Towards Long Narrative Video Generation

Junfei Xiao[1], Feng Cheng[2], Lu Qi[2], Liangke Gui[2], Jiepeng Cen[3], Zhibei Ma[2], Alan Yuille[1], Lu Jiang[2]

[1]Johns Hopkins University [2]ByteDance Seed [3]ByteDance

Project Page: https://long-narrative-video-gen.github.io

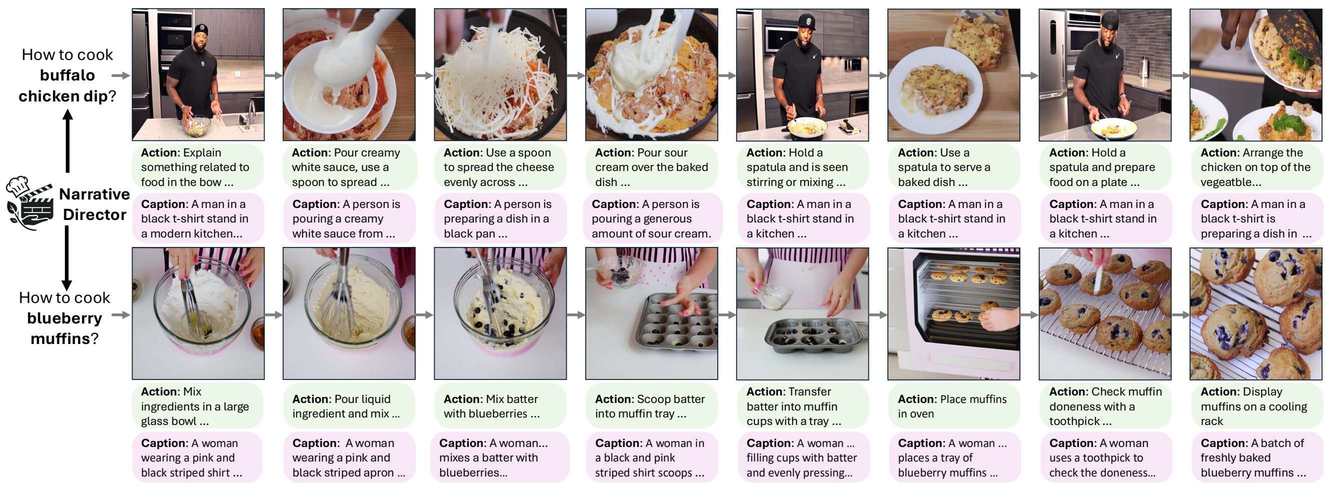


Figure 1. **Long Narrative Video Generation.** We curate a large-scale cooking video dataset to develop an interleaved auto-regressive model that acts as a narrative director, sequentially generating actions, captions, and keyframes (two generated examples here). These elements condition a video generation model to create long narrative videos. See video demos at this website.

Abstract

Recent video generation models have shown promising results in producing high-quality video clips lasting several seconds. However, these models face challenges in generating long sequences that convey clear and informative events, limiting their ability to support coherent narrations. In this paper, we present a large-scale cooking video dataset designed to advance long-form narrative generation in the cooking domain. We validate the quality of our proposed dataset in terms of visual fidelity and textual caption accuracy using state-of-the-art Vision-Language Models (VLMs) and video generation models, respectively. We further introduce a Long Narrative Video Director to enhance both visual and semantic coherence in generated videos and emphasize the role of aligning visual embeddings to achieve improved overall video quality. Our method demonstrates substantial improvements in generating visually detailed and semantically aligned keyframes, supported by finetuning techniques that integrate text and image embeddings within the video generation process. Codes and data will be made publicly available.

1. Introduction

Video generation [5, 6, 17, 18, 39, 49, 60] has recently witnessed remarkable advancements with diffusion [2, 19, 32, 55] and auto-regressive models [22, 41, 42, 51]. A primary objective is to generate coherent video clips from text prompts. In addition, a text-to-video model is able to serve as a foundation that can be adapted for various downstream applications, such as image animation [8, 52], video editing [4, 11], video stylization [20], *etc*.

With the maturity of generating high-fidelity short video clips, researchers begin setting their sights on the next north-star: creating videos capable of conveying a complete narrative which captures an account of events unfolding over time. The literature has extensively highlighted narratives. The importance of narratives has been extensively documented in the literature, emphasizing their fundamental role for mankind. For example, Bruner argues that narratives are essential tools for organizing experiences and memories [3]. Similarly, Harari indicates that the ability to share narratives—stories and abstract ideas—has been pivotal in human development, setting humans apart from other animals and leading to the creation of nations and economic systems in our society [15]. It is not unexpected that all the key challenges associated with short-video will resurface in the context of long narrative video generation (NVG), along with new, unprecedented challenges across multi-clips, including constructing semantic consistency within complex event sequences, preserving object/character identity across multiple scenes, and more.

A notble challenge researchers face is the lack of high-quality video data for learning narratives in video generation. While our community has developed many video datasets, most are unsuitable for NVG due to several reasons. First, most videos are tagged with descriptions that are partially or irrelevant to NVG. Second, even for the relevant descriptions, the quality may be insufficient, because these descriptions may be either too coarse or lack detailed actions needed for NVG. Finally, the type of video itself is a factor; not all videos contain meaningful narratives suitable for learning. Learning from videos that have meaningful, complete, and unambiguous narratives is essential not only for training but for evaluating NVG progress and benchmarking different approaches.

Partly due to the data challenge, progress in narrative video generation has been sluggish, especially when compared to story generation through a series of images, such as comics, cartoon or illustrations [14, 21, 30, 54]. Some of these images, when treated as keyframes, may be further processed into video clips [16, 28, 59, 62, 63]. A drawback of the keyframe-based approach is that the narrative video creation process is scattered across different modules and fine-tuned in multiple steps, making it difficult to optimize the overall process. Very recently, Yang *et al.* [54] proposed a story generation method using a vision-language model (VLM) to generate both images and text. Despite the promising results, there has yet to be a comprehensive study on using VLM models for narrative video generation.

This paper contributes to advancing research in narrative video generation in two ways. First, we curate and annotate a large-scale video dataset focused on the cooking domain. The samples in our dataset are structured with clear narrative flows, each composed of sequential actions and visual states. Our dataset consists of approximately 200,000 video clips, with an average duration of 9.5 seconds per clip. We chose cooking videos due to their complete and unambiguous narratives, which are more objective to annotate and evaluate consistently. To address video copyright concerns, we source videos from existing video datasets, YouCook2 [61] and HowTo100M [31]. Beyond the video pre-processing such as quality filtering and captioning, we conduct a caption-action matching mechanism to extract narrative clips, following the strict, step-by-step process inherent in cooking tasks.

Furthermore, we present a general auto-regressive pipeline for long narrative video generation, comprising two main components: a long narrative director and a visual-conditioned video generation model. The long narrative director produces a coherent narrative flow by generating a sequence of visual embeddings or keyframes that represent the story's logical progression, as illustrated in Figure 1.

Extensive experiments on our dataset demonstrate the effectiveness of the proposed pipeline for long narrative video generation. To sum up, our contributions are as follows:

- A large, structured dataset and a comprehensive data pipeline designed to benchmark long-form narrative video generation. We will open-source the data along with the necessary functionalities to support future long video generation research.
- We propose an effective data-driven pipeline for automatic long video generation. We emperically explore the design and training of an interleaved image-text auto-regressive model for generating visual states and a visual-conditioned video generation model.

2. Related Works

Text-to-Image/Video Generation Text-to-image [7, 23, 33, 34, 36, 48, 56] and video generation [5, 6, 17, 18, 39, 49, 60] have made remarkable progress to generate high-fidelity video clip of 5-10 seconds. For example, latent design [36] has become mainstream, balancing effectiveness with efficiency. Building upon this design, diffusion-based models like DiT [32], Sora [2], and CogVideo [19, 55] leveraged larger datasets and explored refined architectures and loss functions to enhance performance. In contrast, auto-regressive models such as VideoPoet [22] and Emu series [41, 42, 51] sequentially predict image or video tokens. Instead, our work focuses on the model's ability to generate long narrative videos beyond 5 seconds.

Interleaved Image-Text Modeling Interleaved image-text generation [1, 9, 13, 13, 43, 53] has garnered attention as a compelling research area that merges visual and textual modalities to produce rich outputs. Earlier approaches [26, 35, 35, 40] primarily relied on large-scale image-text paired datasets [12, 37] but were often confined to single-modality tasks, such as captioning or text-to-image generation. With the emergence of large language models (LLMs) [45], various vision-language models (VLMs) [25, 29, 50] have ushered in a new era of unified representations, leveraging well-curated datasets for interleaved generation. However, most existing works focus on the one-time generation and do not address the coherence of generated content, which is the focus of our work.

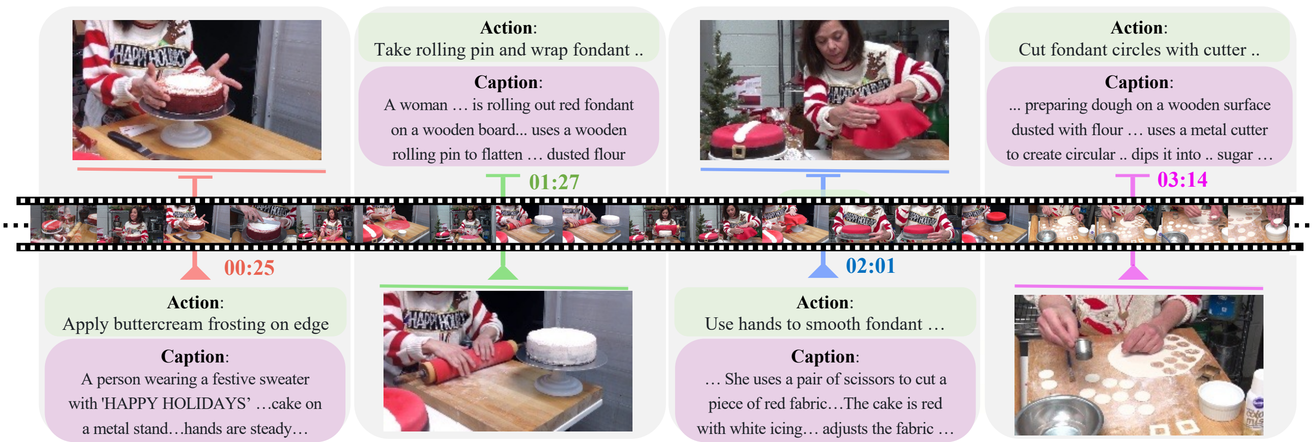


Figure 2. **Long narrative videos annotated with actions and captions.** Each source video is cut into clips and matched with the labeled "actions". We use refined pseudo labels from ASR for Howto100M videos and use manual annotations for Youcook2 videos. We use state-of-the-art VLMs (*i.e.* GPT-4o and a finetuned video captioner) to provide high-quality captions for all video clips.

Narrative Visual Generation The narrative visual generation lies in ensuring consistency across generated images. With this motivation, several works like SEED-Story [54], StoryDiffusion [62], MovieDreamer [59], LVD [27], VideoDirectorGPT [28], Vlogger [63], Animate-a-story [16], VideoTeris [44] predominantly employ conditional generation in diffusion or auto-regressive models. Our work closely relates to the very recent work [54], which generates multimodal, long-form coherent narratives based on user-provided images and texts as story beginnings. However, we aim to generate coherent stories through videos, presenting greater identity and motion preservation challenges.

3. Long Narrative Video Data

To the best of our knowledge, the dataset for long narrative video generation research is extremely limited. To enable in-depth exploration and establish an experimental setting, we establish a large video dataset with detailed annotations on captions, actions, and annotations. As the data example provided in Figure 2, our dataset focuses on cooking videos. We prioritize cooking over other video categories because each dish follows a pre-defined, strict sequence of action steps. These structured and unambiguous objectives in cooking videos are essential for learning and evaluating long video narrative generation.

3.1. Overview

We source over 30,000 raw videos about from two existing video datasets: YouCook2 [61] and HowTo100M [31]. Each video is filtered and cropped with processing to remove obvious logos or watermarks. Table 2 provides detailed information about the dataset statistics, video and clip details, and the train/validation partitioning.

Table 1 compares our dataset with existing datasets most relevant to multimodal narrative generation. Unlike existing datasets that primarily focus on image-based comic story generation, our real-world narrative dataset offers several advantages. First, the videos in our dataset depict procedural activities (*i.e.*, cooking), providing unambiguous narratives that are easier to annotate and evaluate. Second, our dataset contains 150× the number of frames compared to the previous largest dataset, StoryStream. Third, we offer 5× denser textual descriptions, with an average of 763.8 words per video. These advantages make our dataset a better resource for narrative video generation.

Datasets	Modality	Type	# Images	Text Length
Flintstones	Image	Comic	122k	86
Pororo	Image	Comic	74k	74
StorySalon	Image	Comic	160k	106
StoryStream	Image	Comic	258k	146
VIST	Image	Real world	210K	∼70
Ours	Video	Real world	39M	763.8

Table 1. **Comparison with multi-modal narrative datasets.** Most existing datasets focus on image-based comic story generation. In contrast, our dataset consists of long narrative videos, containing 150× the number of frames and 5× the dense text annotations compared to the previous largest dataset, StoryStream.

3.2. Annotation and Processing

To ensure scalability and quality, we design an efficient annotation pipeline to support the annotation as below.

Captions. For open-source and scalability, we train a video captioner based on open-sourced VLM. Inspired by LLaVA-Hound [57], we begin by collecting a caption dataset using GPT-4o, with a focus on object attributes, subject-object interactions, and temporal dynamics. Subsequently, we fine-tune a captioning model based on LLaVA-NeXT [58] to optimize captioning performance.

Data Source	# Vid. (train/val)	# Clips	Clip Len.	# Clips / Vid.
YouCook2	1333 / 457	∼10K	19.6s	7.7
HowTo100M (subset)	30039 / 933	∼183K	9.5s	5.9

Table 2. **Long narrative dataset sources.** Our dataset is built upon Youcook2 and a cooking subset of Howto100M.

Actions. We use HowTo100M ASR-based pseudo labels for 'actions' in each video, further refined by LLMs to provide enhanced annotations of the actions throughout the video [38]. This refinement improves the action quality to capture events and narrative context. However, the annotations are still noisy and sometimes not informative due to the inherent errors in ASR scripts.

Caption-Action Matching and Filtering. To ensure alignment between captions and actions, we implement a matching process based on time intervals. Using Intersection-over-Union (IoU) as a metric, we evaluate whether the overlap between the captioned clip time and action time meets a threshold. An action is considered a match if the following conditions are met: the difference between the clip start time and the action start time (`start_diff`) is less than 5 seconds; the clip end time is later than the action end time; and the IoU between the clip and action time intervals is greater than 0.25, or if IoU>0.5. Here, `clip_time` and `action_time` represent the time intervals for the clip and action, respectively. Using this rule, we filter and match captions to actions, ensuring that each caption aligns with the relevant action. We found this step is important for creating narrative consistency throughout the video.

3.3. Evaluation: Generation and Understanding

High-quality captions are essential for narrative visual generation. To assess the quality of our annotations, we evaluate them from two perspectives: inverse generation (§3.3.1) and visual understanding through VLM experts (§3.3.2).

3.3.1 Inverse Video Generation

This evaluation is motivated by the understanding that high quality captions, when combined with ground truth keyframes, more effectively reconstruct the original videos. We evaluate the dataset's ability to reconstruct original videos using the annotated captions, with and without conditioning with ground truth keyframes. For this evaluation, we assess the validation set (∼5,000 video clips) We measure reconstruction quality using FVD [47]. The results, shown in Table 3, indicate that our captions capture sufficient semantic information, enabling effective representation of the original videos. When generating with ground-truth keyframes, the video quality is very high and closely aligned with the original videos, as shown by the low FVD score (116.3). Without keyframes, the captions alone still provide reasonable alignment. Examples of reconstructed videos are included in the supplementary materials.

Validation Set	w/. GT keyframe	W/o. GT keyframe
# Clips	FVD	FVD
5504	**116.3**	561.1

Table 3. **Inverse video generation.** Evaluation of caption quality through inverse video generation with and without keyframes. FVD scores reflect reasonable video reconstruction quality.

	GPT-4o Evaluation		Human Evaluation	
Score (0-100)	Qwen2-VL-72B	**Ours**	Qwen2-VL-72B	**Ours**
	98.0	95.2	79.3	**82.0**

Table 4. **Caption Quality Evaluation**. We compare the caption quality between our captioner and the Qwen2-VL-72B model by both GPT-4o and human annotators. Our model achieves competitive results despite a much smaller model size.

3.3.2 Semantic Consistency across VLM Experts

GPT-4o & human evaluation. We evaluate the quality of our captions using both GPT-4o and six human annotators, in which we ask humans and GPT-4o to rate our dataset provided captions according to two criteria: the coverage of video elements and the absence of hallucinations in the caption. Following [57], hallucination refers to the model generating content absent or unsupported by the video, such as incorrect details about objects, actions, or counts.

To demonstrate the quality, we compare our captions with those generated by a state-of-the-art open-source VLM (Qwen2-VL-72B). As shown in Table 4, our dataset's captions receive a decent score of 95.2 out of 100, showing slightly better alignment with rigorous human evaluation than the Qwen2-VL-72B model. Results from both human evaluators and GPT-4 assessments indicate that the dataset contains high-quality captions.

4. Method

Given the text input, the task of long narrative video generation aims at generating a coherent long video $\mathcal{Y} \in \mathbb{R}^{H\times W\times F}$ that aligns with the progression of the text input. The H, W, and F are generated videos' height, width, and frame numbers. To achieve this, we propose a pipeline that involves two main components: a long narrative video director and visual-conditioned video generation. The long narrative video director is used to generate a sequence of visual embeddings or keyframes that capture the narrative flow (§4.1), while the visual-conditioned video generation generate video clips based on these visual conditions (§4.2).

4.1. Long Narrative Video Director

As shown in Figure 3, the long narrative video director generates a sequence of visual embeddings (or keyframes) that capture the narrative flow. In the following subsections, we

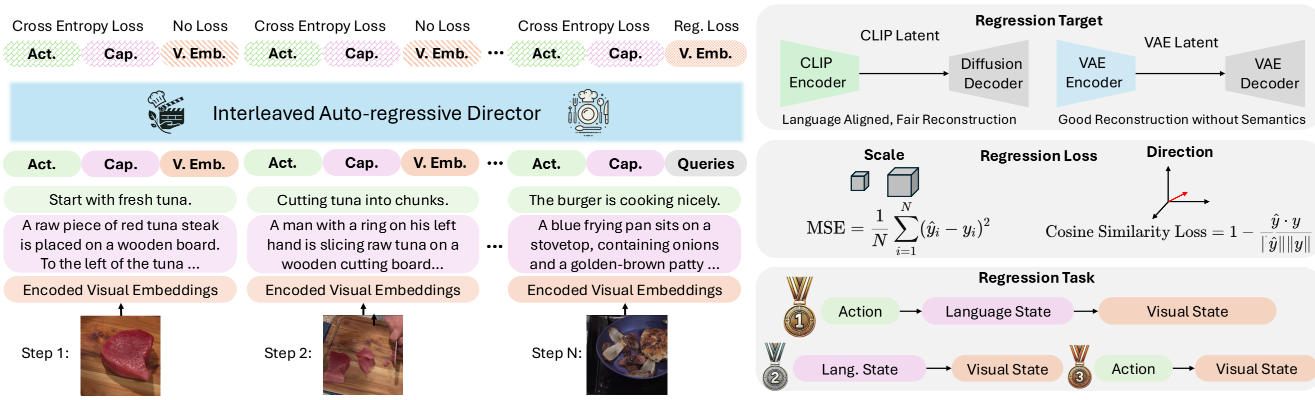


Figure 3. **Long Narrative Video Director**. The video director, a Visual Language Model (VLM), takes a user query (e.g., "How to cook a tuna sandwich?") and an initial image-text pair as input. It then generates captions, actions, and visual states step-by-step. Each video clip is created using either visual embeddings or a keyframe derived from these embeddings. This study focuses on key design choices, such as the visual latent space, regression loss for visual embeddings, and the regression task.

first conduct a analysis of two kinds of video directors based on interleaved generation (§4.1.1) and language-centric visual generation (§4.1.2).

4.1.1 Interleaved Image-Text Director

The interleaved image-text director creates a sequence where text tokens and visual embeddings are interleaved, integrating narrative and visual content tightly. Using an auto-regressive model, it predicts the next token based on the accumulated context of both text and images, maintaining narrative coherence and aligning visuals with the text.

Interleaved auto-regressive model. Our model performs next-token prediction for cross-modal generation, learning from sequences of interleaved image-text pairs with a context window size T. Each text token is supervised with cross-entropy loss, and the final visual embedding $\mathbf{z}_T$ is regressed using learnable query tokens, as illustrated in Figure 3. The auto-regressive conditioning is given by:

$$p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1}) = p(\mathbf{c}_t \mid \mathbf{c}_{1:t-1}) \cdot p(\mathbf{z}_t \mid \mathbf{c}_{1:t}, \mathbf{z}_{1:t-1}), \quad (1)$$

where $\mathbf{c}_t$ represents texts and $\mathbf{z}_t$ denotes visual embeddings.

Regression latent space. We utilize a CLIP-Diffusion visual autoencoder with a CLIP encoder E_{clip} and a diffusion decoder D_{diff} to encode raw images $\mathbf{x}$ to visual embeddings for auto-regressive generation:

$$\mathbf{z} = E_{\text{clip}}(\mathbf{x}), \quad \hat{\mathbf{x}} = D_{\text{diff}}(\mathbf{z}) \quad (2)$$

This setup generates language-aligned visual embeddings and reconstructs images from them.

Regression loss. To align the generated visual latents $\mathbf{z}_{\text{pred}}$ with the target latents $\mathbf{z}_{\text{target}}$, we use a combined loss:

$$L_{\text{reg}} = \alpha\left(1 - \frac{\mathbf{z}_{\text{pred}} \cdot \mathbf{z}_{\text{target}}}{\|\mathbf{z}_{\text{pred}}\|\|\mathbf{z}_{\text{target}}\|}\right) + \beta\frac{1}{N}\sum_{i=1}^{N}(\hat{z}_i - z_i)^2 \quad (3)$$

where α and β balance the contributions of cosine similarity and mean squared error to regress both scale and direction.

Narrative from "actions" to "visual states". The interleaved model generates a coherent narrative sequence by progressively conditioning each step on the cumulative context from previous steps, Figure 3. At each time step t, the model generates an action $\mathbf{a}_t$, a caption $\mathbf{c}_t$, and a visual state $\mathbf{z}_t$, conditioned on the cumulative history $\mathcal{H}_{t-1}$:

$$\begin{aligned} \mathcal{H}_{t-1} &= \{\mathbf{a}_{1:t-1}, \mathbf{c}_{1:t-1}, \mathbf{z}_{1:t-1}\} \\ \mathbf{a}_t \mid \mathcal{H}_{t-1} \rightarrow \mathbf{c}_t \mid \{\mathcal{H}_{t-1}, \mathbf{a}_t\} &\rightarrow \mathbf{z}_t \mid \{\mathcal{H}_{t-1}, \mathbf{a}_t, \mathbf{c}_t\} \end{aligned} \quad (4)$$

This layered conditioning ensures coherence across the sequence, aligning actions, language, and visuals.

4.1.2 Language-Centric Keyframe Director

The interleaved auto-regressive model can also act as a language-centric director when reduced to using only text-based guidance. In this case, keyframes are synthesized using a text-conditioned diffusion model with only captions. For each caption $\mathbf{c}_t$, the diffusion model D_{text} produces the visual state $\mathbf{x}_t = D_{\text{text}}(\mathbf{c}_t)$. This text-only approach benefits from straightforward integration with off-the-shelf text-conditioned diffusion models (*e.g.*, FLUX-1 [24]) and hence enjoys high-fidelity image generation. However, this approach structures the narrative and maintains visual consistency without regressed embeddings. Therefore, it lacks

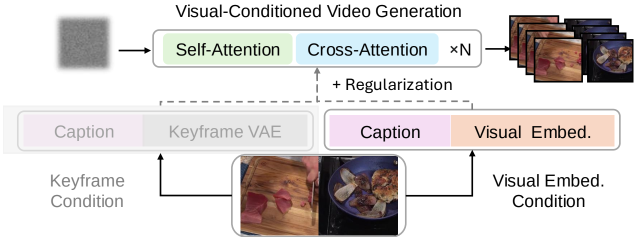


Figure 4. **Visual-Conditioned video generation.** Our interleaved auto-regressive director generates both text and visual conditions, enabling the video generation process to be conditioned either on keyframes (VAE embeddings) or on CLIP latents regressed by the interleaved director. We apply Gaussain noise, random masking and random shuffling as regularization during the training process to improve robustness with the imperfect visual embeddings.

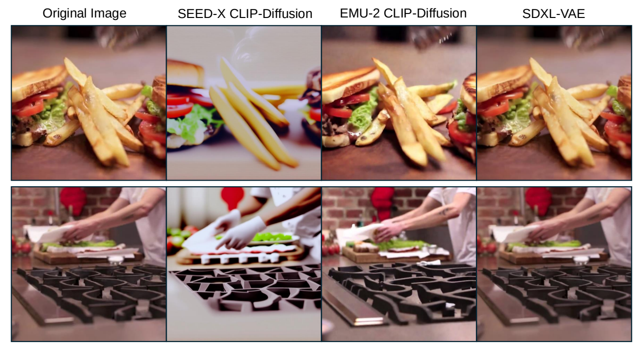


Figure 5. **Auto-encoded results with different latent spaces.** While SEED-X and EMU-2 both use a CLIP vision encoder and a diffusion model (*i.e.* finetuned SDXL) as decoder for autoencoding visual latents, SEED-X is semantic-biased and EMU-2 keeps much more visual details. SDXL-VAE shows the best image reconstruction ability, however, the latent space is not aligned with language (*i.e.* without pretraining on image-text pairs like CLIP).

nuanced transitions between keyframes compared to the interleaved model, which enhances coherence by directly incorporating visual embeddings into the generation process.

4.2. Visual-Conditioned Video Generation

Using the sequence of actions $\mathbf{a}_t$, captions $\mathbf{c}_t$, and visual states $\mathbf{z}_t$ generated by the narrative director, we condition a video generation model to produce coherent long narrative videos. Unlike the classic Image-to-Video (I2V) pipeline that uses an image as the starting frame, our approach leverages the regressed visual latents $\mathbf{z}_t$ as continuous conditions throughout the sequence (see §4.2.1). Furthermore, we improve the robustness and quality of the generated videos by adapting the model to handle noisy visual embeddings, since the regressed visual latents may not be perfect due to regression errors (see §4.2.2).

4.2.1 Visual Conditions Beyond Keyframes

Traditional visual-conditioned video generation typically uses initial keyframes to guide the model, where each frame $\mathbf{x}_t$ is generated as $\mathbf{x}_t = D_{\text{visual}}(I_t)$. Our interleaved auto-regressive director extends this by generating visual states $\mathbf{z}_t$ in a semantically aligned latent space, allowing direct conditioning without biases from a pretrained visual decoder, as shown in Figure 4. By using these regressed visual latents $\mathbf{z}_t$ directly, each frame is generated as $\mathbf{x}_t = D_{\text{visual}}(\mathbf{z}_t)$, ensuring that the video accurately follows the narrative and enhancing consistency by relying on narrative-aligned embeddings rather than potentially biased keyframes (illustrated in Figure 5).

4.2.2 Learning from Noisy Visual Conditions

To enhance the video generation model's robustness to imperfect visual embeddings $\mathbf{z}_t$ from the auto-regressive director, we fine-tune the model using noisy embeddings $\mathbf{z}'_t$ defined by:

$$\mathbf{z}'_t = \mathcal{S}(\mathcal{M}(\mathbf{z}_t + \boldsymbol{\epsilon})) \tag{5}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbf{z}_t)$ represents Gaussian noise, $\mathcal{M}$ is a masking operator that sets a fraction of elements to zero, and $\mathcal{S}$ is a shuffling operator that permutes some embedding dimension. Training with $\mathbf{z}'_t$ improves the model's ability to handle noisy visual conditions, improving generation quality and robustness with imperfect embeddings.

5. Experiments

5.1. Experimental Setup

Models. Following Seed-Story [54], we use the SEED-X [13] pretrained 7B multi-modal LLM as our base model and apply LoRA finetuning on our narrative dataset. For video generation, we employ a Sora-like model [2], which has been pretrained on large-scale video-text pairs and could accept both text and visual conditions.

Data. We use a total of ~32K narrative videos for model developing and also use ~1K videos for validation. All the videos are resized to 448 (short-side) and then center-cropped with 448x448 resolution.

Training & Evaluation. Experiments of interleaved auto-regressive director model are trained with 5,000 steps by default. Training loss is a combination of cosine similarity loss and MSE loss for visual tokens and CrossEntropy loss for language tokens. For visual-conditioned video generation, we use the diffusion loss following DiT [32] and Stable Diffusion 3 [10]. Narrative generation is mostly evaluated on the Youcook2 validation set because of the high-quality of action annotations and the Howto100M validation set is mostly used for data quality evaluation and I2V generation.

Method	Autoencoder Style	VL Aligned.	Recon. Ability	CLIP-T	FID
SDXL-VAE	Variational U-Net	✗	High	13.2	286.6
EMU-2	CLIP-Diffusion	✓	Medium	**25.4**	76.7
SEED-X	CLIP-Diffusion	✓	Low	25.1	**30.1**

Table 5. **Visual latent spaces for visual regression.** The VAE latent space is challenging for auto-regressive models to regress in a single step due to its limited correlation with language. In contrast, the language-aligned latent spaces (EMU-2 and SEED-X) allow for easier and effective regression in an interleaved manner.

5.2. Interleaved Narrative Director

In this section, we discuss three key aspects we explored to improve interleaved auto-regressive model for interleaved narrative visual generation: 1) different visual latent space for visual regression, 2) loss design for accurate latent regression, and 3) the cross-modality regression task.

CLIP beats VAE for interleaved generation. We experiment with three different auto-encoded visual latent spaces for regression: the EMU-2 [42] CLIP-Diffusion autoencoder, the SEED-X CLIP-Diffusion autoencoder, and the KL Variational autoencoder (VAE) used by SDXL. Both SEED-X and EMU-2 use a CLIP vision encoder and a fine-tuned SDXL diffusion model as the decoder for encoding visual latent. From Figure 5, we can observe that SDXL-VAE achieves the best reconstruction quality. However, in terms of visual generation quality, as shown in Table 5, the CLIP-Diffusion based autoencoders significantly outperform VAE (*i.e.*, **+12.2** CLIP-T score and **256.6** better FID). This suggests that CLIP embeddings are more suitable for interleaved visual generation compared to VAE's latent space. This is reasonable, as SDXL-VAE is not aligned with language and lacks semantic understanding.

Latent scale and direction matters. To determine an effective supervision strategy for visual embeddings, we firstly test the robustness of the latents to pseudo regression errors by rescaling (multiplying by a factor) and adding random Gaussian noise. Figure 6 indicates that both scale and direction are critical in latent regression. Notably, rescaling primarily affects object shape while preserving key semantic information (*i.e.* object type and location), whereas adding noise drastically impacts reconstruction quality. As shown in Table 6, combining MSE loss (minimizing scale error) and cosine similarity (minimizing direction error) leads to the best generation quality (*i.e.*, **+1.9** CLIP score and **3.4** better FID), which further verifies our findings.

From "Actions" to "Visual States". We also explore how different regression tasks influence the director's capability in narrative visual generation. Specifically, we compare various reasoning settings for the interleaved director, examining transitions from sequential actions to language states, and ultimately to visual embeddings. As shown in Table 7, a chain of reasoning that progresses from actions to language states and then to visual states proves effective for long narrative visual generation. This approach enhances both training convergence, achieving a lower L2 distance (**0.41** vs. **0.43**), and generation quality, reflected in a superior FID score of 25.3 (an improvement of **+0.8**).

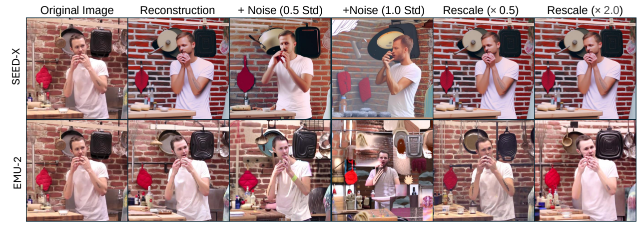


Figure 6. **Both Scale and Direction Matters.** We experiment with pseudo regression errors by altering latent direction and scale using Gaussian noise and scaling factors. The reconstruction results confirm that preserving both scale and direction is essential for accurate latent regression.

Loss Type		SEED-X Latent				EMU-2 Latent			
		Training		Validation		Training		Validation	
MSE	Cos.	L2 Dist.	Cosine.	CLIP	FID	L2 Dist.	Cosine.	CLIP	FID
✓	✗	**0.41**	0.82	23.6	31.9	**1.3**	0.78	25.1	80.1
✗	✓	1.1	0.82	24.1	32.1	2.5	0.79	23.5	115.3
✓	✓	**0.41**	**0.83**	**25.1**	**30.1**	1.4	**0.79**	**25.4**	**76.7**

Table 6. **Regression loss with scale and direction.** We track the training convergence and evaluate models with the CLIP-T and FID metrics on the validation set. Both Seed-X and EMU-2 latent space show that a combination of both MSE loss and Cosine Similarity loss considering both scale and direction performs best. SEED-X and EMU-2 original regression loss setting is grayed .

Regression Task	Training		Validation	
	L2 Dist.	Cosine Sim.	CLIP-T	FID
Action → Vis. Embed.	0.43	0.82	22.7	27.9
Caption → Vis. Embed.	0.41	0.82	25.7	26.1
Action → Caption → Vis. Embed.	**0.41**	**0.83**	**26.1**	**25.3**

Table 7. **From "Actions" to "Visual States".** We report the L2 distance and cosine similarity scores for tracking the training convergence and evaluate the generation images with CLIP score and FID score. Models are trained and evaluated on the collected Howto100M subset. SEED-X latent is used for visual regression.

Interleaved or language-centric? We compare the interleaved auto-regressive model, with a language-centric approach that generates images based solely on language states (captions) using models like SD-XL and FLUX.1-schnell. As illustrated in Figure 7, the language-centric method using text-to-image models generates visually appealing keyframes but lacks consistency across frames due to limited mutual information. In contrast, the interleaved generation approach, leveraging language-aligned visual latents, learns a realistic visual style through training. However, it occasionally produces images with repetitive or noisy visual elements, as the auto-regressive model struggles to generate accurate embeddings in a single forward.

Figure 7. **Quality Comparison Example**. Given a global system prompt—*"Step-by-step guide to cooking mapo tofu:"*—along with the action, caption, and image embeddings of the first step keyframe, our interleaved director sequentially generates "actions," "captions," and image embeddings to construct a narrative on how to cook the dish step by step. The first two rows display the directly generated keyframes (decoded from the image latents) using the EMU-2 and SEED-X latent spaces. The generated images are realistic with strong visual consistency but are less aesthetically refined than those produced by state-of-the-art text-to-image models, *i.e*. SDXL and FLUX.1-s.

Latent condition	Gen. strategy	Aesthetic	Realistic	Visual consist.	Narrative
EMU-2 Latent	Interleaved	0.7	1.2	2.9	2.2
SEED-X Latent	Interleaved	2.1	**4.3**	**4.5**	**4.4**
Text (SDXL)	Language-centric	4.0	2.9	3.3	4.0
Text (FLUX.1-s)	Language-centric	**4.8**	3.1	3.4	**4.4**

Table 8. **Human Evaluation – Interleaved vs. Language-centric.** This table compares different latent conditioning and generation strategies based on semantic alignment, aesthetic quality, visual consistency, and narrative coherence. Each aspect is scored with five tiers: 1∼5, score higher is better.

The human evaluation (Table 8) further validates this point. From Table 8, we can observe that interleaved methods achieve the highest scores in realism (**4.3** vs. **3.1**) and visual consistency (**4.5** vs. **3.4**), while language-centric methods achieve the best aesthetic score (**4.8** vs. **2.1**). We argue that visual consistency is more crucial in long narrative video generation, making interleaved methods preferable.

5.3. Visual-Conditioned Video Generation

As detailed in Section 4.2, we fine-tune the model to be directly conditioned on the visual latents generated by our interleaved director, rather than relying solely on initial keyframes. Table 9 compares the keyframe-conditioned approach with our visual embedding-conditioned strategy. Our method improves CLIP-T scores on both validation sets—from 25.9 to 26.4 on YouCook2 and from 26.6 to 27.3 on HowTo100M. Additionally, FVD scores decrease, indicating better video quality (557.7 vs. 512.6 on YouCook2, 541.1 vs. 520.7 on HowTo100M). Videos conditioned on visual embeddings demonstrate higher semantic alignment and improved generation quality. Video demos are provided on the demo page and in the supplementary materials.

Visual Condition	YouCook2		HowTo100M	
	CLIP-T	FVD	CLIP-T	FVD
Keyframe	25.9	557.7	26.6	541.1
Embedding	**26.4**	**512.6**	**27.3**	**520.7**

Table 9. **Keyframes vs. Visual Embeddings.** Evaluate CLIP-T and FVD scores for video generation conditioned on keyframes versus visual embeddings generated by our interleaved director.

6. Conclusion

In this paper, we tackle the challenges of generating long-form narrative videos and empirically evaluate its efficacy in the cooking domain. We curate and annotate a large-scale cooking video dataset, capturing clear and high-quality narratives essential for training and evaluation. Our proposed two-stage auto-regressive pipeline, which includes a long narrative director and visual-conditioned video generation, demonstrates promising improvements in semantic consistency and visual fidelity in generated long narrative videos. Through experiments on our dataset, we observe enhancements in spatial and temporal coherence across video sequences. We hope our work can facilitate further research in long narrative video generation.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 2

[2] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators, 2024. 1, 2, 6

[3] Jerome Bruner. The narrative construction of reality. *Critical inquiry*, 18(1):1–21, 1991. 2

[4] Duygu Ceylan, Chun-Hao P Huang, and Niloy J Mitra. Pix2video: Video editing using image diffusion. In *CVPR*, 2023. 1

[5] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. In *arXiv*, 2023. 1, 2

[6] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *CVPR*, 2024. 1, 2

[7] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *CVPR*, 2024. 2

[8] Xi Chen, Zhiheng Liu, Mengting Chen, Yutong Feng, Yu Liu, Yujun Shen, and Hengshuang Zhao. Livephoto: Real image animation with text-guided motion control. In *ECCV*, 2025. 1

[9] Runpei Dong, Chunrui Han, Yuang Peng, Zekun Qi, Zheng Ge, Jinrong Yang, Liang Zhao, Jianjian Sun, Hongyu Zhou, Haoran Wei, et al. Dreamllm: Synergistic multimodal comprehension and creation. In *arXiv*, 2023. 2

[10] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, pages 12606–12633. PMLR, 2024. 6

[11] Ruoyu Feng, Wenming Weng, Yanhui Wang, Yuhui Yuan, Jianmin Bao, Chong Luo, Zhibo Chen, and Baining Guo. Ccedit: Creative and controllable video editing via diffusion models. In *CVPR*, 2024. 1

[12] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. In *NeurIPS*, 2024. 2

[13] Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. Seed-x: Multimodal models with unified multi-granularity comprehension and generation. In *arXiv*, 2024. 2, 6

[14] Tanmay Gupta, Dustin Schwenk, Ali Farhadi, Derek Hoiem, and Aniruddha Kembhavi. Imagine this! scripts to compositions to videos. In *Proceedings of the European conference on computer vision (ECCV)*, pages 598–613, 2018. 2

[15] Yuval Noah Harari. *Sapiens: A brief history of humankind*. Random House, 2014. 2

[16] 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

[17] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. In *arXiv*, 2022. 1, 2

[18] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022. 1, 2

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

[20] Nisha Huang, Yuxin Zhang, and Weiming Dong. Style-a-video: Agile diffusion for arbitrary text-based video style transfer. *IEEE Signal Processing Letters*, 2024. 1

[21] Ting-Hao Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. Visual storytelling. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1233–1239, 2016. 2

[22] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. In *ICML*, 2024. 1, 2

[23] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *CVPR*, 2023. 2

[24] Black Forest Labs. Flux.1 [dev]: A 12 billion parameter rectified flow transformer for text-to-image generation, 2024. Accessed: 2024-11-13. 5

[25] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 2

[26] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *CVPR*, 2023. 2

[27] Long Lian, Baifeng Shi, Adam Yala, Trevor Darrell, and Boyi Li. LLM-grounded video diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[28] Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. In *COLM*, 2024. 2, 3

[29] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2024. 2

[30] Adyasha Maharana and Mohit Bansal. Integrating visuospatial, linguistic and commonsense structure into story visualization. *arXiv preprint arXiv:2110.10834*, 2021. 2

[31] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2630–2640, 2019. 2, 3

[32] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 1, 2, 6

[33] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[34] Lu Qi, Lehan Yang, Weidong Guo, Yu Xu, Bo Du, Varun Jampani, and Ming-Hsuan Yang. Unigs: Unified representation for image generation and segmentation. In *CVPR*, 2024. 2

[35] 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 *ICML*, 2021. 2

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2

[37] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. 2

[38] Nina Shvetsova, Anna Kukleva, Xudong Hong, Christian Rupprecht, Bernt Schiele, and Hilde Kuehne. Howtocaption: Prompting llms to transform video annotations at scale. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 4

[39] 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. In *arXiv*, 2022. 1, 2

[40] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. In *arXiv*, 2023. 2

[41] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Emu: Generative pretraining in multimodality. In *ICLR*, 2023. 1, 2

[42] Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. In *CVPR*, 2024. 1, 2, 7

[43] Changyao Tian, Xizhou Zhu, Yuwen Xiong, Weiyun Wang, Zhe Chen, Wenhai Wang, Yuntao Chen, Lewei Lu, Tong Lu, Jie Zhou, et al. Mm-interleaved: Interleaved image-text generative modeling via multi-modal feature synchronizer. In *arXiv*, 2024. 2

[44] Ye Tian, Ling Yang, Haotian Yang, Yuan Gao, Yufan Deng, Jingmin Chen, Xintao Wang, Zhaochen Yu, Xin Tao, Pengfei Wan, et al. Videotetris: Towards compositional text-to-video generation. *arXiv preprint arXiv:2406.04277*, 2024. 3

[45] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[46] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 6

[47] 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:1812.01717*, 2018. 4

[48] Chaoyang Wang, Xiangtai Li, Lu Qi, Henghui Ding, Yunhai Tong, and Ming-Hsuan Yang. Semflow: Binding semantic segmentation and image synthesis via rectified flow. In *NeurIPS*, 2024. 2

[49] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. In *arXiv*, 2023. 1, 2

[50] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. In *Advances in Neural Information Processing Systems*, 2024. 2

[51] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. In *arXiv*, 2024. 1, 2

[52] Zhongcong Xu, Jianfeng Zhang, Jun Hao Liew, Hanshu Yan, Jia-Wei Liu, Chenxu Zhang, Jiashi Feng, and Mike Zheng Shou. Magicanimate: Temporally consistent human image animation using diffusion model. In *CVPR*, 2024. 1

[53] Shuai Yang, Yuying Ge, Yang Li, Yukang Chen, Yixiao Ge, Ying Shan, and Yingcong Chen. Seed-story: Multimodal long story generation with large language model. In *arXiv*, 2024. 2

[54] Shuai Yang, Yuying Ge, Yang Li, Yukang Chen, Yixiao Ge, Ying Shan, and Yingcong Chen. Seed-story: Multimodal

long story generation with large language model. *arXiv preprint arXiv:2407.08683*, 2024. 2, 3, 6

[55] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *arXiv*, 2024. 1, 2

[56] Xuanyu Yi, Zike Wu, Qingshan Xu, Pan Zhou, Joo-Hwee Lim, and Hanwang Zhang. Diffusion time-step curriculum for one image to 3d generation. In *CVPR*, 2024. 2

[57] Ruohong Zhang, Liangke Gui, Zhiqing Sun, Yihao Feng, Keyang Xu, Yuanhan Zhang, Di Fu, Chunyuan Li, Alexander Hauptmann, Yonatan Bisk, et al. Direct preference optimization of video large multimodal models from language model reward. *arXiv preprint arXiv:2404.01258*, 2024. 3, 4, 8

[58] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 3

[59] Canyu Zhao, Mingyu Liu, Wen Wang, Jianlong Yuan, Hao Chen, Bo Zhang, and Chunhua Shen. Moviedreamer: Hierarchical generation for coherent long visual sequence. *arXiv preprint arXiv:2407.16655*, 2024. 2, 3

[60] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. In *arXiv*, 2022. 1, 2

[61] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. 2, 3

[62] Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. Storydiffusion: Consistent self-attention for long-range image and video generation. *arXiv preprint arXiv:2405.01434*, 2024. 2, 3

[63] Shaobin Zhuang, Kunchang Li, Xinyuan Chen, Yaohui Wang, Ziwei Liu, Yu Qiao, and Yali Wang. Vlogger: Make your dream a vlog. In *CVPR*, 2024. 2, 3

Supplementary Material

Project page: https://long-narrative-video-gen.github.io

This appendix provides comprehensive supplementary materials to support our study. Below are brief descriptions of all the sections covered in the appendix. Please visit our project page for more visualization.

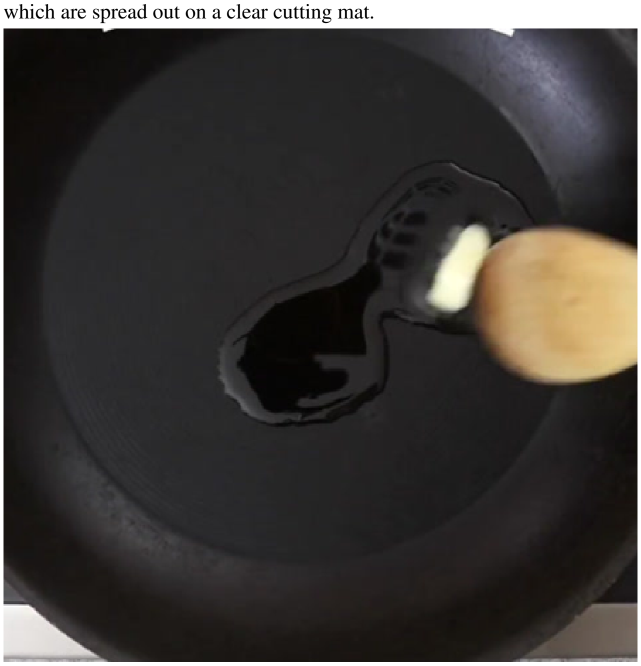
- Appendix A: **Data Examples with Annotations**
  - Presents data examples from our CookGen dataset.
  - Showcases annotated "actions" and "captions" that provide detailed multimodal information of cooking processes.
- Appendix B: **Additional Data Statistics**
  - Offers distributions of video lengths, clip lengths, and textual annotations.
  - Demonstrates the dataset's richness and suitability for long narrative video generation.
- Appendix C: **Data Evaluation Details**
  - Details our data evaluation process.
  - Includes inverse video generation results, the prompts used for video captioning, GPT-4o evaluations, and human evaluation results.
- Appendix D: **Implementation Details**
  - Outlines the implementation details of our models.
  - Provides key hyperparameters and training & inference configurations.
- Appendix E: **Action-Caption Matching Pseudo Code**
  - Includes the pseudo code for our action-caption matching algorithm.
  - Essential for aligning video clips with their corresponding annotations.
- Appendix F: **Generated Video Examples**
  - Showcases generated video examples.
  - Illustrates the effectiveness of our pipeline in producing long narrative videos for cooking recipes like "Fried Chicken" and "Shish Kabob."
- Appendix G: **Limitations**
  - Discusses the limitations of our approach.
  - Includes issues with noisy "actions" from automatic speech recognition and potential failure cases in video generation.

A. Data Examples with Annotations

Figures 8 and 9 shows two data examples from our CookGen dataset, annotated with high-quality descriptions that provide detailed multi-modal information of cooking processes. The examples clearly show structured annotations of key actions and corresponding visual descriptions, making the dataset ideal for generating long narrative videos.

(a) **Action:** Elise works with chicken thighs, advises to trim excess skin and fat
Caption: A person is preparing chicken on a wooden cutting board. He uses a pair of black-handled scissors to cut through the chicken pieces, which are spread out on a clear cutting mat.

(b) **Action:** She offers alternatives with chicken breast bone-in skin-on or chicken drumsticks
Caption: A person with light skin is preparing raw chicken pieces on a wooden surface. He places several pieces of chicken on a white cutting board.



(c) **Action:** Elise heats up a large skillet with two teaspoons of olive oil and a teaspoon of butter
Caption: A person is seen in a kitchen setting, holding a wooden spoon. He places a small piece of butter into a black frying pan on a gas stove.

(d) **Action:** Turn over the chicken pieces and cook for another 4 minutes Remove the chicken from the pan but keep the browned pieces in the pan
Caption: Golden-brown chicken pieces are sizzling in a black frying pan on a gas stove.

(e) **Action:** Use the remaining oil in the pan to brown the orzo Cook the orzo like a traditional rice pilaf, using the same method as before
Caption: A person is cooking rice in a black frying pan on a gas stove. He pours the rice from a glass bowl into the pan, then uses a wooden spatula to spread and stir the rice.

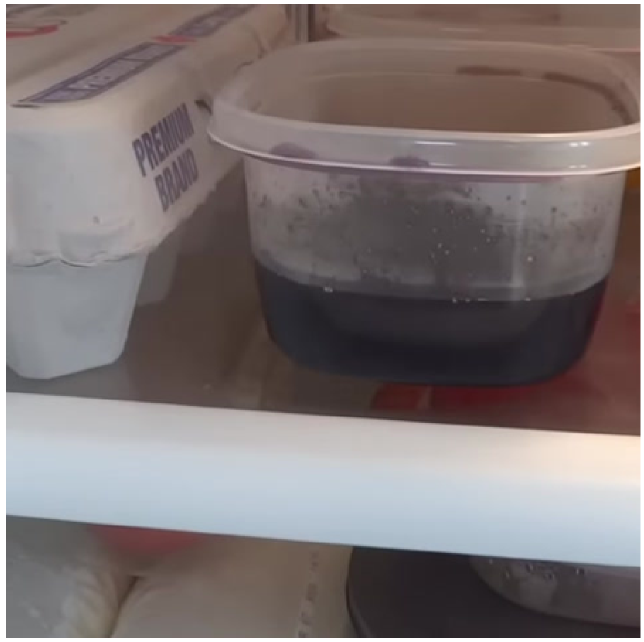
(f) **Action:** Add 2 cups of gordo's to a hot pan
Caption: A person wearing a blue shirt is cooking rice in a black frying pan on a stovetop. Using a wooden spatula, he stirs the rice, ensuring it is evenly cooked.

(g) **Action:** Combine the mixture with the orzo and cook for a few minutes until the sauce thickens
Caption: A woman is cooking on a stovetop, adding pieces of breaded chicken to a pan filled with chopped onions and rice.

(h) **Action:** Stock is cooked until orzo has fully absorbed liquid and chicken is cooked through, about 10-12 minutes Dish is removed from heat and left to sit for five minutes Dish is sprinkled with unspecified seasoning
Caption: A delicious dish of roasted chicken pieces is presented in a black skillet, surrounded by a colorful mix of diced vegetables and grains.

Figure 8. **Data examples with annotated "actions" and "captions"**. A video of cooking recipe of *"One Pot Chicken and Orzo"*.

(a) **Action:** Hi everyone, this one's called rainbow broken glass jello
Caption: A colorful, multi-layered dessert is displayed on a black surface. The dessert features vibrant red, green, blue, and purple segments, arranged in a geometric pattern.

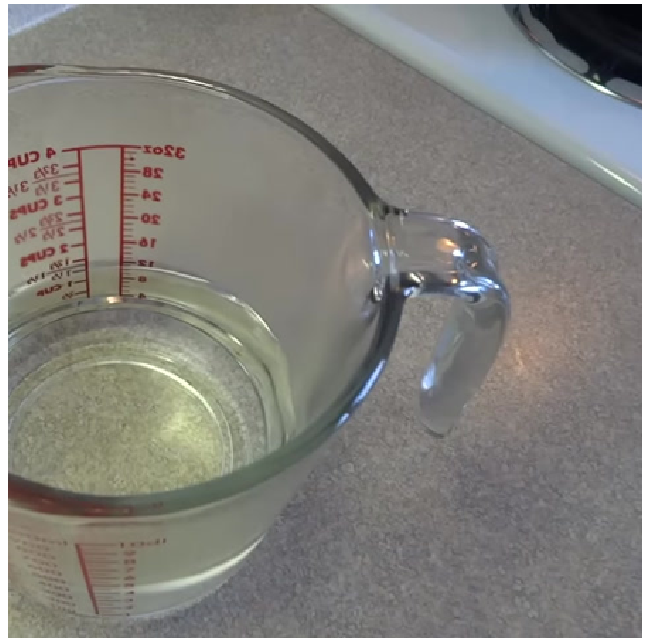


(b) **Action:** Now normally when you make jello you use two cups of boiling water, but in this case we're only using one cup because we want the jello to be extra firm
Caption: The video shows the interior of a refrigerator, focusing on the door shelf. The containers are filled with dark, blue, orange, and red liquids.

(c) **Action:** I find the easiest way to do this is to put the small container into a larger container of hot water
Caption: A person with light skin is holding a clear plastic container filled with a yellow liquid, inspecting its contents.

(d) **Action:** Loosen the edges of the Jello piece Slide the Jello piece out and cut it into cubes Cut the Jello cubes into half-inch pieces
Caption: A person is slicing a block of yellow gelatin on a wooden cutting board, cutting it into uniform strips.

(e) **Action:** Spread out the different colored Jello pieces in a 9 by 13 inch baking dish
Caption: A person is arranging colorful gelatin cubes in a glass baking dish, adjusting the placement of green, orange, purple, and black cubes.



(f) **Action:** Make a separate gelatin mixture by boiling two cups of water and adding two envelopes of gelatin
Caption: A clear glass measuring cup is placed on a countertop, containing water. A person pours a white powder into it.

(g) **Action:** Stir the sweetened condensed milk into the gelatin and water mixture
Caption: A person is vigorously whisking a creamy mixture in a clear glass measuring cup.

(h) **Action:** Let it set for several hours, then cut it into squares and serve
Caption: A glass baking dish is filled with a creamy white liquid, topped with colorful, triangular-shaped glass pieces.

Figure 9. **Data examples with annotated "actions" and "captions"**. A video of preparing *"Rainbow Broken Glass Jello"*.

B. Additional Data Statistics

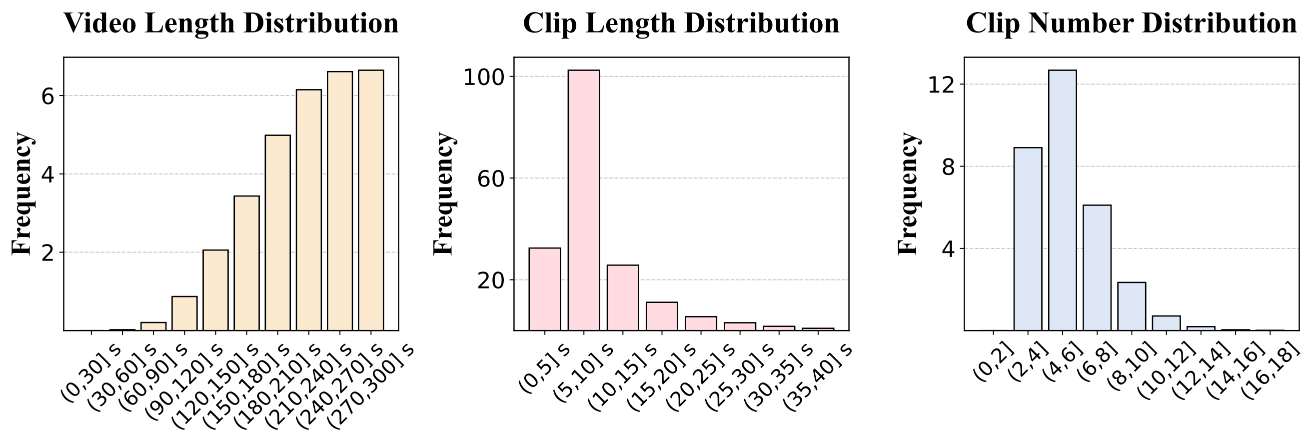


Figure 10. **Statistics on the video data.** We do statistics on the video lengths of the collected whole videos, the clip lengths of the scene-cut video clips, and the number of clips selected for each video.

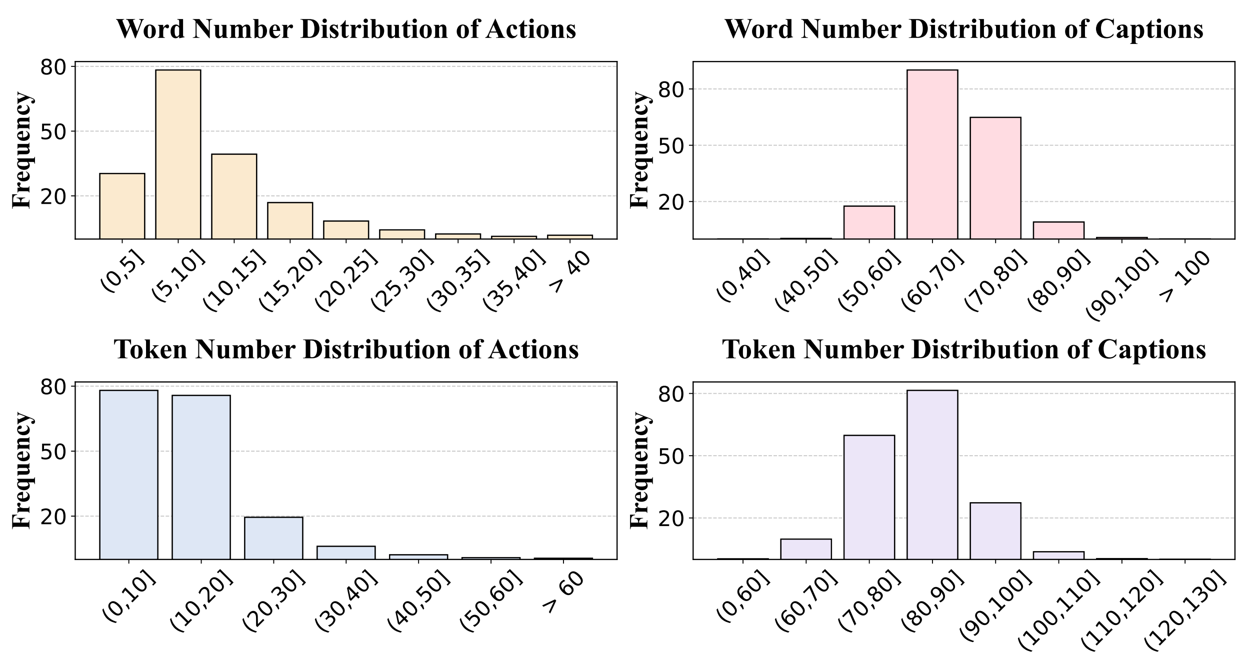


Figure 11. **Statistics on the text annotations.** We do statistics on the number of words and tokens (Llama [46] tokenized) of annotated "actions" and "captions," respectively.

The statistics in Figure 10 and Figure 11 demonstrate the high quality and suitability of our dataset for long narrative video generation. Figure 10 reveals that the video lengths range broadly, with most videos falling between **30** and **150** seconds. Clip lengths are primarily distributed between **5** and **30** seconds, ensuring manageable segments for modeling. Additionally, the majority of videos contain **4** to **12** clips, providing a balanced structure for narrative flow. Figure 11 shows that the word counts for "actions" predominantly range from **10** to **25**, while "captions" range from **40** to **70**. Token distributions further highlight their richness, with "actions" having **20** to **60** tokens and "captions" extending up to **120** tokens. These detailed annotations ensure well-aligned and contextually rich representations of the video content.

Overall, the dataset's design ensures coherent sequences of actions and captions with reasonable clip and video lengths, making it well-suited for generating high-quality, long-form narrative videos.

C. Data Evaluation Details

C.1. Inverse Video Generation Results

As discussed in Section 3.3.1, high-quality captions, especially with ground truth keyframes, enable effective video reconstruction. We compare ground truth video frames with inversely generated frames using the GT first keyframe and annotated captions, as shown in Figures 12 to 14. The reconstruction aligns well with the narrative, accurately capturing actions, though patterns and interactions differ slightly from the original video. This shows that while the captions convey crucial information for reconstruction, they lack finer visual details, a limitation for current vision-language models and human annotators.

For example, in Figure 12, the ground truth shows a hand pouring creamy liquid into a slow cooker and stirring, while the generated frames replicate the actions with slight differences in texture and liquid mixing. Similarly, in Figure 14, the ground truth shows a face drawn with cream on orange liquid, but the generated frames vary in precision and interaction details. These examples highlight the captions' strength in preserving narrative flow while exposing gaps in capturing fine-grained visual detail.

Figure 12. **Left:** Ground truth, **Right:** Inverse generation with GT keyframe. **Caption:** Chunks of meat are simmering in a dark-colored slow cooker. A hand pours a creamy liquid into the pot, causing the liquid to mix with the meat and broth. The mixture bubbles and thickens as the liquid is added. The person stirs the contents with a black spoon, ensuring the ingredients are well combined. The slow cooker continues to cook the meat, which appears tender and well-cooked.

Figure 13. **Left:** Ground truth, **Right:** Inverse generation with GT keyframe. **Caption:** A person wearing a black sleeve is whisking a creamy mixture in a clear glass bowl. The mixture appears to be a batter or dough, gradually becoming smoother and more uniform. The person's left hand holds the bowl steady on a light-colored countertop. The whisking motion is consistent and thorough, ensuring the mixture is well-blended. The background is plain, focusing attention on the mixing process.

Figure 14. **Left:** Ground truth, **Right:** Inverse generation with GT keyframe. **Caption:** A red bowl filled with a thick, orange liquid is placed on a stovetop. A woman's hand, holding a white spoon, appears and begins to draw on the surface of the liquid. She creates a face with white cream, adding details to the eyes and mouth. The background shows a granite countertop with a bunch of red tomatoes and a white pot. The woman continues to add finishing touches to the face.

C.2. Prompt for Video Captioning

Below is the prompt we designed to effectively caption video clips and also for benchmarking VLMs, ensuring detailed and accurate descriptions while avoiding redundancy:

```
You are an expert in describing videos and catching the sequential motions from video frames.

For the given ten video frames, you need to generate a detailed good description within five
sentences / 80 words. Please do not include the word 'frame' or 'frames' in your answer. If
the gender of a person is clear, use 'he' or 'she' instead of they. Do not describe a single
motion/action twice like 'xxx continues doing yyy'. Don't assume actions like discussion or
having a conversation unless it is very clear in the frames. Describe the video given the
frame sequence. Describe both the appearance of people (gender, clothes, etc), objects,
background in the video, and the actions they take.
```

Listing 1. **Video Captioning Prompt**

C.3. GPT-4o Evaluation on Captions

Below is the evaluation prompt designed to objectively assess the quality of video captions generated by a Large Multimodal Model (LMM), focusing on coverage and hallucination.

```
Your role is to serve as an impartial and objective evaluator of a video caption provided by
a Large Multimodal Model (LMM). Based on the input frames of a video, assess primarily on
two criteria: the coverage of video elements in the caption and the absence of hallucinations
in the response. In this context, 'hallucination' refers to the model generating content not
present or implied in the video, mainly focused on incorrect details about objects, actions,
counts, temporal order, or other aspects not evidenced in the video frames.

To evaluate the LMM's response:
    Start with a brief explanation of your evaluation process.
    Then, assign a rating from the following scale:
        Rating 6: Very informative with good coverage, no hallucination
        Rating 5: Very informative, no hallucination
        Rating 4: Somewhat informative with some missing details, no hallucination
        Rating 3: Not informative, no hallucination
        Rating 2: Very informative, with hallucination
        Rating 1: Somewhat informative, with hallucination
        Rating 0: Not informative, with hallucination
Do not provide any other output symbols, text, or explanation for the score.
```

Listing 2. **GPT-4o Evaluation Prompt**

C.4. Human Evaluation on Captions

Matching Tier	Action (Important Info.)	Object (Important Info.)	Score
Very Match	Good Coverage, No Hallucination	Good Coverage, No Hallucination	**100**
Good Match	Good Coverage, Limited Hallucination	Good Coverage, Limited Hallucination	**85**
Somehow Match	Fair Coverage, Some Hallucination	Fair Coverage, Some Hallucination	**70**
Not Match	Little Coverage or High Hallucination	Little Coverage or High Hallucination	**0**

Table 10. **Human Evaluation Matching Rules.** Captions are rated based on coverage and hallucination levels, using four matching tiers.

We assess the quality of our captions through evaluations by six human annotators, who rate the captions based on two key criteria: the coverage of video elements (such as objects and actions) and the absence of hallucinations, defined as generating content unsupported or absent in the video [57]. As shown in Table 4, our captions achieve a high human evaluation score of **82.0**, surpassing the state-of-the-art open-source VLM (Qwen2-VL-72B) score of **79.3**. These results demonstrate the superior quality of our captions, which are more aligned with human preferences and exhibit better narrative accuracy.

For evaluation, annotators rate the captions across four tiers—Very Match, Good Match, Somehow Match, and Not Match—based on consistency with video content. The scoring rubric, detailed in Table 10, considers both coverage and hallucination levels. Our captioner consistently achieves high scores in the top tiers, validating its reliability and quality for narrative video generation.

D. Implementation Details

We provide the training and inference hyperparameters for the interleaved auto-regressive model and the visual-conditioned video generation model in Table 11 and Table 12, respectively. The interleaved auto-regressive model is trained on images with a resolution of 448×448, using a batch size of 512 and bfloat16 precision. It employs AdamW as the optimizer, with a peak learning rate of 2×10^{-4} and a cosine decay schedule, training for 2,500 steps. Training context pairs vary between 2 and 8, while inference always uses 8 pairs for consistency. The visual-conditioned video generation model processes video data at a resolution of $448 \times 448 \times T$, with a batch size of 64 and bfloat16 precision. It uses AdamW with a peak learning rate of 1×10^{-5} and a constant decay schedule, training for 20,000 steps to handle temporal conditioning effectively.

Configuration	Setting
Image resolution	448×448
Optimizer	AdamW
Optimizer hyperparameters	$\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 10^{-6}$
Peak learning rate	2×10^{-4}
Learning rate schedule	Linear warm-up, cosine decay
Gradient clip	1.0
Total training steps	$2,500$
Warm-up steps	200
Batch size	512
Numerical precision	bfloat16
Training context pairs	[2, 8]
Inference context pairs	8

Table 11. **Implementation details of the interleaved auto-regressive model.**

Configuration	Setting
Image/Video resolution	$448 \times 448 \times T$
Optimizer	AdamW
Optimizer hyperparameters	$\beta_1 = 0.9, \beta_2 = 0.95, \epsilon = 10^{-8}$
Peak learning rate	1×10^{-5}
Learning rate schedule	Linear warm-up, constant
Gradient clip	1.0
Total training steps	$20,000$
Warm-up steps	$1,000$
Batch size	64
Numerical precision	bfloat16

Table 12. **Implementation details of the visual-conditioned video generation model.**

E. Action-Caption Matching Pseudo Code

The action-caption matching algorithm detailed in Algorithm 1 aligns video clips with actions based on temporal overlap and specific rules. It uses the Intersection over Union (IoU) to measure the overlap between the time intervals of video clips and actions. A match is identified if either the IoU exceeds 0.5 or all of the following conditions are met: the start time difference (`start_diff`) is less than 5 seconds, the clip's end time exceeds the action's end time, and the IoU is greater than 0.2.

The algorithm processes each video iteratively. For each video, it retrieves all associated actions and their time intervals. Then, for each clip in the video, it calculates the IoU with every action and evaluates the matching conditions. Valid matches, along with their metadata such as clip IDs and descriptions, are stored in a list $\mathcal{M}$. This systematic approach ensures that the matched actions and captions are temporally and semantically consistent, providing high-quality annotations for keyframe visual states.

Algorithm 1 **Pseudo code for action-caption matching.**

```
function IoU([s1, e1], [s2, e2])
    intersection ← max(0, min(e1, e2) − max(s1, s2))
    union ← max(e1, e2) − min(s1, s2)
    if union > 0 then
        return intersection / union
    else
        return 0
    end if
end function
Initialize an empty list M ← []
for all v ∈ V do
    v_id ← v.id
    if v_id ∈ A then
        A_v ← A[v_id]
        action_times ← A_v.times
        action_descriptions ← A_v.descriptions
        for all c ∈ v.clips do
            [s_c, e_c] ← c.start_end
            for all a ∈ action_times do
                [s_a, e_a] ← a
                start_diff ← |s_c − s_a|
                iou ← IoU([s_c, e_c], [s_a, e_a])
                if (start_diff < 5 ∧ e_c > e_a ∧ iou > 0.2) ∨ iou > 0.5 then
                    Create match: M ← M ∪ m
                end if
            end for
        end for
    end if
end for
return M
```

F. Generated Video Examples

Figures 15 and 16 present two examples of long narrative video generation for cooking "Fried Chicken" and "Shish Kabob," illustrated step-by-step. The generation process begins with our interleaved auto-regressive director, which generates keyframe visual embeddings and their corresponding captions. These embeddings and captions are then used as conditions for the video generation model, which produces high-quality video clips that effectively narrate the cooking process and emphasize the crucial "action" information. The resulting video clips demonstrate excellent performance in capturing the step-by-step cooking instructions. All video clips are also included in the supplementary materials for further review.

(a) **Action:** Add raw chicken pieces and seasoning to a bowl of flour.

(b) **Action:** Mix yogurt or buttermilk with seasoning in a bowl.

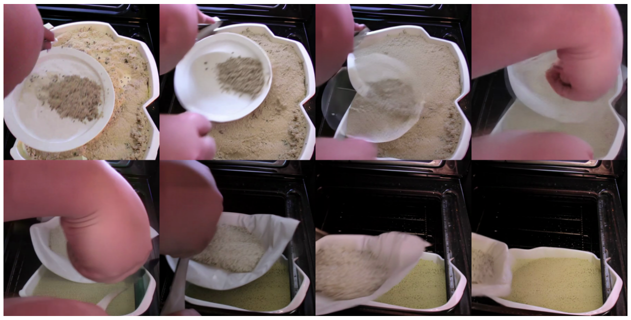
(c) **Action:** Dip chicken pieces into the batter to coat evenly.

(d) **Action:** Coat the battered chicken in the flour mixture.

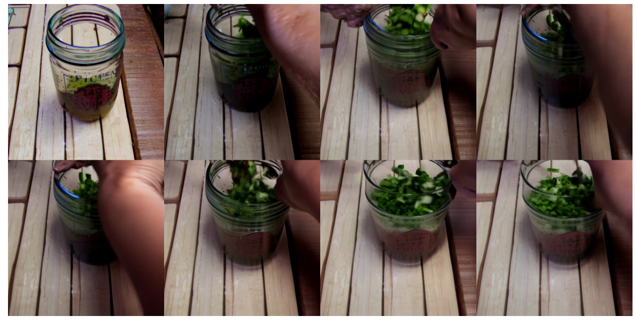
(e) **Action:** Fry the coated chicken in hot oil until crispy and golden.

(f) **Action:** Sprinkle seasoning on the fried chicken and serve.

Figure 15. **Video generation example.** Our pipeline effectively accomplishes long narrative video generation by producing six essential steps (*i.e.*, video clips) for cooking "Fried Chicken." It delivers a clear, structured, and instructional step-by-step narrative, showcasing the model's capability to generate coherent and comprehensive videos.



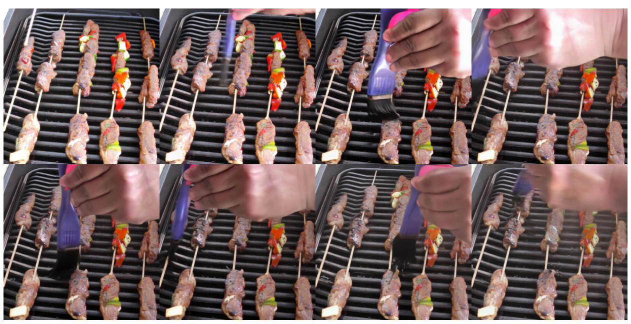
(a) **Action:** Mix chopped vegetables in a glass bowl.



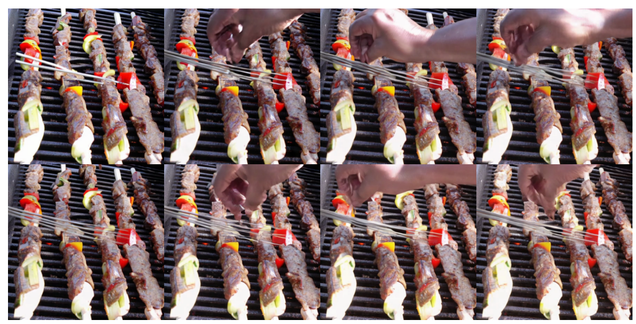
(b) **Action:** Add seasoning to the mixture of chopped vegetables.

(c) **Action:** Thoroughly mix the seasoned vegetable mixture.

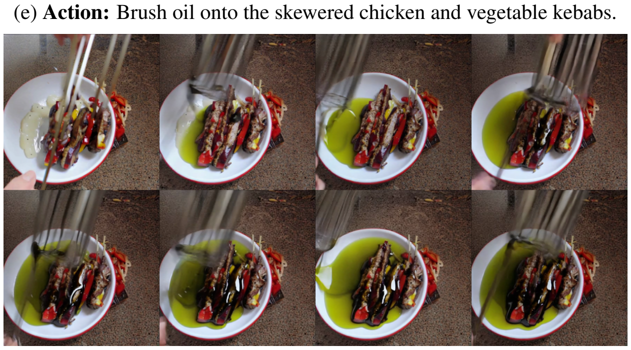
(d) **Action:** Add chicken pieces to vegetable and chicken mixture.



(e) **Action:** Brush oil onto the skewered chicken and vegetable kebabs.



(f) **Action:** Place the prepared chicken and vegetable kebabs onto a grill.



(g) **Action:** Drizzle olive oil over the chicken and vegetable kebabs.

(h) **Action:** Check on the grilling skewered chicken and vegetable kebabs.

Figure 16. **Video generation example.** Our pipeline successfully generates eight crucial steps (*i.e.*, video clips) to prepare the dish "Shish Kabob." This showcases a clear, structured, and instructional step-by-step narrative, demonstrating the model's capability to produce coherent and comprehensive video content.

G. Limitations

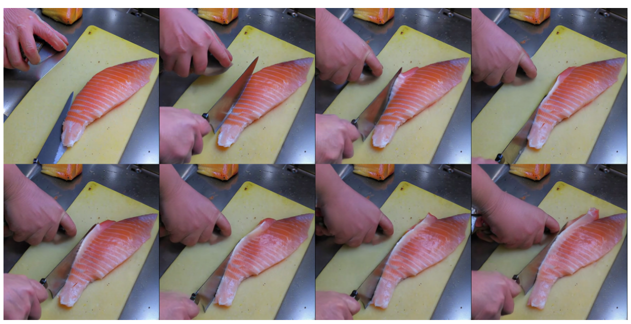
G.1. Noisy “Actions" from ASR

While our CookGen dataset provides high-quality visual and contextual annotations, the action annotations derived from automatic speech recognition (ASR) have notable limitations. ASR-generated text often contains noise, resulting in action descriptions that are incomplete, ambiguous, or not directly informative for capturing the crucial steps in cooking processes. For instance, in Figure 9(a), the action annotation *“Hi everyone, this one’s called rainbow broken glass jello”* offers little value for understanding the cooking process, while another annotation in Figure 9(b) *“Now normally when you make jello you use two cups of boiling water”* provides vague guidance without specific details about the method. Such noisy annotations fail to align with the detailed and instructive nature of cooking instructions, which require precision and clarity to guide long narrative video generation effectively. This limitation underscores the importance of refining action annotations to improve their informativeness and utility for modeling cooking tasks.

G.2. Failure Cases

While our method generates high-quality long narrative videos, there are instances where the model fails to produce meaningful cooking steps, and the rendered video clips contain unrealistic or irrelevant content due to hallucination.

Auto-regressive Director: Repeated “Steps”. Figure 17 illustrates a failure case where the auto-regressive director repeatedly generates similar visual embeddings, resulting in redundant and uninformative cooking video clips. For example, in the provided frames, the generated steps involve repeatedly cutting the salmon fillet, which adds little value to the narrative and fails to progress meaningfully. This issue is a known limitation of auto-regressive models, often caused by a lack of diversity in the embedding generation process. A potential solution is to introduce penalties for repeated embeddings or add constraints to encourage greater variability in visual outputs.



(a) **Action:** Cutting away the salmon fillet from the backbone

(b) **Action:** Slicing the salmon fillet into even pieces

Figure 17. **Failure Case.** Auto-regressive model could generate repeated “Steps”, which is not informative to viewer.

Video Generation Model: Unrealistic Hallucination.Unrealistic hallucination occurs when a video generation model produces content inconsistent with the intended narrative. In Figure 18(a), the action ”placing the fried chicken into an oven set to preheat” is misrepresented as frying chicken in a pan, with an unrealistic increase in the quantity of chicken, showing a lack of object continuity. In Figure 18(b), the action ”adding a drizzle of sauce to a plate of grilled skewers” introduces an illogical appearance of new grilled food items, deviating from the intended action and disrupting narrative coherence.

(a) **Action:** Placing the fried chicken into a oven set to preheat

(b) **Action:** Adding a drizzle of sauce to a plate of grilled skewers

Figure 18. **Failure Case.** Video generation model could make unrealisc hallucination to generate things from “air”.