

Demystify Mamba in Vision: A Linear Attention Perspective

Dongchen Han Ziyi Wang Zhuofan Xia Yizeng Han Yifan Pu Chunjiang Ge
Jun Song Shiji Song Bo Zheng Gao Huang



清华大学
Tsinghua University

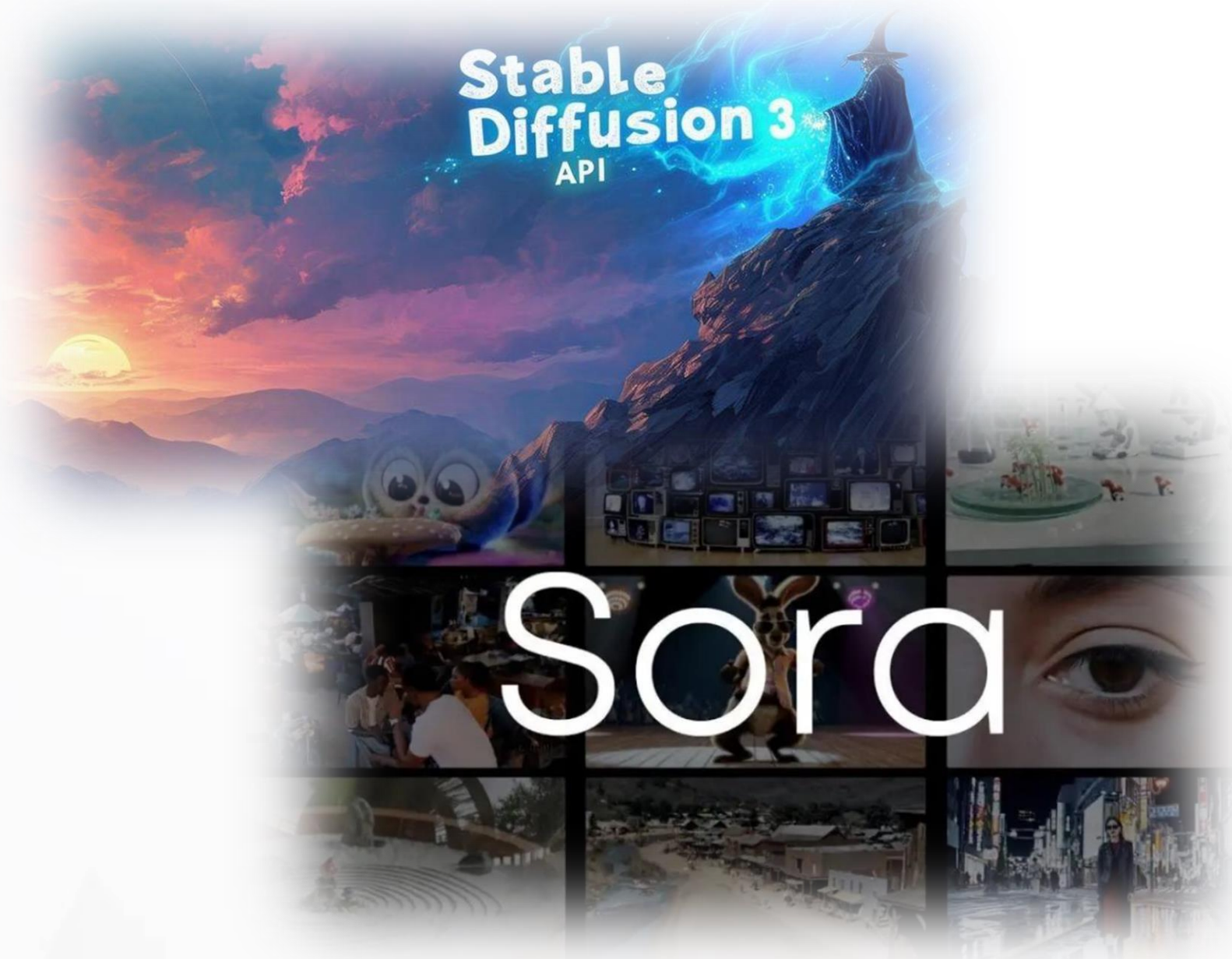


Background



Transformers has a **Quadratic Complexity** $\mathcal{O}(N^2 d)$ *with respect to sequence length.*

High Resolution Images



Videos



Mamba: a Powerful Selective State Space Model



Mamba

- ✓ *High expressive capability*
- ✓ *Linear complexity $\mathcal{O}(Nd^2)$*
- ✓ *Global modeling*

A promising method to deal with high-resolution images!

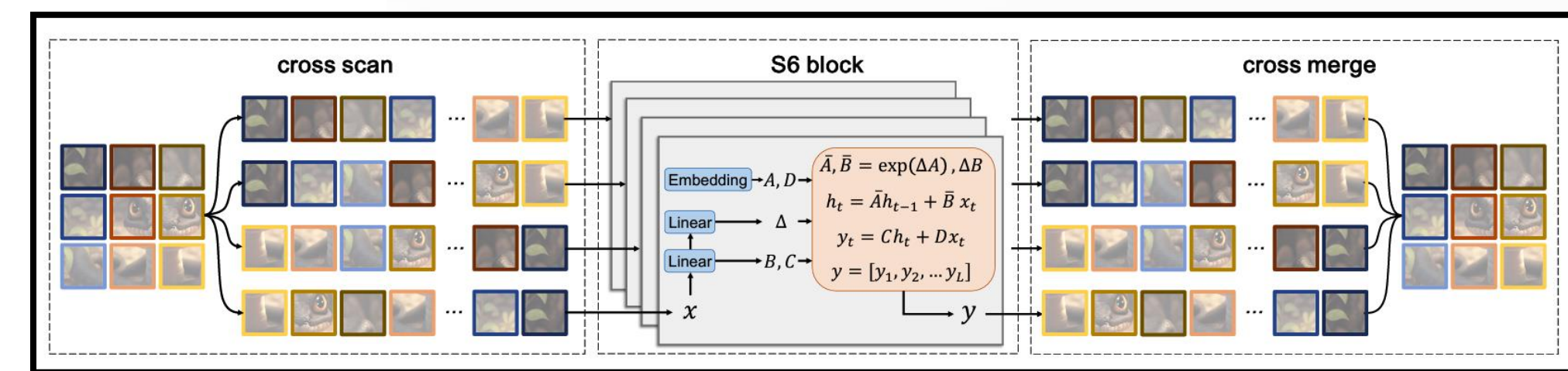
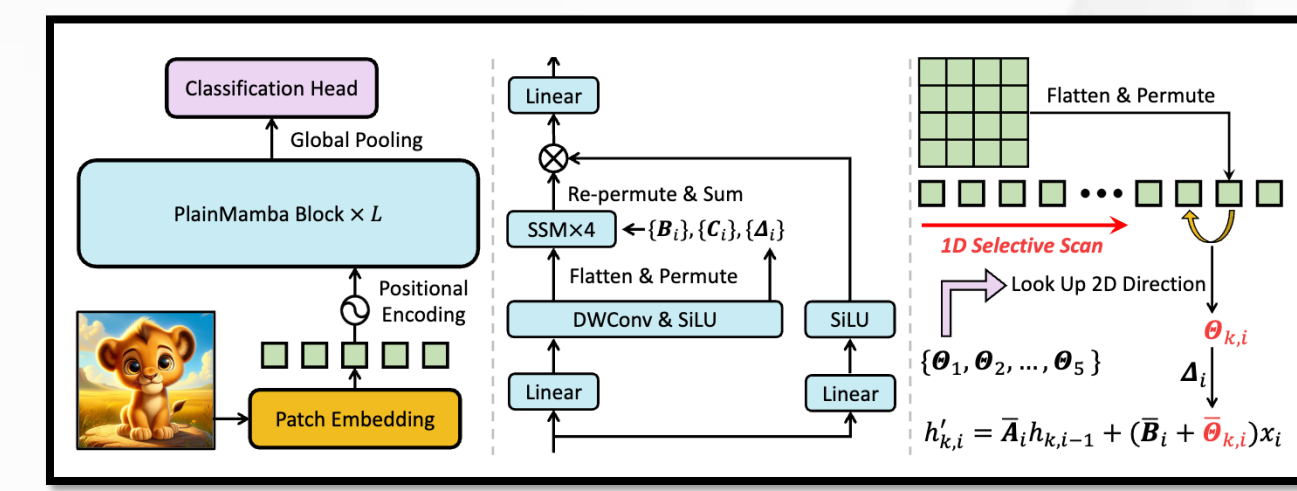
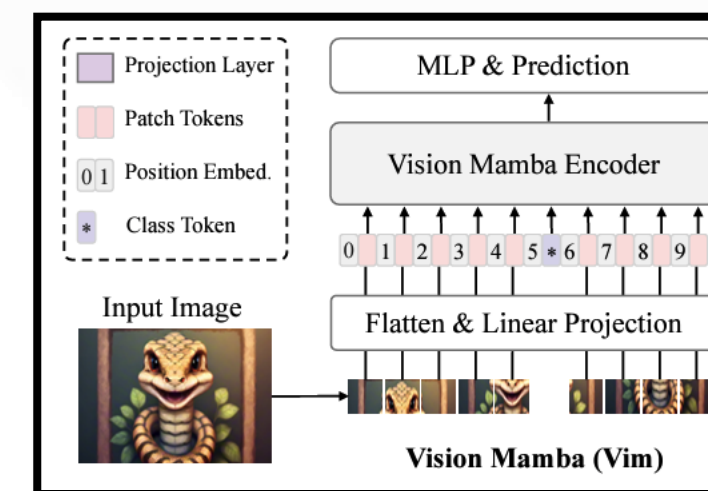


Mamba: Linear-time sequence modeling with selective state spaces

A Gu, T Dao - arXiv preprint arXiv:2312.00752, 2023 - arxiv.org

... to million-length **sequences**. As a general **sequence model** backbone, **Mamba** achieves state-... On language **modeling**, our **Mamba-3B model** outperforms Transformers of the same

☆ Save 📄 Cite **Cited by 339** Related articles All 4 versions ⇨





Mamba

- ✓ Linear complexity $\mathcal{O}(N)$
- ✓ Global modeling
- ✓ **High expressive capability**

$$\begin{aligned} \mathbf{h}_i &= \tilde{\mathbf{A}}_i \odot \mathbf{h}_{i-1} + \mathbf{B}_i(\Delta_i \odot \mathbf{x}_i), \\ \mathbf{y}_i &= \mathbf{C}_i \mathbf{h}_i / 1 + \mathbf{D} \odot \mathbf{x}_i. \end{aligned}$$



Linear Attention

- ✓ Linear complexity $\mathcal{O}(N)$
- ✓ Global modeling
- × **Inferior performance**

$$\mathbf{y}_i = \sum_{j=1}^N \frac{Q_i \mathbf{K}_j^\top}{\sum_{j=1}^N Q_i \mathbf{K}_j^\top} \mathbf{V}_j = \frac{Q_i \left(\sum_{j=1}^N \mathbf{K}_j^\top \mathbf{V}_j \right)}{Q_i \left(\sum_{j=1}^N \mathbf{K}_j^\top \right)}$$

Linear Attention

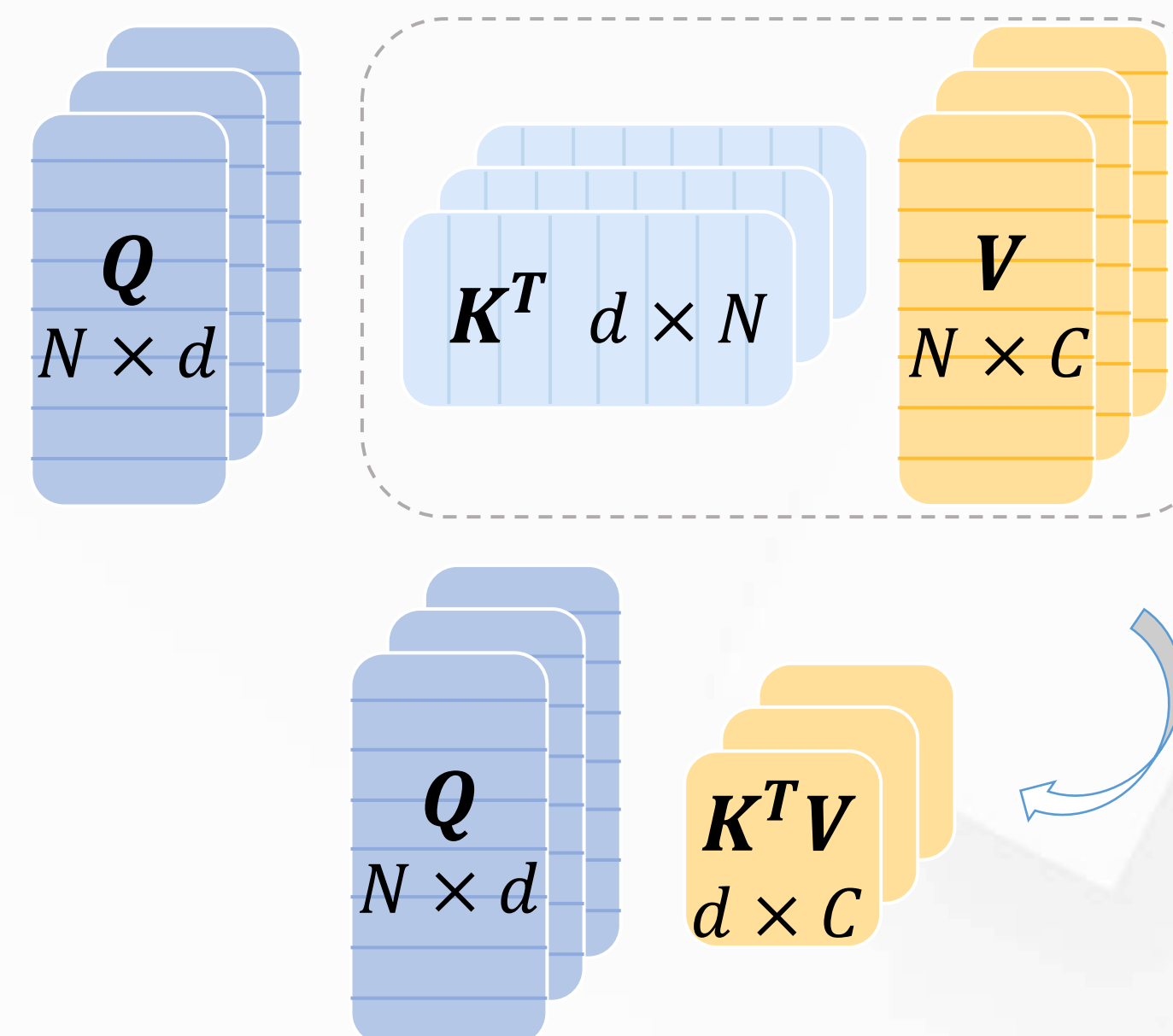


Linear Attention $O = QK^T V$

Carefully designed kernels are introduced as the approximation of the original similarity function:

$$Q = \phi(xW_Q), K = \phi(xW_K), V = xW_V$$

$$y_i = \sum_{j=1}^N \frac{Q_i K_j^T}{\sum_{j=1}^N Q_i K_j^T} V_j = \frac{Q_i \left(\sum_{j=1}^N K_j^T V_j \right)}{Q_i \left(\sum_{j=1}^N K_j^T \right)}$$



- ✗ **Inferior performance**
- ✓ Linear complexity $O(Nd^2)$

Recurrent Linear Attention

paper



code



Non-causal linear attention (common linear attention):

$$y_i = \sum_{j=1}^N \frac{Q_i K_j^\top}{\sum_{j=1}^N Q_i K_j^\top} V_j = \frac{Q_i \left(\sum_{j=1}^N K_j^\top V_j \right)}{Q_i \left(\sum_{j=1}^N K_j^\top \right)}$$

Causal linear attention:

$$y_i = \frac{Q_i \left(\sum_{j=1}^i K_j^\top V_j \right)}{Q_i \left(\sum_{j=1}^i K_j^\top \right)} \triangleq \frac{Q_i S_i}{Q_i Z_i}, \quad S_i = \sum_{j=1}^i K_j^\top V_j, \quad Z_i = \sum_{j=1}^i K_j^\top$$

Recurrent linear attention form:

$$S_i = S_{i-1} + K_i^\top V_i, \quad Z_i = Z_{i-1} + K_i^\top, \quad y_i = Q_i S_i / Q_i Z_i.$$

$$S_i = S_{i-1} + K_i^\top \cdot V_i ;$$

$$Z_i = Z_{i-1} + K_i^\top$$

$$y_i = \frac{Q_i \cdot S_i}{Q_i \cdot Z_i}$$

Selective State Space Model (Scalar Input)

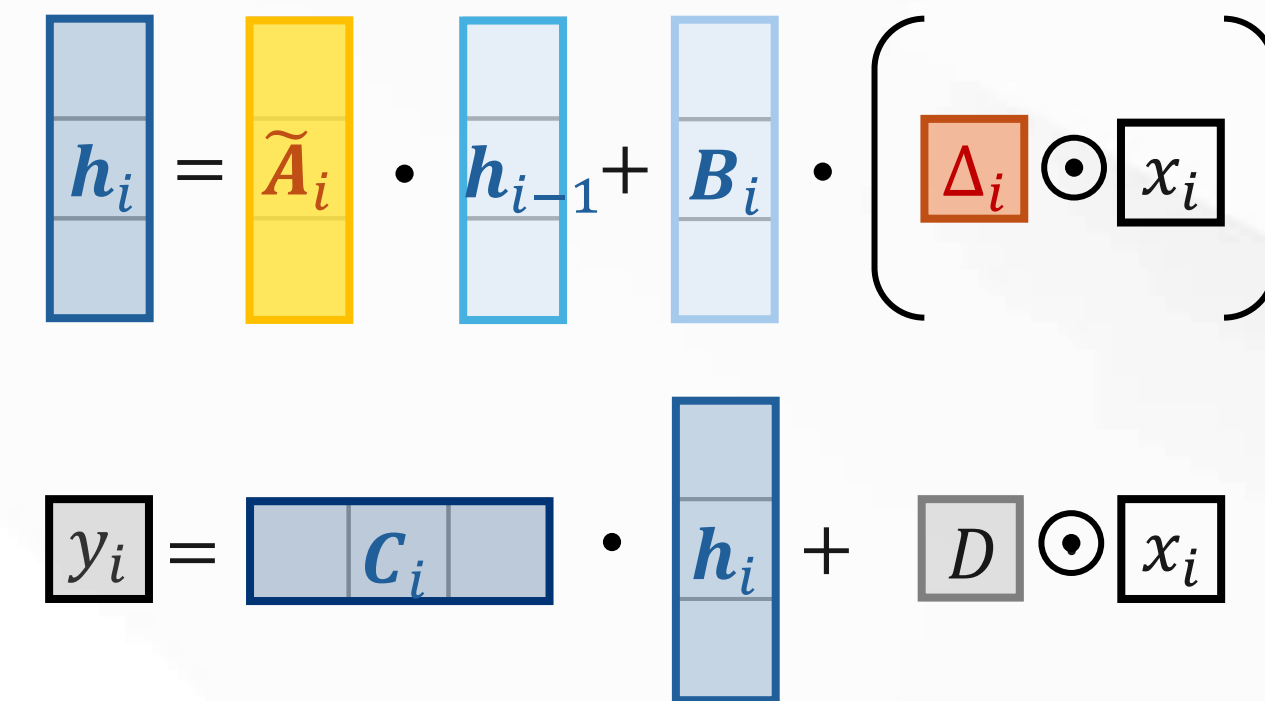
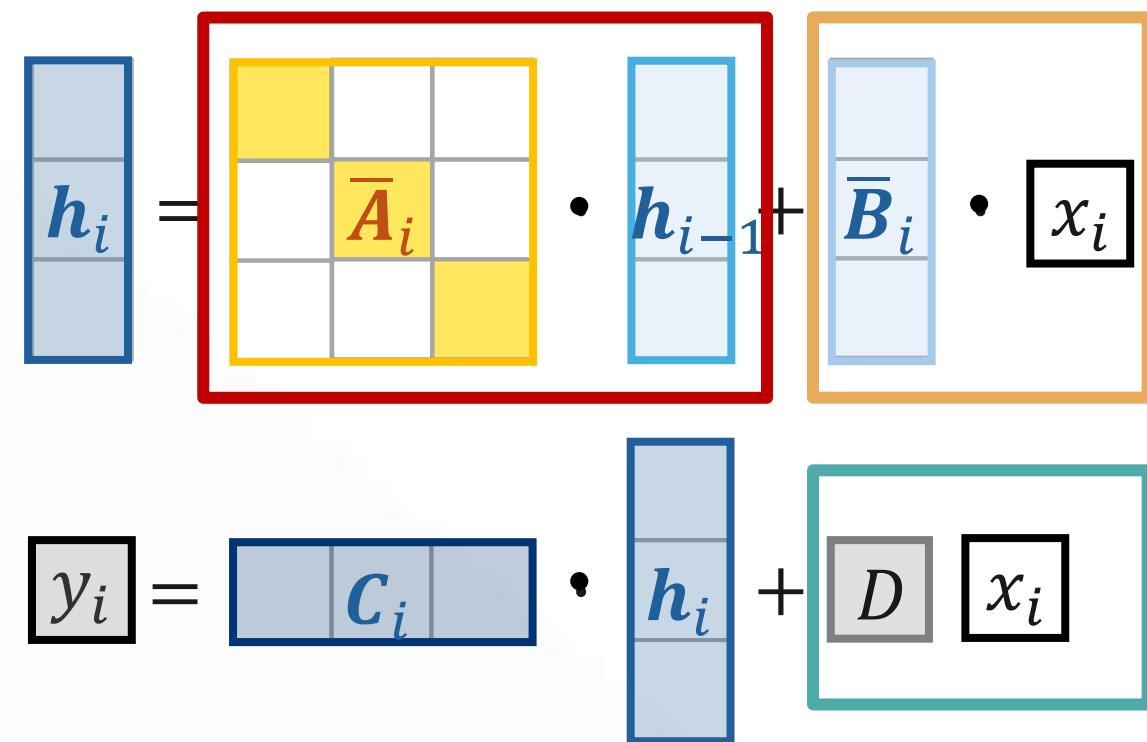


$$\begin{aligned} \mathbf{h}_i &= \bar{\mathbf{A}}_i \mathbf{h}_{i-1} + \bar{\mathbf{B}}_i x_i, \\ y_i &= \mathbf{C}_i \mathbf{h}_i + D x_i, \end{aligned}$$

$$\begin{aligned} x_i \in \mathbb{R}, \quad \bar{\mathbf{A}}_i \in \mathbb{R}^{d \times d}, \quad \bar{\mathbf{B}}_i, \mathbf{h}_{i-1}, \mathbf{h}_i \in \mathbb{R}^{d \times 1}, \\ y_i \in \mathbb{R}, \quad \mathbf{C}_i \in \mathbb{R}^{1 \times d}, \quad D \in \mathbb{R}. \end{aligned}$$

$$\begin{aligned} \mathbf{h}_i &= \tilde{\mathbf{A}}_i \odot \mathbf{h}_{i-1} + \mathbf{B}_i (\Delta_i \odot x_i), \\ y_i &= \mathbf{C}_i \mathbf{h}_i + D \odot x_i, \end{aligned}$$

$$\begin{aligned} x_i, \Delta_i \in \mathbb{R}, \quad \tilde{\mathbf{A}}_i, \mathbf{B}_i, \mathbf{h}_{i-1}, \mathbf{h}_i \in \mathbb{R}^{d \times 1}, \\ y_i \in \mathbb{R}, \quad \mathbf{C}_i \in \mathbb{R}^{1 \times d}, \quad D \in \mathbb{R}. \end{aligned}$$



(1) $\bar{\mathbf{A}}_i \mathbf{h}_{i-1} = \tilde{\mathbf{A}}_i \odot \mathbf{h}_{i-1}$

(2) $\bar{\mathbf{B}}_i x_i = \Delta_i \mathbf{B}_i x_i = \mathbf{B}_i (\Delta_i x_i) = \mathbf{B}_i (\Delta_i \odot x_i)$

(3) $D x_i = D \odot x_i$

Selective State Space Model (Vector Input)

paper



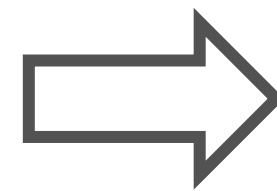
code



$$\begin{aligned} h_i &= \tilde{A}_i \odot h_{i-1} + B_i(\Delta_i \odot x_i), & x_i, \Delta_i &\in \mathbb{R}^{1 \times C}, \tilde{A}_i, h_{i-1}, h_i \in \mathbb{R}^{d \times C}, B_i \in \mathbb{R}^{d \times 1} \\ y_i &= C_i h_i + D \odot x_i, & y_i &\in \mathbb{R}^{1 \times C}, C_i \in \mathbb{R}^{1 \times d}, D \in \mathbb{R}^{1 \times C}, \end{aligned}$$

$$h_i = \tilde{A}_i \cdot h_{i-1} + B_i \cdot \left(\Delta_i \odot x_i \right)$$

$$y_i = C_i \cdot h_i + D \odot x_i$$



$$h_i = \tilde{A}_i \odot h_{i-1} + B_i \cdot \left(\Delta_i \odot x_i \right)$$

$$y_i = C_i \cdot h_i + D \odot x_i$$

Mamba v.s. Linear Attention Transformer

paper

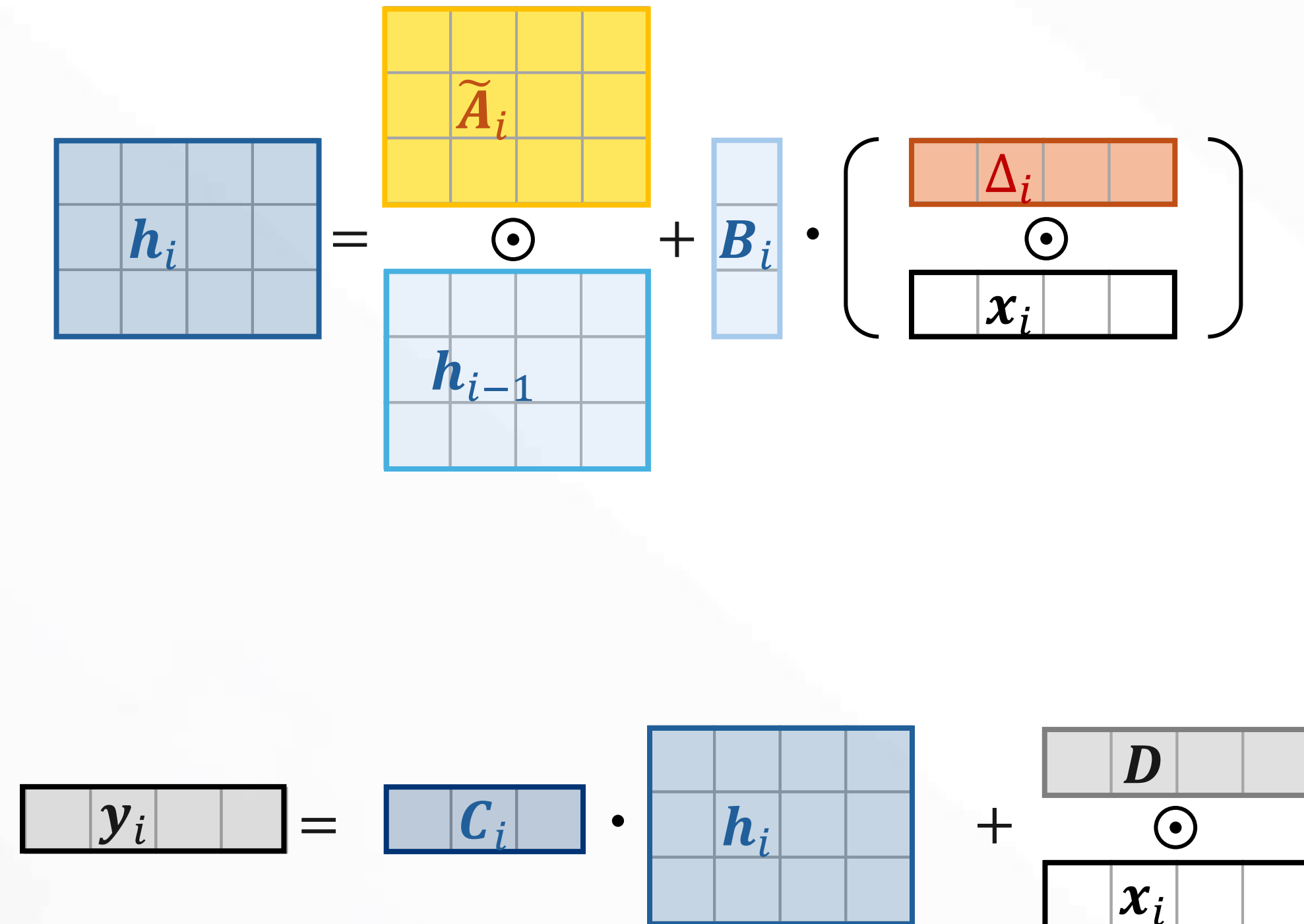


code



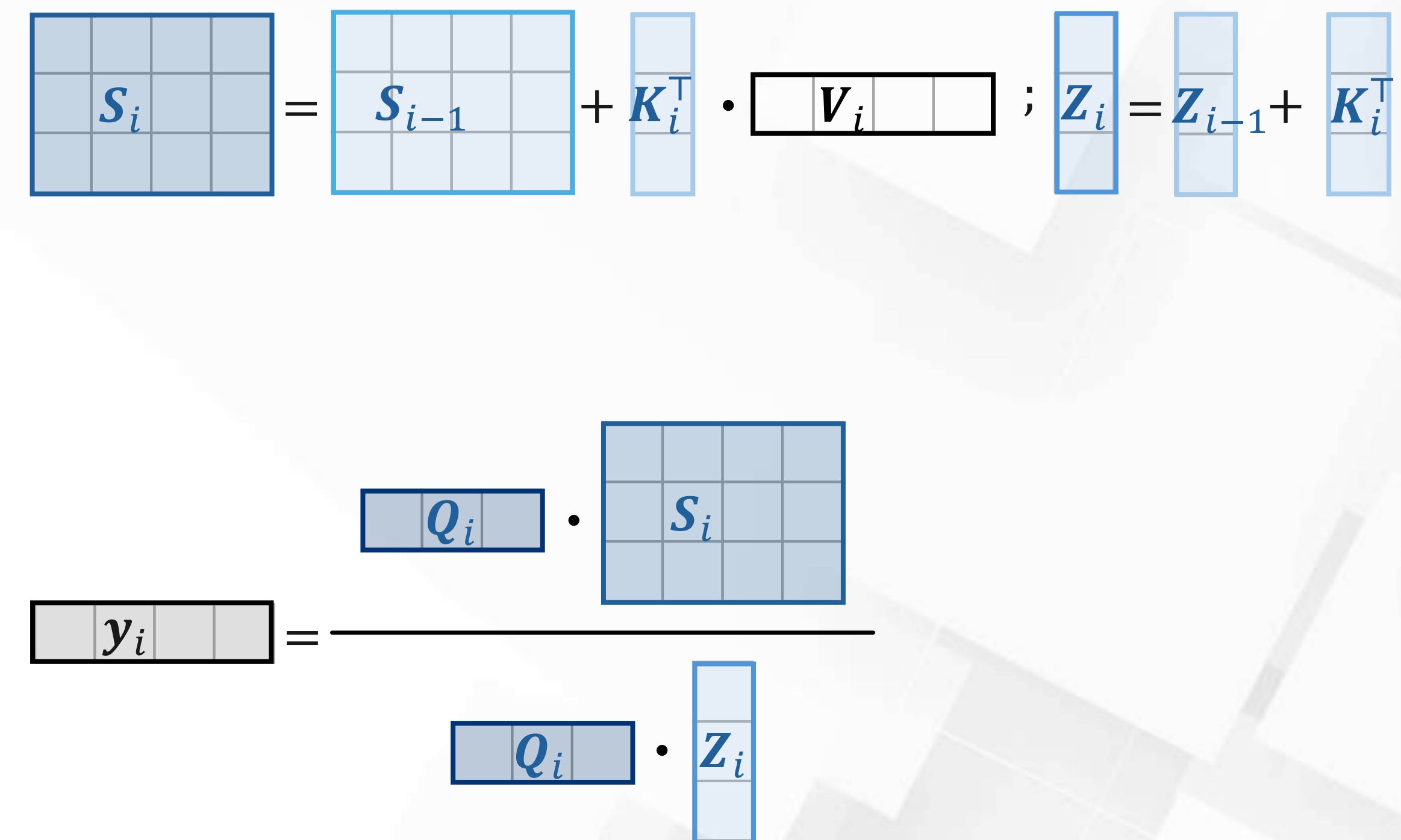
Selective SSM in Mamba

$$h_i = \tilde{A}_i \odot h_{i-1} + B_i(\Delta_i \odot x_i),$$
$$y_i = C_i h_i / 1 + D \odot x_i.$$



Single-head Linear Attention

$$S_i = \mathbf{1} \odot S_{i-1} + K_i^\top (\mathbf{1} \odot V_i),$$
$$y_i = Q_i S_i / Q_i Z_i + \mathbf{0} \odot x_i.$$



Mamba v.s. Linear Attention Transformer



Four differences:

(1) Δ_i : input gate

$$h_i = \tilde{A}_i \odot h_{i-1} + B_i \cdot \left(\begin{array}{c} \Delta_i \\ \odot \\ x_i \end{array} \right)$$
$$y_i = C_i \cdot h_i + \frac{D}{\odot} x_i$$

$$S_i = S_{i-1} + K_i^T \cdot \left(\begin{array}{c} \Delta_i \\ \odot \\ V_i \end{array} \right); Z_i = Z_{i-1} + K_i^T$$
$$y_i = \frac{Q_i \cdot S_i}{Q_i \cdot Z_i}$$

Mamba v.s. Linear Attention Transformer



Four differences:

(1) Δ_i : input gate

(2) \tilde{A}_i : forget gate

$$h_i = \tilde{A}_i \odot h_{i-1} + B_i \cdot \left(\Delta_i \odot x_i \right)$$
$$y_i = C_i \cdot h_i + D \odot x_i$$
The diagram illustrates the Mamba cell equations. The first equation shows the hidden state h_i as a 3x4 grid, which is the element-wise product of a forget gate \tilde{A}_i (a 3x4 yellow grid) and the previous hidden state h_{i-1} (a 3x4 light blue grid). This is then added to the product of a bias matrix B_i (a 3x1 light blue vertical bar) and a vector $\Delta_i \odot x_i$ (a 3x1 black horizontal bar). The second equation shows the output y_i (a 1x4 black horizontal bar) as the product of a bias matrix C_i (a 1x4 light blue horizontal bar) and the hidden state h_i , plus the element-wise product of a bias matrix D (a 1x4 grey horizontal bar) and the input x_i (a 1x4 black horizontal bar).

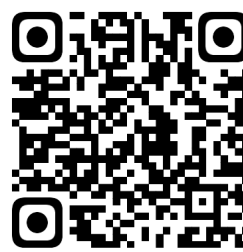
$$s_i = s_{i-1} + K_i^T \cdot V_i$$
$$z_i = z_{i-1} + K_i^T$$
$$y_i = \frac{Q_i \cdot s_i}{Q_i \cdot z_i}$$
The diagram illustrates the Linear Attention Transformer cell equations. The first equation shows the state s_i (a 3x4 light blue grid) as the sum of the previous state s_{i-1} (a 3x4 light blue grid) and the product of a key matrix K_i^T (a 3x1 light blue vertical bar) and a value vector V_i (a 1x4 black horizontal bar). The second equation shows the state z_i (a 3x1 light blue vertical bar) as the sum of the previous state z_{i-1} (a 3x1 light blue vertical bar) and the key matrix K_i^T . The third equation shows the output y_i (a 1x4 black horizontal bar) as the element-wise product of a query vector Q_i (a 1x4 light blue horizontal bar) and the state s_i , divided by the element-wise product of the query vector Q_i and the state z_i .

Mamba v.s. Linear Attention Transformer

paper



code



Four differences:

(1) Δ_i : input gate

(2) \tilde{A}_i : forget gate

(3) $D \odot x_i$: shortcut

The diagram illustrates the Mamba state transition and output calculation. It is divided into two parts by a vertical dashed line. The left part shows the state transition: a blue grid representing the current state h_i is equal to a yellow grid representing the forget gate \tilde{A}_i element-wise multiplied (\odot) with a blue grid representing the previous state h_{i-1} , plus a blue vector B_i multiplied by a vector x_i element-wise multiplied (\odot) by an orange vector Δ_i . The right part shows the output calculation: a grey vector y_i is equal to a blue vector C_i multiplied by the current state h_i , plus a grey vector D element-wise multiplied (\odot) by the input x_i . The label (3) is placed above the D vector.

$$h_i = \tilde{A}_i \odot h_{i-1} + B_i \cdot \left(\Delta_i \odot x_i \right)$$
$$y_i = C_i \cdot h_i + D \odot x_i \quad (3)$$

The diagram illustrates the Linear Attention Transformer state transition and output calculation. It is divided into two parts by a vertical dashed line. The left part shows the state transition: a blue grid representing the current state S_i is equal to a blue grid representing the previous state S_{i-1} plus a blue vector K_i^T multiplied by a vector V_i element-wise multiplied (\odot) by a dashed orange box. The right part shows the output calculation: a grey vector y_i is equal to a blue vector Q_i multiplied by a blue grid S_i , divided by a blue vector Q_i multiplied by a blue vector Z_i , plus a dashed orange box. The label (3) is placed above the dashed orange box. The state transition equation also includes the update rule for Z_i : $Z_i = Z_{i-1} + K_i^T$.

$$S_i = S_{i-1} + K_i^T \cdot \left(V_i \odot \right); Z_i = Z_{i-1} + K_i^T$$
$$y_i = \frac{Q_i \cdot S_i}{Q_i \cdot Z_i} + \quad (3)$$

Mamba v.s. Linear Attention Transformer

paper



code



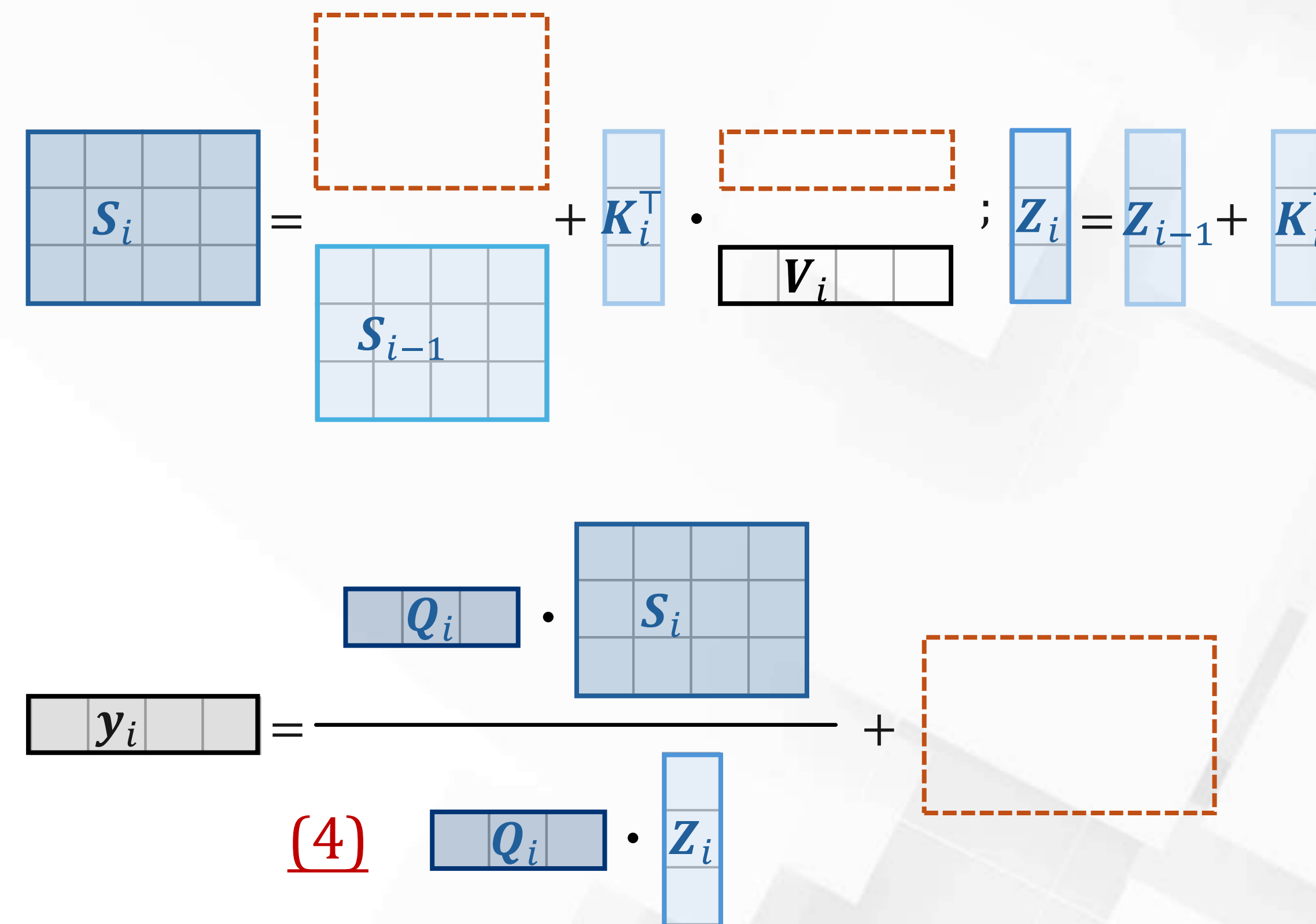
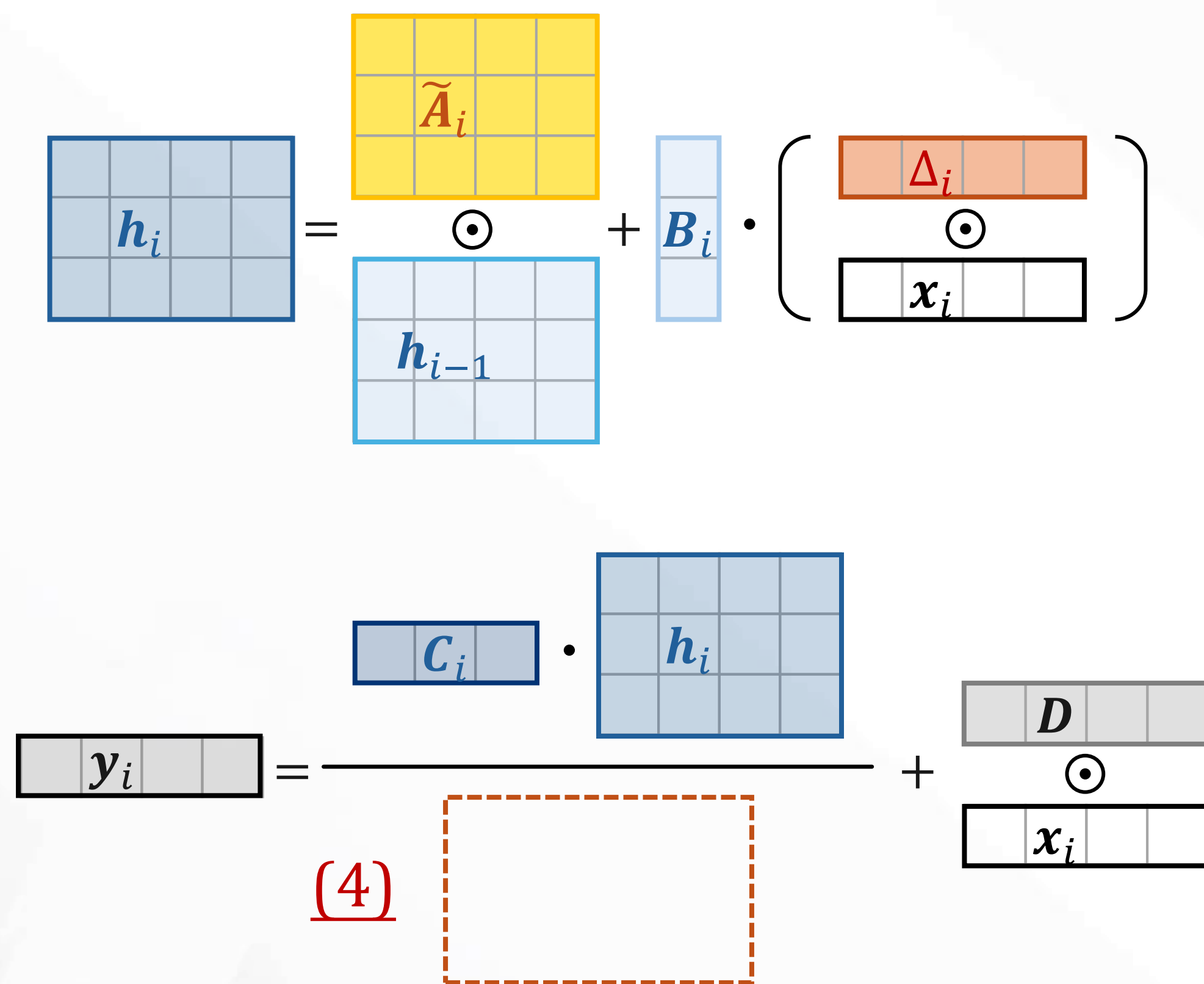
Four differences:

(1) Δ_i : input gate

(2) \tilde{A}_i : forget gate

(3) $D \odot x_i$: shortcut

(4) $Q_i Z_i$: attention normalization



Mamba v.s. Linear Attention Transformer

paper



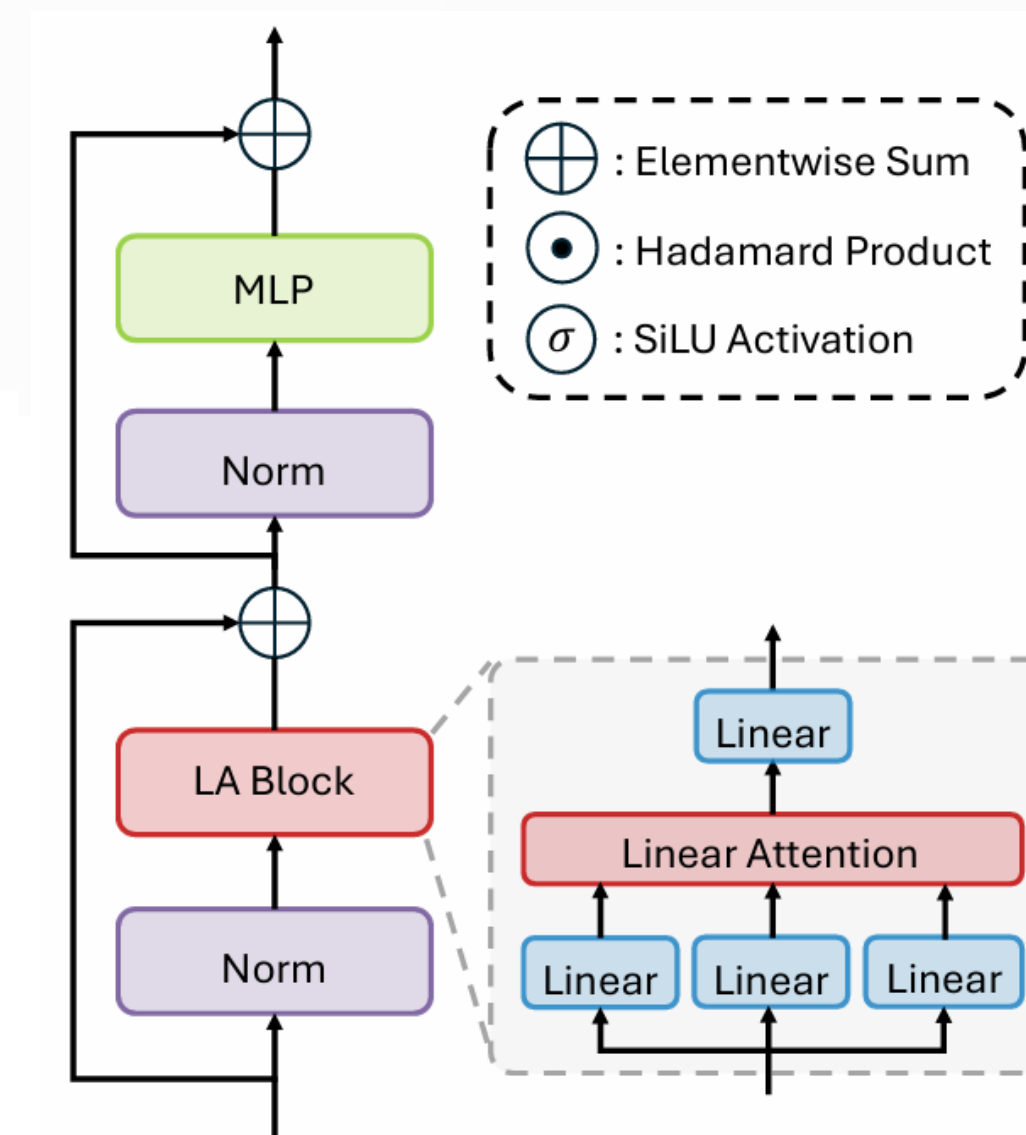
code



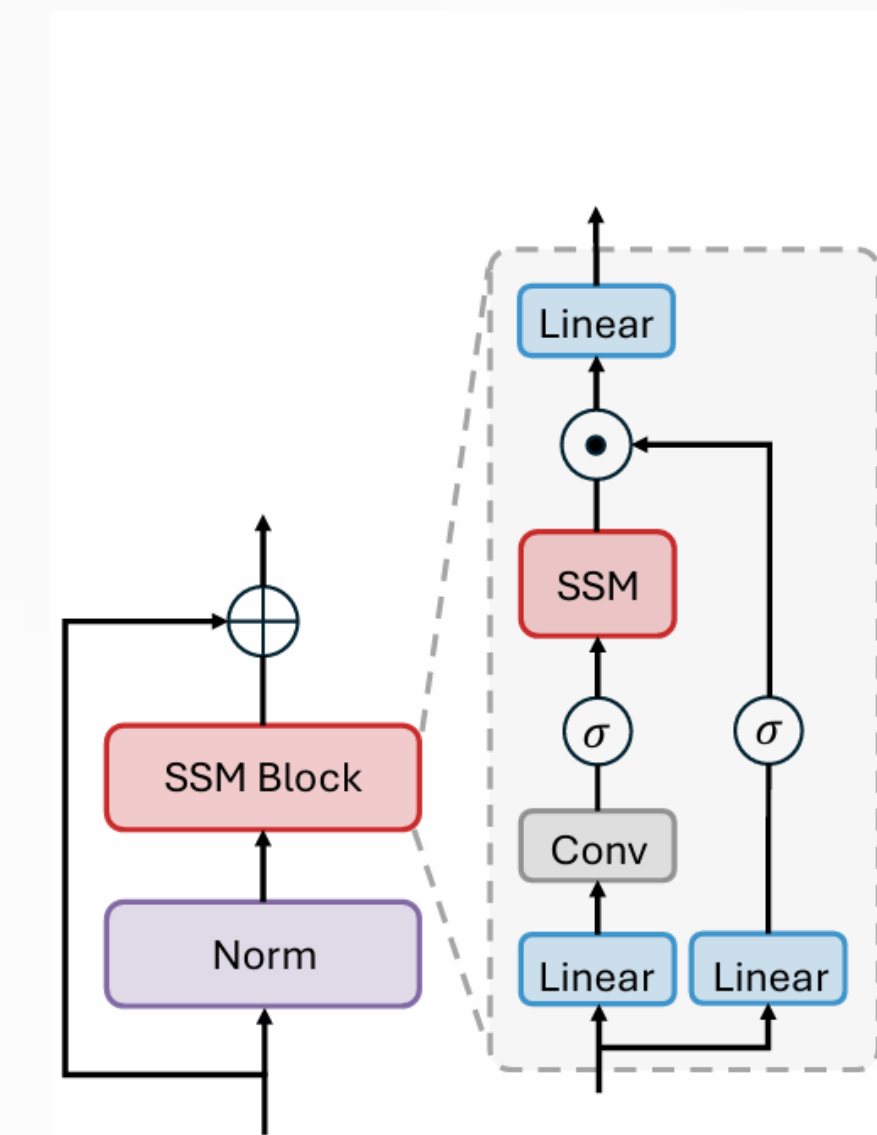
(5) Multi-head design:

- Selective SSM resembles single-head attention
- Linear attention commonly employ multi-head design

(6) Different macro design:



Linear Attention Transformer



Mamba

Mamba v.s. Linear Attention Transformer

paper



code



Mamba can be viewed as
linear attention Transformer
with **six special designs**:

- (1) *input gate*
- (2) *forget gate*
- (3) *shortcut*
- (4) *no attention normalization*
- (5) *single-head design*
- (6) *modified block structure*



Empirical Study

paper



code



	#Params	FLOPs	Throughput	Top-1
Baseline	28M	4.5G	1152	77.6
(1) + Input Gate	29M	4.5G	1069	77.8
(2) + Forget Gate	29M	4.8G	743	78.4
(3) + Shortcut	28M	4.5G	1066	77.8
(4) – Normalization	28M	4.5G	1215	72.4
(5) – Multi-head Design	24M	3.9G	1540	73.5
(6) + Block Design	31M	4.8G	1010	80.9

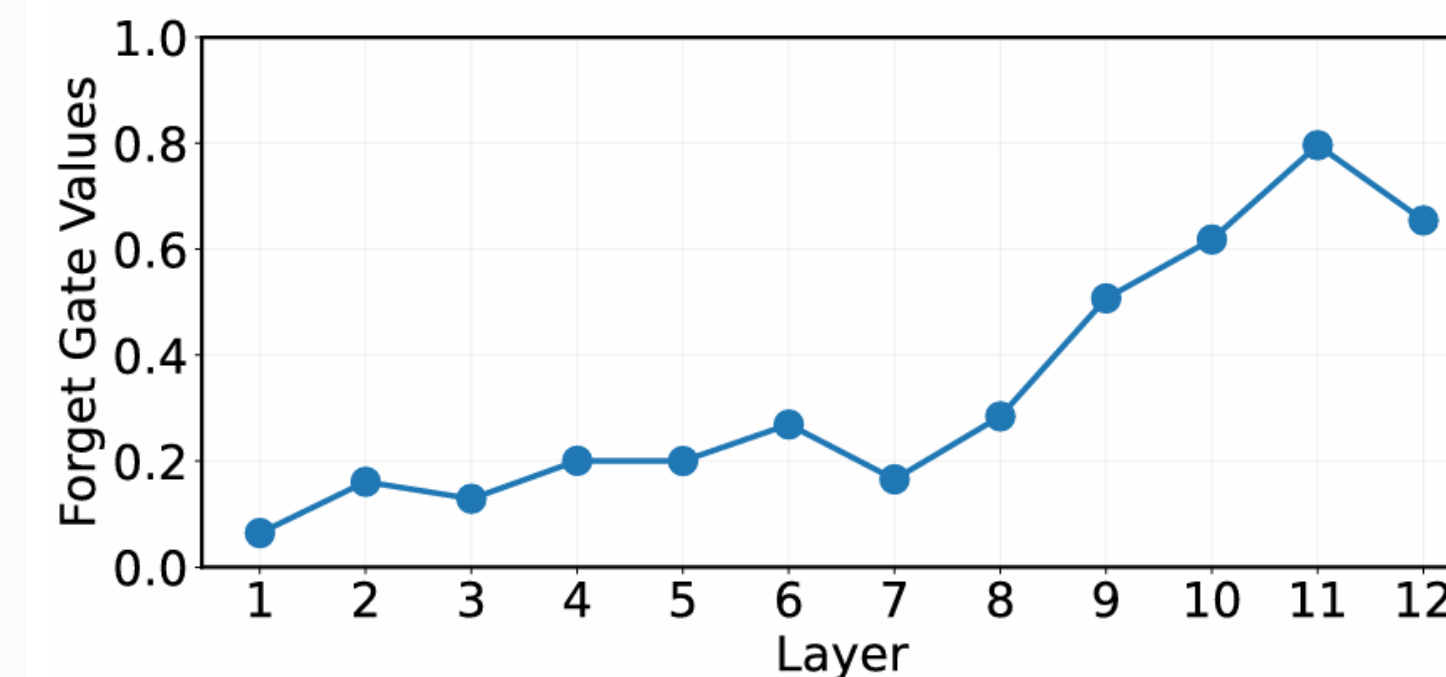
The forget gate and block design tend to be the core contributors!

Empirical Study

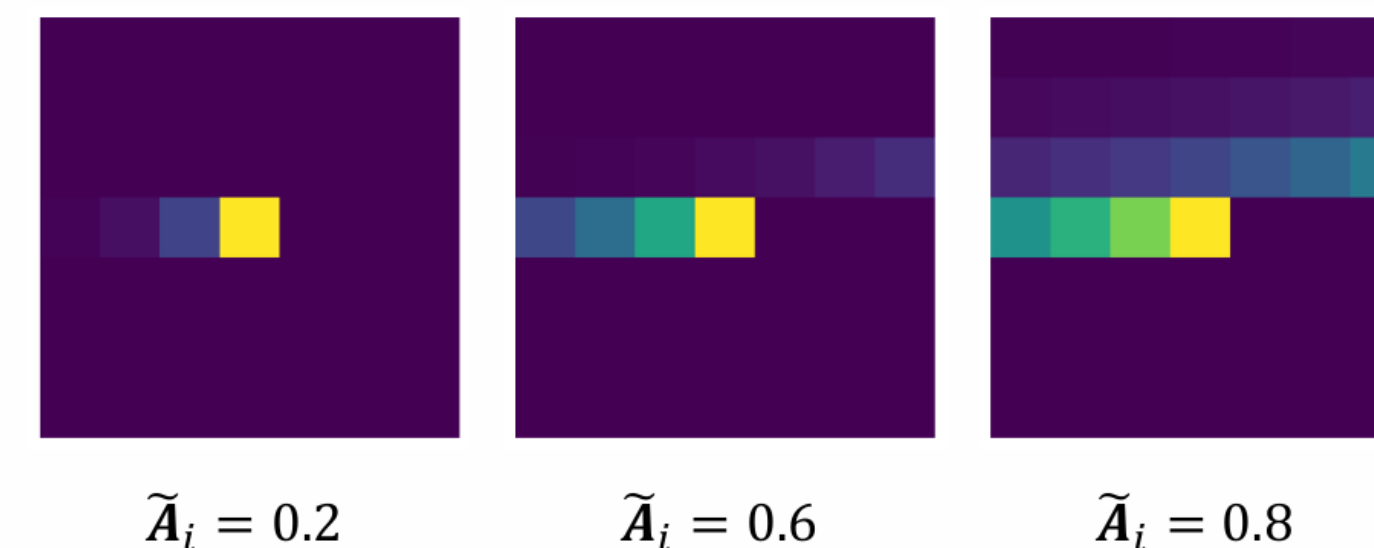


- The **forget gate** needs *recurrent calculation*, which is not ideal for vision models.
- Proper **positional encoding** can function as the forget gate in vision tasks, while preserving *parallelizable computation*.

	#Params	FLOPs	Throughput	Top-1
Baseline	28M	4.5G	1152	77.6
+ Forget Gate	29M	4.8G	743	78.4
+ APE [8]	30M	4.5G	1132	80.0
+ LePE [7]	28M	4.5G	1074	81.6
+ CPE [4]	28M	4.5G	1099	81.7
+ RoPE [33]	28M	4.5G	1113	80.0



(a) Forget Gate Average



(b) Forget Gate Illustration



Based on these findings, we propose a
Mamba-Inspired Linear Attention (MILA) model
by incorporating the merits of Mamba's two key designs
into linear attention.

Empirical Study: ImageNet Classification

paper



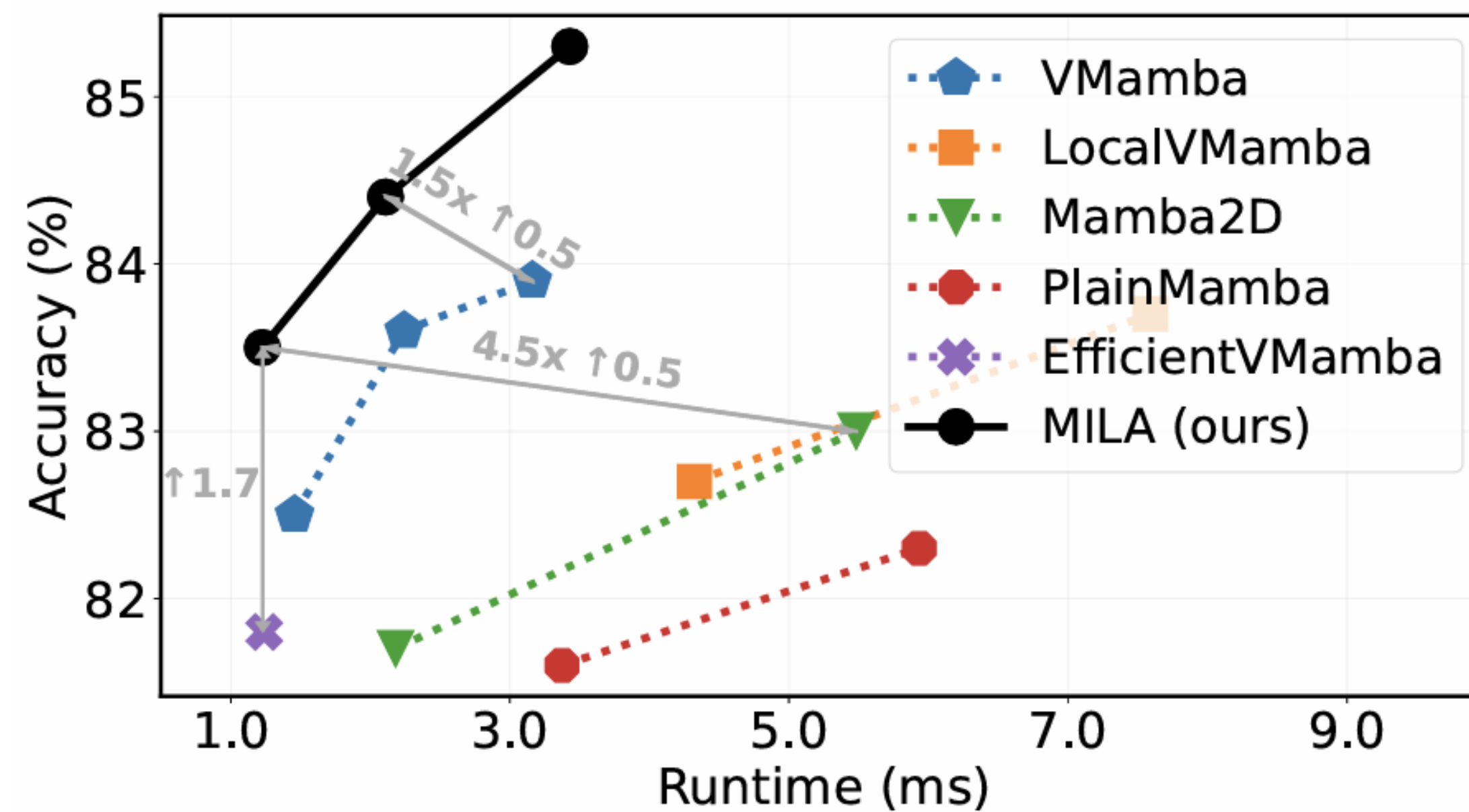
code



Method	Type	#Params	FLOPs	Top-1
ConvNeXt-T [33]	CNN	29M	4.5G	82.1
MambaOut-T [51]	CNN	27M	4.5G	82.7
Swin-T [32]	Transformer	29M	4.5G	81.3
PVTv2-B2 [44]	Transformer	25M	4.0G	82.0
Focal-T [50]	Transformer	29M	4.9G	82.2
MViTv2-T [28]	Transformer	24M	4.7G	82.3
CSwin-T [9]	Transformer	23M	4.3G	82.7
DiNAT-T [19]	Transformer	28M	4.3G	82.7
NAT-T [20]	Transformer	28M	4.3G	83.2
PlainMamba-L1 [49]	Mamba	7M	3.0G	77.9
Vim-S [57]	Mamba	26M	5.1G	80.3
LocalVim-S [25]	Mamba	28M	4.8G	81.2
PlainMamba-L2 [49]	Mamba	25M	8.1G	81.6
Mamba2D-S [27]	Mamba	24M	—	81.7
EfficientVMamba-B [38]	Mamba	33M	4.0G	81.8
VMamba-T [31]	Mamba	31M	4.9G	82.5
LocalVMamba-T [25]	Mamba	26M	5.7G	82.7
MILA-T	MILA	25M	4.2G	83.5

Method	Type	#Params	FLOPs	Top-1
ConvNeXt-S [33]	CNN	50M	8.7G	83.1
MambaOut-S [51]	CNN	48M	9.0G	84.1
PVTv2-B3 [44]	Transformer	45M	7.9G	83.2
CSwin-S [9]	Transformer	35M	6.9G	83.6
Focal-S [50]	Transformer	51M	9.4G	83.6
MViTv2-S [28]	Transformer	35M	7.0G	83.6
VMamba-S [31]	Mamba	50M	8.7G	83.6
LocalVMamba-S [25]	Mamba	50M	11.4G	83.7
MILA-S	MILA	43M	7.3G	84.4
ConvNeXt-B [33]	CNN	89M	15.4G	83.8
MambaOut-B [51]	CNN	85M	15.8G	84.2
PVTv2-B5 [44]	Transformer	82M	11.8G	83.8
Focal-B [50]	Transformer	90M	16.4G	84.0
CSwin-B	Transformer	78M	15.0G	84.2
NAT-B [20]	Transformer	90M	13.7G	84.3
PlainMamba-L3 [49]	Mamba	50M	14.4G	82.3
Mamba2D-B [27]	Mamba	94M	—	83.0
VMamba-B [31]	Mamba	89M	15.4G	83.9
MILA-B	MILA	96M	16.2G	85.3

Empirical Study: Efficiency



Empirical Study: Object Detection

paper



code



(b) Mask R-CNN 3x on COCO

Method	Type	#Params	FLOPs	AP ^b	AP ₅₀ ^b	AP ₇₅ ^b	AP ^m	AP ₅₀ ^m	AP ₇₅ ^m
ConvNeXt-T [33]	CNN	48M	262G	46.2	67.9	50.8	41.7	65.0	44.9
Swin-T [32]	Transformer	48M	267G	46.0	68.1	50.3	41.6	65.1	44.9
PVTv2-B2 [44]	Transformer	45M	309G	47.8	69.7	52.6	43.1	66.8	46.7
FocalNet-T [50]	Transformer	49M	268G	48.0	69.7	53.0	42.9	66.5	46.1
Vmamba-T [31]	Mamba	50M	270G	48.9	70.6	53.6	43.7	67.7	46.8
LocalVMamba-T [25]	Mamba	45M	291G	48.7	70.1	53.0	43.4	67.0	46.4
MILA-T	MILA	44M	255G	48.8	71.0	53.6	43.8	68.0	46.8
ConvNeXt-S [33]	CNN	70M	348G	47.9	70.0	52.7	42.9	66.9	46.2
Swin-S [32]	Transformer	69M	354G	48.2	69.8	52.8	43.2	67.0	46.1
PVTv2-B3 [44]	Transformer	65M	397G	48.4	69.8	53.3	43.2	66.9	46.7
FocalNet-S [50]	Transformer	72M	365G	49.3	70.7	54.2	43.8	67.9	47.4
CSWin-S [9]	Transformer	54M	342G	50.0	71.3	54.7	44.5	68.4	47.7
Vmamba-S [31]	Mamba	70M	384G	49.9	70.9	54.7	44.2	68.2	47.7
LocalVMamba-S [25]	Mamba	69M	414G	49.9	70.5	54.4	44.1	67.8	47.4
MILA-S	MILA	63M	319G	50.5	71.8	55.2	44.9	69.1	48.2

Take-away Messages

paper



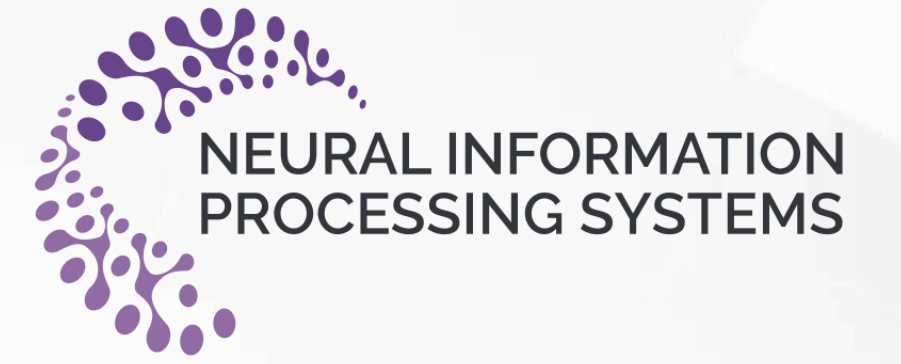
code



- ✓ We reveal the *surprisingly close relationship* between the powerful Mamba and subpar linear attention Transformer
- ✓ We identify that the *forget gate* and *block design* are the core factors behind Mamba's success
- ✓ We propose *Mamba-Inspire Linear Attention (MILA)* model, enjoying *high performance* while maintaining *parallel computation* and *fast inference speed*.



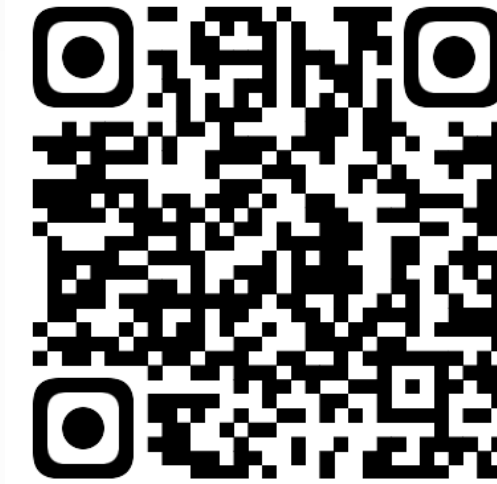
清华大学
Tsinghua University



paper



code



Thank you!

Contact: hdc23@mails.tsinghua.edu.cn