Corgi: Cached Memory Guided Video Generation

Xindi Wu[1,2*] Uriel Singer[1] Zhaojiang Lin[1] Xide Xia[1] Andrea Madotto[1]
Yifan Xu[1] Paul Crook[1] Xin Luna Dong[1] Seungwhan Moon[1]
[1]FAIR, Meta & Meta Reality Labs [2] Princeton University

Figure 1. **Illustration of our proposed method *Corgi* for multi-scene video generation.** Given a few input images of the target subject and an instruction prompt (*left*), *Corgi* – our multi-scene video generation method can generate consistent, faithful and diverse videos (*right*) conditioned on the generated intermediate story prompts.

Abstract

Text-to-Video generation has achieved remarkable progress with the rise of diffusion models. In this work, we introduce Cached Memory-Guided Video Generation (Corgi), aiming to generate multi-scene videos with arbitrary number of video clips, conditioned on input images and instruction prompts. This is a challenging task, as traditional T2V methods often struggle to maintain the quality of longer videos due to the difficulties in preserving visual context from earlier scenes. We address this by introducing a cached memory mechanism that stores the key frames. Our multi-scene video generation process is explicitly conditioned on the cached memories to avoid forgetting the visual appearance of target subjects. Corgi shows significant improvement in multi-scene video generation compared to the prior art, with up to 59.2% in long-term consistency and 7.6% in diversity.

1. Introduction

Recent advances in Text-to-Video (T2V) generation have enabled diffusion models [12, 33] to generate coherent videos from text descriptions. However, most of these methods are limited to generating videos with a single scene. Multi-scene video generation, which creates videos that span multiple scenes, is still a challenging task that has not been fully explored in the current literature. Despite its importance in applications such as filmmaking or game design, where there is a demand for maintaining consistency and character appearances across multiple video clips, there remains a significant gap between the current state-of-the-art and the desired capabilities. To tackle multi-scene video generation, we propose *Corgi*, with the cached latent memory bank as a core component, aiming to generate arbitrarily long videos by concatenating multiple video clips.

Why is it hard? *Multi-scene video generation*, the process of generating multi-scene long videos with multimodal inputs (Fig. 1), primarily faces challenges in consistency, faithfulness, and diversity. First, the **consistency** constraint

*Work partially done at Meta

is two-fold: long-term and short-term. Long-term consistency emphasizes maintaining global visual style and subject continuity across all video clips, while short-term consistency focuses on smooth transitions between consecutive frames within one video clip, avoiding generating flickering, low-quality videos with unrealistic motion changes. Second, **faithfulness** requires that the generated content is aligned with both image and text inputs. Balancing control between vision-language conditions is crucial. Lastly, the results generated should demonstrate **diversity**. While current open-source video generation methods are capable of creating realistic animations based on Text-to-Image (T2I) models, they often lack diversity in both motion and visual appearance. For example, the generated results may still resemble static images with minimal movement, or subjects may consistently face the same direction, indicating a limitation in achieving diverse and dynamic visualizations. All of these make multi-scene video generation challenging.

Corgi. In this work, we propose *Corgi*, a novel framework designed to generate multi-scene videos guided by cached memories. *Corgi* is inspired by neuroscience research on how human brains remember long videos [14]. Key repeating moments appear to cause similar activations in the viewers' brains and thus help them understand the storyline. We propose a multi-scene video generation method where key frames, serving as core memories, are generated first and stored in a cached latent memory bank. Given our goal of generating arbitrarily-long multi-scene videos, we fine-tune the T2I base model to encode the visual appearance of input reference images. We then cache these visual memories in a latent bank as an intermediate step. This memory-guided generation process thus allows the multi-scene video output to be consistently and faithfully conditioned on these latents stored in the bank. Moreover, we select diverse latents from the cached latent bank to improve the overall diversity of the generated videos to avoid repetitiveness. We summarize our **main contributions** as follows:

1. We introduce *Corgi* to generate *multi-scene videos* guided by cached memory and subject finetuning.
2. To ensure consistent, faithful, and diverse output, we propose a *cached memory mechanism*. Latents, or 'memories', are selectively stored as key frames.
3. We demonstrate empirically that our approach is effective and outperforms SOTA methods across key metrics for high-quality multi-scene video generation. *Corgi* improves long-term consistency by 59.21% and diversity by 7.57%. We further conduct human evaluation which aligns with our quantitative results observation.

2. Related Work

Text-to-Video Generation. Recent advancements in diffusion-based T2I generation [20, 23, 25, 28, 30, 38], have led to the production of high-quality images. Building on T2I models, T2V generation also shows promising results. Make-A-Video [31] extends the T2I model to T2V with a spatio-temporal diffusion model and super-resolution techniques. Align-Your-Latents [2] trains separate temporal layers in a T2I model. AnimateDiff [7] shows impressive results with motion module which can be used to bridge the gap between the T2I and T2V models. Text2Video-Zero [15] offers a training-free animation approach via latent wrapping given a predefined affine matrix.

However, those video generation methods are limited to short video generation and struggle with generating long coherent videos across multi-scenes.

Long Video Generation. In order to generate coherent long videos, recent works proposed hierarchical architectures and extrapolation methods [8, 35, 40, 43]. Phenaki [34] uses a transformer-based method and masked tokens to generate variable-length videos, compressing videos into discrete tokens with causal attention. NUWA-Infinity [40] and NUWA-XL [43] explore autoregressive and diffusion over diffusion approaches, respectively, for patch generation and long-term coherence. Animate-A-Story [10] addresses inconsistencies by introducing a retrieval-augmented pipeline and TimeInv to finetune on personalized concepts. Gen-L-Video [36], a tuning-free method, splices short video sub-segments under multiple text conditions for smooth extensions. Freenoise [26] reschedules noise sequences for long-range correlation and uses window-based fusion for temporal attention. SEINE [5] introduces a random mask video diffusion model to generate transitions based on textual descriptions.

Unlike existing long video generation approaches, which focus on generating more frames simultaneously, we sample the videos clip-by-clip sequentially to promote long-term inconsistency. Following Gen-l-video [36], we compare our method with existing methods in Tab. 1.

Subject-driven Customized Generation. Customization (or personalization) image/video generation usually condition on a few user-provided reference images. Several works have explored customized image generation with pre-trained diffusion models, where the unseen visual subjects will be embedded in the output space. DreamBooth [29] finetunes a diffusion model to learn the rare token. Textual Inversion [6] optimizes a learnable text token to represent a given subject. Custom Diffusion [18] proposes a lightweight parameter-efficient finetuning method to customize multiple concepts. Similarly, for customized video generation, previous works [13, 44] also explored the use of reference images to personalize the video diffusion model. VideoComposer [37] decomposes videos into different types of conditions, and jointly customizes the spatial and temporal patterns. VideoDreamer [4] focuses on

Table 1. **Comparison to different methods.** Key features evaluated include the ability to generate long videos (**Long**), the support for multi-text conditions guiding the generation (**Multi-Text Condition**), do not rely on a vast video corpus for training or generation (**Sample Efficient**), the utilization of parallel denoising techniques for efficiency (**Parallel Denoise**), versatility in generating a wide range of video types (**Versatile**), and the ability to generate personalized videos conditioned on subjects (**Image Condition**).

Method	Long	Multi-Text Condition	Sample Efficient	Parallel Denoise	Versatile	Image Condition
Tune-A-Video [41]	✗	✗	✗	✗	✗	✓
LVDM [9]	✓	✗	✗	✗	✗	✗
NUWA-XL [43]	✓	✓	✗	✓	✗	✗
Gen-L-Video [36]	✓	✓	✓	✓	✓	✗
Animate-A-Story [10]	✓	✓	✗	✓	✓	✓
FreeNoise [26]	✓	✓	✓	✓	✓	✗
Corgi (ours)	✓	✓	✓	✓	✓	✓

multi-subject driven customization with LoRA finetuning. CustomVideo [39] proposes a co-occurrence and attention mechanism to disentangle multi-subject customization.

Our work tackles multi-scene video generation with customization and focuses on global coherence across all video clips. Our approach finetunes the T2I base model following Dreambooth [29] and selectively stores intermediate latents in a memory bank to guide multi-scene video generation.

3. Method

We propose a multi-scene video generation method. In contrast to existing long video generation methods that generate thousands of frames simultaneously, which often results in temporal inconsistency, we sample the videos clip-by-clip sequentially. As shown in Fig. 2, given multimodal input of reference images and an instruction prompt, we finetune EMU [30], a diffusion T2I model (Stage 1), to generate and selectively cache the key frames (Stage 2) which are then used for sampling long videos clip-by-clip (Stage 3).

3.1. Problem Formulation

Given a set of reference images and an instruction prompt, we aim to generate multi-scene videos with a cached latent bank serving as the memory guidance. Concretely, let $X = \{x_1, x_2, ..., x_r\}$ denote a set of reference images, where r typically ranges between 3 and 5, let $\mathbf{P}_{\text{instruct}}$ denote the instruction prompt. With the reference images x_r^k as input, the Multimodal-LLM (MLLM) produces story prompts $\mathbf{P}_{\text{story}} = \{p_1, p_2, ..., p_n\}$. Our core component is a cached latent memory bank $B = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$, where each $\mathbf{z}_i$ represents the latent generated by the finetuned T2I model $\hat{\mathbf{x}}_\theta$ corresponding to the i-th story prompt p_i from the MLLM. Our goal is to generate n video clips $\{v_1, v_2, ..., v_n\}$ with N frames each ($N = 16$ in our setting), each based on one pair of story prompt p_k and cached latent $\mathbf{z}_k$. The final long multi-scene video is obtained by concatenating the n generated video clips.

3.2. Pipeline Overview

Our *Corgi* framework, featuring the core component of a cached latent memory bank, is illustrated in Fig. 2. It consists of three stages: finetuning (Stage 1, Sec. 3.3), caching (Stage 2, Sec. 3.4), and sampling (Stage 3, Sec. 3.4).

For stage 1, given a set of 3-5 reference images of a subject along with an instruction prompt $\mathbf{P}_{\text{instruct}}$, we use a Multimodal-LLM (MLLM) [1, 21] to produce a set of story prompts $\mathbf{P}_{\text{story}}$, leveraging its multimodal understanding capabilities. Concurrently, the base T2I model $\hat{\mathbf{x}}_\theta$ is finetuned using the reference images to generate a sequence of intermediate images $\mathbf{x}_{\text{gen},i} = \hat{\mathbf{x}}_\theta(\boldsymbol{\epsilon}, \mathbf{c}_i)$, each corresponding to different story prompts. $\mathbf{c}_i$ is the text embedding, generated by the text encoder with story prompt p_i.

For stage 2, the latent $\mathbf{z}_i$ of $\mathbf{x}_{\text{gen},i}$ is obtained via a pretrained Variational Autoencoder (VAE) [16] and is further stored in a cached latent memory bank. The latents serve as the basis of the initial image conditioning and, along with the story prompts, guide the video generation process.

For stage 3, we introduce motion dynamics to personalized T2I models via a pretrained off-the-shelf temporal transformer with self-attention blocks operating on the temporal axis. Trained on video clips, the temporal layers capture and distill motion priors. We obtain our multi-scene video with the concatenation of generated video clips.

3.3. Subject-Guided Finetuning

One challenge of multi-scene video generation is that the relationship between the reference images and text prompts is not always straightforward, and there is an inherent trade-off between these two conditioning sources. We propose customizing video generation at the T2I level without finetuning the temporal layers, since motion representations are agnostic to object appearance customization and primarily capture temporal changes in visual content.

We use subject-guided finetuning [29] for T2I model to encode the visual representations of reference images. Given 3 - 5 images of a new subject (e.g. "corgi"), our objective is to embed it into the output domain of the pretrained T2I model, similar to adding a new concept to the memory space of diffusion models. With pretrained T2I diffusion model $\hat{\mathbf{x}}_\theta$ and ground-truth image $\mathbf{x}$, the finetuning objective is:

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},\boldsymbol{\epsilon}',t}[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2 + \lambda w_{t'} \|\hat{\mathbf{x}}_\theta(\alpha_{t'} \mathbf{x}_{\text{pr}} + \sigma_{t'} \boldsymbol{\epsilon}', \mathbf{c}_{\text{pr}}) - \mathbf{x}_{\text{pr}}\|_2^2], \quad (1)$$

where $\mathbf{c} = \tau_\theta(p)$ is the conditioning vector, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is the initial noise, α_t, σ_t, w_t control the noise schedule and sample quality. $\mathbf{x}_{\text{pr}}$ is from the pretrained and frozen T2I model. $\mathbf{c}_{\text{pr}} := \tau_\theta(f(\text{"a [name of class]"}))$ is the text conditioning vector. We use unique tokens (e.g., V* for "A V* dog")) in text descriptions which has min-

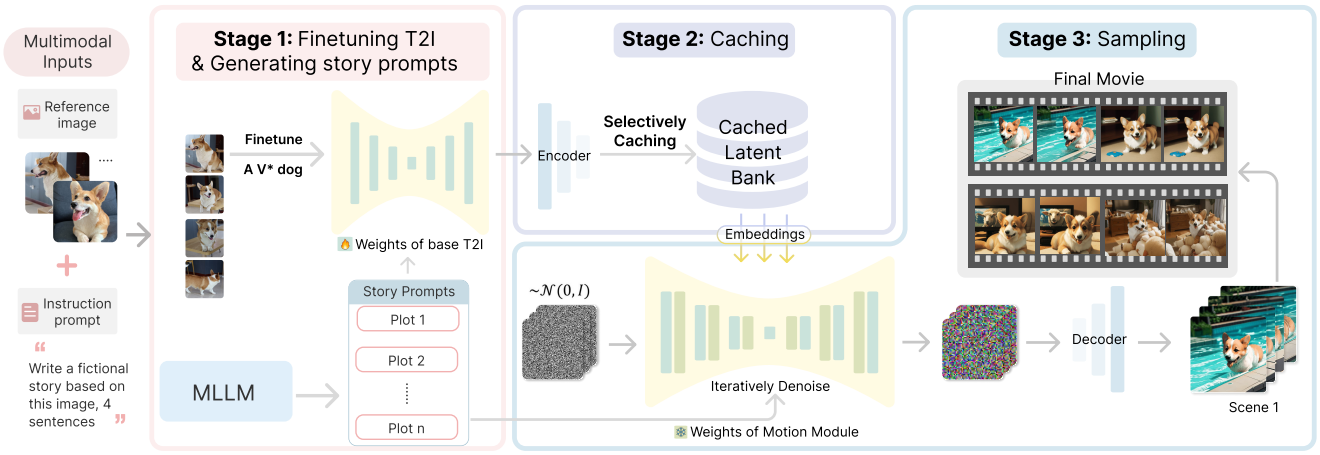

Figure 2. **Corgi pipeline**. Given a set of reference images and instruction prompt, we use Multimodal-LLM (MLLM) [21, 22] to generate arbitary number of story lines. We fine-tune the pretrained T2I diffusion model and store the generated key frame latents to a cached latent memory bank to maintain visual faithfulness and long-term consistency. The final multi-scene video is the concatenation of video clips based on each story prompt.

imal prior in both the pretrained text encoders and diffusion models. More details on the fine-tuning preliminaries can be found in Appendix Sec. B. As shown in Fig. 2, we perform subject-guided finetuning on T2I model in stage 1. During inference, intermediate images are generated conditioned on the story prompts, and their latents are cached in a cached latent memory bank to serve as keyframe (Sec. 3.4).

3.4. Cached Latent Memory Bank

We introduce the cache latent memory bank mechanism, a core component of our method, to improve faithfulness, consistency, and diversity of the generated videos. Since a single keyframe often falls short in representing the entire multi-scene long videos, our method generates subject-consistent multi-scene videos based on multiple latents as memories from the finetuned T2I base model and selectively cached in the bank.

We preserve long-term subject-consistency through cached latent memory bank B, and we explicitly encourage the generated results to share a cohesive global visual appearance across all video clips. As defined in Sec. 3.1, the latent bank $B = \{\mathbf{z}_1, ..., \mathbf{z}_n\}$ caches n latents. Each latent $\mathbf{z}_i$ is encoded via a pretrained VAE [16] from intermediate image $\mathbf{x}_{\text{gen},i}$. Intermediate images are generated by the finetuned T2I model customized for the reference images (Sec. 3.3) and conditioned on the story prompt p_i. We use a memory bank to cache the latents and preserve the visual representations of the target subject, enabling memorization capability and therefore allowing us to generate infinitely long multi-scene videos.

Coverage Caching. To avoid repetitiveness and improve diversity in backgrounds, poses, *etc.*, we selectively preserve and suppress information through the memory bank. The caching process begins by saving the latent of the first generated intermediate image $\mathbf{x}_{\text{gen},0} = \hat{\mathbf{x}}_\theta(\boldsymbol{\epsilon}, \mathbf{c}_0)$. For each subsequent generation conditioned on the same story prompt, we sample k times to obtain k latents (in practice we use $k = 10$). We compute the Euclidean distance from these new latents to the center of the existing latents in the bank, and the one that is farthest from its predecessors is then selected and cached. The coverage score can be formulated as:

$$D = \|\mathbf{z}_{\text{new}} - \mathbf{z}_{\text{centroid}}\|, \tag{2}$$

where $\mathbf{z}_{\text{centroid}} = \frac{1}{r}\sum_{i=1}^{r} \mathbf{z}_i$ is the center of all existing cached latents.

This coverage caching, conducted within the compact and manipulable VAE latent space, aims to maximize the coverage of the cached latents. The diversity is greatly improved and can be propagated to the multi-scene video generation next step. Furthermore, it introduces another layer of flexibility in user interaction and customization. This process maximizes coverage- and diversity-based measures in the feature space and the diverse range of latents contributes to consistent and diverse video output.

Cached Latent Conditioning. During sampling, we concatenate cached latents with random noise across video frames with a gradually decreasing weight over the frames. By leveraging the cached latents as control signals during the denoising process, we can achieve fine-grained control over generated images rather than relying *solely* on prompts. To condition on the cached latent signals during the video generation process, we add weighted $\mathbf{z}_i$, which corresponding to i_{th} video scene/clip, to all the frame noise ϵ_k and construct the input for the model as:

$$\hat{\epsilon} = \{\epsilon_1 + \lambda_1\mathbf{z}_i, \epsilon_2 + \lambda_2\mathbf{z}_i, ..., \epsilon_N + \lambda_N\mathbf{z}_i\}, \tag{3}$$

$\{\lambda_k\}_{k=1}^{N}$ are weights that control how much influence the cached latent will have on the generation of subsequent

frames and N is the maximum number of frames for each clip. Considering a minimum value m, the formula for λ_k would be:

$$\lambda_k = \lambda_0 - k \times \left(\frac{\lambda_0 - m}{N - 1} \right), \tag{4}$$

λ_k starts with the highest value for λ_1 (in our setting, λ_1=0.02) and then decreases at a constant rate for each subsequent frame. This is primarily because as the video evolves, the visual appearance is expected to change and move away from the first frame while still remaining faithful to the text descriptions. Thus, the weight of the cached latent should be decreased proportionally. The decreasing weight allows a smooth transition from strict visual consistency with the input references in early frames to increased diversity and flexibility in later frames, enabling a natural progression of the scene. While maintaining overall faithfulness to input subjects, strictly stick to their exact visual details is unnecessary. Text descriptions provide richer information and guide precise attribute & motion generation.

Clip-by-clip Sampling. We use pretrained temporal transformers which consists of several self-attention blocks along the temporal axis to insert motion dynamics into the finetuned T2I model. The attention mechanism allows the generation of the current frame to include information from other frames, capturing the visual content changes over time that constitute motion dynamics in an animation clip. Given the hidden state z_k of the k-th frame, the self-attention can be formulated as:

$$Q = W^Q z_k, K = W^K z_k, V = W^V z_k, \tag{5}$$

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^\top}{\sqrt{d_k}})V, \tag{6}$$

where W^Q, W^K and W^V are projection matrices. d_k is the dimension of the query and key vectors. To improve consistency and faithfulness of the generated video, we adjust the key and value vectors in the self-attention layers to include features from the corresponding cached latent $\mathbf{z}_i$ ($\oplus$ is the concatenation operation):

$$Q = W^Q z_k, K' = W^K (z_k \oplus \mathbf{z}'_i), V' = W^V (z_k \oplus \mathbf{z}'_i). \tag{7}$$

Cached latents $\mathbf{z}_i$ are passed through UNet to get noised and obtain features $\mathbf{z}'_i$ with correct spatial and channel dimensions. Then $\mathbf{z}'_i$ are concatenated with the intermediate UNet features of frame z_k along feature dimension. Concatenating intermediate UNet features with cached latents provides motion module with semantic information about target subjects, allowing it to capture semantics from cached latents. Serving as a "memory" of target subjects, this builds inter-frame correspondences, and guides consistent and faithful generation while allows motion dynamics.

Through this finetuning and caching process, our method is able to embed the unique features of target subjects and generate videos that are consistent across scenes and remain faithful to the reference images, thereby improving overall generation quality. After generating n video clips, we can then concatenate the clips into a long multi-scene video.

4. Experiments

In this section, we first discuss our experiment setup (Sec. 4.1) and evaluation metrics (Sec. 4.2). Then we conduct qualitative and quantitative evaluations and compare it with the latest state-of-the-art methods (Sec. 4.3). Furthermore, we perform ablation studies on cached latents to evaluate their impact on the diversity of generated videos (Sec. 4.4). Finally, we conduct a human evaluation study and provide the results (Sec. 4.5).

4.1. Evaluation Test-Bed

Datasets. We collect a dataset of 21 subjects, named MainCharacter21, which includes unique subjects such as dogs, cats, stuffed animals, etc. For each subject, we provide 3 - 5 sample images, along with an instruction prompt (See Sec. C in the Appendix for examples). To robustly measure the performance of our method, we generate five video clips for each story prompt and report their average performance.

Implementation Details. Our pipeline builds upon EMU [30] foundation set. We finetune the EMU T2I base model with data from MainCharacter21. The learning rate was set to 5×10^{-6}, with a batch size of 4, over a total of 400 optimization steps. The hyperparameter minimum value m in Eqn. 4, is adjusted based on the output performance and typically falls within the range of 0.005 to 0.001. Each video clip we generated contains 16 frames at a 512 $\times$ 512 resolution. We use the DDIM [32] sampler with 50 steps and classifier-free guidance [11] with a scale of 7.5. We save the video at a rate of 8 FPS.

Runtime. The subject-guided finetuning process takes roughly 2 minutes for one set of 3 to 5 images. On average, caching a single latent takes 33 seconds, as the process involves sampling $k = 10$ times and selectively caching one latent, with the caching process itself varying in duration. For clip-by-clip sampling, the average time spent generating each video clip is approximately 2 to 2.5 minutes.

4.2. Metrics

The three aspects we evaluate are: consistency (short/long-term), faithfulness (visual/textual), diversity (realistic shot change [43]).

Consistency: We evaluate the multi-scene video generation for both short-term and long-term consistency. *Long-term consistency* verifies that the subjects (a.k.a. protagonists) remain unchanged throughout the entire video, while *Short-term consistency* aims to have smooth transition between individual consecutive within a single video clip.

For *long-term consistency*, we segment target subjects using SAM [17], extract frame embeddings with CLIP ViT-B [27], and measure semantic similarity of the subjects across all video clips via average cosine similarity:

$$\frac{1}{n(n-1)}\sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n}\cos(\bar{s}_i,\bar{s}_j),\ \bar{s}_i=\frac{1}{N}\sum_{k=1}^{N}\text{CLIP}(s_{i,k}),$$

where n is the number of video clips. $\bar{s}_i$ is the average subject embedding for clip i. $s_{i,k}$ is the segmented subject in the k-th frame of the i-th video clip and N is number of frames within each video clip. $\cos(a,b)$ is the cosine similarity between vectors a and b and is defined as $\cos(a,b)=\frac{a\cdot b}{\|a\|\|b\|}$.

For *short-term consistency*, as the CLIP embedding metric is not constructed to distinguish between highly similar text descriptions, not to mention the same text prompt, we use DINO [3, 24] following [29] to compute the average cosine similarity of consecutive frames within one video clip using ViT-S/16 DINO embeddings. The self-supervised training objective of DINO encourages distinction of subject-wise unique features and thus helps measure how similar adjacent frames are. For each video clip, the short-term consistency score is:

$$\frac{1}{N-1}\sum_{i=1}^{N-1}\cos(\text{DINO}(f_i),\text{DINO}(f_{i+1})),$$

here, f_i is the frames of the video clips.

Faithfulness: We evaluate both textual and visual faithfulness of the generated videos. Following [5, 19], we quantify *textual faithfulness*, which is the semantic correlation between the generated videos and their corresponding story prompts. It is computed by the cosine similarity between the CLIP text embeddings and the CLIP image embeddings of each frame; then we average the scores from all frames to obtain the alignment score between a text and a generated video clip. Similarly, for *visual faithfulness*, we compute the average pairwise cosine similarity between the CLIP embeddings of each frame of the generated clip and the reference image. We do not use DINO embeddings because unlike CLIP, its visual self-supervised pretraining objective does not explicitly model subject semantics and focuses more on pixel-wise details. However, our goal of visual faithfulness is not to generate videos that look exactly like the subjects' poses from the reference images but to focus more on semantic similarity [3].

Diversity: Multi-scene videos should not only be consistent and faithful, but should also include realistic shot changes [43] to avoid monotony. We analyze different scenes and character actions within the video. Following Lamp [42], we use generation diversity metrics to evaluate the distinctiveness of consecutive video clips. Each video is represented by the average CLIP image embedding of all of its frames. We then calculate and average the cosine distances across all pairs of videos. Lower scores mean less similarity and better diversity. Each video is represented by the average CLIP image embedding of all of its frames. *Diversity* is calculated by the average cosine distance across all video pairs:

$$\frac{1}{n(n-1)}\sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n}(1-\cos(\bar{v}_i,\bar{v}_j)),\ \bar{v}_i=\frac{1}{N}\sum_{k=1}^{N}\text{CLIP}(f_{i,k}),$$

where $\bar{v}_i$ is the average embedding for the i-th video clip, and $f_{i,k}$ is the k-th frame of the i-th video clip.

Table 2. **Baseline Comparisons.** We compare our method with open-sourced multi-scene video generation methods Baseline 1 and Baseline 2. All results are presented as percentage values. Note that the visual faithfulness metric is not applicable to the baseline methods as they do not condition on image inputs. Our method significantly outperforms baselines by a large margin.

Method	Consistency (↓)		Faithfulness (↑)		Diversity (↑)
	Short-term	Long-term	Visual	Textual	
Baseline 1	30.53 ± 7.41	28.51 ± 5.49	–	32.76 ± 3.49	42.26 ± 2.98
Baseline 2	28.97 ± 4.12	32.83 ± 7.33	–	21.18 ± 0.48	49.12 ± 5.92
Corgi (ours)	**12.58** ± 5.76	**11.63** ± 5.23	**85.83** ± 6.38	**37.11** ± 4.27	**52.84** ± 3.28

4.3. Results

Baselines. We compare *Corgi* with two SOTA methods: (i) Baseline 1[1], which introduces a training-free noise rescheduling approach for long video generation, and (ii) Baseline 2, which treats long videos as temporally overlapping short videos and generates long videos with existing short T2V models.

Quantitative Results. For quantitative comparisons, following the evaluation metrics introduced in Sec. 4.2, we evaluate the consistency, faithfulness, and diversity of the generated results. As shown in Tab. 2, *Corgi* is especially strong in both short-term and long-term consistency, achieving average scores of 12.58% and 11.63%, respectively. It shows an improvement of 56.58% in short-term and 59.21% in long-term consistency, outperforming baselines by a large margin. Furthermore, while Baseline 2 shows a competitive edge in diversity with a score of 49.12%, our approach has a 7.57% increase over this next best-performing method, achieving a diversity score of 52.84%. Our experiments demonstrate the robustness of our method in generating consistent and faithful videos while also showing its capability to improve diversity, making it a well-rounded pipeline for multi-scene video generation.

Qualitative Results. In Figs. 3 & 4, we show a few samples of our generated results, as well as comparisons with the baseline methods. Fig. 3 presents multi-scene video frames generated using our Corgi method, highlighting our

[1]Baseline details have not been disclosed due to legal considerations.

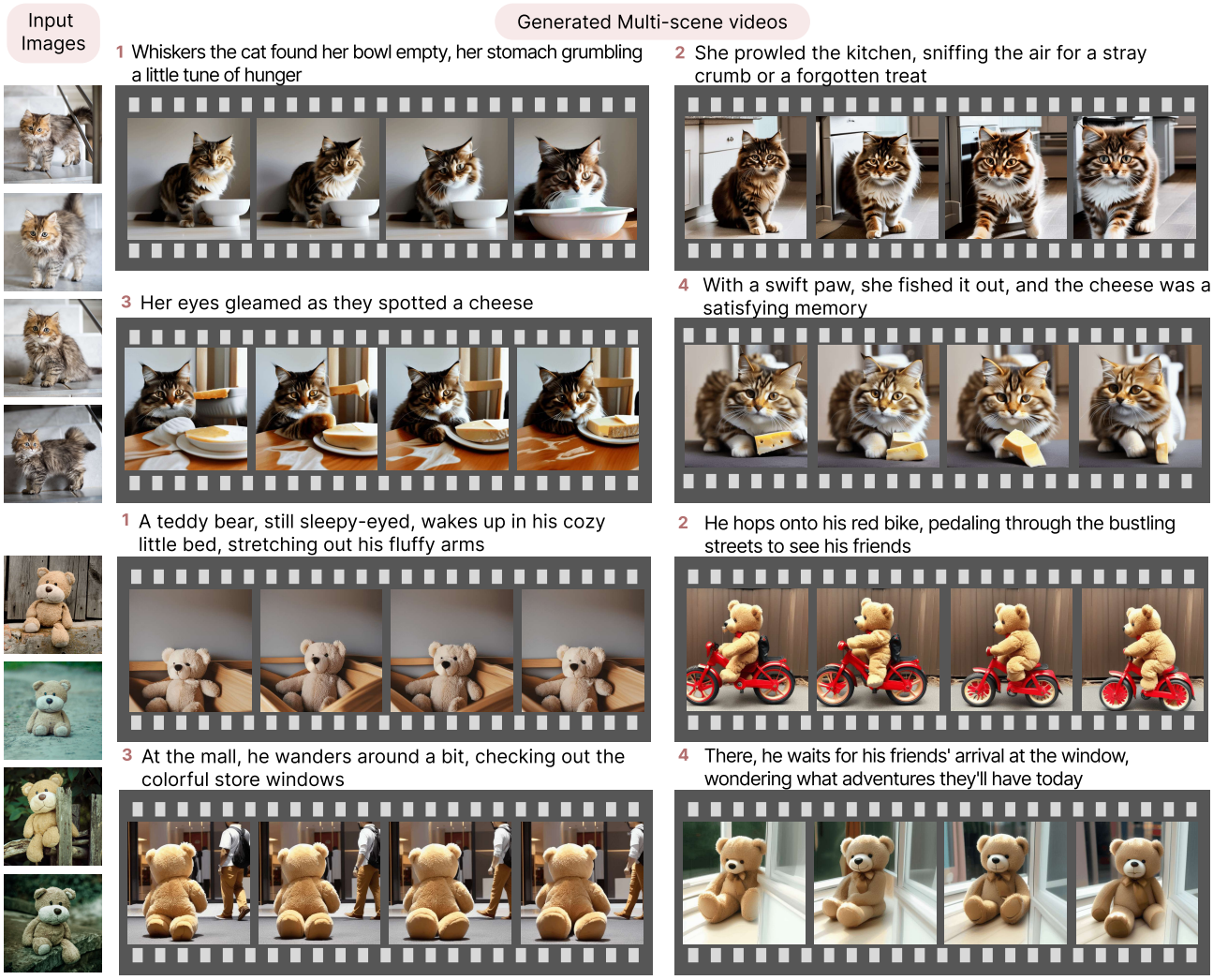

Figure 3. **Multi-scene video generation results.** We present two sets of multi-scene video sample frames generated via our Corgi method. The input reference images used for customization are provided (*left*), and the instruction prompt we used for MLLM is: `Based on this image, generate a 10-sentence story`. Due to space constraints, we only show 4 out of the 10 scenes and leave the rest in the supplementary folder. We provide the generated story prompts at the top of each video clip.

model's strength in maintaining consistency, faithfulness, and diversity. In Fig. 4, we offer a side-by-side comparison of video clips from Baseline 1 (*left*) and our *Corgi* (*right*), using the same multi-scene prompts for fair comparisons.

4.4. Ablation Study

We conduct an ablation study to measure the effectiveness of coverage selection strategy used during the caching stage. Specifically, we compare the performance of two variants: (1) latents saved without coverage score selection and (2) latents selected based on our proposed coverage caching mechanism. The primary goal of this ablation is to evaluate the impact of our caching strategies on the diversity and overall performance of the generated videos. We provide quantitative results in Tab. 3 and qualitative results in Fig. 5. The performance of both approaches demonstrated negligible differences in consistency and faithfulness, yet coverage-score selected latents significantly boost the diversity of multi-scene videos. These results suggest that our coverage selection based caching effectively propagates the cached image latent quality to multi-scene video generation and is a more effective strategy to prioritize diversity. As shown in Fig. 5, latents selected with coverage score help avoid generating videos with similar poses, sizes or facing directions, leading to an overall quality improvement. Additional ablation studies are provided in Appendix Sec. A.

4.5. Human Evaluation

We conduct human evaluations for our method against several baselines. Each pair of generated results was eval-

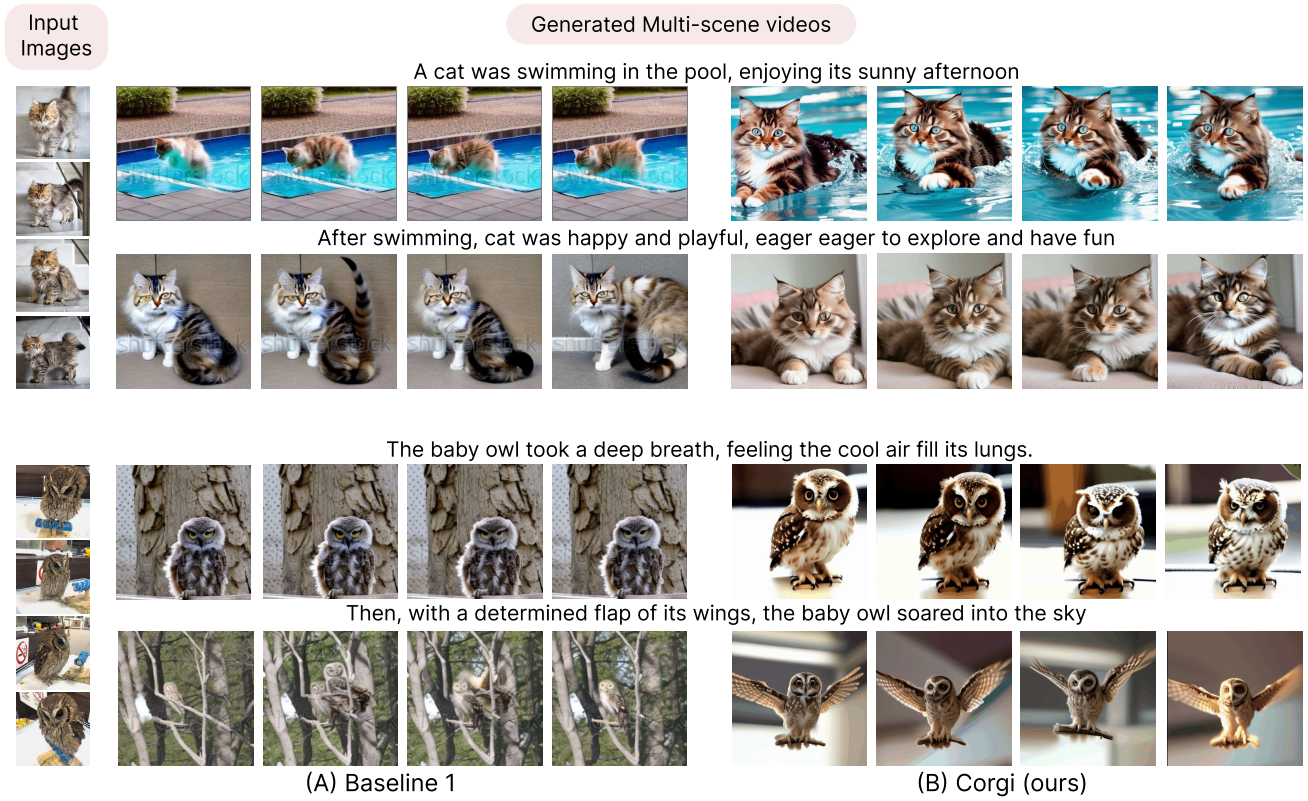

Figure 4. **Qualitative results.** We present side-by-side comparisons of video clips generated by Baseline 1 (*left*) and our *Corgi* method (*right*). We provided two sets of input images and use the same multi-scene prompts for both methods for fair comparisons. Note that Baseline 1 does not condition on the image inputs. Our method shows significantly better visual quality and realistic motion changes.

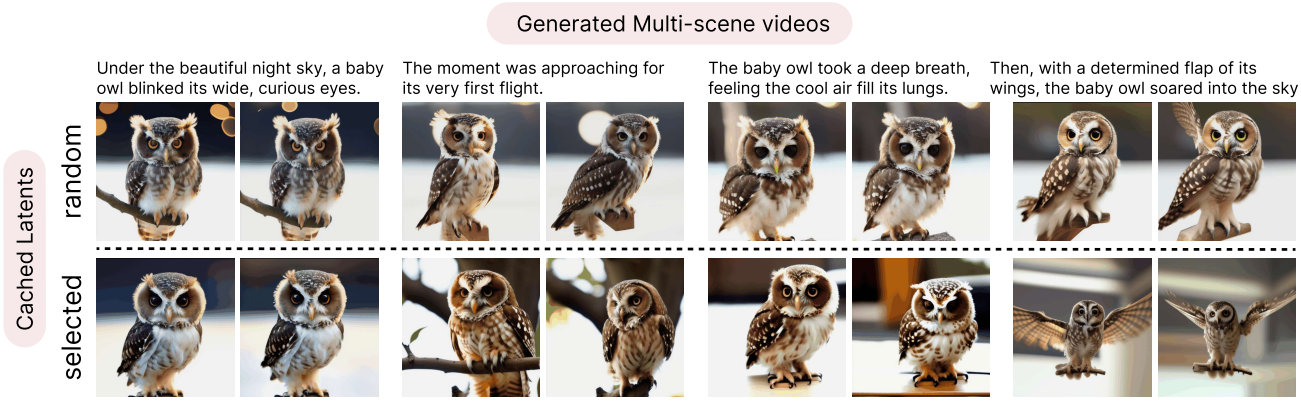

Figure 5. **Ablation study on Cached Latent Selection.** We examine two variants: (1) latents saved without coverage score selection **(Random)**, (2) latents selected based on our proposed coverage caching mechanism **(Selected)**. For **Random** ones, the owls in the generated samples all face roughly the same direction (*right*), while **Selected** ones introduces more diversity of poses (e.g., flying, sitting) and directions (*left*).

uated by five participants, including both experts in the field and individuals without specific background knowledge. Our evaluation set includes 21 pairs of generated results from *Corgi* and open-source baseline methods (Baseline 1 and Baseline 2), as well as generated multi-scene video samples from closed-source methods (three videos from Baseline 3 and five videos from Baseline 4), where we directly use the provided prompts for generation. For comparisons with Baseline 1 and Baseline 2, we collected 105 responses (21 video pairs, evaluated by 5 participants), and for Baseline 3 and Baseline 4, we had 15 (3 video pairs, evaluated by 5 participants) and 25 (5 video pairs, evaluated by 5 participants) responses, respectively. We mixed our generated results with those from baselines, present-

Table 3. **Ablation on Cached Latent Selection.** We conduct an ablation study on cached latents, comparing those without a coverage score selection mechanism (**Random**) and those with it (**Selected**). The results show that latents selected based on a coverage score significantly improve the video diversity.

	Consistency (↓)		Faithfulness (↑)		Diversity (↑)
Cached Latents	Short-term	Long-term	Visual	Textual	
Random	**11.64** ± 5.89	**10.85** ± 6.71	85.33 ± 5.91	36.58 ± 3.49	40.27 ± 4.12
Selected	12.58 ± 5.76	11.63 ± 5.23	**85.83** ± 6.38	**37.11** ± 4.27	**52.84** ± 3.28

Table 4. **Human Preference.** We conduct a human evaluation to compare our *Corgi* method against four baseline methods: Baseline 1 (**B1**), Baseline 2 (**B2**), Baseline 3 (**B3**), and Baseline 4 (**B4**). In each paired comparison, our method was preferred predominantly (over 50%) over the baselines across various metrics. It is important to note that **F** and **G** do not utilize input images for conditioning, hence visual faithfulness was not evaluated for these methods. For **B3** due to limited access to only one set of images used for generating a single video, we report the visual faithfulness score solely for this specific comparison.

Evaluation (%)		Ours > **B1**	Ours > **B2**	Ours > **B3**	Ours > **B4**
Consistency	Short	87.62	77.14	66.67	88.00
	Long	84.76	94.28	46.67	92.00
Faithfulness	Visual	–	–	40.00	96.00
	Textual	63.81	78.09	60.00	84.00
Diversity		63.81	70.48	53.33	84.00
Overall Quality		81.90	84.76	66.67	92.00

ing the participants with story prompts and corresponding videos generated by these methods in a randomized order. Participants were prompted to compare the consistency, faithfulness, diversity, and overall video quality of the multi-scene videos, asking, e.g., "Which video is more consistent/faithful/diverse/has higher quality?" We present the proportion of samples where a higher number of users preferred our examples as being better, as shown in Tab. 4. The results show that our *Corgi* method consistently outperforms the baseline methods across key metrics. Particularly notable are its high preference scores in both short-term and long-term consistency, as well as diversity score and overall video quality, with a remarkable 92% preference over Baseline 4 for overall quality. Although *Corgi* shows a lower preference in visual faithfulness and long-term consistency compared to Baseline 3, this may be due to the limited comparison set, as we had access to only one group of conditioning images from Baseline 3. These results show the effectiveness of *Corgi* in multi-scene video generation.

5. Limitations

While our method offers promising results in multi-scene video generation, it still has its limitations. For example, we observed that when novel subjects in the story prompts are not specified, e.g., in Fig. 6 (A), with only the corgi images as input, the generated results will merge the features of multiple subjects (corgi and squirrel). Another failure case we observed is if in the input images, there is always some part attached to the target subject (e.g., in Fig. 6 (B), the tree branch is attached to the owl), then this feature will be propagated via the cached latents to the final generated videos. Additionally, our diversity metric does not capture whether this diversity aligns with the intended story. As in some cases, it could be preferable for subsequent clips to have similar visual content. Quantifying "desirable" or "reasonable" diversity is subjective and context-dependent. An interactive UI is ideal but beyond our scope. Future work e.g. adaptive weighting or human-in-the-loop approaches for user-selected intermediate images could further improve quality. These challenges open up new opportunities for future research exploration.

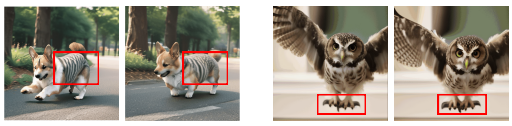

Figure 6. **Limitations.** Feature disentanglement for image-conditioned video generation still remains challenging. As shown in (A), the features of a corgi and a squirrel are mistakenly combined when the input images only includes the corgi. Additionally, in (B), the base T2I model's limitations in contextual understanding and a tendency to overfit to features that appear across all images used for fine-tuning result in incorrect feature attachment.

Negative Impact. While our method aims to enable multi-scene video generation, there is a risk that it could be exploited to create misleading or inappropriate content, which underscores the need for robust filters and stricter regulatory frameworks to prevent misuse in the future.

6. Conclusion

In summary, we propose *Corgi*, a multi-scene video generation method that takes multimodal inputs to create coherent multi-scene videos. We introduce a cached latent memory bank module to selectively store the customized latents from the finetuned T2I model and guide the multi-scene generation process. Our experiments show that *Corgi* can generate results that maintain a high level of consistency, faithfulness, and diversity throughout the entire multi-scene video. We hope that the cached latent memory bank can serve as an essential building block for multi-scene video generation, and the insights we have gathered can provide a strong basis for future research in this field.

Acknowledgments. We thank many people for their helpful discussion and feedback, listed in alphabetical order by last name: Meta GenAI (Rohit Girdhar, Sachit Menon, Ishan Misra), Princeton Visual AI lab (Allison Chen, Olga Russakovsky, William Yang, Tyler Zhu, Ye Zhu) and Carlos E. Jimenez, Tiffany Ling, Zhiqiu Lin, Zirui Wang, Eric Zelikman.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 3

[2] Andreas 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 6

[4] Hong Chen, Xin Wang, Guanning Zeng, Yipeng Zhang, Yuwei Zhou, Feilin Han, and Wenwu Zhu. Videodreamer: Customized multi-subject text-to-video generation with disen-mix finetuning. *arXiv preprint arXiv:2311.00990*, 2023. 2

[5] Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction. *arXiv preprint arXiv:2310.20700*, 2023. 2, 6

[6] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 2

[7] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 2

[8] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2022. 2

[9] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. 2022. 3

[10] Yingqing He, Menghan Xia, Haoxin Chen, Xiaodong Cun, Yuan Gong, Jinbo Xing, Yong Zhang, Xintao Wang, Chao Weng, Ying Shan, et al. Animate-a-story: Storytelling with retrieval-augmented video generation. *arXiv preprint arXiv:2307.06940*, 2023. 2, 3

[11] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[13] Yuming Jiang, Tianxing Wu, Shuai Yang, Chenyang Si, Dahua Lin, Yu Qiao, Chen Change Loy, and Ziwei Liu. Videobooth: Diffusion-based video generation with image prompts. *arXiv preprint arXiv:2312.00777*, 2023. 2

[14] Janne Kauttonen, Yevhen Hlushchuk, Iiro P Jääskeläinen, and Pia Tikka. Brain mechanisms underlying cue-based memorizing during free viewing of movie memento. *NeuroImage*, 172:313–325, 2018. 2

[15] 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

[16] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3, 4

[17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 6

[18] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 2

[19] Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. *arXiv preprint arXiv:2309.15091*, 2023. 6

[20] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. *arXiv preprint arXiv:2404.01291*, 2024. 2

[21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 3, 4

[22] Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024. 4, 3

[23] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 2

[24] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 6

[25] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 2

[26] Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. Freenoise: Tuning-free longer video diffusion via noise rescheduling. *arXiv preprint arXiv:2310.15169*, 2023. 2, 3

[27] 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. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 6

[28] 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*, pages 10684–10695, 2022. 2

[29] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 2, 3, 6, 1

[30] Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks. *arXiv preprint arXiv:2311.10089*, 2023. 2, 3, 5

[31] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 2

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

[33] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 1

[34] 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 description. *arXiv preprint arXiv:2210.02399*, 2022. 2

[35] Vikram Voleti, Alexia Jolicoeur-Martineau, and Chris Pal. Mcvd-masked conditional video diffusion for prediction, generation, and interpolation. *Advances in Neural Information Processing Systems*, 35:23371–23385, 2022. 2

[36] Fu-Yun Wang, Wenshuo Chen, Guanglu Song, Han-Jia Ye, Yu Liu, and Hongsheng Li. Gen-l-video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023. 2, 3

[37] 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. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[38] Zirui Wang, Zhizhou Sha, Zheng Ding, Yilin Wang, and Zhuowen Tu. Tokencompose: Grounding diffusion with token-level supervision. *arXiv preprint arXiv:2312.03626*, 2023. 2

[39] Zhao Wang, Aoxue Li, Enze Xie, Lingting Zhu, Yong Guo, Qi Dou, and Zhenguo Li. Customvideo: Customizing text-to-video generation with multiple subjects. *arXiv preprint arXiv:2401.09962*, 2024. 3

[40] Chenfei Wu, Jian Liang, Xiaowei Hu, Zhe Gan, Jianfeng Wang, Lijuan Wang, Zicheng Liu, Yuejian Fang, and Nan Duan. Nuwa-infinity: Autoregressive over autoregressive generation for infinite visual synthesis. *arXiv preprint arXiv:2207.09814*, 2022. 2

[41] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023. 3

[42] Ruiqi Wu, Liangyu Chen, Tong Yang, Chunle Guo, Chongyi Li, and Xiangyu Zhang. Lamp: Learn a motion pattern for few-shot-based video generation. *arXiv preprint arXiv:2310.10769*, 2023. 6

[43] Shengming Yin, Chenfei Wu, Huan Yang, Jianfeng Wang, Xiaodong Wang, Minheng Ni, Zhengyuan Yang, Linjie Li, Shuguang Liu, Fan Yang, et al. Nuwa-xl: Diffusion over diffusion for extremely long video generation. *arXiv preprint arXiv:2303.12346*, 2023. 2, 3, 5, 6

[44] Haoyu Zhao, Tianyi Lu, Jiaxi Gu, Xing Zhang, Zuxuan Wu, Hang Xu, and Yu-Gang Jiang. Videoassembler: Identity-consistent video generation with reference entities using diffusion model. *arXiv preprint arXiv:2311.17338*, 2023. 2

Supplementary Material
Corgi: Cached Memory Guided Video Generation

Overview

In this supplement, we first extend ablation study to analyze different aspects of *Corgi* and their influence on overall performance in Sec. A. Additional preliminary details for subject-guided finetuning are provided in Sec. B and details of our customized dataset MainCharacter21 are in Sec. C.

A. Additional Ablation Study

In the main paper, we provide ablation studies to evaluate the impact of coverage-based selective caching (Sec. 4.4). Here we ablate two other method design choices of *Corgi*: cached latent conditioning and clip-by-clip sampling.

Cached Latent Conditioning. In our proposed method, cached latent conditioning plays an important role in controlling the generation process across video clips. To evaluate the effectiveness of this design choice, we conduct ablation studies to compare different scenarios:

1. Removing linear weight degradation (as in Eqn. 4) and maintaining a constant degree of influence across all frames, thus $\lambda_k = 0.02$ (**Constant**).
2. Setting the initial weight (λ_0) too low while still maintaining the linear weight degradation, reducing the cached latent influence, which may result in generated videos that are not visually faithful to the input subjects, $\lambda_0 = 0.002$ (**Low**).
3. Setting the initial weight (λ_0) too high while still maintaining the linear weight degradation, resulting in cached latents having an excessive influence on the generated frames, potentially limiting diversity, $\lambda_0 = 0.5$ (**High**).
4. Using the default setting with linear weight degradation, $\lambda_0 = 0.02$ (**Linear**).

As shown in Tab. 5 and Fig. 7, linear weight degradation enables for a gradual transition, allowing the generated frames to deviate from the initial frame while still maintaining visual faithfulness to the input subjects. However, maintaining a constant degree of influence across all frames, without the linear weight degradation, leads to an overly rigid adherence to the cached latents. This affects the natural transition of the generated videos, resulting in minimal motion movement throughout the clips. Setting the cached latents weight too high limits diversity by overly constraining the content to the initial frame cached latents, while a too low weight diminishes visual faithfulness and consistency as frames have little influence from cached latents, deviating from earlier frames, both compromising overall video consistency. While constant weight outperforms others in terms of short-term consistency and visual faithfulness as expected, it significantly affected diversity and long-term consistency.

Table 5. **Ablation on Cached Latent Conditioning.** We compare different scenarios: constant weight (**Constant**), low weight (**Low**), high weight (**High**) and linear weight degradation (**Linear**). The results show that our proposed linear weight degradation approach achieves the optimal tradeoff of consistency, faithfulness, and diversity.

	Consistency (↓)		Faithfulness (↑)		Diversity (↑)
Weight Setting	Short-term	Long-term	Visual	Textual	
Constant	**7.42** ± 4.37	17.93 ± 5.02	**86.44** ± 8.24	35.94 ± 5.73	38.64 ± 6.74
Low	21.36 ± 6.15	23.48 ± 4.63	75.89 ± 8.06	32.18 ± 7.93	49.27 ± 5.15
High	8.57 ± 5.82	25.14 ± 4.85	54.38 ± 9.53	21.49 ± 3.81	34.96 ± 7.36
Linear (ours)	12.58 ± 5.76	**11.63** ± 5.23	85.83 ± 6.38	**37.11** ± 4.27	**52.84** ± 3.28

Clip-by-clip Sampling. Furthermore, we conduct an ablation study to evaluate the impact of the self-attention operation with cached latents concatenation in clip-by-clip sampling. Keeping the same experiment settings for other parts, we evaluate **w/ concatenation** (Eqn. 5) and **w/o concatenation** (Eqn. 7), the results are in Tab. 6. Our ablation study shows that incorporating the proposed cached latent concatenation for self-attention improves performance. When the cached latent concatenation was omitted for self-attention, the ability to preserve the visual appearance of the input subjects was largely weakened and it frequently results in jittery motion and object distortions (Fig. 8).

Table 6. **Ablation on Clip-by-Clip Sampling.** We conduct an ablation study on self-attention concatenation during clip-by-clip sampling, comparing scenarios with and without cached latent concatenation. The results show that with concatenation improves video quality and consistency. The ✓denotes using concatenation.

	Consistency (↓)		Faithfulness (↑)		Diversity (↑)
Concatenation	Short-term	Long-term	Visual	Textual	
✓	**12.58** ± 5.76	**11.63** ± 5.23	**85.83** ± 6.38	37.11 ± 4.27	**52.84** ± 3.28
	14.31 ± 6.58	12.95 ± 4.17	74.23 ± 7.82	**40.03** ± 5.22	50.17 ± 5.39

B. Finetuning Preliminary

Here we provide additional preliminary details for the subject-guided finetuning [29]. Diffusion models are a type of probabilistic generative model designed to learn data distributions. They achieve this by progressively denoising a sample initially drawn from a Gaussian distribution, effectively reducing its noise through each step of the process. As denoted in Sec. 3.2, with pretrained T2I diffusion model $\hat{\mathbf{x}}_\theta$ and conditioning vector $\mathbf{c} = \tau_\theta(\mathbf{p})$, and initial noise map $\boldsymbol{\epsilon}$ drawn from a normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, as well as the ground-truth image $\mathbf{x}$, the original training objective is:

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},t}\left[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2\right], \tag{8}$$

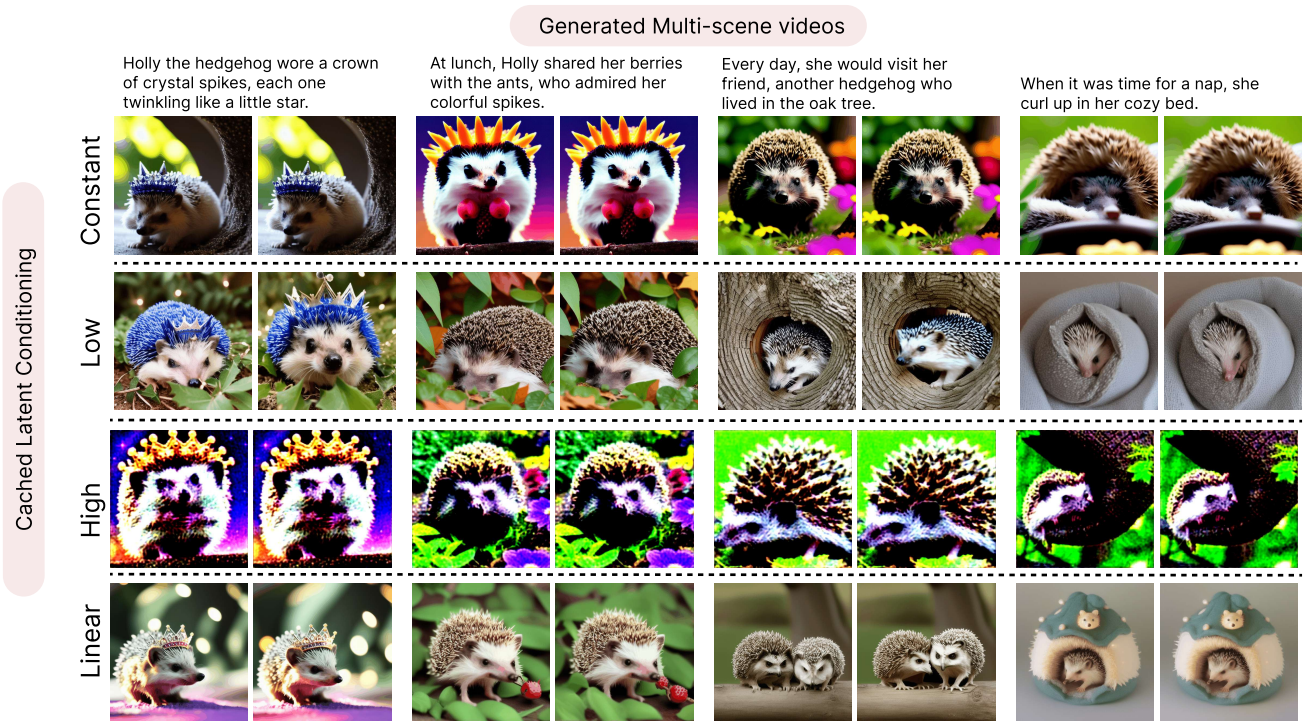

Figure 7. **Ablation study on Cached Latent Conditioning.** We examine different weight settings for cached latent conditioning: constant weight across all frames **(Constant)**, low weight **(Low)**, high weight **(High)**, linear weight degradation **(Linear)**. The **Linear** approach achieves the best balance between consistency, faithfulness, and diversity. **Constant** leads to overly rigid adherence and the videos have minimum motion and appear similar to static images rather than dynamic video sequences, **High** limits diversity and the generated results look unrealistic, and **Low** diminishes visual faithfulness to input subjects.

Figure 8. **Ablation study on Clip-by-Clip Sampling.** We compare the impact of cached latent conditioning on the generated videos. The model without cached latent **(w/o concatenation)** suffers from jittery motion and object distortions, while the model with cached latent **(w/ concatenation)** maintains visual appearance of input subjects and generates more stable and high-quality videos. This demonstrates the effectiveness of the proposed clip-by-clip sampling approach in preserving visual consistency and faithfulness to the input subjects.

α_t, σ_t, w_t control the noise schedule and sample quality. We follow Dreambooth [29] and leverage the class-specific prior preservation loss during finetuning:

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},\boldsymbol{\epsilon}',t}[w_{t'}\|\hat{\mathbf{x}}_\theta(\alpha_{t'}\mathbf{x}_{\text{pr}} + \sigma_{t'}\boldsymbol{\epsilon}', \mathbf{c}_{\text{pr}}) - \mathbf{x}_{\text{pr}}\|_2^2], \quad (9)$$

where $\mathbf{x}_{\text{pr}} = \hat{\mathbf{x}}(\mathbf{z}_{t_1}, \mathbf{c}_{\text{pr}})$ from the pretrained and frozen T2I model. $\mathbf{z}_{t_1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is random initial noise and $\mathbf{c}_{\text{pr}} := \tau_\theta(f(\text{"a [name of class]"}))$ is a conditioning vector. The loss of T2I finetuning is the combination of the both training objectives above:

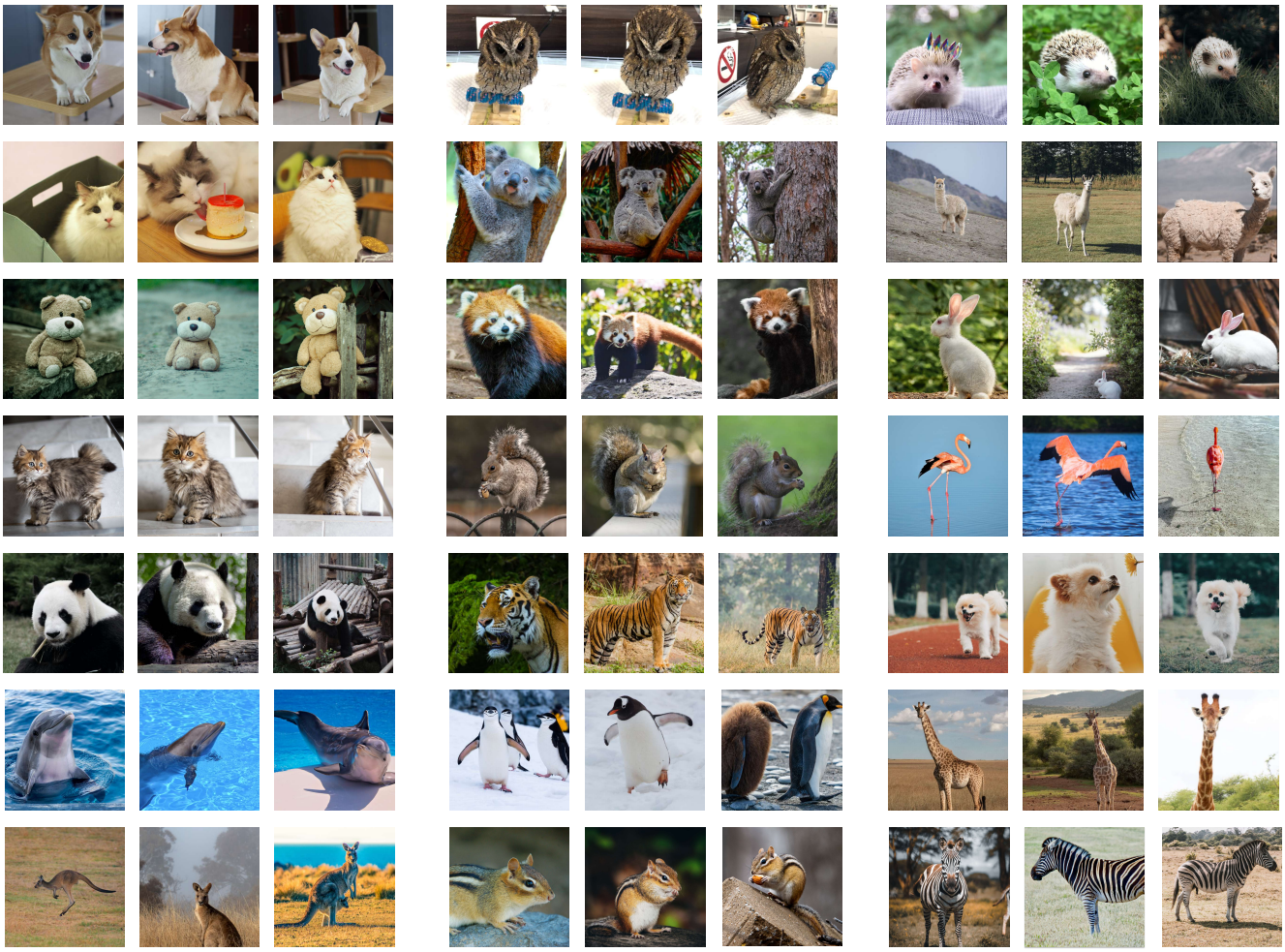

Figure 9. **MainCharacter21.** This figure illustrates our dataset MainCharacter21, including images from 21 distinct subjects, with three sample images per subject.

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},\boldsymbol{\epsilon}',t}[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2 + \lambda w_{t'} \|\hat{\mathbf{x}}_\theta(\alpha_{t'} \mathbf{x}_{\text{pr}} + \sigma_{t'} \boldsymbol{\epsilon}', \mathbf{c}_{\text{pr}}) - \mathbf{x}_{\text{pr}}\|_2^2]. \tag{10}$$

C. MainCharacter21

In our study, we introduce the **MainCharacter21** dataset, including 21 unique subjects, with each subject represented by 3 to 5 images. Fig. 9 shows three images of each subject. We show a list of sample instruction prompts (Tab. 7) used to generate story prompts, along with three example story prompts (Tab. 8, 9, 10) created by MLLM [21, 22] given instruction prompts. It is important to note that, as we use rare tokens (e.g. V*) plus subjects during the T2I finetuning stage, we similarly added the rare tokens to the story prompts before subjects and pronouns in the story prompts were adjusted accordingly during inference.

Table 7. Sample Instruction Prompts

`Inspired by the photo, write a story for a children's book, consisting of 7 sentences.`
`Write a 9-sentence tale about two individuals reuniting under surprising circumstances using the image as inspiration.`
`Narrate a 4-sentence adventure about discovering something invaluable, drawing inspiration from the image.`
`Craft a 5-sentence story about unexpected turns in life, drawing from the image's atmosphere.`
`Using the image as a foundation, write a 9-sentence tale about a life lesson.`

Table 8. Sample Story Prompts 1

Sidney the squirrel scurried around the park, his little heart full of glee. He found a perfect acorn, shiny and brown, right for tea. His fluffy tail flickered as he nibbled away, happy as can be. He played peek-a-boo with the children, who laughed merrily. Sidney had a secret stash, hidden under the oak tree. He'd jump from branch to branch, the leaves whispering, "Catch me!" His friends, the birds, would sing as he danced. When it rained, he snuggle in his cozy warm and dry. And as the stars appeared, Sidney would dream of tomorrow's joyous spree.

Table 9. Sample Story Prompts 2

Holly the hedgehog wore a crown of crystal spikes, each one twinkling like a little star. She loved to explore the garden, her crown catching the light and casting rainbows everywhere. She snuffled through the leaves, her tiny feet padding softly on the earth. Every day, Holly would visit her friend, another hedgehog who lived in the oak tree. At lunch, Holly shared her berries with the ants, who admired her colorful spikes. In the evening, Holly would sit and watch the stars, her crown shimmering along with them. When it was time for a nap, she curl up in her cozy bed.

Table 10. Sample Story Prompts 3

Fiona the flamingo stood gracefully on her legs, her feathers a fiery orange against the setting sun. She loved to watch the ripples in the water, each one telling a story. One by one, her friends flew in, splashing softly in the shallow waters. The water glimmered, turning gold and pink as the sun dipped lower. Fiona and her friend danced in the twilight, creating a whirlpool of colors with their wings. As the stars began to twinkle, she settled down, nestling together in the warm sand.