D^3-TOK: SINGLE-PASS DYNAMIC-LENGTH 1D VIDEO DIFFUSION TOKENIZER

Anonymous authors
Paper under double-blind review

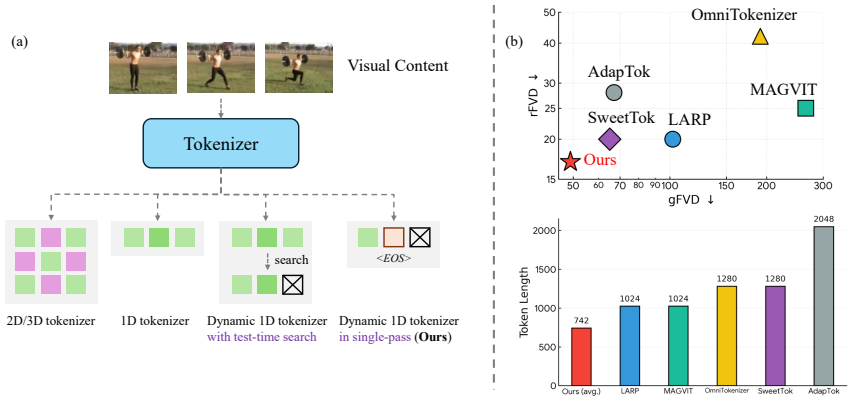


Figure 1: **Comparison between existing visual tokenizers and our proposed D^3-Tok.** (a) 2D/3D tokenizers like VQVAE (van den Oord et al., 2018) and fixed-length 1D tokenizers like Titok (Yu et al., 2024) allocate a predetermined number of tokens regardless of visual complexity, while search-based dynamic methods like ElasticTok (Yan et al., 2025) rely on test-time search to determine optimal token length. In contrast, our approach directly determines the end-of-sequence (<EOS>) in a single-pass manner without search. (b) Our D^3-Tok achieves the best rFVD and gFVD with the fewest tokens, significantly outperforming existing tokenizers.

ABSTRACT

Fixed-length visual tokenization with 2D/3D inductive bias is suboptimal in learning rich semantics and facilitating visual autoregressive modeling. 1D visual tokenizers mitigate these limitations, but existing flexible-length approaches often require test-time search, reducing inference efficiency. We propose D^3-Tok, a single-pass **D**ynamic-length 1**D** video **D**iffusion **Tok**enizer. D^3-Tok maps each video to a 1D latent sequence and predicts a content-dependent end-of-sequence position in a single pass. Specifically, we design a detector to learn to remove redundant tokens in a content-adaptive manner, guided by an off-the-shelf video complexity prior and regularized by reconstruction fidelity. The retained token prefix is decoded by a diffusion decoder. Experimental results show that D^3-Tok surpasses all baselines, setting a new state-of-the-art rFVD/gFVD among video tokenizers with the highest token compression rate. It also achieves comparable performance to its fixed-length counterparts while enabling semantic representations. Downstream tasks on video generation demonstrate that our method achieves better results on autoregressive modeling compared to 2D or 3D methods. Code will be released.

1 INTRODUCTION

Over the past few years, autoregressive (AR) modeling in large language models (OpenAI, 2024; Llama3, 2024; Gemini, 2025; Qwen, 2025) has demonstrated its potential in image and video generation. This success relies on discrete 2D/3D visual tokenizers and motivates the exploration of how AR could be used to unify more distinct tasks (Chameleon, 2024). However, such approaches still lag behind in visual generation and multimodal understanding compared to unified models (Xie

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: **Comparison among different visual tokenizers.**

Method	No inductive bias	Dynamic Length	Single-pass
VAE (Kingma and Welling, 2013)	×	×	✓
OmniTokenizer (Wang et al., 2024b)	×	×	✓
MAGVIT (Yu et al., 2023)	×	×	✓
LARP (Wang et al., 2025b)	✓	×	✓
ElasticTok (Yan et al., 2025)	✓	✓	×
AdapTok (Li et al., 2025)	✓	✓	×
D^3-Tok (Ours)	✓	✓	✓

et al., 2025a;b; Deng et al., 2025; Pan et al., 2025) that employ continuous vision encoders like SigLIP (Zhai et al., 2023) and diffusion modeling. This limitation stems from the 2D/3D inductive bias, which is less compatible with visual autoregressive modeling and the redudant tokens hamper rich semantic representation, as demonstrated by prior works (Fan et al., 2024; Wu et al., 2025).

1D visual tokenizers (Yu et al., 2024; Wang et al., 2025a; Sargent et al., 2025; Wang et al., 2025b; Tan et al., 2025) such as TiTok have demonstrated notably stronger generative capabilities under the AR paradigm. Moreover, Flextok (Bachmann et al., 2025) has shown that randomly dropping tokens during training encourages the remaining tokens to capture higher-level semantic representations and increase the information entropy contained within each token. However, these methods (Bachmann et al., 2025; Duggal et al., 2025; 2024) rely on a tedious process to determine the optimal token length by searching over possible token lengths during the inference stage. This significantly bottlenecks token dynamics and compute efficiency.

In this paper, we aim to bridge the visual tokenizer with single-pass dynamics, better AR generative property, and richer semantics. We propose D^3-Tok, a 1D video diffusion tokenizer that attains dynamic token length in a single-pass manner, without requiring any search during test-time. Specifically, given an input video, an encoder is first learned to map the visual contents to the learnable latent tokens. We draw inspiration from the <EOS> (End-of-Sequence) token in LLMs and enable the visual tokenizer to automatically terminate once it reaches the optimal token length. Our key insight is that content-dependent sequence truncation can be formulated as predicting the position of an <EOS> token. We implement this formulation with a detector trained from a noisy complexity prior and reconstruction feedback. <EOS> token removes the need for autoregressive models to regress redundant tokens, thereby improving AR generative performance. It also encourages tokens to capture high-level semantics. To get a better decoding fidelity and further enhance semantic representation, we employ a flow matching (Xingchao et al., 2022)-based decoder to decode the remaining tokens. The inherent stochasticity of flow matching enables the one-to-many mapping from tokens to videos, which effectively forms semantic clusters. This property directly benefits tasks requiring rich semantics. We provide a detailed comparison among different methods in Fig. 1 (a) and Tab. 1.

Extensive experimental results demonstrate that D^3-Tok outperforms all existing baselines, establishing a new state-of-the-art among 1D video tokenizers. Moreover, it attains performance comparable to fixed-length counterparts while providing the most compact and semantically rich representations. Downstream video generation experiments further validate that our 1D tokens are inherently more compatible with autoregressive modeling than 2D or 3D tokens.

Collectively, the main contributions of this paper can be summarized as:

- We propose D^3-Tok, the first single-pass dynamic-length 1D video diffusion tokenizer, requiring no test-time search.
- We demonstrate that D^3-Tok surpasses all baselines and achieves comparable performance to fixed-length counterparts on video reconstruction and generation tasks.
- Our high-compression tokenizer facilitates more effective visual autoregressive modeling by eliminating redundant token dependencies and enabling richer semantic representations for understanding tasks.

2 Related Works

Visual Tokenizer. The development of visual tokenizers has been driven by the needs of visual auto-regressive (AR) generation (Chowdhery et al., 2022; Chang et al., 2022; Esser et al., 2021; van den Oord et al., 2018) and understanding (Carion et al., 2020; Jin et al., 2024a;b; Lin et al., 2024; Wang et al., 2024a) tasks. VQ-VAE (van den Oord et al., 2018) injects a learnable discreate codebook to continous VAE (Kingma and Welling, 2013) latent to discretize its feature space. MAGVIT (Yu et al., 2023) introduce a 3D tokenizer to quantize a video into spatial-temporal visual tokens. These methods enforce a one-to-one correspondence between latent tokens and video patches. In contrast, 1D tokenizer (Yu et al., 2024) removes the predefined grids by mapping the visual content into a 1D token sequence, achieving higher compression rates. Subsequent works (Sargent et al., 2025; Gao and Shou, 2025) replaces the decoder with a diffusion model for better reconstruction, while SelfTok scales up 1D tokenizers and demonstrates its scalability across understanding, generation, and reasoning tasks. Moreover, 1D tokenizers possess several properties that 2D tokenizers inherently lack. Recent work (Beyer et al., 2025) demonstrates its zero-shot editing capability, FlexTok, VideoFlexTok and Elastictok (Bachmann et al., 2025; Atanov et al., 2026; Yan et al., 2025) show that 1D tokenizers can reconstruct visual content under variable token lengths. FlexTok further observes that highly compressed tokens can activate semantic information. However, all these methods are unable to automatically adjust the trade-off between reconstruction quality and token length. Their implementations of dynamic token length largely rely on searching over all possible lengths, making the process inefficient and limiting broader applicability.

Dynamic Length Tokenizer. As dynamic length tokenizers can directly improve efficiency, researchers have begun to explore this direction. ElasticTok (Yan et al., 2025) allocates a variable number of tokens to each video frame via a masking technique, enabling efficient representation of both images and videos. ALIT (Duggal et al., 2024) proposes a recurrent encoder that incrementally refines 2D tokens into 1D latent tokens. KARL (Duggal et al., 2025) draws on algorithmic information theory to predict the required number of tokens in a single forward pass and halts further token generation once reconstruction quality meets a target. AdapTok (Li et al., 2025) introduces a video tokenizer that allocates tokens over time in a temporally causal manner using a block-wise masking strategy and integer linear programming for token budget allocation per frame. All of these methods require search during the inference stage, limiting their applicability to downstream tasks. Concurrent work EVATok (Tianwei et al., 2026) bypasses searching via local block-wise token routing, while our method fundamentally diverges by formulating dynamic tokenization as a global 1D sequence truncation problem, avoiding semantic fragmentation in local area.

3 Method

We organize our method into four parts. First, we introduce the preliminary knowledge of flow matching in Sec. 3.1. Then, in Sec. 3.2, we present the overall architecture of our 1D video diffusion tokenizer. In Sec. 3.3 and Sec. 3.4, we describe how single-pass dynamic-length tokenization is achieved. Finally, in Sec. 3.5, we detail our training strategy.

3.1 Preliminary

Flow Matching. Flow-based generative models aim to learn a velocity field v_θ that transports Gaussian noise $\epsilon \sim N(0, I)$ to the distribution of real data x_0. Flow matching (Xingchao et al., 2022) adopts linear interpolation between noise ϵ and data x_0:

$$x_t = (1-t)x_0 + t\epsilon, \epsilon \sim N(0, I). \tag{1}$$

It trains a neural network v_θ to match the velocity field and then solves the ODE for a given boundary condition ϵ to obtain the flow. The flow matching loss function is as follows:

$$\min_\theta E_{x_0 \sim p(x_0), \epsilon \sim N(0,I), t \sim \text{ Uniform } (0,1)} \left\| (\epsilon - x_0) - v_\theta\left(x_t, t\right) \right\|_2^2. \tag{2}$$

3.2 1D Video Diffusion Tokenizer

To initiate our exploration of dynamic token length, and given the absence of prior baselines, we first design a 1D video diffusion tokenizer with a fixed number of tokens, which we refer to as the

fixed-length tokenizer. It consists of three main components: an encoder with latent tokens, a vector quantization (VQ) module, and a flow matching-based decoder.

Encoder and VQ module. Given a video $x_0 \in \mathbb{R}^{T\times H\times W\times 3}$, we apply a patchify operator with strides (p_t, p_h, p_w) to obtain $N = \frac{T}{p_t} \times \frac{H}{p_h} \times \frac{W}{p_w}$ patches. Each patch is linearly projected to a d-dimensional embedding and flattened into a 1D sequence $p \in \mathbb{R}^{N\times d}$. The flattened patch tokens p will be concatenated with M learnable latent tokens $q \in \mathbb{R}^{M\times d}$ and fed into the encoder as input. A ViT (Dosovitskiy et al., 2021) encoder $\mathcal{E}_\phi$ mixes information between q and p using bidirectional full attention. In the output of the encoder, we only retain the latent tokens. We then discretize the latent tokens through a VQ module (van den Oord et al., 2018). Formally, the encoded tokens $\mathbf{Z}$ and quantized tokens $\hat{\mathbf{Z}}$ is defined as:

$$\mathbf{Z} = \mathcal{E}_\phi(q, p) \in \mathbb{R}^{M\times d}, \tag{3}$$

$$\hat{\mathbf{Z}} = quant(\mathbf{Z}). \tag{4}$$

Diffusion decoder. Our decoder is a flow matching-based DiT (William and Xie, 2023) model. To obtain x_t, we corrupt input video x_0 using Eq. 1 with Gaussian noise $\epsilon \sim N(0, I)$. We then concatenate the noisy video tokens with the quantized tokens $\hat{\mathbf{Z}}$, which serve as the condition, and feed them into the decoder as input. Our tokenizer is optimized through the flow-matching objective as define in Eq. 2:

$$\mathcal{L}_{\text{fm}} = \mathbb{E}\left[\left\|(\epsilon - x_0) - D_\theta\big(x_t, quant(\mathcal{E}_\phi(q,p)), t\big)\right\|_2^2\right], \tag{5}$$

where D_θ represents the decoder, t is randomly sampled from a mode distribution following SD3 (Esser et al., 2024). We also use a perceptual distance d_{perc} (Johnson et al., 2016) to supervise the 1-step denoised prediction of the network via:

$$\mathcal{L}_{\text{perc}} = \mathbb{E}\left[d_{\text{perc}}\big(x_0, x_t - tD_\theta(x_t, quant(\mathcal{E}_\phi(q,p)), t)\big)\right]. \tag{6}$$

Finally, a commitment loss (van den Oord et al., 2018) of vector quantization $\mathcal{L}_{\text{vq}}$ is applied:

$$\mathcal{L}_{\text{vq}} \;=\; \|\mathbf{Z} - \text{sg}[\hat{\mathbf{Z}}]\|_2^2, \tag{7}$$

where sg is the stop-gradient operation. Our codebook is updated using exponential moving average (EMA) following (Razavi et al., 2019). Overall, our training loss for the fixed-length tokenizer is:

$$\mathcal{L}_{\text{rec}} \;=\; \mathcal{L}_{\text{fm}} \;+\; \lambda_{\text{perc}}\,\mathcal{L}_{\text{perc}} \;+\; \lambda_{\text{vq}}\,\mathcal{L}_{\text{vq}}, \tag{8}$$

with loss weights and further details given in Sec. 4. In the next section, we will elaborate on how we extend the fixed-length tokenizer to a dynamic-length one.

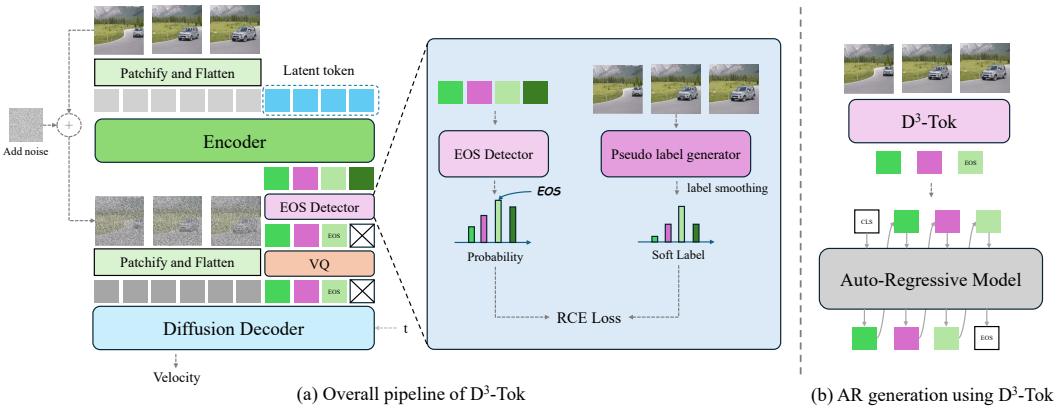
3.3 End-Of-Sequence (<EOS>) Detector

We aim to train the detector to autonomously predict a content-dependent truncation length. Drawing inspiration from the <EOS> (End-of-Sequence) token in Large Language Models (LLMs), we adapt this mechanism for our tokenizer by defining end-position prediction as a classification task. We employ an <EOS> detector to identify the termination point of the 1D token sequence. The detector estimates the probability of being the <EOS> for each token conditioned on the full sequence context. The token with the highest probability is selected as the <EOS>, while all subsequent tokens are discarded. Formally, given the encoded token sequence $\mathbf{Z}$:

$$p_{EOS}(\mathbf{Z}) = E_\psi(\mathbf{Z}), \tag{9}$$

where E_ψ is the <EOS> detector and $p_{EOS}(\mathbf{Z})$ is the probability distribution of each token being the <EOS>. At inference, we select $k = \arg\max_i p_{\text{EOS},i}(Z)$, where k denotes the number of retained tokens. We keep the prefix $\mathbf{Z}_{1:k}$ and mask all positions $j > k$.

However, unlike conventional classification tasks that rely on ground-truth labels for supervision, our <EOS> detector does not have explicit supervision. In the following section, we describe how we train the <EOS> detector.



(a) Overall pipeline of D³-Tok (b) AR generation using D³-Tok

Figure 2: **Overview of our approach.** (a) Our method is composed of a 1D video diffusion tokenizer and a <EOS> detector. The 1D tokenizer is composed of an encoder, a vector quantization (VQ) module, and a diffusion decoder. The <EOS> detector is a lightweight transformer that estimates the probability of each token being the <EOS> token. We jointly train the detector using a video-complexity loss to encourage token compression and a reconstruction loss to preserve reconstruction quality, enabling it to predict a content-dependent truncation length. (b) The encoded tokens can be directly used in Auto-Regressive (AR) models in the same way as LLMs.

3.4 <EOS> Detector Training Framework

Previous approaches like InfoTok (Haotian et al., 2025), formulate theoretic bounds for video token lengths. However, the number of tokens required by a video depends on implementation choices such as the tokenizer architecture, vector quantization, and decoder capacity. We therefore use a heuristic complexity estimate only as a noisy prior and learn a content-dependent correction through reconstruction feedback. The resulting prediction represents an empirical operating point rather than a theoretically optimal sequence length. We first describe the complexity prior and then introduce the detector objective.

Video Complexity Quantification. We consider two alternative proxies for video complexity: an H.264-based pixel-level proxy and a DINO-based semantic proxy. Our pixel-level method is based on modern video coding techniques (Dominguez et al., 2014; Sikora, 1997; Sullivan et al., 2012) which achieve high compression rates by removing inner- and inter-frame redundancy. Videos with higher compression ratios tend to exhibit simpler pixel-level structures, while those with lower compression ratios are typically more complex. Therefore, we adopt the reciprocal compression ratio of H.264 (Dominguez et al., 2014) as a quantitative proxy for video complexity. Specifically, given a video $x_0 \in \mathbb{R}^{T\times H\times W\times 3}$ and its H.264 encoded file with a size of m bytes, we define the complexity c as:

$$c = \frac{m}{T \times H \times W \times 3}, \tag{10}$$

where $T \times H \times W \times 3$ represents the bytes of raw pixels in the uncompressed RGB video. The video-wise pseudo label p is obtained by normalizing c using the following formula:

$$p = \text{sigmoid}(\frac{c - \mu_c}{\sigma_c}), \tag{11}$$

where μ_c and σ_c represents the mean and standard deviation of c in the whole training dataset, respectively. Given a maximum sequence length M, the prior-implied target length is $M \times p$, which we refer to as the hard pseudo-label.

We further incorporate a semantic-level quantification of video complexity while we empirically observe that supervision via semantic-level yields suboptimal performance compared to pixel-level. For the sake of brevity and to maintain focus on our primary architecture, the details of semantic-level labels are deferred to the Appendix B.

Label Smoothing. While our complexity proxy p provides an initial target, it serves merely as a noisy prior and is susceptible to heuristic bias. Empirically, we observe that strict adherence to p often

induces excessive token compression, yielding sub-optimal reconstruction quality (detailed in Sec. 4). To mitigate this bias and prevent the model from overfitting to heuristic inaccuracies (Song et al., 2022), we recast the hard pseudo-label into a soft probabilistic distribution. Assigning probability density to adjacent tokens explicitly enables the detector to account for ambiguity and effectively smooth the label noise. Formally, for the ith token in the token sequence, its probability to be the <EOS> token is:

$$\hat{p}_i = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(i - p \times M)^2}{2\sigma^2}\right), i < M \tag{12}$$

where M is the total number of tokens, σ controls the softness of the label.

Detector Optimization Objective. Even after label smoothing, standard cross-entropy may overfit residual errors in the heuristic prior. We therefore combine cross-entropy and reverse cross-entropy (RCE) (Wang et al., 2019) with a reconstruction term. The classification terms provide a content-dependent compression signal around the noisy prior, while $\mathcal{L}_{\text{rec}}$ penalizes truncation that degrades visual fidelity. Their joint optimization yields a learned trade-off between sequence length and reconstruction quality. The composite loss function for the <EOS> detector is formally defined as:

$$\mathcal{L}_{\text{EOS}} = \lambda_{\text{ce}} \sum_{i=1}^{M} -\hat{p}_i \log p_{\text{EOS}}(\mathbf{Z}_i) + \lambda_{\text{rce}} \sum_{i=1}^{M} -p_{\text{EOS}}(\mathbf{Z}_i) \log \hat{p}_i + \lambda_{\text{re}} \mathcal{L}_{\text{rec}}, \tag{13}$$

where λ_{ce}, λ_{rce} and λ_{re} are hyperparameters, $\mathcal{L}_{\text{rec}}$ is defined in Eq. 8.

3.5 Two Stage Training

Our training proceeds in two stages. In the first stage, we train the fixed-length tokenizer using $\mathcal{L}_{\text{rec}}$ in Eq. 8. In the second stage, we jointly optimize the tokenizer and the <EOS> detector. To preserve hard truncation in the forward pass while allowing the reconstruction loss to update the detector, we use a straight-through estimator (STE). Let $\mathbf{p} = \text{softmax}(E_\psi(Z))$ be the predicted distribution over the number of retained tokens, and let $\mathbf{h} = \text{onehot}(\arg\max_i p_i)$ denote the corresponding hard length decision. During training, we define $\tilde{\mathbf{h}} = \mathbf{h} + \mathbf{p} - \text{sg}(\mathbf{p})$, and condition the decoder on the masked tokens $\mathbf{m} \odot \hat{Z}, m_j = \sum_{k=j}^{M} \tilde{h}_k$. Since $\tilde{\mathbf{h}} = \mathbf{h}$ in the forward pass, the model always uses a hard prefix, while the backward pass retains the gradient path from $\mathcal{L}_{\text{rec}}$ to the <EOS> detector. When optimizing $\mathcal{L}_{\text{EOS}}$, we stop its gradients to the tokenizer parameters while retaining gradients to the detector through the mask.

4 Experiments

4.1 Experimental Setting

Dataset. Our training and evaluation is conducted under UCF-101 (Soomro et al., 2012) and Kinetics-600 (K600) (Carreira et al., 2018). For all videos, we uniformly sample 16 frames and resize them to resolutions of 128×128 and 256×256, respectively.

Implementation Details. We patchify all the input video using setting: $p_t = 4$, $p_h = 8$, $p_w = 8$. The number of latent tokens is set to 1024. For the VQ module, we set up the vanilla vector quantization (van den Oord et al., 2018) with a codebook size of 8192 and a dimension of 16. The <EOS> detector is a lightweight, 33M-parameter transformer that performs global attention across 1024 tokens, ensuring that the probability at each position is conditioned on the entire sequence. Label smoothing hyperparameter σ is set to 100. In the first stage, we train a tokenizer with a fixed token length of 1024 for 100 epochs. In the second stage, we jointly train the <EOS> detector and the tokenizer for 50 epochs. The batch size is set to 64 for both stages. All training is conducted on 8 NVIDIA H100 GPUs. AdamW (Loshchilov and Hutter, 2019) is employed as the optimizer, with a learning rate of 1×10^{-4} in the first stage and 4×10^{-5} in the second stage.

To evaluate the scaling behavior, we train two model variants, D^3-Tok-L and D^3-Tok-XL, with 175M and 368M tokenizer parameters, respectively. The two variants differ only in model capacity. Unless otherwise specified, D^3-Tok refers to the L variant.

Table 2: **Quantitative comparison across diverse benchmarks.** rFVD and gFVD for the UCF-101 (UCF) and Kinetics-600 (K600) datasets are reported. Gray text denotes models trained at 256×256 resolution.

Method	Token Type	#Params	#Tokens	rFVD ↓		gFVD ↓	
				UCF	K600	UCF	K600
SweetTok (Tan et al., 2025)	1D	128M	1280	20	25	65	–
D^3-Tok-L (Ours)	1D	175M	933 (Avg.)	18	22	57	6.5
D^3-Tok-XL (Ours)	1D	368M	961 (Avg.)	**15**	**18**	**52**	**5.8**
CogVideo (Hong et al., 2022)	3D	–	2065	–	–	626	109.2
TATS (Ge et al., 2022)	3D	32M	1024	162	–	332	–
OmniTokenizer (Wang et al., 2024b)	3D	82.2M	1280	42	26	191	32.9
MAGVIT (Yu et al., 2023)	3D	158M	1024	25	–	265	–
ElasticTok (Yan et al., 2025)	1D	200M	2048	93	–	–	–
AdapTok (Li et al., 2025)	1D	259M	2048	28	–	67	11
LARP-L-Long (Wang et al., 2025b)	1D	173M	1024	20	13	102	6.2
D^3-Tok-L (Ours)	1D	175M	706 (Avg.)	18	10.2	52	4.9
D^3-Tok-XL (Ours)	1D	368M	742 (Avg.)	**16**	**9.6**	**47**	**4.1**

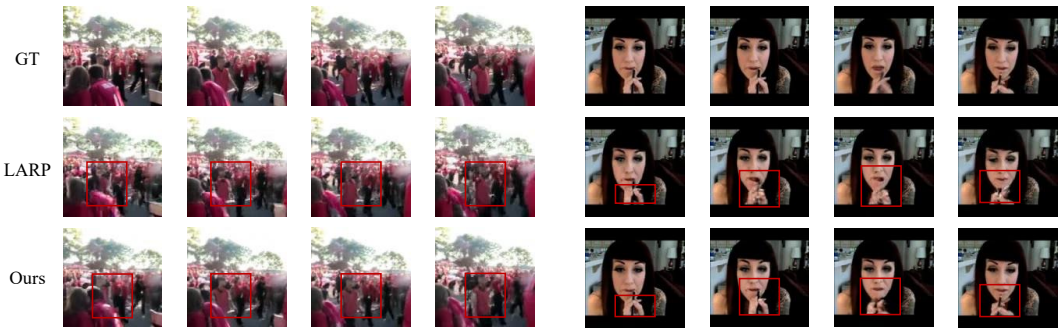


Figure 3: **Qualitative comparison between our method and LARP (Wang et al., 2025b).**

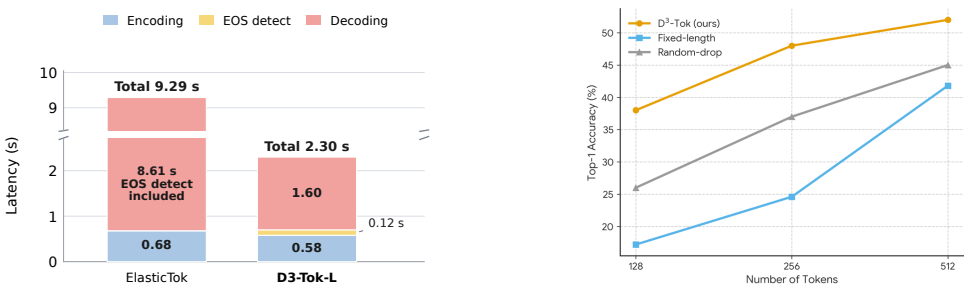
For AR video generation, we employ a Llama model (Touvron et al., 2023) with 327M parameters as our AR generative model following LARP (Wang et al., 2025b). A class token <CLS> and end-of-sequence token <EOS> are employed in the class-conditional generation task on UCF-101 (Soomro et al., 2012) and frame prediction task on K600. All tokens following the <EOS> token are replaced with padding tokens and excluded from loss computation. A comprehensive model and training details are provided in the Appendix A.

Baselines. We compare our method with 3D grid based methods including CogVideo (Hong et al., 2022), TATS (Ge et al., 2022), MAGVIT (Yu et al., 2023), OmniTokenizer (Wang et al., 2024b) and 1D tokenizer methods, including LARP (Wang et al., 2025b) and SweetTok (Tan et al., 2025). Note that SweetTok is trained under the resolution of 256×256 and its results are compared with our model trained at the same resolution. We also compare our method with search-based dynamic length video tokenizer, including AdapTok (Li et al., 2025) and ElasticTok (Yan et al., 2025)

Evaluation. For video reconstruction experiments, we adopt Reconstruction Frechet Video Distance (rFVD) (Thomas et al., 2018) as our metric. For video generation, we use Generation Frechet Video Distance (gFVD) metric. Total diffusion steps are set to 3 in all experiments and Classifier-Free Guidance (CFG) (Ho and Salimans, 2022) is not applied as it degrades reconstruction performance.

4.2 Video Reconstruction and Generation

We evaluate our tokenizer on video reconstruction and generation against recent 1D and 3D tokenizers. Tab. 2 reports the rFVD and gFVD results. Among 1D tokenizers, SweetTok (Tan et al., 2025)



(a) Efficiency comparison

(b) Linear probing results

Figure 4: **(a) Efficiency comparison.** We compare the inference latency between our method and ElasticTok. **(b) Linear probing results.** We perform linear probing on different token numbers to evaluate semantic representation.

achieves competitive gFVD but uses a fixed sequence length of 1280, resulting in redundancy for low-complexity videos. ElasticTok (Yan et al., 2025) and AdapTok (Li et al., 2025) reduce such redundancy through search-based strategies but yield suboptimal reconstruction quality. LARP (Wang et al., 2025b) improves generation with a prior model at the cost of reconstruction fidelity.

In contrast, D^3-Tok achieves the lowest rFVD and gFVD across resolutions on both datasets while maintaining the highest compression rate. These results demonstrate that dynamic-length tokenization effectively balances compactness and reconstruction fidelity. By eliminating redundant visual tokens, it also reduces the modeling burden on the AR generator and improves generation quality.

Compared with 3D tokenizers such as TATS (Ge et al., 2022), MAGVIT (Yu et al., 2023), and CogVideo (Hong et al., 2022), which use fixed spatiotemporal grids and much larger codebooks, D^3-Tok achieves substantially lower rFVD with fewer tokens. This suggests that fixed-grid representations introduce considerable redundancy. Its superior gFVD further demonstrates that compact 1D token sequences are better suited to AR generation. Additional results on PSNR, SSIM, LPIPS, and text-to-video generation are provided in Appendix C.

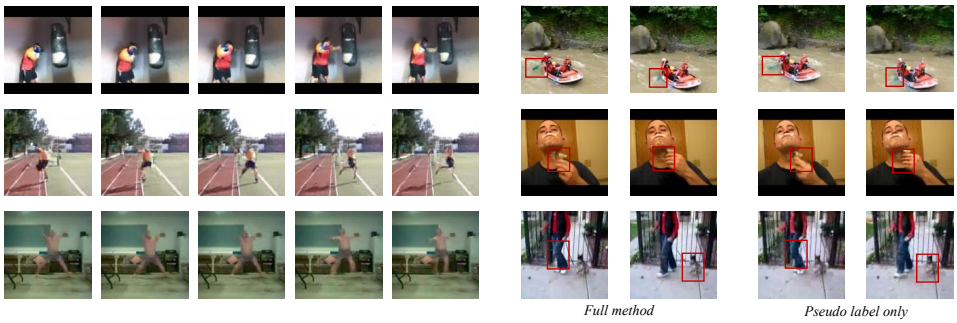
Fig. 3 compares D^3-Tok with LARP (Wang et al., 2025b), showing that our method produces fewer artifacts and sharper fine-grained details. Fig. 5(a) presents class-conditional AR video generation. With an AR-specific <EOS> token, the model can terminate adaptively instead of predicting unnecessary tokens, leading to improved synthesis quality.

4.3 Single-pass efficiency

We measure inference latency between our method and ElasticTok (Yan et al., 2025) on the same GPU with batch size 1. For ElasticTok, we search lengths from 256 in increments of 256 and stop when the rFVD improvement over the previous candidate is below 5%. Our method requires 0.58 s for encoding and 0.12 s for <EOS> prediction. The detector therefore accounts for 17% of encoding latency and only 5% of the total latency. D^3-Tok-L achieves approximately 4× higher inference efficiency than ElasticTok in terms of end-to-end latency.

4.4 Semantic Representation Analysis

We evaluate the learned semantic properties via linear probing for action recognition on UCF-101. As depicted in Fig. 4 (b), we compare D^3-Tok against fixed-length and random-drop counterparts (similar to VideoFlexTok (Atanov et al., 2026)) under varying token budgets. D^3-Tok consistently outperforms all baselines across all lengths. This demonstrates that fixed-length struggles to learn representations due to redundant tokens, and random drop has limited semantics due to over- and under-dropping. Conversely, driven by reconstruction-aware content-adaptive compression, our method leads to more semantically compact and discriminative representations.



(a) Class-conditional video generation.

(b) Comparison between our full method and pseudo label only.

Figure 5: (a) Class-conditional AR video generation in UCF-101 dataset. (b) Visual comparison between our full method and only use pseudo label.

Table 3: **Ablation study.** We ablate different components of our method.

Method	gFVD	rFVD	Len.	Method	gFVD	rFVD	Len.
Fixed length	132	**17**	1024	Hard h.264 + CE	111	32	697
Hard DINO label	167	44	702	Soft h.264 + CE	91	30	670
Soft DINO + RCE + $\mathcal{L}_{\text{rec}}$	115	29	914	Soft h.264 + RCE	76	24	692
Hard h.264 label	105	36	681	Soft h.264 + RCE + $\mathcal{L}_{\text{rec}}$	**52**	<u>18</u>	706

4.5 Ablation Study

We conduct a series of ablation experiments to analyze the effectiveness of different components in our dynamic-length method, as shown in Tab. 3.

Effect of Label Source. We first compare pseudo labels derived from DINO features and H.264 compression ratios. The former captures semantic dispersion, while the latter reflects pixel-level complexity. Using H.264-based labels leads to better reconstruction and generation quality, indicating that pixel-based complexity estimation provides more precise guidance for token-length supervision.

Effect of Label Smoothing. We then convert the hard H.264 labels into soft distributions to prevent <EOS> detector overfitting on hard labels. The superior performance of soft labels suggests that the hard labels contain considerable noise; however, the lack of guidance on the optimal length results in suboptimal performance.

Effect of Loss Function. We replace the standard CE loss with the RCE loss (Wang et al., 2019) to further enhance learning under noisy supervision. The RCE loss encourages the model to trust its own predictions rather than relying solely on the pseudo labels; however, the H.264 pseudo-label merely provides a prior distribution based on pixel redundancy, whereas $\mathcal{L}_{rec}$ establishes the lower bound required to maintain visual fidelity and achieves the best performance. We also provide the visual comparison between our full method and using the pseudo labels only as shown in Fig. 5. It demonstrates that the failure cases using pseudo labels are mitigated by using our method.

5 Conclusion

In this work, we present D^3-Tok, the first single-pass dynamic-length 1D video diffusion tokenizer that removes the need for inference-time search By introducing a lightweight <EOS> detector to the 1D tokenizer, our method adaptively allocates tokens according to visual complexity while discarding redundant tokens. This design achieves state-of-the-art performance with substantially fewer tokens, enhances semantic representation as evidenced by strong linear-probing results, and enables more effective AR video generation. Overall, D^3-Tok establishes a new framework that unifies high compression, dynamic-length flexibility, and semantic richness, providing a foundation for future multi-modal AR models.

AI USE STATEMENT

In this work, we did not use generative AI tools for any research tasks for which disclosure is required, including generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims or assisting with proofs, proposing or refining hypotheses, designing research methodologies or experiments, implementing methods, translating text, cleaning or reformatting datasets, conducting qualitative or thematic data analysis, or interpreting results. Any remaining required-disclosure tasks were not applicable to this work.

We used generative AI tools only to assist with drafting and revising portions of the manuscript and to improve its grammar, clarity, and readability. All AI-assisted text was critically reviewed and edited by the authors. The authors verified that the final text accurately reflects the research methods, results, and conclusions, and that all claims and citations are supported by the underlying work and cited sources. Generative AI tools were not used to produce research ideas, scientific claims, experimental results, or citations. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work adheres to the Code of Ethics. Our study does not involve human subjects, sensitive personal data, or applications with foreseeable harmful consequences. All datasets used are publicly available and described in detail in Section 4, with careful consideration of potential biases and fairness concerns. The proposed methods are intended solely for academic research, and we have taken steps to ensure compliance with ethical research standards and legal requirements.

REPRODUCIBILITY STATEMENT

We have taken several steps to ensure the reproducibility of our work. Section 4 provides detailed descriptions of the model architecture, training objectives, and evaluation protocols. Additional implementation details, hyperparameter settings are included in the Appendix A, while the supplementary materials contain further results.

REFERENCES

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba, Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Andrei Atanov, Jesse Allardice, Roman Bachmann, Oğuzhan Fatih Kar, R. Devon Hjelm, David Griffiths, Peter Fu, Afshin Dehghan, and Amir Zamir. Videoflextok: Flexible-length coarse-to-fine video tokenization. *arXiv preprint arXiv:2604.12887*, 2026.

Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. Flextok: Resampling images into 1d token sequences of flexible length. *arXiv preprint arXiv:2502.13967*, 2025.

L. Lao Beyer, T. Li, X. Chen, S. Karaman, and K. He. Highly compressed tokenizer can generate without training. *arXiv preprint arXiv:2506.08257*, 2025.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020.

Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018.

Chameleon. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. doi: 10.48550/arXiv.2405.09818. URL https://github.com/facebookresearch/chameleon.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. *arXiv preprint arXiv:2202.04200*, 2022.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Humberto de Jesus Ochoa Dominguez, Osslan Osiris Vergara Villegas, Vianey Guadalupe Cruz Sanchez, Efren David Gutierrez Casas, and K.R. Rao. The h.264 video coding standard. *IEEE Potentials*, 2014.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2021.

Shivam Duggal, Phillip Isola, Antonio Torralba, and William T. Freeman. Adaptive length image tokenization via recurrent allocation. *arXiv preprint arXiv:2411.02393*, 2024.

Shivam Duggal, Sanghyun Byun, William T. Freeman, Antonio Torralba, and Phillip Isola. Single-pass adaptive image tokenization for minimum program search. *arXiv preprint arXiv:2507.07995*, 2025.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *arXiv preprint arXiv:2012.09841*, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024.

Lijie Fan, Tianhong Li, Siyang Qin, Yuanzhen Li, Chen Sun, Michael Rubinstein, Deqing Sun, Kaiming He, and Yonglong Tian. Fluid: Scaling autoregressive text-to-image generative models with continuous tokens. *arXiv preprint arXiv:2410.13863*, 2024.

Ziteng Gao and Mike Zheng Shou. D-ar: Diffusion via autoregressive models. *arXiv preprint arXiv:2505.23660*, 2025.

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. *arXiv preprint arXiv:2204.03638*, 2022.

Google Gemini. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Ye Haotian, Qiyuan He, Jiaqi Han, Puheng Li, Jiaojiao Fan, Zekun Hao, and Fitsum Reda et al. Infotok: Adaptive discrete video tokenizer via information-theoretic compression. *arXiv preprint arXiv:2512.16975*, 2025.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

Su J, Ahmed M, Lu Y, and et al. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024.

Yang Jin, Zhicheng Sun, Kun Xu, Kun Xu, Liwei Chen, Hao Jiang, Quzhe Huang, Chengru Song, Yuliang Liu, Di Zhang, Yang Song, Kun Gai, and Yadong Mu. Video-lavit: Unified video-language pre-training with decoupled visual-motional tokenization. *arXiv preprint arXiv:2402.03161*, 2024a.

Yang Jin, Kun Xu, Kun Xu, Liwei Chen, Chao Liao, Jianchao Tan, Quzhe Huang, Bin Chen, Chenyi Lei, An Liu, Chengru Song, Xiaoqiang Lei, Di Zhang, Wenwu Ou, Kun Gai, and Yadong Mu. Unified language-vision pretraining in llm with dynamic discrete visual tokenization. *arXiv preprint arXiv:2309.04669*, 2024b.

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

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. *arXiv preprint arXiv:1603.08155*, 2016.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arxiv:1312.6114*, 2013.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, Krishna Somandepalli, Hassan Akbari, Yair Alon, Yong Cheng, Josh Dillon, Agrim Gupta, Meera Hahn, Anja Hauth, David Hendon, Alonso Martinez, David Minnen, Mikhail Sirotenko, Kihyuk Sohn, Xuan Yang, Hartwig Adam, Ming-Hsuan Yang, Irfan Essa, Huisheng Wang, David A. Ross, Bryan Seybold, and Lu Jiang. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2024.

Yan Li, Changyao Tian, Renqiu Xia, Ning Liao, Weiwei Guo, Junchi Yan, Hongsheng Li, Jifeng Dai, Hao Li, and Xue Yang. Learning adaptive and temporally causal video tokenization in a 1d latent space. *arXiv preprint arXiv:2505.17011*, 2025.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2024.

Meta Llama3. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2019.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2024.

Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, Ji Hou, and Saining Xie. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.

Qwen. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *arXiv preprint arXiv:1906.00446*, 2019.

Kyle Sargent, Kyle Hsu, Justin Johnson, Li Fei-Fei, and Jiajun Wu. Flow to the mode: Mode-seeking diffusion autoencoders for state-of-the-art image tokenization. *arXiv preprint arXiv:2503.11056*, 2025.

T. Sikora. Mpeg digital video-coding standards. *IEEE Signal Processing Magazine*, 1997.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *arXiv preprint arXiv:2007.08199*, 2022.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 2012.

Zhentao Tan, Ben Xue, Jian Jia, Junhao Wang, Wencai Ye, Shaoyun Shi, Mingjie Sun, Wenjin Wu, Quan Chen, and Peng Jiang. Sweettok: Semantic-aware spatial-temporal tokenizer for compact video discretization. *arXiv preprint arXiv:2412.10443*, 2025.

Unterthiner Thomas, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Xiong Tianwei, Jun Hao Liew, Zilong Huang, Zhijie Lin, Jiashi Feng, and Xihui Liu. Evatok: Adaptive length video tokenization for efficient visual autoregressive generation. *arXiv preprint arXiv:2603.12267*, 2026.

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

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *arXiv preprint arXiv:1711.00937*, 2018.

Bohan Wang, Zhongqi Yue, Fengda Zhang, Shuo Chen, Li'an Bi, Junzhe Zhang, Xue Song, Kennard Yanting Chan, Jiachun Pan, Weijia Wu, Mingze Zhou, Wang Lin, Kaihang Pan, Saining Zhang, Liyu Jia, Wentao Hu, Wei Zhao, and Hanwang Zhang. Selftok: Discrete visual tokens of autoregression, by diffusion, and for reasoning. *arXiv preprint arXiv:2505.07538*, 2025a.

Hanyu Wang, Saksham Suri, Yixuan Ren, Hao Chen, and Abhinav Shrivastava. Larp: Tokenizing videos with a learned autoregressive generative prior. *arXiv preprint arXiv:2410.21264*, 2025b.

Junke Wang, Dongdong Chen, Chong Luo, Bo He, Lu Yuan, Zuxuan Wu, and Yu-Gang Jiang. Omnivid: A generative framework for universal video understanding. *arXiv preprint arXiv:2403.17935*, 2024a.

Junke Wang, Yi Jiang, Zehuan Yuan, Binyue Peng, Zuxuan Wu, and Yu-Gang Jiang. Omnitokenizer: A joint image-video tokenizer for visual generation. *arXiv preprint arXiv:2406.09399*, 2024b.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. *arXiv preprint arXiv:1908.06112*, 2019.

Yuqing Wang, Tianwei Xiong, Daquan Zhou, Zhijie Lin, Yang Zhao, Bingyi Kang, Jiashi Feng, and Xihui Liu. Loong: Generating minute-level long videos with autoregressive language models. *arXiv preprint arxiv:2410.02757*, 2024c.

Peebles William and Saining Xie. Scalable diffusion models with transformers. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, Song Han, and Yao Lu. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2025.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2025a.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025b.

Liu Xingchao, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Wilson Yan, Volodymyr Mnih, Aleksandra Faust, Matei Zaharia, Pieter Abbeel, and Hao Liu. Elastictok: Adaptive tokenization for image and video. *arXiv preprint arXiv:2410.08368*, 2025.

Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G. Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, and Lu Jiang. Magvit: Masked generative video transformer. *arXiv preprint arXiv:2212.05199*, 2023.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *arXiv preprint arXiv:2406.07550*, 2024.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *arXiv preprint arXiv:2303.15343*, 2023.

David Junhao Zhang, Jay Zhangjie Wu, Jia-Wei Liu, Rui Zhao, Lingmin Ran, Yuchao Gu, Difei Gao, and Mike Zheng Shou. Show-1: Marrying pixel and latent diffusion models for text-to-video generation. *arXiv preprint arxiv:2309.15818*, 2023.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arxiv:2211.11018*, 2022.

Appendix

A Additional training details

We provide additional details for the implementation details described in Sec. 4. Our D^3-Tok-L configs are shown in Tab. S1. Our encoder is a plain transformer with standard sin&cos positional encoding (PE), exactlt the same as the encoder used in LARP (Wang et al., 2025b). The dimension of the decoder is larger than that of the encoder. We empirically find that when the decoder dimension is no larger than the encoder dimension, the outputs contain noticeable residual noise. Similar observations have also been reported in RAE (Zheng et al., 2025). Our decoder employs dual PEs: Sin&Cos PE is applied to input visual tokens and discreate tokens. 3D RoPE (J et al., 2024) is applied to the Query and Key of visual tokens in every attention layer. The <EOS> detector takes 1D token sequence as input so 1D RoPE is applied. Our vector quantization (VQ) module uses a vanilla codebook (van den Oord et al., 2018) and we use cosine distance to measure the distance between input features and codebook. The codebook is optimized using exponential moving average (EMA) and the decay rate is 0.99. Entropy loss is not applied in our method. Our D^3-Tok-XL model differs only in the decoder: we increase the number of attention heads to 18 and the hidden dimension to 1,152. We also scale the number of layers to 16.

In the first stage, we use $\mathcal{L}_{\text{rec}}$ defined in Eq. 8 to train the tokenizer, consisting of a encoder, VQ module and a decoder. The loss weight hyperparameter for $\mathcal{L}_{\text{rec}}$ is: $\lambda_{\text{perc}} = 0.5,\ \lambda_{\text{vq}} = 1.0$. In the second stage, we introduce <EOS> detector to the tokenizer trained in the first stage and an additional $\mathcal{L}_{\text{EOS}}$ is introduced. The loss weight hyperparameter for $\mathcal{L}_{\text{EOS}}$ is: $\lambda_{\text{ce}} = 0.01,\ \lambda_{\text{rce}} = 0.001,\ \lambda_{\text{re}} = 1.0$. The setting of $\mathcal{L}_{\text{rec}}$ is the same as in the first stage.

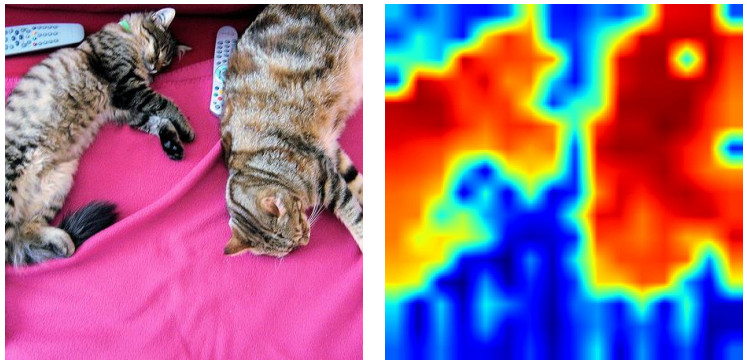
Table S1: Comprehensive configuration of the encoder, decoder, and <EOS> Detector.

Settings	Encoder	Decoder	<EOS> Detector
Module-Specific Settings			
Model Type	ViT	DiT	Transformer
Dimension	768	896	768
Num of Heads	12	14	12
Layers	12	12	4
Positional Encoding	Sin&Cos	Sin&Cos + RoPE	RoPE
QK Normalization	LayerNorm	LayerNorm	None
Shared Settings			
MLP Ratio		4.0	

B Semantic-level video complexity quantification

Self-supervised visual encoders (He et al., 2021; Assran et al., 2025; Oquab et al., 2024) produce features that capture rich semantic information. As an alternative label source in our ablation, we use DINOv2 (Oquab et al., 2024) features to quantify the semantic complexity of a video. We observe that, after applying Principal Component Analysis (PCA) to the DINO features, the dominant principal components (PCs) tend to activate the main subjects of the image, as shown in Fig. S1. Moreover, the degree of feature concentration reflects the intrinsic complexity of an area: low-complexity areas exhibit repetitive, highly similar feature patterns, whereas high-complexity objects are composed of more diverse and heterogeneous features. Based on this observation, we compute semantic complexity separately for the activated regions (main subject) and the non-activated regions (background). Given the ith frame x_0^i in video x_0, we first obtain its DINO feature F_0^i and apply PCA to it. Then, we keep the first n PCs v_n^i accounting for more than 95% of the total variance. The activation score for the jth token is:

$$a_0^{i,j} = \langle F_0^{i,j},\, v_n^i \rangle, \tag{1}$$



Original Image 1st PC Activated Area

Figure S1: **Visualization of area activated by DINO feature.** The first principal component (PC) of DINO features activates the main subject of the image.

Table S2: **Additional reconstruction metrics on UCF-101 and Kinetics-600.** We report PSNR, SSIM, and LPIPS at 128×128 resolution.

Method	UCF-101			Kinetics-600		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
TATS (Ge et al., 2022)	24.22	–	–	–	–	–
OmniTokenizer (Wang et al., 2024b)	27.76	–	0.073	–	–	0.095
MAGVIT (Yu et al., 2023)	22.00	0.7010	0.099	–	–	–
AdapTok (Li et al., 2025)	25.72	–	0.114	–	–	–
LARP-L-Long (Wang et al., 2025b)	30.78	–	0.120	31.17	0.8864	0.176
D^3-Tok-L (Ours)	30.84	0.9065	0.083	31.34	0.8933	0.091
D^3-Tok-XL (Ours)	**31.39**	**0.9177**	**0.064**	**31.48**	**0.8975**	**0.086**

where $\langle\rangle$ denotes the Euclidean inner product. We define the foreground region $\mathcal{R}^i_{\text{fg}}$ as tokens with activations $a_0^{i,j}$ above zero, and assign all remaining tokens to the background region $\mathcal{R}^i_{\text{bg}}$. Semantic complexity is measured as the feature dispersion within each region:

$$\mathcal{C}^i(\mathcal{R}) = \text{tr}\left(\frac{1}{|\mathcal{R}|}\sum_{j\in\mathcal{R}} \tilde{F}_0^{i,j}\left(\tilde{F}_0^{i,j}\right)^\top\right), \tag{2}$$

where tr denotes the matrix trace operator, i.e., the sum of its diagonal entries. It computes the total variance of feature responses within the region. We compute $\mathcal{C}^i_{\text{fg}}$ and $\mathcal{C}^i_{\text{bg}}$, and aggregate them to obtain the frame-level semantic complexity. The video-level semantic complexity c is computed as the mean semantic complexity across all frames. The pseudo label p is obtained by normalizing c using Eq. 11.

C Additional Metrics and Experiments

Additional metrics. We expand our evaluation to include PSNR, SSIM and LPIPS on UCF-101 and Kinetic-600 (K-600) dataset as shown in Tab. S2. D^3-Tok-L achieves competitive reconstruction quality with similarly sized tokenizers while using substantially fewer tokens. Scaling from D^3-Tok-L to D^3-Tok-XL consistently improves all three reconstruction metrics on both datasets.

Text-conditioned Video Generation. To evaluate whether D^3-Tok transfers beyond class-conditional generation, we pretrain D^3-Tok-XL and a 2.1B autoregressive model on a filtered subset of WebVid-10M containing 2.5M video-text pairs, and evaluate zero-shot text-to-video generation on MSR-VTT. As shown in Table S3, D^3-Tok achieves a CLIPSim of 0.3056 and an FVD of 256, ranking second

Table S3: **Zero-shot text-to-video generation on MSR-VTT.**

Method	CLIPSim ↑	FVD ↓
CogVideo (Hong et al., 2022)	0.2631	1294
MagicVideo (Zhou et al., 2022)	–	998
ModelScope (Jiuniu et al., 2023)	0.2930	550
Show-1 (Zhang et al., 2023)	**0.3072**	538
VideoPoet (Kondratyuk et al., 2024)	0.3049	**213**
Loong (Wang et al., 2024c)	0.2903	274
D^3-Tok-XL (Ours)	<u>0.3056</u>	<u>256</u>

among the compared methods on both metrics. Achieving this performance with the smallest model and the least amount of training data provides strong evidence that our method transfer effectively to text-conditioned video generation. Since the compared systems differ in architecture, training data, and optimization recipe, we interpret these results as evidence of transferability rather than a strictly controlled state-of-the-art comparison.