to give way to the ego-vehicle. In the second future, the opposite occurs and the ego-vehicle slows down to give way to the white vehicle. 2) We see two plausible ego-behaviors: going straight or turning right at the roundabout. 3) The model predicts two futures with varying traffic levels.



(a) Weather.



(b) Illumination.

Figure 12: Generation from a text prompt, showing that the world model has learned different concepts such as weather or illumination.

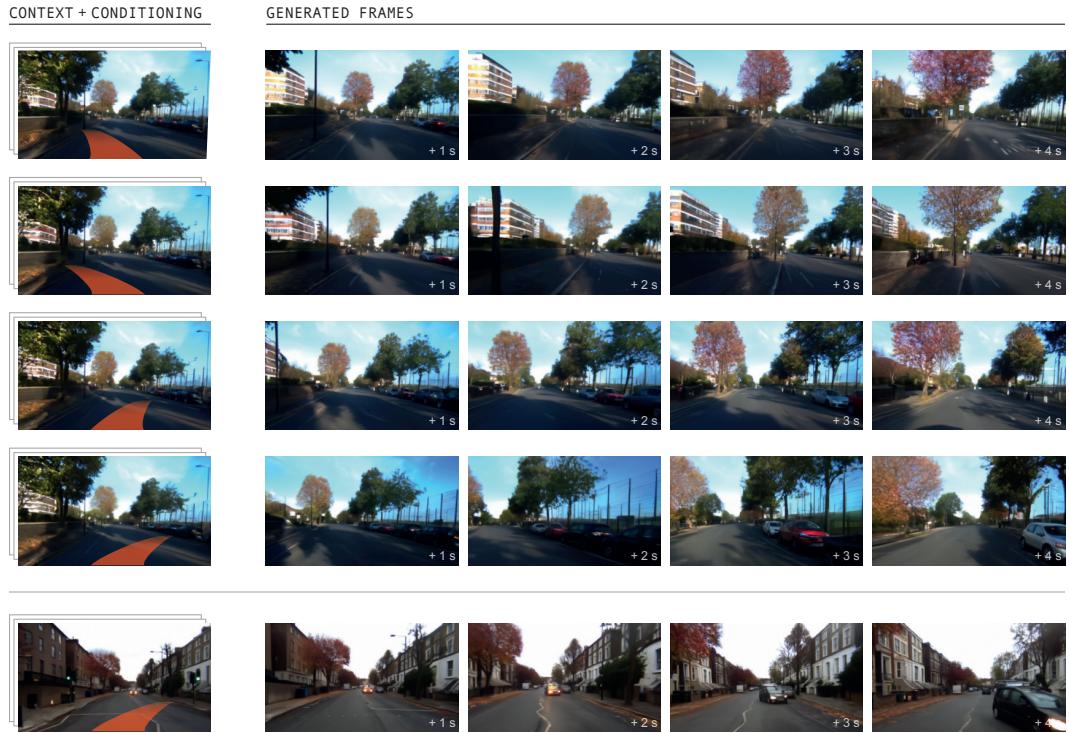


Figure 13: The world model can predict different outcomes conditioned on its actions. In the first four rows, we execute different out-of-distribution actions (left, strong left, right, strong right — while maintaining speed) from a given context video. The world model can predict the corresponding states with accurate geometry. In the last row, we force the ego-vehicle to steer right while maintaining speed, and as a consequence, we observe the oncoming vehicle reacting and making a maneuver to avoid a collision.

8 Related Work

Video generative models. Video generative models are neural networks that can generate realistic video samples. They can be grouped in four categories: VAE-based (variational autoencoder [50]), GAN-based (generative adversarial network [51]), diffusion-based [52], and autoregressive-based [53].

Latent-variable video models (VAE-based) try to infer the underlying latent process that generated the videos [54, 55, 56, 57, 58]. One known limitation of those models is that they tend to generate blurry outputs due to limited representational power, inadequate choice of prior distribution, and the optimization of a lower-bound instead of the true likelihood. GAN-based methods produce more realistic videos [59, 60, 61, 62, 63, 64] but are known to suffer from training instability and a lack of generation diversity [65]. Diffusion-based methods have yielded significant enhancements in realism, controllability, and temporal consistency. They can operate either at the pixel level [38, 17, 66, 67, 68, 69, 16] or in the latent space of a pre-trained image tokenizer [70, 18, 71]. Diffusion models are expressive neural networks that can fit complex data distributions, but rely on a long Markov chain of diffusion steps to generate samples. Lastly, autoregressive-based methods are conceptually simple and rely on tractable exact likelihood optimization (fits the entire data distribution). Likewise, they can operate at the pixel level [72, 73], or in a discrete learned token space [74, 75, 76, 77]. A known limitation is the slow generation speed, but this issue could be alleviated by future research on parallel sampling [78, 79, 80], reducing the number of latent variables [81], and improvements in hardware accelerators.

World models. A world model is a predictive model of the future that learns a general representation of the world in order to understand the consequences of its actions [7, 8]. The main use cases are: pure representation learning, planning (look-ahead search), or learning a policy in the world model (neural simulator).

World modeling has been used as a pre-training task to learn a compact and general representation in a self-supervised way [82, 83]. Subsequently, using this representation as a state for traditional reinforcement learning (RL) algorithms significantly accelerated convergence speed. World models can also be utilized for look-ahead search, in order to plan by imagining the outcomes of future actions. They have proven to be highly effective in game environments or board games [9, 84]. Additionally, world models can be a solution to the sample efficiency issues of RL algorithms by acting as a simulator of the environment [7, 85, 86, 62, 13, 15, 87], although this assumes the world model is an accurate model of the environment.

A recent line of work suggests casting world modeling as a single sequence model, treating states, actions and rewards as simply a stream of data [10, 19, 14, 88, 12, 89]. The advantage of such a perspective is that world models can benefit from scaling properties of high-capacity sequence model architectures applied to large-scale unsupervised training [26]. This is the approach that GAIA-1 takes, leveraging those scaling properties to model complex environments such as real-world driving scenes.

Scaling. Large language models have shown clear benefits in scaling model size and data [90, 24, 26, 20, 21, 22, 23]. In particular, [49] showed predictable relationships between model/data size and loss over multiple orders of magnitude. [49] derived power laws for transformer based language models in order to optimally allocate the compute budget between the model and data size. Those laws were then refined by [27] by adapting the learning rate schedule when changing the dataset size. Another direction of research to improve the training efficiency of language models is data quality. [91] showed that the quality of the training data plays a critical role in the performance of language models in downstream tasks.

Transferring the scaling principles from large language models to the visual domain holds the potential for delivering consistent and expected performance improvements [92, 93, 43, 16, 94]. In this work, by casting the problem of world modeling as unsupervised sequence modeling, we have shown that similar scaling trends from language models also applied to world models.

9 Conclusions and Future Work

GAIA-1 is a generative world model for autonomous driving. The world model uses vector-quantized representations to turn the task of future prediction into a next token prediction task, a technique that has been successfully employed in large language models. GAIA-1 has demonstrated its capability to acquire a comprehensive understanding of the environment, distinguishing between various concepts such as cars, trucks, buses, pedestrians, cyclists, road layouts, buildings, and traffic lights — all through self-supervision. Further, GAIA-1 harnesses the capabilities of video diffusion models to generate realistic driving scenarios, thereby functioning as an advanced neural simulator. GAIA-1 is a multimodal approach that enables the control of the ego-vehicle’s actions and other scene attributes through a combination of textual and action-based instructions.

While our method demonstrated promising results that have the potential to push the boundaries of autonomous driving, it is important to acknowledge current limitations. For instance, the autoregressive generation process, while highly effective, does not yet run at real-time. Nevertheless, it is noteworthy that this process lends itself well to parallelization, allowing for the concurrent generation of multiple samples.

The significance of GAIA-1 extends beyond its generative capabilities. World models represent a crucial step towards achieving autonomous systems that can understand, predict, and adapt to the complexities of the real world. Furthermore, by incorporating world models into driving models, we can enable them to better understand their own decisions and ultimately generalize to more real-world situations. Lastly, GAIA-1 can also serve as a valuable neural simulator, allowing the generation of unlimited data, including adversarial examples, for training and validating autonomous driving systems.

Acknowledgments

This work was made possible through the expertise and generous help of many teams and people across Wayve. In particular we would like to thank: Giulio D’Ippolito, Dan Reisman, Alex Persin, Przemyslaw Mazur, Oleg Sinavski, Long Chen, Fergal Cotter, Corina Gurau, Shu Ishida, Remi Tachet, Rudi Rankin, Tilly Pielichaty, Rod Bauer, Charlie Lyons-Rothbart, Harriett-Rose Follas, Robert Weston, Becky Goldman, Sasha Harrison, Saurabh Nair, Prajwal Chidananda, Tom Newton, Benoit Hanotte, Ana-Maria Marcu, Thomas Sajot, Giacomo Gallino, Alex Garcia Mayans, Tim Geypens, Robin Tweedie, Rebecca Hills, Tim Williams-Silvera, Darren Jenner, Matt Wood, Dave Chilvers, Danny Ly, Joseph Rodrigo, Will Dias, Naomi Standard, and Theepa Balasubramaniam.

References

- [1] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah. Learning to drive in a day. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2019.
- [2] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuScenes: A multimodal dataset for autonomous driving. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [3] A. Hu, F. Cotter, N. Mohan, C. Gurau, and A. Kendall. Probabilistic Future Prediction for Video Scene Understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [4] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. Qi, Y. Zhou, Z. Yang, A. Chouard, P. Sun, J. Ngiam, V. Vasudevan, A. McCauley, J. Shlens, and D. Anguelov. Large Scale Interactive Motion Forecasting for Autonomous Driving : The Waymo Open Motion Dataset. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.
- [5] A. Hu, Z. Murez, N. Mohan, S. Dudas, J. Hawke, V. Badrinarayanan, R. Cipolla, and A. Kendall. FIERY: Future Instance Prediction in Bird's-Eye View From Surround Monocular Cameras. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 15273–15282, 2021.
- [6] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [7] D. Ha and J. Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [8] Y. LeCun. A Path Towards Autonomous Machine Intelligence. In *arXiv preprint*, 2022.
- [9] J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, T. Lillicrap, and D. Silver. Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model. In *Nature*, 2020.
- [10] M. Janner, Q. Li, and S. Levine. Offline reinforcement learning as one big sequence modeling problem. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- [11] A. Hu, G. Corrado, N. Griffiths, Z. Murez, C. Gurau, H. Yeo, A. Kendall, R. Cipolla, and J. Shotton. Model-Based Imitation Learning for Urban Driving. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [12] V. Micheli, E. Alonso, and F. Fleuret. Transformers are sample-efficient world models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.
- [13] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap. Mastering diverse domains through world models. In *arXiv preprint*, 2023.
- [14] S. Reed, K. Zolna, E. Parisotto, S. Gómez, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. Freitas. A generalist agent. In *Transactions on Machine Learning Research (TMLR)*, 2022.
- [15] P. Wu, A. Escontrela, D. Hafner, P. Abbeel, and K. Goldberg. Daydreamer: World models for physical robot learning. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2023.
- [16] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, and T. Salimans. Imagen video: High definition video generation with diffusion models. In *arXiv preprint*, 2022.
- [17] W. Harvey, S. Naderiparizi, V. Masrani, C. Weilbach, and F. Wood. Flexible diffusion modeling of long videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

- [18] P. Esser, J. Chiu, P. Atighehchian, J. Granskog, and A. Germanidis. Structure and content-guided video synthesis with diffusion models. In *arXiv preprint*, 2023.
- [19] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- [20] S. Smith, M. M. A. Patwary, B. Norick, P. Legresley, S. Rajbhandari, J. Casper, Z. Liu, S. Prabhumoye, G. Zerveas, V. Korthikanti, E. Zhang, R. Child, R. Aminabadi, J. Bernauer, X. Song, M. Shoeybi, Y. He, M. Houston, S. Tiwary, and B. Catanzaro. Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, A Large-Scale Generative Language Model. In *arXiv preprint*, 2022.
- [21] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. M. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. C. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. García, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Díaz, O. Firat, M. Catasta, J. Wei, K. S. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel. PaLM: Scaling language modeling with pathways. *arXiv preprint*, 2022.
- [22] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambo, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models. In *arXiv preprint*, 2023.
- [23] OpenAI. GPT-4 Technical Report. In *arXiv preprint*, 2023.
- [24] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 2020.
- [25] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint*, 2019.
- [26] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- [27] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, O. Vinyals, J. W. Rae, and L. Sifre. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [28] A. van den Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [29] Z. Peng, L. Dong, H. Bao, Q. Ye, and F. Wei. BEiT v2: Masked Image Modeling with Vector-Quantized Visual Tokenizers. *arXiv preprint*, 2022.
- [30] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.
- [31] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

- [32] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.
- [33] P. Esser, R. Rombach, and B. Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [34] J. Yu, X. Li, J. Y. Koh, H. Zhang, R. Pang, J. Qin, A. Ku, Y. Xu, J. Baldridge, and Y. Wu. Vector-quantized image modeling with improved VQGAN. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.
- [35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [36] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [37] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [38] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet. Video diffusion models. In *arXiv preprint*, 2022.
- [39] J. H. Tim Salimans. Progressive distillation for fast sampling of diffusion models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.
- [40] I. Loshchilov and F. Hutter. Decoupled Weight Decay Regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.
- [41] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [42] J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He. DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020.
- [43] E. Hoogeboom, J. Heek, and T. Salimans. simple diffusion: End-to-end diffusion for high resolution images. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.
- [44] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. The curious case of neural text degeneration. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- [45] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint*, 2022.
- [46] H. Chang, H. Zhang, J. Barber, A. Maschinot, J. Lezama, L. Jiang, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint*, 2023.
- [47] Negative prompt. <https://github.com/AUTOMATIC1111/stable-diffusion-webui/wiki/Negative-prompt>, 2022.
- [48] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.
- [49] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei. Scaling laws for neural language models. In *arXiv preprint*, 2020.

- [50] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.
- [51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.
- [52] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2015.
- [53] A. van den Oord, N. Kalchbrenner, L. Espeholt, k. kavukcuoglu, O. Vinyals, and A. Graves. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.
- [54] M. Babaeizadeh, C. Finn, D. Erhan, R. H. Campbell, and S. Levine. Stochastic variational video prediction. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.
- [55] E. Denton and R. Fergus. Stochastic Video Generation with a Learned Prior. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.
- [56] R. Villegas, A. Pathak, H. Kannan, D. Erhan, Q. Le, and H. Lee. High fidelity video prediction with large stochastic recurrent neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- [57] J.-Y. Franceschi, E. Delasalles, M. Chen, S. Lamprier, and P. Gallinari. Stochastic latent residual video prediction. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.
- [58] M. Babaeizadeh, M. Saffar, S. Nair, S. Levine, C. Finn, and D. Erhan. Fitvid: Overfitting in pixel-level video prediction. In *arXiv preprint*, 2021.
- [59] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.
- [60] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz. MoCoGAN: Decomposing motion and content for video generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [61] A. Clark, J. Donahue, and K. Simonyan. Adversarial Video Generation on Complex Datasets. In *arXiv preprint*, 2019.
- [62] S. W. Kim, J. Philion, A. Torralba, and S. Fidler. DriveGAN: Towards a controllable high-quality neural simulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [63] I. Skorokhodov, S. Tulyakov, and M. Elhoseiny. StyleGAN-V: A continuous video generator with the price, image quality and perks of StyleGAN2. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [64] T. Brooks, J. Hellsten, M. Aittala, T.-C. Wang, T. Aila, J. Lehtinen, M.-Y. Liu, A. Efros, and T. Karras. Generating long videos of dynamic scenes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [65] I. Goodfellow. NIPS 2016 Tutorial: Generative Adversarial Networks. In *arXiv preprint*, 2016.
- [66] V. Voleti, A. Jolicoeur-Martineau, and C. Pal. MCVD: Masked conditional video diffusion for prediction, generation, and interpolation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [67] T. Höppe, A. Mehrjou, S. Bauer, D. Nielsen, and A. Dittadi. Diffusion models for video prediction and infilling. In *Transactions on Machine Learning Research (TMLR)*, 2022.

- [68] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni, D. Parikh, S. Gupta, and Y. Taigman. Make-A-Video: Text-to-video generation without text-video data. In *arXiv preprint*, 2022.
- [69] E. Molad, E. Horwitz, D. Valevski, A. R. Acha, Y. Matias, Y. Pritch, Y. Leviathan, and Y. Hoshen. Dreamix: Video diffusion models are general video editors. *arXiv preprint*, 2023.
- [70] D. Zhou, W. Wang, H. Yan, W. Lv, Y. Zhu, and J. Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint*, 2022.
- [71] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [72] N. Kalchbrenner, A. van den Oord, K. Simonyan, I. Danihelka, O. Vinyals, A. Graves, and K. Kavukcuoglu. Video Pixel Networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.
- [73] D. Weissenborn, O. Täckström, and J. Uszkoreit. Scaling autoregressive video models. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- [74] W. Yan, Y. Zhang, P. Abbeel, and A. Srinivas. VideoGPT: Video generation using vq-vae and transformers. In *arXiv preprint*, 2021.
- [75] G. L. Moing, J. Ponce, and C. Schmid. CCVS: Context-aware controllable video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- [76] S. Ge, T. Hayes, H. Yang, X. Yin, G. Pang, D. Jacobs, J.-B. Huang, and D. Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.
- [77] Y. Seo, K. Lee, F. Liu, S. James, and P. Abbeel. HARP: Autoregressive latent video prediction with high-fidelity image generator. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2022.
- [78] W. Yan, D. Hafner, S. James, and P. Abbeel. Temporally consistent transformers for video generation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.
- [79] R. Villegas, M. Babaeizadeh, P.-J. Kindermans, H. Moraldo, H. Zhang, M. T. Saffar, S. Castro, J. Kunze, and D. Erhan. Phenaki: Variable length video generation from open domain textual description. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.
- [80] L. Yu, Y. Cheng, K. Sohn, J. Lezama, H. Zhang, H. Chang, A. G. Hauptmann, M.-H. Yang, Y. Hao, I. Essa, and L. Jiang. MAGVIT: Masked Generative Video Transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [81] C. Hawthorne, A. Jaegle, C. Cangea, S. Borgeaud, C. Nash, M. Malinowski, S. Dieleman, O. Vinyals, M. Botvinick, I. Simon, H. Sheahan, N. Zeghidour, J.-B. Alayrac, J. Carreira, and J. Engel. General-purpose, long-context autoregressive modeling with Perceiver AR. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2022.
- [82] M. Schwarzer, A. Anand, R. Goel, R. D. Hjelm, A. C. Courville, and P. Bachman. Data-efficient reinforcement learning with self-predictive representations. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- [83] P. Wu, A. Majumdar, K. Stone, Y. Lin, I. Mordatch, P. Abbeel, and A. Rajeswaran. Masked trajectory models for prediction, representation, and control. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.
- [84] W. Ye, S. Liu, T. Kurutach, P. Abbeel, and Y. Gao. Mastering atari games with limited data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

- [85] L. Kaiser, M. Babaeizadeh, P. Milos, B. Osinski, R. Campbell, K. Czechowski, D. Erhan, C. Finn, P. Kozakowski, S. Levine, A. Mohiuddin, R. Sepassi, G. Tucker, and H. Michalewski. Model-based reinforcement learning for atari. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- [86] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba. Mastering atari with discrete world models. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.
- [87] X. Wang, Z. Zhu, G. Huang, X. Chen, and J. Lu. DriveDreamer: Towards Real-world-driven World Models for Autonomous Driving. *arXiv preprint*, 2023.
- [88] F. Liu, H. Liu, A. Grover, and P. Abbeel. Masked autoencoding for scalable and generalizable decision making. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [89] commaVQ. <https://github.com/commaai/commavq>, 2023.
- [90] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.
- [91] N. Du, Y. Huang, A. M. Dai, S. Tong, D. Lepikhin, Y. Xu, M. Krikun, Y. Zhou, A. W. Yu, O. Firat, B. Zoph, L. Fedus, M. Bosma, Z. Zhou, T. Wang, Y. E. Wang, K. Webster, M. Pellat, K. Robinson, K. S. Meier-Hellstern, T. Duke, L. Dixon, K. Zhang, Q. V. Le, Y. Wu, Z. Chen, and C. Cui. GLaM: Efficient scaling of language models with mixture-of-experts. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2022.
- [92] Y. Tay, M. Dehghani, J. Rao, W. Fedus, S. Abnar, H. W. Chung, S. Narang, D. Yogatama, A. Vaswani, and D. Metzler. Scale efficiently: Insights from pre-training and fine-tuning transformers. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.
- [93] M. Dehghani, J. Djolonga, B. Mustafa, P. Padlewski, J. Heek, J. Gilmer, A. Steiner, M. Caron, R. Geirhos, I. Alabdulmohsin, R. Jenatton, L. Beyer, M. Tschannen, A. Arnab, X. Wang, C. Riquelme, M. Minderer, J. Puigcerver, U. Evci, M. Kumar, S. van Steenkiste, G. F. Elsayed, A. Mahendran, F. Yu, A. Oliver, F. Huot, J. Bastings, M. P. Collier, A. Gritsenko, V. Birodkar, C. Vasconcelos, Y. Tay, T. Mensink, A. Kolesnikov, F. Pavetić, D. Tran, T. Kipf, M. Lučić, X. Zhai, D. Keysers, J. Harmsen, and N. Houlsby. Scaling vision transformers to 22 billion parameters. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.
- [94] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *arXiv preprint*, 2023.