Supplementary File
QuantCache: Adaptive Importance-Guided Quantization with Hierarchical Latent and Layer Caching for Video Generation

Junyi Wu[1*], Zhiteng Li[1*], Zheng Hui[2], Yulun Zhang[1†], Linghe Kong[1], Xiaokang Yang[1]
[1]Shanghai Jiao Tong University, [2]MGTV, Shanhai Academy

1. Detailed Description of Diffusion Model and Diffusion Transformers

Diffusion Models (DMs) [2, 4, 8, 10, 14, 18, 19] and Diffusion Transformers (DiTs) [3, 9, 11, 12, 15, 22] have emerged as transformative paradigms within the realm of generative modeling, captivating the research community with their remarkable ability to synthesize high-fidelity samples across a wide array of applications, including image generation, text-to-image synthesis, video production, and even audio synthesis. These models have revolutionized the field by offering a robust framework for capturing intricate data distributions, making them a cornerstone of modern artificial intelligence research. This section provides an in-depth exploration of their foundational principles, mathematical foundations, architectural designs, and the nuanced differences that set them apart, while also delving into their practical implications and evolving synergies.

1.1. Diffusion Models: Principles and Mechanisms

Diffusion Models are a class of probabilistic generative models that draw inspiration from the principles of non-equilibrium thermodynamics, a field that studies systems far from equilibrium states. These models operate by simulating a meticulously orchestrated process where data is progressively degraded through the addition of noise (the forward process) and subsequently reconstructed through a learned reverse process that seeks to recover the original data distribution from a state of pure noise. This dual-process approach allows DMs to transform complex, high-dimensional data distributions into simpler, more manageable forms—such as isotropic Gaussian noise—and then reverse-engineer this transformation to regenerate meaningful outputs. The elegance of this methodology lies in its ability to iteratively refine noisy inputs into coherent and visually appealing results, a feat that has garnered significant attention in both academic and industrial settings.

1.1.1. Forward Process

The forward diffusion process [4] is a systematic procedure that incrementally introduces Gaussian noise into the data over a predefined series of T timesteps, effectively blurring the original structure until it resembles random noise. Starting with an initial data sample x_0 drawn from the true data distribution $q(x_0)$, the forward process unfolds as a Markov chain, where each step introduces a controlled amount of noise. This is mathematically expressed as:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I), \tag{1}$$

where $\beta_t \in (0,1)$ represents a variance schedule that dictates the intensity of noise added at timestep t, and I denotes the identity matrix. As the number of timesteps increases, the process gradually erases the original data's structure, converging toward an isotropic Gaussian distribution, $x_T \sim \mathcal{N}(0, I)$, where all traces of the initial data are lost in a sea of randomness.

A particularly valuable property of the forward process is its closed-form expression, which allows direct sampling at any timestep t from the initial data x_0. This is captured by the equation:

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\alpha_t}x_0, (1-\alpha_t)I), \tag{2}$$

where $\alpha_t = \prod_{s=1}^{t}(1-\beta_s)$ quantifies the cumulative retention of the original signal, and $1-\alpha_t$ reflects the accumulated noise variance. This property simplifies the modeling process by providing a direct link between the original data and its noisy counterparts at any point in the diffusion timeline.

[*] Equal contribution

1.1.2. Reverse Process

The reverse process is the heart of the Diffusion Model, where the goal is to undo the noise corruption and regenerate the original data x_0 from the fully noisy state x_T. This is achieved by training a parameterized model $p_\theta(x_{t-1}|x_t)$ to approximate the true posterior distribution $q(x_{t-1}|x_t)$, effectively learning to reverse the forward diffusion steps. This reverse process is also modeled as a Markov chain:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \tag{3}$$

where $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ represent the mean and covariance predicted by a neural network parameterized by θ. In practical implementations, the covariance is often simplified and fixed (e.g., $\Sigma_\theta = \beta_t I$), allowing the model to focus its efforts on accurately predicting the mean, which guides the denoising process.

The training of Diffusion Models involves optimizing the variational lower bound (VLB) on the data likelihood, a complex objective that ensures the model learns a faithful representation of the data distribution. However, a more streamlined and widely adopted approach, introduced by [4], employs a simplified loss function:

$$L_{\text{simple}} = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right], \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, I)$ is the noise introduced at timestep t, and $\epsilon_\theta(x_t, t)$ is the noise predicted by the model based on the noisy input $x_t = \sqrt{\alpha_t}x_0 + \sqrt{1-\alpha_t}\epsilon$. This noise prediction framework reframes the problem in a way that is both computationally efficient and empirically robust, enabling the model to learn the reverse process with remarkable precision.

1.1.3. Architecture

The architectural backbone of traditional Diffusion Models often relies on U-Net [16, 17, 21] architectures, which incorporate convolutional layers to effectively process spatial data such as images. The U-Net takes the noisy data x_t and the timestep t—encoded through sinusoidal embeddings—as inputs, leveraging its encoder-decoder structure to predict the noise component $\epsilon_\theta(x_t, t)$ at each step. The U-Net's skip connections facilitate the preservation of spatial details across different resolution levels, ensuring that the model can reconstruct fine-grained features during the reverse process. This design choice has proven particularly effective for tasks requiring pixel-level precision, such as high-resolution image synthesis.

1.2. Diffusion Transformers: A Paradigm Shift

Diffusion Transformers (DiTs) [3, 9, 11, 12, 15, 22] mark a significant evolution in the diffusion framework by replacing the convolutional U-Net architectures with Transformer-based designs, harnessing the latter's inherent strengths in modeling long-range dependencies, scalability, and adaptability to diverse data types. Introduced by [15], DiTs represent a paradigm shift that adapts the diffusion process to operate on tokenized or latent representations rather than raw pixel data, making them exceptionally well-suited for handling high-dimensional inputs and tackling complex cross-modal tasks such as text-to-image generation or video synthesis.

1.2.1. Key Innovations

A defining characteristic of DiTs is their departure from the local spatial convolutions of U-Nets in favor of self-attention mechanisms, which enable the model to capture global relationships across the entire input sequence. The input data—typically an image—is first encoded into a sequence of patches or latent tokens using a pre-trained encoder, such as a Variational Autoencoder (VAE). The noisy latent representation z_t at timestep t is then processed by the Transformer architecture:

$$z_t = \sqrt{\alpha_t}z_0 + \sqrt{1-\alpha_t}\epsilon, \tag{5}$$

where z_0 is the latent representation of the original data x_0, and ϵ is additive Gaussian noise. This latent diffusion approach reduces computational complexity by operating in a compressed latent space, allowing the model to focus on high-level semantic features rather than pixel-level details.

The Transformer architecture predicts the noise $\epsilon_\theta(z_t, t)$ using a stack of multi-head self-attention layers, conditioned on the timestep t (encoded via embeddings) and, in many cases, additional contextual information such as text prompts through cross-attention mechanisms. The final output is then decoded back into the data space—such as an image—using a corresponding decoder, bridging the gap between latent representations and tangible outputs.

1.2.2. Training and Sampling

The training objective for DiTs aligns closely with that of standard DMs, employing the simplified loss function from Equation 4, but applied within the latent space to predict the noise component. The sampling phase follows the reverse diffusion process, iteratively denoising the initial noisy latent representation $z_T \sim \mathcal{N}(0, I)$ to reconstruct z_0, which is subsequently decoded into the final data output x_0. The Transformer's ability to process tokens in parallel offers a potential advantage over the sequential nature of convolutional operations, paving the way for accelerated training and inference, especially on modern hardware optimized for parallel computation.

1.2.3. Advantages and Trade-offs

DiTs shine in scenarios that demand global coherence and contextual understanding, such as text-conditioned image generation, where the Transformer's capacity to model long-range interactions proves invaluable. This makes them particularly effective in applications like Stable Diffusion [17], where a latent diffusion process is seamlessly integrated with Transformer-based text conditioning to generate highly detailed and contextually relevant images. However, this power comes with trade-offs: DiTs may demand greater computational resources than U-Nets, especially for smaller-scale tasks, and their performance hinges on the efficacy of the tokenization strategy used to manage input dimensionality. Careful design choices, such as the selection of the VAE encoder and the tuning of attention mechanisms, are critical to optimizing their efficiency and output quality.

1.3. Comparison and Synergy

While Diffusion Models with U-Net architectures excel in their computational efficiency and are ideally suited for tasks requiring precise pixel-level synthesis—such as generating photorealistic images with fine textures—Diffusion Transformers offer a higher degree of flexibility and superior performance in scenarios involving complex dependencies or multimodal data integration. The architectural distinction between the two approaches is evident: U-Nets leverage local convolutions to build spatial hierarchies, whereas DiTs employ global self-attention to model relationships across the entire input. This fundamental difference underpins their respective strengths, with DiTs finding particular success in advanced applications like Stable Diffusion, where the combination of latent diffusion and Transformer-based text conditioning has set new benchmarks in generative modeling.

From a mathematical perspective, both DMs and DiTs adhere to the same underlying diffusion framework, as outlined in Equations 1 through 4, with the primary divergence arising from their neural architectures and the representation spaces they operate within. This shared foundation opens the door to hybrid approaches that blend convolutional and Transformer components, aiming to synergize the local detail preservation of U-Nets with the global context awareness of Transformers. Such hybrid models are beginning to emerge, promising to deliver the best of both worlds by balancing computational efficiency with expressive power.

In conclusion, Diffusion Models and Diffusion Transformers represent complementary pillars of advancement in generative modeling, each tailored to leverage specific strengths. DMs provide a robust foundation for fine-grained spatial synthesis, excelling in tasks where pixel-level accuracy is paramount, while DiTs push the boundaries of scalability and context-rich generation, enabling breakthroughs in multimodal and high-dimensional applications. Together, they form a dynamic duo that continues to shape the future of generative AI, with ongoing research exploring new ways to integrate their capabilities for even greater innovation.

2. Detailed Evaluation Metrics

Our evaluation framework, inspired by [1, 7, 23], uses VBench [5, 6] to conduct quality experiments. We select eight representative indicators to measure the quality of video generation, which are introduced as follows:

- **Aesthetic Quality**: Measures the artistic merit and attractiveness of individual frames as perceived by viewers.
- **Motion Smoothness**: Judges the smoothness and realism of motion, aligned with natural physical principles.
- **Imaging Quality**: Analyzes the presence of imperfections such as noise or excessive brightness in each frame.
- **Scene Consistency**: Verifies if the depicted scenes match the intended narrative of the prompt.
- **Subject Consistency**: Determines if key subjects maintain a uniform appearance across the video.
- **Background Consistency**: Assesses whether background elements remain stable throughout the sequence.
- **Dynamic Degree**: Quantifies the level of activity by computing the average optical flow across frames.
- **Overall Consistency**: Evaluates consistency in both thematic content and stylistic execution relative to the prompt.

To conduct these assessments, we leverage three prompt collections from the VBench GitHub repository, generating one video per prompt for analysis:

- **subject_consistency.txt**: Features 72 prompts targeting dynamic degree, motion smoothness, and subject consistency.
- **overall_consistency.txt**: Contains 93 prompts to examine overall consistency, aesthetic quality, and imaging quality.
- **scene.txt**: Includes 86 prompts focused on overall consistency and scene consistency.

2.1. Metrics

We apply a set of specialized metrics with Open-Sora prompts to quantify video quality across the defined categories:
DOVER's VQA. Using the DOVER approach [20], we assess video quality in two dimensions. The Technical Score (VQA-T) identifies flaws like blur, noise, or overexposure, while the Aesthetic Score (VQA-A) evaluates artistic attributes, including color balance, layout harmony, realism, and overall visual artistry.
CLIPSIM and CLIP-temp. Implemented via EvalCrafter [13], these metrics utilize the CLIP-ViT-B/32 model. CLIPSIM calculates the average similarity between all video frames and the input prompt, assessing how well the video captures the described concept. CLIP-temp, conversely, measures similarity between adjacent frames, averaging these scores to evaluate the video's semantic continuity over time.

3. Cosine Similarity Analysis Across Timesteps

To further validate the generalizability of our *Structural Redundancy-Aware Pruning (SRP)* mechanism, we analyze cosine similarity across multiple timesteps, as introduced in the main text (*Quantifying Redundancy with Cosine Similarity*). We present cosine similarity plots for various timesteps, with examples shown in Figs. 1 to 4. These plots measure layer-wise similarity $S_t^{(l,l+1)}$ between spatial-temporal attention, cross-attention, and FFN.

3.1. Analysis of Cosine Similarity Plots

In Fig. 1, we illustrate cosine similarity at timestep 50 for spatial differences (a) and temporal differences (b) across different components: spatial self-attention (sp_st_att), spatial feed-forward (sp_ffn), spatial cross-attention (sp_ca) in (a), and temporal self-attention (tp_st_att), temporal feed-forward (tp_ffn), temporal cross-attention (tp_ca) in (b). The x-axis represents the layer index (0 to 27), and the y-axis shows cosine similarity (0 to 1.0). In (a), sp_ffn consistently exhibits higher similarity (often above 0.6) across layers 10–25, indicating greater spatial redundancy, while sp_ca shows lower similarity (around 0.2–0.4), reflecting its role in capturing diverse spatial features. In (b), tp_ffn also displays higher similarity (peaking near 0.7), suggesting temporal redundancy, whereas tp_st_att and tp_ca fluctuate more (0.1–0.5), indicating sensitivity to temporal dynamics.

3.2. Generalizability Across Timesteps

We observe that cosine similarity patterns remain consistent across timesteps, with plots at other timesteps in Figs. 2 to 4 exhibiting similar trends: feed-forward components (sp_ffn, tp_ffn) frequently show higher redundancy, while attention mechanisms (sp_ca, tp_st_att, tp_ca) display greater variability. This consistency suggests that our SRP method, which skips layers based on $S_t^{(l,l+1)}$ thresholds, can generalize effectively across the diffusion process. By leveraging these stable redundancy patterns, our approach ensures robust layer-skipping decisions, enhancing efficiency without compromising video quality.

4. Additional Results

We present additional results of generated video frames by Open-Sora [24], ViDiT-Q [23], AdaCache [7], and QuantCache in Figs. 5 to 7. Our method achieves a remarkable speedup of 6.72× compared to the baseline Open-Sora, while incurring almost no loss in precision. Due to limited hardware support for W4A6 quantization, our W4A6 configuration is implemented on the W8A8 CUDA kernel, ensuring compatibility while maintaining efficiency. Notably, our approach excels in delivering superior aesthetic quality, characterized by vivid colors and harmonious compositions. In Fig. 5, our method enhances the vividness of the red and yellow rock strata, creating a striking contrast with the turquoise sea that feels almost painterly in its beauty. In terms of imaging quality, our generated frames exhibit exceptional sharpness, fine-grained details, and robust texture preservation. For Fig. 6, the intricate details of snow-laden branches and the subtle ruts in the dirt road are captured with precision, lending the scene a tangible realism that draws viewers in. In Fig. 7, the waterfall's descent into the lake is depicted with unbroken continuity, as the cliff's rugged outline and the lake's calm surface remain perfectly aligned frame after frame, enhancing the video's representation of nature's power and tranquility. Both excellent generation quality and high speedup position our approach as a highly competitive solution, balancing efficiency and visual quality with unparalleled effectiveness.

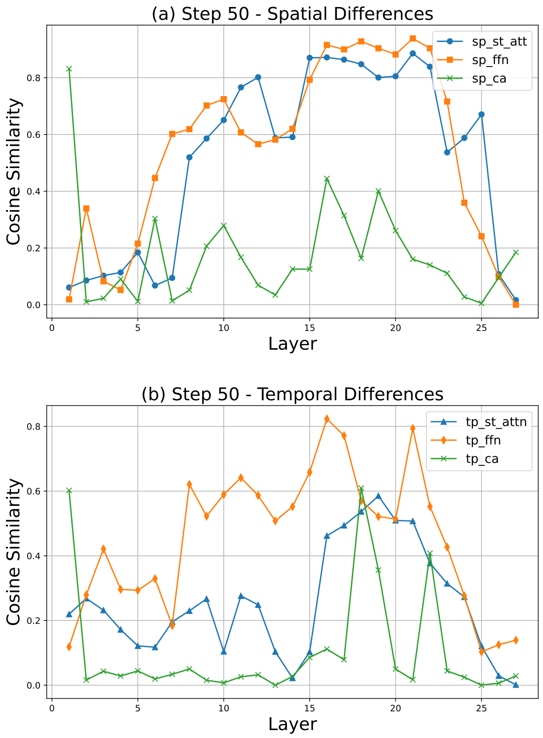


Figure 1. Cosine similarity for the 50th timestep

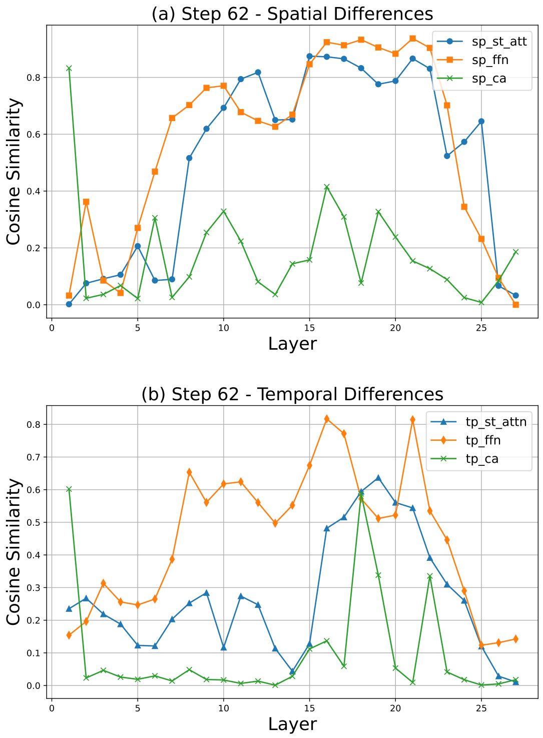


Figure 2. Cosine similarity for the 62nd timestep

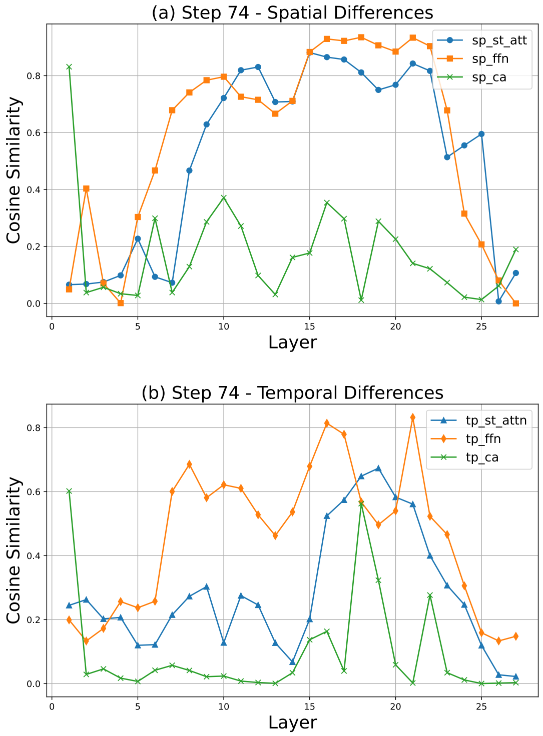


Figure 3. Cosine similarity for the 74th timestep

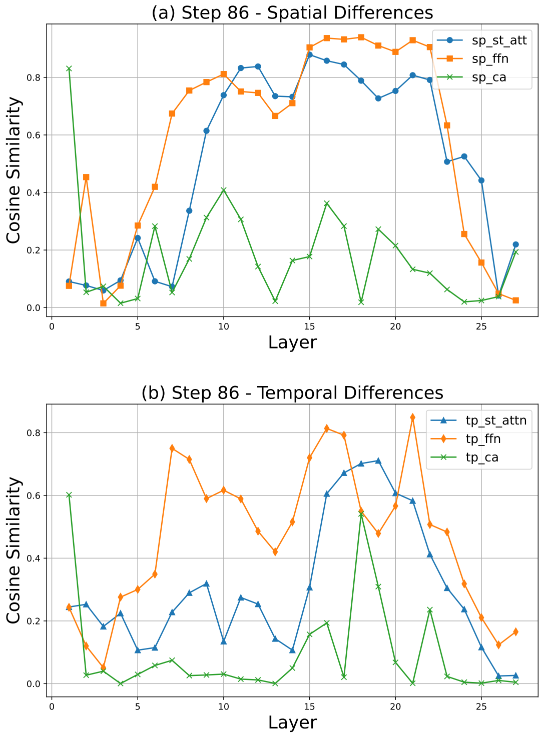
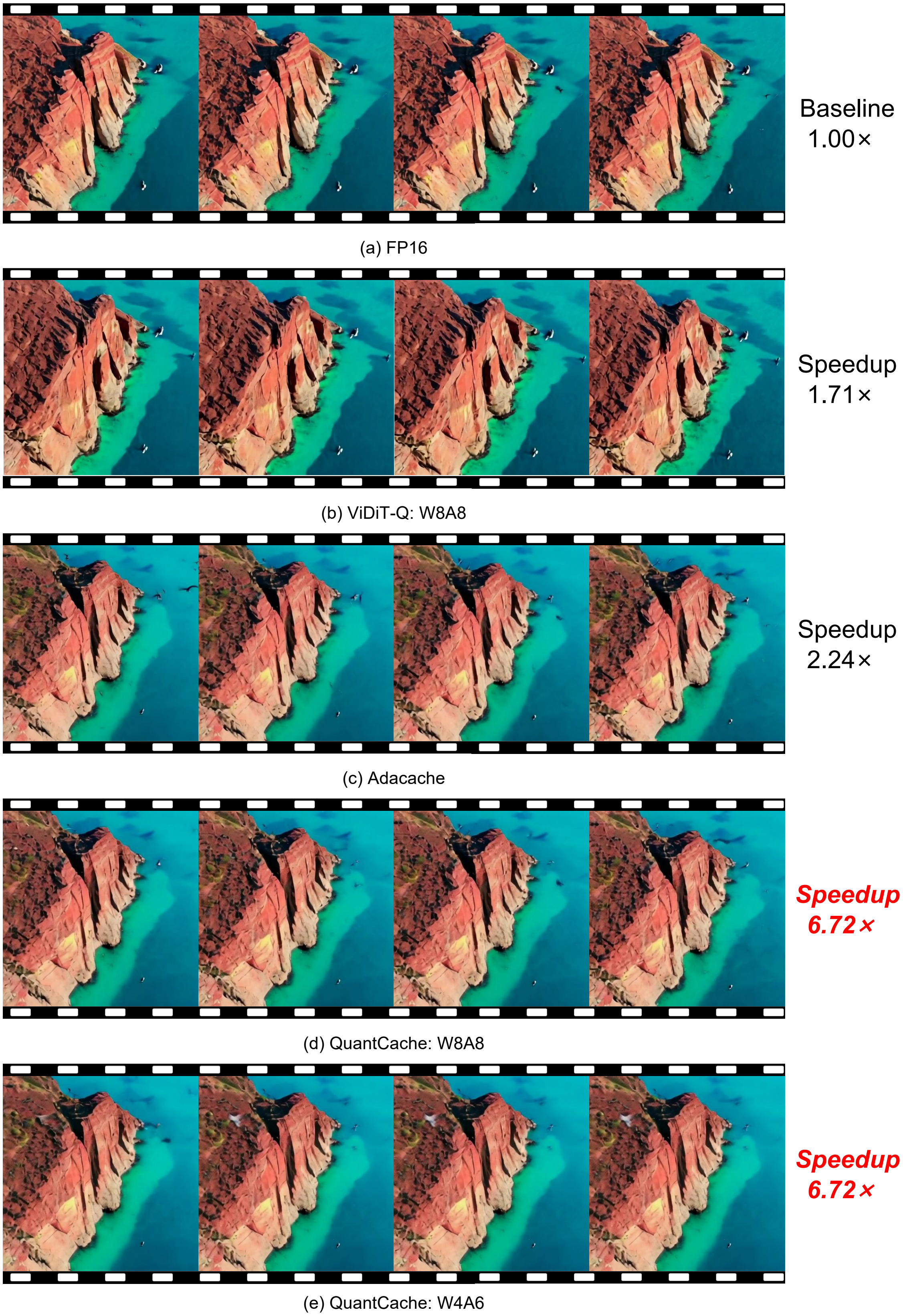


Figure 4. Cosine similarity for the 86th timestep



(a) FP16

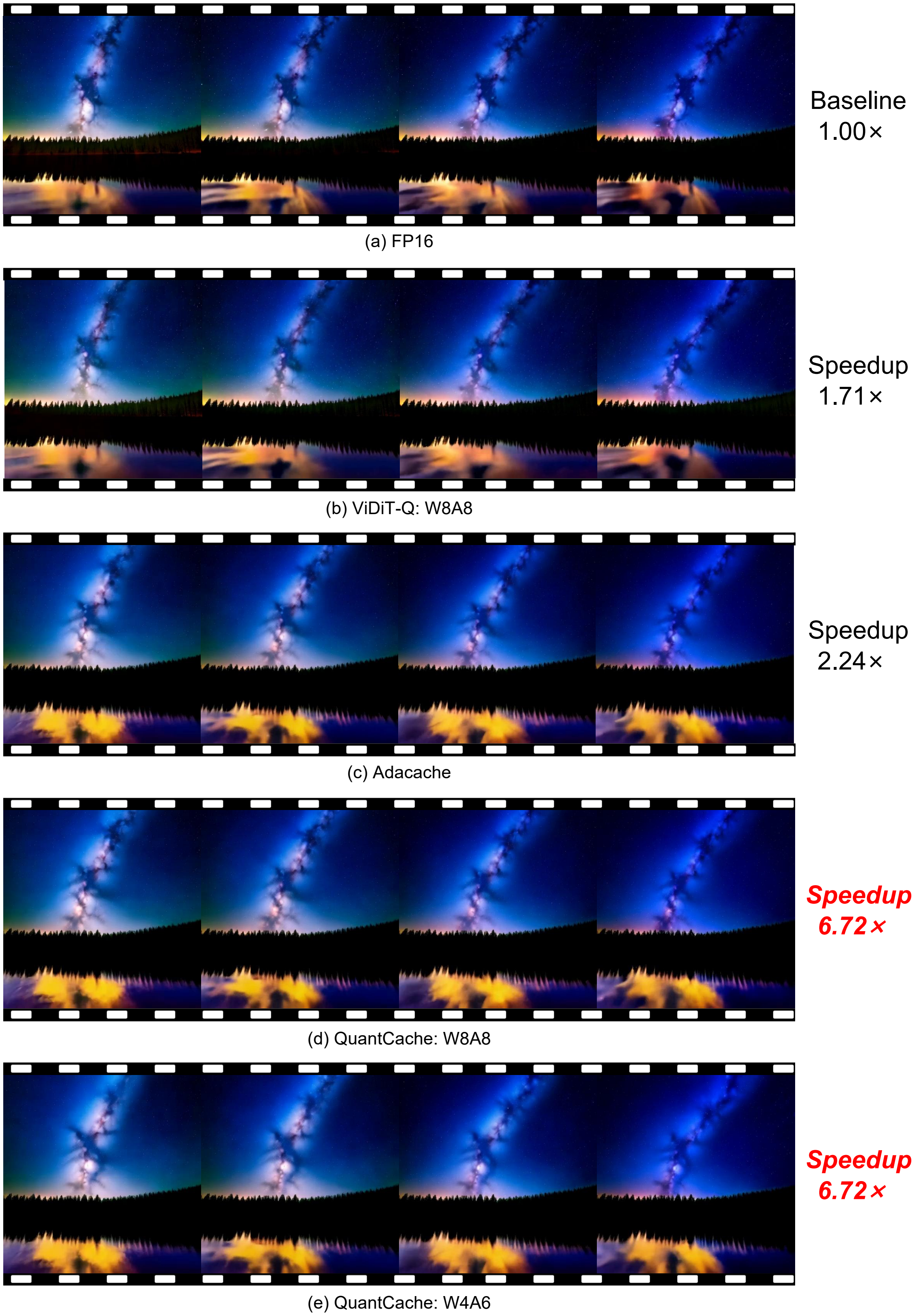
(b) ViDiT-Q: W8A8

(c) Adacache

(d) QuantCache: W8A8

(e) QuantCache: W4A6

Figure 5. Visual Comparison with prompt: “A soaring drone footage captures the majestic beauty of a coastal cliff, its red and yellow stratified rock faces rich in color and against the vibrant turquoise of the sea. ... The video captures the essence of pristine natural beauty untouched by human structures.”



(a) FP16

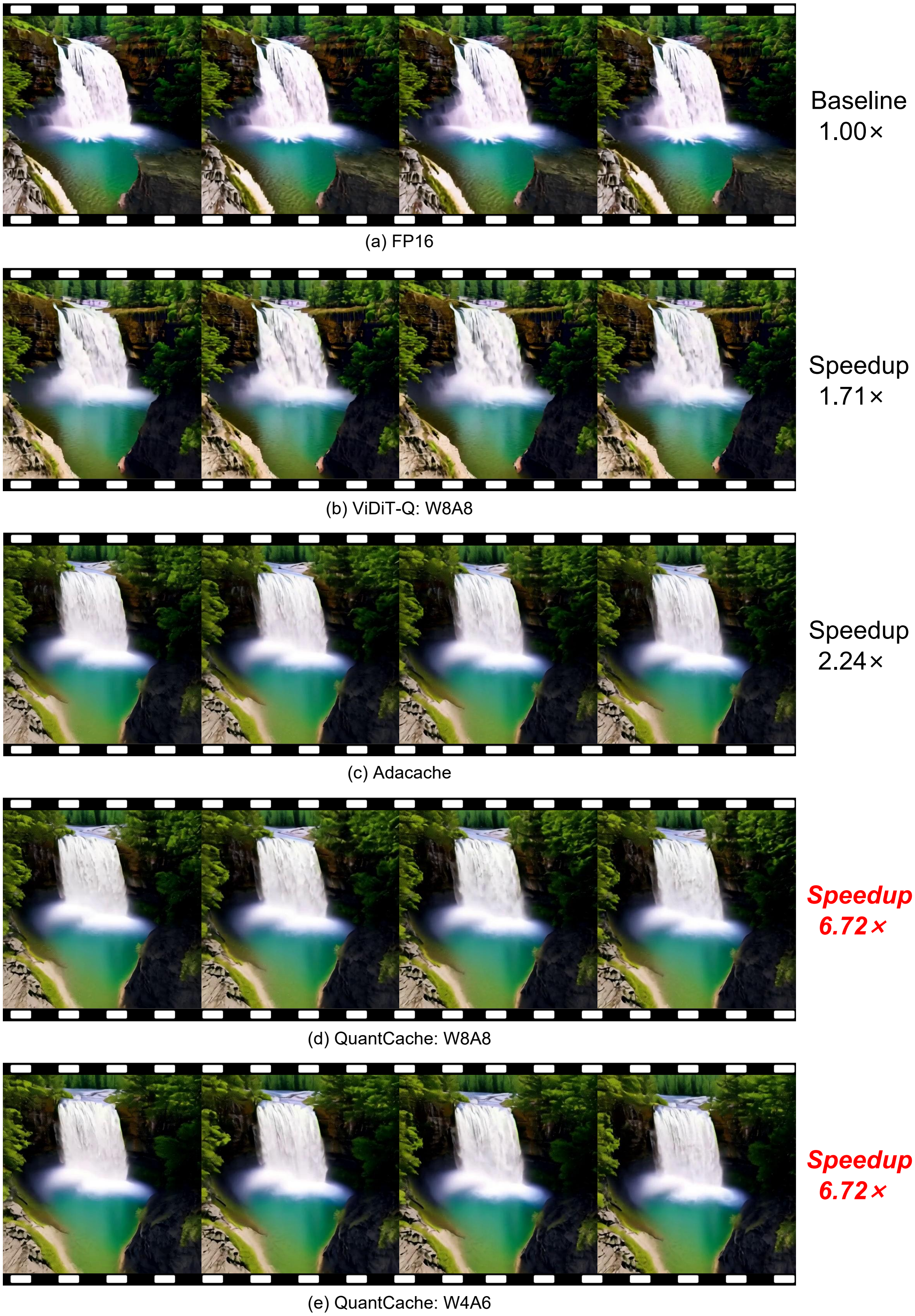
(b) ViDiT-Q: W8A8

(c) Adacache

(d) QuantCache: W8A8

(e) QuantCache: W4A6

Figure 6. Visual Comparison with prompt: "A serene night scene in a forested area. The first frame shows a tranquil lake reflecting the star-filled sky above. ... The style of the video is naturalistic, emphasizing the beauty of the night sky and the peacefulness of the forest. "



(a) FP16

(b) ViDiT-Q: W8A8

(c) Adacache

(d) QuantCache: W8A8

(e) QuantCache: W4A6

Figure 7. Visual Comparison with prompt: “A majestic beauty of a waterfall cascading down a cliff into a serene lake. ... The video is a stunning representation of nature’s power and beauty.”

References

[1] Lei Chen, Yuan Meng, Chen Tang, Xinzhu Ma, Jingyan Jiang, Xin Wang, Zhi Wang, and Wenwu Zhu. Q-dit: Accurate post-training quantization for diffusion transformers. *arXiv preprint arXiv:2406.17343*, 2024. 3

[2] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. In *ICLR*, 2023. 1

[3] Ali Hatamizadeh, Jiaming Song, Guilin Liu, Jan Kautz, and Arash Vahdat. Diffit: Diffusion vision transformers for image generation. In *ECCV*, 2024. 1, 2

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2

[5] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024. 3

[6] Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Vbench++: Comprehensive and versatile benchmark suite for video generative models. *arXiv preprint arXiv:2411.13503*, 2024. 3

[7] Kumara Kahatapitiya, Haozhe Liu, Sen He, Ding Liu, Menglin Jia, Chenyang Zhang, Michael S Ryoo, and Tian Xie. Adaptive caching for faster video generation with diffusion transformers. *arXiv preprint arXiv:2411.02397*, 2024. 3, 4

[8] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 1

[9] Gwanghyun Kim, Alonso Martinez, Yu-Chuan Su, Brendan Jou, José Lezama, Agrim Gupta, Lijun Yu, Lu Jiang, Aren Jansen, Jacob Walker, et al. A versatile diffusion transformer with mixture of noise levels for audiovisual generation. *arXiv preprint arXiv:2405.13762*, 2024. 1, 2

[10] Diederik P Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. On density estimation with diffusion models. In *NeurIPS*, 2021. 1

[11] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2024. 1, 2

[12] Zhenghao Lin, Yeyun Gong, Yelong Shen, Tong Wu, Zhihao Fan, Chen Lin, Weizhu Chen, and Nan Duan. Genie: Large scale pre-training for text generation with diffusion model. *arXiv preprint arXiv:2212.11685*, 2022. 1, 2

[13] Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *CVPR*, 2024. 4

[14] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICCV*, 2021. 1

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

[16] Vidya Prasad, Chen Zhu-Tian, Anna Vilanova, Hanspeter Pfister, Nicola Pezzotti, and Hendrik Strobelt. Unraveling the temporal dynamics of the unet in diffusion models. *arXiv preprint arXiv:2312.14965*, 2023. 2

[17] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3

[18] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *ArXiv*, abs/2202.00512, 2022. 1

[19] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 1

[20] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou Hou, Annan Wang, Wenxiu Sun Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *ICCV*, 2023. 4

[21] Zhaohu Xing, Liang Wan, Huazhu Fu, Guang Yang, and Lei Zhu. Diff-unet: A diffusion embedded network for volumetric segmentation. In *MICCAI*, 2023. 2

[22] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025. 1, 2

[23] Tianchen Zhao, Tongcheng Fang, Enshu Liu, Wan Rui, Widyadewi Soedarmadji, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation. In *ICLR*, 2025. 3, 4

[24] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, march 2024. *URL https://github. com/hpcaitech/Open-Sora*, 2024. 4