Mamba Fusion: Learning Actions Through Questioning

Apoorva Beedu[*‡], Zhikang Dong[*‡†], Jason Sheinkopf[‡], Irfan Essa[‡⇳]

Georgia Institute of Technology[‡], Stony Brook University[†], Google Research[⇳]

[*] Indicates equal contribution

Challenges Faced By Transformers

Quadratic computational complexity

High GPU memory usage

Difficulty capturing long-term dependencies

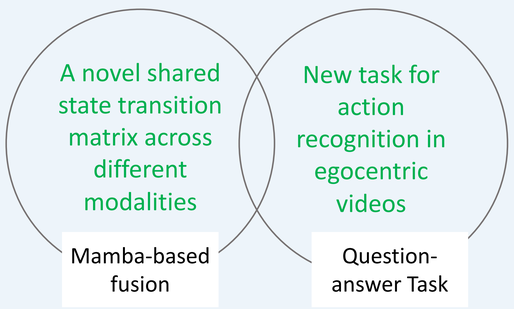
Selective State Space Models like Mamba are a solution!

LLM Descriptions for an action – Open Door

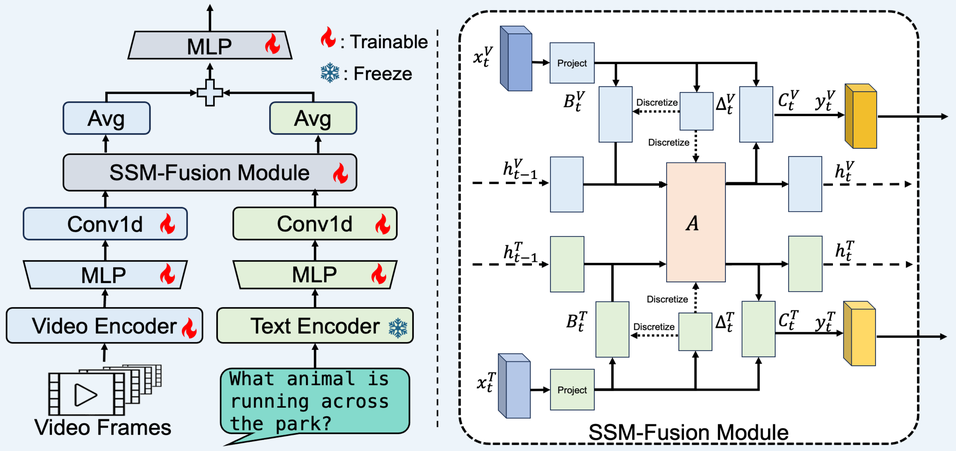
1.Grasp the handle, apply pressure, and shift the barrier aside to reveal an entryway.
2.Turn the knob, push or pull, and make way for movement through the passage.
3.Unlatch the panel, move it out of the way, and step through the opening.
4.Push against the wooden slab, allowing space to pass through.
5.Rotate the handle, displace the obstruction, and access the next area.
6.Release the latch, shift the divider, and enter the adjacent space.
7.Pull on the frame, creating an opening to step forward.
8.Apply force to the entryway's barrier, making room to pass.
9.Press against the surface, moving it aside to clear the way.
10.Twist the lever, slide or swing the partition, and proceed through.

- Captions should not contain the actions for end-to-end training!
- LLM based caption generation does not provide the right context!

Solution!



MambaVL Pipeline



Algorithm 1 SSM-Fusion Module

Require: $\mathbf{x}^V$: (B, F, D), $\mathbf{x}^T$: (B, L, D)
Ensure: $\mathbf{y}^V$: (B, F, D), $\mathbf{y}^T$: (B, L, D)

$\boldsymbol{A}$: (D, N) ← Parameter
$\boldsymbol{B}^V$: (B, F, N) ← $\text{Linear}_{\text{B}}^V(\mathbf{x}^V)$
$\boldsymbol{B}^T$: (B, L, N) ← $\text{Linear}_{\text{B}}^T(\mathbf{x}^T)$
$\boldsymbol{C}^V$: (B, F, N) ← $\text{Linear}_{\text{C}}^V(\mathbf{x}^V)$
$\boldsymbol{C}^T$: (B, L, N) ← $\text{Linear}_{\text{C}}^T(\mathbf{x}^T)$
Δ^V: (B, F, D) ← $\text{Softplus}(\text{Parameter} + \text{Linear}_{\Delta}^V(\mathbf{x}^V))$
Δ^T: (B, L, D) ← $\text{Softplus}(\text{Parameter} + \text{Linear}_{\Delta}^T(\mathbf{x}^T))$
$\overline{\boldsymbol{A}^V}, \overline{\boldsymbol{B}^V}$: (B, F, D, N) ← $\text{Discretize}(\Delta^V, \boldsymbol{A}, \boldsymbol{B}^V)$
$\overline{\boldsymbol{A}^T}, \overline{\boldsymbol{B}^T}$: (B, L, D, N) ← $\text{Discretize}(\Delta^T, \boldsymbol{A}, \boldsymbol{B}^T)$
$\mathbf{y^V}$: (B, F, D) ← $\text{SSM}(\overline{\boldsymbol{A}^V}, \overline{\boldsymbol{B}^V}, C^V)$
$\mathbf{y^T}$: (B, L, D) ← $\text{SSM}(\overline{\boldsymbol{A}^T}, \overline{\boldsymbol{B}^T}, C^T)$
return $\mathbf{y}^V$ and $\mathbf{y}^T$

- Videos and Questions are used as inputs to train MambaVL to recognize actions.
- State Transition Matrix A is shared across modalities
- Projection matrices (B and C) and Transition matrix Δ are modality specific!
- when the action is "Open Door", we replace the words -- "Open" or "Door" -- with a <MASK> token

EK100 Action Recognition

Model(Backbone)	Pretrain data	Verb	Noun	Action
MeMViT (24x3)	K600	71.4	60.3	48.4
Omnivore (swin-B)	IN-(21k+1k)+K400+SUN	69.5	61.7	49.9
MeMViT (16x4)	K400	70.6	58.5	46.2
ORViT (MF-HR)	IN-21k+K400	68.4	58.7	45.7
MambaVL (ORViT)	IN-21k+K400	**69.1**	**63.9**	**48.6**
AVION (VIT-B)	WIT + Ego4D	70.0	59.8	**49.1**
LaViLa (TSF-B)	WIT + Ego4D	69.0	58.4	46.9
MambaVL (ViT-B)	WIT + Ego4D	**70.9**	**61.1**	**49.1**
AVION (ViT-L)	WIT + Ego4D	73.0	65.4	54.4
LaViLa (TSF-L)	WIT + Ego4D	72.0	62.9	51.0
MambaVL (ViT-L)	WIT + Ego4D	**74.3**	**67.1**	**55.0**

Action Recognition: across model-sizes, MambaVL is stronger.

EK100 Action Anticipation

Method	Pretrain data	Overall		
		Verb	Noun	Action
AVT+ [34]	IN21K + EPIC boxes	28.2	32.0	15.9
MeMViT (32x3) [20]	K700	**32.2**	**37.0**	17.7
MeMViT (16x4) [20]	K400	**32.8**	33.2	15.1
AFFT [35]	IN-21K	22.8	34.6	18.5
ORViT-MF [31]	IN-21k+K400	26.9	34.2	**23.3**
MambaVL (ORViT)	IN-21k+K400	29.1	35.1	**23.9**

Action Anticipation: MambaVL performs better than base model ORViT

Fusion Method Comparison

Fusion Method	Verb	Noun	Action
MLP	62.8	51.6	39.6
Transformer (6x4)	62.9	51.9	40.0
Transformer (12x12)	62.5	51.8	39.5
MambaVL	69.1	63.9	48.6

MambaVL performs better than other MLP and Transformer methods!

Model	GFLOPS	Params
ORViT	405.0	148M
ORViT + Transformer Fusion	413.5	242M
MambaVL	413.0	157M

Mamba fusion block adds negligible number of FLOPs and trainable parameters!

Conclusion

- Mamba: Linear complexity for long-range sequence modeling
- Effective cross-modal information sharing
- Flexible integration with existing methods
- New Task! Question-Answering for Action Recognition