EMU VIDEO:
Factorizing Text-to-Video Generation by Explicit Image Conditioning

Rohit Girdhar[†,*] Mannat Singh[†,*] Andrew Brown[*] Quentin Duval[*] Samaneh Azadi[*]
Sai Saketh Rambhatla Akbar Shah Xi Yin Devi Parikh Ishan Misra[*]
GenAI, Meta

https://emu-video.metademolab.com/

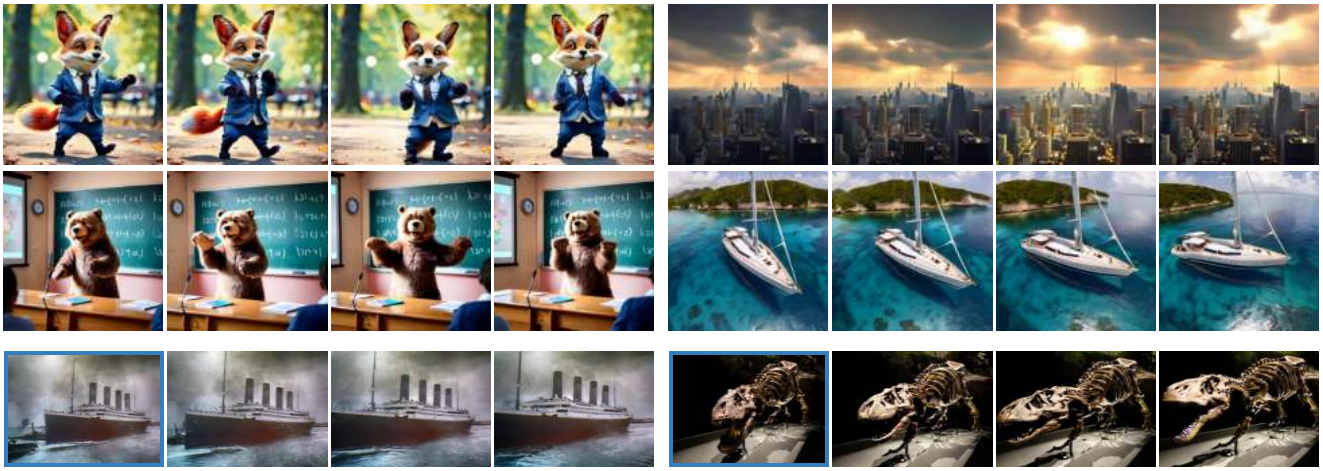


Figure 1. EMU VIDEO can generate high quality and temporally consistent videos while using a text prompt as input (top two rows), or an additional user-provided image (bottom row). Prompts: (top-left) A fox dressed in a suit dancing in a park, (top-right) The sun breaks through the clouds from the heights of a skyscraper, (middle-left): A bear is giving a presentation in the classroom, (middle-right): A 360 shot of a sleek yacht sailing gracefully through the crystal-clear waters of the Caribbean, (bottom-left): A ship driving off the harbor, (bottom-right): The dinosaur slowly comes to life. In the bottom two examples, a user-image is provided as an additional conditioning (shown in a blue border) and brought to life by EMU VIDEO. The first one is a historical picture of the RMS Titanic departing from Belfast, Northern Ireland; and the second is a picture of a Tyrannosaurus rex fossil.

Abstract

We present EMU VIDEO*, a text-to-video generation model that factorizes the generation into two steps: first generating an image conditioned on the text, and then generating a video conditioned on the text and the generated image. We identify critical design decisions–adjusted noise schedules for diffusion, and multi-stage training–that enable us to directly generate high quality and high resolution videos, without requiring a deep cascade of models as in prior work. In human evaluations, our generated videos are strongly preferred in quality compared to all prior work–* 81% *vs. Google's Imagen Video,* 90% *vs. Nvidia's PYOCO, and* 96% *vs. Meta's Make-A-Video. Our model outperforms commercial solutions such as RunwayML's Gen2 and Pika Labs. Finally, our factorizing approach naturally lends itself to animating images based on a user's text prompt, where our generations are preferred* 96% *over prior work.*

1. Introduction

Large text-to-image models [17, 21, 28, 38, 55, 62] trained on web-scale image-text pairs generate diverse and high quality images. While these models can be further adapted for text-to-video (T2V) generation [7, 30, 38, 41, 68] by using video-text pairs, video generation still lags behind image generation in terms of quality and diversity. Compared to image generation, video generation is more challenging as it requires modeling a higher dimensional spatiotemporal output space while still being conditioned only on a text prompt. Moreover, video-text datasets are typically an order of magnitude smaller than image-text datasets [17, 38, 68].

The dominant paradigm in video generation uses diffusion models [38, 68] to generate all video frames at once.

[†]Equal first authors [*]Equal technical contribution

In stark contrast, in NLP, long sequence generation is formulated as an autoregressive problem [11]: predicting one word conditioned on previously predicted words. Thus, the conditioning signal for each subsequent prediction progressively gets stronger. We hypothesize that strengthening the conditioning signal is also important for high quality video generation, which is inherently a time-series. However, autoregressive decoding with diffusion models is challenging since generating a single frame from such models itself requires many iterations.

We propose EMU VIDEO to strengthen the conditioning for diffusion based text-to-video generation with an explicit intermediate image generation step. Specifically, we factorize text-to-video generation into two subproblems: (1) generating an image from an input text prompt; (2) generating a video based on the stronger conditioning from the image *and* the text. Intuitively, giving the model a starting image and text makes video generation easier since the model only needs to predict how the image will evolve in the future.

Since video-text datasets are much smaller than image-text datasets, we also initialize [7, 68] our factorized text-to-video model using a pretrained text-to-image (T2I) model whose weights are kept frozen. We identify critical design decisions–changes to the diffusion noise schedule and multi-stage training–to directly generate videos at a high resolution of 512px. Unlike direct T2V methods [38, 68], at inference, our factorized approach explicitly generates an image, which allows us to easily retain the visual diversity, style, and quality of the text-to-image model (examples in Figure 1). This allows EMU VIDEO to outperform direct T2V methods, even when accounting for the same amount of training data, compute, and trainable parameters.

Contributions. We show that text-to-video (T2V) generation quality can be greatly improved by factorizing the generation into first generating an image and using the generated image and text to generate a video. Our multi-stage training enables us to directly generate videos at a high resolution of 512px, bypassing the need for a deep cascade of models used in prior work [38, 68]. We design a robust human evaluation scheme–JUICE–where we ask evaluators to justify their choices when making the selection in the pairwise comparisons. As shown in Figure 2, EMU VIDEO significantly *surpasses all prior work* including commercial solutions with an average win rate of 91.8% for quality and 86.6% for text faithfulness. Beyond T2V, EMU VIDEO can be used out-of-the-box for image-to-video where the model generates a video based on a user-supplied image and a text prompt. In this setting, EMU VIDEO's generations are preferred 96% of the times over VideoComposer [76].

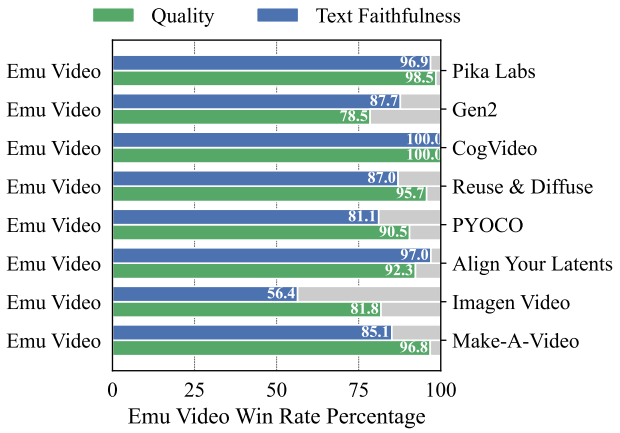


Figure 2. EMU VIDEO *vs.* prior work in text-to-video in terms of video quality and text faithfulness win-rates evaluated by majority score of human evaluator preferences. Since most models from prior work are not accessible, we use the videos released by each method and their associated text prompt. The released videos are likely the *best* generations and we compare without any cherry-picking of our own generations. We also compare to commercial solutions (Gen2 [54] and PikaLabs [47]) and the open source model CogVideo [41] using the prompt set from [7]. EMU VIDEO significantly outperforms all prior work across both metrics.

2. Related Work

Text-to-Image (T2I) diffusion models. Diffusion models [69] are a state-of-the-art approach for T2I generation, and out-perform prior GAN [8, 43, 66] or auto-regressive methods [1, 22, 29, 59]. Diffusion models learn a data distribution by gradually denoising a normally distributed variable, often called 'noise', to generate the output. Prior work either denoises in the pixel space with pixel diffusion models [19, 36, 37, 56, 60, 63], or in a lower-dimensional latent space with latent diffusion models [17, 62]. In this work, we leverage latent diffusion models for video generation.

Video generation/prediction. Many prior works target the constrained settings of unconditional generation, or video prediction [45, 46, 53]. These approaches include training VAEs [4, 5, 18], auto-regressive models [25, 41, 42, 61, 81], masked prediction [27, 32, 86], LSTMs [67, 77], or GANs [2, 9, 16, 75]. However, these approaches are trained/evaluated on limited domains. In this work, we target the broad task of open-set T2V generation.

Text-to-Video (T2V) generation. Most prior works tackle T2V generation by leveraging T2I models. Several works take a training-free approach [40, 44, 49, 87] for *zero-shot* T2V generation by injecting motion information in the T2I models. Tune-A-Video [79] targets *one-shot* T2V generation by fine-tuning a T2I model with a single video. While these methods require no or limited training, the quality and diversity of the generated videos is limited.

Many prior works instead improve T2V generation by learning a *direct mapping* from the text condition to the generated videos by introducing temporal parameters to a T2I model [7, 30, 33, 39, 41, 48, 72, 74, 78, 82, 84]. Make-

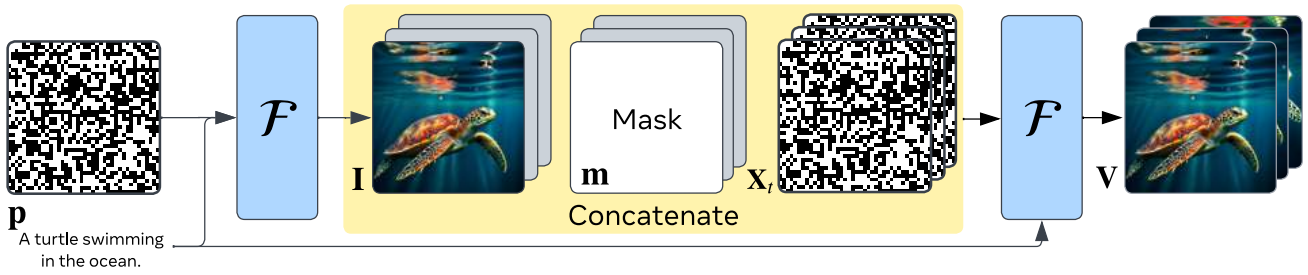


Figure 3. Factorized text-to-video generation involves first generating an image $\mathbf{I}$ conditioned on the text $\mathbf{p}$, and then using stronger conditioning–the generated image *and* text–to generate a video $\mathbf{V}$. To condition our model $\mathcal{F}$ on the image, we zero-pad the image temporally and concatenate it with a binary mask indicating which frames are zero-padded, and the noised input.

A-Video [68] utilizes a pre-trained T2I model [60] and the prior network of [60] to train T2V generation without paired video-text data. Imagen Video [38] builds upon the Imagen T2I model [63] with a cascade of diffusion models [37, 39]. To address the challenges of modeling the high-dimensional spatiotemporal space, several works instead train T2V diffusion models in a lower-dimensional latent space [3, 7, 24, 30, 31, 34, 80], by adapting latent diffusion T2I models. Blattmann *et al*. [7] freeze the parameters of a pre-trained T2I model and train new temporal layers, whilst Ge *et al*. [30] build on [7] and design a noise prior tailored for T2V generation. The limitation of these approaches is that learning a direct mapping from text to the high dimensional video space is challenging. We instead strengthen our conditioning signal by taking a factorization approach. Unlike prior work that enhancing the conditions for T2V generation including leveraging large language models (LLMs) to improve textual description and understanding [24, 40, 50], or adding temporal information as conditions [13, 76, 83, 87], our method does not require any models to generate the conditions as we use the first frame of a video as the image condition.

Factorized generation. The most similar works to EMU VIDEO, in terms of factorization, is CogVideo [41] and Make-A-Video [68]. CogVideo builds upon the pretrained T2I model [20] for T2V generation using auto-regressive Transformer. The auto-regressive nature is fundamentally different to our explicit image conditioning in both training and inference stages. Make-A-Video [68] leverages the image embedding condition learnt from a shared image-text space. Our factorization leverage the first frame as is, which is a stronger condition. Moreover, Make-A-Video initializes from a pretrained T2I model but finetunes all the parameters so it cannot retain the visual quality and diversity of the T2I model as we do.

3. Approach

The goal of text-to-video (T2V) generation is to construct a model f that takes as input a text prompt $\mathbf{p}$ to generate a video $\mathbf{V}$ consisting of T RGB frames. Recent methods [7, 30, 38, 68] directly generate the T video frames at once using text-only conditioning. Our approach builds on the hypothesis that stronger conditioning by way of both text *and* image can improve video generation (*cf*. § 3.2).

3.1. Preliminaries

Conditional Diffusion Models [36, 69] are a class of generative models that are trained to generate the output using a conditional input $\mathbf{c}$ by iteratively denoising from gaussian noise. At training time, time-step $t \in [0, N]$ dependent gaussian noise $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ is added to the original input signal $\mathbf{X}$ to obtain a noisy input $\mathbf{X}_t = \alpha_t \mathbf{X} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t$. α_t defines the "noise schedule", *i.e*., noise added at timestep t and N is the total number of diffusion steps. The diffusion model is trained to denoise $\mathbf{X}_t$ by predicting either $\boldsymbol{\epsilon}_t$, $\mathbf{X}$, or $v_t = \alpha_t \boldsymbol{\epsilon}_t - \sqrt{1-\alpha_t}\mathbf{X}$ (called v-prediction [64]). The signal-to-noise ratio (SNR) at timestep t is given by $(\frac{\alpha_t}{1-\alpha_t})^2$ and decreases as $t \to N$. At inference, samples are generated by starting from pure noise $\mathbf{X}_N \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ and denoising it. Note that at inference time $\mathbf{X}_N$ has no signal, *i.e*., zero SNR which has significant implications for video generation as we describe in § 3.2.

3.2. EMU VIDEO

We factorize text-to-video generation into two steps (1) generating the first frame (image) given the text prompt $\mathbf{p}$ and (2) generating T frames of a video by leveraging the text prompt and the image conditioning. We implement both steps using a latent diffusion model $\mathcal{F}$, illustrated in Figure 3. We initialize $\mathcal{F}$ with a pre-trained text-to-image model to ensure that it is capable of generating images at initialization. Thus, we only need to train $\mathcal{F}$ to solve the second step, *i.e*., extrapolate a video conditioned on a text prompt and a starting frame. We train $\mathcal{F}$ using video-text pairs by sampling a starting frame $\mathbf{I}$ and asking the model to predict the T frames using both the text prompt $\mathbf{p}$xw and the image $\mathbf{I}$ conditioning. We denote a video $\mathbf{V}$ consisting of T RGB frames of spatial dimensions H', W' as a

4D tensor of shape $T \times 3 \times H' \times W'$. Since we use latent diffusion models, we first convert the video $\mathbf{V}$ into a latent space $\mathbf{X} \in \mathbb{R}^{T \times C \times H \times W}$ using a image autoencoder applied frame-wise, which reduces the spatial dimensions. The latent space can be converted back to the pixel space using the autoencoder's decoder. The T frames of the video are noised independently to produce the noised input $\mathbf{X}_t$, which the diffusion model is trained to denoise.

Image conditioning. We condition on the starting frame, $\mathbf{I}$, by concatenating it with the noise. Our design choice allows the model to use all the information in $\mathbf{I}$ unlike other choices [68, 76] that lose image information by using a semantic image embedding for conditioning. We represent $\mathbf{I}$ as a single-frame video, *i.e.*, $T = 1$ and zero-pad it to obtain a $T \times C \times H \times W$ tensor. We use a binary mask $\mathbf{m}$ of shape $T \times 1 \times H \times W$ that is set to 1 at the first temporal position to indicate the position of the starting frame, and zero otherwise. The mask $\mathbf{m}$, starting frame $\mathbf{I}$, and the noised video $\mathbf{X}_t$ are concatenated channel-wise as the input to the model.

Model. We initialize our latent diffusion model $\mathcal{F}$ using the pretrained T2I model [17]. Like prior work [68], we add new learnable temporal parameters: a 1D temporal convolution after every spatial convolution, and a 1D temporal attention layer after every spatial attention layer. The original spatial convolution and attention layers are applied to each of the T frames independently and are kept frozen. The pretrained T2I model is already text conditioned and combined with the image conditioning described above, $\mathcal{F}$ is conditioned on both text and image.

Zero terminal-SNR noise schedule. We found that the diffusion noise schedules used in prior work [17, 62] have a train-test discrepancy which prevents high quality video generation (reported for images in [12, 51]). At training, the noise schedule leaves some residual signal, *i.e.*, has non-zero signal-to-noise (SNR) ratio even at the terminal diffusion timestep N. This prevents the diffusion model from generalizing at test time when we sample from random gaussian noise with no signal about real data. The residual signal is higher for high resolution video frames, due to redundant pixels across both space and time. We resolve this issue by scaling the noise schedule and setting the final $\alpha_N = 0$ [51], which leads to zero SNR at the terminal timestep N during training too. We find that this design decision is *critical* for high resolution video generation.

Interpolation model. We use an interpolation model $\mathcal{I}$, architecturally the same as $\mathcal{F}$, to convert a low frame-rate video of T frames into a high frame-rate video of T_p frames. The interpolation model operates on $T_p \times C \times H \times W$ inputs/outputs. For frame conditioning, the input T frames are zero-interleaved to produce T_p frames, and a binary mask $\mathbf{m}$ indicating the presence of the T frames are concatenated to the noised input (similar to the image conditioning for $\mathcal{F}$). The model is trained on video clips of T_p frames of which T frames are fed as input. For efficiency, we initialize $\mathcal{I}$ from $\mathcal{F}$ and only train the temporal parameters of the model $\mathcal{I}$ for the interpolation task.

Simplicity in implementation. EMU VIDEO can be trained using standard video-text datasets, and does not require a deep cascade of models, *e.g.*, 7 models in [38], for generating high resolution videos. At inference, given a text prompt, we run $\mathcal{F}$ without the temporal layers to generate an image $\mathbf{I}$. We then use $\mathbf{I}$ and the text prompt as input to $\mathcal{F}$ to generate T video frames, directly at high resolution. We can increase the fps of the video using $\mathcal{I}$. Since the spatial layers are initialized from a pretrained T2I model and kept frozen, our model retains the conceptual and stylistic diversity learned from large image-text datasets, and uses it to generate $\mathbf{I}$. This comes at no additional training cost unlike approaches [38] that do joint finetuning on image and video data to maintain such style. Many direct T2V approaches [7, 68] also initialize from a pretrained T2I model and keep the spatial layers frozen. However, they do not employ our image-based factorization and thus do not retain the quality and diversity in the T2I model.

Robust human evaluation (JUICE). Similar to recent studies [17, 38, 57, 68], we find that the automatic evaluation metrics [73] do not reflect improvements in quality. We primarily use human evaluation to measure T2V generation performance on two orthogonal aspects - (a) video generation quality denoted as Quality (Q) and (b) the alignment or 'faithfulness' of the generated video to the text prompt, denoted as Faithfulness (F). We found that asking human evaluators to JUstify their choICE (JUICE) when picking a generation over the other significantly improves the inter-annotator agreement (details in Appendix C). The annotators select one or more pre-defined reasons to justify their choice. The reasons for picking one generation over the other for Quality are: pixel sharpness, motion smoothness, recognizable objects/scenes, frame consistency, and amount of motion. For Faithfulness we use two reasons: spatial text alignment, and temporal text alignment.

3.3. Implementation Details

We provide complete implementation details in the supplement Appendix A and highlight salient details next.

Architecture and initialization. We adapt the text-to-image U-Net architecture from [17] for our model and initialize all the spatial parameters with the pretrained model. The pretrained model produces square 512px images using an 8 channel 64×64 latent as the autoencoder downsamples spatially by $8\times$. The model uses both a frozen T5-XL [15] and a frozen CLIP [58] text encoder to extract features from the text prompt. Separate cross-attention layers in the U-Net attend to each of the text features. After initialization, our model contains 2.7B spatial parameters which are kept frozen, and 1.7B temporal parameters that are learned.

The temporal parameters are initialized as identity operations: identity kernels for convolution, and zeroing the final MLP layer of the temporal attention block. In our preliminary experiments, the identity initialization improved the model convergence by $2\times$. For the additional channels in the model input due to image conditioning, we add $C + 1$ additional learnable channels (zero-initialized) to the kernel of the first spatial convolution layer. Our model produces 512px square videos of $T = 8$ or 16 frames and is trained with square center-cropped video clips of 1, 2 or 4 seconds sampled at 8fps or 4fps. We train all our models with a batch size of 512 and describe the details next.

Efficient multi-stage multi-resolution training. To reduce the computational complexity, we train in two stages - (1) for majority of the training iterations (70K) we train for a simpler task: 256px 8fps 1s videos, which reduces per-iteration time by $3.5\times$ due to the reduction in spatial resolution; (2) we then train the model at the desired 512px resolution on 4fps 2s videos for 15K iterations. The change in spatial resolution does not affect the 1D temporal layers. Although the frozen spatial layers were pretrained at 512px, changing the spatial resolution at inference to 256px led to no loss in generation quality. We use the noise schedule from [62] for 256px training, and with zero terminal-SNR for 512px training using the v-prediction objective [64] with $N = 1000$ steps for the diffusion training. We sample from our models using 250 steps of DDIM [70]. Optionally, to increase duration, we further train the model on 16 frames from a 4s video clip for 25K iterations.

Finetuning for higher quality. Similar to the observation in image generation [17], we find that the motion of the generated videos can be improved by finetuning the model on a small subset of high motion and high quality videos. We automatically identify a small finetuning subset of 1.6K videos from our training set which have high motion (computed using motion signals stored in `H.264` encoded videos). We follow standard practice [62] and also apply filtering based on aesthetic scores [62] and CLIP [58] similarity between the video's text and first frame.

Interpolation model. We initialize the interpolation model from the video model $\mathcal{F}$. Our interpolation model takes 8 frames as input and outputs $T_p = 37$ frames at 16fps. During training, we use noise augmentation [37] where we add noise to the frame conditioning by randomly sampling timesteps $t \in \{0, ...250\}$. At inference time, we noise augment the samples from $\mathcal{F}$ with $t = 100$.

4. Experiments

Dataset. We train EMU VIDEO on a dataset of 34M licensed video-text pairs. Our videos range from 5-60 seconds and cover a variety of natural world concepts. These videos were not curated for any particular task and were *not* filtered for any text-frame similarity or aesthetics. Unless noted otherwise, we train the model on the full set, and do not use the 1.6K high motion quality finetuning subset described in § 3.3.

Text prompt sets for human evaluation. We use the text prompt sets from prior work (*cf*. Appendix Table 10) to generate videos. The prompts cover a wide variety of categories that can test our model's ability to generate natural and fantastical videos, and compose different visual concepts. We use our proposed JUICE evaluation scheme (Sec. 3) for reliable human evaluation and use the majority vote from 5 evaluators for each comparison.

4.1. Ablating design decisions

We study the effects of our design decisions using the 8 frame generation setting and report human evaluation results in Table 1 using pairwise comparisons on the 307 prompt set of [68].

Factorized *vs*. Direct generation. We compare our factorized generation to a direct T2V generation model that generates videos from text condition only. We ensure that the pretrained T2I model, training data, number of training iterations, and trainable parameters are held constant for this comparison. As shown in Table 1a, the factorized generation model's results are strongly preferred both in Quality and Faithfulness.The strong preference in Quality is because the direct generation model does not retain the style and quality of the text-to-image model despite frozen spatial parameters, while also being less temporally consistent (examples in Figure 4).

Zero terminal-SNR noise schedule. We compare using zero terminal-SNR for the high resolution 512px training against a model that is trained with the standard noise schedule. Table 1b shows that generations using zero terminal-SNR are *strongly* preferred. This suggests that the zero terminal-SNR noise schedule's effect of correcting the train-test discrepancy as described in § 3.2 is critical for high resolution video generation. We also found that zero terminal-SNR has a stronger benefit for our factorized generation compared to a direct T2V model possibly. Similar to images [51], in the direct T2V case, this decision primarily affects the color composition. For our factorized approach, this design choice was critical for object consistency and high quality as our qualitative results in Figure 4 show.

Multi-stage multi-resolution training. We spend most training budget ($4\times$) on the 256px 8fps stage compared to the $3.5\times$ slower (due to increased resolution) 512px 4fps stage. We compare to a baseline that trains only the 512px stage with the same training budget. Table 1c shows that our multi-stage training yields significantly better results.

High quality finetuning. We study the effect of finetuning our model on automatically identified high quality videos in Table 1d. We found that this finetuning improves on both metrics. In particular, finetuning improves the model's abil-

Method	Q	F
Factorized	70.5	63.3

(a)

Method	Q	F
Zero SNR	96.8	88.3

(b)

Method	Q	F
Multi-stage	81.8	84.1

(c)

Method	Q	F
HQ finetuned	65.1	79.6

(d)

Method	Q	F
Frozen spatial	55.0	58.1

(e)

Table 1. Key design decisions in EMU VIDEO. Each table shows the preference, in terms of the Quality (Q) and Faithfulness (F), on adopting a design decision *vs*. a model that does not have it. Our results show clear preference to a) factorized generation that uses both image and text conditioning (against a direct video generation baseline that is only text conditioned), b) adopting zero terminal-SNR noise schedule for directly generating high resolution 512px videos, c) adopting the multi-stage training setup compared to training directly at the high resolution, d) incorporating the high quality (HQ) finetuning, and e) freezing the spatial parameters. See § 4.1 for details.

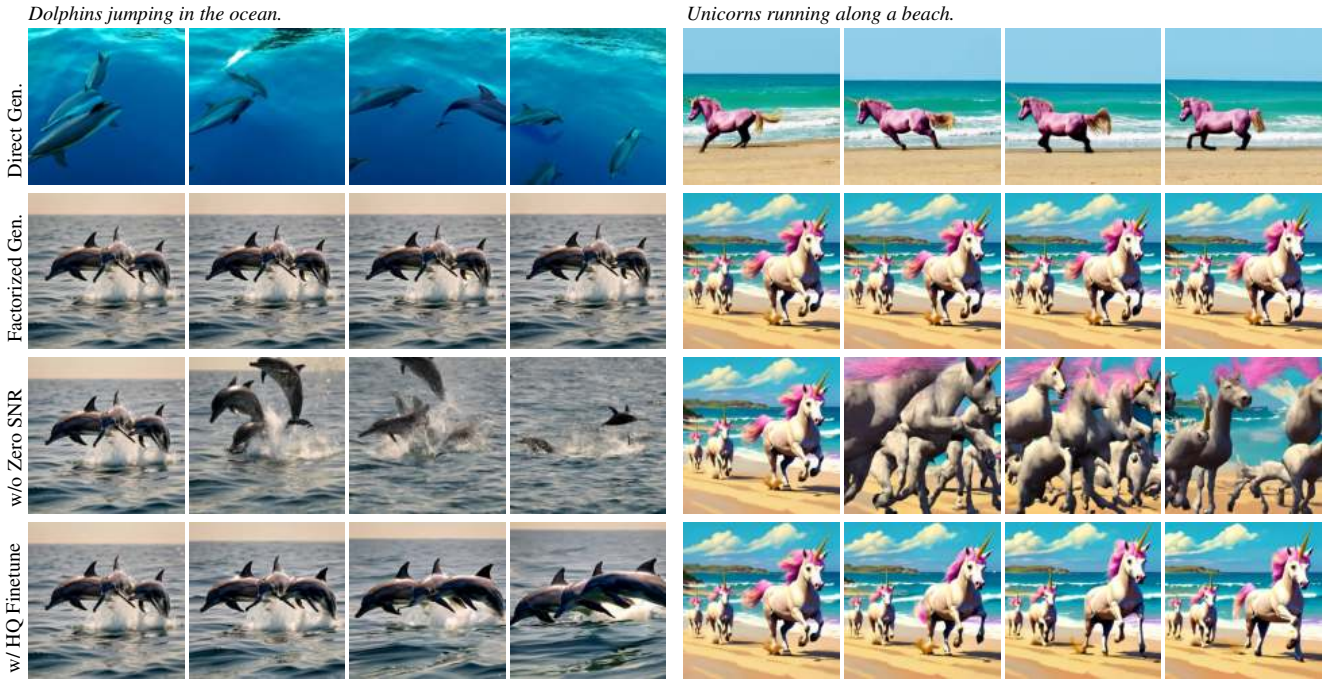


Figure 4. Design choices in EMU VIDEO. *Top row:* Direct text-to-video generation produces videos that have low visual quality and are inconsistent. *Second row:* We use a factorized text-to-video approach that produces high quality videos and improves consistency. *Third row:* Not using a zero terminal-SNR noise schedule at 512px generation leads to significant inconsistencies in the generations. *Bottom row:* Finetuning our model (second row) with HQ data increases the motion in the generated videos.

ity to respect the motion specified in the text prompt as reflected by the strong gain in Faithfulness.

Parameter freezing. We test if freezing the spatial parameters of our model affects performance. We compare against a model where all parameters are finetuned during the second 512px training stage. For fair comparison, we use the same conditioning images $\mathbf{I}$ across our model and this baseline. Table 1e suggests that freezing the spatial parameters produces better videos, while reducing training cost.

4.2. Comparison to prior work

We evaluate EMU VIDEO against prior work and train $\mathcal{F}$ to produce 16 frame 4 second long videos and use the best design decisions from § 4.1, including high quality finetuning. We use the interpolation model $\mathcal{I}$ on our generations to get 16fps videos. Please see Appendix A for details on how we interpolate 16-frame videos with $\mathcal{I}$.

Human evaluation of text-to-video generation. Since many recent prior methods in text-to-video generation are closed source [7, 30, 31, 38], we use the publicly released examples from each of these methods. Note that the released videos per method are likely to be the 'best' representative samples from each method and may not capture their failure modes. For Make-A-Video, we obtained non cherry-picked generations through personal communication with the authors. For CogVideo [41], we perform T2V generation on the prompt set from [7] using the open source models. We also benchmark against commercially engineered black-box text-to-video solutions, Gen2 [54] and PikaLabs [47], for which we obtain generations through their respective websites using the prompts from [7]. We do not cherry-pick or contrastively rerank [59, 85] our videos,

Figure 5. Qualitative comparison. EMU VIDEO produces higher quality generations compared to Imagen Video [38] and Align Your Latents [7] in terms of style and consistency.

and generate them using a deterministic random noise seed that is not optimized in any way.

Since each method generates videos at different resolutions, aspect-ratios, and frame-rates, we reduce annotator bias in human evaluations by postprocessing the videos for each comparison in Figure 2 so that they match in these aspects. Full details on this postprocessing and the text prompts used are in Appendix D. As shown in Figure 2, EMU VIDEO's generations significantly outperform all prior work, including commercial solutions, both in terms of Quality (by an average of 91.8%) and Faithfulness (by an average of 86.6%). We show some qualitative comparisons in Figure 5 and some additional generations in Figure 1. EMU VIDEO generates videos with significantly higher quality, and overall faithfulness to both the objects and motion specified in the text. Since our factorized approach explicitly generates an image, we retain the visual diversity and styles of the T2I model, leading to far better videos on fantastical and stylized prompts. Additionally, EMU VIDEO generates videos with far greater temporal consistency than prior work. We hypothesize that since we use stronger conditioning of image and text, our model is trained with a relatively easier task of predicting how an image evolves into the future, and thus is better able to model the temporal nature of videos. Please see Appendix E for more qualitative comparisons. We include human evaluations where videos are not postprocessed in the supplement Appendix D, where again EMU VIDEO's generations significantly outperform all prior work. The closest model in performance compared to ours is Imagen Video when measured on Faithfulness, where we outperform Imagen Video by 56%. Imagen Video's released prompts ask for generating text characters, a known failure mode [17, 62] of latent diffusion models used in EMU VIDEO.

We inspect the reasons that human evaluators prefer EMU VIDEO generations over the two strongest competitors

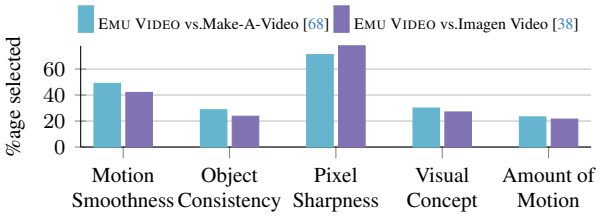


Figure 6. Percentage of each reason selected for samples where EMU VIDEO wins against Make-A-Video [68] or Imagen Video [38] on Quality. Human evaluators pick EMU VIDEO generations primarily because of their pixel sharpness and motion smoothness, with an overall preference of 96.8% and 81.8% to each baseline, respectively.

Method	Automated FVD ↓	IS ↑
MagicVideo [88]	655.0	-
Align Your Latents [7]	550.6	33.5
Make-A-Video [68]	367.2	33.0
PYOCO [30]	355.2	47.8
EMU VIDEO	606.2	42.7

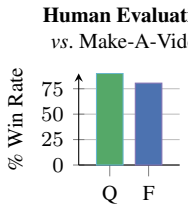


Table 2. Automated metrics are flawed for zero-shot text-to-video evaluation on UCF101. (Left) We present automated metrics and observe that EMU VIDEO does not outperform prior work. (Right) We use human evaluations to compare EMU VIDEO and Make-A-Video where EMU VIDEO significantly outperforms Make-A-Video both in Quality (90.1%) and Faithfulness (80.5%).

in Figure 6. A more detailed inspection is provided in Appendix C. EMU VIDEO generations are preferred due to their better pixel sharpness and motion smoothness. While being state-of-the-art, EMU VIDEO is also simpler and has a two model cascade with a total of 6.0B parameters (2.7B frozen parameters for spatial layers, and 1.7B learnable temporal parameters each for $\mathcal{F}$ and $\mathcal{I}$), which is much simpler than methods like Imagen Video (7 model cascade, 11.6B parameters), Make-A-Video (5 model cascade, 9.6B parameters) trained using similar scale of data.

Automated metrics. In Table 2, we compare against prior work using the zero-shot T2V generation setting from [68] on the UCF101 dataset [71]. EMU VIDEO achieves a comptetitive IS score [65] and a higher FVD [73]. Prior works suggest that the automated metrics are flawed and do not capture human preferences [6, 14, 17, 38, 57, 68]. We believe FVD penalizes our high quality generations that are different from the UCF101 videos, while IS is biased towards its training data [6, 14]. To confirm this, we use human evaluations to compare our generations to Make-A-Video. We use a subset of 303 generated videos (3 random samples per UCF101 class) and find that our generations are strongly prefered (Table 2 Right). Qualitative comparisons

Method	#Prompts	Q	F
EMU VIDEO *vs*. VideoComposer [76]	65 [7]	96.9	96.9
EMU VIDEO *vs*. PikaLabs I2V [47]		84.6	84.6
EMU VIDEO *vs*. Gen2 I2V [54]		70.8	76.9
EMU VIDEO *vs*. VideoComposer [76]	307 [68]	97.4	91.2

Table 3. Human evaluation of EMU VIDEO *vs*. prior work in text-conditioned image animation. We compare EMU VIDEO against three methods across two prompt sets using the generations from [57] as the starting images. EMU VIDEO's animated videos are strongly prefered on both the prompt sets over all baselines.

can be found in Appendix E.

Animating images. A benefit of our factorized generation is that the same model can be used out-of-the-box to 'animate' user-provided images by supplying them as the conditioning image **I**. We compare EMU VIDEO's image animation with three methods, concurrent work [76] and commercial image-to-video (I2V) solutions [47, 54], on the prompts from [68] and [7]. All the methods are shown the same image generated using a different text-to-image model [57] and expected to generate a video according to the text prompt. We use the API for [57] in our comparisons since the official training data and model is not available. We report human evaluations in Table 3 and automated metrics in the supplement (*cf*. Appendix Table 9). Human evaluators strongly prefer EMU VIDEO's generations across all the baselines. These results demonstrate the superior image animation capabilities of EMU VIDEO compared to methods specifically designed for the image-to-video task.

4.3. Analysis

Nearest neighbor baseline. We expect good and useful generative models to outperform a nearest neighbor retrieval baseline and create videos not in the training set. We construct a strong nearest neighbor baseline that retrieves videos from the full training set (34M videos) by using the text prompt's CLIP feature similarity to the training prompts. When using the evaluation prompts from [68], human evaluators prefer EMU VIDEO's generations 81.1% in Faithfulness over real videos confirming that EMU VIDEO outperforms the strong retrieval baseline. We manually inspected and confirmed that EMU VIDEO outperforms the baseline for prompts not in the training set.

Extending video length with longer text. Recall that our model conditions on the text prompt and a starting frame to generate a video. With a small architectural modification, we can also condition the model on T frames and *extend* the video. Thus, we train a variant of EMU VIDEO to generate the future 16 frames conditioned on the 'past' 16 frames. While extending the video, we use a *future* text prompt different from the one used for the original video and visualize results in Figure 7. We find that the extended videos respect the original video as well as the future text prompt.

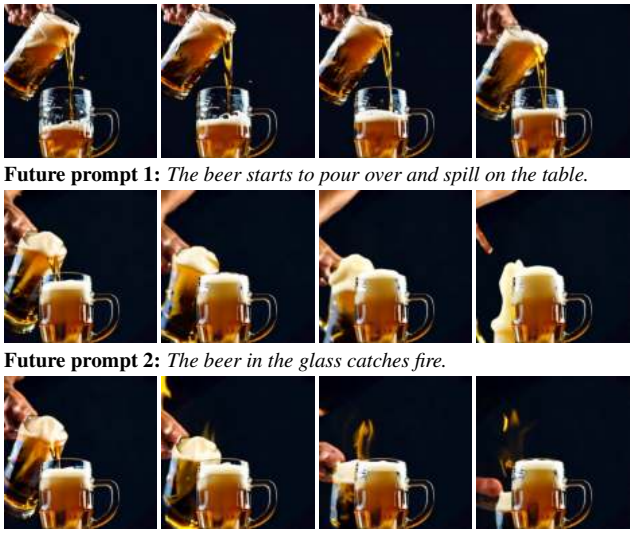


Figure 7. Extending to longer videos. We test a variant of EMU VIDEO that is conditioned on all the frames from the original video, and generates new videos conditioned on a future prompt. For two different future prompts, our model generates plausible extended videos that respect the original video and the future text.

5. Limitations and Future Work

We presented EMU VIDEO, a factorized approach to text-to-video generation that leverages strong image and text conditioning. EMU VIDEO significantly outperforms all prior work including commercial solutions. There is a difference in the image conditioning used for our model at train and inference: at training, we use a video frame sampled from real videos, while at inference we use a generated image (using the spatial parameters of the model). In practice, this difference does not affect the quality of the generated video for most scenarios. However, in cases where the generated image used for conditioning at inference is not representative of the prompt, our model has no way to recover from this error. We believe that improving the models ability to recover from such errors is an important direction for future work. Strengthening the conditioning for video models using pure autoregressive decoding with diffusion models is not currently computationally attractive. However, further research may provide benefits for longer video generation.

Ethical considerations. We propose advancements in generative methods specifically to improve the generation of high dimensional video outputs. Generative methods can be applied to a large variety of different usecases which are beyond the scope of this work. A careful study of the data, model, its intended applications, safety, risk, bias, and societal impact is necessary before any real world application.

Acknowledgments. We are grateful for the support of multiple collaborators at Meta who helped us in this work. Baixue Zheng, Baishan Guo, Jeremy Teboul, Milan Zhou, Shenghao Lin, Kunal Pradhan, Jort Gemmeke, Jacob Xu, Dingkang Wang, Samyak Datta, Guan Pang, Symon Perriman, Vivek Pai, Shubho Sengupta for their help with the data and infra. We would like to thank Uriel Singer, Adam Polyak, Shelly Sheynin, Yaniv Taigman, Licheng Yu, Luxin Zhang, Yinan Zhao, David Yan, Emily Luo, Xiaoliang Dai, Zijian He, Peizhao Zhang, Peter Vajda, Roshan Sumbaly, Armen Aghajanyan, Michael Rabbat, and Michal Drozdzal for helpful discussions. We are also grateful to the help from Lauren Cohen, Mo Metanat, Lydia Baillergeau, Amanda Felix, Ana Paula Kirschner Mofarrej, Kelly Freed, Somya Jain. We thank Ahmad Al-Dahle and Manohar Paluri for their support.

References

[1] Armen Aghajanyan, Po-Yao (Bernie) Huang, Candace Ross, Vladimir Karpukhin, Hu Xu, Naman Goyal, Dmytro Okhonko, Mandar Joshi, Gargi Ghosh, Mike Lewis, and Luke Zettlemoyer. Cm3: A causal masked multimodal model of the internet. *ArXiv*, abs/2201.07520, 2022. 2

[2] Nuha Aldausari, Arcot Sowmya, Nadine Marcus, and Gelareh Mohammadi. Video generative adversarial networks: A review. *ACM Comput. Surv.*, 55(2), 2022. 2

[3] Jie An, Songyang Zhang, Harry Yang, Sonal Gupta, Jia-Bin Huang, Jiebo Luo, and Xi Yin. Latent-shift: Latent diffusion with temporal shift for efficient text-to-video generation, 2023. 3

[4] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic variational video prediction. In *ICLR*, 2018. 2

[5] Mohammad Babaeizadeh, Mohammad Taghi Saffar, Suraj Nair, Sergey Levine, Chelsea Finn, and Dumitru Erhan. Fitvid: Overfitting in pixel-level video prediction. *arXiv preprint arXiv:2106.13195*, 2020. 2

[6] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018. 7

[7] A. Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22563–22575, 2023. 1, 2, 3, 4, 6, 7, 8, 14, 15, 16, 20

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. 2

[9] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei A Efros, and Tero Karras. Generating long videos of dynamic scenes. In *NeurIPS*, 2022. 2

[10] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 15

[11] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *preprint arXiv:2005.14165*, 2020. 2

[12] Ting Chen. On the importance of noise scheduling for diffusion models. *arXiv preprint arXiv:2301.10972*, 2023. 4

[13] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *arXiv preprint arXiv:2305.13840*, 2023. 3

[14] Min Jin Chong and David Forsyth. Effectively unbiased fid and inception score and where to find them. In *CVPR*, pages 6070–6079, 2020. 7

[15] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 4

[16] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets, 2019. 2

[17] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. 1, 2, 4, 5, 7

[18] Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1174–1183. PMLR, 2018. 2

[19] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 2

[20] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *NeurIPS*, 2022. 3

[21] J. Donahue, P. Krahenbühl, and T. Darrell. Adversarial feature learning. In *ICLR*, 2016. 1

[22] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 2

[23] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models, 2023. 15

[24] Hao Fei, Shengqiong Wu, Wei Ji, Hanwang Zhang, and Tat-Seng Chua. Empowering dynamics-aware text-to-video diffusion with large language models, 2023. 3

[25] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, page 64–72, Red Hook, NY, USA, 2016. Curran Associates Inc. 2

[26] Joseph L Fleiss and Jacob Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619, 1973. 16

[27] Tsu-Jui Fu, Licheng Yu, Ning Zhang, Cheng-Yang Fu, Jong-Chyi Su, William Yang Wang, and Sean Bell. Tell me what happened: Unifying text-guided video completion via multimodal masked video generation. In *CVPR*, pages 10681–10692, 2023. 2

[28] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv preprint arXiv:2203.13131*, 2022. 1

[29] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *European Conference on Computer Vision*, 2022. 2

[30] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models, 2023. 1, 2, 3, 6, 7, 16, 20

[31] Jiaxi Gu, Shicong Wang, Haoyu Zhao, Tianyi Lu, Xing Zhang, Zuxuan Wu, Songcen Xu, Wei Zhang, Yu-Gang Jiang, and Hang Xu. Reuse and diffuse: Iterative denoising for text-to-video generation, 2023. 3, 6, 16, 20

[32] Agrim Gupta, Stephen Tian, Yunzhi Zhang, Jiajun Wu, Roberto Martín-Martín, and Li Fei-Fei. Maskvit: Masked visual pre-training for video prediction. In *ICLR*, 2023. 2

[33] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. In *NeurIPS*, pages 27953–27965. Curran Associates, Inc., 2022. 2

[34] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation, 2023. 3

[35] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 13

[36] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arxiv:2006.11239*, 2020. 2, 3

[37] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *arXiv preprint arXiv:2106.15282*, 2021. 2, 3, 5

[38] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models, 2022. 1, 2, 3, 4, 6, 7, 16, 20

[39] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, pages 8633–8646. Curran Associates, Inc., 2022. 2, 3

[40] Susung Hong, Junyoung Seo, Sunghwan Hong, Heeseong Shin, and Seungryong Kim. Large language models are frame-level directors for zero-shot text-to-video generation, 2023. 2, 3

[41] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers, 2022. 1, 2, 3, 6

[42] Nal Kalchbrenner, Aäron van den Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video pixel networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1771–1779. PMLR, 2017. 2

[43] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[44] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. *arXiv preprint arXiv:2303.13439*, 2023. 2

[45] Taesup Kim, Sungjin Ahn, and Yoshua Bengio. *Variational Temporal Abstraction*. Curran Associates Inc., Red Hook, NY, USA, 2019. 2

[46] Manoj Kumar, Mohammad Babaeizadeh, Dumitru Erhan, Chelsea Finn, Sergey Levine, Laurent Dinh, and Durk Kingma. Videoflow: A conditional flow-based model for stochastic video generation. In *ICLR*, 2020. 2

[47] Pika Labs. Pika labs. https://www.pika.art/. 2, 6, 8

[48] Ivan Laptev and Tony Lindeberg. Space-time interest points. In *ICCV*, 2003. 2

[49] Seungwoo Lee, Chaerin Kong, Donghyeon Jeon, and Nojun Kwak. Aadiff: Audio-aligned video synthesis with text-to-image diffusion, 2023. 2

[50] Long Lian, Baifeng Shi, Adam Yala, Trevor Darrell, and Boyi Li. Llm-grounded video diffusion models. *arXiv preprint arXiv:2309.17444*, 2023. 3

[51] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. *arXiv preprint arXiv:2305.08891*, 2023. 4, 5, 13

[52] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 13

[53] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error, 2016. 2

[54] Runway ML. Gen2. https://research.runwayml.com/gen2. 2, 6, 8, 15

[55] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 1

[56] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022. 2

[57] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 4, 7, 8, 15

[58] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,

Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. 2021. 4, 5

[59] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021. 2, 6

[60] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2, 3

[61] Marc'Aurelio Ranzato, Arthur Szlam, Joan Bruna, Michaël Mathieu, Ronan Collobert, and Sumit Chopra. Video (language) modeling: a baseline for generative models of natural videos. *ArXiv*, abs/1412.6604, 2014. 2

[62] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 2, 4, 5, 7, 13

[63] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 2, 3

[64] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models, 2022. 3, 5

[65] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. 7

[66] Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. StyleGAN-T: Unlocking the power of GANs for fast large-scale text-to-image synthesis. 2023. 2

[67] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun WOO. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *NeurIPS*. Curran Associates, Inc., 2015. 2

[68] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *ICLR*, 2023. 1, 2, 3, 4, 5, 7, 8, 14, 15, 16, 20

[69] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2256–2265, Lille, France, 2015. PMLR. 2, 3

[70] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, 2020. 5, 13

[71] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. *CRCV-TR-12-01*, 2012. 7

[72] Zineng Tang, Ziyi Yang, Chenguang Zhu, Michael Zeng, and Mohit Bansal. Any-to-any generation via composable diffusion, 2023. 2

[73] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. 2019. 4, 7

[74] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *International Conference on Learning Representations*, 2023. 2

[75] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 613–621, 2016. 2

[76] Xiang* Wang, Hangjie* Yuan, Shiwei* Zhang, Dayou* Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *arXiv preprint arXiv:2306.02018*, 2023. 2, 3, 4, 8, 15

[77] Nevan Wichers, Ruben Villegas, D. Erhan, and Honglak Lee. Hierarchical long-term video prediction without supervision. In *International Conference on Machine Learning*, 2018. 2

[78] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *ArXiv*, abs/2104.14806, 2021. 2

[79] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *ICCV*, 2023. 2

[80] Zhen Xing, Qi Dai, Han Hu, Zuxuan Wu, and Yu-Gang Jiang. Simda: Simple diffusion adapter for efficient video generation, 2023. 3

[81] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers, 2021. 2

[82] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*, 2022. 2

[83] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*, 2023. 3

[84] Shengming Yin, Chenfei Wu, Huan Yang, Jianfeng Wang, Xiaodong Wang, Minheng Ni, Zhengyuan Yang, Linjie Li, Shuguang Liu, Fan Yang, Jianlong Fu, Gong Ming, Lijuan Wang, Zicheng Liu, Houqiang Li, and Nan Duan. Nuwa-xl: Diffusion over diffusion for extremely long video generation, 2023. 2

[85] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 6

[86] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alex Hauptmann, Ming-Hsuan

Yang, Yuan Hao, Irfan Essa, and Lu Jiang. Magvit: Masked generative video transformer. In *CVPR*, 2023. 2

[87] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation, 2023. 2, 3

[88] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models, 2023. 7

Table Of Contents

A. Implementation Details

In this section we include details on the architectures and hyper-parameters used for training the models in the main paper, and on the use of multiple conditionings for classifier-free guidance. For both our text-to-video ($\mathcal{F}$) and interpolation ($\mathcal{I}$) models we train with the same U-Net architecture. We share the exact model configuration for our U-Net in Table 4, and the configuration for our 8-channel autoencoder in Table 5.

Setting	Value
input_shape	[17, T, 64, 64]
output_shape	[8, T, 64, 64]
model_channels	384
attention_resolutions	[4, 2, 1]
num_res_blocks	[3, 4, 4, 4]
channel_multipliers	[1, 2, 4, 4]
use_spatial_attention	True
use_temporal_attention	True
transformer_config:	
d_head	64
num_layers	2
context_dim_layer_1	768
context_dim_layer_2	2048

Table 4. U-Net architecture details. Our U-Net contains 4.3B total parameters, out of which 2.7B are initialized from our pre-trained text-to-image model and kept frozen, resulting in 1.7B trainable parameters. T is the total frames produced by the model.

Table 6 shares the training hyperparameters we used for various stages of our training – 256px training, 512px training, High Quality finetuning, and frame interpolation. For inference, we use the DDIM sampler [70] with 250 diffusion steps. We use Classifier Free Guidance (CFG) [35] with w_{img} of 7.5 for image generation, and w_{img} of 2.0 and w_{txt} of 7.5 for both video generation and frame interpolation. We share more details about handling multiple conditionings for Classifier Free Guidance next.

Setting	Value
type	AutoencoderKL [62]
z_channels	8
in_channels	3
out_channels	3
base_channels	128
channel_multipliers	[1, 2, 4, 4]
num_res_blocks	2

Table 5. VAE architecture details. We use an image based VAE and apply it to videos frame-by-frame. Our VAE encoder downsamples videos spatially by 8×8 and produces 8 channel latents.

Setting	Training stage: 256px $\mathcal{F}$	512px $\mathcal{F}$	HQ FT $\mathcal{F}$	FI $\mathcal{I}$
Diffusion settings:				
Loss	Mean Squared Error			
Timesteps	1000			
Noise Schedule	quad	quad*		
Beta start	8.5×10^{-4}	$8.5 \times 10^{-4^*}$		
Beta end	1.2×10^{-2}	$1.2 \times 10^{-2^*}$		
Var type	Fixed small			
Prediction mode	eps-pred	v-pred		
0-term-SNR rescale	False	True [51]		
Optimizer	AdamW [52]			
Optimizer Momentum	$\beta_1 = 0.9, \beta_2 = 0.999$			
Learning rate:				
Schedule	Constant			
Warmup Schedule	Linear			
Peak	1e-4		2.5e-5	1.5e-4
Warmup Steps	1K		10K	1.5K
Weight decay	0.0		1e-4	0.0
Dataset size	34M		1.6K	34M
Batch size	512		64	384
Transforms:				
Clip Sampler	Uniform			
Frame Sampler	Uniform			
Resize				
interpolation	Box + Bicubic			
size	256px	512px		
Center Crop	256px	512px		
Normalize Range	[-1, 1]			

Table 6. Training hyperparameters for various stages in our pipeline: 256px training, 512px training, High Quality finetuning (HQ FT), and frame interpolation (FI). *: noise schedules are changed afterwards with zero terminal-SNR rescaling [51].

Multiple Conditionings for CFG. For video generation, our model receives two conditioning signals (image $\mathbf{I}$, text prompt $\mathbf{p}$), which we use in conjunction for Classifier Free Guidance [35]. Eq 1 lists the combined CFG equation we

use.

$$\tilde{\mathbf{X}} = \mathbf{X} + w_i(\mathbf{X}(\mathbf{I}) - \mathbf{X}(\varnothing)) + w_p(\mathbf{X}(\mathbf{I}, \mathbf{p}) - \mathbf{X}(\mathbf{I})) \quad (1)$$

Eq 1 was chosen such that: (1) if the CFG scales for image w_i and text prompt w_p are both equal to 1, the resulting vector $\tilde{\mathbf{X}}$ should be equal to the prediction $\mathbf{X}(\mathbf{I}, \mathbf{p})$ conditioned on the image and text, without Classifier Free Guidance. (2) if the CFG scales for image w_i and text w_p are both equal to 0, the resulting vector $\tilde{\mathbf{X}}$ should be equal to the un-conditioned prediction $\mathbf{X}(\varnothing)$.

In Eq 1 there is an ordering on the conditionings. We also considered alternate orderings in which we start with the text conditioning first instead of the image conditioning:

$$\tilde{\mathbf{X}} = \mathbf{X} + w_p(\mathbf{X}(\mathbf{p}) - \mathbf{X}(\varnothing)) + w_i(\mathbf{X}(\mathbf{I}, \mathbf{p}) - \mathbf{X}(\mathbf{p})) \quad (2)$$

Eq 2 did not lead to improvement over Eq 1, but required significantly different values for w_i and w_p to work equally well. We also considered formulas without ordering between the two conditionings, for instance:

$$\tilde{\mathbf{X}} = \mathbf{X} + w_i(\mathbf{X}(\mathbf{I}) - x(\varnothing)) + w_p(\mathbf{X}(\mathbf{p}) - x(\varnothing))$$

and

$$\tilde{\mathbf{X}} = \mathbf{X}(\mathbf{I}, \mathbf{p}) + w_i'(\mathbf{X}(\mathbf{I}, \mathbf{p}) - \mathbf{X}(\mathbf{p})) + w_p'(\mathbf{X}(\mathbf{I}, \mathbf{p}) - x(\mathbf{I}))$$

where $w_i' = (w_i - 1)$ and $w_p' = (w_p - 1)$

Similar to Eq 2, those formulas did not improve over Eq 1, and in addition miss the useful properties listed above.

Selecting CFG scales. Eq 1 requires to find the guidance factor w_i for image and w_p for text. We found that these factors influence the motion in the generated videos. To quantify this, we measure a 'motion score' on the generated videos by computing the mean energy of the motion vectors in the resulting `H.264` encoding. We found that the motion score was a good proxy for the amount of motion, but did not provide signal into consistency of the motion. Higher motion as computed through motion vectors does not necessarily translate to interesting movement, as it could be undesirable jitter, or reflect poor object consistency. Table 7 shows how the CFG scales directly influence the amount of motion in the generated videos.

After narrowing down a few CFG value combinations by looking at the resulting motion score, we identified the best values by visual inspection and human studies. Qualitatively, we found that the (1) higher w_i for a fixed w_p, the more the model stays close to the initial image and favors camera motion; and (2) the higher w_p for a fixed w_i, the more the model favors movement at the expense of object consistency.

Frame Interpolation Model. Here, we include extra details on the frame interpolation model, $\mathcal{I}$. First we explain our masked zero-interleaving strategy. Second we explain how we interpolate 16-frame 4fps videos from $\mathcal{F}$. § 3.3 in the main paper details how $\mathcal{I}$ is trained to take 8 zero-interleaved frames (generated from $\mathcal{F}$ at 4fps) as conditioning input and generate 37 frames at 16fps. One option for training an interpolation model that increases the fps by 4-fold is to generate 3 new frames between every pair of input frames (as in [7]). However, the downside to this approach is that the resulting interpolated video has a slightly shorter duration than the input video (since every input frame has 3 new generated frames after it, except the last input frame). We instead take the approach of using $\mathcal{I}$ to increase the duration of the input video, and we design a zero-interleaving scheme accordingly. Our interpolation model is trained to generate 3 new frames between every pair of frames, and also 4 new frames either side of the input video. As a result, during training $\mathcal{I}$ takes as conditioning input a 2s video, and generates a 2.3s video.

For interpolating 16-frame input videos from $\mathcal{F}$ (as described in § 4.2 in the main paper), we simply split the videos into two 8-frame videos and run interpolation on both independently. In order to construct our final interpolated video, we discard the *overlapping* frames (the last 5 frames of the first interpolated video, and the first 4 of the second), and concatenate the two videos frame-wise. The resulting interpolated video is 65 frames long at 16fps (4.06 seconds in duration – we refer to these videos as 4 seconds long in the main paper for brevity).

Model	w_p	w_i	Motion Score
w/o HQ finetuning	2.0	1.0	1.87
w/o HQ finetuning	8.0	1.0	2.87
w/o HQ finetuning	16.0	1.0	3.86
w/o HQ finetuning	8.0	1.0	2.87
w/o HQ finetuning	8.0	2.0	0.61
w/o HQ finetuning	8.0	3.0	0.25
HQ finetuned	2.0	2.0	11.1
HQ finetuned	8.0	2.0	12.7
HQ finetuned	16.0	2.0	13.5
HQ finetuned	8.0	1.0	14.9
HQ finetuned	8.0	2.0	12.7
HQ finetuned	8.0	3.0	11.3

Table 7. We measure the amount of motion in the generated videos using an automated motion score where a higher value reflects more motion. We use the prompts from [68]. The ratio of text CFG scale w_p to image CFG scale w_i influences the amount of motion in the video. We also observe that, w/o HQ fine-tuning, motion is much less and that the relative effect of CFG scales is even more pronounced.

B. Additional experiments

We detail additional experiments, viz. (i) an investigation into the effect of the initial image on our video generations, (ii) a quantitative comparison to prior work in image anima-

Method	#Prompts	Q	F
Gen2 *vs*. Gen2 I2V	65 [7]	41.5	44.6
EMU VIDEO *vs*. Gen2 I2V		72.3	78.4
EMU VIDEO *vs*. Gen2		78.5	87.7

Table 8. Image conditioning for commercial T2V We compare EMU VIDEO against two video generation variants of Gen2 API: (1) Gen2 which accepts only a text prompt as input and (2) Gen2 I2V which accepts an input image (generated using [57]) and a text prompt. We observe that the second variant (Gen2 I2V) outperforms the text-to-video Gen2 variant. EMU VIDEO's generations are strongly preferred to both the variants of the Gen2 API.

tion with automated metrics, (iii) a joint investigation into the effect of the number of training steps and data, and finally (iv) an analysis into the effect of the amount of training data.

Image conditioning for commercial T2V systems. We study the effect of image conditioning on the commercial T2V solution from Gen2 [54] in Table 8. The Gen2 API has two video generation variants: (1) A pure T2V API that accepts a text prompt as input and generates a video; and (2) an "image + text" API, denoted as Gen2 I2V, that takes an image and a text prompt as input to generate a video. We use images generated from [57] for the Gen2 I2V variant.

We observe that the Gen2 I2V variant outperforms the Gen2 API that only accepts a text prompt as input. We benchmark EMU VIDEO against both variants of the API and observe that it outperforms Gen2 and the stronger Gen2 I2V API. In Table 3, we also compare EMU VIDEO using the same images as Gen2 I2V for "image animation" and observe that EMU VIDEO outperforms Gen2 I2V in that setting as well.

Automated metrics for image animation. We follow the setting from Table 3 and report automated metrics for comparison in Table 9. Following [23, 76], we report Frame consistency (FC) and Text consistency (TC). We also report CLIP Image similarity [10] (IC) to measure the fidelity of generated frames to the conditioned image. We use CLIP ViT-B/32 model for all the metrics. Compared to VideoComposer [76], EMU VIDEO generates smoother motion, as measure by frame consistency, maintains a higher faithfulness to the conditioned image, as measured by the image score, while adhering to the text on both the prompt sets. EMU VIDEO fares slightly lower compared to PikaLabs and Gen2 on all three metrics. Upon further inspection, EMU VIDEO (motion score of 4.98) generates more motion compared to PikaLabs and Gen2 (motion scores of 0.63 and 3.29 respectively). Frame and image consistency favour static videos resulting in the lower scores of EMU VIDEO on these metrics.

Effect of the number of training steps and data. In Figure 8, we vary the number of training steps in the initial

Method	Dataset	FC (↑)	IC (↑)	TC(↑)
VideoComposer [76]	AYL [7]	96.8	86.4	33.3
PikaLabs I2V		99.9	95.0	34.6
Gen2 I2V		99.9	96.8	34.3
EMU VIDEO		99.3	94.2	34.2
VideoComposer [76]	MAV [68]	95.2	82.6	31.3
EMU VIDEO		98.9	91.3	32.1

Table 9. Automatic evaluation of EMU VIDEO *vs*. prior work in text-conditioned image animation. We compare EMU VIDEO against three contemporary methods following the settings from 3 using Frame consistency (FC), Image similarity (IC), and Text consistency (TC). EMU VIDEO outperforms VideoComposer across both the prompt sets and all three metrics. Automatic metrics favor static videos to ones with motion, resulting in lower scores for EMU VIDEO compared to PikaLabs and Gen2.

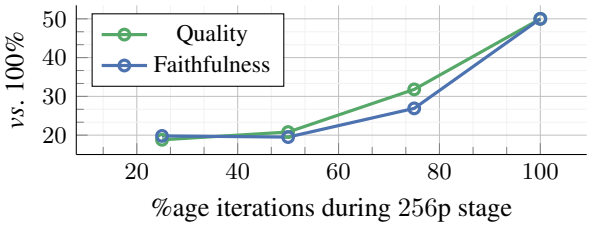


Figure 8. Performance *vs*. training iterations. On training the 256px stage for fewer iterations, we compare the generations after the same 512px finetuning to the 100% trained model via human evaluations. We observe a gradual drop in performance, indicating the importance of the low-resolution high-FPS pretraining stage.

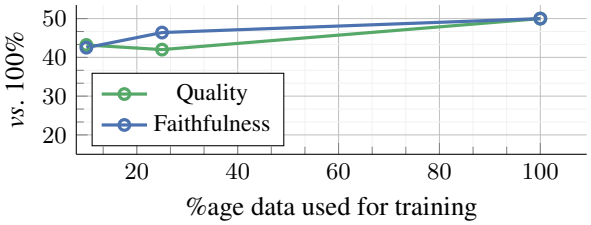


Figure 9. Performance *vs*. training data. We train our model with less data (for both 256px and 512px stages) while keeping the training steps constant, and compare the generations with the the 100% data model via human evaluations. We observe that even with 10% data, we only see a slight degradation in performance ($\sim$ 43% on both Quality and Faithfulness), showcasing that our method works well even with a fraction of the data.

low-resolution high-FPS pretraining stage. Note that since we run one full epoch through the data during this training stage, reducing the steps correspondingly also reduces the amount of training data seen. We finetune each of these models at higher resolution/low FPS (512px, 4fps) for the same (small) number of steps – 15K. We compare the model trained with 100% low-resolution pretraining with models with less low-resolution pretraining using human evaluations. We observe a gradual drop in performance as we reduce the low-resolution pretraining iterations to 75%, 50%

and 25%, indicating the importance of that stage.

Effect of the amount of training data. In Figure 9, we vary the amount of training data, while keeping the training iterations fixed for both the training stages, and perform a similar comparison as in Figure 8. Here we find a much smaller drop in performance as we reduce the amount of data. This suggests that EMU VIDEO can be trained effectively with relatively much smaller datasets, as long as the model is trained long enough (in terms of training steps).

Source	#prompts
Make-A-Video [68]	307
Imagen Video [38]	55
Align Your Latents [7]	65
PYOCO [30]	74
Reuse & Diffuse [31]	23

Table 10. Text prompt sets used for evaluation in our work. We use the text prompt sets from prior work to generate videos.

C. Human evaluations

We rely on human evaluations for making quantitative comparisons to prior work. In Sec. 4 in the main paper, we introduce our method for robust human evaluations. We now give extra details on this method, termed JUICE, and analyse how it improves robustness, and explain how we ensure fairness in the evaluations. Additionally, in Table 10 we summarize the prompt datasets used for evaluations.

C.1. Robust Human Evaluations with JUICE

When comparing to prior work, we use human evaluations to compare the generations from pairs of models. Unlike the naive approach, where evaluators simply pick their choice from a pair of generations, we ask the evaluators to select a reason when making their choice. We call this approach JUICE, where evaluators are asked to 'justify your choice'. We show an example of the templates used for human evaluations for both video quality and text faithfulness in Figure 10, where the different possible justifying reasons are shown. One challenge faced when asking evaluators to justify their choice is that human evaluators who are not experts in video generation may not understand what is meant by terms such as "Object/scene consistency" or "Temporal text alignment" or may have subjective interpretations, which would reduce the robustness of the evaluations. To alleviate this challenge, for each justifying option we show the human evaluators examples of generated video comparisons where each of the factors could be used is used in determining a winner. It is important that when giving human evaluators training examples such as these that we do not bias them towards EMU VIDEO's generations over those of prior work. Thus, to ensure fairness in the comparisons, we make sure that these training examples include cases where generated videos from different prior works are superior to EMU VIDEO and vice-versa. As detailed in the main paper, for each comparison between two videos from two different models, we use the majority vote from 5 different human evaluators. To further reduce annotator bias we make sure that the relative positioning of the generated videos being shown to the human evaluators is randomized. For details on how we ensure fairness in human evaluations when comparing videos with different resolutions, see Appendix D.

Next, we analyze quantitatively how JUICE improves human evaluation reliability and robustness. To identify unbiased JUICE factors differentiating any two video generation models on Quality and Faithfulness, we made an initial pool of random video samples generated by a few models, and asked internal human raters to explicitly explain their reasoning for picking one model over another. We then categorized them into five reasons for Quality and two for Faithfulness as mentioned in Section 3.2.

Effect of JUICE on improving evaluation reliability and robustness of human evaluations. We measure the reliability of our human evaluations when evaluators are required to justify their choice. For each pair of videos which are compared, we look at the votes for model A *vs*. model B and call the agreement between annotators either 'split' (2|3 or 3|2 votes), 'partial' (4|1 or 1|4 votes), or 'complete' (5|0 or 0|5 votes). We run human evaluations comparing our generations *vs*. Make-A-Video, first using a naive evaluation template and then with JUICE, and show the results in Figure 11. We observe that the number of samples with 'split' agreement is decreased significantly by 28%, and the number of 'complete' agreements is increased by 24%.

Next, we use Fleiss' kappa [26] as a statistical measure for inter-rater reliability for a fixed number of raters. This metric stands for the amount by which the observed agreement exceeds the agreement by chance, *i.e*., when the evaluators made their choices completely randomly. Fleiss' kappa works for any number of evaluators giving categorical ratings and we show the values in Figure 12. The value of kappa is always in the range of $[-1, 1]$, with positive kappa values representing an agreement. To better understand its behavior and range of scores in our evaluation setup, we perform an experiment on a simulated data representing our specific case of 304 tasks with two classes, model A-vs-B, and five evaluators per task. We begin with computing the kappa value when we have a 'complete' agreement among evaluators on all tasks, i.e. when all five evaluators choose either model A or model B in each task. This run receives a kappa value of 1 (blue dot in Figure 12). We gradually decrease the number of samples with complete agreement by introducing samples with 'partial' agreement when four out of five evaluators picked model A or model B (green line in Figure 12) Similarly, we decrease the number of samples with complete agreement by replac-

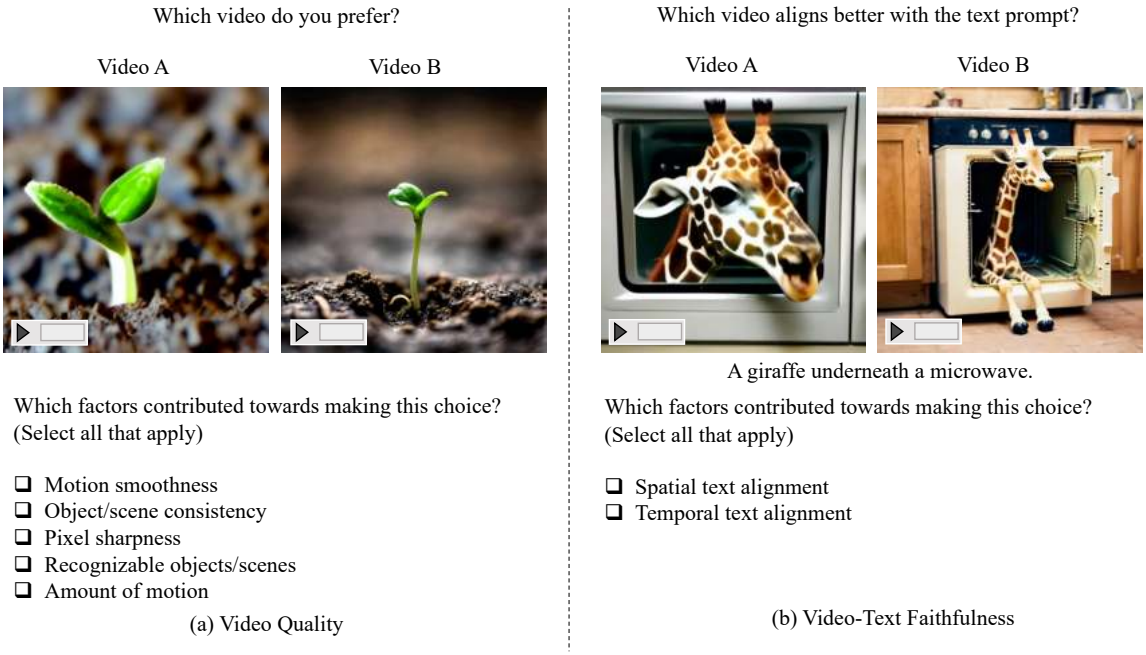


Figure 10. The JUICE template to compare two models in terms of (a) video quality and (b) video-text alignment. Here, human evaluators must justify their choice of which generated video is superior through the selection of one or more contributing factors, shown here. To ensure that human evaluators have the same understanding of what these factors mean, we additionally provide training examples of video comparisons where each of the justifying factors could be used in selecting a winner.

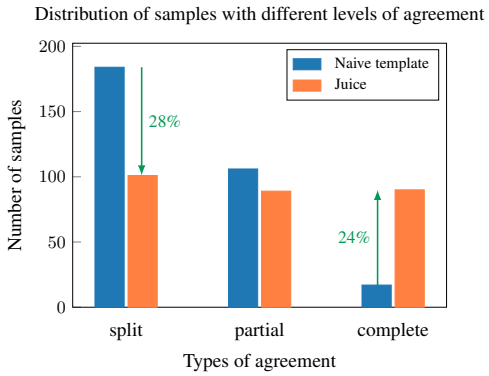


Figure 11. Human agreement in EMU VIDEO *vs*. Make-A-Video. Distribution of samples with 'split' (2|3 or 3|2 votes), 'partial' (4|1 or 1|4 votes), or 'complete' (5|0 or 0|5 votes) agreement when using a naive evaluation *vs*. JUICE. Our JUICE evaluation reduces ambiguity in the task and results in a 28% reduction in the number of samples with 'split' agreement and a 24% increase in the number of samples with 'complete' agreement. This improves Fleiss' kappa from 0.004 to 0.31.

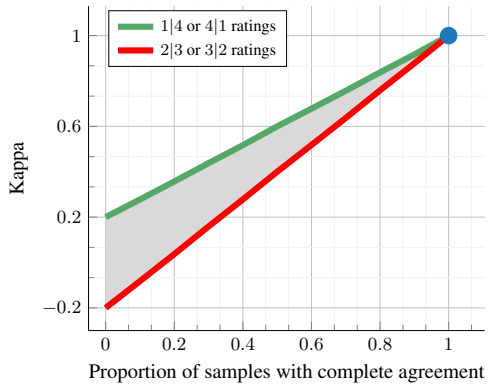
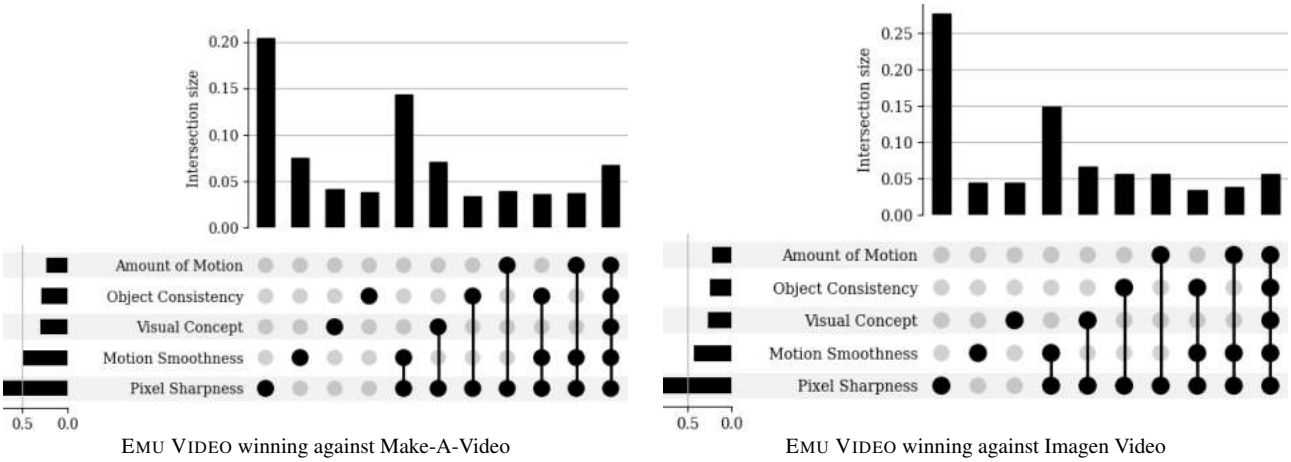


Figure 12. Analysis of Fleiss' kappa for a simulated two-class five-raters evaluation task. The blue dot shows the kappa value when we have a complete agreement among evaluators on all the samples. We progressively replace samples with 5|0 or 0|5 votes (complete agreeement) with either 1|4 or 4|1 or 3|2 or 2|3 votes and compute the Fleiss' kappa (shown in green and red). The shaded region shows the kappa value for different proportions of samples with complete, partial or split agreements.

ing them with samples where three out of the five evaluators picked model A or model B, illustrated with a red line. As shown in the plot, the kappa value ranges from -0.2 (ratings always being 'split') to 1.0 (ratings always having 'complete' agreement). Different proportions of samples with 'complete', 'partial' or 'split' agreements result in a kappa value in the shaded region. We compute and compare kappa values for the naive evaluation and JUICE evaluation–0.004 and 0.31, respectively–confirming the improvement in the inter-rater reliability of JUICE.



EMU VIDEO winning against Make-A-Video

EMU VIDEO winning against Imagen Video

Figure 13. Vertical bars show percentage of each reason and its co-occurrence with other reasons picked for EMU VIDEO against Make-A-Video (left) and Imagen Video (right). Horizontal bars depict the overall percentage of each reason, similar to Figure 6. Pixel sharpness and motion smoothness are the two most contributing factors in the EMU VIDEO win against both baselines.

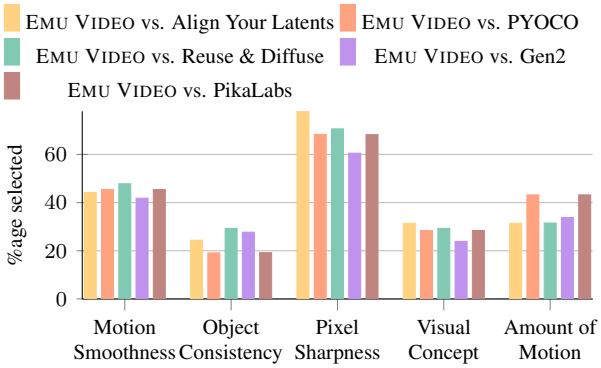


Figure 14. Percentage of each reason selected for samples where EMU VIDEO wins against each baseline model on Quality. Reasons that human evaluators pick EMU VIDEO generations over the baseline models from Figure 2 are primarily pixel sharpness and motion smoothness of our videos for most models. Amount of motion in EMU VIDEO generations is also an impactful winning factor against PYOCO and PikaLabs.

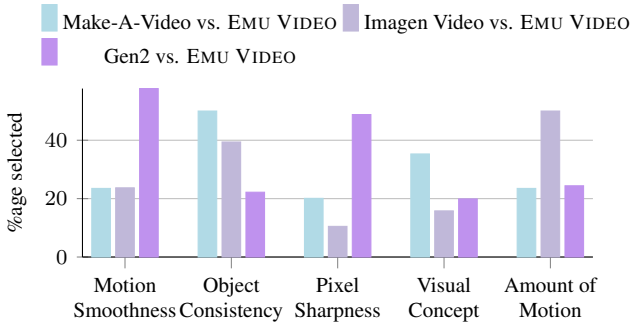


Figure 15. Percentage of each reason selected for samples where each baseline model wins against EMU VIDEO on Quality. Among the few preferred Make-A-Video generations from Figure 2 against EMU VIDEO, object consistency has been the primary reason, while for Imagen Video generations, amount of motion has been an additional considerable reason. Gen2 generations preferred over EMU VIDEO are mainly selected due to their motion smoothness and pixel sharpness.

Analyzing human evaluations. To clearly understand the strengths of each model in our evaluations, we find the most contributing factors when EMU VIDEO generations are preferred to each baseline in Figures 6, 14. A more detailed distribution of each reason and its co-occurrence with other factors is illustrated in Figure 13. We similarly, plot the percentage of each reason picked for the best three baseline generations preferred to EMU VIDEO in Figure 15.

D. Comparisons to Prior Work

In § 4.2 in the main paper, we conduct human evaluations comparing EMU VIDEO to prior work. Here, we share further details and include human evaluation results using a different setup. Specifically, in Appendix D.1 we outline the prompt datasets that are used in comparisons to prior work. In Appendix D.2 we detail how we sampled from the commercial models that we compare to in the main paper. In Appendix D.3 we give details on the postprocessing done for the human evaluations in Figure 2 in the main paper. In Appendix D.4 we include further human evaluations conducted without postprocessing the videos from EMU VIDEO or prior work.

D.1. Datasets used for Prior Work Comparisons

Since many of the methods that we compare to in Figure 2 are closed source, we cannot generate samples from all of them with one unified prompt dataset, and instead must construct different datasets via each method's repsective publicly released example generated videos. In total, we use 5

different prompt datasets. The human evaluations in Figure 2 for Make-A-Video, Imagen Video, Align Your Latents, PYOCO, and Reuse & Diffuse were conducted using the prompt datasets from the respective papers (see Table 10 for details). Certain methods that we compare to are either open-source (CogVideo) or can be sampled from through an online interface (Gen2 and Pika Labs). For these, human evaluations are conducted using the prompt set from Align Your Latents.

Model	Video Dimensions		
	T × H × W	Frame Rate	Duration (s)
EMU VIDEO	65 × 512 × 512	16	4.06
Pika	72 × 768 × 768	24	3.00
Gen2	96 × 1024 × 1792	24	4.00
CogVideo	32 × 480 × 480	8	4.00
Reuse & Diffuse	29 × 512 × 512	24	1.21
PYOCO	76 × 1024 × 1024	16	4.75
Align Your Latents	112 × 1280 × 2048	30	3.73
Imagen Video	128 × 768 × 1280	24	5.33
Make-A-Video	92 × 1024 × 1024	24	3.83
VideoComposer	16 × 256 × 256	8	2

Table 11. Video Dimensions. The dimensions of the generated videos from EMU VIDEO and each of the prior work. The top and bottom part of the table shows the specifications of Text-to-Video and Image-to-Video models respectively. Each of the prior works generates videos at different dimensions, making unbiased human evaluation a challenge.

D.2. Sampling from Commercial Models

The commercially engineered black-box text-to-video models that we compare to (Pika Labs and Gen2) can be sampled from through an online interface. Here we include details for how we sampled from these models. In both cases, these interfaces allow for certain hyper-parameters to be chosen which guide the generations.

We selected optimal parameters for each of the models by varying the parameters over multiple generations and choosing those that consistently resulted in the best generations. For Pika Labs, we use the arguments "-ar 1:1 -motion 2" for specifying the aspect ratio and motion. For Gen2, we use the "interpolate" and "upscale" arguments and a "General Motion" score of 5. All samples were generated on October 24th 2023.

D.3. Postprocessing Videos for Comparison

Our goal with our main human evaluations in Figure 2 is to ensure fairness and reduce any human evaluator bias. To ensure this fairness, we postprocess the videos from each model being compared (as outlined in § 4.2 in the main paper). Here, we give further details on the motivation behind this decision, and explain how this postprocessing is done.

Models Compared	Dimensions after Postprocessing		
	T × H × W	Frame Rate	Duration (s)
EMU VIDEO vs. Pika Labs	48 × 512 × 512	16	3.00
EMU VIDEO vs. Gen2	65 × 512 × 512	16	4.06
EMU VIDEO vs. CogVideo	32 × 480 × 480	8	4.00
EMU VIDEO vs. Reuse & Diffuse	19 × 512 × 512	16	1.19
EMU VIDEO vs. PYOCO	65 × 512 × 512	16	4.06
EMU VIDEO vs. Align Your Latents	65 × 512 × 512	16	4.06
EMU VIDEO vs. Imagen Video	65 × 512 × 512	16	4.06
EMU VIDEO vs. Make-A-Video	61 × 512 × 512	16	3.81
EMU VIDEO vs. VideoComposer	16 × 256 × 256	8	2

Table 12. Video Dimensions after postprocessing for human evaluations.. To ensure fairness in the human evaluations in in Figure 2 in the main paper, we postprocess the videos for each comparison so that they have equal dimensions and hence are indistinguishable aside from their generated content. The top and bottom part of the table shows the specifications of Text-to-Video and Image-to-Video models respectively.

Results for human evaluations conducted without any postprocessing are discussed in Appendix D.4.

As outlined in Appendix C, our human evaluations are conducted by showing evaluators repeated comparisons of videos generated by two different models for the same prompt, and asking them which model they prefer in terms of the metric being evaluated. It is key for the fairness of the human evaluation that the evaluator treats each comparison independently. It is hence important that the evaluator does not know which model generated which video, otherwise they can become biased towards one model over the other. Since each method generates videos at different dimensions (see Table 11), conducting the human evaluations without postprocessing the videos would lead to this annotator bias. Hence we decide to postprocess the videos being compared such that they have the same aspect-ratios, dimensions and frame rates so that they are indistinguishable aside from their generated content. For each pair of models being compared, we downsample these dimensions to the minimum value between the two models (see Table 12 for details). Next, we detail how we postprocess the videos.

Aspect Ratio. Since EMU VIDEO generates videos at a 1:1 aspect ratio, all videos are postprocessed to a 1:1 aspect ratio by centre cropping.

Spatial Dimension. The height and width of videos are adjusted using bilinear interpolation.

Video Duration. The duration of videos is reduced via temporal centre cropping.

Frame rate. The frame rate is adjusted using torchvision. The number of frames is selected according to the desired frame rate and video duration.

Next we discuss human evaluation results where videos are compared without any postprocessing.

	Make-A-Video	Imagen Video	Align Your Latents	PYOCO	Reuse & Diffuse	CogVideo	Gen2	PikaLabs
#Prompts	307 [68]	55 [38]	65 [7]	74 [30]	23 [31]	65 [7]	65 [7]	65 [7]
Quality	96.8	90.9	96.9	93.2	95.7	100.0	83.1	93.9
Faithfulness	86.0	69.1	90.8	89.2	100.0	100.0	98.5	100.0

Table 13. EMU VIDEO *vs*. prior work where videos are not postprocessed. We evaluate text-to-video generation in terms of video quality and text faithfulness win-rates evaluated by the majority votes of human evaluators for EMU VIDEO vs. Prior work methods. We compare methods here with their original dimensions (aspect ratio, duration, frame rate). EMU VIDEO significantly outperforms all prior work across all settings and metrics.

D.4. Prior Work at Original Dimensions

In this Section, we include further human evaluation results between EMU VIDEO and prior work where we do not perform any postprocessing on the videos and conduct the evaluations with the original dimensions (as detailed in Table 11). In this system-level comparison, human evaluators are comparing between videos that may have very different aspect ratios, durations, and frame rates, and in turn may become biased towards one model over another after seeing repeated comparisons. We note that since the dimensions of the videos here are so large, we must scale the height of each video so that both compared videos can fit on one screen for human evaluators. All other dimensions remain as in the original sampled videos. The results are in Table 13. Similar to the human evaluations conducted with postprocessed videos in Figure 2 in the main paper, EMU VIDEO significantly outperforms prior work in terms of both text faithfulness and video quality. Even when comparing EMU VIDEO's generated videos to generated videos with longer durations (including PYOCO, Imagen Video), wider aspect ratios (incliding Gen2, Align Your Latents), or higher frame rates (including Pika, Gen2), human evaluators still prefer EMU VIDEO's generated videos in both metrics. We hypothesize that the vastly improved frame quality and temporal consistency of EMU VIDEO still outweighs any benefits that come from any larger dimensions in the prior work's videos.

Interestingly, EMU VIDEO wins by larger margins here than in the postprocessed setting (an average win rate of 93.8% in quality and 93.1% in faithfulness here, vs. 91.8% and 86.6% in the postprocessed comparison). We conjecture that this improvement in win rates for EMU VIDEO may be due to the potential evaluator bias introduced in this evaluation setting. This introduced bias tends to favor EMU VIDEO since our video generations are on average superior in terms of quality and faithfulness than those of prior work. Hence in this paper we primarily report and refer to the human evaluation scores from the fairer postprocessed setting.

E. Qualitative Results

In this Section, we include additional qualitative results from EMU VIDEO (in Appendix E.1), and further qualitative comparisons between EMU VIDEO and prior work (in Appendix E.2)

E.1. Further EMU VIDEO qualitative Results

Examples of EMU VIDEO's T2V generations are shown in Figure 16, and EMU VIDEO's I2V generations are shown in Figure 17. As shown, EMU VIDEO generates high quality video generations that are faithful to the text in T2V and to both the image and the text in I2V. The videos have high pixel sharpness, motion smoothness and object consistency, and are visually compelling. EMU VIDEO generates high quality videos for both natural prompts and fantastical prompts. We hypothesize that this is because EMU VIDEO is effectively able to retain the wide range of styles and diversity of the T2I model due to the factorized approach.

E.2. Qualitative Comparisons to Prior Work

We include further qualitative comparisons to prior work in Figs. 18, 19, 20, 21, 22 and 23. This Section complements § 4.2 in the main paper where we quantatively demonstrate via human evaluation that EMU VIDEO significantly outperforms the prior work in both video quality and text faithfulness. EMU VIDEO consistently generates videos that are significantly more text faithful (see Figs. 19 and 21), with greater motion smoothness and consistency (see Figs. 20 and 22), far higher pixel sharpess (see Figure 23), and that are overall more visually compelling (see Figure 18) than the prior work.

(Ours - EMU VIDEO) *Prompt:* A hamster wearing virtual reality headsets is a dj in a disco.

(EMU VIDEO) *Prompt:* A massive tidal wave crashes dramatically against a rugged coastline.

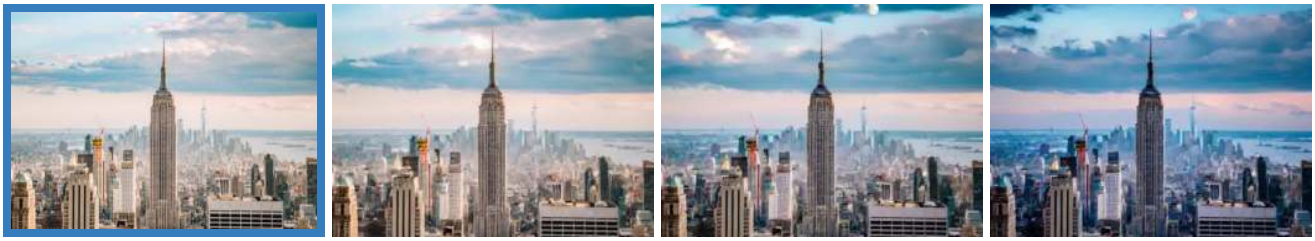
(EMU VIDEO) *Prompt:* A majestic white unicorn with a golden horn walking in slow-motion under water.

(EMU VIDEO) *Prompt:* A grizzly bear hunting for fish in a river at the edge of a waterfall, photorealistic.

Figure 16. Example T2V generations from EMU VIDEO for a selection of diverse prompts (shown above each row of frames). EMU VIDEO generates natural-looking videos which are faithful to the text and high in visual quality. The videos are highly temporally consistent, with smooth motion. EMU VIDEO is able to generate high quality videos for both natural prompts (rows 2 and 4) depicting scenes from the natural world, and also fantasical prompts including DJing hamsters (row 1) and underwater unicorns (row 3).

(Ours - EMU VIDEO) *Prompt:* The American flag waving during the moon landing with the camera panning.

(EMU VIDEO) *Prompt:* The sun sets and the moon rises.



(EMU VIDEO) *Prompt:* Satellite flies across the globe.

(EMU VIDEO) *Prompt:* horse moving its legs.

Figure 17. Example I2V generations from EMU VIDEO for a selection of diverse prompts (shown above each row of frames). EMU VIDEO generates natural-loooking videos from the conditioning image (shown in a blue box on the left side of each row of frames) and the text prompt, that have smooth and consistent motion.

(Ours - EMU VIDEO) *Prompt:* An astronaut flying in space, 4k, high resolution.

(Gen2) *Prompt:* An astronaut flying in space, 4k, high resolution.

(PikaLabs) *Prompt:* An astronaut flying in space, 4k, high resolution.

(Align Your Latents) *Prompt:* An astronaut flying in space, 4k, high resolution.

(CogVideo) *Prompt:* An astronaut flying in space, 4k, high resolution.

Figure 18. Example T2V generations from EMU VIDEO and a selection of prior work methods that we compare to in the main paper for the same prompt, namely Gen2, Pika Labs, Align your latents, and CogVideo. EMU VIDEO generates higher quality videos that are more faithful to the text, have realistic & smooth movement, and are visually compelling. In this example, CogVideo cannot generate a natural-looking video (see 5th row). PikaLabs is not faithful to the text and does not generate a realistic looking astronaut (see 3rd row), whereas Align Your Latents generates a video with low visual quality. Gen2's video, although visually superior to other prior work, lacks pixel sharpness and is not as visually compelling as EMU VIDEO.

(Ours - EMU VIDEO) *Prompt:* Teddy bear walking down 5th Avenue, front view, beautiful sunset, close up, high definition, 4k.

(Gen2) *Prompt:* Teddy bear walking down 5th Avenue, front view, beautiful sunset, close up, high definition, 4k.

(PikaLabs) *Prompt:* Teddy bear walking down 5th Avenue, front view, beautiful sunset, close up, high definition, 4k.

(Align Your Latents) *Prompt:* Teddy bear walking down 5th Avenue, front view, beautiful sunset, close up, high definition, 4k.

(CogVideo) *Prompt:* Teddy bear walking down 5th Avenue, front view, beautiful sunset, close up, high definition, 4k.

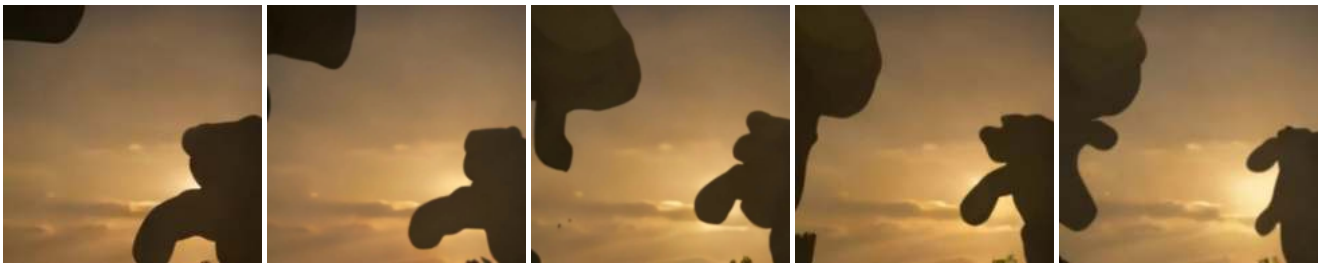


Figure 19. Example T2V generations from EMU VIDEO and a selection of prior work methods that we compare to in the main paper for the same prompt, namely Gen2, Pika Labs, Align your latents, and CogVideo. CogVideo and PikaLabs's videos are not faithful to the text and lack on visual quality. Gen2 correctly generates a video of a bear on a street, but the bear is not moving, and there is limited motion in the vidoeo. Align Your Latents's video lacks motion smoothness and pixel sharpness. On the other had, EMU VIDEO's video has very high visual quality and high text faithfulness, with smooth and consistent high motion.

(Ours - EMU VIDEO) *Prompt:* A clear wine glass with turquoise-colored waves inside it.

(Imagen Video) *Prompt:* A clear wine glass with turquoise-colored waves inside it.

(Ours - EMU VIDEO) *Prompt:* A panda bear driving a car.

(Imagen Video) *Prompt:* A panda bear driving a car.

Figure 20. Example T2V generations from EMU VIDEO and Imagen Video on two prompts (which are shown above each row of frames). Imagen Video generates videos that are faithful to the text, however the videos lack in pixel sharpness and motion smoothness. Additionally Imagen Video's generations lack fine-grained high-quality details such as in the panda's hair (see 4th row) and the water movements (see 2nd row). EMU VIDEO on the other hand generates high quality videos that are faithful to the text, and with high pixel sharpness and motion smoothness. EMU VIDEO accurately generates natural looking fine-grained details such as the hair on the panda (see 3rd row) and the water droplets in the waves (see 1st row).

(Ours - EMU VIDEO) *Prompt:* A robot dj is playing the turntable, in heavy raining futuristic tokyo rooftop cyberpunk night, sci-fi, fantasy, intricate, elegant, neon light, highly detailed, concept art, soft light, smooth, sharp focus, illustration.

(PYOCO) *Prompt:* A robot dj is playing the turntable, in heavy raining futuristic tokyo rooftop cyberpunk night, sci-fi, fantasy, intricate, elegant, neon light, highly detailed, concept art, soft light, smooth, sharp focus, illustration.

(Ours - EMU VIDEO) *Prompt:* A cute funny robot dancing, centered, award winning watercolor pen illustration, detailed, isometric illustration, drawing.

(PYOCO) *Prompt:* A cute funny robot dancing, centered, award winning watercolor pen illustration, detailed, isometric illustration, drawing.

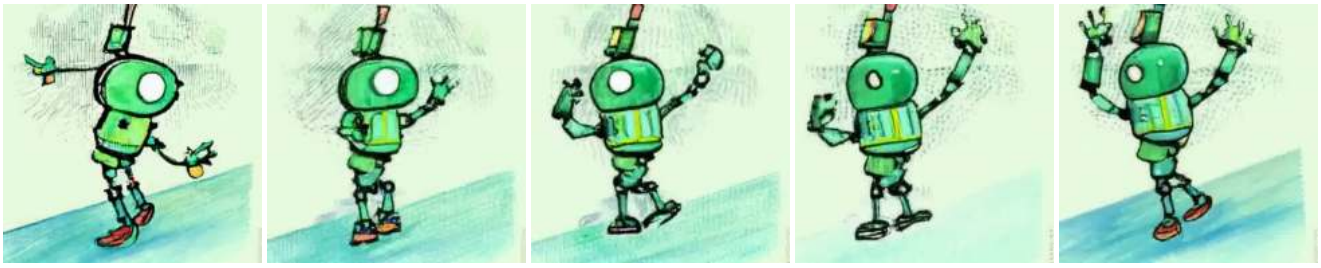


Figure 21. Example T2V generations from EMU VIDEO and PYOCO on two prompts (which are shown above each row of frames). Whereas PYOCO's videos lack motion smoothness or consistency and cannot generate fine-grained details, EMU VIDEO instead generates highly realistic videos that are smooth and consistent. EMU VIDEO can generate high quality videos given fantastical prompts.

(Ours - EMU VIDEO) *Prompt:* There's a dog with a harness on that is running through an open field and flying a kite.

(Make-A-Video) *Prompt:* There's a dog with a harness on that is running through an open field and flying a kite.

(Ours - EMU VIDEO) *Prompt:* A person standing in the ocean fishing.

(Make-A-Video) *Prompt:* A person standing in the ocean fishing.

Figure 22. Example T2V generations from EMU VIDEO and Make-A-Video on two prompts (which are shown above each row of frames). whereas Make-A-Video's videos lack pixel sharpness and object consistency, EMU VIDEO generates high quality and natural-looking videos. EMU VIDEO's videos have high motion smoothness and object consistency.

(Ours - EMU VIDEO) *Prompt:* A sailboat is sailing on a sunny day in a mountain lake.

(Reuse & Diffuse) *Prompt:* A sailboat is sailing on a sunny day in a mountain lake.

(Ours - EMU VIDEO) *Prompt:* Waves are crashing against a lone lighthouse, ominous lighting.

(Reuse & Diffuse) *Prompt:* Waves are crashing against a lone lighthouse, ominous lighting.

Figure 23. Example T2V generations from EMU VIDEO and Reuse & Diffuse on two prompts (which are shown above each row of frames). whereas Reuse & Diffuse's videos lack in visual quality both in terms of pixel sharpness, and temporal consistency, EMU VIDEO instead generates visually compelling and natural-looking videos which accurately follow the prompt.

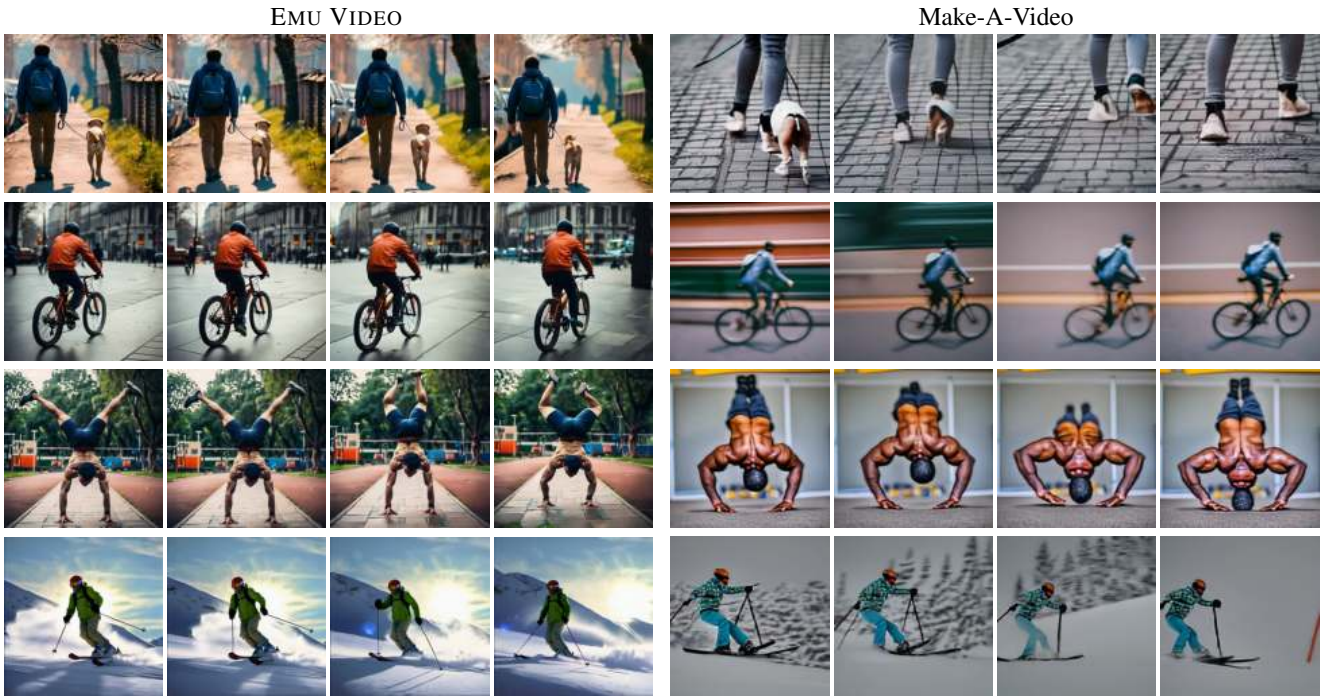


Figure 24. Zero-Shot text-to-video generation on UCF101. The classes for these videos from top to bottom are: walking with a dog, biking, handstand pushups, skiing. Our generations are of higher quality and more coherent than those from Make-A-Video.