LiSA: Linear Stochastic Attention for Long-Range Visual Modeling

Multi-hop token interactions with linear sequence scaling

Sahar Dastani, Fereshteh Shakeri, Samuel Barbeau, Kunal Mahatha,
Ismail Ben Ayed, Herve Lombaert, Christian Desrosiers

arXiv:2611.04216v2 | 4 November 2026

WACV 2027 author project; official code forthcoming

Can linear attention model indirect token relationships?

The paper's question

Can a vision model add multi-hop context while keeping attention cost linear in the number of patches?

- Softmax attention explicitly compares every pair of tokens: quadratic time and memory.
- Kernelized linear attention avoids the $n \times n$ matrix, but aggregates each value through one direct query–key interaction.
- LiSA treats token similarities as transition probabilities and propagates information along short random walks.

Paper, Sec. 1, PDF pp. 1–2.

Kernelized linear attention is a one-hop update

For positive feature maps $Q, K \in \mathbb{R}^{n \times d_h}$:

$$Y = Q(K^\top V).$$

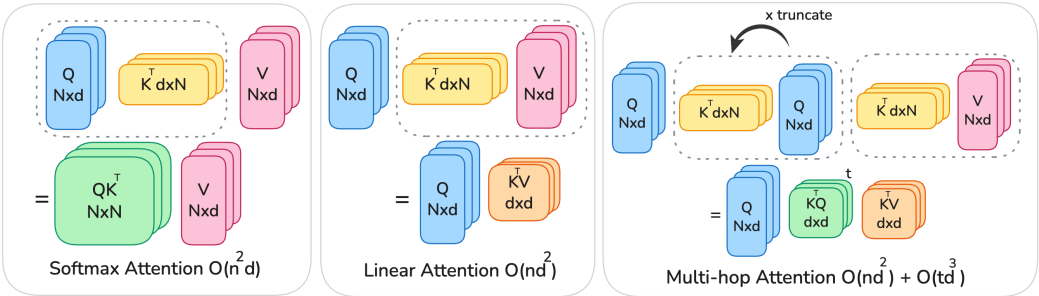
- First aggregate keys and values into a $d_h \times d$ summary.
- Each query reads that summary once: direct, first-order interaction.
- Indirect paths such as A^2V and A^3V, for $A = QK^\top$, are absent.

LiSA's design target

Add useful transitive paths without forming token-by-token attention.

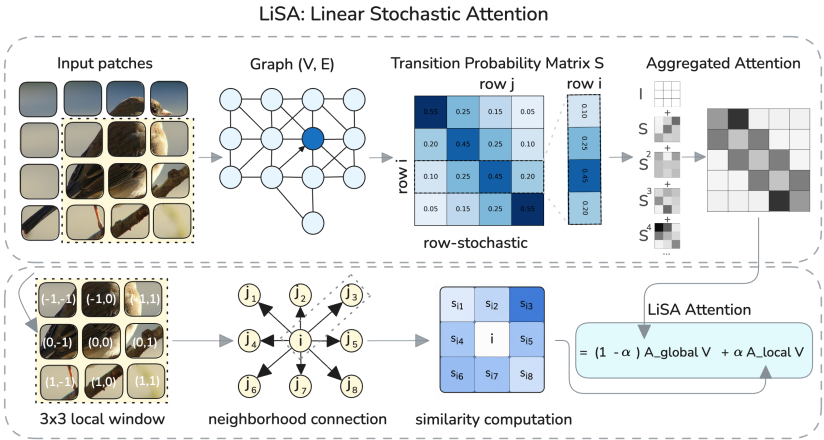
Paper, Secs. 1 and 3.1, PDF pp. 2–3.

LiSA expands the receptive field through low-rank terms



Paper, Fig. 1, PDF p. 1. The right panel contrasts multi-hop terms with the $n \times n$ softmax matrix.

The layer blends global diffusion with local detail



Paper, Fig. 2, PDF p. 4. The schematic shows a global transition branch and a masked 3×3 local branch.

A random walk assigns weight to paths of different lengths

Normalize the positive query–key affinities into a row-stochastic matrix:

$$S = \tilde{Q}K^\top, \qquad \tilde{Q} = D^{-1}Q, \quad D = \mathrm{Diag}\big(Q(K^\top \mathbf{1})\big).$$

Truncate the geometric walk after T terms:

$$P^{(T)} = \frac{1-\beta}{1-\beta^T} \sum_{k=0}^{T-1} \beta^k S^k, \qquad Y_{\mathrm{global}} = P^{(T)} V.$$

- $S^0 = I$ retains the starting token; S^k propagates across k transitions.
- β controls continuation; T is the maximum retained path depth.

Paper, Sec. 3.2, Eqs. (4)–(6), PDF pp. 3–4.

Low rank keeps the recurrence in feature space

Since $S = \tilde{Q}K^\top$, for $k \geq 1$:

$$S^k V = \tilde{Q}(K^\top \tilde{Q})^{k-1} K^\top V.$$

- The repeated core $K^\top \tilde{Q}$ is only $d_h \times d_h$.
- LiSA never needs to materialize the $n \times n$ matrix S.
- The paper reports $O(Td_h^3 + nDd_h)$; with small fixed T, cost is linear in token count n.

Practical choice

Main experiments use $T = 3$; each additional step increases the propagation work.

Paper, Sec. 3.2, Eq. (8) and Algorithm 1, PDF pp. 4–5.

Adaptive hop weights and a local branch protect detail

Global branch

$$Y_g = \beta_0 V + \sum_{k=0}^{T-2} \beta_{k+1} \tilde{Q}(K^\top \tilde{Q})^k K^\top V.$$

- A small MLP predicts nonnegative step weights per token and head.
- Softmax makes the T weights sum to one.

Local branch

- Masked similarity only within a 3×3 spatial neighborhood.
- Includes the token itself as a self-loop.
- Preserves sharp local structure while global diffusion carries long-range context.

Paper, Sec. 3.2, Eq. (10), PDF p. 5; adaptive weights in Supp. Sec. 8, PDF p. 12.

A learned mixture balances global and local evidence

$$Y = (1 - \alpha) Y_{\text{global}} + \alpha Y_{\text{local}}, \qquad \alpha \in [0, 1].$$

- Global-only and local-only variants are both weaker than the learned mixture in the reported ablation.
- Repeated averaging can oversmooth features as T grows.
- The local self-loop and branch retain neighboring detail while diffusion extends reach.

Modeling trade-off

More hops can connect distant regions, but excessive diffusion weakens token distinctions.

Paper, Sec. 3.2 and Table 6, PDF pp. 5, 7; failure cases in Supp. Sec. 9, PDF p. 12.

Evaluation spans classification and dense prediction

- **ImageNet-1K:** DeiT, PVT, Swin, and BiFormer; Top-1 accuracy. Main classification models are trained from scratch for 300 epochs with AdamW and standard augmentation.
- **ADE20K:** Semantic FPN and UPerNet with ImageNet-initialized backbones; mIoU and mAcc.
- **COCO:** Mask R-CNN with PVT-T and Swin-T; $1\times$ schedule; box and mask AP.
- The paper reports eight A100/H100 GPUs for classification; the efficiency table does not state GPU or batch size.

Paper, Sec. 4, PDF pp. 5–7; detailed efficiency table in Supp. Sec. 12, PDF p. 13.

Top-1 accuracy improves across four backbone families

Backbone / input	Self-attention	FLatten	LiSA
DeiT-T, 224^2	72.20	74.10	**74.54**
DeiT-T, 384^2	74.90	76.53	**77.54**
DeiT-S, 224^2	79.80	80.10	**80.78**
PVT-T, 224^2	75.10	77.80	**78.53**
PVT-S, 224^2	79.80	81.70	**81.96**
Swin-T, 224^2	81.30	82.10	**82.25**

- Gains over self-attention range from $+0.95$ to $+3.43$ percentage points in this selection.
- LiSA also exceeds FLatten on each listed row; the margin is smaller on Swin-T ($+0.15$).

Paper, Table 1, PDF p. 6. All rows use the reported main setting $T = 3$.

The largest classification gains appear on PVT and higher resolution

PVT-T at 224^2

LiSA reaches 78.53% Top-1:

- $+3.43$ points over PVT.
- $+0.73$ points over FLatten.
- FLOPs: 2.12G, versus 1.94G for PVT and 2.00G for FLatten.

DeiT-T at 384^2

LiSA reaches 77.54% Top-1:

- $+2.64$ points over DeiT-T.
- $+1.01$ points over FLatten.
- 3.75G FLOPs, below DeiT-T's 4.70G.

BiFormer also improves: $+1.0/+0.3/+0.7$ Top-1 points for T/S/B.

Paper, Tables 1–2, PDF p. 6. Gains are absolute percentage points.

ADE20K gains are strong on PVT-T, mixed against FLatten on Swin

Backbone	Method	FLOPs	mIoU	mAcc
PVT-T / S-FPN	PVT	158G	36.57	46.72
	FLatten	169G	37.21	48.95
	LiSA	170G	**40.85**	**51.75**
Swin-T / UPerNet	Swin	945G	44.51	55.61
	FLatten	946G	44.82	**57.01**
	LiSA	947G	**45.03**	56.40

- PVT-T vs. PVT: $+4.28$ mIoU and $+5.03$ mAcc points.
- Swin-T vs. FLatten: $+0.21$ mIoU, but -0.61 mAcc points.

Paper, Table 3, PDF p. 6. mIoU and mAcc are higher-is-better metrics.

COCO detection gains are larger with the PVT-T backbone

Backbone	FLOPs	Box AP	Mask AP
PVT-T	240G	36.7	35.1
LiSA-PVT-T	245G	**40.5**	**37.6**
Swin-T	267G	43.7	39.8
LiSA-Swin-T	269G	**44.5**	**40.3**

- PVT-T: $+3.8$ box AP and $+2.5$ mask AP for 5G more FLOPs.
- Swin-T: $+0.8$ box AP and $+0.5$ mask AP for 2G more FLOPs.
- Both use Mask R-CNN, a $1\times$ schedule, and 1280×800 input.

Paper, Table 4, PDF p. 7. AP is averaged over COCO IoU thresholds.

Short walks capture most of the ablation gain

Top-1 by truncation depth

T	Mean $\pm$ std. (3 runs)
1	73.34 ± 0.64
2	75.43 ± 0.56
3	75.63 ± 0.39
4	75.16 ± 0.13
5	$\mathbf{75.73 \pm 0.47}$

Learned global/local mix

Model	Global	Local	Mix
DeiT-T	73.7	73.4	**76.3**
DeiT-S	78.3	78.1	**80.7**

Main experiments use $T = 3$ for lower compute; $T = 5$ is only slightly higher in this ablation.

Paper, Fig. 3 and Table 6, PDF p. 7; ablations use mini-ImageNet unless stated otherwise.

The proposed attention core contributes beyond FLatten add-ons

Core	No add-on	+F	+D	+F+D
LA	65.7	65.9	73.3	74.4
GL (LiSA)	**69.2**	**69.3**	**75.2**	**76.3**

- F and D denote FLatten's focusing and depthwise-convolution components.
- GL beats LA with each matched auxiliary configuration.
- LiSA also improves linear-angular DeiT-T from 66.0 to 73.4, and Agent Attention by +2.37 (T) / +2.01 (S).

Paper, Table 7 and Sec. 4.4, PDF p. 8; alternative plug-ins in PDF pp. 7–8.

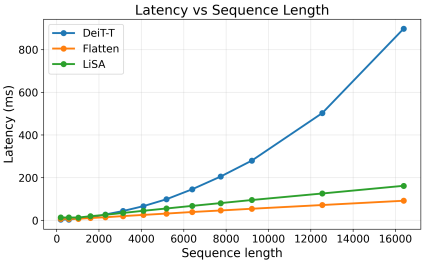
Qualitative maps show the intended transitive behavior



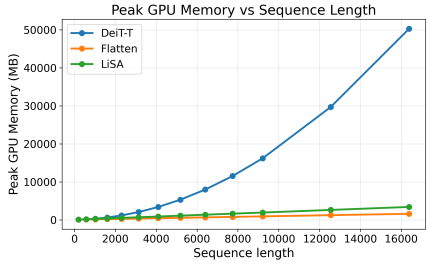
- Local attention stays near the query patch.
- LiSA response spreads to more of the bird as T grows from 3 to 7.

Paper, Fig. 4b, PDF p. 8. This is a qualitative visualization, not an accuracy metric.

At long sequences, LiSA saves memory but trails FLatten



Latency



Peak GPU memory

- At 16,385 tokens: 162 ms / 3.4 GB for LiSA, 92 ms / 1.6 GB for FLatten, and 899 ms / 50.3 GB for DeiT-T.
- At 197 tokens, LiSA latency is already 13.53 ms, versus 7.15 ms for FLatten and 3.50 ms for DeiT-T.

Paper, Fig. 5, PDF p. 8; exact values: Supp. Table 13, p. 13. GPU and batch size are not stated there.

What the evidence supports

- LiSA adds higher-order, transitive interactions to kernelized linear attention through a stochastic diffusion expansion.
- Low-rank algebra avoids an $n \times n$ matrix; with short fixed T, sequence scaling remains linear.
- The strongest reported benefits are on PVT-T, high-resolution DeiT-T, and PVT-T dense prediction.
- There is a real cost trade-off: more latency and memory than FLatten; larger diffusion depth can oversmooth; Swin ADE20K mAcc is lower than FLatten.

Take-home

LiSA is a stronger linear-attention operator when a task benefits from multi-hop context and can absorb the added constant cost.

Evidence: Tables 1–7, PDF pp. 6–8; Supp. Secs. 9–10 and Table 13, PDF pp. 12–13.

Paper status and links

- **Paper:** arXiv:2611.04216v2, revised 4 November 2026.
- **Venue:** Accepted at WACV 2027 based on author project and camera-ready materials; no official proceedings entry was found. See the WACV 2027 author guide.
- **Official code:** author repository; implementation marked “Code Coming Soon”.

Status checked 9 November 2026. The arXiv version remains the source for all method and result claims in this deck.