Video Len.	Ctx. Len.	Trainable Parameters	LR	Schedule	Steps
3 sec	18048	TTT / Pre-trained Params	1×10^{-4} / 1×10^{-5}	Cosine / Constant	5000
9 sec	51456	TTT + Local Attn (QKVO)	1×10^{-5}	Constant	5000
18 sec	99894	TTT + Local Attn (QKVO)	1×10^{-5}	Constant	1000
30 sec	168320	TTT + Local Attn (QKVO)	1×10^{-5}	Constant	500
63 sec	341550	TTT + Local Attn (QKVO)	1×10^{-5}	Constant	250

Table 2. Model configurations for multi-stage fine-tuning. First, the entire pre-trained model is adapted to 3-second segments of *Tom and Jerry*, with higher learning rates assigned to newly introduced TTT layers and gates. After, only TTT layers, gates, and self-attention parameters are fine-tuned at reduced learning rates to preserve general pre-trained knowledge.

A. Experiment Details

Extension stages. For stability, we extend to one-minute generations using a stage-based approach. First, we perform supervised fine-tuning on 3-second segments. We interpret this as a "style transfer" phase to adapt the model to *Tom and Jerry* animations. During this phase, new parameters are trained with a higher learning rate on a cosine decay schedule. When beginning the 9-second stage, we preserve the same number of training steps used in the initial 3-second stage, as this is the model's first exposure to multiple scenes. Each successive extension stage then uses progressively fewer training steps (Table 2).

Diffusion schedule. We fine-tune our models following CogVideoX [54] using v-prediction [34]. This includes a diffusion noise schedule with 1000 steps and Zero-SNR [27] enforced at the final step.

Training configurations. We use the following hyperparameters for all stages of training:

- **Optimizer:** AdamW with $(\beta_1, \beta_2) = (0.9, 0.95)$
- **Learning Rate:** Linear warmup over 2% of training steps
- **Batch Size:** 64
- **Gradient Clipping:** 0.1
- **Weight Decay:** 10^{-4} applied to all params except biases and normalization layers
- **VAE Scale Factor**: 1.0
- **Dropout:** Zero-out text prompt with probability 0.1
- **Precision:** Mixed Precision with PyTorch FSDP2

TTT configurations. A key hyperparameter for TTT layers is η, which determines the learning rate applied to the hidden state model f. We set $\eta = 1.0$ for TTT-Linear and $\eta = 0.1$ for TTT-MLP.

Sampling schedule. We follow the DDIM sampler [41] with 50 steps, applying dynamic classifier-free guidance (CFG) [18] that increases CFG magnitude from 1 to 4 and utilizing negative prompts to further enhance video quality.

B. On-Chip Tensor Parallel Details

We use ThunderKittens [42] to implement the TTT-MLP kernel.

Hidden state sharding. As described in Section 3.4, TTT-MLP is a two-layer MLP with a 4× feature expansion. We follow a standard tensor-parallel strategy, sharding the first layer column-wise and the second layer row-wise. As the GeLU non-linearity is elementwise, the forward pass of the TTT-layer requires a single reduction for computing the inner loss used to update the hidden state.

Further latency optimizations. We incorporate several techniques from FlashAttention-3 [38] to further reduce I/O latency on NVIDIA Hopper GPUs. In particular, we implement a multi-stage pipelining scheme that asynchronously prefetches future mini-batches from HBM, overlapping data transfers with computation on the current mini-batch. This approach, known as *producer-consumer asynchrony*, involves dedicating specialized warpgroups to either data loading (producer) or computation (consumer).

Additionally, we integrate gradient checkpointing along the sequence directly into our fused kernel. To reduce I/O-induced stalls and lighten CUDA thread workloads, we leverage the Tensor Memory Accelerator (TMA) to perform asynchronous memory stores.

C. More Experiment Results

As mentioned in Subsection 4.2, we also evaluate methods at 18-seconds (Table 3), where context length is roughly 100k tokens and modern RNN layers remain competitive. Gated DeltaNet achieves the best overall performance (+29 ELO points). TTT-MLP, TTT-Linear, and Mamba 2 achieve comparable results. Local Attention performs the worst, particularly struggling with scene consistency, highlighting the importance of incorporating global sequence modeling layers for coherent video generation.

	Text Alignment	Motion Smoothness	Aesthetics	Scene Consistency	Average
Local Attention	965	972	969	944	962.50
TTT-Linear	1003	995	1007	1001	1001.50
Mamba 2	**1023**	987	1008	1004	1005.50
Gated DeltaNet	1020	**1039**	**1044**	**1026**	**1032.25**
SWA	995	1004	993	980	993.0
TTT-MLP	994	1002	1002	1019	1004.25

Table 3. Human evaluation results for 18 second video generation. Gated DeltaNet outperforms baselines by an average of +26 ELO points over the next-best model, notably in aesthetics (+36) and motion smoothness (+35). Local Attention performs the worst overall, while the remaining baselines (TTT-Linear, Mamba 2, SWA, and TTT-MLP) show similar performance.

Format 1

Tom is happily eating an apple pie at the kitchen table. Jerry looks longingly wishing he had some. Jerry goes outside the front door of the house and rings the doorbell. While Tom comes to open the door, Jerry runs around the back to the kitchen. Jerry steals Tom's apple pie. Jerry runs to his mouse hole carrying the pie, while Tom is chasing him. Just as Tom is about to catch Jerry, Jerry makes it through the mouse hole and Tom slams into the wall.

Format 2

Segment 1-2: Tom walks into the kitchen carrying an apple pie. He sits at the table and begins eating.

Segment 3-5: The viewpoint shifts behind the countertop, revealing Jerry hiding behind a salt shaker. Jerry steps out, watches Tom eating the pie, and eagerly rubs his tummy. He then darts off-screen to the right.

Segment 6-8: Outside the house, Jerry approaches the front door, jumps to press the doorbell, and quickly runs away.

The story continues...

Format 3

<start_scene>The kitchen has soft yellow walls, white cabinets, and a window with red-and-white checkered curtains letting in gentle sunlight. In the middle, there's a round wooden table with matching chairs, sitting on a clean white-tiled floor. Tom, the blue-gray cat, walks in from the left holding a warm, golden-brown pie on a shiny silver tray. He moves calmly across the room toward the table, carefully places the pie down, pulls out a chair, and sits comfortably. The camera smoothly follows Tom from left to right, clearly showing each of his movements.

The kitchen has soft yellow walls, white cabinets, and a window with red-and-white checkered curtains letting in gentle sunlight. In the middle, there's a round wooden table with matching chairs, sitting on a clean white-tiled floor. Tom, the blue-gray cat, sits comfortably at the table with the golden-brown pie resting on its shiny silver tray directly in front of him. He carefully uses his paw to pick up a slice from the tray, lifts it toward his mouth, and takes a large bite. The camera slowly moves closer, clearly showing Tom enjoying his pie as crumbs lightly fall onto the table.<end_scene>

<start_scene>The kitchen has soft yellow walls, white cabinets, and a window with red-and-white checkered curtains letting in gentle sunlight. The cabinets have white countertops, on which a tall glass salt shaker is sitting. In the background, Tom, the blue-gray cat, sits at the round wooden table, eating the golden-brown pie. Jerry, the brown mouse, stands on the white countertop, hidden behind the salt shaker. Jerry glances around briefly, then steps out from behind the salt shaker. The camera captures Jerry as he emerges from behind the salt shaker and stands on the countertop.

The kitchen has soft yellow walls, white cabinets, and a window with red-and-white checkered curtains letting in gentle sunlight. The cabinets have white countertops, on which a tall glass salt shaker is sitting. In the background, Tom, the blue-gray cat, sits at the round wooden table, eating the golden-brown pie. Jerry, the brown mouse, stands on top of the countertop next to the salt shaker. Jerry rubs his stomach with his paws and looks at Tom, the blue-gray cat. The camera remains in position slightly to the side of Jerry, capturing his hungry expression.

The kitchen has soft yellow walls, white cabinets, and a window with red-and-white checkered curtains letting in gentle sunlight. The cabinets have white countertops, on which a tall glass salt shaker is sitting. In the background, Tom, the blue-gray cat, sits at the round wooden table, eating the golden-brown pie. Jerry, the brown mouse, stands on top of the countertop next to the salt shaker. Jerry gives his belly a final rub, then turns to the left and quickly begins running along the countertop toward the right side of the scene. The camera captures Jerry as he disappears off-screen.<end_scene>

<start_scene>The front of the house has light blue walls, a white wooden front door, a small round white doorbell button beside it, and a small porch with steps leading down to a neat green lawn. Bright flowers in red and yellow line the walkway, and sunlight warmly fills the area. Jerry, the brown mouse, calmly walks up onto the porch from the right side, moving toward the front door and doorbell at the center of the scene. The camera smoothly tracks Jerry's steps, capturing clearly as he crosses the porch and comes to a gentle stop near the steps, glancing cautiously upward at the doorbell.

The front of the house has light blue walls, a white wooden front door, a small round white doorbell button beside it, and a small porch with steps leading down to a neat green lawn. Bright flowers in red and yellow line the walkway, and sunlight warmly fills the area. Jerry, the brown mouse, stands close to the front door, looking upward toward the doorbell button mounted just above his head. He takes two quick jumps upward, reaching with an extended paw, and on his third jump, firmly presses the doorbell. The camera follows Jerry’s jumps closely, clearly capturing each leap and the exact moment he presses the doorbell.

The story continues...

Figure 6. Illustration of the three prompt formats used for model input: (1) short plot summaries, (2) sentence-level plots, and (3) detailed scene storyboards. All inputs are eventually converted to the detailed storyboard format (Format 3) for both fine-tuning and inference.