Adaptive Tokenisation via Temporal Redundancy Masking and Latent Inpainting

Kevin Dave[*1] Sai Aditya Patkuri[*1,2] Chhaya Kumar Das[*1] Gouranga Bala[1,3] Rajesh Kumar S A[1]
R. Venkatesh Babu[2]

[1]Phronetic AI [2]Vision and AI Lab (VAL), Indian Institute of Science
[3]Indian Institute of Technology Bombay [*]Equal contribution

BMVC 2026

Motivation

- Video tokenisers spend a fixed token budget on every clip. A static shot and a fight scene cost exactly the same.
- Adaptive tokenisers fix that, but pay a search or routing tax: ElasticTok runs a binary search ($\log_2 N$ decoder passes), InfoTok needs an extra full-rate decoder pass.
- **Is that overhead necessary?**

Key idea

The latent space of a *frozen* continuous tokeniser already exposes temporal redundancy. If a latent barely changed since the last kept frame, drop it — and inpaint it back before decoding.

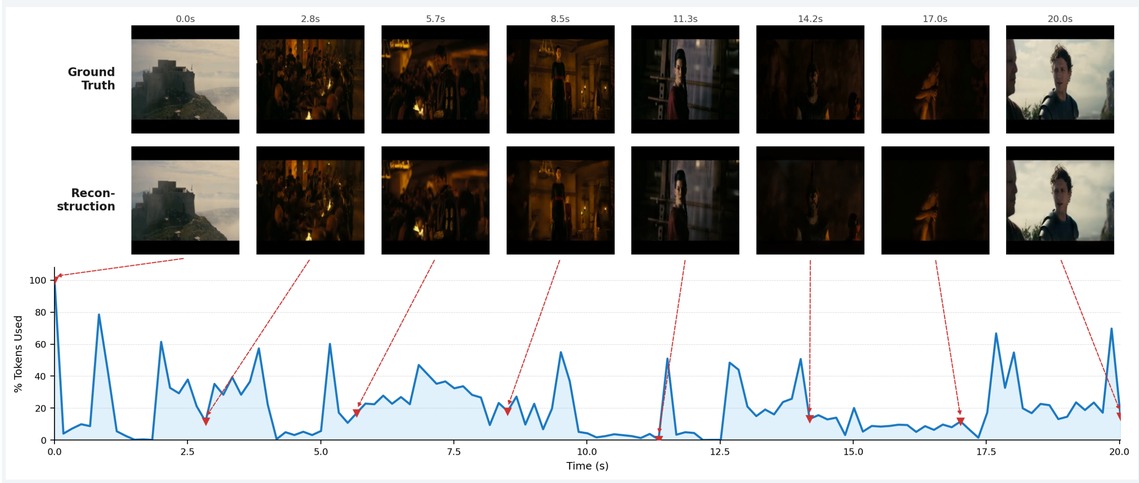
No router. No scorer. No extra decoder pass.

Inference cost (per clip, single A10G)

Method	NFE↓	secs↓	Speed-up↑
ElasticTok-CV (binary search)	$\log_2 N$	35.97	1×
InfoTok (discrete)	1	2.84	13×
Ours	**0**	**1.15**	31×

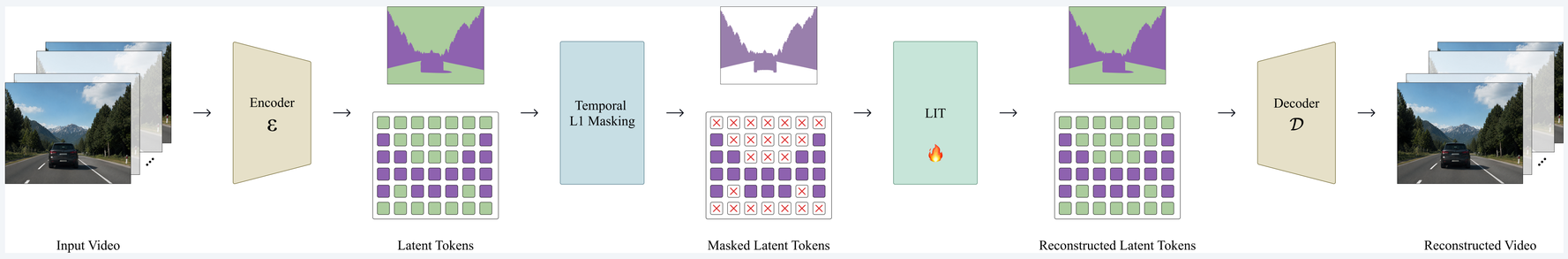
NFE counts the *extra* forward evaluations needed to set the budget. Ours needs none — the mask falls out of the encoder pass.

The budget follows the content



One clip, one fixed threshold: the retained fraction tracks the content — static intervals collapse towards zero, motion and scene cuts spike. Per-video drop rates span 5.15% to 86.10% on UCF-101.

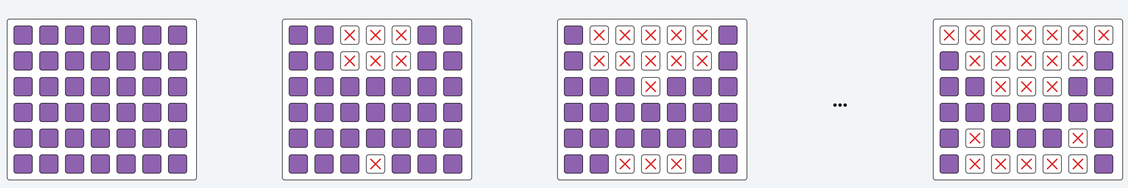
Method



Temporal-L1 masking — parameter free

For every spatial position (y, x), compare the latent against $\rho(y, x, t')$, the most recently kept temporal position:

$$\Delta(t', y, x) = \frac{1}{C}\sum_{c=1}^{C} \big| z[c, t', y, x] - z[c, \rho(y, x, t'), y, x] \big|, \qquad m = \mathbf{1}[\Delta \geq \tau]$$



Keep the position if the latent moved, drop it if it did not. τ is one scalar, calibrated once per backbone (0.3 Cosmos, 1.2 Omni).

Latent Inpainting Transformer (LIT)

- A frozen Cosmos-Tokenize1-CV $4{\times}8{\times}8$ encoder gives $z \in \mathbb{R}^{16\times9\times32\times32}$, i.e. $N = 9216$ tokens.
- LIT refills the dropped positions with interleaved spatial and temporal attention + 2D RoPE: $\approx 9\times$ cheaper than full 3-D attention, 2.7 M trainable parameters.
- Only LIT is trained, on $\mathcal{L} = \|x - \hat{x}\|_1 + \|z - \hat{z}\|_1$; encoder and decoder stay frozen.

Three properties that follow

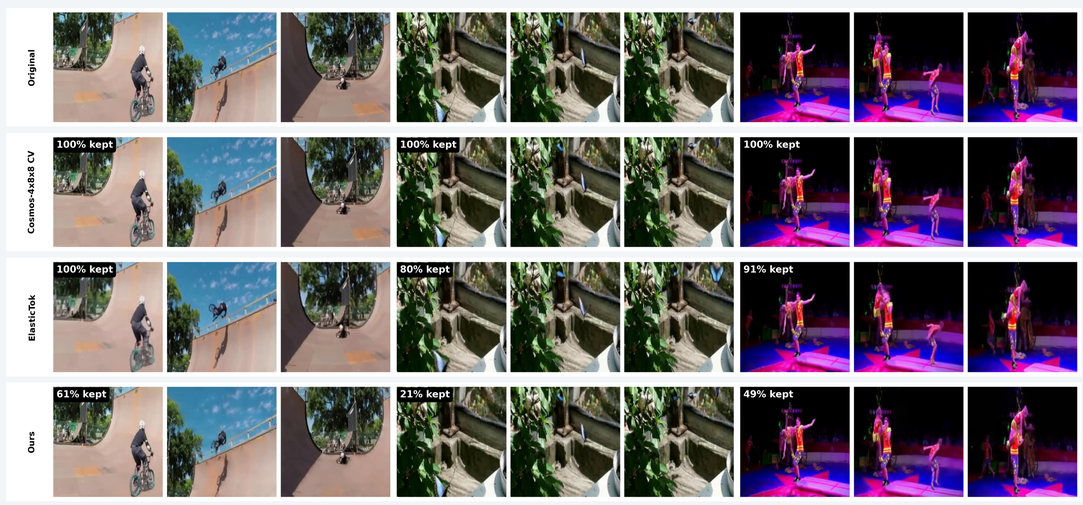
- **Parameter free** — no learned router, one interpretable scalar.
- **Single pass** — the mask needs only z: zero extra forward evaluations to allocate the budget.
- **Content adaptive** — the compression rate emerges from the video rather than being imposed on it.

Quantitative Results

Method	Keep↓	PSNR↑	SSIM↑	LPIPS↓	FVD↓
TokenBench (256×256)					
Cosmos-CV 4x8x8 (fixed)	100%	35.38	0.948	0.078	8.77
OmniTokenizer (fixed)	100%	27.05	0.892	0.128	57.83
ElasticTok-CV (MSE = 0.003)	40%	30.87	0.900	0.154	68.75
InfoTok (discrete)	32%	27.50	0.817	0.234	189.16
Ours (Cosmos, τ=0.3)	**32%**	**30.31**	**0.894**	**0.141**	**81.61**
Ours (Omni, τ=1.2)	**29%**	**23.88**	**0.790**	**0.245**	**168.82**
DAVIS (256×256)					
Cosmos-CV 4x8x8 (fixed)	100%	31.57	0.918	0.109	61.33
OmniTokenizer (fixed)	100%	24.49	0.837	0.194	244.81
ElasticTok-CV (MSE = 0.003)	82%	29.02	0.878	0.172	200.97
InfoTok (discrete)	62%	25.80	0.787	0.244	406.33
Ours (Cosmos, τ=0.3)	**62%**	**28.27**	**0.856**	**0.187**	**312.66**
Ours (Omni, τ=1.2)	**52%**	**22.50**	**0.736**	**0.293**	**468.15**

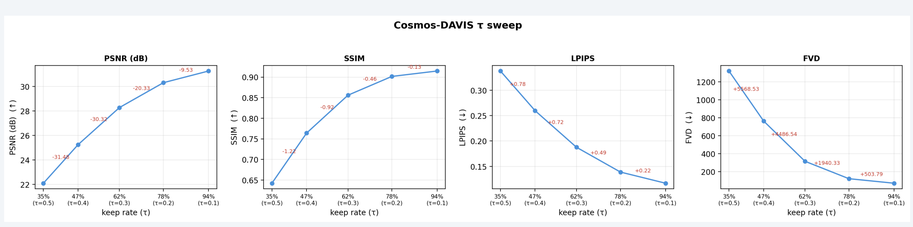
Baselines are matched to us — InfoTok to our emergent keep rate, ElasticTok-CV to a strict error threshold. At an *identical* budget we gain +2.81 dB / +2.47 dB over InfoTok.

Qualitative Results



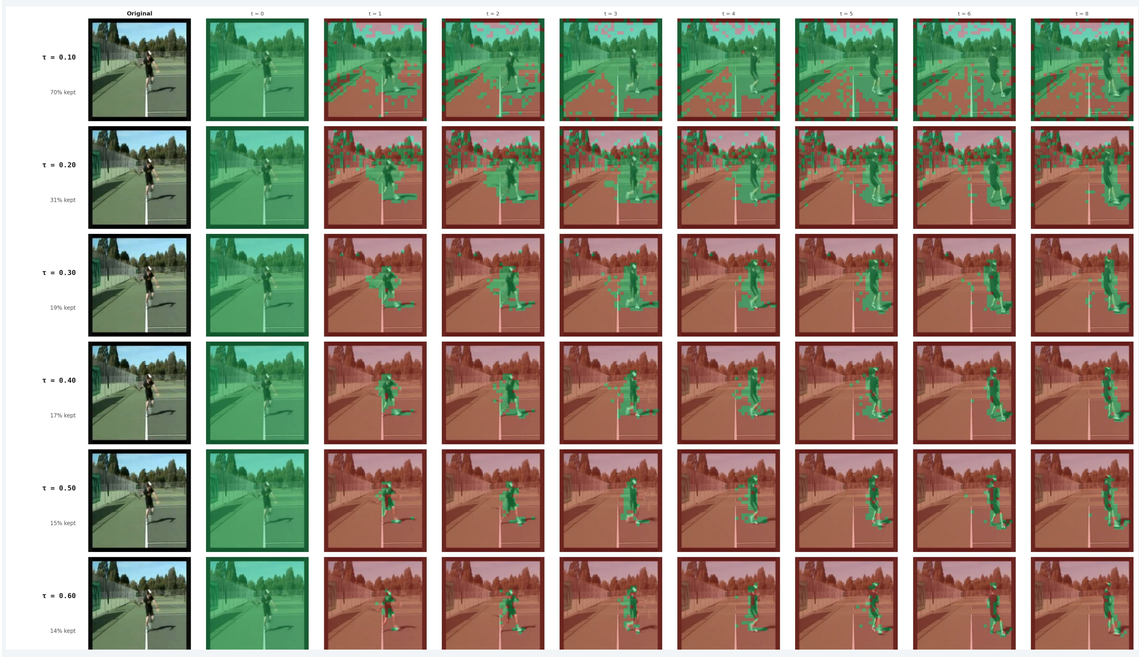
At 21–61% of the tokens we stay on par with ElasticTok at 80–100%, and with the full-rate Cosmos backbone.

Rate–quality: τ is the only knob



Quality degrades smoothly and monotonically as τ rises — no retraining, one calibration per backbone.

What τ does to the mask



Green kept, red dropped; one row per threshold, time running left to right. Raising τ peels the static court away first and keeps the moving player longest: 70% of tokens at τ=0.1, 14% at τ=0.6.

Ablations — DAVIS, every row at a 62% keep rate

Variant	PSNR↑	SSIM↑	LPIPS↓	FVD↓
(a) Masking criterion — ref. last-kept, fill copy				
Pixel-L1	25.59	0.775	0.261	629.25
Latent-L2	27.42	0.827	0.200	**477.90**
Latent-cosine	26.14	0.814	0.209	509.28
Latent-L1 (ours)	**27.54**	**0.829**	**0.198**	483.00
(b) Reference — crit. temporal-L1, fill LIT				
Previous frame + LIT	27.62	0.846	0.194	335.79
Last-kept + LIT (ours)	**28.27**	**0.856**	**0.187**	**312.66**
(c) Fill — crit. temporal-L1, ref. last-kept				
Copy last-kept	27.54	0.829	0.198	483.00
Linear interpolation	26.30	0.814	0.215	579.86
LIT (ours)	**28.27**	**0.856**	**0.187**	**312.66**

Fidelity comes from the *inpainting*: LIT gains +0.7 dB and −170 FVD over copying the last kept latent.