

---

# TaoCache: Structure-Maintained Video Generation Acceleration

---

Zhentao Fan\*

Huawei Inc.

[zhentao.fan@mail.utoronto.ca](mailto:zhentao.fan@mail.utoronto.ca)

Zongzuo Wang

Huawei Inc.

Weiwei Zhang

Huawei Inc.

## Abstract

Existing cache-based acceleration methods for video diffusion models primarily skip early or mid denoising steps, which often leads to structural discrepancies relative to full-timestep generation and can hinder instruction following and character consistency. We present TaoCache, a training-free, plug-and-play caching strategy that, instead of residual-based caching, adopts a fixed-point perspective to predict the model’s noise output and is specifically effective in late denoising stages. By calibrating cosine similarities and norm ratios of consecutive noise deltas, TaoCache preserves high-resolution structure while enabling aggressive skipping. The approach is orthogonal to complementary accelerations such as Pyramid Attention Broadcast (PAB) and TeaCache, and it integrates seamlessly into DiT-based frameworks. Across Latte-1, OpenSora-Plan v110, and Wan2.1, TaoCache attains substantially higher visual quality (LPIPS, SSIM, PSNR) than prior caching methods under the same speedups.

## 1 Introduction

Diffusion models have recently shown remarkable capability in high-quality video generation, particularly with Diffusion Transformers (DiTs) [11, 13]. Despite state-of-the-art fidelity, their iterative denoising inherently incurs heavy computation: producing high-resolution or long-duration videos typically requires hundreds of sequential inference steps, which limits real-time and interactive use.

To reduce this cost without retraining, recent work explores *caching* strategies that reuse intermediate outputs across timesteps. ADACACHE [8] dynamically selects timesteps for recomputation via a feature-distance metric and motion regularization, allocating inference adaptively to video content. TEACACHE [9] leverages timestep embeddings to predict output variation and skips steps whose predicted residuals fall below calibrated thresholds. MAGCACHE [12] further simplifies the criterion using residual-norm magnitude, based on the observation that residual-norm ratios decrease through denoising. However, these approaches primarily skip early or mid stages; the resulting small discrepancies can compound and manifest later as degraded spatial structure and weakened high-frequency details—precisely where late-stage denoising is visually critical.

We address these limitations with **TaoCache**, a training-free, plug-and-play mechanism tailored to effective caching in the late denoising stages. Rather than relying on first-order residual approximations, TaoCache adopts a fixed-point view of the model’s noise prediction and explicitly models *second-order* noise deltas. By calibrating norm ratios and cosine similarities from consecutive late-stage steps, TaoCache predicts the model outputs for skipped timesteps while preserving global geometric consistency—even under aggressive skipping at high resolutions. The method introduces only a single lightweight calibration step and integrates seamlessly with DiT-based frameworks.

---

\*Corresponding author.

We evaluate TaoCache across diverse video generation stacks, including Latte-1 2B, OpenSora-Plan v110, and Wan2.1-1.3B. Under matched speedups, TaoCache delivers higher visual quality—measured by LPIPS, SSIM, and PSNR—than prior caching methods. Moreover, it complements orthogonal accelerations such as TeaCache and Pyramid Attention Broadcast (PAB), further improving end-to-end efficiency.

## 2 Related Work

### 2.1 Diffusion Models for Video Generation

Diffusion probabilistic models [5] have become the leading paradigm for high-quality video generation, surpassing GAN-based and autoregressive approaches in visual fidelity and training stability. Architectures have progressed from UNet-style designs (e.g., Make-A-Video [17]) to Diffusion Transformers (DiTs) [11, 13]. State-of-the-art systems such as Open-Sora-Plan and Wan2.1 produce compelling results at resolutions from various scales. However, the large number of inference steps required still hinders real-time and interactive applications, motivating substantial work on inference acceleration.

### 2.2 Acceleration Methods for Diffusion Inference

**Step reduction and advanced samplers.** Step-reduction methods include training-based approaches such as progressive distillation [15], sampling pseudo-knowledge distillation [1], and distribution matching [21]. Training-free ODE/SDE solvers (e.g., DPM-Solver [10] and UniPC [23]) also accelerate sampling substantially. In practice, training-based methods require model retraining, while training-free solvers can face stability or quality trade-offs at very low step counts, especially under guidance, limiting direct applicability to video DiTs.

**Spatial and temporal sparsity.** Computational sparsity reduces token counts or focuses compute on salient regions. Token merging [2] merges redundant spatial tokens; region-adaptive sampling [7] concentrates resources where motion is present. Pyramid Attention Broadcast (PAB) [24] hierarchically reuses multi-scale context, and Sparse VideoGen [20] leverages temporal attention sparsity. These techniques are orthogonal to timestep skipping and can complement caching.

### 2.3 Feature Caching Strategies

Feature-level caching reuses intermediate activations to avoid redundant computation without re-training. ADACACHE [8] selects recomputation steps using a feature-space distance with motion regularization, adapting compute to content. TEACACHE [9] predicts output variation from timestep embeddings and skips steps whose calibrated residual estimates fall below a threshold. MAGCACHE [12] uses a magnitude-based rule driven by the (near-)decay of residual-norm ratios to enable global skipping after minimal calibration. SKIP-DIT [3] inserts long-skip connections so deep features evolve slowly, enabling extensive reuse at the cost of fine-tuning. DUCA [26] interleaves aggressive layer caching with periodic token-wise recomputation to heal accumulated error, though additional forward passes can dilute net speedup.

### 2.4 Content-Adaptive Computation

Several methods tailor computation to content characteristics, e.g., spatial activity or motion. Region-based strategies refine active regions [7], and ADACACHE adjusts allocation using optical flow [8]. TaoCache is complementary: it operates primarily along the timestep dimension and integrates with existing spatially adaptive techniques.

### 2.5 Positioning TaoCache

TaoCache frames feature caching as a fixed-point estimation problem over *second-order* noise deltas, targeting the visually critical late denoising stages. Unlike first-order approximations, it preserves geometric and structural fidelity at high resolutions, requires no model modification, and needs only lightweight calibration. Experiments show that TaoCache outperforms prior caching baselines under matched speedups and pairs well with orthogonal accelerations in later denoising.

### 3 Methodology

#### 3.1 Observation

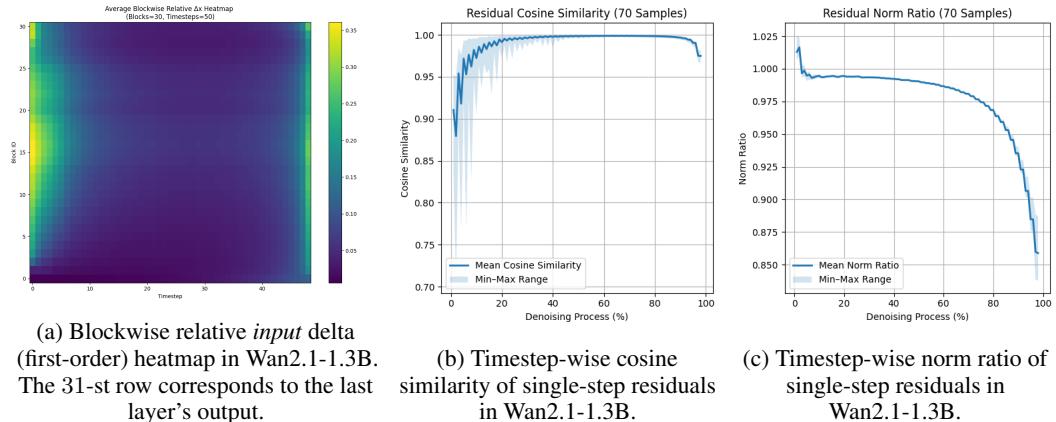


Figure 1: Layerwise/stepwise statistics that inform caching policy. Mid-range steps show smaller first-order changes (a), while early/late steps exhibit higher variance in residual cosine similarity (b) and norm ratios (c), making fixed thresholds less reliable across prompts.

Different choices of *what* to cache naturally induce different caching policies. As shown in Fig. 1 (a), the *blockwise relative input/output delta* is smaller in the mid-range denoising steps, which explains why many feature-caching methods preferentially skip there. For approaches such as TEACACHE and MAGCACHE that rely on single-step residual signals—distance between the input and output of a one-step forward DiT—their cosine similarity and magnitude (norm) ratios display larger variance in the early and late stages (Fig. 1 (b,c)). Since these metrics are *a priori* unknown for a specific prompt, static or globally calibrated thresholds can lead to unstable skip allocations across timesteps.

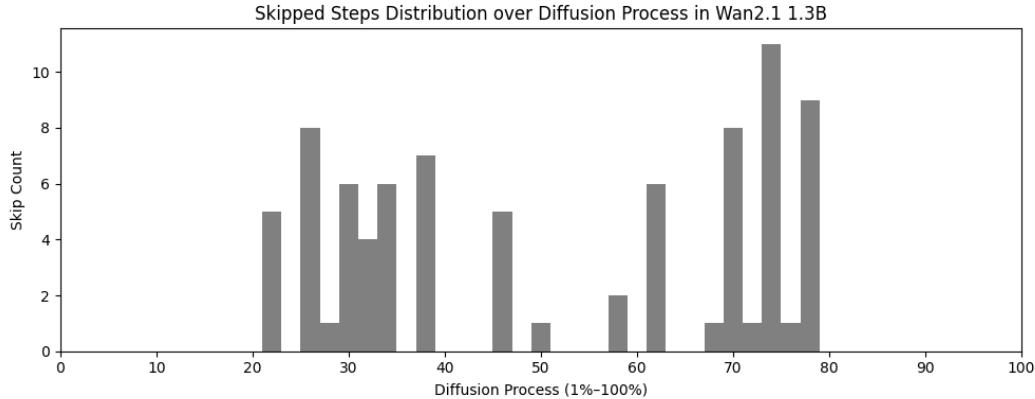


Figure 2: TEACACHE skip counts across denoising steps under total skip budgets of 2%, 4%, 6%, ..., 24%. Early/late stages are harder to skip consistently due to the higher variance of first-order residual signals.

Moreover, early steps are crucial for latent denoising [12]; errors from skipping there can *propagate* and manifest as structural drift. While later steps may recover low-frequency fidelity, the damage to **instruction following** and **character consistency** is harder to repair.

To seek a signal that is stable at *late timesteps* yet light-weight to compute and store, we examine *output-noise deltas*:

$$\Delta\epsilon_t := \epsilon_\theta(\mathbf{x}_t, t) - \epsilon_\theta(\mathbf{x}_{t+1}, t+1),$$

and measure their cosine similarity and norm ratio across consecutive late-stage steps:

$$\cos_{\text{sim}}(t) = \frac{\langle \Delta\epsilon_t, \Delta\epsilon_{t+1} \rangle}{\|\Delta\epsilon_t\| \|\Delta\epsilon_{t+1}\|}, \quad \text{norm\_ratio}(t) = \frac{\|\Delta\epsilon_{t+1}\|}{\|\Delta\epsilon_t\|}.$$

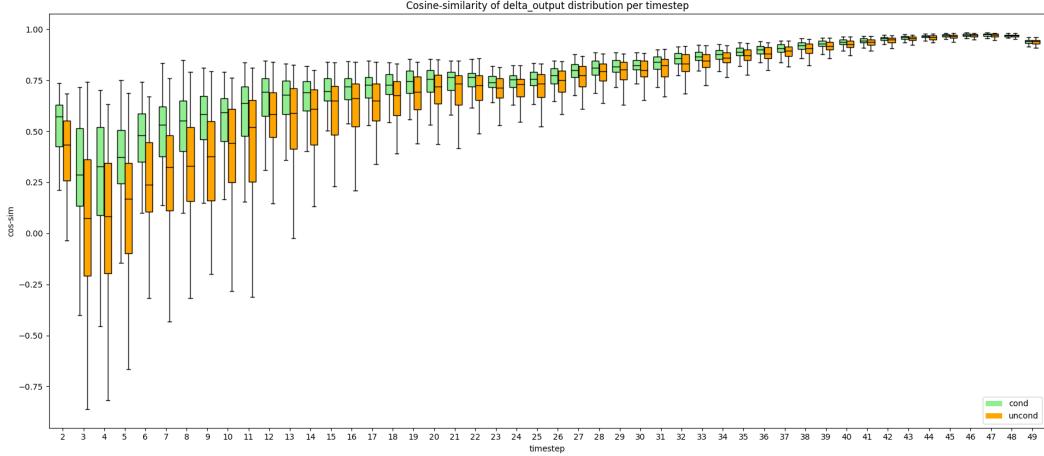


Figure 3: Cosine similarity of *output-noise deltas* in Wan2.1-1.3B (“*cond*” denotes the positive-conditioning stream; “*uncond*” the negative). Late-stage values converge and remain highly correlated.

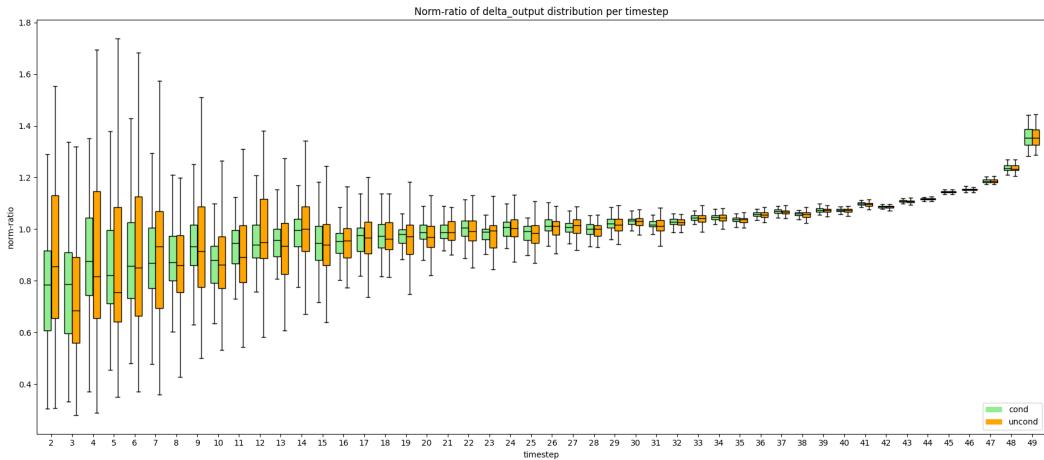


Figure 4: Norm ratio of *output-noise deltas* in Wan2.1-1.3B (cond/uncond as above). Ratios stabilize in late denoising, yielding a predictable scale relationship.

Empirically (Fig. 3–4), both  $\cos_{\text{sim}}(t)$  and  $\text{norm\_ratio}(t)$  *converge* during late denoising for both conditional and unconditional streams. This property provides a simple, robust late-stage cache signal that does not require heavy, layer-wise state. Leveraging it enables a caching policy that maintains global structure while remaining lightweight and training-free—motivating the fixed-point, second-order design of **TaoCache**.

### 3.2 Design

**Fixed-point view of output-related caching.** At late denoising steps, the DiT forward map is close to an identity transform over small neighborhoods of the latent trajectory. Let  $f$  denote a generic one-step map; a classical fixed-point intuition is

$$y + \delta y = f(y), \quad f(y + \delta y) \approx (y + \delta y) + \delta y,$$

i.e., consecutive increments are small and strongly correlated. For video DiTs, we apply this view to the *predicted noise*.

**Notation.** Let  $\hat{\epsilon}_t := \epsilon_\theta(\mathbf{x}_t, t)$  be the model’s noise prediction at timestep  $t$  for latent  $\mathbf{x}_t$ , and SchedulerStep the sampling update (Euler, DPM-Solver, UniPC, etc.). We define the *output-noise delta*

$$\Delta_t := \hat{\epsilon}_t - \hat{\epsilon}_{t+1}.$$

Empirically (Fig. 3–4), in late stages the *direction* of  $\Delta_t$  changes slowly and the *scale* evolves smoothly. Hence we model a second-order relation

$$\Delta_t \approx r_t \Delta_{t+1}, \quad r_t > 0,$$

where  $r_t$  is a scalar *norm ratio* and the direction agreement is quantified by the cosine similarity  $c_t \approx 1$  between  $\Delta_t$  and  $\Delta_{t+1}$ . This “scalar-times-previous-delta” approximation is the core of TAO CACHE.

**Layer/solver abstraction.** Writing the DiT as a composition  $F = F_L \circ \dots \circ F_1$  (self-/cross-attn, MLP, norms) and the sampler as  $G_t(\mathbf{x}_t, \hat{\epsilon}_t)$ , a single step is

$$\hat{\epsilon}_t = \epsilon_\theta(\mathbf{x}_t, t), \quad \mathbf{x}_{t-1} = G_t(\mathbf{x}_t, \hat{\epsilon}_t).$$

Because  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$  are close in late steps,  $\hat{\epsilon}_t$  and  $\hat{\epsilon}_{t+1}$ —and hence  $\Delta_t$  and  $\Delta_{t+1}$ —exhibit high correlation, enabling the above second-order approximation without touching internal layer states.

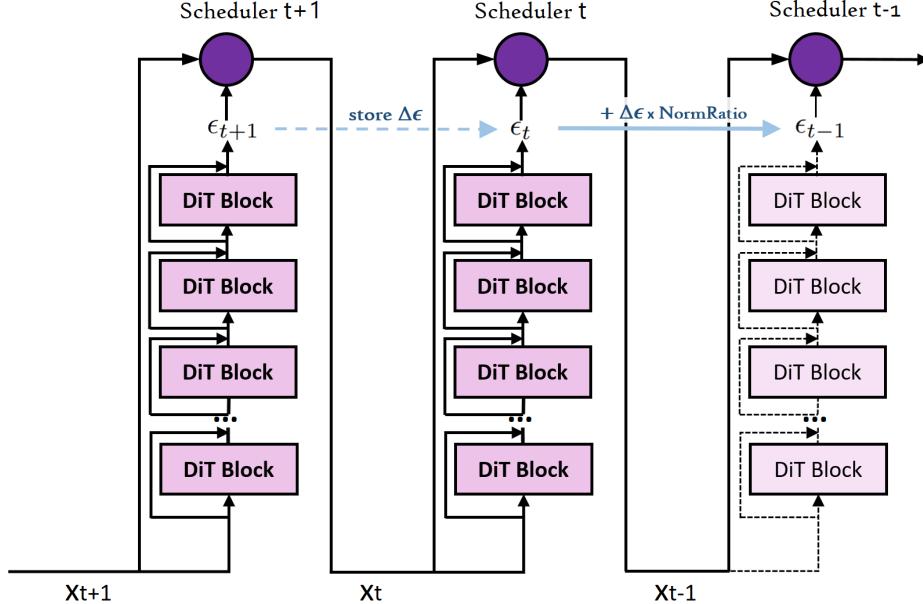


Figure 5: **TaoCache overview.** Instead of predicting  $\hat{\epsilon}_t$  directly, we predict the change  $\Delta_t$  from  $\Delta_{t+1}$  using calibrated late-stage statistics (norm ratio  $r_t$  and cosine  $c_t$ ), then recover  $\hat{\epsilon}_t = \hat{\epsilon}_{t+1} + \Delta_t$  and call the scheduler.

**One-time warmup calibration.** For a new checkpoint, we run a small set of prompts  $\mathcal{P}$  once without acceleration. From each trajectory we record, for every valid  $t$ ,

$$c_t^{(p)} = \frac{\langle \Delta_t^{(p)}, \Delta_{t+1}^{(p)} \rangle}{\|\Delta_t^{(p)}\| \|\Delta_{t+1}^{(p)}\|} \quad r_t^{(p)} = \frac{\|\Delta_t^{(p)}\|}{\|\Delta_{t+1}^{(p)}\|}.$$

We then build lookup tables with both *mean* and *dispersion*:

$$C_{\cos}[t] = \text{mean}_p c_t^{(p)}, \quad S_{\cos}[t] = \text{std}_p c_t^{(p)}; \quad C_{\text{ratio}}[t] = \text{mean}_p r_t^{(p)}, \quad S_{\text{ratio}}[t] = \text{std}_p r_t^{(p)}.$$

**Deviation-aware window selection.** Given a target skip budget  $N_{\text{skip}}$ , we choose a *contiguous late-stage window*  $W$  of length  $N_{\text{skip}}$  either **manually** or by maximizing a variance-penalized score for automated skipping:

$$W^* = \arg \max_W \left( \underbrace{\text{mean}_{t \in W} C_{\cos}[t]}_{\text{directional agreement}} - \lambda \underbrace{\text{mean}_{t \in W} S_{\cos}[t]}_{\text{directional variability}} - \gamma \underbrace{\text{mean}_{t \in W} S_{\text{ratio}}[t]}_{\text{scale variability}} \right),$$

subject to  $W$  lying in the late denoising regime (enforced by an upper-bound  $t$  or by requiring  $C_{\cos}[t] \geq \tau_{\cos}$ ). This “max-cos with deviation penalty” makes the skip region both *predictable* and *stable* across prompts.

**Delta prediction during skipping.** When  $t \in W^*$  (skip), we set

$$\tilde{\Delta}_t = C_{\text{ratio}}[t] \cdot \tilde{\Delta}_{t+1}, \quad \tilde{\hat{\epsilon}}_t = \tilde{\hat{\epsilon}}_{t+1} + \tilde{\Delta}_t,$$

and fall back to a *refresh* (full model call) every  $K$  skips (to bound drift,  $K$  can be large). Finally, we advance the sampler via  $\mathbf{x}_{t-1} = \text{SchedulerStep}(\mathbf{x}_t, \tilde{\hat{\epsilon}}_t, t)$ .

---

**Algorithm 1** TaoCache: Delta–Noise Calibration and Deviation-Aware Inference for DiT

---

```

1: function CALIBRATE( $\mathcal{P}, \epsilon_\theta$ )
2:   Initialize:  $c[t, p] \leftarrow 0, r[t, p] \leftarrow 0$  for  $t = T-2, \dots, 1$ ; ▷ means over prompts
3:   for all  $p \in \mathcal{P}$  do
4:      $\mathbf{x}_T \leftarrow \text{INITNOISE}(p)$ 
5:     for  $t = T, T-1, \dots, 1$  do
6:        $\hat{\epsilon}_t \leftarrow \epsilon_\theta(\mathbf{x}_t, t)$ 
7:        $\mathbf{x}_{t-1} \leftarrow \text{SCHEDULERSTEP}(\mathbf{x}_t, \hat{\epsilon}_t, t)$ 
8:       if  $t \leq T-1$  then
9:          $\Delta_t \leftarrow \hat{\epsilon}_t - \hat{\epsilon}_{t+1}$  ▷ output-noise delta
10:      end if
11:      if  $t \leq T-2$  then
12:         $c[t, p] = \text{COSINESIMILARITY}(\Delta_t, \Delta_{t+1})$ 
13:         $r[t, p] = \|\Delta_t\| / \|\Delta_{t+1}\|$ 
14:      end if
15:    end for
16:  end for
17:  Normalization:  $C_{\cos}[t] \leftarrow \text{MEAN}(c[t]), C_{\text{ratio}}[t] \leftarrow \text{MEAN}(r[t])$  for each  $t$ 
18:  Deviation:  $S_{\cos}[t] \leftarrow \text{STD}(c[t]), S_{\text{ratio}}[t] \leftarrow \text{STD}(r[t])$  for each  $t$ 
19:  return  $C_{\cos}, C_{\text{ratio}}, S_{\cos}, S_{\text{ratio}}$ 
20: end function
21:
22: function TAOCACHEFORWARD( $\mathbf{x}_T, \epsilon_\theta, C_{\cos}, C_{\text{ratio}}, S_{\cos}, S_{\text{ratio}}, N_{\text{skip}}, K$ )
23:    $\mathcal{T}_{\text{skip}} \leftarrow \text{MAXCOSSLIDINGWINDOW}(C_{\cos}, N_{\text{skip}}, S_{\cos}, S_{\text{ratio}}, K)$  or Manually
24:   for  $t = T, T-1, \dots, 1$  do
25:     if ( $t \notin \mathcal{T}_{\text{skip}}$ ) then
26:        $\hat{\epsilon}_t \leftarrow \epsilon_\theta(\mathbf{x}_t, t)$ 
27:       if  $t \leq T-1$  then
28:          $\Delta_t \leftarrow \hat{\epsilon}_t - \hat{\epsilon}_{t+1}$ 
29:       end if
30:     else
31:        $\Delta_t \leftarrow C_{\text{ratio}}[t] \cdot \Delta_{t+1}$ 
32:        $\hat{\epsilon}_t \leftarrow \hat{\epsilon}_{t+1} + \Delta_t$ 
33:     end if
34:      $\mathbf{x}_{t-1} \leftarrow \text{SCHEDULERSTEP}(\mathbf{x}_t, \hat{\epsilon}_t, t)$ 
35:   end for
36:   return  $\mathbf{x}_0$ 
37: end function

```

---

### 3.3 Orthogonality

**Scope.** TAO CACHE operates via output–noise deltas in late denoising stages, which is largely disjoint from spatial–temporal sparsity methods and prior feature-caching that occurs in mid inference steps. It is training-free and applies on top of a given sampler ( $G_t$ ), making it compatible with both cache-based and non-cache accelerations.

As concrete illustrations, we demonstrate one composition along the *timestep* axis (Tea+Tao Cache) and one along the *spatial-temporal* axis (PAB+TaoCache). We did not empirically study other combinations like Delta-DiT or AdaCache; they should be compatible in principle, which we leave for future work.

#### 3.3.1 Orthogonal: Tea + Tao Cache

Integrating TAO CACHE with TEACACHE is straightforward: allocate *late* steps to TAO CACHE and apply TEACACHE to the remaining (earlier/mid) steps, with a short *refresh guard band* between the two ranges to prevent error carryover.

---

#### Algorithm 2 Hybrid TEA+TAO Inference

---

```

1: function HYBRIDFORWARD( $\mathbf{x}_T, C_{\cos}, N_{\text{TaoSkip}}, \text{RefreshSteps}$ )
2:    $\mathcal{T}_{\text{Tao}} \leftarrow \text{MAXCOSSLIDINGWINDOW}(C_{\cos}, N_{\text{TaoSkip}})$             $\triangleright$  contiguous late-stage window
3:    $t_{\text{brk}} \leftarrow \max(\mathcal{T}_{\text{Tao}}) + \text{RefreshSteps}$             $\triangleright$  2–3 steps recommended
4:   for  $t = T, T-1, \dots, t_{\text{brk}}$  do
5:     Apply TEACACHE step
6:   end for
7:   for  $t = t_{\text{brk}}, t_{\text{brk}}-1, \dots, 1$  do
8:     Apply TAO CACHE step
9:   end for
10:  return  $\mathbf{x}_0$ 
11: end function

```

---

### 3.4 Orthogonal: PAB + Tao Cache

PAB reduces FLOPs by reusing multi-scale attention context within each model call, while TAO CACHE reduces the *number* of model calls via timestep skipping. Since they act on different axes (intra-step vs. inter-step), we simply *enable PAB inside each forward* and let TAO CACHE choose the skip window. No change to either mechanism is required, aside from ensuring that the PAB state (if any) is reinitialized on refresh steps.

## 4 Experiments & Results

### 4.1 Experimental Settings

**Base Models and Compared Methods.** To validate the generality of TaoCache, we apply it to three representative DiT-based video generators: Latte-1 2B [11], OpenSora-Plan v110 [14], and Wan2.1-1.3B [4]. We compare against recent training-free caching and acceleration techniques: TeaCache [9] and MagCache [12]. 70 prompts that used for evaluation are sampled from CompBench [18]. Unless otherwise stated, we follow each model’s default inference resolution and hold the sampler and guidance settings fixed across methods.

**Evaluation Metrics.** We measure inference efficiency via the number of inference steps skipped (which is also relative FLOPs reduction). For visual quality, we report: LPIPS (Learned Perceptual Image Patch Similarity; lower is better) [22], which uses deep network activations to approximate human perceptual judgments; SSIM (Structural Similarity Index; higher is better) [19], which quantifies image quality based on luminance, contrast, and structural agreement; PSNR (Peak Signal-to-Noise Ratio; higher is better) [6], which measures pixel-wise fidelity in decibels.

## 4.2 Latte-1 2B

Latte-1 2B is a DiT-based video generator. We compare TAO CACHE with TEACACHE under *matched skip budgets*, keeping the sampler, guidance, and resolution identical to the baseline. The percentage in parentheses denotes the fraction of steps skipped, and speedup can be calculated accordingly.

Table 1: Video Quality Metrics and End-to-End Speedup for Latte-1 2B.

Method	% Speedup (with % inference steps skipped)	LPIPS ↓	SSIM ↑	PSNR ↑ (dB)
Original	–	–	–	–
TeaCache	21.47% (17.68%)	0.1674	0.7507	21.57
TaoCache	21.95% (18.00%)	<b>0.0598</b>	<b>0.8838</b>	<b>29.36</b>
TeaCache	16.27% (14.00%)	0.1541	0.7654	22.21
TaoCache	16.27% (14.00%)	<b>0.0383</b>	<b>0.9194</b>	<b>32.18</b>
TeaCache	11.11% (10.00%)	0.1338	0.7891	23.14
TaoCache	11.11% (10.00%)	<b>0.0184</b>	<b>0.9524</b>	<b>36.07</b>
TeaCache	6.80% (6.37%)	0.1017	0.8334	25.46
TaoCache	6.38% (6.00%)	<b>0.0127</b>	<b>0.9634</b>	<b>38.03</b>

**Full Timesteps:** 50

**Frames:** 16

**Resolution:**  $512 \times 512$

**Prompts:** 70 in total (7 domains, 10 prompts each)

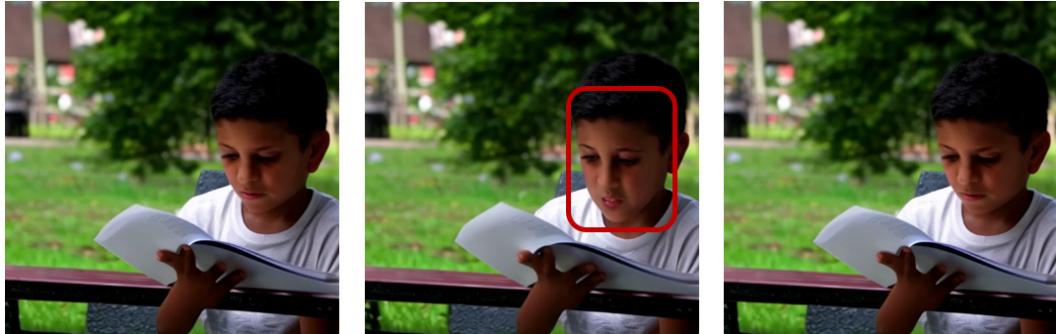


Figure 6: Baseline vs TeaCache vs TaoCache (ours)  
Latte-1 2B (16 frames), 6% Inference Steps Skipped



Figure 7: Baseline vs TeaCache vs TaoCache (ours)  
Latte-1 2B (16 frames), 10% Inference Steps Skipped

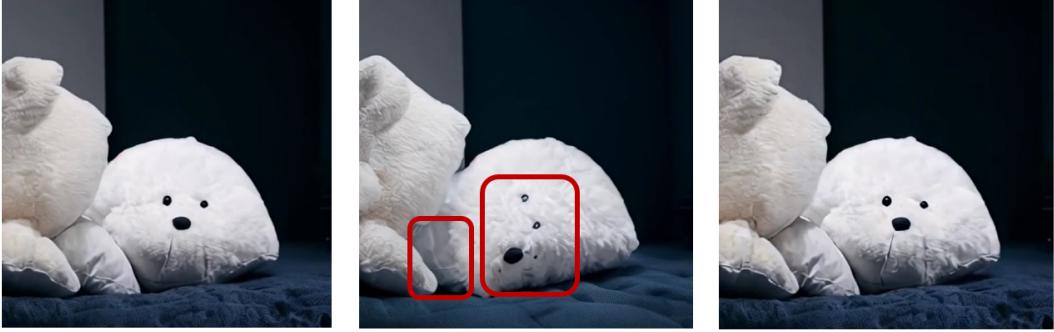


Figure 8: Baseline vs TeaCache vs TaoCache (ours)  
Latte-1 2B (16 frames), 18% Inference Steps Skipped

These are three sample comparisons across different scales (6%, 10%, 18%) of inference step skips on Latte-1 2B. From the results, those undesired caching behaviours like disfigured faces and unexpected limbs are avoided by TaoCache. This is significant for high-standard video generations of instruction following for controller and facial consistency for story alignment.

We should emphasize that not all TeaCache’s generations would have these problems, but they are the issues that TeaCache cannot inherently avoid.

### 4.3 OpenSora-Plan v110

OpenSora-Plan is an open-source DiT-based video generation framework that targets high-fidelity and efficient synthesis at moderate resolutions. Similar to the Latte-1 case, we evaluate TAO CACHE against TEACACHE under matched skip budgets, keeping the sampling algorithm, guidance scale, and resolution fixed. TAO CACHE uses its late-stage, deviation-aware skip placement to improve stability and quality, particularly for temporally consistent content and fine details.

Table 2: Video Quality Metrics and End-to-End Speedup for OpenSora-Plan v110.

Method	% Speedup (with % inference steps skipped)	LPIPS ↓	SSIM ↑	PSNR ↑ (dB)
Original	–	–	–	–
TeaCache	33.68% (25.2%)	0.0780	0.8531	26.27
TaoCache	33.33% (25.0%)	<b>0.0472</b>	<b>0.8873</b>	<b>29.71</b>
TeaCache	26.93% (21.2%)	0.0774	0.8538	26.32
TaoCache	26.58% (21.0%)	<b>0.0330</b>	<b>0.9093</b>	<b>31.58</b>
TeaCache	19.47% (16.3%)	0.0586	0.8842	28.40
TaoCache	19.04% (16.0%)	<b>0.0294</b>	<b>0.9151</b>	<b>32.24</b>
TeaCache	8.69% (8.0%)	0.0512	0.8968	29.13
TaoCache	9.89% (9.0%)	<b>0.0164</b>	<b>0.9485</b>	<b>35.40</b>

**Full Timesteps:** 100 (9 scheduler order + 91)

**Frames:** 65

**Resolution:**  $512 \times 512$

**Prompts:** 70 in total (7 domains, 10 prompts each)



Figure 9: Baseline vs TeaCache (8%) vs TaoCache (ours)  
OpenSora-Plan v110 (65 frames), 9% Inference Steps Skipped



Figure 10: Baseline vs TeaCache vs TaoCache (ours)  
OpenSora-Plan v110 (65 frames), 16% Inference Steps Skipped



Figure 11: Baseline vs TeaCache vs TaoCache (ours)  
OpenSora-Plan v110 (65 frames), 21% Inference Steps Skipped

For OpenSora-Plan, TaoCache has the same expected behaviour as we discussed in Latte-1 2B’s experiments. For TeaCache of 8.0% step skips, these 8 skips (since 100 timesteps in total) come from the very early stages of warmup timesteps and cannot easily be further reduced to 4 skips by tuning the *rel\_l1\_thresh* parameter.

#### 4.4 Wan2.1 1.3B

Wan2.1-1.3B is a state-of-the-art DiT-based video generation model designed for high-fidelity, controllable synthesis. In addition to TEACACHE, we also compare against MAGCACHE [12] on this model. All methods use the model’s default inference settings (sampler, guidance, resolution) to

ensure comparability. Skip budgets are matched across methods; speedup is reported with parentheses indicating the proportion of denoising steps skipped. TAO CACHE applies its calibrated, deviation-aware skip placement in late-stage timesteps to maximize stability and visual quality.

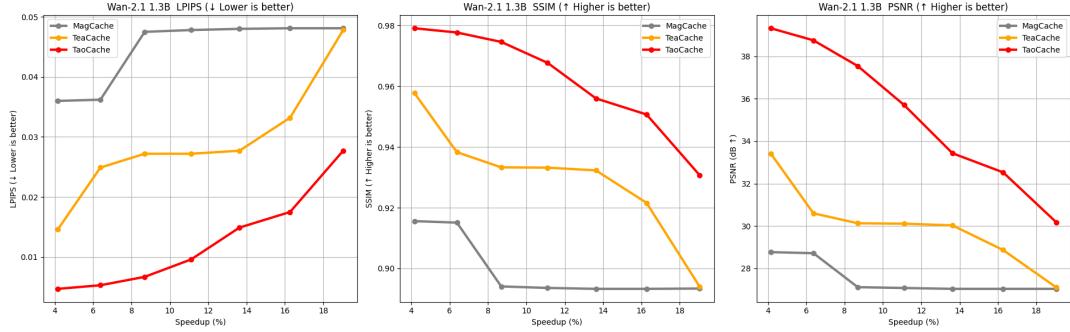


Figure 12: Comparisons between TaoCache, TeaCache, and MagCache on Wan2.1-1.3B.

**Full Timesteps:** 50

**Frames:** 33 (2 s)

**Resolution:** 832 × 480 (480 p)

**Prompts:** 70 in total (7 domains, 10 prompts each)



Figure 13: Baseline vs TeaCache vs TaoCache (ours)  
Wan2.1-1.3B (33 frames), 8% Inference Steps Skipped



Figure 14: Baseline vs TeaCache vs TaoCache (ours)  
Wan2.1-1.3B (33 frames), 8% Inference Steps Skipped

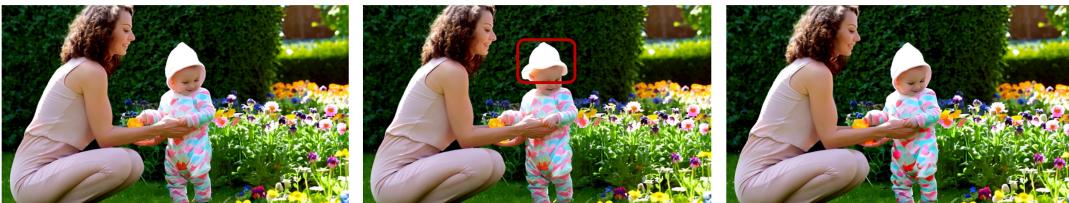


Figure 15: Baseline vs TeaCache vs TaoCache (ours)  
Wan2.1-1.3B (33 frames), 16% Inference Steps Skipped



Figure 16: Baseline vs TeaCache vs TaoCache (ours)  
Wan2.1-1.3B (33 frames), 16% Inference Steps Skipped

These are the sample result from TaoCache and TeaCache on Wan2.1. Figure 12 shows that TaoCache surpasses the other two methods under 20% speedup. However, from Figure 15 we can inspect that the TaoCache strategy also have certain flaws. The color of the baby suit is denser in TaoCache. This can be explained as TaoCache may overestimate the denoising momentum for a certain prompt in a long skip of late denoising stages.

#### 4.5 Ablation for Caching Feature

To compare the residual-based caching and output-delta caching in the later denoising process, we test TeaCache’s residual mechanism on the same timesteps being skipped by TaoCache. This is indicated as TaoSkip + TeaResidual rows in the following table.

From the following table, we see that, on the same timesteps skipping in late denoising procedures, the residual skipping strategy is not as good as the output delta skipping.

Method	% Speedup (with % inference steps skipped)	LPIPS ↓	SSIM ↑	PSNR ↑ (dB)
Original	–	–	–	–
TeaCache	4.16% (4.00%)	0.0146	0.9578	33.41
TaoSkip+TeaResidual	4.16% (4.00%)	0.0088	0.9704	36.63
TaoCache	4.16% (4.00%)	<b>0.0047</b>	<b>0.9791</b>	<b>39.32</b>
TeaCache	8.69% (8.00%)	0.0272	0.9333	30.13
TaoSkip+TeaResidual	8.69% (8.00%)	0.0126	0.9633	35.19
TaoCache	8.69% (8.00%)	<b>0.0067</b>	<b>0.9746</b>	<b>37.54</b>
TeaCache	13.63% (12.00%)	0.0277	0.9323	30.03
TaoSkip+TeaResidual	13.63% (12.00%)	0.0215	0.9465	33.02
TaoCache	13.63% (12.00%)	<b>0.0149</b>	<b>0.9560</b>	<b>33.43</b>
TeaCache	19.04% (16.00%)	0.0478	0.8940	27.11
TaoSkip+TeaResidual	19.04% (16.00%)	0.0333	0.9200	<b>30.87</b>
TaoCache	19.04% (16.00%)	<b>0.0277</b>	<b>0.9307</b>	30.18

Table 3: Video Quality Metrics and End-to-End Speedup for Wan2.1-1.3B.

**Full Timesteps:** 50

**Frames:** 33 (2s)

**Resolution:**  $832 \times 480$  (480p)

**Prompts:** 70 in total (7 domains, 10 prompts each)

#### 4.6 Orthogonality : Tea + Tao Cache

Mentioned in Algorithm 2, the Hybrid Cache firstly generates under TeaCache’s range and then ends with TaoCache’s inference caching. The experiments in Fig 17 are conducted between pure TeaCache and Hybrid Caching with the TeaCache followed by 7 steps of timestep skips from TaoCache.

The results show that with the same percentages of acceleration, the Hybrid Cache can surpass the pure TeaCache method in terms of video quality.

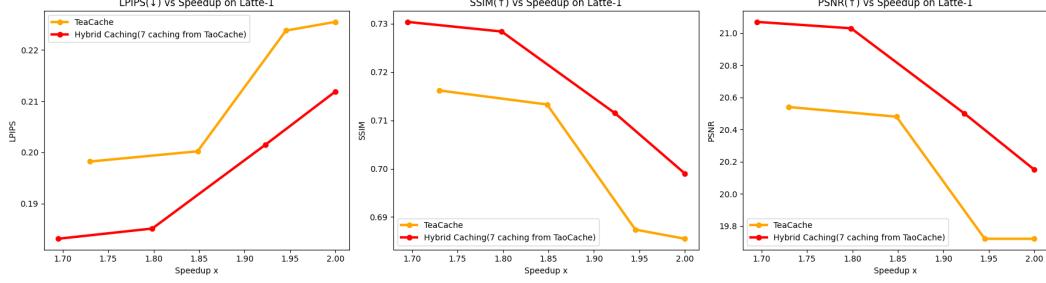


Figure 17: Hybrid Caching(TeaCache + TaoCache) on Latte-1 2B

**Full Timesteps:** 50

**Frames:** 16

**Resolution:**  $512 \times 512$

**Prompts:** 70 in total (7 domains, 10 prompts each)

#### 4.7 Orthogonal : PAB + TaoCache

The experiment for PAB with TaoCache is conducted directly based on PAB<sub>224</sub>, PAB<sub>236</sub>, PAB<sub>347</sub>, PAB<sub>469</sub> settings, where PAB <sub>$\alpha\beta\gamma$</sub>  represents the broadcast ranges of spatial ( $\alpha$ ), temporal ( $\beta$ ), and cross ( $\gamma$ ) attentions [11].

The following graphs show that with comparable video qualities, PAB + TaoCache can significantly speed up the inference.

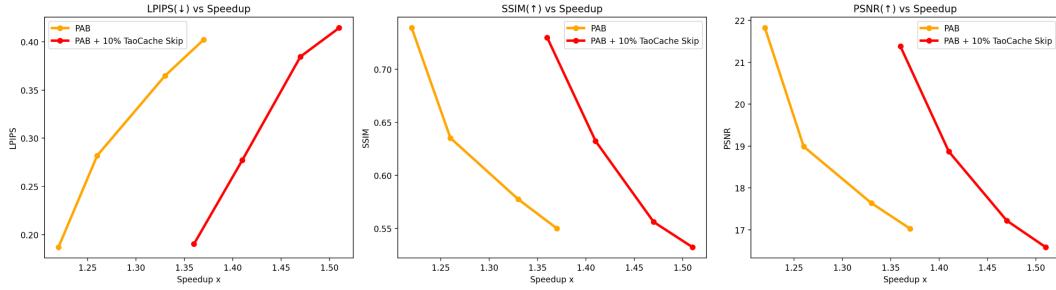


Figure 18: PAB + TaoCache on Latte-1 2B.

**Full Timesteps:** 50

**Frames:** 16

**Resolution:**  $512 \times 512$

**Prompts:** 70 in total (7 domains, 10 prompts each)

## 5 Limitations

There are three main constraints for TaoCache. Firstly, the inference steps range that TaoCache can be applied is narrower than TeaCache and MagCache, as it only applies in late stages, but its orthogonality compensates for this. Secondly, the calibration is as heavy as TeaCache. To inspect the deviation of output delta's cosine similarities and norm ratio, for a new model, 20 prompts from various domains are recommended. Lastly, for a model trained under uniformly distributed timesteps [5, 13], late-stage behavior is relatively predictable, making second-order delta prediction effective. In contrast, for models trained with *log-normal* timestep sampling [16, 25], more structural updates concentrate at late steps; the deltas become less stationary, and TAOCACHE may be less effective.

## 6 Conclusion

In this work, we introduced TaoCache, a novel training-free acceleration method for diffusion-based video generation that prioritizes the preservation of structural integrity. We identified that existing caching methods, which primarily skip early or middle denoising steps, can lead to discrepancies in the final video’s composition and character consistency. To address this, TaoCache introduces a caching strategy based on a fixed-point approximation of the second-order noise delta. By leveraging the observation that this delta becomes highly stable and predictable during the late stages of denoising, our method effectively skips computations where it matters least for structure and most for fine-tuning details.

Our extensive experiments on state-of-the-art models like Latte, OpenSora-Plan, and Wan2.1 demonstrate that TaoCache significantly outperforms prior caching techniques such as TeaCache and MagCache, achieving superior visual quality across LPIPS, SSIM, and PSNR metrics for the same level of speedup. We also showed that TaoCache is orthogonal to other acceleration methods and can be successfully hybridized to yield even greater efficiency. While its effectiveness is currently focused on models trained with uniform timestep distributions, TaoCache presents a robust and principled approach to accelerating video generation while maintaining high fidelity and structural coherence, paving the way for more practical applications of large-scale video diffusion models.

## A Appendix

### A.1 Model Calibrations

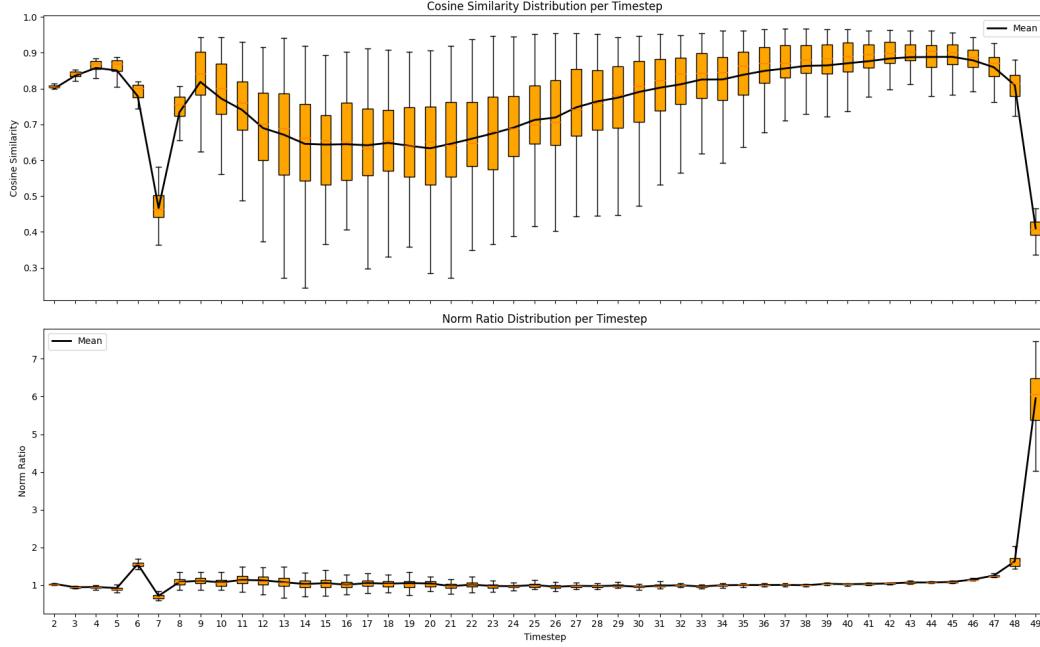


Figure 19: Latte-1 2B (16 frames)

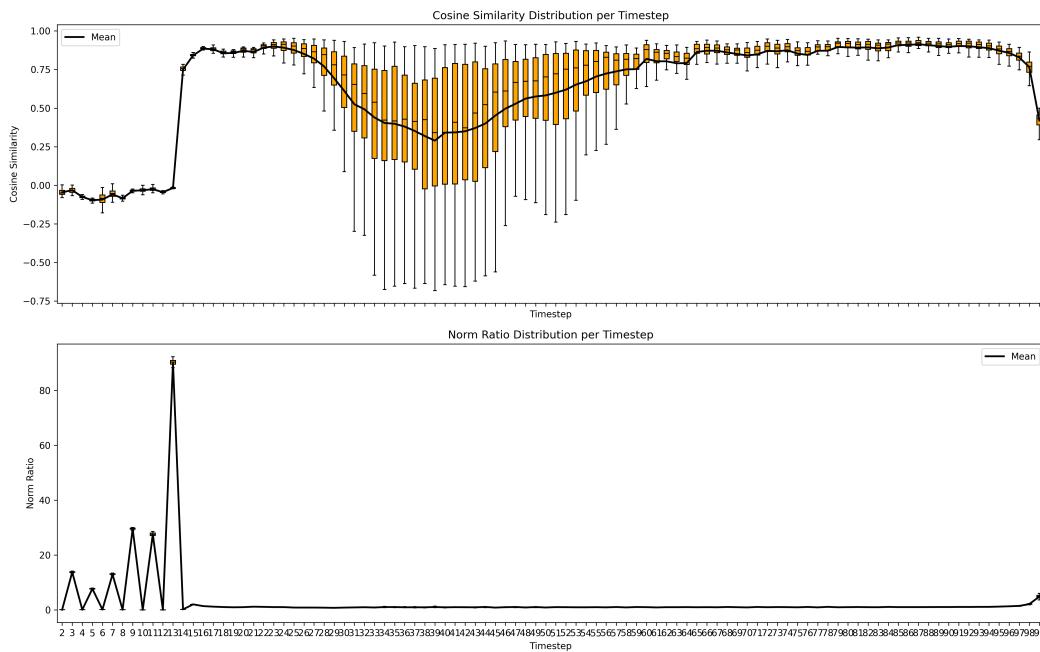


Figure 20: Opensora-Plan V110 (65 frames)

## References

- [1] Bao, C., Tian, Y., Yan, W., Liu, B. & Zhang, H. (2023). SPKD: Sampling Pseudo-Knowledge Distillation for Fast Image Synthesis. *arXiv preprint arXiv:2308.18933*.
- [2] Bolya, D. & Hoffman, J. (2023). Token Merging for Fast Stable Diffusion. In *Proceedings of the CVPR 2023 Workshops*.
- [3] Chen, G. et al. (2025). Towards Stabilized and Efficient Diffusion Transformers through Long-Skip-Connections with Spectral Constraints. *arXiv preprint arXiv:2411.17616*.
- [4] Fan, Z. et al. (2025). Wan 2.1: Scaling Diffusion Transformers for High-Resolution Video Generation. *arXiv preprint arXiv:2503.20314*.
- [5] Ho, J., Jain, A. & Abbeel, P. (2020). Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems 33*, pp. 6840–6851.
- [6] Hore, A., & Ziou, D. (2010). Image quality metrics: PSNR vs. SSIM. In *Proceedings of the 2010 International Conference on Pattern Recognition*, pp. 2366–2369.
- [7] Jeong, W. et al. (2025). Upsample What Matters: Region-Adaptive Latent Sampling for Accelerated Diffusion Transformers. *arXiv preprint arXiv:2507.08422*.
- [8] Kahatapitiya, K. et al. (2024). Adaptive Caching for Faster Video Generation with Diffusion Transformers. *arXiv preprint arXiv:2411.02397*.
- [9] Liu, F. et al. (2024). Timestep Embedding Tells: It's Time to Cache for Video Diffusion Model. *arXiv preprint arXiv:2411.19108*.
- [10] Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C. & Zhu, J. (2022). DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps. *arXiv preprint arXiv:2206.00927*.
- [11] Ma, X., Li, Y., Chen, J. et al. (2024). Latte: Latent Diffusion Transformer for Video Generation. *arXiv preprint arXiv:2401.03048*.
- [12] Ma, Z. et al. (2025). MagCache: Fast Video Generation with Magnitude-Aware Cache. *arXiv preprint arXiv:2506.09045*.
- [13] Peebles, W. & Xie, S. (2023). Scalable Diffusion Models with Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205.
- [14] PKU-YuanGroup (2024). Open-Sora-Plan v1.1: A High-Fidelity Video Synthesis Pipeline. *arXiv preprint arXiv:2412.01234*.
- [15] Sauer, A. et al. (2023). Adversarial Diffusion Distillation. In *Proceedings of the European Conference on Computer Vision 2024*.
- [16] Sauer, A., Rombach, R., Esser, P., Diagne, C., Dockhorn, T., Podell, D. & Black Forest Labs Team (2025). FLUX.1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space. *arXiv preprint arXiv:2506.15742*.
- [17] Singer, U. et al. (2022). Make-A-Video: Text-to-Video Generation without Text-Video Data. *arXiv preprint arXiv:2209.14792*.
- [18] Sun, K., Huang, K., Liu, X., Wu, Y., Xu, Z., Li, Z., & Liu, X. (2024). T2V-CompBench: A Comprehensive Benchmark for Compositional Text-to-video Generation. *arXiv preprint arXiv:2407.14505*.
- [19] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, **13**(4), 600–612.
- [20] Xi, H., Yang, S., Zhao, Y., Xu, C., Li, M., Li, X., Lin, Y. & Han, S. (2025). Sparse VideoGen: Accelerating Video Diffusion Transformers with Spatial-Temporal Sparsity. *arXiv preprint arXiv:2502.01776*.

- [21] Yin, T. et al. (2024). Improved Distribution Matching Distillation for Fast Image Synthesis. *arXiv preprint arXiv:2405.14867*.
- [22] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595.
- [23] Zhao, W., Bai, L., Rao, Y., Zhou, J., & Lu, J. (2023). UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models. *arXiv preprint arXiv:2302.04867*.
- [24] Zhao, X. et al. (2024). Real-Time Video Generation with Pyramid Attention Broadcast. *arXiv preprint arXiv:2408.12588*.
- [25] Zheng, Z., Peng, X., Yang, T., Shen, C., Li, S., Liu, H., Zhou, Y., Li, T., & You, Y. (2024). Open-Sora: Democratizing Efficient Video Production for All. *arXiv preprint arXiv:2412.20404*.
- [26] Zou, C. et al. (2024). Accelerating Diffusion Transformers with Dual Feature Caching. *arXiv preprint arXiv:2412.18911*.