LongDiff: Training-Free Long Video Generation in One Go

Zhuoling Li[1] Hossein Rahmani[1] Qiuhong Ke[2] Jun Liu[1*]
[1] Lancaster University [2] Monash University
{z.li81, h.rahmani}@lancaster.ac.uk, Qiuhong.Ke@monash.edu, j.liu81@lancaster.ac.uk

Abstract

Video diffusion models have recently achieved remarkable results in video generation. Despite their encouraging performance, most of these models are mainly designed and trained for short video generation, leading to challenges in maintaining temporal consistency and visual details in long video generation. In this paper, we propose LongDiff, a novel training-free method consisting of carefully designed components – Position Mapping (PM) and Informative Frame Selection (IFS) – to tackle two key challenges that hinder short-to-long video generation generalization: temporal position ambiguity and information dilution. Our LongDiff unlocks the potential of off-the-shelf video diffusion models to achieve high-quality long video generation in one go. Extensive experiments demonstrate the efficacy of our method.

1. Introduction

With the popularity of generative AI, text-to-video generation, which aims to create video content aligned with provided textual prompts, has become a very important and hot research topic [13]. In recent years, extensive research has been conducted in this area, where video diffusion models [3, 16, 21, 24, 28, 50] have made remarkable progress. Benefiting from training on a large collection of annotated video data, these models can generate high-quality videos that are even nearly indistinguishable from reality. However, despite the impressive generation quality, most existing approaches [6, 21, 23, 24, 50] are mainly designed and trained for short video generation, typically limited to fewer than 24 frames. Yet, in many real-world applications, like filmmaking [44], game development [9] and animation creation [49], long videos are commonly required. Thus, long video generation, with its valuable applications, has become an urgent problem to address, attracting significant attention.

To enable long video generation, a straightforward approach is to re-train video models on long-video datasets. For example, the works of [2, 41, 43] directly train models to generate entire long videos by increasing computational resources and extending model capacities. Additionally, some methods adopt auto-regressive [4, 22] or hierarchical paradigms [15, 55], which break down the complex task of long video generation into manageable processes by focusing on generating individual frames or short clips that are logically assembled to form long videos [29]. However, these training-based methods tend to be resource and time intensive due to the complexity of their models. Moreover, the scarcity of long video datasets makes it hard to meet training needs, thus making it challenging for researchers to obtain optimal parameters for long video generation [29].

Considering the amazing performance of the off-the-shelf short video generation models, another approach is to directly adapt these models for long video generation without any parameter updates. Such a training-free approach requires neither annotated long video data nor computational resources for re-training, and thus has gained much attention, with several attempts made [32, 35, 47]. To extend videos smoothly, some of them [35, 47] split the entire long video to be generated, into several overlapping short clips via a sliding window, and then process the clips asynchronously. While these methods can maintain consistency within individual video clips, the sliding window scheme limits long-range interactions between longer-distance frames, potentially leading to a lack of global temporal consistency. Another method [32] generates the long video in one go (i.e., without using sliding-windows), and improves video quality by mixing spatial and temporal information in the frequency domain. These training-free methods that mainly rely on empirical designs, have made some progress, yet the improvements achieved still remain suboptimal. In this work, we also seek to unlock the inherent potential of the off-the-shelf short video models, to generate high-quality long videos in one go in a training-free manner.

To generate videos with temporal coherence, it is crucial to capture the temporal correlations over video frames. To this end, temporal convolution and temporal transformer are incorporated in previous works [6, 48, 50] to capture

*Corresponding author

positional relationships over frames. Among them, temporal transformer with relative positional encoding techniques [40, 42] becomes more and more popular, and has been widely used in recent video diffusion models [6, 21, 23, 24, 50, 54]. In this paper, we mainly focus on this category of video diffusion models. However, despite the strong sequence modeling capability of temporal transformer, as shown in Fig. 1, directly using short video models for long video generation yields low-quality results with inferior temporal consistency (e.g., abrupt transitions between frames) and a lack of visual details (e.g., blurred textures and missing critical details).

Drawing on [18] and leveraging theoretical insights from pseudo-dimensions and information entropy, these limitations in long video generation can be further attributed to two fundamental challenges: **(1) Temporal Position Ambiguity**. The pseudo-dimensions-based analysis reveals that as the length of the generated video increases, video models struggle to accurately differentiate between the relative positions of frames. This positional ambiguity disrupts the model's ability to maintain frame order, leading to compromised temporal consistency (e.g., abrupt transitions of the polar bear's expression as illustrated in Fig. 1). **(2) Information Dilution**. The information entropy-based analysis shows that as the length of the video sequence increases, the temporal correlation entropy between frames shows an upward trend. This indicates a reduction in per-frame information content during generation, resulting in missing visual details and reduced visual quality in long video outputs (e.g., the missing "*drum*" and "*wooden bowl*", and blurred "*NYC Times Square*" as illustrated in Fig. 1).

Based on the above analysis, these two challenges can be effectively mitigated through targeted, minor modifications to the temporal transformer layers in video diffusion models. We propose **LongDiff**, a simple yet effective method that unlocks the inherent long-video generation capability of pretrained short-video diffusion models in a training-free manner, enabling the generation of high-quality long videos with global temporal consistency and visual details. Our LongDiff consists of two key components: **P**osition **M**apping (PM) and **I**nformative **F**rame **S**election (IFS). PM is designed to ensure that frame positions are accurately differentiated by video models. It maps large numbers of distinct relative frame positions to a manageable range, preserving their distinctiveness through simple `GROUP` and `SHIFT` operations. This mapping addresses the issue of temporal position ambiguity and ensures that the video model maintains the correct frame order of the generated sequence, thereby enhancing temporal consistency. To address the issue of information dilution, the IFS strategy limits temporal correlation of each frame to its neighbor frames and a set of selected key frames. This reduces the risk of excessive entropy of temporal correlation during generation, which can impair the capture of fine details. Both PM and IFS are elegant, with minor modifications to the original temporal transformer layers, which facilitate long video generation in one go.

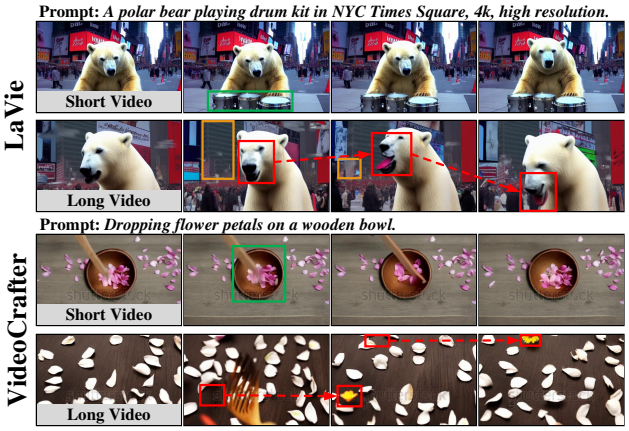

Figure 1. Results of directly applying short-video diffusion models (LaVie [50] and VideoCrafter [6]) for long video generation. Since the spatial transformer layers in these short video models operate independently of video lengths, and the temporal transformer layers can process input sequences of various length, we only need to extend the length of the noise sequences used as the starting point for denoising to achieve long video generation. It can be observed that though short videos have good quality, long videos exhibit inferior temporal consistency (marked by the red boxes), such as abrupt transitions of the polar bear's expression and the sudden appearance and disappearance of the yellow flower. Additionally, the long videos lack some key visual details (marked by the orange boxes), such as the missing "*drum*" and "*wooden bowl*", and blurred "*NYC Times Square*".

Our contributions are as follows. We propose a novel training-free method, LongDiff, consisting of two carefully designed components: position mapping and informative frame selection, to mitigate two key challenges that hinder high-quality long video generation. Our LongDiff enables pretrained short video models to generate high-quality long videos in one go and achieves state-of-the-art performance on the evaluated benchmarks.

2. Related Work

Text-to-Video Diffusion Models are mainly constructed based on text-to-image diffusion models [38, 39], incorporating temporal modules to establish correlations over video frames. Since frame order (i.e., positional relationships over video frames) is essential for maintaining temporal coherence in generated videos, some methods [7, 48] introduce temporal convolution to capture positional relationships of frames within a local range. However, temporal convolution may struggle with building long-range temporal correlations over frames [52]. To this end, many recent methods [3, 6, 16, 28, 50] also incorporate tempo-

ral transformers equipped with positional encoding techniques [40, 42, 45] to capture precise positional relationships over frames, facilitating temporally coherent generation. Among these, relative positional encodings [40, 42], which can effectively capture inter-token relationships in LLMs, have been widely adopted in recent video diffusion models [6, 16, 24, 50], achieving impressive video generation results. In this paper, we mainly focus on video models that incorporate temporal transformers with relative positional encodings. Specifically, we aim to tackle two key challenges in long video generation and propose LongDiff, a training-free method to adapt these short video models to generate high-quality long videos.

Long Sequence Generalization is a common problem arising in various AI tasks, such as automatic speech recognition [33, 57], long text comprehension [8, 18, 26, 53], human activity analysis [1, 12, 14, 36, 56], and long video generation [19, 32, 35, 47]. Among these tasks, long video generation is particularly challenging due to the need to maintain both temporal content consistency and visual details throughout the sequence. To achieve high-quality long video generation, many recent advancements focus on improving visual quality using diffusion-based techniques [19, 21, 46, 55]. Nuwa-XL [55] employs a parallel diffusion process, while StreamingT2V [22] uses an auto-regressive approach with a short-long memory block to enhance the consistency of long video sequences. Despite their effectiveness, these methods require substantial computational resources and large-scale datasets for training.

Recently, training-free methods [32, 35, 47] that adapt off-the-shelf short video models to generate long videos without parameter updates have gained attention. Gen-L-Video [47] extends videos by merging overlapping subsegments using a sliding-window method during denoising. FreeNoise [35] uses rescheduled noise sequences and window-based temporal attention to improve video continuity. Freelong [32] explores long video generation in the frequency domain, improving generation quality by mixing features of different frequencies via Fourier transform. Differently, in this work, we focus on two challenges, revealed by theoretical analysis, that hinder short-to-long generalization of short video models. Motivated by the analysis, we find that simple modifications to temporal transformer can enable high-quality long video generation in one go.

3. Preliminaries

Temporal Transformer in Short Video Diffusion Models. Most of existing short video diffusion models [3, 6, 21, 23, 24, 50] are built on the 3D U-Net architecture, where both spatial and temporal transformer layers play important roles in video generation. In the spatial transformer layer, video features (i.e., hidden states of the sampled video) are processed independently over frames, allowing these layers

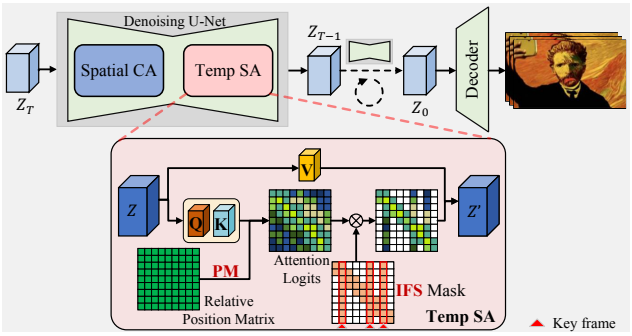

Figure 2. Overview of our proposed method. CA and SA denote cross-attention and self-attention, respectively. During the denoising steps, the video hidden states Z iteratively pass through temporal transformer layers where our LongDiff mechanism is applied. LongDiff comprises two key components: Position Mapping (PM) and Informative Frame Selection (IFS), corresponding to two modifications to temporal self-attention. First, we transform the original relative position matrix (the green matrix) via PM to alleviate the temporal position ambiguity issue. Additionally, a specially designed IFS mask restricts the temporal correlations of each query frame to both its neighbor frames and a set of detected key frames, to avoid the problem of information dilution.

to remain unaffected by the length of the video. Given this property, we mainly focus on the temporal transformer layers, as they play a significant role in handling the challenges of long video generation. The temporal transformer, responsible for establishing correlations over frames, is computed as follows:

$$w_{i,j} = \frac{\exp(a_{i,j})}{\sum_{k=1}^{N} \exp(a_{i,k})}, \quad a_{i,j} = f(\mathbf{q}_i, \mathbf{k}_j, i-j)$$
$$\mathbf{o}_i = \sum_{j=1}^{N} w_{i,j} \mathbf{v}_j \tag{1}$$

where $a_{i,j}$ and $w_{i,j}$ are the attention logit and attention weight between the i-th query frame and the j-th key frame, respectively. $\mathbf{q}_i$, $\mathbf{k}_j$, and $\mathbf{v}_j$ are the query, key, and value vectors, respectively. $f(\mathbf{q}_i, \mathbf{k}_j, i-j)$ is a function that computes the attention logit by taking into account the query frame i and key frame j, together with their relative position $(i-j)$. N is the number of frames in the video, and the output $\mathbf{o}_i$ is the aggregated information for the i-th frame. The form of f varies for different relative encoding mechanisms. For simplicity, we represent $f(\mathbf{q}_i, \mathbf{k}_j, i-j)$ as $f(\mathbf{q}, \mathbf{k}, p)$, where the relative position of the query and key is given by $p = i-j$. Notably, when generating a video with N frames, the positions of query and key frames are typically within the range $[0, N-1]$. Hence, the range of relative positions p between query and key frames is $[-(N-1), N-1]$.

4. Proposed Method

We aim to adapt off-the-shelf short video models to generate high-quality long videos in one go. To this end, we

propose our training-free solution, LongDiff. As shown in Fig. 2, LongDiff involves subtle modifications – Position Mapping (PM) and Informative Frame Selection (IFS) – to temporal attention in short video models, which are carefully designed to tackle two challenges in long video generation, namely *temporal position ambiguity* and *information dilution*. Below, we elaborate on how these two challenges are revealed, and introduce how the proposed solutions address these challenges.

4.1. Maintaining Temporal Consistency

Challenge: Temporal Position Ambiguity. Ensuring temporal consistency in video generation is crucial for maintaining realistic and smooth transitions between frames. This is linked to relative positional encoding (RPE), a technique used in various video generation models to encode relative positions of frames. By capturing positional differences, RPE helps models to maintain temporal coherence, guiding them to generate frames in the correct order. It is reasonable to conjecture that one reason of the loss of temporal consistency in long video generation can be that *RPEs in temporal transformer layers fail to function as intended when generating longer videos.* This is evident in some video diffusion models [6, 24], which employ the RPE mechanism [40] to encode relative positions only within a fixed range. When generating long videos, these models assign (clip) all relative positions exceeding the maximum encoding range to the same boundary position value, which impedes the model's ability to recognize frame order. However, even with recent RPE techniques, like RoPE [42], which theoretically can encode relative positions in sequences significantly longer than that required for long video generation, video diffusion models [50, 54] still struggle to maintain temporal consistency in long video generation. Inspired by [18], leveraging the pseudo-dimension technique [11, 31, 34], which is a metric used to assess the expressive capacity of nonlinear functions, this problem can be further investigated through the following theoretical explanation.

Theorem 1. *Define the attention logit function in temporal attention as $f(\mathbf{q}, \mathbf{k}, p)$, which maps the query frame $\mathbf{q}$, key frame $\mathbf{k}$, and their relative position p to a scalar value. Consider a video generation task with N frames, where the model categorizes the $2N-1$ relative positions[1] into $g(N)$ groups. Here, $g(N) \in \mathbb{N}$ is a non-decreasing and unbounded function representing the model's capability to differentiate relative positions. Additionally, assume that any two relative positions p and p' within the same group satisfy $d_f(p, p') \leq \epsilon$, where d_f is the distance function associated with the attention logit function f. Then the following holds:*

$$\sup_{-(N-1)\leq p\leq N-1} |f(\mathbf{q}, \mathbf{k}, p)| \geq \left(\frac{g(N)}{2}\right)^{\frac{1}{2r}} \frac{\epsilon}{4e} \quad (2)$$

where r is the pseudo-dimension of the function class $\mathcal{H} = \{f(\cdot, \cdot, p) \mid p \in \mathbb{Z}\}$, and e is the Euler's number.

For a detailed proof of this theoretical explanation, please refer to Supplementary. Theorem 3 illustrates that video models' ability to distinguish between different relative positions (i.e., $g(N)$) is limited by the supremum of the temporal attention logits. Notably, to ensure that the video is generated in the correct frame order, the ideal video model should satisfy $g(N) = 2N - 1$. Otherwise, at least one pair of distinct relative positions will be indistinguishable by the model. In other words, there will exist (p, p') (where $p \neq p'$ and $-(N-1) \leq p, p' \leq N-1$) such that $d_f(p, p') \leq \epsilon$, i.e., there will be temporal position ambiguity. Therefore, as the video sequence length N increases, we expect $g(N)$ to also increase accordingly in order to ensure position distinguishability. This necessitates larger supremum of the attention logits to satisfy the inequality in Eq. (2). However, through experiments (with details in Supplementary), we observe in long video generation, e.g., videos with 128 frames, less than 40% (the percentage is even smaller for longer videos) of the query and key features satisfy Eq. (2) when substituting $g(N)$ with $2N-1$, which is the requirement for the model to correctly identify frame order. This indicates the limited ability of using off-the-shelf short video models when distinguishing large numbers of distinct relative positions in long sequences, inevitably leading to degenerated temporal consistency for long video generation.

Solution: Position Mapping (PM). According to Theorem 3, an effective way to alleviate temporal position ambiguity is by limiting the number of distinct relative positions processed by the model. This can be achieved through a simple clipping operation [18], which sets a position threshold and maps all relative positions exceeding it to the threshold value. However, this method prevents the model from distinguishing between different relative positions larger than this threshold, limiting its ability to establish correlations over distant frames. Moreover, Theorem 3 also implies that position interpolation [8], which down-scales input relative position indices to fit within the pretraining position range, remains suboptimal, because it essentially introduces more relative positions within a short range. This requires the model to enhance its ability to distinguish positions, which is challenging in a training-free setting.

To tackle the temporal position ambiguity issue, here we propose a method that divide the relative positions in long video sequences into several position groups. Instead of using the original positions, we use the group indices to reference the corresponding positional encodings for temporal

[1] There are $2N-1$ relative positions, because the indices of relative positions range from $-(N-1)$ to $N-1$, as discussed in Sec. 3.

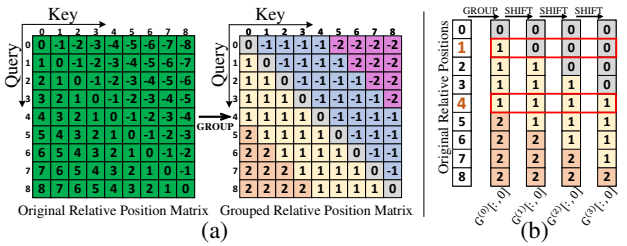

Figure 3. Figure (a) shows the GROUP operation where $N = 9$ and $G = 3$. The query-axis and key-axis of the matrices represent positions of the query frames and key frames, respectively. Each matrix entry represents the relative position between the query and key frames. In the grouped relative position matrix, the 17 original relative positions (ranging from -8 to 8) are grouped into 5 groups (from -2 to 2). Figure (b) shows a simple case of the SHIFT operation on the first column of the shifted relative position matrix $\mathbf{G}^{(m)}$. The red box represents the "assignment record".

attention computation. This GROUP operation reduces the need to manage an excessive number of indistinguishable relative positions by mapping them to a smaller set of indices, while still approximately preserving the overall positional relationships over the video sequence. However, the model still cannot distinguish relative positions within each group. Hence, we further design a SHIFT operation to recover fine-grained position distinguishability. Notably, our solution is compatible with video diffusion models that unitize different RPE techniques.

(1) Position Grouping. For a long video with N frames, we map the $2N-1$ relative positions (from $-(N-1)$ to $N-1$) into $2G-1$ groups, with group indices spanning from $-(G-1)$ to $G-1$, where G is a hyperparameter satisfying $G <= N$. The GROUP operation is formulated as:

$$p_g = \begin{cases} \lceil \frac{p}{\lceil \frac{N-1}{G-1} \rceil} \rceil, & \text{if } p \geq 0, \\ \lfloor \frac{p}{\lceil \frac{N-1}{G-1} \rceil} \rfloor, & \text{if } p < 0. \end{cases} \tag{3}$$

where p_g denotes the group index, and p is the original relative position between the query and key frames. $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the ceil and floor operations, respectively. Notably, except for the 0-*th* and the last groups, each group encapsulates $S = \lceil \frac{N-1}{G-1} \rceil$ original relative positions. The group indices are then used as the positional encoding indices in the temporal attention computation. A visual example of the GROUP operation is provided in Fig. 3(a), where $N = 9$ and $G = 3$.

(2) Position Shifting. While the GROUP enables the model to manage a reduced set of relative positions, it introduces ambiguity in frame order, as the model cannot discern the relative positions within the same position group. For example, as shown in Fig. 3(a), original relative positions $[1, 2, 3, 4]$ are now all mapped into the 1-*th* group, and $[1, 1, 1, 1]$ are then used as their actual relative positions. To tackle this problem, we recover position distinguishability within each group through a tailored SHIFT operation. This SHIFT operation is applied to the grouped position matrix (denoted as $\mathbf{G}^{(0)} \in \mathbb{R}^{N \times N}$) obtained through the GROUP operation. We denote the position matrix after the m-*th* shift as $\mathbf{G}^{(m)}$. To simplify the explanation of the SHIFT operation, we take the entries in the first column of $\mathbf{G}^{(0)}$ as an example. As shown in Fig. 3(b), given $\mathbf{G}^{(0)}[:, 0]$, which contains 3 groups (namely, 0, 1 and 2) mapped from the original relative positions, we apply a downward shift (padding zeros on the top) to the entries in $\mathbf{G}^{(0)}[:, 0]$. After each SHIFT operation, each entry is assigned a relative position, which may differ from its initial one. After three SHIFT operations, each entry accumulates four position assignments (including the initial grouped position). For example, the second entry (with original position value of 1) has an assignment record of $[1, 0, 0, 0]$, while the fifth entry (with original position value of 4) has $[1, 1, 1, 1]$. In this case, each entry thereby receives a unique "assignment record", where the sum of the record values for each entry exactly corresponds to its original position, as shown in the red boxes in Fig. 3(b). This assignment record allows us to distinguish entries even within the same group. As we aim for a training-free approach, we leverage the distinctness of the assignment record to distinguish each entry by separately computing temporal attention using its assigned positions and then simply average the resulting softmax attentions. Below we introduce the details of how the SHIFT operation is applied to the entire position matrix $\mathbf{G}^{(m)}$.

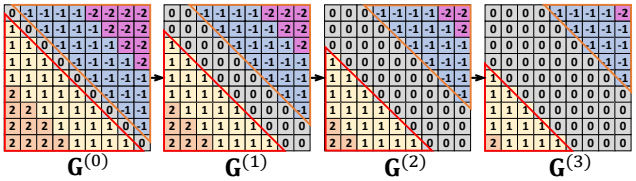

Figure 4. Illustration of the SHIFT operation. In each SHIFT operation, each entry in the upper triangle is shifted to the right by one position, with zeros added at the left. Meanwhile, each entry in the lower triangle is shifted downward by one position, with zeros added at the top.

As shown in Fig. 3(a), relative position matrices are always anti-symmetric, which exhibit beneficial properties, namely, the relative position of a frame with itself is always 0, and for any two frames, swapping their positions results in the relative position value being negated. Hence, when shifting a certain entry in such an anti-symmetric position matrix, the entry at its symmetric position should also change accordingly. With this insight, as shown in Fig. 4, in each SHIFT operation, we keep the diagonal entries as 0 and shift each entry in the lower triangle downward by one position. Additionally, we simultaneously update the corresponding entries in the upper triangle to ensure $\mathbf{G}^{(m)}_{i,j} = -\mathbf{G}^{(m)}_{j,i}$ due to the anti-symmetric property, which can be achieved through shifting entries row-wise in

the upper triangle. Our SHIFT operation is formulated as:

$$\mathbf{G}_{i,j}^{(m+1)} = \begin{cases} \mathbf{G}_{i,j-1}^{(m)}, & \text{if } i < j, \\ \mathbf{G}_{i,j}^{(m)}, & \text{if } i = j, \\ \mathbf{G}_{i-1,j}^{(m)}, & \text{if } i > j. \end{cases} \tag{4}$$

Notably, the number of required shifts (M) is determined by the size of each position group (S), and is given by $M = S - 1$. Then, we use the $M + 1$ relative position matrices $\{\mathbf{G}^{(m)}\}_{m=0}^{M}$ to compute the temporal attention, respectively. We average the resulting softmax-attentions as the final output of the temporal attention:

$$\mathbf{A}_{i,j} = \frac{1}{M+1} \sum_{m=0}^{M} \mathbf{A}_{i,j}^{(m)}, \ \mathbf{A}_{i,j}^{(m)} = \frac{\exp(f(\mathbf{q}_i, \mathbf{k}_j, \mathbf{G}_{i,j}^{(m)}))}{\sum_k \exp(f(\mathbf{q}_i, \mathbf{k}_k, \mathbf{G}_{i,k}^{(m)}))} \tag{5}$$

where $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the final temporal attention output, and f is the function described in Eq. (1) that computes the attention logit, taking the relative position into account.

Through the aforementioned GROUP and SHIFT operations, PM maps large numbers of distinct relative positions into several position groups while maintaining their distinctness. It is worth noting that the temporal attention computation is only a part of the entire denoising process, and the M SHIFT operations can be performed in parallel. Thus, our PM results in only a minor decrease in inference speed.

4.2. Maintaining Visual Details

Challenge: Information Dilution. As shown in Fig. 1, aside from inferior temporal consistency, directly generating long videos using short video models results in a lack of visual details, including blurred textures and missing critical details. This problem can be further investigated with the following theoretical explanation, which is obtained by interpreting the information passing mechanism of the temporal transformer from the perspective of information entropy [18]:

Theorem 2. *When generating a video with N frames, the information entropy H of temporal correlations over frames of the video sequence, is lower bounded by:*

$$H\left(\frac{e^{a_i}}{\Sigma_{j=1}^{N} e^{a_j}} | 1 \le i \le N\right) \ge \ln N - 2B, \tag{6}$$

where $\{a_i\}_{i=1}^{N}$ are the attention logits with boundary $[-B, B]$.

For a detailed proof of Theorem 4, please refer to Supplementary. Eq. (18) implies that as the video length N increases, the information entropy H of the temporal correlations over frames exhibits an increasing trend. This thus indicates a decrease in the effective information carried by each frame during video generation, which may cause the model missing detailed information in some key frames and finally resulting a lack of visual details in long videos.

Solution: Informative Frame Selection (IFS). Based on Theorem 4, the information dilution issue in long video generation is related to the number of frames involved in information passing during attention. Hence, it seemingly can be alleviated by using a fixed-length attention window to restrict information passing to neighbor frames [18, 35]. This approach indeed maintains visual details, which however hinders long-range interactions between remoter frames, leading to a lack of global temporal consistency in the generated videos. Different from this approach [18, 35], we propose to limit the temporal correlations for each frame to both its neighbor frames and a set of selected key frames that serve as a summary of the entire video. This design only uses informative frames for information exchange, reducing the risk of excessive information entropy of temporal correlations during generation, while maintaining global consistency by capturing the entire video's information through selected key frames.

Specifically, to select frames containing important information for video generation, we exploit a simple yet effective key-frame detection mechanism [10] to identify frames that may reflect scene changes and significant events in the video. Additionally, because no real video data exists until the generation process is complete, we convert the input sequential features of each temporal transformer layer into a pseudo-video to ensure compatibility with our key-frame detection pipeline.

(1) Pseudo-Video Construction. The input video feature F (i.e., hidden states of the sampled video) of each temporal transformer layer has dimensions $\mathbb{R}^{N \times C \times hw}$, where N is the number of video frames, h and w are the sizes of each frame's feature map, and C is the number of channels. To make F suitable for the existing key-frame detection pipeline, we perform max-pooling, average-pooling, and min-pooling operations along the channel dimension, and then stack these 3 pooled features along the channel dimension. This yields a down-sampled video feature $F' \in \mathbb{R}^{N \times 3 \times hw}$, reserving essential semantic information from the original video feature while matching the input shape required by the key-frame detection pipeline. We further normalize and map values of F' to integer values within the range $[0, 255]$, simulating real video data. We denote the resulted pseudo-video as V:

$$V = \text{round}\left(\frac{F' - \min(F')}{\max(F') - \min(F')} \times 255\right) \tag{7}$$

(2) Key-Frame Detection. After obtaining the pseudo-video data V, we employ a simple and efficient key-frame detection mechanism [10]. Specifically, we first uniformly divide V into n video shots, and then select one key frame in each video shot. We introduce image entropy, which reflects the complexity and information content of an image, to assess the importance of each frame in the video shot. The image entropy is computed based on the distribution of

pixel values, defined as follows:

$$H(k) = -\Sigma_x p(x,k)\log_2(p(x,k)) \quad (8)$$

where $p(x,k)$ is the probability of the luminance value x in the appearance histogram of the k-*th* frame in the video shot.

In addition, we also consider frame differencing, which compares the pixel-wise differences between consecutive frames to identify video content changes. That is:

$$SAD(k) = \sum_{i,j} |I(i,j,k) - I(i,j,k-1)| \quad (9)$$

where $I(i,j,k)$ denotes the pixel value of the k-th frame at position (i,j). The importance score of the frame is given by combining these two measures:

$$Score(k) = \alpha H(k) + SAD(k) \quad (10)$$

where α is a weighting factor. The frame with the highest score in each video shot is selected as a key frame.

(3) IFS Mask. Finally, we insert a specially designed mask into the temporal attention to restrict the temporal correlations of each frame to only its neighbor frames and the selected key frames. This mask is formulated as:

$$\text{Mask}_{ij} = \begin{cases} 1, & \text{if } |i-j| \le L \text{ or } j \text{ is a key frame,} \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

where L is a hyperparameter determining the range of used neighbor frames.

4.3. Overall Inference

During video generation, features are iteratively passed through temporal attention layers, where LongDiff operates. Specifically, before temporal attention computation, each entry in the original relative position matrix is assigned new positions through the `GROUP` and `SHIFT` operations, resulting in new position matrices ($\{\mathbf{G}^{(m)}\}_{m=0}^{M}$). These position matrices are then used for temporal attention computation. In this process, the IFS mask filters out uninformative frames by making their associated softmax-attention values to 0. The resulting softmax attention values are averaged to obtain the final temporal attention (see Eq. (5)).

5. Experiments

Implementation Details. To evaluate the effectiveness of our proposed LongDiff, we conduct experiments using open-source diffusion-based text-to-video generation models: LaVie [50] and VideoCrafter-512 [6] (more models in Supplementary). LaVie utilizes Rotary Position Embedding (RoPE) [42] as RPE to capture relative positions across frames, while VideoCrafter-512 employs the RPE mechanism from [40]. Both models are trained to generate 16-frame short videos at a 320×512 resolution. We equip them with our LongDiff to generate long videos with 128 frames. More implementation details are in Supplementary.

Evaluation Metrics. Following FreeLong [32], we use metrics from VBench [25] to evaluate video quality, including Subject Consistency (SC), Background Consistency (BC), Motion Smoothness (MS), Temporal Flickering (TF), Imaging Quality (IQ), and Overall Consistency (OC). More details about these metrics are in Supplementary. We use the same 200 test prompts from VBench [25] as FreeLong [32] for evaluation.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
VideoCrafter-512 + Direct	88.62	91.86	85.30	78.53	65.38	20.71
VideoCrafter-512 + Sliding	83.75	91.14	92.12	90.58	67.25	21.51
VideoCrafter-512 + FreeNoise[35]	91.43	93.48	93.33	91.88	68.39	22.69
VideoCrafter-512 + FreeLong [32]	90.84	92.37	89.11	88.46	66.62	21.85
VideoCrafter-512 + Our LongDiff	**93.69**	**95.59**	**94.59**	**93.35**	**70.03**	**23.17**
LaVie + Direct	88.95	93.23	92.77	91.44	64.76	22.34
LaVie + Sliding	85.80	92.83	95.79	94.00	66.57	23.46
LaVie + FreeNoise [35]	92.30	95.87	96.32	94.94	67.14	24.42
LaVie + FreeLong [32]	95.16	96.80	96.85	96.04	67.55	24.56
LaVie + Our LongDiff	**98.10**	**98.23**	**97.46**	**96.84**	**68.83**	**25.24**

Table 1. Quantitative comparisons of longer video generation (128 frames) on the video models VideoCrafter-512 and LaVie. Results on more short video models are provided in Supplementary.

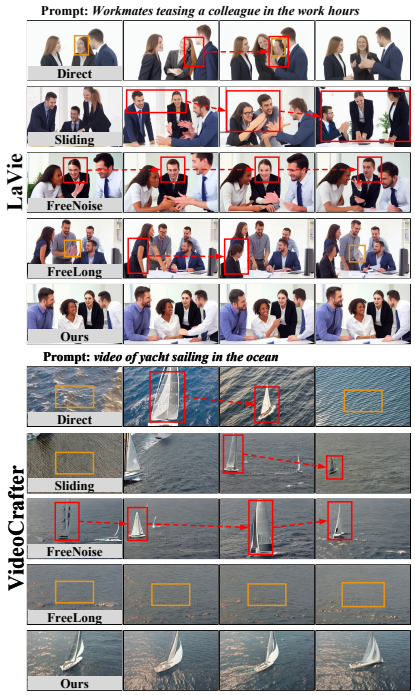

Figure 5. Qualitative comparisons of longer video generation. (128 frames). We illustrate inferior temporal consistency and the lack of visual details using red and orange boxes, respectively. More examples are in Supplementary.

5.1. Evaluation on Long Video Generation

We compare our LongDiff with other training-free diffusion-based long video generation methods [32, 35] and two basic methods, called **Direct** and **Sliding**. In Direct, long videos are generated directly from short video models by extending the initial noise sequence as the starting point

for denoising. In Sliding, we adopt temporal sliding windows [35] to process a fixed number of frames at a time. We generate videos with 128 frames for comparison. Tab. 1 displays the comparative results. We observe that videos generated by Direct suffer significantly in quality due to short-to-long generalization challenges. Compared to Direct, the methods of Sliding, FreeNoise, and FreeLong achieve better performance. Among them, Sliding and FreeNoise improve video quality by employing sliding-windowed attention to build temporal correlations only within neighbor frames. On the other hand, FreeLong improves long video generation through fusing spatial and temporal information in the frequency domain. Compared to these methods, our method achieves the best performance across all metrics, demonstrating the efficacy of the proposed LongDiff. We show qualitative comparisons in Fig. 5.

5.2. Ablation Studies

In this section, we equip the short video model LaVie [50] with our LongDiff for analysis. By default, the short video model is trained on 16-frame videos, and we adapt it to generate 128-frame videos. More results are in Supplementary.

Impact of Main Components of LongDiff. First, we verify the impact of the key components of LongDiff by comparing the following variants: 1) **Ours (w/o PM)**, where we remove the `GROUP` and `SHIFT` operations and thus use the original relative positions to compute temporal attention. 2) **Ours (w/o IFS)**, where we remove the IFS mask in temporal attention computation, requiring each query frame to correlate with all frames for information passing during video generation. The results in Tab. 2 indicate that both PM and IFS significantly contribute to LongDiff's performance.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
Ours (w/o PM)	91.85	94.79	95.12	93.26	65.73	22.58
Ours (w/o IFS)	94.43	96.37	93.65	92.85	65.45	23.46
Ours	98.10	98.23	97.46	96.84	68.83	25.24

Table 2. Ablation study for main components of LongDiff.

Impact of Operations in Position Mapping. Next, we explore the impact of operations used in PM, by comparing the following variants: 1) **Clip**, where relative positions that exceed the range encountered during pretraining are set to the maximum or minimum relative position within that range. 2) **Interpolation**, where we linearly downscale input relative positions to fit within the pretraining position range. 3) **Group**, where we remove the `SHIFT` operation, and only apply the `GROUP` operation for position mapping. As shown in Tab. 3, our method achieves better performance than all other variants, demonstrating the efficacy of PM that allows the model to avoid large numbers of distinct relative positions while preserving position distinguishability.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
Clip	91.02	94.18	94.77	92.87	65.12	22.43
Interpolation	92.52	95.29	95.25	94.76	66.43	23.01
Group	94.49	96.63	96.74	95.79	67.45	24.62
Our PM	98.10	98.23	97.46	96.84	68.83	25.24

Table 3. Ablation study for operations in PM.

Impact of Different Informative Frame Selection Mechanisms. In our LongDiff, the IFS mask is used to select informative frames for information passing in temporal attention computation. Here, we further explore different frame selection mechanisms by comparing the following variants: 1) **Neighbor**, where for each query frame, we remove the detected key frames and only select its neighbor frames following [35] for temporal attention computation. 2) **Neighbor (Plus)**, where for each query frame, compared to the Neighbor variant, we additionally select n (the number of detected key frames) neighbor frames. 3) **Key Frame**, where for each query frame, we remove its neighbor frames and only select the detected key frames. 4) **Key Frame (Plus)**, where for each query frame, compared to the Key Frame variant, we additionally select $2L$ (the number of neighbor frames) detected key frames. 5) **Neighbor+Uniform**, where, besides neighbor frames, we uniformly select frames at equal intervals from the entire sequence and allow each frame to attend to these selected frames for attention computation. 6) **Neighbor+Random**, where, besides neighbor frames, we randomly select frames from the entire sequence. As shown in Tab. 4, we observe that all the variants yield inferior results. This suggests that both neighbor and key frames carry informative content. Our IFS approach, which selects a proper combination of these frames for information passing during video generation, achieves the best performance.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
Neighbor	90.12	93.54	95.31	94.27	66.94	24.15
Neighbor (Plus)	90.33	93.59	94.87	94.29	67.13	24.21
Key Frame	90.29	94.21	93.20	92.33	62.83	22.12
Key Frame (Plus)	90.17	94.08	93.47	92.55	63.00	22.33
Neighbor+Uniform	96.33	97.09	96.51	96.29	68.20	24.85
Neighbor+Random	95.77	96.64	96.23	96.14	67.77	24.69
Our IFS	98.10	98.23	97.46	96.84	68.83	25.24

Table 4. Ablation study for different informative frame selection mechanisms.

6. Conclusion

In this paper, we propose LongDiff, a novel training-free method that involves only minor modifications, namely position mapping and informative frame selection, to temporal transformer layers. These modifications are carefully designed to tackle two key challenges in generating long videos through short video diffusion models: *temporal position ambiguity* and *information dilution*. Extensive experiments demonstrate that LongDiff significantly outperforms existing training-free methods, achieving high-quality long video generation in one go.

Acknowledgement. This research was supported by the Australian Government through the Australian Research Council's DECRA funding scheme (Grant No.: DE250100030).

References

[1] Dasom Ahn, Sangwon Kim, Hyunsu Hong, and Byoung Chul Ko. Star-transformer: a spatio-temporal cross attention transformer for human action recognition. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3330–3339, 2023. 3

[2] Fan Bao, Chendong Xiang, Gang Yue, Guande He, Hongzhou Zhu, Kaiwen Zheng, Min Zhao, Shilong Liu, Yaole Wang, and Jun Zhu. Vidu: a highly consistent, dynamic and skilled text-to-video generator with diffusion models. *arXiv preprint arXiv:2405.04233*, 2024. 1

[3] 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. 1, 2, 3

[4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1

[5] 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. 1

[6] 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. *arXiv preprint arXiv:2310.19512*, 2023. 1, 2, 3, 4, 7

[7] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024. 2

[8] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023. 3, 4

[9] Iván J Pérez Colado, Víctor M Pérez Colado, Antonio Calvo Morata, Rubén Santa Cruz Píriz, and Baltasar Fernández Manjón. Using new ai-driven techniques to ease serious games authoring. In *2023 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE, 2023. 1

[10] Frédéric Dirfaux. Key frame selection to represent a video. In *Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*, pages 275–278. IEEE, 2000. 6

[11] Simone Fioravanti, Michele Flammini, Bojana Kodric, and Giovanna Varricchio. Pac learning and stabilizing hedonic games: towards a unifying approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5641–5648, 2023. 4

[12] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. Unified pose sequence modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13019–13030, 2023. 3

[13] Lin Geng Foo, Hossein Rahmani, and Jun Liu. Ai-generated content (aigc) for various data modalities: A survey. *arXiv preprint arXiv:2308.14177*, 2023. 1

[14] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, and Jun Liu. Action detection via an image diffusion process. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18351–18361, 2024. 3

[15] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *European Conference on Computer Vision*, pages 102–118. Springer, 2022. 1

[16] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*, 2023. 1, 2, 3

[17] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 6

[18] Chi Han, Qifan Wang, Hao Peng, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. Lm-infinite: Zero-shot extreme length generalization for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3991–4008, 2024. 2, 3, 4, 6, 5

[19] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *Advances in Neural Information Processing Systems*, 35:27953–27965, 2022. 3

[20] David Haussler. Decision theoretic generalizations of the pac model for neural net and other learning applications. *Information and computation*, 100(1):78–150, 1992. 3

[21] 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*, 2(3):4, 2022. 1, 2, 3

[22] Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024. 1, 3

[23] 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. *arXiv preprint arXiv:2210.02303*, 2022. 1, 2, 3

[24] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022. 1, 2, 3, 4

[25] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 7, 1

[26] Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen, and Xia Hu. Llm maybe longlm: Self-extend llm context window without tuning. *arXiv preprint arXiv:2401.01325*, 2024. 3

[27] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5148–5157, 2021. 1

[28] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023. 1, 2

[29] Chengxuan Li, Di Huang, Zeyu Lu, Yang Xiao, Qingqi Pei, and Lei Bai. A survey on long video generation: Challenges, methods, and prospects. *arXiv preprint arXiv:2403.16407*, 2024. 1

[30] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9801–9810, 2023. 1

[31] Suqi Liu, Tianxi Cai, and Xiaoou Li. Representation-enhanced neural knowledge integration with application to large-scale medical ontology learning. *arXiv preprint arXiv:2410.07454*, 2024. 4

[32] Yu Lu, Yuanzhi Liang, Linchao Zhu, and Yi Yang. Freelong: Training-free long video generation with spectralblend temporal attention. *Advances in Neural Information Processing Systems*, 2024. 1, 3, 7, 2, 4, 5, 6

[33] Niko Moritz, Takaaki Hori, and Jonathan Le Roux. Triggered attention for end-to-end speech recognition. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5666–5670. IEEE, 2019. 3

[34] David Pollard. Empirical processes: theory and applications. Ims, 1990. 4

[35] Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. Freenoise: Tuning-free longer video diffusion via noise rescheduling. In *International Conference on Learning Representations (ICLR)*, 2024. 1, 3, 6, 7, 8, 2, 4, 5

[36] Haoxuan Qu, Yujun Cai, and Jun Liu. Llms are good action recognizers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18395–18406, 2024. 3

[37] 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. 1

[38] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021. 2

[39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[40] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018. 2, 3, 4, 7

[41] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3626–3636, 2022. 1

[42] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 2, 3, 4, 7

[43] 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. 1

[44] Ketan Totlani. The evolution of generative ai: Implications for the media and film industry. *International Journal for Research in Applied Science and Engineering Technology*, 2023. 1

[45] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 3, 6

[46] 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. 3

[47] 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. 1, 3

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

[49] Xudong Wang and Weiyi Zhong. Evolution and innovations in animation: A comprehensive review and future directions. *Concurrency and Computation: Practice and Experience*, 36 (2):e7904, 2024. 1

[50] Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, et al. Lavie: High-quality video generation with cascaded latent diffusion models. *arXiv preprint arXiv:2309.15103*, 2023. 1, 2, 3, 4, 7, 8

[51] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui

Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023. 1

[52] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*, 2022. 2

[53] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023. 3

[54] 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. *arXiv preprint arXiv:2408.06072*, 2024. 2, 4

[55] 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. 1, 3

[56] Hongguang Zhang, Li Zhang, Xiaojuan Qi, Hongdong Li, Philip HS Torr, and Piotr Koniusz. Few-shot action recognition with permutation-invariant attention. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 525–542. Springer, 2020. 3

[57] Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, et al. Google usm: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*, 2023. 3

LongDiff: Training-Free Long Video Generation in One Go

Supplementary Material

A. More Implementation Details

In our main paper, we equip LaVie [50] and VideoCrafter [6] with our LongDiff to generate 128-frame (i.e., $N = 128$) long videos. Following [32, 35], during sampling, we employ the noise shuffle mechanism and perform DDIM sampling with 50 denoising steps. We set G in Eq.(3) to 16. The weighting factor α in Eq.(10) is set to 2. We set the neighbor range L in Eq.(11) to 8. In addition to the neighbor frames, we also select $n = 8$ key frames for temporal attention computation. Notably, we uniformly sample 50% of the temporal attention layers in the short video model and replace them with our LongDiff module. All experiments are conducted using NVIDIA 6000 Ada GPUs.

B. More Evaluation Metrics Details

Following FreeLong [32], we use metrics from VBench [25] to evaluate video quality. For video consistency, we report: 1) Subject Consistency (SC), measured by DINO [5] feature similarity across frames, to check object appearance stability, and 2) Background Consistency (BC), calculated with CLIP [37] feature similarity across frames. For video fidelity, we assess 1) Motion Smoothness (MS) using AMT [30] motion priors, 2) Temporal Flickering (TF) via mean absolute difference between static frames, and 3) Imaging Quality (IQ), measured by MUSIQ [27]. For video-text consistency, we employ Overall Consistency (OC) from ViCLIP [51] to capture both semantic and style information.

C. Additional Ablation Studies

We here conduct more ablation experiments about our LongDiff based on the short video model LaVie.

C.1. Impact of the Number of Position Groups

In Position Mapping (PM), we map $2N-1$ (from $-(N-1)$ to $(N-1)$) original relative positions into $2G-1$ groups to make the model avoid handling large numbers of distinct positions. In our main paper, we set $G = 16$ for LaVie. Here we evaluate other choices of G, and report the results in Tab. 5. We find that performance improves when we increase G, until G reaches 16, where the improvement tapers off. Thus, we set G = 16.

C.2. Impact of the Number of Key Frames

In our LongDiff, we establish temporal correlations between each frame and its neighboring frames, as well as n key frames selected using a key-frame detection pipeline. Here, we also evaluate other choices of n. As shown in Tab. 6, the model performance reaches optimal results at $n = 8$. Thus, we set $n = 8$ in the experiments to achieve a good result.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
$G = 8$	93.47	95.68	95.72	94.19	66.53	23.77
$G = 12$	96.19	97.17	96.74	95.74	67.88	24.42
$G = 16$	98.10	98.23	97.46	96.84	68.83	25.24
$G = 20$	97.25	97.76	97.14	96.45	68.41	24.98

Table 5. Ablation study for the number of position groups.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
$n = 4$	92.96	95.98	96.97	94.67	67.65	24.65
$n = 6$	97.19	97.52	97.18	95.74	68.11	25.01
$n = 8$	98.10	98.23	97.46	96.84	68.83	25.24
$n = 10$	97.52	97.85	97.29	96.18	68.31	25.13

Table 6. Ablation study for the number of key frames.

C.3. Impact of the Number of Neighbor Frames

Here, we also explore the impact of using different numbers L of neighbor frames for temporal attention computation. As shown in Tab. 7, the model's performance reaches its highest value at $L = 8$. We thus set $L = 8$ in our experiments.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
$L = 2$	94.11	96.18	95.28	94.54	65.77	23.61
$L = 4$	96.90	97.61	96.84	96.15	67.91	24.76
$L = 8$	98.10	98.23	97.46	96.84	68.83	25.24
$L = 16$	96.43	97.37	96.55	95.87	67.58	24.59

Table 7. Ablation study for the number of neighbor frames.

C.4. Impact of the Mechanism to Down-Sample Video Features

In our LongDiff, we use a combination of max-pooling, average-pooling, and min-pooling operations to reduce the channel dimension of the video feature F to three channels, aligning it with the input shape required by the key-frame detection pipeline. Here, we evaluate the efficacy of this mechanism by comparing the following variants: 1) **Max**, where only the max-pooling operation is used, and the result is replicated three times along the channel dimension. 2) **Min**, where only the min-pooling operation is used, and

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
Max	97.28	97.70	97.02	96.58	68.37	25.06
Min	96.71	97.33	96.71	96.40	68.33	24.93
Average	97.67	97.95	97.23	96.73	68.68	25.14
Ours	98.10	98.23	97.46	96.84	68.83	25.24

Table 8. Ablation study for the mechanism to downsample video features.

the result is replicated three times along the channel dimension. 3) **Average**, where only the average-pooling operation is used, and the result is replicated three times along the channel dimension. As shown in Tab. 8, we observe that using the combination of max-pooling, average-pooling, and min-pooling operations achieves the best performance. Notably, all of these variants of our LongDiff consistently outperform the previous state-of-the-art methods[32, 35].

C.5. Impact of the Weighting Factor α

In our LongDiff, we use two measures—image entropy and frame differencing—to select the most important frame in each shot of the pseudo-video as a key frame. Here, we evaluate the impact of α, which weights these two measures, and report the results in Tab. 9. As shown, the model achieves the best performance with $\alpha = 2$. Therefore, we set α to 2 in our experiments to obtain optimal results.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
$\alpha = 0$	97.10	97.58	96.92	96.53	68.47	25.02
$\alpha = 1$	97.44	97.83	97.21	96.69	68.61	25.11
$\alpha = 2$	98.10	98.23	97.46	96.84	68.83	25.24
$\alpha = 3$	97.94	98.12	97.37	96.79	68.77	25.20

Table 9. Ablation study for the weighting factor α.

C.6. Impact of the Key-Frame Selection Measures

In LongDiff, we use the combination of two measures, image entropy and frame differencing, to select informative key frames by comparing the following variants: 1) **w/o Entropy**, where only image entropy is used as the sole measure to select key frames. 2) **w/o Differencing**, where frames are selected solely based on the frame differencing measure. As shown in Tab. 10, the combination of these two measures yields the best results. In addtion, both variants outperform previous training-free methods[32, 35].

C.7. Impact of the Proportion of LongDiff Modules

In our main experiments, we uniformly replace 50% of the temporal attention layers in the short video model with our LongDiff modules. Here, we explore the impact of varying the proportion of the number of replaced temporal attention layers with LongDiff modules. As shown in Tab. 11, the performance improves noticeably when the proportion of LongDiff is below 50%, and the improvement trend plateaus beyond this point. Based on this observation, we choose to uniformly replace 50% of the temporal attention layers with our LongDiff modules to achieve good results while maintaining efficiency.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
w/o Entropy	97.10	97.58	96.92	96.53	68.47	25.02
w/o Differencing	97.35	97.75	97.06	96.61	68.50	25.02
Ours	98.10	98.23	97.46	96.84	68.83	25.24

Table 10. Ablation study for the key-frame selection measures.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
Proportion 25.0%	94.86	96.14	96.07	95.30	66.53	23.94
Proportion 50.0%	98.10	98.23	97.46	96.84	68.83	25.24
Proportion 75.0%	98.40	98.51	97.63	96.98	68.92	25.25

Table 11. Ablation study for the proportion of LongDiff modules.

D. More Qualitative Results

In this section, we provide more qualitative results regarding the ablation study of the main components of LongDiff (see Fig. 6), longer video generation (see Fig. 7), multi-prompt video generation (see Fig. 8), and more generated videos (see Fig. 9 and Fig. 10).

E. Proofs

E.1. Detailed Proof of Theorem 1

Here, we provide a detailed proof of Theorem 1, which is mainly based on [18]. For ease of reading, we restate Theorem 1 from the main paper below.

Theorem 3. *Define the attention logit function in temporal attention as $f(\mathbf{q}, \mathbf{k}, p)$, which maps the query frame $\mathbf{q}$, key frame $\mathbf{k}$, and their relative position p to a scalar value. Consider a video generation task with N frames, where the model categorizes the $2N-1$ relative positions into $g(N)$ groups. Here, $g(N) \in \mathbb{N}$ is a non-decreasing and unbounded function representing the model's capability to differentiate relative positions. Additionally, assume that any two relative positions p and p' within the same group satisfy $d_f(p, p') \leq \epsilon$, where d_f is the distance function associated with the attention logit function f. Then the following holds:*

$$\sup_{-(N-1)\leq p\leq N-1} |f(\mathbf{q}, \mathbf{k}, p)| \geq \left(\frac{g(N)}{2}\right)^{\frac{1}{2r}} \frac{\epsilon}{4e} \qquad (12)$$

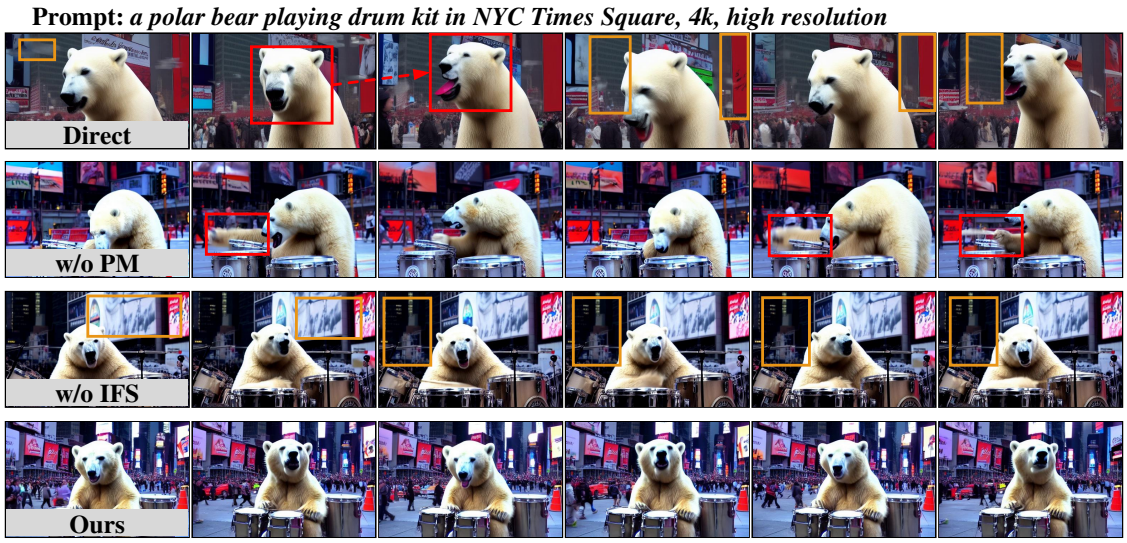

Figure 6. **Ablation Study of the Main Components of LongDiff.** Here, we show the qualitative comparison of LongDiff with two variants: **Ours (w/o PM)**, where we remove the `GROUP` and `SHIFT` operations and thus use the original relative positions to compute temporal attention. 2) **Ours (w/o IFS)**, where we remove the IFS mask in temporal attention computation, requiring each query frame to correlate with all frames for information passing during video generation. As shown, videos generated from the (w/o PM) variant exhibit abrupt temporal transition between frames, particularly noticeable in the bear's hand. On the other hand, videos generated from the (w/o IFS) variant lack some visual details, manifesting as as a blurry "NYC Times Square". We illustrate inferior temporal consistency and the the visual detail issues using red and orange boxes, respectively.

Figure 7. **Longer Video Generation.** Here, we equip VideoCrafter[6] with our LongDiff to generate 256-frame videos. As shown, these generated videos maintain temporal consistency and visual details. This further demonstrates the efficacy of out method.

where r is the pseudo-dimension of the function class $\mathcal{H} = \{f(\cdot,\cdot,p) \mid p \in \mathbb{Z}\}$, and e is the Euler's number.

The distance function d_f can be rewritten in a more detailed form as:

$$d_f(p,p') = \mathbb{E}_{\mathbf{q}\sim\mathbf{Q},\mathbf{k}\sim\mathbf{K}}(f(\mathbf{q},\mathbf{k},p) - f(\mathbf{q},\mathbf{k},p'))^2 \quad (13)$$

where $\mathbf{Q}$ and $\mathbf{K}$ are the trained distributions for $\mathbf{q}$ and $\mathbf{k}$. To assist in proving the inequality in Theorem 1, the following lemma is introduced from [20].

Lemma 1. *Let $\mathcal{H} = \{h(z)\}$ be a family of functions that map a set $\mathbf{Z}$ into $[0, M]$ with pseudo-dimension $\dim_P(\mathcal{H}) = r$, where $1 \le r < \infty$. Let P be a probability measure on $\mathbf{Z}$. Then, for all $0 < \epsilon \le M$, the ϵ-cover of $\mathcal{H}$ under the metric $d(h_1,h_2) = \mathbb{E}_{z\sim P}(h_1(z) - h_2(z))^2$ is bounded by:*

$$\mathcal{N}_P(\epsilon,\mathcal{H},d) \le 2\left(\frac{2eM}{\epsilon}\ln\frac{2eM}{\epsilon}\right)^r \quad (14)$$

where $N_P(\epsilon,\mathcal{H},d)$ is the cover size, defined as the smallest cardinal of a cover-set $\mathcal{H}'$ such that for every entry $h \in \mathcal{H}$, there exists at least one entry $h' \in \mathcal{H}'$ within ϵ distance from h.

Based on Lemma 1, Theorem 1 can be proven by contradiction as follows.

Proof. First, let the negation of Eq. (12) in Theorem 1 be assumed to hold:

$$\sup_{-(N-1)\le p\le N-1} |f(\mathbf{q},\mathbf{k},p)| < \left(\frac{g(N)}{2}\right)^{\frac{1}{2r}} \frac{\epsilon}{4e} = a \quad (15)$$

This indicates that the function family $\mathcal{H} = \{f(\cdot,\cdot,p)|p \in \mathbb{Z}\}$ maps the input to the range $[-a,a]$. Without loss of generality, all values from the range $[-a,a]$ can be shifted to the range $[0,2a]$ to apply Lemma 1. Then, according

Prompt 1: ***A waterfall flows in the mountains under a clear sky***
Prompt 2: ***A waterfall flows in the fall mountains under a clear sky***

Prompt 1: ***There is a beach where there is no one***
Prompt 2: ***The waves hit the deserted beach***
Prompt 3: ***There is a beach that has been swept away by waves***

Figure 8. **Multi-Prompt Video Generation.** Our LongDiff can be easily adapted for multi-prompt video generation by assigning distinct prompts to each video segment following [32, 35]. As shown, the output of our LongDiff maintains temporal consistency and visual details across different segments.

to Lemma 1[2], the ϵ-cover size $\mathcal{N}_P(\epsilon, \mathcal{H}, d_f)$ of $\mathcal{H}$ satisfies that:

$$\mathcal{N}_P(\epsilon, \mathcal{H}, d_f) \leq 2\left(\frac{4ea}{\epsilon}\ln\frac{4ea}{\epsilon}\right)^r \tag{16}$$

By substituting $a = \left(\frac{g(N)}{2}\right)^{\frac{1}{2r}}\frac{\epsilon}{4e}$ (defined in Eq. (15)) into Eq. (16), the following expression is obtained:

$$\begin{aligned}\mathcal{N}_P(\epsilon, \mathcal{H}, d_f) &\leq 2\left(\left(\frac{g(N)}{2}\right)^{\frac{1}{2r}}\ln\left(\frac{g(N)}{2}\right)^{\frac{1}{2r}}\right)^r \\ &< 2\left(\left(\frac{g(N)}{2}\right)^{\frac{1}{2r}}\left(\frac{g(N)}{2}\right)^{\frac{1}{2r}}\right)^r = g(N)\end{aligned} \tag{17}$$

This indicates that, if the assumption in Eq. (15) holds, the ϵ-cover size $\mathcal{N}_P(\epsilon, \mathcal{H}, d_f)$ is smaller than $g(N)$. In other words, we cannot find $g(N)$ distinct functions in the function family $\mathcal{H} = \{f(\cdot,\cdot,p) \mid p \in \mathbb{Z}\}$ such that the pairwise distances (measured by d_f) between them are greater than ϵ. This implies that the number of distinct relative positions differentiated by the model is less than $g(N)$, which contradicts the definition of $g(N)$. Therefore, Eq. (15) does not hold, and thus Eq. (12) in Theorem 1 is proven. □

Pseudo-Dimension of $\mathcal{H}$. As discussed above, Lemma 1 is introduced to assist in proving Theorem 3, which requires that $\mathcal{H}$ has a bounded pseudo-dimension $\dim_P(\mathcal{H}) = r$. Notably, $\mathcal{H} = \{f(\cdot,\cdot,p) \mid p \in \mathbb{Z}\}$ represents the family of attention logit functions, whose form varies depending on the RPE mechanisms. For RoPE, the logit function $f(\cdot,\cdot,p)$ can be expressed as a weighted sum of a finite set of sinusoidal functions $\{\sin(\omega_i p), \cos(\omega_i p)\}$, where the size of this set equals the feature dimension k. Based on the properties of pseudo-dimensions, it follows that $\dim_P(\mathcal{H}_1 + H_2) \leq \dim_P(\mathcal{H}_1) + \dim_P(\mathcal{H}_2)$, and the pseudo-dimension of scaling a single function is at most 2. Therefore, the pseudo-dimension of the whole family is bounded by $\dim_P(\mathcal{H}) \leq 2k$, which satisfies the requirement in Lemma 1.

Analysis of Theorem 1. Theorem 3 implies that, the ability of a video model to distinguish between different relative positions is constrained by the supremum of the model's temporal attention logits. Building on Theorem 3, here, we further analyze whether existing video models can accurately identify frame order during long video generation. Recall that for a video model to correctly identify frame order in a video of length N, it must be capable of distinguishing between $2N-1$ distinct relative positions using its temporal attention logits. According to Theorem 3, this requirement means that a video model capable of correctly identifying frame order must satisfy Eq. (12) when $g(N)$ is set to $2N-1$. Conversely, if the inequality in Eq. (12) fails to hold for $g(N) = 2N-1$, it suggests that the supremum of the temporal attention logits is inadequate for the model to handle $2N-1$ distinct positions. Consequently, the model is unable to correctly identify frame order. Based on the above arguments, here, we perform our analysis taking the LaVie [50] video model as a case study, and use it to directly generate 128-frame (i.e., $N = 128$) videos. Notably, as shown in Eq. (12), to compute it, we need to determine the values of r and ϵ. Below, we then first discuss how we determine the values of r and ϵ for LaVie in our analysis.

Specifically, w.r.t. r, in LaVie, RoPE [42] is employed as the RPE mechanism, and only 32 dimensions (i.e., $k = 32$) of the query and key features are processed by RoPE

[2] A prerequisite for applying Lemma 1 is that $\mathcal{H}$ has a bounded pseudo-dimension $\dim_P(\mathcal{H}) = r$ (i.e., $1 \leq r < +\infty$). It will be shown that $\mathcal{H}$ satisfies this prerequisite later.

in each attention head. Additionally, as discussed earlier, for models using RoPE as the RPE mechanism, $r \leq 2k$ (i.e., $r \leq 64$). Notably, the right-hand side of Eq. (12) is negatively correlated with r. Hence, if $r = 64$ causes the inequality to fail, then the inequality does not hold for any $r < 64$. We then set $r = 64$ for the subsequent analysis here.

Meanwhile, to determine the value of ϵ, we first examine a scenario where these $2N-1$ positions are uniformly distributed to $2N-1$ groups (given $g(N) = 2N-1$). And the boundaries of these clusters (groups) are precisely situated in the middle of two adjacent positions. Consequently, the maximum intra-cluster (group) distance for the cluster that includes position p can be determined by calculating $d_f(p-0.5, p+0.5)$. According to the definition in Theorem 3, ϵ is greater than the maximum intra-cluster distance (measured by d_f) across all position clusters. With the maximum intra-distance of each group, we can determine the lower bound of ϵ, denoted as $\Omega(\epsilon_{\text{uni}})$. Notably, though $\Omega(\epsilon_{\text{uni}})$ is obtained based on the assumption that these $2N-1$ positions are uniformly clustered, for any non-uniformly distributed scenarios, there must exist at least one position cluster of larger size with a greater maximum intra-cluster distance. This means the true lower bound of ϵ is greater than $\Omega(\epsilon_{\text{uni}})$. Additionally, the right-hand side of Eq. (12) is positively correlated with ϵ. Hence, if setting ϵ to $\Omega(\epsilon_{\text{uni}})$ causes the Eq. (12) to fail, then the inequality does not hold for any ϵ. Thus, we here set $\epsilon = \Omega(\epsilon_{\text{uni}})$ for subsequent analysis.

After determining the values of r and ϵ, we extract query and key features from all the temporal attention heads to compute both the left and right sides of Eq. (12). We find that when generating 128-frame videos, only query and key features in 40% of attention heads satisfy the inequality in Eq. (12), and this percentage decreases to 34% when setting $N = 256$. This suggests that the supremum of the temporal attention logits is insufficient for the model to achieve $g(N) = 2N-1$. In other words, the existing video model can struggle in identifying correct frame order.

E.2. Detailed Proof of Theorem 2

Here, we provide a detailed proof of Theorem 2. For ease of reading, we restate Theorem 2 from the main paper below.

Theorem 4. *When generating a video with N frames, the information entropy H of temporal correlations over frames of the video sequence, is lower bounded by [18]:*

$$H\left(\frac{e^{a_i}}{\Sigma_{j=1}^{N} e^{a_j}} | 1 \leq i \leq N\right) \geq \ln N - 2B, \tag{18}$$

where $\{a_i\}_{i=1}^{N}$ are the attention logits with boundary $[-B, B]$.

Proof. The information entropy H of a discrete distribution P is given as $H(P) = -\Sigma_i p_i \ln p_i$. Hence, the information entropy of temporal correlation is computed as follows [18]:

$$\begin{aligned}
&H\left(\frac{e^{a_i}}{\Sigma_{j=1}^{N} e^{a_j}} | 1 \leq i \leq N\right) \\
&= -\sum_i \frac{e^{a_i}}{\sum_j e^{a_j}} \ln \frac{e^{a_i}}{\sum_j e^{a_j}} \\
&= -\sum_i \frac{e^{a_i}}{\sum_j e^{a_j}} \left(a_i - \ln \sum_j e^{a_j}\right) \\
&= -\sum_i \frac{e^{a_i}}{\sum_j e^{a_j}} a_i + \ln \sum_j e^{a_j} \\
&\geq -\max_i a_i + \ln(N e^{-B}) \\
&\geq \ln N - 2B
\end{aligned} \tag{19}$$

□

F. More Experiment Results

F.1. User Study

Following [35], we carried out a user study to assess our results based on human subjective judgment. In this study, participants were shown generated long videos using LaVie as the short video model from all methods (a total of 250 videos), with the examples presented in a random order to eliminate potential bias. Participants were then asked to score the generated videos on a scale of 1 to 5 according to three evaluation criteria: content consistency, video quality, and video-text alignment. The average scores for each method are reported in Tab. 12. As shown, our method received the highest ratings across all metrics.

Method	Content Consistency ↑	Video Quality	Video-Text Alignment
Direct	2.8	1.9	2.3
Sliding	1.8	3.1	2.5
FreeNoise [35]	3.3	3.6	3.5
FreeLong [32]	3.7	3.8	3.9
Ours	4.7	4.6	4.7

Table 12. Comparison based on user study.

F.2. Inference Time

In this section, we compare the inference times (time required for each denoising step) of our LongDiff with other training-free methods [32, 35] and two basic methods, **Direct** and **Sliding**, on the NVIDIA A6000 Ada

Method	Inference Time ↓
Direct	4.0s
Sliding	5.4s
FreeNoise [35]	5.4s
FreeLong [32]	4.7s
Ours	5.5s

Table 13. Comparison of inference time.

GPU. We apply all methods to LaVie and generate 128-frame videos for comparison. As shown in Tab. 13, Our LongDiff significantly improves the quality of long videos generated by the short video model and achieves state-of-the-art results with only a modest increase in inference time compared to the Direct method.

F.3. Evaluation on Video Models with Absolute Positional Encoding

Our LongDiff can also be adapted to video models utilizing absolute positional encoding mechanisms, such as sinusoidal position encoding [45]. This is achieved by performing the `GROUP` and `SHIFT` operations directly on the frame position rather than the relative positions among frames. Here, we take Animatediff [17], which uses sinusoidal position encoding for temporal attention computation, as a case study to evaluate the efficacy of our LongDiff. Specifically, we adapt Animatediff to generate 128-frame long videos with a resolution of 255×255. As shown in Tab. 14, compared to other training-free methods, LongDiff achieves the best performance across all the metrics.

Method	SC ↑	BC ↑	MS ↑	TF ↑	IQ ↑	OC ↑
Direct	92.25	94.35	97.42	96.75	49.27	20.01
Sliding	86.62	92.68	97.86	96.95	60.51	23.42
FreeNoise [35]	95.84	96.75	98.92	98.61	64.69	24.78
FreeLong [32]	95.11	95.86	97.72	98.10	60.23	23.51
Ours	97.54	97.39	98.98	98.70	65.14	25.11

Table 14. Quantitative comparisons of longer video generation (128 frames) on the Animatediff.

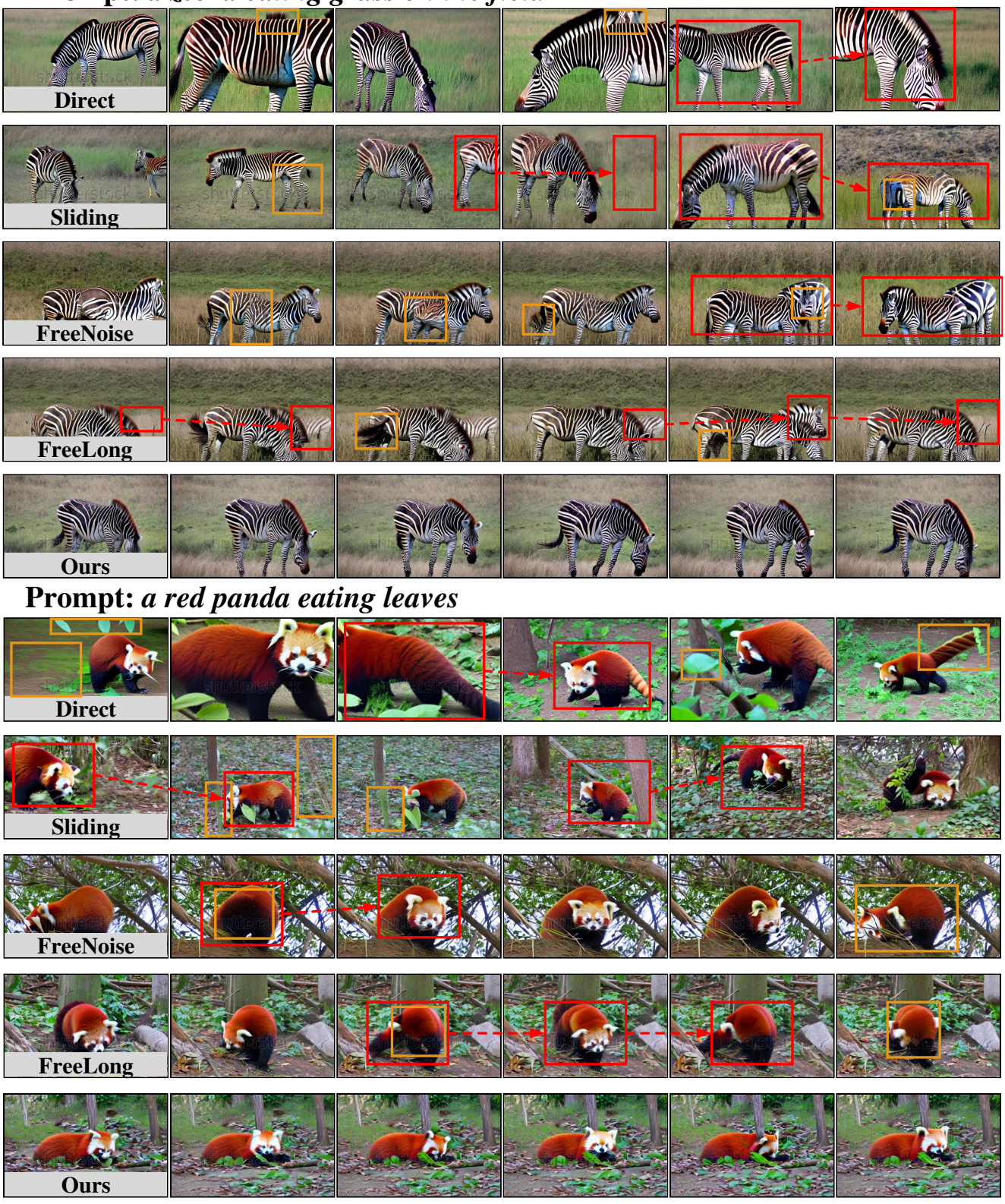

Figure 9. Qualitative comparisons of long video generation (128 frames) based on VideoCrafter[6]. Compared to our LongDiff, videos generated by other methods lack temporal consistency to some extent (e.g., zebras that suddenly appear and disappear in the videos generated from the first prompt; drastic motion changes of the red panda in the videos generated from the second prompt), and suffer from visual detail issues (e.g., blurred zebra bodies in the videos generated from the first prompt; fuzzy leaves and red pandas in the videos generated from the second prompt). We illustrate inferior temporal consistency and visual detail issues using red and orange boxes, respectively.

Figure 10. Qualitative comparisons of long video generation (128 frames) based on LaVie[50]. Compared to our LongDiff, videos generated by other methods lack temporal consistency to some extent (e.g., altered yacht structures in the videos generated from the first prompt; changing river surfaces and trees that suddenly appear and disappear in the videos generated from the second prompt), and suffer from visual detail issues (e.g., the fuzzy yacht in the videos generated from the first prompt; the blurred forest in the videos generated from the second prompt). We illustrate inferior temporal consistency and the visual detail issues using red and orange boxes, respectively.