RIPE-MAMBASPIKE: RESOLUTION-INDEPENDENT SPIKING–STATE-SPACE INTERFACES FOR PARAMETER-EFFICIENT EVENT-BASED VISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Spiking–Mamba hybrids reach strong accuracy on event-based vision, but existing designs often require tens of millions of parameters. Much of that cost comes from how the spiking front-end is connected to the state-space backbone rather than from the hybrid architecture itself. In a representative model, a single resolution-dependent projection accounts for 33.55M of 36.25M parameters. To that end, we introduce RIPE-MambaSpike (**R**esolution-**I**ndependent, **P**arameter-**E**fficient), which replaces that projection with a hierarchical multi-resolution bridge of fixed channel width. Its deployed footprint is 0.870M parameters, constant at fixed time steps and widths across a 43× range of input areas. Reparameterized spiking stages, temporal decoupled modulation, and a dynamic convex-hull-bounded dual-stream membrane-potential attention preserve accuracy under this compact design. Result-wise, RIPE-MambaSpike is pareto-optimal on CIFAR10-DVS, N-Caltech101, and DailyDVS-200. Notably, on the 200-class DailyDVS-200, a scaled 8.04M configuration achieves 45.7% top-1 accuracy, the best reported spiking result on that benchmark, and outperforms prior spiking methods with 3.0–15.1× fewer parameters than dense ANNs. Overall, our findings demonstrate that competitive event-based recognition does not require resolution-dependent parameter growth.

1 INTRODUCTION

Spiking Neural Networks (SNNs) match event cameras representationally. Their temporal, event-driven dynamics align with the sparse asynchronous streams of Dynamic Vision Sensors. A second, more conditional advantage motivates much of the field: binary spikes let a layer replace multiply–accumulate with accumulate-only operations, which costs less energy on supporting silicon (Davies et al., 2018; Merolla et al., 2014; Horowitz, 2014). That advantage is contingent on every convolution operand being binary. Under standard sequential execution, many otherwise-spiking architectures do not meet that condition, ours included. On conventional GPU/edge runtimes we therefore make no energy, operation-count or latency claim (§5). An exact algebraic reparameterization recovers the binary-operand property for neuromorphic deployment (Proposition 3).

Two obstacles separate these advantages from deployment. Accuracy is the familiar one. SNNs have trailed dense ANNs on temporally rich benchmarks such as DVS-Gesture (Amir et al., 2017), CIFAR10-DVS (Li et al., 2017) and N-Caltech101 (Orchard et al., 2015). The less examined obstacle is *footprint*. Whatever the arithmetic, the model must fit on the device beside the sensor. On embedded platforms the parameter budget usually binds first, held as it is in on-chip SRAM or modest flash, and it binds absolutely: a model whose weights do not fit cannot run at all, however its arithmetic is counted. Event cameras ride on micro-aerial vehicles, robots and wearables (Gallego et al., 2022). Our weight storage is 1.7–16.1 MB depending on benchmark, against 153–294 MB for the models leading the largest event-action benchmarks, and that gap decides what can sit beside the sensor at all (§C.4). This paper addresses that second axis: *parameter*-efficiency.

Where the parameters actually go. A growing line of work closes this gap by hybridizing SNNs with high-capacity sequence models. Spiking–Mamba architectures (Qin & Liu, 2024a) are the most compelling, since Mamba's selective state-space scan (Gu & Dao, 2023) models long event

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.

streams in linear time with content-aware gating. These hybrids reach their accuracy at tens of millions of parameters, and the reason is narrower, and more fixable, than sheer size. Mamba-Spike's released model deploys $36.25\,\mathrm{M}$ parameters at 128^2. Of those, $33.55\,\mathrm{M}$ (92%) sit in a single layer, the projection carrying the spiking front-end into the scan, which flattens the feature map and projects it densely:

$$\texttt{Linear}(C\lfloor H/4\rfloor\lfloor W/4\rfloor, d). \tag{1}$$

Its parameter count is *quadratic in sensor resolution and independent of model capacity*. Measured by direct instantiation (Table 6), it grows $49\times$ from 34×34 to 224×224, reaching 97.4% of all parameters. The cost is an artifact of *how the SNN is joined to the SSM*, not an intrinsic price of the pairing. Spiking *transformer* baselines (Zhou et al., 2023; Yao et al., 2023; Lee et al., 2025a) are already flat at $2.57\,\mathrm{M}$ across resolutions, so the pathology is specific to flatten-then-project bridges rather than to attention models in general.

Thesis: the interface, not the capacity. That measurement gives the thesis its shape. The parameter cost of Spiking–Mamba hybrids is a property of the SNN–SSM interface, not of the capacity the task demands. An interface built on a fixed token width d_{m} removes that cost at no loss of accuracy: our deployed count is constant at $0.870\,\mathrm{M}$ across a $43\times$ range of input areas (from 34^2 to 224^2). Accuracy-competitive Spiking–Mamba modeling therefore does not need a large parameter budget, provided the architecture is co-designed under a strict *reparameterization-first* principle: pay the parameter cost at training time only, and recover at inference a compact single-kernel model with sparse activations and linear attention. The result is a sub-million-parameter deployed network, $\sim41\times$ smaller than Mamba-Spike, that nonetheless exceeds it on every shared benchmark. Each multi-branch convolution admits a closed-form fusion into one 3×3 kernel, generalizing RepVGG (Ding et al., 2021) to SNNs with BatchNorm-over-Time and spike non-linearities. Attention is linear in token count with a non-saturating feature map. The SNN–Mamba interface carries the front-end into the scan at a fixed width, so no state-space parameter tracks the sensor.

Contributions.

- **C1.** RIPE-MambaSpike, a compact hierarchical Spiking–Mamba hybrid. Reparameterized spiking backbones (Wang et al., 2024a) and spiking state-space hybrids (Qin & Liu, 2024a) have been explored separately. We claim their *combination*: a resolution-decoupled interface carrying bounded membrane-potential attention, co-designed under one reparameterization-first principle, at under one million deployed parameters (Table 4 sets the two architectures side by side).
- **C2.** *Temporal Decoupled Modulation (TDM)*, a per-channel learnable temporal filter with three analytical views (discrete-derivative, z-domain FIR, gradient-decoupling). Its commutation with the following convolution (Proposition 3) refactors the pair into two static kernels over binary operands, a result that is not specific to TDM.
- **C3.** *Dual-Stream Membrane-Potential Attention (DS-MPA)*, a linear attention with a non-negative ELU feature map between the membrane-potential (query/key) and prior-spike (value) streams, provably bounded by construction within the dynamic convex hull of its values (and in $[0, 1]$ for unprojected binary spikes), avoiding the gate-saturation problem of earlier spike-driven linear attention.
- **C4.** A *Multi-Resolution Spiking–State-Space Aggregation (MR-S3A) bridge* that fuses the spiking stages at a *fixed* model width, so the scan's parameter count is independent of $H\times W$, verified by instantiation across 34^2 to 224^2 ($43\times$ in area).
- **C5.** Across five event benchmarks, RIPE-MambaSpike beats a controlled Mamba-Spike reproduction wherever the two overlap, at 1.2–$41\times$ fewer parameters, and establishes a Pareto-optimal trade-off across the object and large-scale action benchmarks. DailyDVS-200, one of the largest event action benchmarks published and the hardest we evaluate, is where just one of the twelve methods we tabulate clears 50% top-1; there RIPE-MambaSpike is the strongest spiking method by 8.80 pp, within 6.25 pp of the state of the art at an order of magnitude fewer parameters.

2 Related Work

Vision SNNs. Direct-training vision SNNs split into spike-driven transformers (Zhou et al., 2023; Yao et al., 2023; 2024) and spiking residual networks (Fang et al., 2021a; Hu et al., 2024) with

learnable membrane constants (Fang et al., 2021b). TET loss (Deng et al., 2022) and RecDis (Guo et al., 2022) are near-universal training tools, orthogonal to our architecture and adopted in our recipe. The strongest current results on our benchmarks come from this transformer family. QK-Former (Zhou et al., 2024) reaches 84.0% on CIFAR10-DVS at 1.50 M, STAtten (Lee et al., 2025a) adds spatial–temporal attention, and the attention-free Spiking Wavelet Transformer (Fang et al., 2024) holds the best N-Caltech101 result we know of at 88.45%, setting the accuracy ceiling. RIPE-MambaSpike does not surpass them, and instead competes on the accuracy–parameter trade-off, since none provides an inference-time-fused reparameterized backbone and so each carries its full multi-branch parameter cost into deployment. SpikePool (Lee et al., 2025b) replaces self-attention with max-pooling ($80.20/81.62\%$ at 0.55 M, $82.70/85.01\%$ at 2.19 M on CIFAR10-DVS/N-Caltech101, Table 1); RIPE-MambaSpike beats both on N-Caltech101 at up to $2.5\times$ fewer parameters, trading 1.3 pp for a $2.5\times$ smaller footprint on CIFAR10-DVS.

State-space models and Spiking–Mamba hybrids. The SSM lineage (Gu et al., 2022; Gu & Dao, 2023) offers linear-time sequence modeling with content-aware gating, adapted to vision by Vim (Zhu et al., 2024) and VMamba (Liu et al., 2024). **Mamba-Spike** (Qin & Liu, 2024a), our primary baseline, couples a small spiking encoder to a Mamba stack. SpikMamba (Chen et al., 2024) adds window-based spiking linear attention tuned for fixed-resolution gesture sequences (99.01% on DVS-Gesture at 0.18 M, Table 1); RIPE-MambaSpike instead keeps its parameter count invariant across input scales via MR-S3A's fixed-width bridge. Vision SmolMamba (Bai et al., 2026) prunes tokens inside a spiking SSM. Its target is token count; ours is the *parameter* coupling between resolution and the interface, a different and complementary lever. We prune nothing (every MR-S3A token enters the scan), and our invariance follows instead from the bridge's fixed width and the length-independent $\mathcal{O}(d_{\mathrm{m}}^2)$ cost of Mamba blocks. A system could adopt both. FLAMES (Chakraborty & Mukhopadhyay, 2025) takes a complementary event-native SSM route, adapting HiPPO to inter-spike intervals rather than touching the interface. SPARTA (Jang & Kim, 2025) prioritizes tokens by spike timing, reaching $98.78/83.06\%$ on DVS-Gesture/CIFAR10-DVS at $\sim$13.8 M, $15\times$ our footprint. TP-Spikformer (Wei et al., 2026) prunes progressively. TS-SNN (Yu et al., 2025) is a FLOP-free temporal-shift operator in TDM's spirit; it holds an accumulate-only backbone by construction, where TDM recovers one by an exact commutation (§5). RIPE-MambaSpike occupies the sub-million-parameter corner these leave open.

Structural reparameterization. RepVGG (Ding et al., 2021), MobileOne (Vasu et al., 2023) and RepGhost (Chen et al., 2022) fuse multi-branch training graphs into single kernels. Closest is RTFormer (Wang et al., 2024a), which reparameterizes a spiking block and folds temporal BN into the neuron *threshold*. We instead retain BNTT as a per-timestep affine absorbed into a final scale–shift, target the sub-million regime, and pair reparameterization with a state-space rather than an attention backbone. Our claim is not that reparameterized spiking backbones are novel; we position RIPE-MambaSpike as the first to combine one with a hierarchical Spiking–Mamba interface at this scale (Appendix B). DS-MPA fixes the normalizer-collapse issue of linear attention (Katharopoulos et al., 2020; Choromanski et al., 2021) by construction (§4.3).

3 PRELIMINARIES

We accumulate a DVS event stream into T time bins to form $\mathbf{X} \in \mathbb{R}^{B\times T\times 2\times H\times W}$ (two polarity channels). A leaky integrate-and-fire (LIF) neuron integrates input x_t into a membrane potential $u_t = \beta(u_{t-1} - v_{\mathrm{th}} s_{t-1}) + x_t$ and fires $s_t = \Theta(u_t - v_{\mathrm{th}})$ with a soft reset. Because Θ' is a Dirac delta, we backpropagate through the ATan surrogate (Neftci et al., 2019), whose sharpness α we anneal from 2 to 4, or to 3 on CIFAR10-DVS and N-Caltech101 (Appendix C.6). That single α is also the temperature of the soft-spike sigmoid in the consistency loss (Section 4.5), and is distinct from the per-channel TDM coefficients λ and the scalar loss weights. We use per-timestep BatchNorm (BNTT) (Kim & Panda, 2021), which keeps T independent affine pairs. Mamba (Gu & Dao, 2023) evolves a hidden state via a selective, input-dependent state-space recurrence at $\mathcal{O}(Ld_{\mathrm{state}}d)$ cost (full discretization in Appendix B). We use the standard synaptic-operation (SOP) accounting convention (Horowitz, 2014): a spiking convolution with average pre-synaptic spike rate $\bar{r}$ contributes $\bar{r}\cdot\Phi$ accumulate-only (AC) operations *only when its operand is binary*, and dense multiply-accumulate (MAC) operations otherwise. In §5 we use this convention to state why, under standard sequential execution, the deployed backbone does not qualify for accumulate-only accounting, and we report

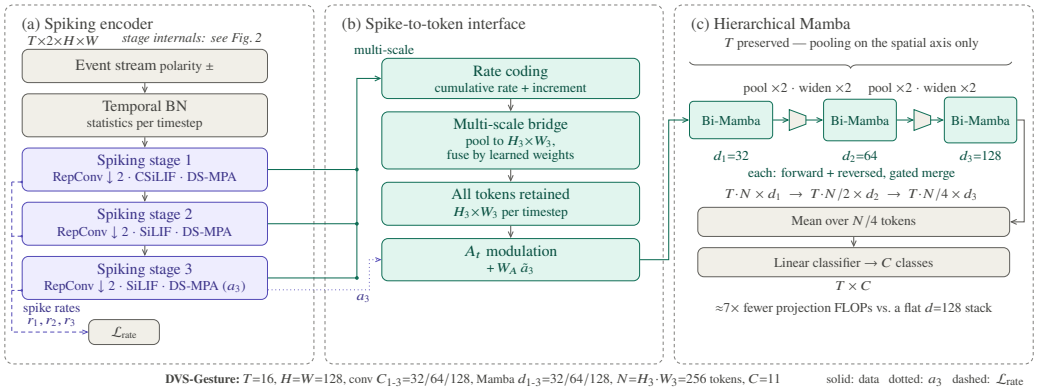


Figure 1: Overview of RIPE-MambaSpike (DVS-Gesture). **(a)** Three RepSpikeStage blocks (TDM + reparameterizable convolutions + DS-MPA, stride-2) emit the final attention map a_3 and firing rates. **(b)** Multi-scale rates/increments are fused (MR-S3A) into a fixed-width token sequence. **(c)** Three gated bidirectional Mamba blocks process the tokens, each transition pooling the spatial axis $\times 2$ and doubling width.

picojoule figures only for the dual-kernel reparameterization (Proposition 3) that recovers it for neuromorphic deployment.

4 METHOD: RIPE-MAMBASPIKE

RIPE-MambaSpike maps a 5-D event tensor to per-timestep class logits (averaged over T at evaluation, supervised per-timestep by TET) through: an entry BNTT; three *RepSpikeStage* blocks of channels $[32, 64, 128]$ with stride-2 downsampling and heterogeneous neurons (CSiLIF, SiLIF, SiLIF); a SpikeToRate+MR-S3A bridge into a d=32 token space widened to 128; a three-level hierarchical bidirectional Mamba backbone; and a linear head (Fig. 1). DS-MPA is embedded inside each stage.

4.1 REPARAMETERIZED SPIKING STAGE

At training time each backbone convolution is a sum of seven parallel branches (3×3, 1×1, 1×3, 3×1, 1×1→3×3, identity, avg-pool), each with its own BatchNorm (Fig. 4b). This sum fuses exactly to one 3×3 kernel at inference.

Proposition 1 (Branch fusion). *Let $s_b = \gamma_b/\sqrt{\sigma_b^2 + \varepsilon}$ be each branch's BN scale. With $W^\star = \sum_b \mathrm{Pad}_{3\times3}(s_b W_b)$ and $b^\star = \sum_b(\beta_b - \mu_b s_b)$, we have $\mathrm{RepConv}(x) \equiv W^\star * x + b^\star$ for all x.*

Proof and the 1×1→3×3 composition are in Appendix A. The seven branches have different kernel areas, so the saving is $26/9 \approx 2.9\times$ per unit. Branch BNs are standard 2D normalizations across the merged $B\times T$ batch and absorb completely into the single static 3×3 kernel $W^\star$ and bias $b^\star$; BNTT sits downstream of the sum as a per-timestep affine vector ($2Tc_{\text{out}}$ scalars), so fusion does not replicate weights across time. Each stage chains two TDM→RepConv→BNTT→neuron units with DS-MPA bridging the two neurons:

$$y_1 = \mathrm{BNTT}(\mathrm{RepConv}_1(\mathrm{TDM}_{\lambda_1}(x))), \quad (S_1, U_1) = \mathcal{N}_1(y_1), \tag{2}$$

$$y_2 = \mathrm{BNTT}(\mathrm{RepConv}_2(\mathrm{TDM}_{\lambda_2}(S_1))), \quad \hat{y}_2 = \text{DS-MPA}(U_1, S_1^{\text{prev}}, y_2),\ (S_2, U_2) = \mathcal{N}_2(\hat{y}_2). \tag{3}$$

4.2 TEMPORAL DECOUPLED MODULATION (TDM)

For a per-channel $\lambda \in \mathbb{R}^C$, TDM is the first-order filter $\mathrm{TDM}_\lambda(x)_t = (1 + \lambda_c)x_t - \lambda_c x_{t-1}$ (with x_0:=0), adding one scalar per input channel per insertion (322 total, 0.04% of the model). It admits three views, with full forms in Appendix B; we state the two load-bearing ones here. As a learnable discrete derivative $\tilde{x}_t = x_t + \lambda_c(x_t - x_{t-1})$, it is exactly the identity at λ=0, so zero-initialization makes TDM inert at iteration zero and training starts from the TDM-free network rather than a perturbed one. As a z-domain filter $H_c(z) = (1 + \lambda_c) - \lambda_c z^{-1}$, each channel learns

its own bandwidth: $|H_c(e^{j\omega})|^2 = 1 + 2\lambda_c(1+\lambda_c)(1-\cos\omega)$ is high-pass for $\lambda_c > 0$ (Prop. 4, Appendix B; proof in Appendix A). The deployed model learns exactly this depth profile. The first stage's coefficient reaches $\lambda \approx 0.72$, high-pass and motion-emphasizing, while the deepest stage settles near $\lambda \approx 0$, close to identity, consistent with early-pathway motion sensitivity. TDM also provides a surrogate-free temporal gradient path parallel to the spike path (gradient-decoupling lemma, Appendix A).

Heterogeneous neurons. Stage 1 uses CSiLIF, a complex-valued sigmoid LIF whose per-channel pole $\rho_c = -e^{\ell_c} + j\omega_c \in \mathbb{C}^-$ turns a real LIF's low-pass response into a damped oscillation, a learnable band-pass channel. The form $-e^{\ell_c}$ keeps it stable for every ℓ_c without clipping (readout $U = 2\,\mathrm{Re}(m)$; Appendix B). Stages 2–3 use the real-valued SiLIF. Neither neuron is our contribution: SiLIF follows PLIF (Fang et al., 2021b), CSiLIF the resonate-and-fire idea (Izhikevich, 2001). Our finding is that the assignment CSiLIF–SiLIF–SiLIF attains the highest accuracy of the eight sweep configurations (Appendix C), with resonance helping most at the high-resolution first stage.

4.3 Dual-Stream Membrane-Potential Attention

DS-MPA is a per-timestep linear attention with queries/keys from the current membrane U_1 and values from the previous spike tensor S_1^{prev}. With the non-negative feature map $\phi(z) = \mathrm{ELU}(z) + 1$,

$$A = \frac{\phi(Q)\,(\phi(K)^\top V)}{\phi(Q)\,(\phi(K)^\top \mathbf{1}) + \varepsilon} \in \mathbb{R}^{N\times C}, \tag{4}$$

$$\hat{X} = X + \gamma \odot (A \odot X), \tag{5}$$

with a per-channel gate γ (init 1). Dimensionally, the $N{=}HW$ spatial positions are the sequence axis and the C channels the feature axis. The key–value contraction $\phi(K)^\top V \in \mathbb{R}^{C\times C}$ is over the sequence axis at cost $\mathcal{O}(NC^2)$, so DS-MPA mixes *channels* conditioned on membrane content rather than tokens, and the normalizer keeps A bounded. Width does not inflate that $O(NC^2)$ term here, since N shrinks $16\times$ as C grows $4\times$. Isolated, DS-MPA is 7.9% of a step, 4.3% of parameters and 7.4% of the deployed count; a $3.3\times$ cheaper gate recovers 5.5% of step time and a $4\times$ low-rank cut (which provably retains the bound below) only $0.875 \to 0.827$ M, both against the 3.16 pp DS-MPA is worth on N-Caltech101, so we deploy the full operator (Tables 16 and 18, the latter untrained; pseudo-code in Fig. 6).

Proposition 2 (Boundedness)**.** *If $\phi \geq 0$ element-wise, then $A_{i,\cdot}$ is a convex combination of the* rows of V *contracted by $\eta_i = d_i/(d_i{+}\varepsilon) \in [0,1)$, so $\min(0, \min_j V_{j,c}) \leq A_{i,c} \leq \max(0, \max_j V_{j,c})$; unprojected binary V gives $0 \leq A_{i,c} \leq 1$.*

The output can therefore neither diverge nor be driven to a constant by a saturating scalar; there is none, and the bound is *dynamic* rather than a fixed $[0,1]$ interval (proof in Appendix A). It holds for any V and any parameterization of W_Q, W_K, W_V, so the reduced variants of Table 18 inherit it. We do not constrain W_V to make V non-negative. A parameter-matched sigmoid gate $A = \sigma(QK^\top)V$ instead saturates: $\sigma'(z) \to 0$ compounds with surrogate-gradient attenuation and starves the early stages, whereas $\phi'(z) > 0$ everywhere does not (derivation in Appendix B). We have not trained that control, so this argument is structural rather than measured. DS-MPA costs $\mathcal{O}(NC^2)$ time against $\mathcal{O}(N^2C)$ for softmax attention; learned attention maps and per-class accuracy are in Appendix C.12.

4.4 MR-S3A Bridge

Each stage's spikes are encoded as a 2-channel cumulative-rate+first-difference summary, pooled to the deepest resolution and fused by softmax-weighted multi-scale projection into the bridge token space. The bridge width d_{m} is a *fixed* hyperparameter, independent of $H{\times}W$. This is what decouples the deployed parameter count from resolution, since the Mamba blocks downstream are $\mathcal{O}(d_{\text{m}}^2)$ and independent of sequence length. All TH_3W_3 tokens enter the scan. We apply no token pruning, and the parameter invariance holds without it (Table 6). A learnable projection additionally injects the stage-3 DS-MPA attention map into the token sequence (pseudo-code in Fig. 7, exact modulation term in Appendix B).

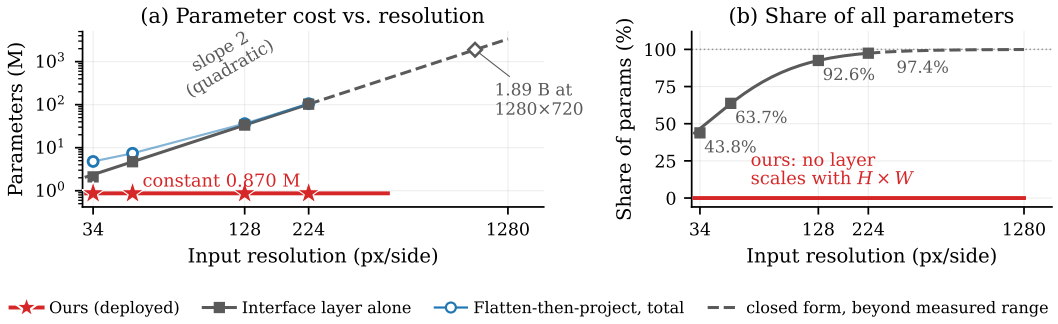


Figure 2: **The SNN→SSM interface's parameter cost is quadratic in sensor resolution; RIPE-MambaSpike's is constant.** Markers are *instantiated and counted* (Appendix C.1); the curve is Eq. (1), which reproduces all four exactly, dashed where extrapolated. At the 1280×720 of an HD event sensor that one layer reaches 1.89 B parameters (≈7 GiB `fp32`); RIPE-MambaSpike is tested flat at 0.870 M from 34^2 to 224^2.

4.5 Training Objective

The loss is $\mathcal{L} = \mathcal{L}_{\text{TET}} + \lambda_{\text{SGC}}\mathcal{L}_{\text{SGC}} + \lambda_{\text{L1}}\bar{r}$. Beyond TET (Deng et al., 2022), a Surrogate-Gradient Consistency term $\mathcal{L}_{\text{SGC}} = \text{MSE}(z^{\text{disc}}, z^{\text{cont}})$ penalizes the gap between the hard-spike pass and a soft-spike pass ($\Theta \to \sigma(\alpha\cdot)$) through shared weights (symmetric, not distillation; dual-branch schematic in Appendix B), and a small spike-rate L1 ($\lambda_{\text{L1}}{=}10^{-4}$) holds a depth-dependent operating point $r_\ell \in [0.015, 0.19]$. The full composite loss is used for DVS-Gesture and N-MNIST; CIFAR10-DVS and N-Caltech101 use a TET-only objective ($\lambda_{\text{SGC}}{=}\lambda_{\text{L1}}{=}0$), and DailyDVS-200 disables SGC while retaining the rate penalty (Table 9).

5 Parameter and Deployment Cost

Parameters. Each RepConv fuses from $26c_{\text{in}}c_{\text{out}}$ training weights to $9c_{\text{in}}c_{\text{out}}$ at inference (Prop. 1, derivation in Appendix B), a $26/9 \approx 2.89\times$ reduction per convolution unit. Across the full backbone that is 2.76×, 1.014 → 0.368 M, not a naive 7×. Fusion leaves the hierarchical Mamba, bridge and head untouched. The hierarchy is a parameter-economy device, not an operation-count one: a bidirectional block costs $\mathcal{O}(d^2)$, so a flat three-block stack at $d{=}128$ would cost 2.21× the hierarchy's 361,760 (1.31 vs. 0.875 M deployed), though at a *matched* budget it is not faster. We quote two deployed counts: 869,593 ≈ **0.870 M** with a 10-class head at $T{=}10$, and 875,122 ≈ **0.875 M** for DVS-Gesture's 11-class head at $T{=}16$, down from 1.522 M at training time by branch fusion (Table 5). They differ only in T and class count, neither of which depends on $H{\times}W$; the head-independent core (868,303) and the full reconciliation are in Appendix B. At fixed T and widths no parameter scales with $H{\times}W$: the Mamba blocks are $O(d^2)$ and independent of sequence length, which we verify by direct instantiation from 34^2 through 224^2 (43× range in pixel area). Mamba-Spike instead flattens the grid into one dense $\texttt{Linear}(C\lfloor H/4\rfloor\lfloor W/4\rfloor, d)$, exactly quadratic in $\lfloor H/4\rfloor$ and independent of capacity. Fig. 2 plots both regimes.

Constant parameters do not mean constant cost. Over the same range peak activation memory grows ∼30× and throughput falls (Table 19), so weight storage is decoupled from resolution but total memory is not. Measured FLOP/MACs and their k, T, C scaling are in Table 20: 4.50 GMACs deployed, and 8× the token budget costs only 11% more compute.

Accumulate-only inference by linear commutation. A linear temporal filter commutes with the convolution that follows it, and the pair refactors exactly into two static kernels over the filter's *original* operands, which for TDM are binary spikes. Run sequentially TDM emits $\{-\lambda_c, 0, 1, 1{+}\lambda_c\}$ and a GPU evaluates dense MACs (4.50 GMACs deployed); a neuromorphic target evaluates this instead:

Proposition 3 (Dual-Kernel AC Inference)**.** *Let $W^\star \in \mathbb{R}^{C_{out}\times C_{in}\times 3\times 3}$ and $b^\star \in \mathbb{R}^{C_{out}}$ be the fused RepConv parameters, and $\lambda \in \mathbb{R}^{C_{in}}$ be the TDM parameter. With scaled static kernels $W_a^\star = W^\star \cdot \text{diag}(1+\lambda)$ and $W_b^\star = W^\star \cdot \text{diag}(\lambda)$, the composition exactly satisfies:*

$$\text{Conv}(W^\star, \text{TDM}_\lambda(x))_t \;=\; W_a^\star * x_t - W_b^\star * x_{t-1} + b^\star. \tag{6}$$

The identity applies exactly when x_t, x_{t-1} are *original binary spikes*, where Eq. (6) executes as **two accumulate-only (AC) convolutions** plus an element-wise subtraction. It holds for five of six backbone convolutions, failing only for the first, whose operand is the binned event tensor after entry normalization and is real-valued; that one stays MAC, and is 1.54% of backbone convolution operations at 128^2, leaving 98.5% AC-eligible. DS-MPA, the MR-S3A projections, the complex CSiLIF state and the selective scan remain dense and uncosted for neuromorphic silicon. At 45 nm (E_{MAC}=4.6 pJ, E_{AC}=0.9 pJ) the two AC convolutions cost $1.8\bar{r}$ pJ: at most 1.8 pJ, and 0.09–0.36 pJ over the measured $\bar{r} \in [0.05, 0.20]$. That counts synaptic operations for those layers only, against a dense MAC at 100% activity, and excludes memory access. Proof in Appendix A.

Latency and memory. Fusing the multi-branch graph cuts B=8 latency $1.53\times$. At B=1, the edge operating point, RIPE-MambaSpike's footprint is below the baseline's; at B=8 its activations are $\sim 3.8\times$ larger, so it is slower and larger in total memory: the trade is weight memory for activation memory (Appendix C.13).

6 Experiments

Setup. We evaluate on DVS-Gesture (T=16, 128^2), CIFAR10-DVS (T=10, 128^2), N-Caltech101 (T=10, 128^2, 101 classes), DailyDVS-200 (Wang et al., 2024b) (T=10, 224^2, 200 action classes, official subject-disjoint split, 14,773/3,180/4,093 clips) and N-MNIST (Orchard et al., 2015) (T=10, 34^2), using each dataset's standard split. All numbers are single-pass top-1 (no test-time augmentation) after clean batch-norm recalibration. The identical protocol, schedule and split are used for the headline results and all ablations, so every internal comparison is like-for-like. Every headline result is a five-seed measurement: DVS-Gesture 98.18 ± 0.31 (best 98.48), CIFAR10-DVS 81.28 ± 0.11, N-Caltech101 85.49 ± 0.18, N-MNIST 99.55 ± 0.02, DailyDVS-200 45.74 ± 0.40. Tables quote the best run, matching how the published baselines report; σ is used in Section 7 to say which ablation effects clear noise. Per-seed values and the per-dataset σ-in-samples conversions are in Appendix C.5. Full hyperparameters are in Appendix C; we implement RIPE-MambaSpike in PyTorch with the official `mamba_ssm` kernels, training on an NVIDIA A100 GPU. The full training code is submitted as supplementary material.

Baselines. Mamba-Spike (Qin & Liu, 2024a) is the primary baseline, reported three ways. The first is its published figures, which are for a larger configuration with no parameter count stated. The other two run the *released* code (Qin & Liu, 2024b), under its own protocol (94.32/48.90/99.36%) and under ours (95.83/65.30/99.49%). Our protocol *improves* it everywhere, by 16.40 pp on CIFAR10-DVS, so the gap to the published numbers is not under-training on our part. All quoted Mamba-Spike parameters are measured from that code (36.25 M at 128^2) and do not pair with the published accuracies, as detailed in Appendix C.3. Secondary baselines (SpikFormer, SDT/V2, MS-ResNet, QKFormer, SWformer, STAtten, RTFormer, TET-VGG) are quoted from their papers and marked *; those rows are cross-protocol.

Results. Table 1 reports all five benchmarks, and three patterns hold across the four with published parameter counts. RIPE-MambaSpike exceeds the Mamba-Spike reproduction on every shared benchmark by 2.65/16.10/0.08 pp (best-seed). The parameter ratio is benchmark-dependent: $41\times$ on DVS-Gesture and CIFAR10-DVS (36.25 M), but $1.2\times$ on N-MNIST, whose 34^2 input shrinks the baseline to 1.03 M. On CIFAR10-DVS, N-Caltech101 and DailyDVS-200, RIPE-MambaSpike is Pareto-optimal among published models with reported parameter counts (Fig. 3). On DVS-Gesture, the action-specific SpikMamba (Chen et al., 2024) reaches 99.01% at 0.18 M; RIPE-MambaSpike achieves 98.48% ($98.18 \pm 0.31\%$ mean, trailing by $\sim$1–2 clips on this 264-clip test set) while maintaining resolution-independent parameters across both object and action datasets. Widening the stages shows the deployed point is an operating choice, not a ceiling. On CIFAR10-DVS, $[64, 128, 256]$ (3.35 M) gives 82.80 and $[128, 256, 512]$ (13.14 M) gives 83.40, beating SPARTA (13.80 M, 83.06) but not QKFormer's 84.00 at 1.50 M. It leads on accuracy nowhere: SDT (99.30), QKFormer (84.00) and SWformer (88.45) remain ahead, so the claim is frontier position, not dominance. On CIFAR10-DVS the published Mamba-Spike figure (92.5) far exceeds our reproduction, for the protocol reasons of Appendix C.3. DailyDVS-200, at 200 classes and 224^2 the largest scale we test, is nowhere near solved: only one of the twelve entries in Table 1 clears 50% top-1, their median is 41.95 and the ceiling 51.99. RIPE-MambaSpike reaches 45.74 ± 0.40 top-1 and 69.50 ± 0.49 top-5 at 8.04 M: the

Table 1: State-of-the-art comparison across all five event benchmarks (inference-time parameters). * published; ‡/† the released Mamba-Spike code under its own / our protocol; § no parameter count is attributable (Appendix C.3). Among general-purpose backbones, RIPE-MambaSpike is smallest in three of four and Pareto-optimal across datasets (action-specific SpikMamba Chen et al. (2024) is smaller at 0.18 M on DVS-Gesture), leading nowhere on accuracy: it concedes at most 2.9 pp to models 1.7–41× its size, 6.25 pp to the best DailyDVS-200 result. DailyDVS-200 baselines are frame-based video models with no spiking time-step, so their T column reads "–". Best per block underlined.

Method	T	Params	Acc. (%)
DVS-Gesture (T=16)			
TET Deng et al. (2022)	16	3.66 M	97.92*
SpikFormer Zhou et al. (2023)	16	2.57 M	98.30*
SDT Yao et al. (2023)	16	2.57 M	99.30*
MS-ResNet Hu et al. (2024)	16	11.20 M	98.61*
RTFormer Wang et al. (2024a)	16	—	98.61*
SPARTA Jang & Kim (2025)	20	13.80 M	98.78*
SpikMamba Chen et al. (2024)	16	0.18 M	99.01*
Mamba-Spike Qin & Liu (2024a)*	16	—§	97.80
Mamba-Spike Qin & Liu (2024a)‡	16	36.25 M	94.32
Mamba-Spike Qin & Liu (2024a)†	16	36.25 M	95.83
RIPE-MambaSpike (ours)	16	**0.88 M**	**98.48**
CIFAR10-DVS (T=10)			
TET Deng et al. (2022)	10	3.66 M	83.17*
SpikFormer Zhou et al. (2023)	10	2.57 M	80.90*
SDT V2 Yao et al. (2024)	10	4.34 M	82.00*
SWformer Fang et al. (2024)	10	2.05 M	82.90*
RTFormer Wang et al. (2024a)	10	—	83.60*
QKFormer Zhou et al. (2024)	16	1.50 M	84.00*
SPARTA Jang & Kim (2025)	16	13.80 M	83.06*
STAtten Lee et al. (2025a)	10	2.57 M	83.90*
SpikePool Lee et al. (2025b)	16	0.55 M	80.20*
SpikePool Lee et al. (2025b)	16	2.19 M	82.70*
Mamba-Spike Qin & Liu (2024a)*	10	—§	92.50
Mamba-Spike Qin & Liu (2024a)‡	10	36.25 M	48.90
Mamba-Spike Qin & Liu (2024a)†	10	36.25 M	65.30
RIPE-MambaSpike (ours)	10	**0.87 M**	**81.40**

Method	T	Params	Acc. (%)
N-Caltech101 (T=10, 101 classes)			
SDT Yao et al. (2023)	10	2.57 M	82.50*
SDT V2 Yao et al. (2024)	10	4.34 M	83.70*
STAtten Lee et al. (2025a)	10	2.57 M	84.25*
SWformer Fang et al. (2024)	10	2.05 M	88.45*
SpikePool Lee et al. (2025b)	10	0.55 M	81.62*
SpikePool Lee et al. (2025b)	10	2.19 M	85.01*
RIPE-MambaSpike (ours)	10	**0.88 M**	**85.54**
DailyDVS-200 (T=10, 200 classes, 224×224)			
C3D Wang et al. (2024b)	–	147.2 M	21.99*
ESTF Wang et al. (2024b)	–	46.7 M	24.68*
SDT Wang et al. (2024b)	–	—	35.43*
SpikFormer Wang et al. (2024b)	–	—	36.94*
TSM Wang et al. (2024b)	–	24.3 M	40.87*
SlowFast Wang et al. (2024b)	–	33.6 M	41.49*
TimeSformer Wang et al. (2024b)	–	121.2 M	44.25*
Swin-T Wang et al. (2024b)	–	27.8 M	48.06*
EvMamba Wang et al. (2024c)	–	76.5 M	49.65*
DarkShake-DVS Chen et al. (2026)	–	—	51.99*
RIPE-MambaSpike (ours)	10	**8.04 M**	**45.74**
lite	10	3.82 M	42.41
N-MNIST (T=10, saturated)			
PLIF Fang et al. (2021b)	10	0.87 M	99.61*
SpikFormer Zhou et al. (2023)	10	2.57 M	99.61*
SDT Yao et al. (2023)	10	2.57 M	99.55*
Mamba-Spike Qin & Liu (2024a)‡	10	1.03 M	99.36
Mamba-Spike Qin & Liu (2024a)†	10	1.03 M	99.49
RIPE-MambaSpike (ours)	10	**0.87 M**	**99.57**

best spiking method by 8.80 pp, fourth of twelve overall (3.79 pp above that median), and Pareto-optimal among entries with a published count, at 3.5× smaller than Swin-T, 9.6× than EvMamba and 15.1× than TimeSformer, which it beats outright. Width is a capacity requirement at this scale, not an operating choice: the lite/wide pair gives 42.41 ± 0.34 at 3.82 M against 45.74 at 8.04 M (Δ=3.33 pp, Welch's t=14.1). Further per-benchmark detail, including the top-5 asymmetry against TSM, is in Appendix C.5.

7 Ablations

All ablations use the identical protocol, so each Δ characterizes the deployed model directly. The DVS-Gesture Δ is against the five-seed mean (98.18), never the best seed; the single-seed C10 and N-Caltech101 columns are differenced against Table 1's single-run reference (81.40, 85.54; five-seed means 81.28 ± 0.11, 85.49 ± 0.18).

Table 2: Component-wise ablation and cross-dataset transfer. DVSG Δ uses the 5-seed mean; C10/NCal are single-seed.

#	Ablation	DVSG (Δ)	Δ C10	Δ NCal
A1	−DS-MPA → id.	97.73 (–0.45)	–0.80	**–3.16**
A2	−RepConv → 3×3	97.73 (–0.45)	–0.60	–1.94
A3	−TDM (λ=0)	98.11 (–0.07)	–1.40	–2.43
A4	−MR-S3A → st.-3 only	98.11 (–0.07)	–1.10	–1.46
RIPE-MambaSpike (full)		$\mathbf{98.18 \pm 0.31}$	**81.40**	**85.54**

Component-wise (Table 2). Each headline component is removed and reverted to its closest fallback. On DVS-Gesture the four effects span one to two misclassified clips. Anchored to the five-seed mean 98.18 ± 0.31 rather than the best seed (98.48), the removals are $\approx 1.5\sigma$ for DS-MPA and RepConv and $\approx 0.2\sigma$ for TDM and MR-S3A: none clears noise here. The finer-grained benchmarks resolve all four against their own five-seed σ: every component clears 5.5σ on CIFAR10-DVS (σ=0.11 pp) and 8.0σ on N-Caltech101 (σ=0.18 pp), per-component values in Appendix C.8. DVS-Gesture is therefore too saturated to resolve two of them. DS-MPA dominates on N-Caltech101 (−3.16 pp, the largest single effect anywhere), and every

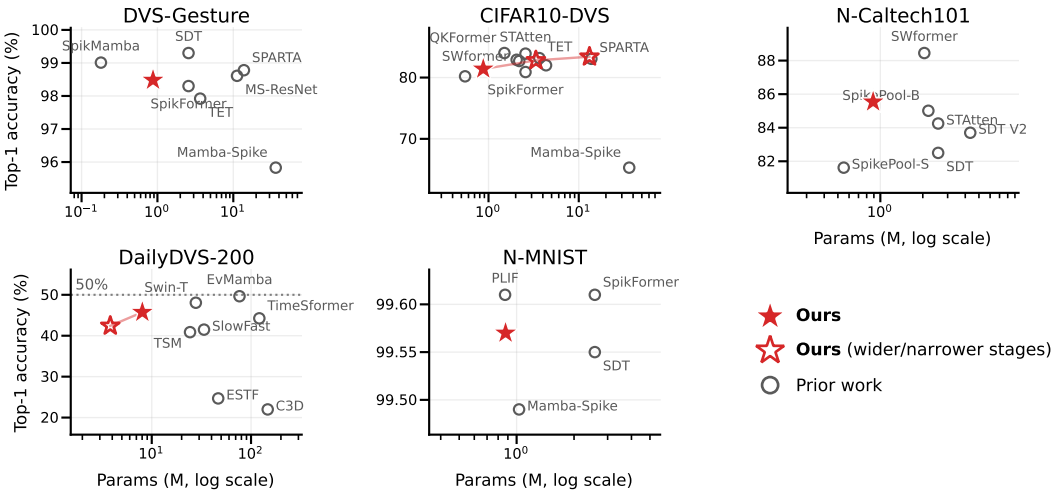


Figure 3: Accuracy vs. inference-time parameters (log scale). Panels plot exactly the Table 1 entries reporting a parameter count and label the frontier, so the DailyDVS-200 panel's *visible* ceiling is EvMamba (49.65) not the benchmark best (51.99); its dotted line marks 50% top-1. RIPE-MambaSpike (filled star) is Pareto-optimal on CIFAR10-DVS, N-Caltech101 and DailyDVS-200, but not on DVS-Gesture, where SpikMamba is smaller and more accurate, nor on the saturated N-MNIST. Hollow stars are rescaled variants: off-frontier on CIFAR10-DVS, but on DailyDVS-200 the *lite* config is itself a frontier point.

Δ is largest on the hardest benchmark, so DVS-Gesture understates their importance rather than contradicting it.

Sequence module, parameter-matched (Table 3). Replacing the sequence mixer alone at a fixed budget (five seeds each), DVS-Gesture again cannot separate the arms: a depthwise Conv1D is indistinguishable from Mamba (-0.22 pp, $p{=}0.22$). N-Caltech101 separates the arms decisively: Transformer -2.75, GRU -3.01, Conv1D -3.26 pp, all $p < 0.001$, deleting the mixer entirely costs -6.05 pp, confirming the DVS-Gesture ordering was noise. A flat, budget-matched Mamba reaches comparable accuracy on CIFAR10-DVS and N-Caltech101, so the hierarchy is an efficiency choice, not an accuracy one.

Table 3: **Parameter-matched sequence-module ablation** (5 seeds/cell, mean $\pm$ s.d.). Each arm is matched to within 0.1% of hierarchical-Mamba (None is 23.7% smaller). $^{***}p{<}0.001$, $^{*}p{<}0.05$, n.s. $p{\geq}0.05$.

Module	DVS-Gest. (Δ)	N-Cal101 (Δ)
Hier. Mamba (ours)	$\mathbf{98.18 \pm 0.31}$	$\mathbf{85.49 \pm 0.18}$
Transformer	$97.65\ (-0.53^{*})$	$82.75\ (-2.75^{***})$
GRU	$97.73\ (-0.45^{*})$	$82.48\ (-3.01^{***})$
Conv1D	$97.96\ (-0.22^{\text{n.s.}})$	$82.24\ (-3.26^{***})$
None	$97.20\ (-0.98^{***})$	$79.44\ (-6.05^{***})$

Sweeps (Appendix C). CSiLIF–SiLIF–SiLIF achieves the top accuracy among eight neuron mixes; accuracy plateaus at $T{=}16$; and performance is insensitive to λ_{L1} across four orders of magnitude.

8 Discussion and Conclusion

Where the parameters go in a Spiking–Mamba hybrid is settled at the interface: in the baseline we profile, 92% of them sit in the one projection joining front-end to scan, quadratic in resolution. A fixed-width bridge holds the deployed count at 0.870 M across a $43\times$ area range. That point covers the object and gesture benchmarks; DailyDVS-200 needs width, and its 8.80 pp lead over spiking methods is at 8.04 M. The diagnosis rests on one released model, so we scope it to flatten-then-project bridges; swapping that projection for a pooled one, nothing else changed, costs no accuracy (Appendix C.2). A front-end this small can carry attention at all only because DS-MPA is bounded by construction (Proposition 2), its normalizer unable to collapse the way a saturating gate does, and removing it costs more than removing anything else we ablate (-3.16 pp on N-Caltech101). Proposition 3 restores binary operands to all but the first backbone convolution, and the commutation it rests on holds for any linear temporal filter placed before a convolution.

AI USE STATEMENT

Generative AI tools were used to critique the theoretical framing, to help verify code and quantitative result checking, to identify literature and to help compose and edit the manuscript. They were not used to fabricate or alter experimental observations. Numerical claims were verified with the original numerical records and experimental figures; mathematical claims were re-derived and verified by the authors who take responsibility for the final manuscript and released materials.

REPRODUCIBILITY STATEMENT

Splits, protocols and per-dataset hyperparameters are in Section 6 and Appendix C.6; the backbone specification and all proofs are in Appendices C.7 and A. Headline accuracies are five-seed measurements, with per-seed values in Appendix C.5. The resolution sweep behind Fig. 2 is in Appendix C.1. The training code, covering all five benchmarks, is submitted as supplementary material. It ships with the released Mamba-Spike code (Qin & Liu, 2024b) and both of the scripts behind its Table 1 rows, one running the authors' own protocol (‡) and one running ours (†), so the 48.90 and 65.30 CIFAR10-DVS figures can be reproduced directly rather than taken on trust. The pooled-bridge control of Appendix C.2 is included on the same footing.

REFERENCES

Arnon Amir, Brian Taba, David Berg, Timothy Melano, Jeffrey McKinstry, Carmelo Di Nolfo, Tapan Nayak, Alexander Andreopoulos, Guillaume Garreau, Marcela Mendoza, Jeff Kusnitz, Michael Debole, Steve Esser, Tobi Delbruck, Myron Flickner, and Dharmendra Modha. A low power, fully event-based gesture recognition system. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Dewei Bai, Hongxiang Peng, Yunyun Zeng, Ziyu Zhang, Hong Qu, and Yi Zhang. Vision Smol-Mamba: Spike-guided token pruning for energy-efficient spiking state-space vision models. *arXiv preprint arXiv:2604.25570*, 2026.

Biswadeep Chakraborty and Saibal Mukhopadhyay. FLAMES: A hybrid spiking-state space model for adaptive memory retention in event-based learning. *arXiv preprint arXiv:2504.01257*, 2025.

Chengpeng Chen, Zichao Guo, Haien Zeng, Pengfei Xiong, and Jian Dong. RepGhost: A hardware-efficient ghost module via re-parameterization. *arXiv preprint arXiv:2211.06088*, 2022.

Jiaqi Chen, Yan Yang, Shizhuo Deng, Da Teng, and Liyuan Pan. SpikMamba: When SNN meets mamba in event-based human action recognition. In *ACM International Conference on Multimedia in Asia (MMAsia)*, 2024. arXiv:2410.16746.

Jiaqi Chen, Qinfu Xu, and Liyuan Pan. Darkshake-dvs: Event-based human action recognition under low-light and shaking camera conditions. *arXiv preprint arXiv:2605.20680*, 2026.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations (ICLR)*, 2021.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, Yuyun Liao, Chit-Kwan Lin, Andrew Lines, Ruokun Liu, Deepak Mathaikutty, Steven McCoy, Arnab Paul, Jonathan Tse, Guruguhanathan Venkataramanan, Yi-Hsin Weng, Andreas Wild, Yoonseok Yang, and Hong Wang. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1):82–99, 2018.

Shikuang Deng, Yuhang Li, Shanghang Zhang, and Shi Gu. Temporal efficient training of spiking neural network via gradient re-weighting. In *International Conference on Learning Representations (ICLR)*, 2022.

Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. RepVGG: Making vgg-style convnets great again. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021a.

Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021b.

Yuetong Fang, Ziqing Wang, Lingfeng Zhang, Jiahang Cao, Honglei Chen, and Renjing Xu. Spiking wavelet transformer. In *European Conference on Computer Vision (ECCV)*, 2024. arXiv:2403.11138.

Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J Davison, Jörg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180, 2022.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations (ICLR)*, 2022.

Yufei Guo, Xinyi Tong, Yuanpei Chen, Liwen Zhang, Xiaode Liu, Zhe Ma, and Xuhui Huang. RecDis-SNN: Rectifying membrane potential distribution for directly training spiking neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Mark Horowitz. 1.1 computing's energy problem (and what we can do about it). In *IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers*, 2014.

Yifan Hu, Lei Deng, Yujie Wu, Man Yao, and Guoqi Li. Advancing spiking neural networks towards deep residual learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

Eugene M Izhikevich. Resonate-and-fire neurons. *Neural Networks*, 14(6–7):883–894, 2001.

Minsuk Jang and Changick Kim. SPARTA: Advancing sparse attention in spiking neural networks via spike-timing-based prioritization. *arXiv preprint arXiv:2508.01646*, 2025.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning (ICML)*, 2020.

Youngeun Kim and Priyadarshini Panda. Revisiting batch normalization for training low-latency deep spiking neural networks from scratch. *Frontiers in Neuroscience*, 15:773954, 2021. arXiv:2010.01729.

Quoc V. Le, Navdeep Jaitly, and Geoffrey E. Hinton. A simple way to initialize recurrent networks of rectified linear units. *arXiv preprint arXiv:1504.00941*, 2015.

Donghyun Lee, Yuhang Li, Youngeun Kim, Shiting Xiao, and Priyadarshini Panda. Spiking transformer with spatial-temporal attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a. arXiv:2409.19764.

Donghyun Lee, Alex Sima, Yuhang Li, Panos Stinis, and Priyadarshini Panda. SpikePool: Event-driven spiking transformer with pooling attention. *arXiv preprint arXiv:2510.12102*, 2025b.

Hongmin Li, Hanchao Liu, Xiangyang Ji, Guoqi Li, and Luping Shi. CIFAR10-DVS: An event-stream dataset for object classification. *Frontiers in Neuroscience*, 11:309, 2017.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. VMamba: Visual state space model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Paul A Merolla, John V Arthur, Rodrigo Alvarez-Icaza, Andrew S Cassidy, Jun Sawada, Filipp Akopyan, Bryan L Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, Bernard Brezzo, Ivan Vo, Steven K Esser, Rathinakumar Appuswamy, Brian Taba, Arnon Amir, Myron D Flickner, William P Risk, Rajit Manohar, and Dharmendra S Modha. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668–673, 2014.

Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019.

Garrick Orchard, Ajinkya Jayawant, Gregory K Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in Neuroscience*, 9:437, 2015.

Jiahao Qin and Feng Liu. Mamba-Spike: Enhancing the mamba architecture with a spiking front-end for efficient temporal data processing. In *Computer Graphics International (CGI)*, volume 15339 of *Lecture Notes in Computer Science*, 2024a. arXiv:2408.11823.

Jiahao Qin and Feng Liu. Mamba-Spike: official implementation. `https://github.com/ecnu-cross-innovation-lab/mamba-spike`, 2024b. GitHub repository.

Pavan Kumar Anasosalu Vasu, James Gabriel, Jeff Zhu, Oncel Tuzel, and Anurag Ranjan. MobileOne: An improved one millisecond mobile backbone. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Hongzhi Wang, Xiubo Liang, Mengjian Li, and Tao Zhang. RTFormer: Re-parameter TSBN spiking transformer. In *International Joint Conference on Neural Networks (IJCNN)*, 2024a. arXiv:2406.14180.

Qi Wang, Zhou Xu, Yuming Lin, Jingtao Ye, Hongsheng Li, Guangming Zhu, Syed Afaq Ali Shah, Mohammed Bennamoun, and Liang Zhang. Dailydvs-200: A comprehensive benchmark dataset for event-based action recognition. In *European Conference on Computer Vision (ECCV)*, 2024b.

Xiao Wang, Shiao Wang, Pengpeng Shao, Bo Jiang, Lin Zhu, and Yonghong Tian. Event stream based human action recognition: A high-definition benchmark dataset and algorithms. *arXiv preprint arXiv:2408.09764*, 2024c.

Wenjie Wei, Xiaolong Zhou, Malu Zhang, Ammar Belatreche, Qian Sun, Yimeng Shan, Dehao Zhang, Zijian Zhou, Zeyu Ma, Yang Yang, and Haizhou Li. TP-Spikformer: Token pruned spiking transformer. *arXiv preprint arXiv:2603.00527*, 2026.

Man Yao, Jiakui Hu, Zhaokun Zhou, Li Yuan, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Man Yao, Jiakui Hu, Tianxiang Hu, Yifan Xu, Zhaokun Zhou, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips. In *International Conference on Learning Representations (ICLR)*, 2024.

Kairong Yu, Tianqing Zhang, Qi Xu, Gang Pan, and Hongwei Wang. TS-SNN: Temporal shift module for spiking neural networks. In *International Conference on Machine Learning (ICML)*, 2025. arXiv:2505.04165.

Chenlin Zhou, Han Zhang, Zhaokun Zhou, Liutao Yu, Liwei Huang, Xiaopeng Fan, Li Yuan, Zhengyu Ma, Huihui Zhou, and Yonghong Tian. QKFormer: Hierarchical spiking transformer using q-k attention. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2403.16552.

Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, Shuicheng Yan, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer. In *International Conference on Learning Representations (ICLR)*, 2023.

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision Mamba: Efficient visual representation learning with bidirectional state space model. In *International Conference on Machine Learning (ICML)*, 2024.

A Proofs of Propositions

A.1 Proof of Proposition 1 (Branch Fusion)

We show that the seven-branch training-time RepConv of Eq. (26) is algebraically equivalent to the single 3×3 convolution-plus-bias defined by Eqs. (27) and (28). The argument has three parts: (i) BN absorption, (ii) reduction of asymmetric and identity branches to 3×3 kernels, and (iii) composition of the sequential branch.

BN absorption. For any branch $b \in \mathcal{B}$ with weight W_b and BatchNorm parameters $(\gamma_b, \beta_b, \mu_b, \sigma_b^2, \varepsilon)$, the affine action of BN on the convolution output is

$$\begin{aligned}\mathrm{BN}_b(W_b * x) &= \gamma_b \cdot \frac{W_b * x - \mu_b}{\sqrt{\sigma_b^2 + \varepsilon}} + \beta_b \\ &= s_b\,(W_b * x) + (\beta_b - s_b\,\mu_b),\end{aligned} \tag{7}$$

where $s_b = \gamma_b/\sqrt{\sigma_b^2 + \varepsilon}$ and we have used the fact that running mean μ_b is a constant per output channel and therefore acts as a bias. Each branch thus decomposes into a *linear* convolution with effective weight $s_b W_b$ and an additive bias $\beta_b - s_b\mu_b$. Importantly, each internal branch batch-normalization BN_b is a standard 2D BatchNorm evaluated across the combined batch and temporal dimensions ($B \times T$), computing time-invariant channel statistics (μ_b, σ_b^2) and affine parameters (γ_b, β_b). Consequently, the absorbed weights $W^\star$ and bias $b^\star$ are strictly static and time-independent, yielding a single static 3×3 kernel for deployment. The temporal batch-normalization (BNTT) is applied exclusively *downstream* of the fused convolution (i.e., $\mathrm{BNTT}_t(\mathrm{Conv}(W^\star, \cdot)_t)$) as an elementwise affine transformation across timesteps, and does not interfere with the spatial kernel fusion.

Asymmetric and identity branches. The asymmetric kernels 1×1, 1×3 and 3×1 are zero-padded to 3×3 support:

$$\widetilde{W}^{1\times1} = \mathrm{Pad}_{[1,1,1,1]}(W^{1\times1}), \tag{8}$$

$$\widetilde{W}^{1\times3} = \mathrm{Pad}_{[1,1,0,0]}(W^{1\times3}), \tag{9}$$

$$\widetilde{W}^{3\times1} = \mathrm{Pad}_{[0,0,1,1]}(W^{3\times1}), \tag{10}$$

where the padding tuples are $[\text{top}, \text{bottom}, \text{left}, \text{right}]$. These paddings preserve the convolutional action because multiplication by zero is the identity for addition.

The identity branch is represented by the centered Kronecker delta kernel $\mathbf{I} \in \mathbb{R}^{c\times c\times3\times3}$ with $\mathbf{I}_{i,j,1,1} = \delta_{ij}$ and zero elsewhere. The AvgPool branch is represented by the constant kernel $\mathbf{A}_{i,j,p,q} = \frac{1}{9}\,\delta_{ij}$ for $p, q \in \{0, 1, 2\}$.

Sequential branch. The $1\times1\to3\times3$ sequential branch composes a 1×1 projection with a 3×3 convolution. By associativity of convolution, this is equivalent to a single 3×3 convolution with weight

$$W_{\text{fused}}^{\text{seq}} \;=\; W_{\text{seq}}^{3\times3} \;*\; \big(W_{\text{seq}}^{1\times1}\big)^{\top}, \tag{11}$$

where the transpose swaps input and output channel axes for the 1×1 kernel. Its BN is then absorbed as in the general case.

Summation. The seven branches act *in parallel* on the same input x and their outputs are summed in Eq. (26). Because convolution is distributive over addition,

$$\sum_b \widetilde{W}_b^{\text{eff}} * x + b_b^{\text{eff}} \;=\; \Big(\sum_b \widetilde{W}_b^{\text{eff}}\Big) * x + \sum_b b_b^{\text{eff}}, \tag{12}$$

which is exactly Eqs. (27)–(28). The equivalence holds pointwise for every input x. □

A.2 Proof of Proposition 4 (TDM Spectral Shape)

The transfer function of TDM is $H_c(z) = (1+\lambda_c) - \lambda_c\, z^{-1}$ (Eq. (31)). Evaluating on the unit circle $z = e^{j\omega}$,

$$\begin{aligned} |H_c(e^{j\omega})|^2 &= \big((1+\lambda_c) - \lambda_c \cos\omega\big)^2 + \big(\lambda_c \sin\omega\big)^2 \\ &= (1+\lambda_c)^2 - 2\lambda_c(1+\lambda_c)\cos\omega + \lambda_c^2 \\ &= 1 + 2\lambda_c(1+\lambda_c)(1-\cos\omega). \end{aligned} \tag{13}$$

Since $1 - \cos\omega \geq 0$ for $\omega \in [0, \pi]$ and is strictly increasing on $[0, \pi]$, the squared magnitude is strictly increasing in ω if and only if $\lambda_c(1+\lambda_c) > 0$, i.e. $\lambda_c \in (-\infty, -1) \cup (0, \infty)$ (high-pass), and strictly decreasing if and only if $\lambda_c(1+\lambda_c) < 0$, i.e. $\lambda_c \in (-1, 0)$ (low-pass). At $\lambda_c \in \{-1, 0\}$ the response is flat. □

A.3 Proof of Proposition 2 (DS-MPA Boundedness)

Let $\phi_i := \phi(Q_i) \in \mathbb{R}^C_{\geq 0}$ and $\psi_j := \phi(K_j) \in \mathbb{R}^C_{\geq 0}$, and let $w_{ij} := \phi_i^\top \psi_j \geq 0$. The attention numerator and denominator for token i, channel c are

$$n_{i,c} = \sum_{j=1}^{N} w_{ij}\, V_{j,c}, \qquad d_i = \sum_{j=1}^{N} w_{ij}. \tag{14}$$

We have $A_{i,c} = n_{i,c}/(d_i + \varepsilon)$. Define $p_{ij} := w_{ij}/d_i \in [0, 1]$ with $\sum_j p_{ij} = 1$ (a probability distribution over tokens). Factoring out the normalizer yields the ε-exact form:

$$A_{i,c} \;=\; \frac{d_i}{d_i + \varepsilon} \sum_{j=1}^{N} p_{ij}\, V_{j,c}. \tag{15}$$

Because $d_i \geq 0$ and $\varepsilon > 0$, the scaling factor satisfies $\eta_i := \frac{d_i}{d_i+\varepsilon} \in [0, 1)$. Hence $A_{i,c}$ is a convex combination of $\{V_{j,c}\}_{j=1}^N$ contracted by η_i, which lies strictly in the extended convex hull $\mathrm{Conv}(\{V_{j,c}\}_{j=1}^N \cup \{0\})$. In particular, for non-negative values $V_{j,c} \geq 0$,

$$0 \;\leq\; A_{i,c} \;\leq\; \max_j V_{j,c}. \tag{16}$$

More generally, regardless of sign,

$$\min\Big(0,\, \min_j V_{j,c}\Big) \;\leq\; A_{i,c} \;\leq\; \max\Big(0,\, \max_j V_{j,c}\Big). \tag{17}$$

No assumption on the sign or magnitude of V is needed: the bound follows from $\phi \geq 0$ (making every $p_{ij} \geq 0$) together with the normalizer. If additionally $V_{j,c} \in [0, 1]$, for instance if V carries raw spikes rather than a linear projection of them, the convex hull is contained in $[0, 1]$ and $A_{i,c} \in [0, 1]$ as a special case. In all cases, $A_{i,c}$ is dynamically bounded and cannot saturate downstream activations. □

A.4 Proof of Proposition 5 (DS-MPA Complexity)

The dominant cost in Eq. (4) is the product $\phi(K)^\top V \in \mathbb{R}^{C\times C}$, which requires $\mathcal{O}(NC^2)$ multiply-adds, followed by $\phi(Q) \cdot (\phi(K)^\top V)$ which is $\mathcal{O}(NC^2)$ multiply-adds. The denominator $\phi(K)^\top \mathbf{1}$ is $\mathcal{O}(NC)$ and $\phi(Q) \cdot (\phi(K)^\top \mathbf{1})$ is $\mathcal{O}(NC)$, both subdominant. The element-wise division and the modulation Eq. (5) are $\mathcal{O}(NC)$. Total: $\mathcal{O}(NC^2)$ operations and $\mathcal{O}(NC + C^2)$ memory (the $C\times C$ key–value product plus the $N\times C$ output).

In comparison, standard softmax attention requires forming the $N \times N$ score matrix at $\mathcal{O}(N^2C)$ cost and $\mathcal{O}(N^2 + NC)$ storage, plus $\mathcal{O}(N^2C)$ for the value matmul. For our deepest stage with $N = 256$ and $C = 128$, the ratio is

$$\frac{N^2C}{NC^2} = \frac{N}{C} = \frac{256}{128} = 2, \tag{18}$$

confirming the constant-factor speed-up cited in the body. □

A.5 Proof of Lemma 1 (TDM Gradient Decoupling)

Let $\tilde{x}_t = (1+\lambda_c)\, x_t - \lambda_c\, x_{t-1}$ be the TDM output, and let f be a smooth downstream operator. By the chain rule,

$$\frac{\partial f(\tilde{x}_t)}{\partial x_{t-1}} \;=\; \frac{\partial f(\tilde{x}_t)}{\partial \tilde{x}_t}\,\frac{\partial \tilde{x}_t}{\partial x_{t-1}} + \frac{\partial f(\tilde{x}_t)}{\partial x_t^{(\text{spike})}}\,\frac{\partial x_t^{(\text{spike})}}{\partial x_{t-1}}, \tag{19}$$

where the second term collects the gradient flowing through the spiking state chain (involving the surrogate gradient Eq. (30)). The first term is

$$\frac{\partial f(\tilde{x}_t)}{\partial \tilde{x}_t}\cdot(-\lambda_c) \;=\; (-\lambda_c)\, f'(\tilde{x}_t), \tag{20}$$

where f' is the ordinary derivative of f at $\tilde{x}_t$. This term contains *no surrogate gradient*, since TDM is a smooth linear operator. The decomposition is therefore exactly the one claimed in the lemma statement. □

A.6 Proof of Proposition 3 (Dual-Kernel Accumulate-Only Inference)

Let $x_t \in \{0,1\}^{C_{\text{in}}\times H\times W}$ denote the binary spike tensor at timestep t. By the definition of TDM (§4.2), for each input channel $c \in \{1,\dots,C_{\text{in}}\}$, the temporally modulated feature is

$$\tilde{x}_{t,c} \;=\; (1+\lambda_c)\, x_{t,c} - \lambda_c\, x_{t-1,c}. \tag{21}$$

Let $W^\star \in \mathbb{R}^{C_{\text{out}}\times C_{\text{in}}\times 3\times 3}$ and $b^\star \in \mathbb{R}^{C_{\text{out}}}$ denote the fused spatial convolution kernel and bias (Proposition 1). The 2D spatial convolution at output channel $k \in \{1,\dots,C_{\text{out}}\}$ evaluates as:

$$\big[\mathrm{Conv}(W^\star,\tilde{x}_t)\big]_k \;=\; \sum_{c=1}^{C_{\text{in}}} W^\star_{k,c} * \tilde{x}_{t,c} + b^\star_k. \tag{22}$$

Substituting the linear definition of $\tilde{x}_{t,c}$:

$$\begin{aligned}\big[\mathrm{Conv}(W^\star,\tilde{x}_t)\big]_k &= \sum_{c=1}^{C_{\text{in}}} W^\star_{k,c} * \big((1+\lambda_c)\, x_{t,c} - \lambda_c\, x_{t-1,c}\big) + b^\star_k \\ &= \sum_{c=1}^{C_{\text{in}}} \big((1+\lambda_c)\, W^\star_{k,c}\big) * x_{t,c} - \sum_{c=1}^{C_{\text{in}}} \big(\lambda_c\, W^\star_{k,c}\big) * x_{t-1,c} + b^\star_k. \end{aligned} \tag{23}$$

Defining the channel-scaled kernels $[W^\star_a]_{k,c} = (1+\lambda_c)\, W^\star_{k,c}$ and $[W^\star_b]_{k,c} = \lambda_c\, W^\star_{k,c}$, or in compact matrix notation:

$$W^\star_a \;=\; W^\star\cdot\mathrm{diag}(1+\lambda), \qquad W^\star_b \;=\; W^\star\cdot\mathrm{diag}(\lambda), \tag{24}$$

we obtain the exact identity:

$$\mathrm{Conv}(W^\star,\mathrm{TDM}_\lambda(x))_t \;=\; W^\star_a * x_t - W^\star_b * x_{t-1} + b^\star. \tag{25}$$

Because $x_t \in \{0,1\}^{C_{\text{in}}\times H\times W}$ and $x_{t-1} \in \{0,1\}^{C_{\text{in}}\times H\times W}$ are unquantized binary spikes, every spatial operation in $W^\star_a * x_t$ and $W^\star_b * x_{t-1}$ performs conditional addition (accumulation) over active event indices ($x_{t,c,u,v} = 1$) with zero multiplications. At $t = 1$, boundary condition $x_0 = 0$ reduces the right-hand side to a single accumulate-only pass $W^\star_a * x_1 + b^\star$. The subtraction of the two pre-activations is an element-wise vector operation per timestep. □

B Method Details

This appendix expands the modules of Section 4 and states the two propositions and the lemma whose proofs appear in Appendix A.

B.1 Architectural Comparison with Mamba-Spike

Table 4 summarizes the structural differences between RIPE-MambaSpike and Mamba-Spike along six design axes; the subsections below expand each.

Table 4: Architectural comparison of RIPE-MambaSpike and Mamba-Spike.

Design axis	Mamba-Spike	RIPE-MambaSpike
Reparameterized convs	×	✓
Per-channel temporal filter	×	✓(TDM)
Linear membrane attention	×	✓(DS-MPA)
Fixed-width SNN→SSM bridge	×	✓(MR-S3A)
Multi-resolution SNN bridge	×	✓
Inference-fused 3×3 kernel	×	✓

B.2 Mamba Discretization

Given $\mathbf{u} \in \mathbb{R}^{L\times d}$, Mamba evolves $\dot{\mathbf{h}}(t) = A\mathbf{h}(t) + Bu(t)$, $y(t) = C\mathbf{h}(t)$, discretized with step Δ by zero-order hold: $\bar{A} = \exp(\Delta A)$, $\bar{B} = (\Delta A)^{-1}(\exp(\Delta A) - I)\Delta B$, giving $\mathbf{h}_l = \bar{A}\mathbf{h}_{l-1} + \bar{B}u_l$, $y_l = C\mathbf{h}_l$, implemented as a parallel selective scan at $\mathcal{O}(LNd)$. B, C, Δ are input-dependent.

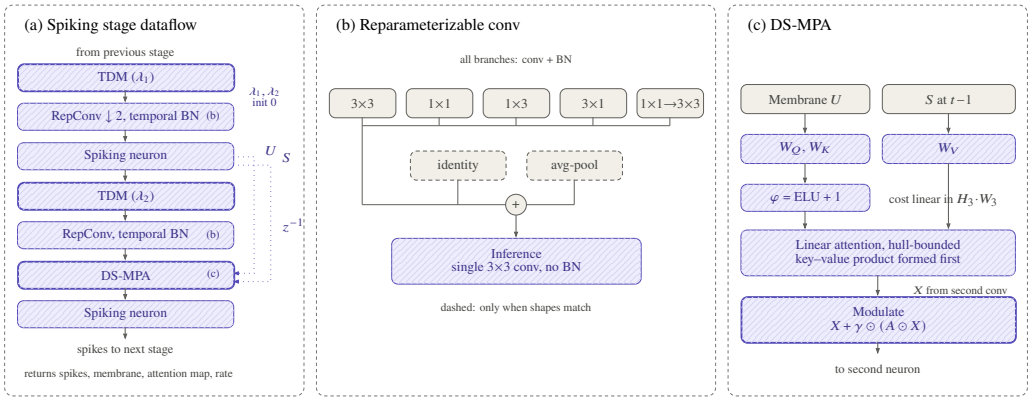


Figure 4: RepSpikeStage internals. **(a)** Two TDM→RepConv→BNTT→neuron units. **(b)** The seven-branch RepConv fuses to a single 3×3 kernel (Prop. 1). **(c)** DS-MPA, a dynamic convex-hull bounded linear attention with ϕ=ELU+1, queries/keys from the membrane U, values from the previous spike tensor.

B.3 Reparameterization: Definitions and Parameter Algebra

At training time each convolution is the seven-branch sum, each branch with its own BatchNorm:

$$\mathrm{RepConv}(x) = \sum_{b\in\mathcal{B}} \mathrm{BN}_b(W_b * x), \quad \mathcal{B} = \{3{\times}3, 1{\times}1, 1{\times}3, 3{\times}1, 1{\times}1{\to}3{\times}3, \mathrm{Id}, \mathrm{AvgPool}\}. \tag{26}$$

With $s_b = \gamma_b/\sqrt{\sigma_b^2 + \varepsilon}$ the fused kernel and bias are

$$W^\star = \textstyle\sum_{b\in\mathcal{B}} \mathrm{Pad}_{3\times3}(s_b W_b), \tag{27}$$

$$b^\star = \textstyle\sum_{b\in\mathcal{B}} (\beta_b - \mu_b s_b). \tag{28}$$

The training-time count (six weight-bearing branches + *seven* BNs, since the identity and average-pool branches carry a BN but no weights) is $P^{\text{train}} = (9{+}1{+}3{+}3{+}9{+}1)c_{\text{in}}c_{\text{out}} + 14c_{\text{out}}$, fusing to $P^{\text{infer}} = 9c_{\text{in}}c_{\text{out}} + c_{\text{out}}$, a ratio $\to 26/9 \approx 2.89$ as $c_{\text{in}}{=}c_{\text{out}} \to \infty$.

Execution order and stage-by-stage parameter accounting. The execution sequence in each spiking block is strictly:

$$x_t \xrightarrow{\text{TDM}} \tilde{x}_t \xrightarrow{\text{RepConv}} y_t \xrightarrow{\text{BNTT}} \hat{y}_t \xrightarrow{\text{Neuron}} s_t, \tag{29}$$

where for the second block in each stage, DS-MPA modulates the post-BNTT features prior to the second neuron ($y_t \to \hat{y}_t \to \text{DS-MPA}(U_1, S_{1,\text{prev}}, \hat{y}_t) \to s_t$). Within each RepConvBlock, the

seven branches possess internal 2D Batch Normalizations (BN_b) computed across all batch and temporal items ($B \cdot T$). Because these internal BNs operate over the pooled temporal dimension, their running statistics and affine scales are time-invariant. Upon deployment, all seven branches and their internal BNs collapse into a single, static 3×3 convolutional weight $W^\star \in \mathbb{R}^{c_{\text{out}} \times c_{\text{in}} \times 3 \times 3}$ and bias $b^\star \in \mathbb{R}^{c_{\text{out}}}$ via Eqs. (27)–(28). Downstream of this fused convolution, BNTT applies an elementwise, per-timestep affine vector $(\gamma_t, \beta_t) \in \mathbb{R}^{2c_{\text{out}}}$ for $t \in [1, T]$ ($2Tc_{\text{out}}$ scalars in total). BNTT does *not* replicate the spatial convolution kernels across timesteps.

We verify the exact stage-by-stage parameter breakdown by direct instantiation of our implementation:

- **Stage 1 (c=32):** $\mathrm{RepConv}_1$ ($2 \to 32$, stride 2: $10{,}624 \to 608$ fused), $\mathrm{RepConv}_2$ ($32 \to 32$, stride 1: $27{,}072 \to 9{,}248$ fused), $\mathrm{BNTT}_1 + \mathrm{BNTT}_2$ ($2 \times 2 \times T \times 32 = 1{,}280$ parameters at T=10), CSiLIF Neurons ($2 \times 5 \times 32 = 320$ parameters), DS-MPA$_1$ ($3 \times 32^2 + 32 = 3{,}104$), and TDM scalars λ_1, λ_2 ($2 + 32 = 34$). Stage 1 total: $42{,}434$ training $\to 14{,}594$ fused parameters (T=10).
- **Stage 2 (c=64):** $\mathrm{RepConv}_1$ ($32 \to 64$, stride 2: $72{,}320 \to 18{,}496$ fused), $\mathrm{RepConv}_2$ ($64 \to 64$, stride 1: $107{,}392 \to 36{,}928$ fused), $\mathrm{BNTT}_1 + \mathrm{BNTT}_2$ ($2 \times 2 \times T \times 64 = 2{,}560$ parameters at T=10), SiLIF Neurons ($2 \times 3 \times 64 = 384$ parameters), DS-MPA$_2$ ($3 \times 64^2 + 64 = 12{,}352$), and TDM scalars ($32 + 64 = 96$). Stage 2 total: $195{,}104$ training $\to 70{,}816$ fused parameters (T=10).
- **Stage 3 (c=128):** $\mathrm{RepConv}_1$ ($64 \to 128$, stride 2: $288{,}000 \to 73{,}856$ fused), $\mathrm{RepConv}_2$ ($128 \to 128$, stride 1: $427{,}776 \to 147{,}584$ fused), $\mathrm{BNTT}_1 + \mathrm{BNTT}_2$ ($2 \times 2 \times T \times 128 = 5{,}120$ parameters at T=10), SiLIF Neurons ($2 \times 3 \times 128 = 768$ parameters), DS-MPA$_3$ ($3 \times 128^2 + 128 = 49{,}280$), and TDM scalars ($64 + 128 = 192$). Stage 3 total: $771{,}136$ training $\to 276{,}800$ fused parameters (T=10).

Across the three stages, training parameters collapse from $1{,}008{,}674$ down to $362{,}210$ (a $2.78\times$ structural compression). All six fused convolutions sum to exactly $286{,}720$ parameters, which are strictly static and time-invariant. The BNTT layers across all three stages account for $8{,}960$ parameters at T=10 (or $14{,}336$ at T=16). Together with the entry temporal norm (40 params at T=10, 64 at T=16), the entire fused spiking backbone comprises $362{,}250$ parameters (or $367{,}650$ at T=16). The MR-S3A bridge adds $133{,}411$: three per-scale projections ($18{,}496$+$36{,}928$+$73{,}792$), three softmax fusion weights, the bridge norm (64), and the attention-injection projection W_A ($4{,}128$). The hierarchical Mamba stack adds $372{,}640$: three blocks at $361{,}760$, two transitions at $10{,}816$, and an input norm of 64. With the classification head ($1{,}290$) and two frozen scalar coefficients retained from the token-ranking study of Table 13, this accounts for exactly $869{,}593$ parameters for a 10-class head at T=10 (≈ 0.870 M). The head-independent core is $868{,}303$; at T=16 with an 11-class head the same model is $875{,}122$ (§5).

The ATan surrogate used throughout is

$$\frac{\partial s_t}{\partial u_t} \approx \frac{\alpha}{2\big(1 + (\frac{\pi}{2}\alpha(u_t - v_{\text{th}}))^2\big)}. \tag{30}$$

B.4 TDM: The Two Additional Views

Spectral shape. TDM's transfer function is

$$H_c(z) = (1 + \lambda_c) - \lambda_c z^{-1}. \tag{31}$$

Proposition 4 (TDM spectral shape)**.** *The squared magnitude response of H_c is $|H_c(e^{j\omega})|^2 = 1 + 2\lambda_c(1+\lambda_c)(1-\cos\omega)$, strictly increasing in $\omega \in [0, \pi]$ (high-pass) for $\lambda_c \in (-\infty, -1) \cup (0, \infty)$ and strictly decreasing (low-pass) for $\lambda_c \in (-1, 0)$.*

Gradient decoupling.

Lemma 1 (TDM gradient decoupling)**.** *With TDM before a smooth operator f and the LIF reset treated as a stop-gradient, $\partial f(\tilde{x}_t)/\partial x_{t-1} = (-\lambda_c) f'(\tilde{x}_t) + (\partial f/\partial x_t^{(spike)})(\partial x_t^{(spike)}/\partial x_{t-1})$; the first term carries no surrogate gradient.*

The TDM path therefore carries a *surrogate-independent* temporal gradient. It survives regardless of whether the neuron is in its firing band, since the first term contains no surrogate factor. We say "surrogate-independent" rather than "unbiased" deliberately: the lemma establishes a differentiable path that bypasses the surrogate, not a statistical property of a gradient estimator.

Table 5: Train-time vs. inference-time parameter count of RIPE-MambaSpike (hierarchical configuration, stage channels $[32, 64, 128]$, hierarchical Mamba level dimensions $(32, 64, 128)$). Structural reparameterization fuses the multi-branch spiking backbone into single 3×3 kernels, reducing backbone parameters by $\sim2.8\times$ at inference and yielding a deployed model of **0.875 M** parameters (Proposition 1).

Component	P^{train}	P^{infer}
RepSpikeStage backbone (6 RepConv units)	1.014 M	0.368 M
Hierarchical Mamba + MR-S3A bridge + classifier	0.508 M	0.508 M
Full model	**1.522 M**	**0.875 M**

B.5 DS-MPA Complexity

Proposition 5 (DS-MPA complexity)**.** *Eq.* (4) *evaluates in $\mathcal{O}(NC^2)$ time and $\mathcal{O}(NC+C^2)$ memory, versus $\mathcal{O}(N^2C)$ time and $\mathcal{O}(N^2+NC)$ memory for softmax attention (time ratio N/C; $2\times$ at the deepest stage, N=256, C=128).*

Mathematical failure mode of sigmoid-gated attention vs. DS-MPA. A parameter-matched sigmoid attention gate is the natural alternative to DS-MPA's normalized feature map, and it carries a specific failure mode:

$$A_{\text{sig}} = \sigma(QK^\top)V, \tag{32}$$

where $\sigma(z) = (1+e^{-z})^{-1}$ and Q, K, V share the identical projections as DS-MPA. In backpropagation, the gradient with respect to pre-activation $z = QK^\top$ is:

$$\frac{\partial A_{\text{sig}}}{\partial z} = \sigma'(z)\cdot V = \sigma(z)\big(1-\sigma(z)\big)\cdot V. \tag{33}$$

When input event rates burst, membrane potentials U fluctuate over large dynamic ranges, pushing $|z| \gg 0$ into the flat saturation tails where $\sigma'(z) \to 0$. In spiking neural networks, backward gradients must already traverse the surrogate gradient non-linearity $\frac{\partial s}{\partial u} = \frac{\alpha}{2(1+(\frac{\pi}{2}\alpha u)^2)}$ of the spiking neurons. Cascading this surrogate factor with $\sigma'(z) \to 0$ causes compound *gradient starvation*, extinguishing gradient flow into the early convolutional layers.

In contrast, DS-MPA's kernel feature map $\phi(z) = \mathrm{ELU}(z) + 1 \geq 0$ possesses an asymmetrical, strictly non-vanishing derivative:

$$\phi'(z) = \begin{cases} 1, & z > 0, \\ e^z > 0, & z \leq 0. \end{cases} \tag{34}$$

Because $\phi'(z) \geq \min(1, e^z) > 0$ everywhere and never saturates for positive activations ($z > 0$), gradients flow smoothly into U_1 regardless of burst amplitude. Combined with the linear normalizer $\sum_j \phi(K_j) + \varepsilon$ that contracts the output into the dynamic convex hull of V (Proposition 2), DS-MPA prevents both output divergence and gradient starvation. The argument is structural. Ablation A1 replaces DS-MPA with the identity, so its -3.16 pp on N-Caltech101 establishes that the module matters rather than isolating boundedness as the reason; separating the two needs the sigmoid-gated control above, which we do not train here.

B.6 Dual-Branch Training with the SGC Loss

Every Mixup batch runs twice through the shared weights: a discrete pass (hard spikes, ATan surrogate) supervised by $\mathcal{L}_{\text{TET}}$, and a continuous pass (soft spikes $\sigma(\alpha(u-\theta))$, exact gradient). The symmetric $\mathcal{L}_{\text{SGC}} = \mathrm{MSE}(z_d, z_c)$ couples them (a mutual-consistency penalty, not distillation). At inference the continuous branch is dropped, RepConv is fused, BN is recalibrated on clean data, and one discrete pass is averaged over T (Fig. 5). The consistency gap falls from a peak of ≈ 0.11 to $\approx 1.8 \times 10^{-3}$ over training.

B.7 MR-S3A Bridge Detail

Pseudo-code and the exact per-scale tensor shapes for the bridge introduced in §4.4 are below. A learnable projection $W_A \in \mathbb{R}^{C_3 \times d_{\text{m}}}$ additionally injects the stage-3 DS-MPA attention map $\bar{a}_3$ into

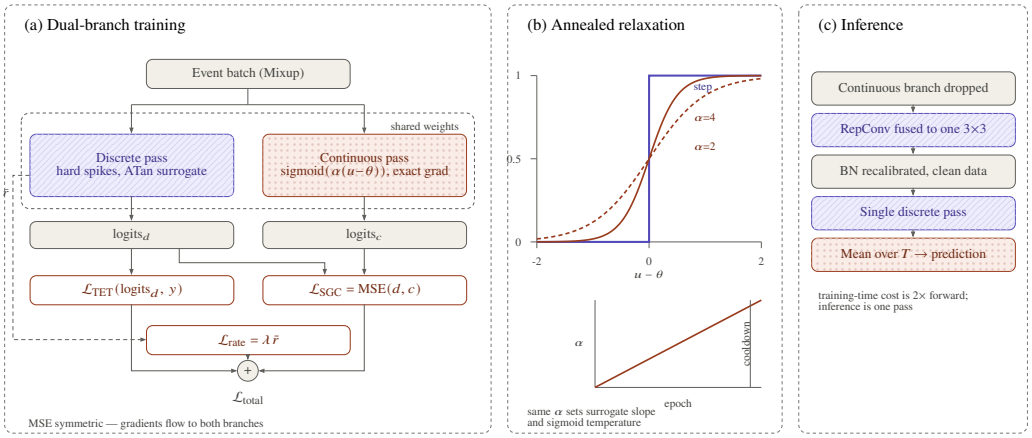


Figure 5: Dual-branch training with the SGC loss. Coral marks training-time-only components removed at deployment. During the continuous pass the BatchNorm momentum is frozen so running statistics are not updated twice per step.

```
# DS-MPA. U, S_prev, X : (B,T,C,H,W).  N = H*W tokens, C channels.
#   U      = membrane potentials at t      -> queries and keys
#   S_prev = BINARY spikes at t-1          -> values
#   X      = features to modulate
Q, K = W_Q(U), W_K(U)                # 1x1 convs; reshape to (B*T, N, C)
V    = W_V(S_prev)                   # 1x1 conv on spikes; unconstrained
phiQ, phiK = elu(Q) + 1, elu(K) + 1        # feature map, phi >= 0

KV  = phiK.transpose(1, 2) @ V       # (B*T, C, C)  contraction over TOKENS
num = phiQ @ KV                      # (B*T,N,C)  O(N C^2), no NxN matrix
den = (phiQ * phiK.sum(1, keepdim=True)).sum(-1, keepdim=True)
A   = num / (den + eps)              # convex comb. of V's rows -> Prop. 2
A   = A.permute(0, 2, 1).view(B, T, C, H, W)  # (B*T,N,C) -> (B,T,C,H,W)

return X + gamma * (A * X), A        # gamma: per-channel, init 1
```

Figure 6: **DS-MPA in full.** Channel mixing, not token mixing: the key–value contraction is over the N token positions and produces a $C \times C$ matrix, so cost is $\mathcal{O}(NC^2)$ and no $N \times N$ attention matrix is ever materialized (contrast softmax attention's $\mathcal{O}(N^2C)$). Queries and keys read the *current* membrane potential while values read the *previous* binary spikes, and the "dual-stream" of the name. Two consequences are visible directly in the code: $\phi \geq 0$ plus the normalizer `den` make `A` a convex combination of the rows of `V` (Prop. 2), and there is no additive learnable gate bias anywhere, so no saturating scalar can zero the gradient. `W_V` is deliberately unconstrained; forcing $V \in [0, 1]$ would require a sigmoid and reintroduce exactly that failure mode.

the token sequence via $\mathbf{tokens} \mathrel{+}= W_A \bar{a}_3$, a resolution-independent modulation (same $C_3 \to d_{\text{m}}$ regardless of $H \times W$) contributing 0.5% of the deployed parameters.

B.8 Formal Derivation of the CSiLIF Neuron

The Complex-valued Sigmoid LIF (CSiLIF) neuron used in Stage 1 (§4.1) is a discrete-time resonate-and-fire model (Izhikevich, 2001) with learnable complex poles, derived below.

Continuous dynamics. Each channel c maintains a complex-valued membrane state $m(t) \in \mathbb{C}$ evolving under the linear ODE

$$\dot{m}(t) = \rho_c \, m(t) + b_c \, x(t), \tag{35}$$

where $\rho_c = -e^{\ell_c} + j\,\omega_c \in \mathbb{C}^-$ is a pole in the open left half-plane (the parameterization $-e^{\ell_c}$ enforces $\mathrm{Re}(\rho_c) < 0$ for every $\ell_c \in \mathbb{R}$) and $b_c \in \mathbb{R}$ is a per-channel input gain.

```
# MR-S3A bridge. S_i: (B, T, C_i, H_i, W_i) spikes, i in {1,2,3}.
#   d_m: token width (fixed, independent of H x W)
#   N  = H_3 * W_3: token count at deepest resolution
tok = []                                          # collect per-scale
for i in 1, 2, 3:
    r     = cumsum(S_i, dim=T) / arange(1, T+1)   # cum. rate
    d     = r - shift(r, 1, dim=T)                # first diff.
    f_i   = concat([r, d], dim=C)                 # 2*C_i chans
    f_i   = Pool_{H_3,W_3}(f_i)                   # H_3 x W_3
    tok_i = Proj_i(f_i)                           # 3x3 conv + BN -> d_m
    tok.append(tok_i.reshape(B, T, N, d_m))       # flatten spatial

w = softmax(w_hat)                                # in R^3
# Partition of unity across scales: (B, T, N, d_m)
tokens = w[0]*tok[0] + w[1]*tok[1] + w[2]*tok[2]
tokens = tokens + W_A(a3_pooled)                  # + attention mod.

return tokens  # all N tokens enter Mamba scan
```

Figure 7: **MR-S3A in full.** Each stage's binary spike tensor is summarized by a cumulative rate r and its first difference δ (rate *and* instantaneous coding, $2C_i$ channels), pooled to the deepest stage's spatial size $H_3 \times W_3$, and projected to the shared token width d_{m} by a per-scale 3×3 convolution with BatchNorm. The three scales are combined by a learnable softmax weight $w \in \Delta^2$ (a partition of unity), not concatenated or gated, so the output is always $N{=}H_3W_3$ tokens of width d_{m} regardless of the input resolution $H\times W$; only H_1, H_2, H_3 (and W_1, W_2, W_3) change with resolution, and Pool absorbs that into a fixed output size before d_{m} is ever touched. This is the mechanism behind the resolution-independence of Section 4.4: no weight in Proj_i or in the softmax gate depends on H or W. A learnable projection W_A injects the stage-3 DS-MPA attention map into the token sequence (matching the $+W_A\bar{a}_3$ of the code and Fig. 1). All N tokens are retained, with no top-k selection or pruning is applied anywhere in this path. Exact per-level shapes (d_{m}, token count, pooling factors) are in Table 10.

Zero-order hold discretization. With a learnable step $\Delta t_c = e^{\tau_c} > 0$ (parameterized as $\tau_c =$ `log_dt` in the code), the exact zero-order hold gives the discrete pole

$$\alpha_c = \exp(\rho_c\,\Delta t_c) = \exp\big((-e^{\ell_c} + j\,\omega_c)\,e^{\tau_c}\big). \tag{36}$$

Unconditional Schur stability.

Proposition 6 (CSiLIF stability)**.** *For every $(\ell_c, \omega_c, \tau_c) \in \mathbb{R}^3$, the discrete pole satisfies $|\alpha_c| < 1$.*

Proof. $|\alpha_c| = |\exp(-e^{\ell_c}\Delta t_c)| \cdot |\exp(j\,\omega_c\,\Delta t_c)| = \exp(-e^{\ell_c}\Delta t_c) \cdot 1 = \exp(-e^{\ell_c}\,e^{\tau_c})$. Since $e^{\ell_c} > 0$ and $e^{\tau_c} > 0$, the exponent is strictly negative, so $|\alpha_c| < 1$. No clipping or projection is needed. □

Discrete recurrence with soft reset. The membrane evolves as

$$m_t = \alpha_c\,(m_{t-1} - \kappa\,s_{t-1}) + b_c\,x_t, \tag{37}$$

where $\kappa = 0.5$ is the reset factor (a hyperparameter; $\kappa = 1$ corresponds to hard reset). The real-valued readout and spike are

$$U_t = 2\ \mathrm{Re}(m_t), \qquad s_t = \Theta(U_t - v_{\text{th},c}), \tag{38}$$

with $v_{\text{th},c}$ a learnable per-channel threshold and Θ replaced by the ATan surrogate (Eq. (30)) during backpropagation.

Interpretation. The imaginary part of ρ_c gives each channel a learnable oscillation frequency ω_c, turning the low-pass response of a real LIF into a damped band-pass. The readout $U_t = 2\,\mathrm{Re}(m_t)$ discards the imaginary component, which carries phase information only; the factor 2 compensates for the energy split between real and imaginary parts. Compared to SiLIF (Stages 2–3), which has $\rho_c \in \mathbb{R}^-$ and no oscillation, CSiLIF adds one learnable scalar (ω_c) per channel at negligible parameter cost.

Table 6: **The SNN→SSM interface's parameter cost scales with input resolution; RIPE-MambaSpike's does not.** Both models instantiated and counted directly at each resolution (RIPE-MambaSpike after fusion, i.e. the deployed graph); configuration held fixed down each column, so this isolates resolution alone.

Input	RIPE-MambaSpike	Mamba-Spike		
resolution	deployed (M)	total (M)	`input_proj` (M)	share
34×34	**0.870**	4.79	2.10	43.8%
48×48	**0.870**	7.41	4.72	63.6%
128×128	**0.870**	36.25	33.55	92.6%
224×224	**0.870**	105.46	102.76	97.4%
growth	$\mathbf{1.00}\times$	$22.0\times$	$49.0\times$	—

C Experimental Details and Extended Ablations

This appendix supplies the material needed to reproduce every number in the main text. That includes the exact per-dataset recipe (§C.6), the backbone specification the parameter counts are derived from (§C.7), the four sweeps summarized in one paragraph of §7 (§C.9–§C.10), the optimization behaviour behind the reported schedule (§C.11), and qualitative evidence for the bounded-attention claim together with the per-class breakdown behind the headline DVS-Gesture accuracy (§C.12). Unless stated otherwise every number is a single-seed measurement under the protocol of §6; the five-seed results are confined to the five headline accuracies.

C.1 Instantiated Resolution Sweep

The counts behind Fig. 2, tabulated. Both models are instantiated at each resolution and counted directly (no training, no estimation); RIPE-MambaSpike is counted after reparameterization fusion, i.e. the deployed graph, and the configuration is held fixed down each column so this isolates resolution alone.

C.2 The Interface Swap in Isolation

Table 6 shows what the flatten-then-project bridge costs in parameters. Whether it earns them is a separate question, and it is answerable by changing that one layer in the released Mamba-Spike model. In place of flattening the front-end feature map and projecting it densely (Eq. 1), we average-pool it to a fixed 8×8 grid and project from there. The spiking front-end, neuron models, Mamba blocks and classifier are the released ones, and both arms share a trainer, optimizer, schedule and epoch budget; `input_proj` is the only module that differs.

Table 7: Interface swap on CIFAR10-DVS at 128^2, released Mamba-Spike code under its own protocol. Parameters are counted from the trained checkpoints.

Bridge	Total params	`input_proj`	Top-1
Flatten-then-project (released)	36.25 M	33.55 M (92.6%)	48.90
Pool to 8×8, then project	**4.80 M**	**2.10 M** (43.7%)	**53.30**

Pooling removes $16\times$ of the bridge and $7.6\times$ of the whole model, and gains 4.40 pp on top (Table 7). The measured 92.6% also confirms from a trained checkpoint the share quoted in §1 from instantiation alone. This protocol carries no augmentation and drives both arms to 100% train accuracy, so both sit well below the augmented CIFAR10-DVS figures in Table 1; the comparison is internal to the pair, and it bounds what the resolution-coupled projection buys rather than measuring either bridge at its best.

C.3 Protocol Markers in the SOTA Table

Table 1 carries four markers, expanded here so the caption can stay short. * marks a figure quoted from the method's own paper. We did not retrain those, so they are cross-protocol comparisons. ‡

marks the released Mamba-Spike code run under *its own* protocol, and † the same code under *ours*, which is the like-for-like comparison. § marks published Mamba-Spike figures that are for a larger configuration whose size the paper does not state, so no parameter count is attributable to them; the 36.25 M we quote is measured from the released code instead. Its published 99.4% is on a MNIST variant it calls Sequential MNIST (Le et al., 2015), not N-MNIST, so we do not map the two onto each other. On CIFAR10-DVS the published Mamba-Spike figure (92.50) far exceeds our reproduction, and its released baselines there likewise exceed those methods' values elsewhere. That points to a different protocol, so we rest the comparison on our controlled reproduction instead. DailyDVS-200 parameter counts are from (Wang et al., 2024c), the only source reporting them. Five-seed means for RIPE-MambaSpike are in §6.

C.4 Why Parameter Count Is the Binding Constraint

The efficiency claim is a deployed parameter count, not operations or energy (§8). That axis is the one that binds on the platforms event cameras are actually mounted on.

Event-based sensors are used predominantly where power and payload are budgeted: micro-aerial vehicles, mobile robots, automotive perception and wearable devices (Gallego et al., 2022). On such platforms weights are held in on-chip SRAM or modest external flash, and on neuromorphic accelerators the constraint is tighter still, since synaptic memory is distributed across cores and is not expandable (Merolla et al., 2014; Davies et al., 2018). Unlike an arithmetic budget, which trades against latency and can be met by running slower, a weight budget binds *absolutely*: a model whose parameters do not fit does not run at reduced speed, it does not run at all.

Table 8 makes the resulting gap concrete at half precision.

Table 8: Deployed weight storage at FP16 (2 bytes per parameter). Parameter counts are the deployed/fused values used throughout; baseline counts are as cited in the corresponding SOTA table. Storage is weights only; activations and the runtime are additional.

Model	Params	FP16 weights
RIPE-MambaSpike (small benchmarks)	0.870 M	**1.7 MB**
RIPE-MambaSpike *lite* (DailyDVS-200)	3.82 M	**7.6 MB**
RIPE-MambaSpike (DailyDVS-200)	8.04 M	**16.1 MB**
TSM	24.3 M	48.6 MB
Swin-T	27.8 M	55.6 MB
Mamba-Spike	36.25 M	72.5 MB
EvMamba	76.5 M	153.0 MB
TimeSformer	121.2 M	242.4 MB
C3D	147.2 M	294.4 MB

The practical reading is that the top of the DailyDVS-200 table is not reachable on this class of device. Conceding 3.91 pp to EvMamba is a different trade once EvMamba needs 153 MB of weight storage against our 16.1 MB. The larger model is not a faster-or-slower option; it is an unavailable one.

C.5 Per-Seed Values and Noise Floors

Five seeds per headline result. DVS-Gesture: $\{97.73, 98.11, 98.11, 98.48, 98.48\}$, i.e. 98.18 ± 0.31, best 98.48; on the 264-sample split one clip is 0.38 pp, so $\sigma \approx 0.83$ clips. CIFAR10-DVS: $\{81.1, 81.3, 81.3, 81.3, 81.4\}$, i.e. 81.28 ± 0.11, best 81.40, a much tighter $\sigma \approx 1.1$ of its 1,000 test samples. N-Caltech101: 85.49 ± 0.18, and the reported 85.54 is the median and modal outcome (3/5); on 823 test samples $\sigma \approx 1.5$ samples. N-MNIST: 99.55 ± 0.02, best 99.57, i.e. $\sigma \approx 1.8$ of its 10,000 test samples, smaller than the gaps separating the leading methods there, which is why we treat that benchmark as saturated. DailyDVS-200: $\{45.91, 45.74, 45.91, 45.05, 46.08\}$ top-1, i.e. 45.74 ± 0.40, and $\{69.38, 69.58, 69.73, 68.75, 70.07\}$ top-5, i.e. 69.50 ± 0.49, on the 4,093-clip test split where one clip is 0.024 pp.

C.6 PER-DATASET HYPERPARAMETERS

Table 9 gives the complete recipe per dataset as a set of deltas against one default column, so that what actually varies is visible at a glance. Three groups of differences are worth calling out, since each was forced by a property of the data rather than tuned freely. The two harder, smaller-sample benchmarks, CIFAR10-DVS and N-Caltech101, need more regularization: drop-path doubles to 0.2, weight decay drops to 0.03, Mixup strengthens to $\alpha{=}0.4$ at probability 0.5, and warm-up lengthens to 15 epochs. This is the standard response to a few-thousand-sample event dataset, and we did not search it per-dataset beyond these values. The auxiliary losses, SGC and the spike-rate L_1, are enabled only for DVS-Gesture and N-MNIST; on CIFAR10-DVS and N-Caltech101 we found a TET-only objective at least as good, so those runs use it. N-Caltech101 and DailyDVS-200 were trained on Google Colab at the default batch of 32; N-MNIST is the one departure, compensating for its small 34^2 frames with batch 64.

Table 9: Per-dataset hyperparameters for RIPE-MambaSpike. "—" means identical to "Default".

Hyperparameter	Default	DVSG	C10-DVS	NCal101	NMNIST	DDVS-200
Architecture						
Stage channels $[c_1, c_2, c_3]$	$[32, 64, 128]$	—	—	—	—	$[64, 128, 256]$ *(lite)* / $[96, 192, 384]$
Hier. level dims	$(32, 64, 128)$	—	—	—	—	$(64, 128, 256)$ *(lite)* / $(96, 192, 384)$
Hier. pool ratios	$(1, 2, 2)$	—	—	—	—	—
d_{state}	16	—	—	—	—	—
Drop-path rate	0.1	—	0.2	0.2	—	0.2
Classifier head width	—	—	—	—	—	$1024^{\dagger}$
Optimization						
Optimizer / Init LR / Min LR	AdamW / 10^{-3} / 10^{-6}	—	—	—	—	AdamW/5×10^{-4}/10^{-6}
Weight decay (decay group)	0.05	—	0.03	0.03	—	0.02
LR warm-up / Epochs	10 ep / 300	—	15 ep/—	15 ep/—	—	2000 it.¶/250
Batch size	32	—	—	—	64	—
Grad-accum steps	1	—	—	—	2	—
Grad-clip $\|g\|_2$	1.0	—	—	—	—	—
Surrogate α (start→end)	$2 \to 4$	—	$2 \to 3$	$2 \to 3$	—	—
Loss						
TET λ	5×10^{-3}	—	10^{-2}	10^{-2}	—	—
SGC λ_{SGC}	1.0	—	$0^{\#}$	$0^{\#}$	—	$0^{\ddagger}$
L1 λ_{L1}	10^{-4}	—	$0^{\#}$	$0^{\#}$	—	—
Mixup α / prob	0.2 / 0.3	—	0.4/0.5	0.4/0.5	0.4/0.5	0.2/0.5
Label smoothing	0.1	—	0.05	0.05	0.05	—
EventCutMix α	—	—	0.4	0.2	0.2	—
Data						
T (time bins)	16	—	10	10	10	10
Spatial (H, W)	128^2	—	—	—	34^2	224^2
NDA n_{ops} / mag	1 / 0.3	—	2/0.5	2/0.2	2/—	n/a§
NDA roll / cutout cap	0.25 / 0.25	—	—	0.15/0.15	0.15/0.20	n/a§
Polarity flip / H-flip / T-rev prob	0.5/—/—	—	—/0.5/—	—	—/0.5/0.5	n/a§/0.5/n/a§

† DailyDVS-200 inserts a widening projection (`Linear`→`LayerNorm`→`GELU`→`Linear`) before the classifier: a 384-dimensional penultimate feature cannot form the optimal simplex for 200 classes, so it is widened to 1024 first. The other four benchmarks use a single linear head. ¶ DailyDVS-200's warm-up is specified in optimizer iterations (steps), not epochs, in the training script; the other columns' warm-up is epoch-based. ‡ Only the SGC consistency term is disabled for DailyDVS-200; spike-rate L_1 stays at its default rate, unlike C10-DVS and N-Caltech101 (marked $^{\#}$), where both are disabled for a TET-only objective. § Not implemented in the DailyDVS-200 data pipeline (no NDA transform, no polarity flip or temporal reversal); DailyDVS-200's own horizontal-flip augmentation (prob. 0.5) is unrelated to the NDA/reversal pipeline used by the other four datasets.

C.7 HIERARCHICAL MAMBA BACKBONE SPECIFICATION

Table 10 specifies the hierarchical scan that the complexity analysis of §5 and the parameter counts of Table 6 are computed from. The design principle is that each level balances sequence length against representational width. Moving down a level halves the token count L_ℓ and doubles the model dimension d_ℓ: the state-space scan cost $\mathcal{O}(L_\ell d_{\text{state}} d_\ell)$ stays exactly constant, while the input/output projection cost $\mathcal{O}(L_\ell d_\ell^2)$ scales by only $2\times$ per level (subquadratic compared to unpooled widening). Total cost is therefore a *sum* of three compact level costs rather than one large monolithic scan, which is what allows a wide final Mamba dimension ($d_{\text{m}}{=}128$) at a sequence length of only 64 retained

tokens by the deepest level. The state dimension is held at d_{state}=16 and expansion at 2 throughout; these are Mamba defaults and we did not tune them. The Level-1 token count is H_3W_3, the deepest spiking stage's spatial grid (§4.4), and halves at each subsequent level by the pool ratios; none of this is a function that is tuned per resolution, which is precisely the property that makes the deployed parameter count resolution-independent (§5).

Table 10: Per-level configuration of the hierarchical bidirectional Mamba backbone (DVS-Gesture, H_3=W_3=16). Each level is a forward+backward scan pair; "tokens" is the retained spatial-token count (preserved across all T).

Level	Model dim d_{m}	Tokens k_ℓ	State d_{state}	Expansion
1	32	256	16	2
2	64	128	16	2
3	128	64	16	2

C.8 Per-Component Ablation Significance

Section 7 judges each ablation against the five-seed standard deviation measured on the *same* dataset, rather than against a single pooled figure. Table 11 gives the per-component values behind the summary stated there. DVS-Gesture, at 264 test clips and σ=0.31 pp, simply lacks the resolution to separate two of the four components, while the finer-grained benchmarks separate all four by a wide margin. We treat DVS-Gesture as the weakest of the three ablation substrates despite being our headline benchmark, and read the CIFAR10-DVS and N-Caltech101 columns as the load-bearing evidence.

Table 11: Ablation effect sizes in units of each dataset's own five-seed σ. DVS-Gesture σ=0.31 pp, CIFAR10-DVS 0.11 pp, N-Caltech101 0.18 pp. Effects below $\approx 2\sigma$ are not claimed.

Component removed	DVSG	CIFAR10-DVS	NCal101
DS-MPA	1.5σ	7.3σ	17.4σ
RepConv	1.5σ	5.5σ	10.7σ
TDM	0.2σ	12.8σ	13.4σ
MR-S3A	0.2σ	10.0σ	8.0σ

C.9 Neuron-Mix and Pruning Sweeps

Two design choices in the deployed model are reported here as swept rather than asserted. For the *neuron mix* (Table 12), we evaluated eight depth-by-depth assignments of CSiLIF and SiLIF across the three spiking stages. The deployed CSiLIF–SiLIF–SiLIF ordering is the best of the eight, and placing CSiLIF in both early stages is the most harmful configuration. Spike-rate variance is comparable across all eight, so the choice rests on accuracy alone and we make no stability claim for it. The result is consistent with the intended reading of CSiLIF: a complex-valued resonate-and-fire neuron is most useful where the input still carries raw temporal structure, i.e. at stage 1.

Token pruning is the wrong lever for this architecture (Table 13). An earlier version selected a top-k subset of spatial tokens before the scan; we removed it, and the sweep below is why. On DVS-Gesture, accuracy is identical at 98.48% for $k \in \{64, 128, 256\}$, where k=256 is the full 16×16 grid, i.e. no pruning at all, so pruning bought no accuracy. It also bought no parameters, since the invariance in Table 6 holds with pruning disabled because Mamba blocks are $\mathcal{O}(d_{\text{m}}^2)$ and independent of sequence length, and it bought little compute, since an 8× change in sequence length moves total FLOPs by only 11% (Table 20). A component that buys none of the three is better removed than defended, so the deployed model scans every token MR-S3A produces. One honest limit of the sweep: the DVS-Gesture test split has 0.38 pp granularity per clip, so the tied entries are indistinguishable at this sample size rather than provably equal.

Table 12: Heterogeneous neuron-mix ablation on DVS-Gesture. Each row assigns a neuron family {LIF, SiLIF, CSiLIF} to each of the three RepSpikeStages. "SR-var" is the cross-epoch standard deviation of the mean spike rate during the last 50 epochs (lower = more stable). All rows are measured single-seed runs; the deployed configuration (**bold**) attains the highest accuracy, while training is comparably stable across all mixes (SR-var $\leq 10^{-3}$). Rows sorted by accuracy.

Configuration (Stage 1, 2, 3)	Acc. (%)	Params (M)	SR-var
CSiLIF, SiLIF, SiLIF (ours)	**98.48**	**0.88**	**0.0005**
SiLIF, SiLIF, CSiLIF	98.11	0.88	0.0004
SiLIF, CSiLIF, CSiLIF	98.11	0.88	0.0003
LIF, SiLIF, SiLIF	97.73	0.88	0.0003
LIF, LIF, LIF	97.73	0.88	0.0009
SiLIF, SiLIF, SiLIF	97.35	0.88	0.0002
CSiLIF, CSiLIF, SiLIF	96.59	0.88	0.0005
CSiLIF, CSiLIF, CSiLIF	96.21	0.88	0.0005

Table 13: **Why the deployed model prunes no tokens.** An earlier version of this architecture selected a top-k subset of spatial tokens before the scan; this sweep is why we removed it. k is the number of spatial tokens retained per timestep and the full sequence is $N{=}H_3W_3{=}256$, so $k{=}256$ *is* the unpruned model. Accuracy is identical for $k \in \{64, 128, 256\}$: pruning to a quarter of the tokens bought no accuracy. It also bought no parameters — the resolution invariance of Table 6 holds with pruning disabled, since Mamba blocks are $\mathcal{O}(d_{\text{m}}^2)$ and independent of sequence length — and little compute, an $8\times$ change in sequence length moving total FLOPs by 11% (Table 20). The deployed model therefore scans every token. Caveat: the DVS-Gesture split has 0.38 pp granularity per clip, so the tied entries are indistinguishable at this sample size rather than provably equal; $k{=}32$ costing 0.75 pp is two clips.

k	32	64	128	**256 (full, deployed)**
Tokens / sample	512	1024	2048	**4096**
Acc. (%)	97.73	98.48	98.48	**98.48**
Δ (pp)	–0.75	0.00	0.00	**—**

C.10 Time-Step and Spike-Rate Sweeps

Table 14 sweeps the number of time bins. Accuracy rises steeply to $T{=}16$ and is then flat (98.48% at both 16 and 20), while measured compute (MACs, not SOPs, §5) grows strictly linearly in T. $T{=}16$ is the knee; everything beyond it buys computation with no accuracy. This is the single most consequential efficiency knob in the model. Halving T to 8 halves the operation count at a cost of 1.89 pp, a trade a deployment might reasonably take, whereas raising T past 16 is never worthwhile.

Figure 8 sweeps the spike-rate L_1 coefficient across four orders of magnitude. Both accuracy and the mean spike rate $\bar{r}$ are nearly flat over the whole range, with the default 10^{-4} giving the best accuracy at a low rate. This is a negative result worth reporting: sparsity here is governed mainly by the architecture and the surrogate-gradient schedule, not by the explicit penalty, so λ_{L1} is not a sensitive hyperparameter and should not be presented as one.

Table 14: Sensitivity to time bins T on DVS-Gesture. All measured single-seed. MACs rather than SOPs because TDM makes backbone convolutions dense (§5).

T	4	8	12	16	20
Acc. (%)	94.32	96.59	97.73	**98.48**	98.48
MACs (M)	430.5	861.0	1291.4	1721.9	2152.4

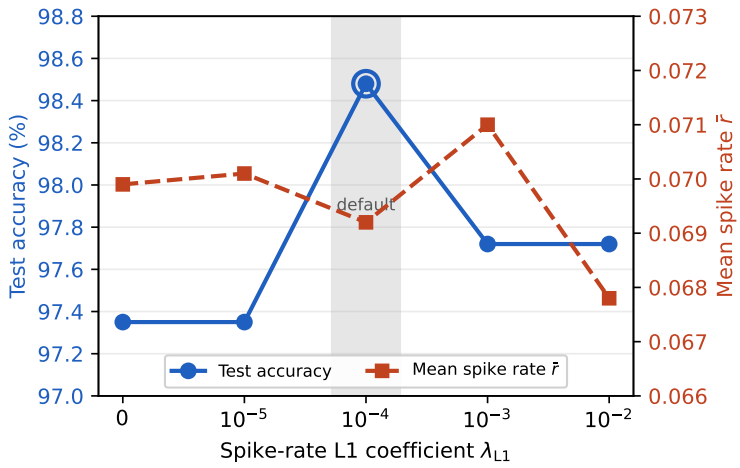


Figure 8: Sensitivity to the spike-rate L1 coefficient λ_{L1} on DVS-Gesture. The default 10^{-4} gives the highest accuracy at a low spike rate; accuracy and $\bar{r}$ are otherwise nearly flat across four orders of magnitude.

C.11 Training Dynamics

Figure 9 tracks per-stage spike rates during training. The pattern is depth-dependent and stable: stage 1 settles near $\bar{r} \approx 0.2$ while deeper stages fall well below it. This matters for two claims made elsewhere. It supports the firing-rate range assumed by the accumulate-only accounting of Proposition 3 (§5). It also explains why the spike-driven top-k ranking was informative in the pruning ablation (Table 13): with deep-stage activity this sparse, the saliency signal cleanly separated active from inactive positions, even though pruning to it bought no accuracy.

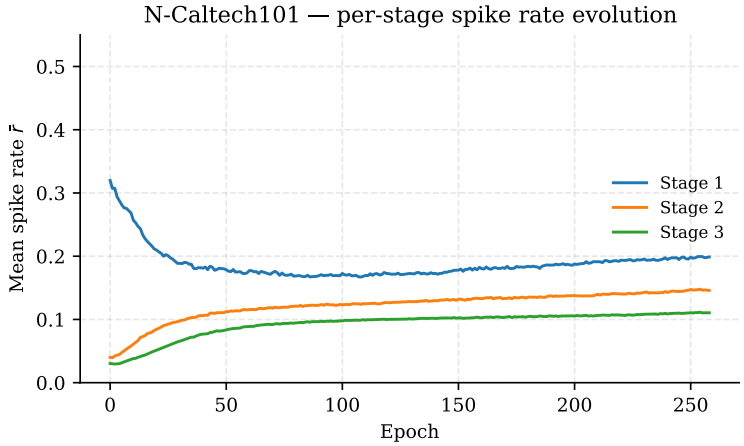


Figure 9: Per-stage spike rate evolution (N-Caltech101; the same depth-dependent pattern holds on DVS-Gesture): stage 1 stays near $\bar{r} \approx 0.2$, deeper stages settle well below it.

C.12 DS-MPA Attention and Per-Class Accuracy

Figure 10 visualizes stage-3 DS-MPA attention across four timesteps of a DVS-Gesture "hand-clap" sample. Two things are visible. The attention map is broadly distributed and peaks at the mid-gesture motion maximum (t=6), then localizes as the hands meet (t=14): the module tracks where evidence is, rather than collapsing onto a fixed region. Every value lies inside $[0, 1]$ here, which is a property of this checkpoint's value stream rather than a prediction of Prop. 2: with V a learned projection of prior spikes, the proposition bounds A by the convex hull of V and the origin, not by $[0, 1]$. The bound is constructive, so this figure is a check that the implementation matches the proof rather than evidence for the bound itself.

Table 15 decomposes the headline 98.48%. Nine of the eleven classes are perfect. The entire error budget sits in *air-drums* and *air-guitar* at 91.67% each, two misclassified clips apiece out of twenty-four. These two are also the classes a human observer finds closest, both sustained two-handed motions without the large directional sweep that separates the arm gestures, so the residual error concentrates in the semantically hardest pair rather than spreading across the task. With 264 test clips (24 per class), one clip is 0.38 pp, so these per-class figures should be read as coarse: the difference between a perfect class and a 91.67% class is just two samples.

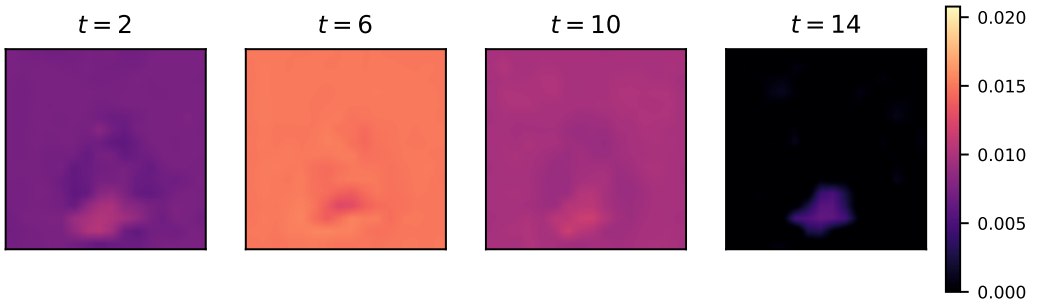


Figure 10: Stage-3 DS-MPA attention A_t (mean over channels) at $t \in \{2, 6, 10, 14\}$ on a DVS-Gesture "hand-clap" sample. Attention is broadly distributed and strongest at the mid-gesture motion peak (t=6), then localizes as the hands meet (t=14); values are dynamic convex-hull bounded (Prop. 2, strictly within $[0, 1]$ for unprojected binary spikes).

Table 15: Per-class top-1 accuracy of RIPE-MambaSpike on DVS-Gesture.

Class	Acc. (%)
hand-clap, R/L-hand wave, R/L-arm cw, R/L-arm ccw, arm-roll, other	100.00
air-drums	91.67
air-guitar	91.67
Mean	**98.48**

Scope of the resolution-independence diagnosis. The coupling we remove is specific to flatten-then-project SNN–to–SSM bridges. Spiking *transformer* baselines are already flat in parameters across input resolutions, so the mechanism we identify does not describe a deficiency of attention architectures generally. It describes only the dense projection that carries a spiking front-end into a state-space scan.

Accumulate-only accounting. A temporal operator placed *before* a convolution silently makes that convolution dense under standard sequential execution, since its operands are no longer binary. The standard synaptic-operation methodology, applied uncritically, would then credit our backbone with accumulate-only cost it does not have in that execution mode. We flag this as a general accounting caveat for any spiking backbone that uses pre-convolution temporal operators, not something specific to our design; §5 (Proposition 3) gives an exact linear-commutation reparameterization that recovers accumulate-only execution for neuromorphic deployment, and we report an energy advantage only under that reparameterization, not for the standard sequential path.

Full limitations. All five headline results are five-seed measurements (98.18 ± 0.31 on DVS-Gesture, 81.28 ± 0.11 on CIFAR10-DVS, 85.49 ± 0.18 on N-Caltech101, 99.55 ± 0.02 on N-MNIST, 45.74 ± 0.40 on DailyDVS-200), as is the parameter-matched sequence-module study of Table 3. The component-wise ablations remain single-seed, and we judge them against those σ rather than treating small Δ as real. On conventional GPU/edge runtimes the model is parameter-efficient only, not energy-efficient: executed sequentially, TDM precedes each backbone convolution, so its operands are real-valued and accumulate-only accounting does not apply, and the dense DS-MPA, bridge and Mamba stages dominate the remaining cost. The accumulate-only property is recoverable for neuromorphic deployment by the exact commutation of Proposition 3, which we verify algebraically and numerically but have not measured on neuromorphic silicon (§5). Weights shrink under fusion,

Table 16: **DS-MPA cost, isolated.** Measured at the exact (B, T, C, H, W) each DS-MPA instance receives in the deployed DVS-Gesture model (B=4 — the batch the recipe trains at — T=16, 128^2 input), summed over the three stages on one RTX 3050; CUDA-event timing, 30 reps after warm-up, activation = peak allocated during one forward+backward. *removed* returns X unchanged with A=0, so the downstream A_t projection is inert. Against a full training step of 405.8 ms, DS-MPA is 7.9% of step time and 4.3% of parameters; substituting the cheap gate recovers only 5.5% of step time while forfeiting the 3.16 pp DS-MPA contributes on N-Caltech101 (Table 2) — the largest single-component effect we measure. Two properties make this more than a single-GPU anecdote. First, the DS-MPA/gate *ratio* is 3.30/3.33/3.31/3.28 at B=1/2/4/8, stable to ±0.8% across an 8× change in utilisation, and cost scales linearly with B from B=1, so the operator is compute-bound rather than launch-bound: the ratio reflects arithmetic, not this device's idle time. Absolute latencies are of course hardware-specific; the parameter and activation figures are not. Second, the $O(NC^2)$ term does *not* grow with channel width here: N falls 16× as C rises 4×, so NC^2 is constant (4.19 M per stage) and measured forward time *decreases* with depth.

Channel-mixing operator	Params	Fwd (ms)	Fwd+bwd (ms)	Act. (MiB)
DS-MPA (ours)	64,736	11.48	32.26	790.6
1×1 conv + sigmoid gate	21,728	3.50	9.75	336.0
removed (identity)	0	0.29	1.30	112.0
does the $O(NC^2)$ term worsen with width? DS-MPA at B=1:				
[32, 64, 128]	64,736	3.11	8.87	197.7
[64, 128, 256]	258,496	5.56	16.12	397.6
[128, 256, 512]	1,033,088	12.21	34.76	805.4

Table 17: Measured deployment cost on DVS-Gesture (T=16, 128×128, fused; CUDA events, median of 30 runs after 10 warm-ups, RTX 3050). "Weights" is parameters + buffers; "act." is peak allocation minus weights. RIPE-MambaSpike is 41× smaller in weights and its fused graph is 1.53× faster than its training graph, but its activations are larger than the baseline's — the trade is weight memory for activation memory and batched throughput.

Model	Weights	B=1 lat.	B=8 lat.	B=8 peak / act.
Mamba-Spike (baseline)	138.3 MiB	38.2 ms	**51.5 ms**	**456 / 272 MiB**
RIPE-MambaSpike, training graph	5.9 MiB	36.5 ms	159.3 ms	1073 / 1043 MiB
RIPE-MambaSpike, deployed	**3.4 MiB**	**34.2 ms**	103.8 ms	1070 / 1043 MiB

but activation memory grows under batching. The selective-scan kernel is CUDA-specific, and quantized deployment on neuromorphic silicon is untested, though the fused 3×3 backbone is directly amenable to it. Several design choices are reported as deployed rather than swept: alternative DS-MPA feature maps, per-dataset k, the grouped/low-rank DS-MPA variants of Table 18 (untrained), and the bridge width d_{m} and pooling schedule. The dual-kernel accumulate-only path of Proposition 3 is likewise proven and numerically verified rather than benchmarked on hardware. DailyDVS-200 uses uniform event-to-frame binning only, and sensitivity to that choice is unverified. The DailyDVS-200 lite/wide pair (§6) is an axis distinct from the resolution-invariance claim of Fig. 2: that claim fixes the bridge width d_{m} and varies $H \times W$, while the lite/wide pair fixes $H \times W = 224^2$ and deliberately varies d_{m} as a capacity choice.

C.13 Latency and Memory

Table 17 reports CUDA-event latency and peak memory. Fusing the multi-branch graph cuts B=8 latency 1.53×, beyond the 1.74× parameter cut. At B=1, the edge operating point, RIPE-MambaSpike's total footprint is below the baseline's, since the baseline's 138 MiB of weights dominates there. That ordering reverses at B=8: RIPE-MambaSpike's activations are ∼3.8× larger, so it becomes slower and larger in total memory under batching. The trade is weight memory for activation memory and batched throughput.

Table 18: **Cheaper DS-MPA parameterizations are available and preserve the bound, but we do not adopt them.** The bound is structural: $\phi \geq 0$ and the normalizer make $A_{i,\cdot}$ a convex combination of the rows of V (Prop. 2), which holds for *any* factorization of W_Q, W_K, W_V — verified at every stage width by checking each $A_{i,c}$ against the componentwise range of $V_{\cdot,c}$ that variant produces $(9/9)$. So the $O(C^2)$ projection cost *can* be reduced without theoretical loss. We nevertheless deploy the full operator, for a reason the last column makes explicit: DS-MPA is only 7.4% of the deployed $875{,}122$ parameters, so even the $4\times$ reduction moves the headline figure by 5.5%, while removing DS-MPA altogether costs 3.16 pp on N-Caltech101 (Table 2) — the largest single-component effect we measure. We are not willing to risk a measured 3.16 pp for an unmeasured 5.5% of weight storage, and we have not trained these variants, so we make no accuracy claim for them. They are reported as a sound design option, not as a result.

DS-MPA parameterization	Params	vs. dep.	Bound	Deployed total
deployed: $3 \times (C \to C)$	64,736	$1.00\times$	✓	$\mathbf{875{,}122}$
low-rank: $3 \times (C \to \frac{C}{4} \to C)$	32,480	$0.50\times$	✓	842,866 (-3.7%)
grouped: $3 \times (C \to C)$, g=4	16,352	$0.25\times$	✓	826,738 (-5.5%)

Table 19: **What resolution-independence does and does not buy.** Complements Table 6: deployed parameters are *constant* across a $43\times$ range in pixel count, but activation memory and latency are not — they scale with resolution as in any convolutional model. Measured on one RTX 3050, T=16, batch 4, RIPE-MambaSpike's DVS-Gesture configuration; parameters counted after fusion. †At 224^2 the training step reaches this card's 8 GiB capacity, so the training throughput there is allocator-limited rather than a clean measurement (inference, which needs no gradients, still runs at 18.2 samples/s). This is the trade the paper claims and no more: weight storage is decoupled from resolution, total memory is not.

Input resolution	Deployed (M)	Train (samp/s)	Infer (samp/s)	Peak (GiB)
34×34	**0.875**	25.4	102.0	0.27
48×48	**0.875**	25.4	83.4	0.45
128×128	**0.875**	10.3	50.1	2.67
224×224	**0.875**	$0.8^\dagger$	18.2	$8.00^\dagger$

Table 20: **Compute cost and how it scales with k, T, C and resolution.** FLOPs are *measured* with PyTorch's flop counter on the fused inference graph (torch's convention: one multiply–accumulate is 2 FLOPs; MACs = FLOPs/2), so the selective scan's matmuls and the DS-MPA contraction are counted as executed rather than derived from a formula. Batch 1; the deployed DVS-Gesture point is $[32, 64, 128]$, T=16, 128^2, all tokens scanned. Three things are worth reading off this table. *First*, an $8\times$ change in the SSM sequence length ($32 \to 256$ tokens per timestep) costs only 11% more compute, because the convolutional front-end dominates and not the scan — so the scan's length is not the lever that matters here. *Second*, cost is exactly linear in T and exactly quadratic in C ($4\times$ per width doubling). *Third*, deployed *parameters* are flat across resolution while compute is not: this is the same trade Table 19 makes for memory. The small parameter variation down the T block is BNTT, whose affine terms are per-timestep. This is a compute count and *not* an energy claim: TDM precedes each backbone convolution, so accumulate-only accounting does not apply (§5), and arithmetic counts say nothing about memory traffic.

Configuration	Params (M)	GFLOPs	GMACs
deployed (T=16, 128^2)	**0.875**	**9.00**	**4.50**
SSM sequence length (tokens entering the scan; T=16, 128^2)			
L/T=32	0.875	8.11	4.05
L/T=128	0.875	8.49	4.25
L/T=256	0.875	9.00	4.50
time bins T (128^2)			
T=4	0.864	2.06	1.03
T=8	0.868	4.12	2.06
T=32	0.890	16.47	8.23
stage widths C (T=16, 128^2)			
$[64, 128, 256]$	3.361	32.73	16.36
$[128, 256, 512]$	13.165	130.49	65.25
input resolution (T=16) — params flat, compute not			
34×34	0.875	0.78	0.39
48×48	0.875	1.27	0.63
224×224	0.875	24.69	12.34