KLAS: Using Similarity to Stitch Neural Networks for Improved Accuracy-Efficiency Tradeoffs

Debopam Sanyal, Anantharaman Iyer, Alind Khare, Trisha Jain, Akshay Jajoo, Myungjin Lee, Clayton Kerce, Alexey Tumanov

Georgia Tech · SAIL · CISCO

tl;dr: Automate stitch configuration selection in pretrained model zoos using similarity between intermediate activations

- We show existing stitching heuristics are suboptimal and fail to generalize
- KL divergence captures representational and functional similarity, enabling principled stitch selection with no extra finetuning cost

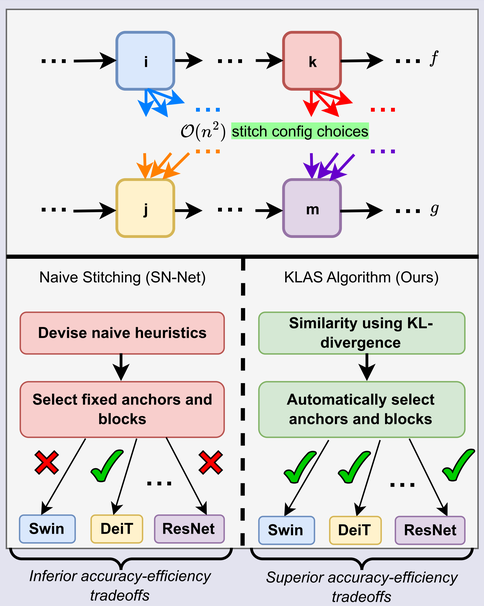
Model Stitching for Many-to-Many NAS

Goal: Given two pretrained anchors f and g with blocks (i.e., layers) $i \in f, j \in g$, learn a stitch layer T to produce a stitched network:

$$g_{>j} \circ T \circ f_{\leq i}$$

Key challenge: For two anchors of depth n, there are $\mathcal{O}(n^2)$stitch configurations → exhaustive search is infeasible

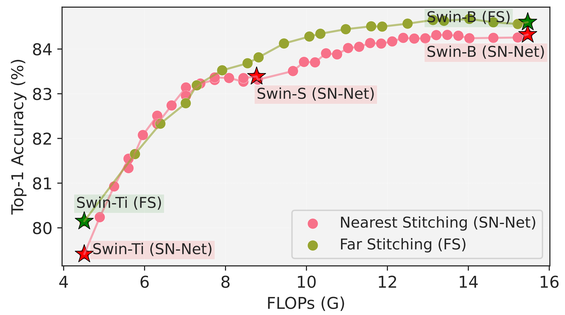
Existing methods: Use heuristics that do not generalize across model families (e.g., SN-Net[1])



(1) Can g produce similar outputs when the inputs are transformed internal representations from f?

(2) Can such transformations be effective with minimal finetuning?

Existing Heuristics / Similarity Metrics are Suboptimal



Direct far-stitching beats SN-Net's nearest stitching by 0.9% accuracy

Metric	Type	Overlap
CKA	✗ (Unsupervised)	5.5%
MSE	✓ (Supervised)	27.8%
CE	✓ (Supervised)	22.2%
DM	✓ (Supervised)	33.3%
SN-Net	✗ (Unsupervised)	61.1%
KL-Div(Ours)	✓ (Supervised)	**88.9%**

Existing similarity metrics fail to recover good stitch configs

The KLAS Framework

Key Intuition: A good similarity metric must capture both similarities

1. *Representational*: can T learn a simple linear map from A_i^f to A_j^g?
2. *Functional:* does the source preserve task-relevant information?

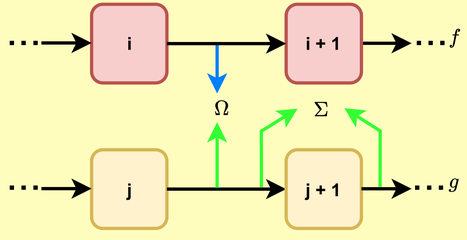
*CKA captures only (1); MSE / CE / DM only (2); **KL divergence** captures both (1) and (2)*

$$\Theta(P_i^f, P_j^g) = \frac{\sum_{x \in \mathcal{D}_v} D_{KL}\left(P_i^f(x) \,||\, P_j^g(x)\right)}{|\mathcal{D}_v|}$$

Linear Probes: Attach a 1X1 conv probe after each block → get $P_i^f(x)$. Jointly trained via **ProbeNet** (one probe active per forward pass).

Step 1 – Anchor Selection: Compute last-block KL divergence between candidate anchors. Smallest values → most compatible anchor pair. No probes needed (softmax already computed).

Step 2 – Block Selection via Stitch Score: Lower $\Gamma(i,j)$ is better



$$\Gamma(i,j) = \frac{\overbrace{\Theta(P_i^f, P_j^g)}^{\text{cross-anchor activation distance } (\Omega)}}{\underbrace{\Theta(P_j^g, P_{j+1}^g)}_{\text{intra-anchor block capacity } (\Sigma)}}$$

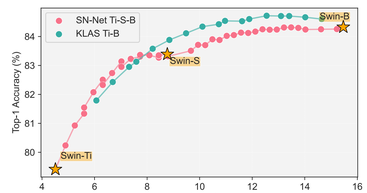
Step 3 – Candidate Selection with Buckets: Pick set $\mathcal{S}$ for finetuning

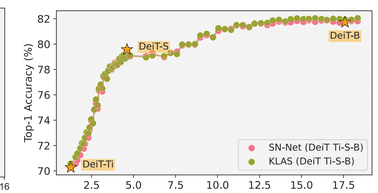
$$\mathcal{S} = \bigcup_{b \in \mathcal{B}} \mathcal{R}_b^*;\ \mathcal{R}_b^* = \left(\left\{ (i^*, j^*) \mid \Gamma(i,j) \leq \tau,\ \forall (i,j) \in b \right\} \bigcup \left\{ (i^*, j^*) = \arg\min_{(i,j) \in b} \Gamma(i,j) \right\} \right)$$

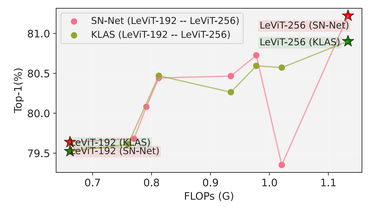
Experimental Results

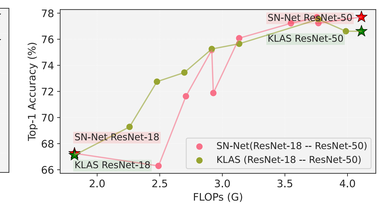
(a) Accuracy-Efficiency Pareto fronts

Metric	CKA	MSE	CE	DM	SN-Net	*KLAS (ours)*
AUC	0.8124	0.7564	0.8023	0.7642	0.8345	**0.8950**









(b) LLMs / Dense Prediction tasks / Ablations

Model	Method	ROUGE-1	ROUGE-2
Stitched LLaMa 1.6 B	ESTA	0.576	0.304
Stitched LLaMa 1.4 B	**KLAS**	**0.593**	**0.337**
Stitched LLaMa 2.7 B	ESTA	0.631	0.353
Stitched LLaMa 2.6 B	**KLAS**	**0.645**	**0.379**

Model	FLOPs(G)	mIoU(%)
SN-Net-1	152	29.4
KLAS-1	145	**29.8**
SN-Net-2	274	32.6
KLAS-2	277	**33.5**
SN-Net-3	327	37.7
KLAS-3	316	**37.8**

Threshold (τ)	Avg Top-1(%)	AUC
1%	83.72	0.8934
3%	83.74	0.8942
5%	**83.76**	**0.8950**
10%	83.69	0.8931

#Buckets	Avg Top-1(%)	AUC
10	83.65	0.8902
15	83.75	0.8947
20	**83.76**	**0.8950**

[1] Pan, Zizheng, Jianfei Cai, and Bohan Zhuang. "Stitchable neural networks." CVPR 2023.