PEEKABOO: Interactive Video Generation via Masked-Diffusion

Yash Jain*
Microsoft
Redmond

Anshul Nasery*
University Of Washington
Seattle

Vibhav Vineet
Microsoft Research
Redmond

Harkirat Behl
Microsoft Research
Redmond

Figure 1. **Zero-training interactive video generation.** PEEKABOO allows users to control the output (object size, location and motion) for off-the-shelf video diffusion models, through specially designed masking modules. First row shows a panda playing PEEKABOO by following an expanding mask on left.

Abstract

Recently there has been a lot of progress in text-to-video generation, with state-of-the-art models being capable of generating high quality, realistic videos. However, these models lack the capability for users to interactively control and generate videos, which can potentially unlock new areas of application. As a first step towards this goal, we tackle the problem of endowing diffusion-based video generation models with interactive spatio-temporal control over their output. To this end, we take inspiration from the recent advances in segmentation literature to propose a novel spatio-temporal masked attention module - PEEKABOO. *This module is a training-free, no-inference-overhead addition to off-the-shelf video generation models which enables spatio-temporal control. We also propose an evaluation benchmark for the interactive video generation task. Through extensive qualitative and quantitative evaluation, we establish that* PEEKABOO *enables control video generation and even obtains a gain of upto* $3.8\times$ *in mIoU over baseline models.*

1. Introduction

Generating realistic videos from natural language descriptions is a challenging but exciting task that has recently made significant progress [18, 32, 35, 37]. This is largely due to the development of powerful generative models and latent diffusion models (LDMs [29]), which can produce high-quality and diverse videos from text. These models have opened up new possibilities for creative applications and expression.

As the generation quality continues to improve, we can expect more innovation and potential in this domain. An important aspect is to enable more interactivity and user control over the generated videos (or better *alignment*), by allowing the user to control the spatial and temporal aspects of the video, such as the size, location, pose, and movement of the objects. This enables users to express their creativity and imagination through generating videos that match their

vision and preferences. It can also be useful for various applications, such as education, entertainment, advertising, and storytelling, where users can create engaging and personalized video content.

While current models are capable of producing temporally and semantically coherent videos, the user cannot have spatio-temporal control [38]. Moreover, these models sometimes fail to produce the main object in the video [1]. In order to control the output of videos interactively, a model would need to incorporate inputs about spatial layouts into its generation process. One set of approaches to achieve spatial control on the network output involves training the entire network or specialized adaptors on spatially grounded data [26, 36]. However, such methods involve retraining which is resource and data intensive, limiting their access to the wider community. This raises the question - Can we create a training-free technique that can introduce interactivity through desired control in videos while utilising large scale pretrained Text-to-Video (T2V) models?

In this work, we propose PEEKABOO, a training-free method to augment any off-the-shelf LDM based video-generation model with spatial control. Further, our method has negligible inference overhead. For control over individual object generation, we propose to use local context instead of global context. We propose an efficient strategy to achieve controlled generation within the T2V inference pipeline. PEEKABOO works by refocusing the spatial-, cross-, and temporal-attention in the UNet [31] blocks.

Figures 1, 3 and 5 demonstrate outputs that our method produces for a variety of masks and prompts. Our method is able to maintain a high quality of video generation, while controlling the output spatio-temporally. To evaluate the spatio-temporal control of video generation method, we propose a new benchmark by adapting an existing dataset [24], and curating a new dataset for our task (Section 5.1.1), and proposing an evaluation strategy for further research in this space. Finally, we show the versatility of our approach on two text-to-video models [35] and a text-to-image model [30]. This demonstrates the wide applicability of our method. In summary:

- We introduce PEEKABOO which i) allows interactive video generation by inducing spatio-temporal and motion control in the output of any UNet based off-the-shelf video generation model, ii) is *training-free* and iii) adds no additional latency at inference time.
- We curate and release a public benchmark, SSv2-ST for evaluating spatio-temporal control in video generation. Further, we create and release the Interactive Motion Control (IMC) dataset to evaluate interactive inputs from a human.
- We extensively evaluate PEEKABOO on i) multiple evaluation datasets, ii) with multiple T2V models (ZeroScope and ModelScope) and iii) multiple evaluation metrics. Our evaluation shows upto $2.9\times$ and $3.8\times$ gain in mIoU score by using PEEKABOO over ZeroScope and ModelScope respectively.
- We present qualitative results on spatio-temporally controlled video generation with PEEKABOO, and also showcase its ability of to overcome some fundamental failure cases present in existing models.

2. Related Work

2.1. Video Generation

Text-based video generation using latent diffusion model has taken a significant leap in recent years [7, 13, 14, 32, 41]. Make-a-video [32] introduced the 3D UNet architecture, by decomposing attention layers into spatial, cross and temporal attention layers. Further progress in this generation pipeline was made by [7, 14, 15, 41], while keeping the core three attention-layer architecture intact. Although these works focus on generating videos with high relevance to the text input, they do not provide spatio-temporal control in each frame. More recent works have tried to equip models with this ability to control generation spatio-temporally. Such methods have integrated guidance from depth maps [11], target motion [6, 16] or a combination of these modalities to generate videos [36]. However, all these works either require re-training the base model or an external adapter with aligned grounded spatio-temporal data, which is a challenging and expensive task.

On the other hand, zero-training works include Text2video-zero [18], which integrates optical flow guidance with image model to get consistent frames, ControlVideo [40], which incorporates sequence of supervising frames (depth maps, stick figures etc.) to control the motion of the video, and Free-Bloom [17], which combines a large language model (LLM) with a text-to-image model to get coherent videos. However, these methods extend specialized image models which were trained on grounded data, and cannot be used with off-the-shelf video-generation models. The closest method to our work is a concurrent work [22]. The paper uses an LLM to generate bounding box co-ordinates across scenes for an object in the prompt. They use an off-the-shelf video generation model in conjunction with a special guidance module. However, their work has a latency overhead due to extra steps in special guidance module which is absent in our method.

2.2. Controllable Text to Image generation

Recent works have explored incorporating spatial and stylistic control while generating images from text using diffusion models. These methods can broadly be categorized into those requiring training of the model [20], and training-free methods [1, 4, 10, 21, 28]. The former line of works require large amounts of compute resources, as well

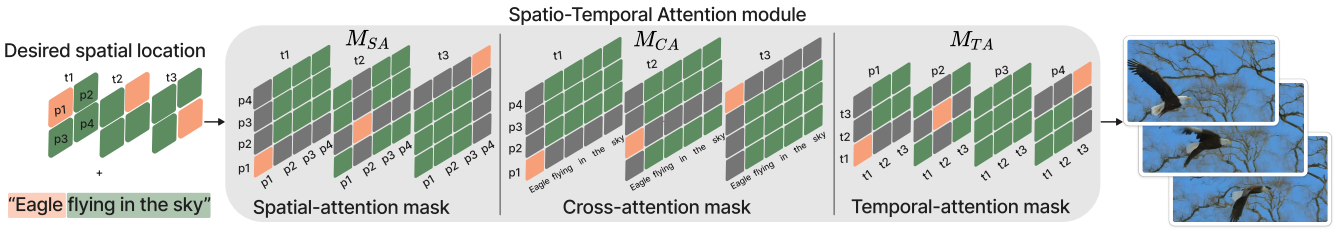


Figure 2. **PEEKABOO Module**: Our method proposes converting attention modules of an off-the-shelf 3D UNet into masked spatio-temporal mixed attention modules. We propose to use local context for generating individual objects and hence, guide the generation process using attention masks. For each of spatial-, cross-, and temporal-attentions, we compute attention masks such that foreground pixels and background pixels attend only within their own region. We illustrate these mask computations for an input mask (size 2×2 and 3 frames) which changes temporally as shown on the left. Green pixels are background pixels and orange are foreground. In the attention masks, both green and orange pixels have a value of 1, and gray pixels have a value of 0. We add the colors for ease of exposition. This masking is applied for a fixed number of steps, after which free generation is allowed. Hence, foreground and background pixels are hidden from each other before being visible, akin to a game of PEEKABOO. Best viewed in color.

as spatially grounded data to train their models. The latter either try to shape the spatial and cross-attention maps using energy function guided diffusion, or through masking. Our method is hence closer to the second type of works, however, directly extending these to videos is non-trivial due to the spatio-temporal nature of video generation.

Guided Attention The idea of guiding attention maps to control the generation in the image domain has gained popularity recently. Agarwal et al. [1] focus on minimizing overlap in attention maps for different prompt words, maintaining object information across diffusion steps. Epstein et al. [10] suggest various energy functions on cross-attention maps to control spatial properties of objects via guided sampling. Phung et al. [28] extend this by ensuring both cross and self-attention maps accurately represent objects, achieving this through optimized noise and segmented attention. Such optimization based methods have inference time overheads, in contrast with our method. Cao et al. [4] uses thresholded cross-attention maps of the object tokens as masks for self-attention, and ensures that foreground pixels only interact with other pixels within the foreground. Their method also requires multiple diffusion inference calls, or requires a source image as an input. Further, they apply their technique only for controlling the pose or actions of objects, which is orthogonal to our task.

3. Preliminaries: Video Diffusion Models

Diffusion models [33] are generative models that generate images or videos through gradually denoising random gaussian noise. The most effective amongst these are Latent Diffusion models (LDMs) [29] including Stable Diffusion. LDMs have two components: First is an image compression auto-encoder, which maps the image $\mathbf{x}$ to and back from a lower dimensional latent $\mathbf{z}$. Second component is a *Denoising Autoencoder* $f_\theta(\mathbf{z})$ which operates in the latent space and gradually converts random noise to the image latent.

Text conditioning Most current text-to-video methods utilize a conditional latent diffusion model which takes a text query as input [27, 32, 35]. The denoising autoencoder is thus conditioned on the text caption $\mathbf{c}$ as

$$\mathbf{z}_{t+1} = f_\theta(\mathbf{z}_t|\mathbf{c}), \tag{1}$$

where f_θ is a 3D UNet [31]. During inference, the input noise is iteratively cleaned and aligned towards the desired text caption. This is achieved by including a cross attention with the text embedding.

4. PEEKABOO

Spatio-temporal conditioning For interactive generation, the denoising should also be conditioned on the *user-desired* spatial location and movement of the objects in the video. This is rather complicated, because unlike Equation 1 where the entire latent $\mathbf{z}_t$ is conditioned on $\mathbf{c}$, in this setting, parts of the video have to be conditioned on parts of the caption. Note that this would become a conditional distribution with multiple conditions.

A possible solution is to encode the extra conditions as grounding pairs (spatio-temporal volume, text embedding) and pass them as context tokens in the cross attention layer, and train accordingly, taking inspiration from the image based method Gligen [20] or even Flamingo [2]. On the other hand, we want to explore using a frozen f_θ.

4.1. Masked Diffusion

We draw a parallel with the segmentation problem, which is the inverse of spatio-temporal conditioned generation prob-

lem. In particular, we take inspiration from MaskFormer [8] and Mask2Former [9] who proposed to formulate segmentation as a mask classification problem. This formulation is widely used and accepted, not just for segmentation but even detection [19] and unified models [42].

Cheng et al. [9] propose to split segmentation into grouping into N regions which are represented with binary masks. Hence, Cheng et al. [9] advocate using *local* features for segmenting individual objects. On the other hand, text-to-video diffusion models operate on conditioning a *global* context, as shown in Eqn 1. Using the above insight to tackle the problem of spatio-temporal conditioned generation we also propose to use local context for generating individual objects, and then add them together. In order to control the spatial locations of objects, we propose to modify the attention computations in the transformer blocks of the diffusion model to *masked* attention calls similar to [9]. This enables better local generation without any additional computation or diffusion steps.

4.2. Masked spatio-temporal mixed attention

Given an input bounding box for a foreground object in the video, we create a binary mask for the foreground object, and downsample it to the size of the latent. We create block sparse attention masks as described below. We use additive masking for attention, i.e. for any query Q, key K, value V a binary 2D attention mask M,

$$\text{MaskedAttention}(Q,K,V,M) = \text{softmax}(\frac{QK^T}{d} + \mathcal{M})V$$
$$\text{where} \quad \mathcal{M}[i,j] = \begin{cases} -\infty & \text{if } M[i,j]=0 \\ 0 & \text{if } M[i,j]=1 \end{cases} \tag{2}$$

Here, the additive mask $\mathcal{M}$ is such that it has a large negative value on the masked out entries in M, leading to the attention scores for such entries being small. Note that $M \in \{0,1\}^{d_q \times d_k}$, where d_q, d_k are the lengths of queries and keys respectively. We denote the length of the text prompt by l_{text}, the length of the video by l_{video}, and the dimensions of the latents by $l_{latents}$. The text input is denoted by T, and the input mask for frame f is denoted by M^f_{input}. For the ease of notation, we assume that the input masks and the latents are flattened along their spatial dimensions. The shape of M_{input} is $l_{video} \times l_{latents}$ We also define the function $\text{fg}(\cdot)$, which takes a pixel or a text token as input, and returns 1 if it corresponds to the foreground of the video, and 0 otherwise.

By nudging the foreground token to attend only to the pixels at the desired location at each frame, we can control the position, size and movement of the object. However, naively enforcing this attention constraint only in the cross-attention layer is not sufficient for spatial control. This is because the foreground and background pixels also interact through spatial- and temporal attention. We now discuss how to effectively localise the generation context.

Masked cross attention For each frame f, we compute an attention mask M^f_{CA}, which is a 2-dimensional matrix of size $l_{latents} \times l_{text}$. For each pixel-token pair, this mask is 1 iff both the pixel and token are foreground, or if both of them are in the background. Formally

$$M^f_{CA}[i,j] = \text{fg}(M^f_{input}[i]) * \text{fg}(T[j]) + (1-\text{fg}(M^f_{input}[i])) * (1-\text{fg}(T[j])) \tag{3}$$

This ensures that the latents attend to the foreground and the background tokens at the correct locations.

Masked spatial attention For each frame f, we compute an attention mask M^f_{SA} which is a 2-dimensional matrix of size $l_{latents} \times l_{latents}$. For each pixel pair, this mask is 1 iff both the pixels are foreground, or if both of them are in the background. Formally

$$M^f_{SA}[i,j] = \text{fg}(M^f_{input}[i]) * \text{fg}(M^f_{input}[j]) + (1-\text{fg}(M^f_{input}[i])) * (1-\text{fg}(M^f_{input}[j])) \tag{4}$$

This additionally focuses the attention to ensure that the foreground and background are generated at the correct locations, by encouraging them to evolve independently for the initial steps. This also helps improve the quality of generation since it leads to adequate interaction within the foreground and background regions. A similar idea in the context of image generation had been explored in MasaCtrl[4] in their self attention layer.

Masked temporal attention For each latent pixel i, we compute a mask M^i_{TA}, which is a 2D matrix of size $l_{video} \times l_{video}$. For each frame pair, the value of this mask is 1 if the pixel i is a foreground pixel in both frames, or if it is a background pixel in both frames. Formally,

$$M^i_{TA}[f,k] = \text{fg}(M^f_{input}[i]) * \text{fg}(M^k_{input}[i]) + (1-\text{fg}(M^f_{input}[i])) * (1-\text{fg}(M^k_{input}[i])) \tag{5}$$

This ensures temporal consistency for the generation since it provides correct local context for foreground and background latents across time.

4.3. Zero-training Pipeline

Putting the selective masks in a diffusion pipeline gives us a zero-training method, *dubbed* PEEKABOO. PEEKABOO integrates in the attention layers of the 3D-UNet architecture

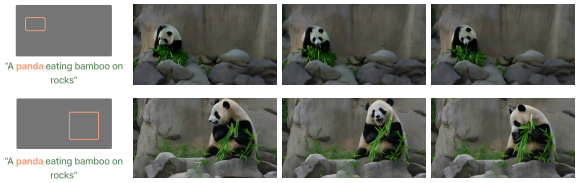


Figure 3. **Spatial control with PEEKABOO:** Changing the bounding box while providing the same prompt leads to generated panda being faithful to the input layout in terms of size and location with our method.

of text-to-video models. We perform selective generation of foreground and background object for a fixed number of steps t and then allow free-generation for the rest of steps. This free generation enables the foreground and background pixels to cohesively integrate with each other on the same canvas as have been done by [3, 21]. In essence, our method ensures that foreground pixels cannot "see" the background pixels for some steps (and vice versa), before being visible to each other. This is akin to a game of PEEKABOO.

Unlike image control methods [3, 21], PEEKABOO does not require extra inference overhead in the form of more number of diffusion steps and works with very low value of fixed step t (refer to Appendix for more details). This ensures that there is no gain in latency during generation while providing extensive spatial control.

Further, since PEEKABOO is a zero-training off-the-shelf technique it is versatile to implement in all diffusion models and can work with present as well as future text-to-video models. Thus, PEEKABOO can give spatio-temporal control in better quality generation models which are not explicitly trained on any spatially-grounded dataset.

4.4. Extensions

Currently, majority of the diffusion pipelines have a UNet-based architecture. This enables PEEKABOO to become versatile and be used not only in Text-to-Video scenario, but in Text-to-Image setup with a possibility in other generation modalities too.

Automatically generated input masks Since our method is orthogonal to the choice of input masks, we can use a large language model to generate the input masks for an object corresponding to a given prompt, in a similar fashion as concurrent works [21, 23]. In Table 2, we demonstrate that doing this leads to videos with better quality than the baseline model. Moreover, it enables our method to be end-to-end in terms of only requiring a text prompt from the user.

Image generation Image generation diffusion method are based of 2D-UNet architecture, with the absence of temporal attention layer. Analogous to our text-to-video setup, we can adapt PEEKABOO for Image Diffusion models. The spatial-attention mask maintains the semantic structure of the image while the cross-attention mask focuses the attention of foreground token on desired location and vice versa for background. In Figure 7, we showcase spatial control on an off-the-shelf diffusion model and highlight the versatility of our method.

5. Experiments

In this section, we demonstrate the effectiveness of our method. The main focus of our technique is to generate objects in specific spatio-temporal locations in videos. We first evaluate this region level control in Sec 5.1.1. In 5.1.2, we compare the generation quality against baselines to show that grounding enables much better generation. We also demonstrate qualitative results of our method, and perform ablation analysis on our method and show the effect of each component on the final generations.

5.1. Quantiative Analysis

We first present quantitative results on evaluating the spatial control and the quality of videos generated by PEEKABOO.

5.1.1 Spatial Control

Evaluation Datasets Evaluating spatial control in multiple text-to-video models is a challenging task and requires creating a common benchmark for (prompt, mask) pairs. We develop a benchmark obtained from a public video dataset with high-quality masks that represent realistic locations for day-to-day subjects. Further, we also curated a set of (prompt, mask) pairs that represent an interactive input from the user in controlling a video and its subject.

- **Something-something v2-Spatio-Temporal (ssv2-ST):** We use Something-Something v2 dataset [12, 24] to obtain the generation prompts and ground truth masks from real action videos. We filter out a set of 295 prompts. The details for this filtering are in the appendix. We then use an off-the-shelf *OWL-ViT-large* open-vocabulary object detector [25] to obtain the bounding box annotations of the object in the videos. This set represents bounding box and prompt pairs of real-world videos, serving as a test bed for both the quality and control of methods for generating realistic videos with spatio-temporal control.
- **Interactive Motion Control (IMC):** We also curate a set of prompts and bounding boxes which are manually defined. We use GPT-4 to generate prompts and pick a set of 34 prompts of objects in their natural contexts. These prompts are varied in the type of object, size of the object and the type of motion exhibited. We then annotate

Method	PEEKABOO	ssv2-ST				Interactive Motion Control (IMC)			
		mIoU % (↑)	Coverage % (↑)	CD (↓)	AP50 % (↑)	mIoU % (↑)	Coverage % (↑)	CD (↓)	AP50 % (↑)
ZeroScope [35]	-	13.9	42.0	0.22	9.3	12.6	88	0.26	0.6
	✓	34.7	56.3	0.17	39.8	36.3	96.3	0.12	33.8
ModelScope [35]	-	12.0	44.7	0.17	6.6	9.6	93.3	0.25	2.35
	✓	33.2	63.7	0.10	35.8	36.1	96.6	0.13	33.3

Table 1. **Evaluation of spatio-temporal control:** We evaluate two different models for video generation with spatio-temporal control on ssv2-ST and IMC datasets. As demonstrated by mIoU and CD, the videos generated by PEEKABOO endow the baselines with spatio-temporal control. PEEKABOO also increases the quality of the main objects in the scene, as seen by higher coverage and AP50 scores.

Method	FVD@MSR-VTT (↓)
CogVideo (English) [15]	1294
MagicVideo [41]	1290
ModelScope [35]	868
ModelScope w/ PEEKABOO	609

Table 2. **Video quality evaluation.** PEEKABOO is able to generate videos with higher quality than other baselines. We use bounding boxes generated by GPT-4 as inputs to the model.

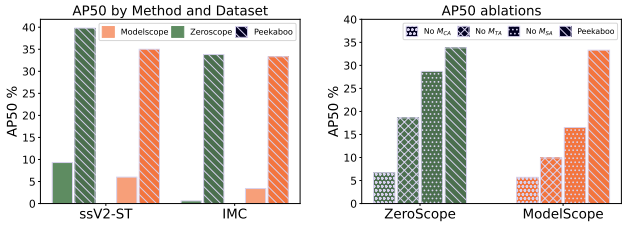


Figure 4. (a) **AP50 Scores for our datasets**: Subplot (a) shows the performance of baseline models with and without PEEKABOO on IMC and ssv2-ST. Our method provides a large gain in the scores. (b) **Ablation Studies on IMC**: The performance of PEEKABOO varies as different attention masks are removed. The AP50 drops the most when cross-attention masks are removed, indicating their importance to spatial control, followed by temporal and spatial attention. Best viewed in color.

3 sets of bounding boxes for each prompt, where the location, path taken, speed and size are varied. This set of 102 prompt-bounding box pairs serve as our custom evaluation set for spatial control. Note that since ssv2-ST dataset has a lot of inanimate objects, we bias this dataset to contain more living objects. This dataset represents possible input pairs that real users may generate.

Experimental Setup We use two base models for our evaluation, Zeroscope and ModelScope [35]. These models are run for the default number of inference steps, with default temperature and classifier guidance parameters. We also experiment with mask guidance steps in the appendix. We provide the model with the text prompt and the set of input bounding boxes. The generated videos are then evaluated for spatio-temporal control and video quality.

Evaluation methodology. After generating videos for each (prompt, mask) pair, we pass these videos through *OWL-ViT-large* detector to compute bounding boxes for each generated video. We first compute the fraction of generated videos for which OwL-ViT detects bounding boxes in more that 50% of the generated frames. We report this fraction as the *Coverage* of the model in Table 1. However, the lack of a detected bounding box does not necessarily imply the lack of an object generated, since OwL-ViT could fail to capture some objects correctly. Hence, to evaluate the spatio-temporal control of the generation method, we first filter out videos where less than 50% frames have a detected bounding box. We then compute the Intersection-over-Union of the detected bounding boxes and the input mask on these filtered videos. We report the mean of these IoU (*mIoU*) scores for each method in Table 1. These two metrics together provide a good proxy of the quality of the generated videos as well as the spatio-temporal control imparted. We compute the *Centroid Distance (CD)* as the distance between the centroid of the generated object and input mask, normalized to 1. This measures control of the generation location. Finally, we report the average precision@50% (*AP50*) of the detected and input bounding boxes averaged over all videos. For generated frames with the object present, AP50 represents the spatial control provided by the method, while mIoU measures the model's ability to match the input bounding boxes exactly and penalizes frames where the object cannot be detected.

Results In Table 1, we demonstrate that our method adds control to the model. We verify that our method enables spatio-temporal control, as evidenced by the lower CD and higher (upto 2.5x) AP@50 scores on both the IMC and ssv2-ST dataset. This means that the generated objects are close to the true centroid of the input mask, and their shape

Figure 5. **PEEKABOO with a moving mask:** As demonstrated, our method can mimic the input mask trajectories to generated spatio-temporally controlled videos with realistic motions. For *e.g.* in the last row, the wolf is jumping following the mask on the left.

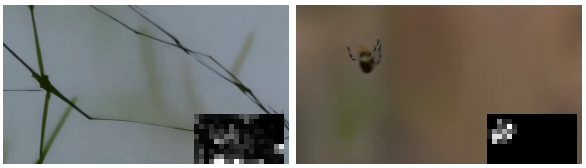
and size are also consistent with the input mask. We observe significant jump in mIoU score with PEEKABOO across different models, highlighting superior spatio-temporal control achieved through PEEKABOO. Finally, we note that PEEKABOO has a higher coverage than the baseline models, indicating that our method is also able to generate objects when the base model could not do so.

5.1.2 Quality control

While the above datasets provide evidence for PEEKABOO's spatio-temporal control, we also benchmark our method on MSR-VTT[39] – a large scale video generation dataset – to evaluate the quality of videos generated. We benchmark PEEKABOO for evaluating quality control using Fréchet Video Distance score (FVD) metric [34]. FVD is calculated based on I3D model trained on Kinetics-400 dataset [5]. Following previous works, we evaluate on the test-set of MSR-VTT containing 2900 videos by randomly sampling one of the 20 captions for each video. We demonstrate the versatility of our method by using bounding boxes generated by GPT-4. We query GPT-4 to generate series of locations for the foreground object depending on the prompt. We evaluate on ModelScope model and compare the scores with PEEKABOO. Table 2 shows that PEEKABOO increases the quality of generated while providing spatial control during video generation. The performance of these methods is also better than other baselines, indicating that PEEKABOO can be integrated in an automated pipeline to use GPT-4 generated bounding boxes and output a coherent video.

5.1.3 Ablation analysis

A Spatio-Temporal attention block consists of three types of attention layers– Spatial, Cross and Temporal. PEEKABOO applies masking on all three layers, however, the effect of each mask on the generation quality is different. In this section, we experiment with PEEKABOO by disabling masking for each attention layer one-by-one. We evaluate the AP50 score for ModelScope and ZeroScope on the IMC dataset, as shown in Figure 4. The performance drops massively when any one of the attention mask is not provided. We observe that not passing M_{CA} hurts the control the most. This is explained by the fact that main object's text token will not focus its attention at the bounding box location, leading to the object being generated at a different location. Surpris-



(c) A spider descending on its web.

(f) A croissant on a wooden table.

Figure 6. **Overcoming model failures**: Frames on the left are generated by zero-scope, and frames on the right are generated by PEEKABOO. Inset in the first row are cross-attention map between the word "spider" and the pixels in the video frame. We can generate objects that are otherwise omitted from the video by the base model. The attention maps also show that explicit masking leads to better generation. The second row depicts a numeracy failure of the baseline where PEEKABOO can control the number of objects.

ingly, not passing M_{TA} is worse than not passing M_{SA}. We conjecture that removing spatial attention mask leads to degraded videos, while removing the temporal attention mask leads to the loss of temporal control. Since the latter model still generates higher quality objects at incorrect locations, it has a lower AP50 score. We notice that the Coverage of the model after removing M_{SA} is much less than the Coverage of the model after removing M_{TA}, providing evidence in support of our hypothesis.

5.2. Qualitative Results

In Figure 1, we present examples of videos generated by PEEKABOO applied on ZeroScope [35]. As demonstrated, the videos follow the bounding box input. Through these qualitative results, we highlight the versatility of bounding box input in capturing the shape, size, location and motion, and show how our method can utilize this information interactively.

Static spatial control. Figure 3 shows videos where the object is statically located in the frame. Our method can control the position of the object, and can also change the size of the object as specified by the user through a bounding box.

Dynamic spatial control. Figure 5 present videos where the main subject is moving on a desired path. Our method generated realistic looking movements for various motion trajectories. The temporal masking of our method also enables it to handle cases where the mask disappears mid-way

(c) A kite flying in the sky. (f) A ship docked at a port.

Figure 7. **Text to Image synthesis:** We augment Stable-Diffusion v2.1 with PEEKABOO to produce images with spatial control including the size and location of the objects. Inset images are the masks passed to the model. Best viewed when zoomed in.

through the scene, as is the case in the first row in Figure 1, while the spatial and cross-attention masking ensures spatial coherence of the generated frames with the input bounding boxes.

Overcoming model failures. Diffusion models can have a bias on their generation capabilities depending on their training data. However, we observe that PEEKABOO can suppress those biases and produce high quality generation by forcing the model to generate foreground object at a specific location. In Figure 6, we present results of prompts where the original model fails to produce the foreground object however, our method can produce the object in the user specified location and motion. The inset figures in Figure 6 reveal the reason for this – while the cross-attention corresponding to the word "spider" is diffused across the entire canvas in the original model, PEEKABOO focuses this attention on the desired region. Further, Figure 6 depicts the example of hallucination by generation model where the subject was generated multiple times. Again, PEEKABOO solves this issue due to spatial-attention mask and cross-attention at a specific location.

Text to image synthesis While PEEKABOO was designed for video synthesis, it can be easily modified and work for the task of Text-to-Image synthesis. Figure 7 shows the versatility of our method. We generate images using Stable-Diffusion v2.1 [30] and gained spatial control through PEEKABOO. We observe that for the same prompt and initialization seed, PEEKABOO is able to control the location of the subject making the generation process interactive. Please refer to appendix for more results.

6. Conclusion

In this work, we explore interactive video generation. We hope that this work will inspire more research in this area. To this end, we propose a new benchmark for this task and PEEKABOO, which is a training-free, no latency overhead method to endow video models with spatio-temporal control. Future work involves exploring PEEKABOO for

image-to-video generation, video-to-video generation and long form video generation.

References

[1] Aishwarya Agarwal, Srikrishna Karanam, K J Joseph, Apoorv Saxena, Koustava Goswami, and Balaji Vasan Srinivasan. A-star: Test-time attention segregation and retention for text-to-image synthesis, 2023. 2, 3

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. 3

[3] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023. 5

[4] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing, 2023. 2, 3, 4

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 7

[6] Tsai-Shien Chen, Chieh Hubert Lin, Hung-Yu Tseng, Tsung-Yi Lin, and Ming-Hsuan Yang. Motion-conditioned diffusion model for controllable video synthesis, 2023. 2

[7] 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, 2023. 2

[8] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021. 4

[9] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation, 2022. 4

[10] Dave Epstein, Allan Jabri, Ben Poole, Alexei A. Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation, 2023. 2, 3

[11] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models, 2023. 2

[12] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850, 2017. 5, 13

[13] J Ho, T Salimans, A Gritsenko, W Chan, M Norouzi, and DJ Fleet. Video diffusion models. arxiv 2022. *arXiv preprint arXiv:2204.03458*. 2

[14] 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. 2

[15] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2, 6

[16] Yaosi Hu, Zhenzhong Chen, and Chong Luo. Lamd: Latent motion diffusion for video generation, 2023. 2

[17] Hanzhuo Huang, Yufan Feng, and ChengShi LanXu JingyiYu SibeiYang. Free-bloom: Zero-shot text-to-video generator with llm director and ldm animator. *arXiv preprint arXiv:2309.14494*, 3, 2023. 2

[18] 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, 2023. 1, 2

[19] Feng Li, Hao Zhang, Huaizhe xu, Shilong Liu, Lei Zhang, Lionel M. Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation, 2022. 4

[20] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation, 2023. 2, 3

[21] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. Llm-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models, 2023. 2, 5

[22] Long Lian, Baifeng Shi, Adam Yala, Trevor Darrell, and Boyi Li. Llm-grounded video diffusion models. *arXiv preprint arXiv:2309.17444*, 2023. 2

[23] Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning, 2023. 5

[24] Farzaneh Mahdisoltani, Guillaume Berger, Waseem Gharbieh, David Fleet, and Roland Memisevic. On the effectiveness of task granularity for transfer learning. *arXiv preprint arXiv:1804.09235*, 2018. 2, 5, 13

[25] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *European Conference on Computer Vision*, pages 728–755. Springer, 2022. 5, 13

[26] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 2

[27] John Mullan, Duncan Crawbuck, and Aakash Sastry. Hotshot-XL, 2023. 3

[28] Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing, 2023. 2, 3

[29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 3

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 2, 8

[31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 2, 3

[32] 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, 2022. 1, 2, 3

[33] 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*. 3

[34] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 7

[35] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023. 1, 2, 3, 6, 8

[36] 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, 2023. 2

[37] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, 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, 2023. 1

[38] Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jinbin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei Huang, Yuanxi Sun, Rui He, Feng Hu, Junhua Hu, Hai Huang, Hanyu Zhu, Xu Cheng, Jie Tang, Mike Zheng Shou, Kurt Keutzer, and Forrest Iandola. Cvpr 2023 text guided video editing competition, 2023. 2

[39] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2016. 7

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

[41] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models, 2022. 2, 6

[42] Xueyan Zou*, Zi-Yi Dou*, Jianwei Yang*, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Jianfeng Wang, Lu Yuan, Nanyun Peng, Lijuan Wang, Yong Jae Lee*, and Jianfeng Gao*. Generalized decoding for pixel, image and language. 2022. 4

Table 3. **Ablation study on PEEKABOO** : We evaluate ModelScope without various attention masks on our user defined dataset. We find that each component of our method impacts the performance significantly

Model	mIoU % (↑)	Coverage % (↑)	CD (↓)	AP50 % (↑)
ModelScope+PEEKABOO	36.1	96.6	0.13	33.3
-w/o Cross Attn Mask	14.2	93.3	0.27	5.7
-w/o Self Attn Mask	19.5	87.7	0.30	16.5
-w/o Temp Attn Mask	19.7	96.6	0.25	9.9
ZeroScope+PEEKABOO	36.3	96.3	0.12	33.8
-w/o Cross Attn Mask	13.3	78	0.23	6.7
-w/o Self Attn Mask	25.6	83	0.21	28.7
-w/o Temp Attn Mask	25.1	91	0.18	18.6

A. Implementation details

ModelScope - We generate videos of 256x256 resolution, and 16 frames. We fix the fixed step t to be 2 for all generations for ssv2-ST and 4 for IMC generation, and diffusion steps to be 40. For numbers on IMC, we generate 24 frames. In the quality evaluation experiments Table 2, we re-evaluated ModeScope performance on our selected set of prompts from the MSR-VTT dataset. PEEKABOO generation results are for videos generated with fixed step t equal to 2 of 40 steps.

ZeroScope - We generate videos of 320 x 576 resolution, and 24 frames. We fix the fixed step t to be 2 for all generations for ssv2-ST and 4 for IMC generation, and diffusion steps to be 40.

B. Ablation studies

B.1. Sensitivity to t

In fig 8, we present results on varying the t parameter for generation on the IMC dataset. As t increases, AP50 increases, but coverage decreases

B.2. More results on masking

In Tab 3, we present detailed results from Fig4.

C. More videos

We have uploaded the videos of the results presented in the main paper with our supplementary material. We also append more video results in the supplementary material for the reader.

D. Dataset Curation and filtering

D.1. IMC

D.1.1 Prompts

List of prompts:

- A woodpecker climbing up a tree trunk.
- A squirrel descending a tree after gathering nuts.
- A bird diving towards the water to catch fish.
- A frog leaping up to catch a fly.
- A parrot flying upwards towards the treetops.
- A squirrel jumping from one tree to another.
- A rabbit burrowing downwards into its warren.
- A satellite orbiting Earth in outer space.
- A skateboarder performing tricks at a skate park.
- A leaf falling gently from a tree.
- A paper plane gliding in the air.
- A bear climbing down a tree after spotting a threat.
- A duck diving underwater in search of food.
- A kangaroo hopping down a gentle slope.
- An owl swooping down on its prey during the night.
- A hot air balloon drifting across a clear sky.
- A red double-decker bus moving through London streets.
- A jet plane flying high in the sky.
- A helicopter hovering above a cityscape.
- A roller coaster looping in an amusement park.
- A streetcar trundling down tracks in a historic district.
- A rocket launching into space from a launchpad.
- A deer standing in a snowy field.
- A horse grazing in a meadow.
- A fox sitting in a forest clearing.
- A swan floating gracefully on a lake.
- A panda munching bamboo in a bamboo forest.
- A penguin standing on an iceberg.
- A lion lying in the savanna grass.
- An owl perched silently in a tree at night.
- A dolphin just breaking the ocean surface.
- A camel resting in a desert landscape.
- A kangaroo standing in the Australian outback.
- A colorful hot air balloon tethered to the ground.

D.1.2 Generating the bounding Boxes

Given the set of prompts, we annotate the main subject in the prompt. Further, the prompts are classified as stationary/moving, along with the object's aspect ratio as square, vertical rectangle, or horizontal rectangle. Specifically, the aspect ratio values are $1:1$, $4:3$, $3:4$ respectively. For prompts with movement, we also classify movement into up/down, left/right or zig-zag.

Three sets of bounding boxes are generated for each prompt. The starting co-ordinate of the bounding box is chosen randomly from 9 centroids of a 3x3 grid that the canvas is divided into. The speed is randomly chosen from 5-20 for moving prompts. The movement direction is randomly flipped as well. The bonding box size is chosen as 0.25 or 0.35 of the canvas size. We then generate a bounding box for each frame according to the random parameters, adding a small jitter for each pixel is well. For moving prompts, the starting location is one of 6 centroids, omitting

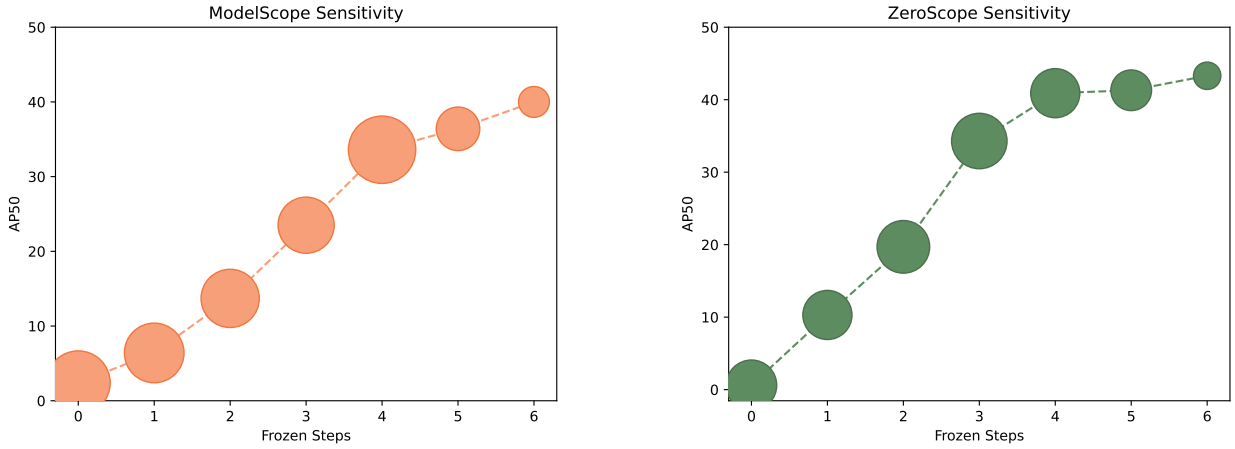


Figure 8. **Sensitivity to frozen steps** We plot AP50 against number of frozen steps t for ModelScope and ZeroScope. The radius of the marker is proportional to the coverage. We find that increasing t increases AP50 at the risk of losing coverage, i.e. degrading quality.

the centroids which align with the direction of motion. We will release the code for generating this dataset as well.

D.2. ssv2-ST

Filtering - We use Something-Something v2 dataset [12, 24] to obtain the generation prompts and ground truth masks from real action videos. We filter out a set of 295 prompts. The details for this filtering are in the appendix. We then use an off-the-shelf *OWL-ViT-large* open-vocabulary object detector [25] to obtain the bounding box annotations of the object in the videos. This set represents bounding box and prompt pairs of real-world videos, serving as a test bed for both the quality and control of methods for generating realistic videos with spatio-temporal control. We filter out the prompts such that they contain a single foreground object and obtain the bounding boxes or masks for the videos. We also further filter out videos with 0 bounding boxes.

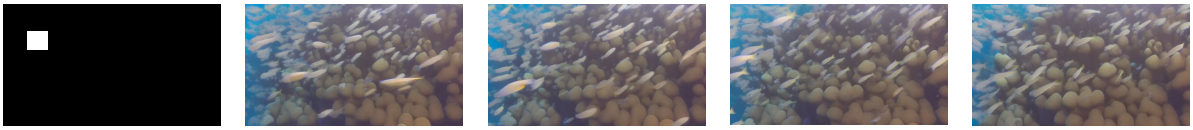
Post-processing bounding boxes - We downsample videos in ssV2 to 5fps and 224x224 resolution. For each video, we consider the first 24 frames for computing bounding boxes. We use OwL-ViT/B16 for getting the bounding boxes of the first 24 frames. Due to frame jittering and low resolution, we observe that obtained masks were not consistently calculated for each frame. Hence, we interpolated the masks between two successive frames. Our final test set contains 295 prompts and masks pairs. We pass the first 16 of these boxes to ModelScope, and all 24 of them to ZeroScope

E. Limitations

In Fig 9, we depict three typical failure modes of our method. These usually happen because there is a mismatch between the prior and the input mask, *ie.*, the bounding boxes should be of sensible size that align with the training data of the model. Further, the generation usually fails for cases where the base model is bad at the target prompt. Moreover, the movement introduced through interactive control should align with the input text prompt.

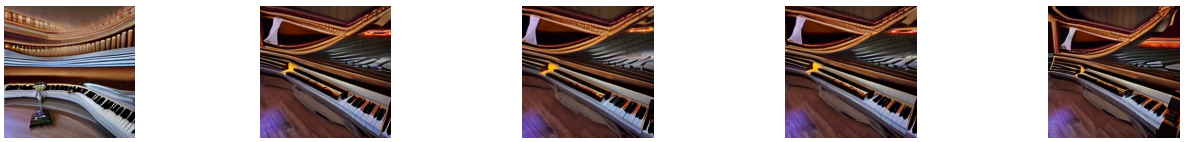
F. Societal Impact

This is a work on controllable video generation and not video generation itself. It is possible that the base model itself reflects some societal biases of the training set which will be propagated with the work. It also inherits the potential for misuse that other such video generation works have.



(a) A school of fish in the ocean.

(b) A rocket launching into space.



(c) A grand piano in a hall.

Figure 9. **Our Failure modes**: Top row shows a failure mode because the mask is too small for the subject. Middle row shows a failure model where the object does not move much, since the direction of motion of the mask contradicts that of the text. Bottom row shows a case where the model inherits a bad generation of the base model.