# Supplementary Material: Accelerating Autoregressive Video Diffusion via History-Guided Cache and Residual Correction

**Kepan Nan**[1*] **Wangbo Zhao**[2*] **Penghao Zhou**[3]
**Jun Li**[1] **Zhenheng Yang**[3] **Jian Yang**[1] **Ying Tai**[1†]
[1]Nanjing University [2]National University of Singapore [3]Independent Researcher

## 1. Further Discussion on Enhanced Residual Correction

We conduct an ablation study to investigate the impact of the multi-step cache order within the Enhanced Residual Correction (ERC) module, as detailed in Table 1. Inspired by previous studies [1, 3], we formulate the residual at each acceleration step for segment $s$ using an $n$-th order ERC mechanism as follows:

$$\mathbf{r}_t^s = \mathbf{r}_{t_0}^s + \sum_{i=1}^{n} \lambda_t^s \cdot \alpha^{i-1} \cdot (\mathbf{r}_{t_0}^s - \mathbf{r}_{t_i}^s), \tag{1}$$

where $t_0 < t_1 < \cdots < t_n$ denote the selected recomputation timesteps, and $\alpha$ is a decay factor, which is set to 0.5 by default. Based on our ERC, the trajectory parameter $\lambda_t^s$ is shared across all segments and is derived from the first segment to ensure temporal stability:

$$\lambda_t^s = \lambda_t^1 = \frac{\mathrm{L1_{rel}}\left(\mathbf{r}_t^1, \mathbf{r}_{t_0}^1\right)}{\sum_{i=1}^{n} \alpha^{i-1} \cdot \mathrm{L1_{rel}}\left(\mathbf{r}_{t_0}^1, \mathbf{r}_{t_i}^1\right)}, \quad s > 1 \tag{2}$$

By substituting $\lambda_t^1$ into Eq. 1, the residual for segment $s$ is rewritten as:

$$\mathbf{r}_t^s = \mathbf{r}_{t_0}^s + \frac{\sum_{i=1}^{n} \mathrm{L1_{rel}}\left(\mathbf{r}_t^1, \mathbf{r}_{t_0}^1\right) \cdot \alpha^{i-1} \cdot \left(\mathbf{r}_{t_0}^s - \mathbf{r}_{t_i}^s\right)}{\sum_{i=1}^{n} \alpha^{i-1} \cdot \mathrm{L1_{rel}}\left(\mathbf{r}_{t_0}^1, \mathbf{r}_{t_i}^1\right)}, \quad s > 1 \tag{3}$$

Experimental results in Table 1 demonstrate that increasing the cache order from 0 to 1 yields notable improvements in visual quality, with PSNR rising from 24.13 to 24.79, SSIM from 0.8169 to 0.8266, and LPIPS decreasing from 0.1159 to 0.1117. Further increasing the cache order to 2 and 4 provides only marginal additional gains while incurring higher memory consumption. Meanwhile, inference speed remains largely unaffected across different cache orders, as ERC involves only lightweight residual computations. Consequently, we adopt the first-order ERC in our main paper, as it strikes an optimal balance between visual quality improvement and computational efficiency.

Table 1. **Ablation study of the multi-step cache order in ERC using FramePack-F1.** Best results are highlighted in **bold**, while the second-best result is underlined.

| Cache Order | Acceleration | | Visual Quality | | | VRAM (GB)↓ |
|---|---|---|---|---|---|---|
| | Latency(s)↓ | Speed↑ | PSNR↑ | SSIM↑ | LPIPS↓ | |
| $n=0$ | **96.40** | **1.52**× | 24.13 | 0.8169 | 0.1159 | **69.24** |
| $n=1$ | 96.74 | 1.51× | 24.79 | 0.8266 | 0.1117 | 70.70 |
| $n=2$ | 96.78 | 1.51× | **24.95** | 0.8227 | 0.1121 | 71.59 |
| $n=4$ | 97.14 | 1.51× | 24.79 | **0.8268** | **0.1110** | 73.38 |

## 2. Long-Range Video Generation

To further validate the robustness of ARCache in long-range video generation and its ability to mitigate error accumulation, we extend our experiments on FramePack-F1 from the standard 4-segment setting to a more challenging 12-segment scenario, effectively tripling the video length. As summarized in Table 2, ARCache-slow achieves the highest visual fidelity among all methods, yielding a PSNR of 20.37, SSIM of 0.6338, and LPIPS of 0.2546, thereby significantly outperforming previous caching-based acceleration approaches. ARCache-fast also maintains competitive quality, outperforming other fast baselines. Conversely, most baseline methods exhibit pronounced degradation in visual quality as video length increases, primarily due to severe error accumulation over extended temporal horizons. We further present qualitative results for long-range video generation in Figure 1 and 2, demonstrating that ARCache produces temporally consistent and visually faithful sequences over extended durations. In contrast, competing methods display noticeable artifacts and temporal drift. Once a minor error emerges (*e.g.*, spherical vegetation in TeaCache, or an incorrect camera direction in TaylorSeer), these methods are prone to propagating it sequentially over time. Ultimately, this unchecked accumulation causes the subsequent frames to deviate significantly from

* Equal contributions. † Corresponding author: yingtai@nju.edu.cn.

the original video. These findings underscore the effectiveness and robust generalizability of ARCache in long-range auto-regressive video synthesis.

Table 2. **Quantitative comparison of long-range video generation on FramePack-F1.** Best results are highlighted in **bold**.

| Model | Method | Visual Quality | | |
|---|---|---|---|---|
| | | PSNR↑ | SSIM↑ | LPIPS↓ |
| FramePack-F1 [4] | Vanilla (100% steps) | - | - | - |
| | Vanilla (50% steps) | 17.99 | 0.5656 | 0.3040 |
| | PAB ($I = 2$) [5] [ICLR'25] | 16.32 | 0.5043 | 0.3718 |
| | TeaCache-slow [2] [CVPR'25] | 18.97 | 0.5941 | 0.2845 |
| | TeaCache-fast [2] [CVPR'25] | 17.84 | 0.5587 | 0.3176 |
| | TaylorSeer [3] [ICCV'25] | 16.92 | 0.5350 | 0.3429 |
| | ARCache-slow (Ours) | **20.37** | **0.6338** | **0.2546** |
| | ARCache-fast (Ours) | 18.58 | 0.5692 | 0.3059 |

## 3. Compatibility with Inverted Anti-Drifting Sampling

To further evaluate the generality of our approach, we conduct experiments on FramePack using the inverted anti-drifting sampling paradigm [4]. Unlike the standard forward generation setting, where videos are synthesized sequentially from the first frame to the last, this inverted paradigm generates videos in a backward fashion, starting from the last frame and proceeding towards the first. Additionally, it leverages extra supervision from the initial frame to further suppress temporal drift during generation. As presented in Table 3, our ARCache consistently achieves superior visual quality under this paradigm. Specifically, ARCache-slow obtaining the best PSNR (31.38), SSIM (0.9127), and LPIPS (0.0352) among all baselines, while ARCache-fast delivers competitive results with significant acceleration. These findings demonstrate that our approach is robust and effective across different auto-regressive video generation paradigms, seamlessly adapting to both forward and backward sampling strategies.

Table 3. **Results on FramePack with inverted anti-drifting sampling.** Best results are highlighted in **bold**.

| Model | Method | Acceleration | | Visual Quality | | |
|---|---|---|---|---|---|---|
| | | Latency(s)↓ | Speedup↑ | PSNR↑ | SSIM↑ | LPIPS↓ |
| FramePack [4] | Vanilla (100% steps) | 144.74 | 1.00× | - | - | - |
| | Vanilla (50% steps) | 74.68 | 1.94× | 25.50 | 0.7956 | 0.0916 |
| | PAB ($I = 2$) [5] [ICLR'25] | 52.50 | 2.76× | 22.42 | 0.7214 | 0.1277 |
| | TeaCache-slow [2] [CVPR'25] | 92.89 | 1.56× | 26.75 | 0.8289 | 0.0708 |
| | TeaCache-fast [2] [CVPR'25] | 63.52 | 2.28× | 25.08 | 0.7948 | 0.0877 |
| | TaylorSeer [3] [ICCV'25] | 73.10 | 1.98× | 23.92 | 0.7694 | 0.0941 |
| | ARCache-slow (Ours) | 92.97 | 1.56× | **31.38** | **0.9127** | **0.0352** |
| | ARCache-fast (Ours) | **51.30** | **2.82**× | 25.85 | 0.7987 | 0.0783 |

## References


[1] Jiazi Bu, Pengyang Ling, Yujie Zhou, Yibin Wang, Yuhang Zang, Dahua Lin, and Jiaqi Wang. Dicache: Let diffusion model determine its own cache. *arXiv preprint arXiv:2508.17356*, 2025. 1

[2] Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7353–7363, 2025. 2

[3] Jiacheng Liu, Chang Zou, Yuanhuiyi Lyu, Junjie Chen, and Linfeng Zhang. From reusing to forecasting: Accelerating diffusion models with taylorseers. *arXiv preprint arXiv:2503.06923*, 2025. 1, 2

[4] Lvmin Zhang and Maneesh Agrawala. Packing input frame context in next-frame prediction models for video generation. *arXiv preprint arXiv:2504.12626*, 2025. 2

[5] Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. *arXiv preprint arXiv:2408.12588*, 2024. 2

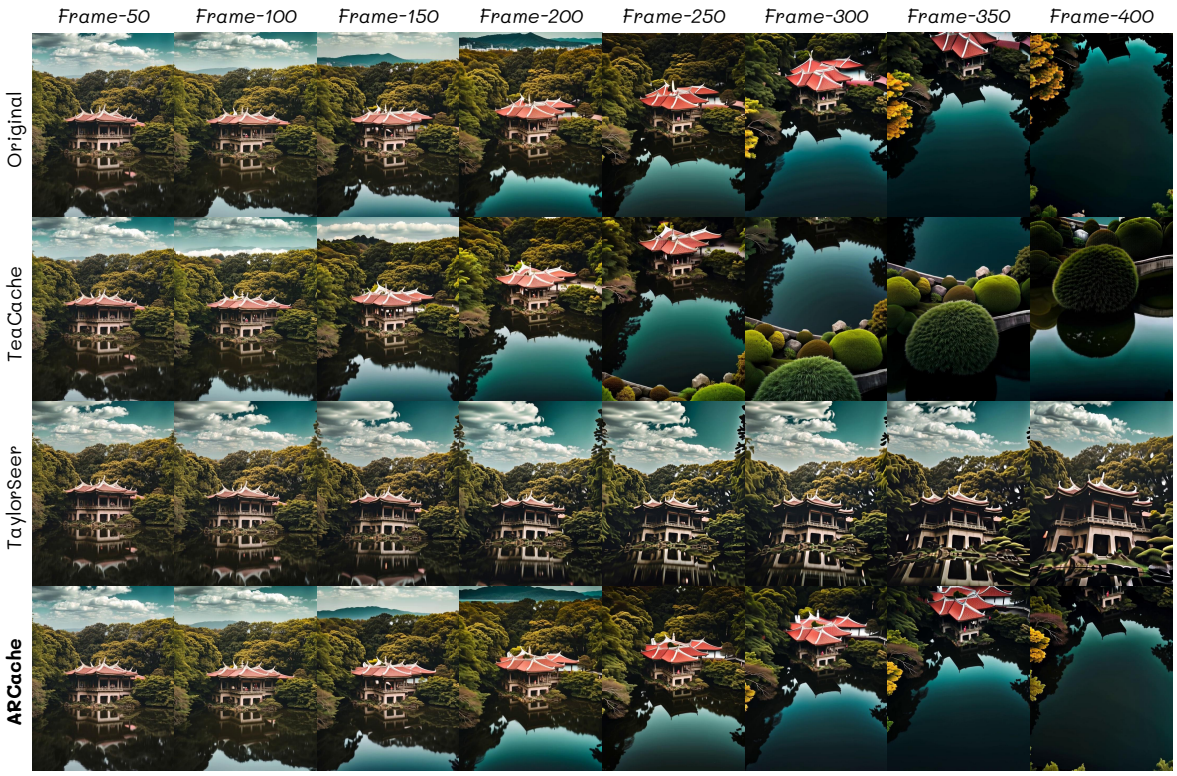


Figure 1. **Qualitative results for long-range video generation.** Please zoom in for more details.

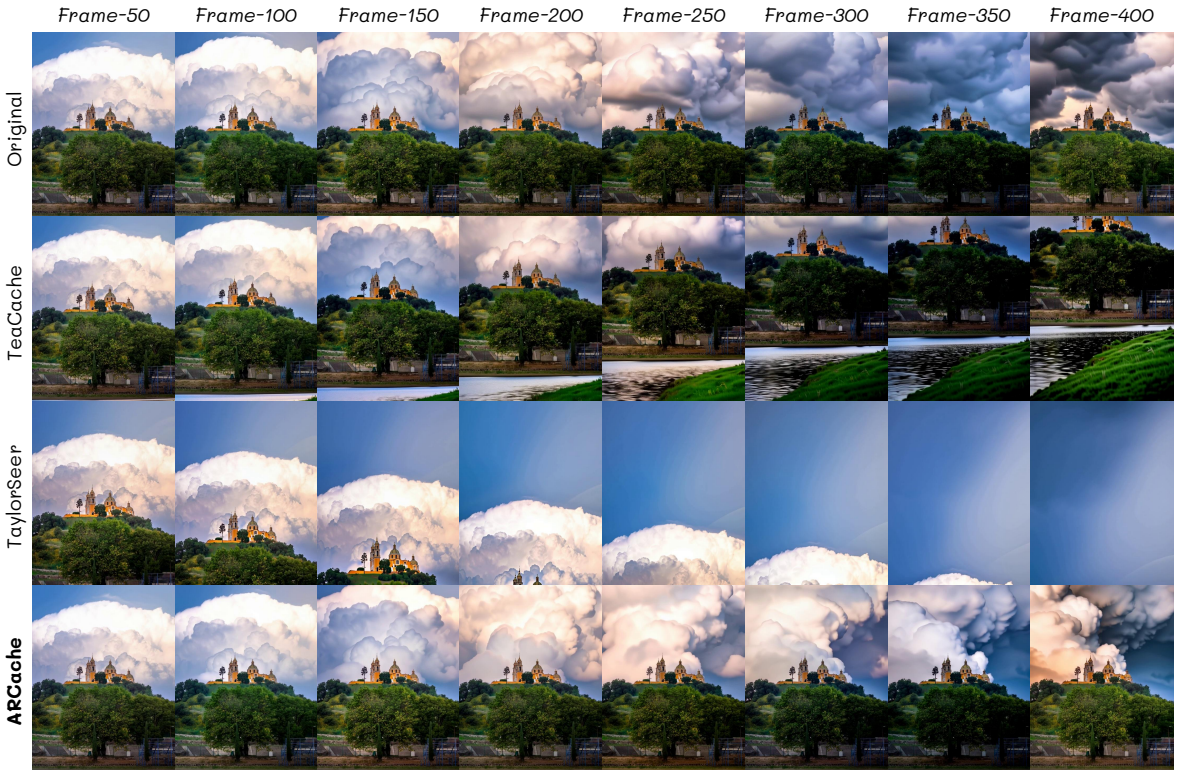


Figure 2. **Qualitative results for long-range video generation.** Please zoom in for more details.