Table 1: Details of teacher models. The head dimensions of all the models are set to 64. 'ks' means kernel size. 'st' means stride. 'oc' means output channel number.

Teacher	Student	Resolution	Depth	Heads	Patch Stem (ks, st, oc)
Our LV-ViT-Ti	SiT-Ti	224^2	14	5	(3×3, 2×2, 40) (3×3, 2×2, 80) (3×3, 2×2, 160) (3×3, 2×2, 320)
Our LV-ViT-S	SiT-XS SiT-S	224^2	16	6	(3×3, 2×2, 48) (3×3, 2×2, 96) (3×3, 2×2, 192) (3×3, 2×2, 384)
Our LV-ViT-M	SiT-M	224^2	20	8	(3×3, 2×2, 64) (3×3, 2×2, 128) (3×3, 2×2, 256) (3×3, 2×2, 512)
Our LV-ViT-L	SiT-L	288^2	24	12	(3×3, 2×2, 96) (3×3, 1×1, 96) (3×3, 1×1, 96) (3×3, 2×2, 192) (3×3, 2×2, 384) (3×3, 2×2, 768)

Table 2: Robustness analysis based on our LV-ViT-S.

Ratio	$\mathcal{L}_{\text{logits}}+\mathcal{L}_{\text{token}}$	$\mathcal{L}_{\text{hard}}$
1	83.3	83.3
0.75	83.2	83.0
0.5	82.6	82.2
0.25	80.9	80.0

A More details about teacher models

Table 1 shows more details about our teacher models. We elaborately design different patch stems for our LV-ViT [1] models.

B More robustness analysis.

We conduct more analysis based on our LV-ViT-S in Table 2. It shows that our self-slimmed learning is also robust to different FLOPs ratios on LV-ViT-S. Moreover, our method still performs better than CNN distillation on larger model.

Table 3: More results on DeiT. "DeiT$_P$" indicates the original DeiT and "DeiT$_C$" refers to the variant with lightweight convolutional patch embedding stacked by four 3×3 convolutions (2×2 stride) and one point-wise convolution.

Model	FLOPs ratio	FLOPs (G)	$\mathcal{L}_{\text{logits}}$ +$\mathcal{L}_{\text{token}}$	$\mathcal{L}_{\text{hard}}$	Throughput (image/s)	ImageNet Top-1(%)
DeiT$_P$-S	0.25	1.1	✗	✗	6413(**3.9**×)	71.6(-8.2)
		1.1	✔	✗	6413(**3.9**×)	75.9(-3.9)
		1.1	✗	✔	6286(**3.8**×)	72.9(-6.9)
		1.1	✔	✔	6286(**3.8**×)	75.3(-4.5)
	0.5	2.3	✗	✗	3308(**2.0**×)	78.6(**−1.3**)
		2.3	✔	✗	3308(**2.0**×)	79.4(**−0.4**)
		2.3	✗	✔	3262(**2.0**×)	78.8(**−1.0**)
		2.3	✔	✔	3262(**2.0**×)	79.8(**+0.0**)
	1	4.6	✗	✗	1637	79.8
DeiT$_C$-S	0.25	1.1	✗	✗	5898(**3.7**×)	76.1(-3.9)
		1.1	✔	✗	5898(**3.7**×)	78.4(-1.6)
		1.1	✗	✔	5830(**3.7**×)	77.5(-2.5)
		1.1	✔	✔	5830(**3.7**×)	78.8(-1.2)
	0.5	2.3	✗	✗	3150(**2.0**×)	79.1(**−0.9**)
		2.3	✔	✗	3150(**2.0**×)	79.9(**−0.1**)
		2.3	✗	✔	3106(**1.9**×)	80.3(**+0.3**)
		2.3	✔	✔	3106(**1.9**×)	80.6(**+0.6**)
	1	4.6	✗	✗	1597	80.0

C More experiments on DeiT

We also verify the effectiveness of our self-slimmed learning on DeiT as illustrated in Table 3. For the FLOPs ratio of 0.5 and 0.25, the stage numbers are {3,4,3,2} and {1,1,1,9} respectively. Specifically, we conduct the experiments on the original DeiT [4] and its variant with lightweight convolutional patch embedding. Both models achieve similar accuracy with the same computational costs. However, we observe the performance of their students is quite different especially at a small FLOPs ratio. DeiT$_P$ suffers severe performance deterioration when 75% computation is reduced, while DeiT$_C$ only drops the accuracy by 2.5%. More importantly, DeiT$_C$ generally obtain higher accuracies than DeiT$_P$ at a relatively higher FLOPs ratio. It demonstrates that the models with convolutional patch embedding are more redundant and friendly to slimming. In addition, we also compare our DKD with the CNN distillation under different settings. The layer-to-layer dense knowledge distillation consistently brings more performance gains than CNN distillation. It is worth mentioning that, self-slimmed learning is also complementary to the extra CNN distillation. Surprisingly, the best student model of DeiT$_C$ even outperforms the teacher by 0.6% top-1 accuracy while running 2× faster under the joint supervision. These results prove the effectiveness and generalization ability of our self-slimmed learning.

Table 4: Comparisons between DynamicViT and our SiT on DeiT.

Model	FLOPs ratio	#FLOPs (G)	DynamicViT Throughput (image/s)	DynamicViT ImageNet Top-1(%)	SiT Throughput (image/s)	SiT ImageNet Top-1(%)
DeiT$_P$-S	0.25	1.1	6254(**3.8**×)	65.6(-14.2)	**6413**(**3.9**×)	**75.9**(**−3.9**)
	0.5	2.3	3248(**2.0**×)	78.4(-1.4)	**3308**(**2.0**×)	**79.4**(**−0.4**)
	1	4.6	1637	79.8	1637	79.8
DeiT$_C$-S	0.25	1.1	5689(**3.6**×)	73.4(-6.6)	**5898**(**3.7**×)	**78.4**(**−1.6**)
	0.5	2.3	3092(**1.9**×)	79.2(-0.8)	**3150**(**2.0**×)	**79.9**(**−0.1**)
	1	4.6	1597	80.0	1597	80.0

As described in Table 4, we further compare our self-slimmed learning with the recent method, *i.e.*, DynamicViT. We observe that our SiT runs slightly faster than DynamicViT with the same FLOPs, which reveals our TSM presents better inference efficiency than the prediction module of DynamicViT. More importantly, thanks to the soft-slimming designs, SiT outperforms DynamicViT by a large margin (5.3%-10.0%) at the FLOPs ratio of 0.25. For the large FLOPs ratio, our SiT still obtains at least 0.7% higher accuracy than DynamicViT, proving the soft slimming triumphs the hard dropping manner.

D More experiments on Swin Transformer

Model	Baseline Throughput	Baseline Top-1	Baseline+SiT Throughput	Baseline+SiT Top-1
Swin-T	1023	81.2	1183 (**+15.6%**)	81.2
Swin-S	652	83.0	855 (**+31.1%**)	83.0

Table 5: SiT for hierarchical networks.

Note that the recent slimming methods [3] only works for vanilla ViTs. Since the hierarchical ViTs generally introduce structured operations like convolution and relative position bias, it's not suitable for arbitrary token dropping. To verify the generality of our SiT, we adapt the typical hierarchical network (i.e., Swin) with SiT, modifying some of the structured operations. Table 5 shows that arming Swin with SiT, we can also improve its throughput without accuracy drop. We will focus on more elegant token slimming method in the future.

E More visualizations

Qualitative token slimming visualization. We present more visualizations of our progressive token slimming in Figure 2.

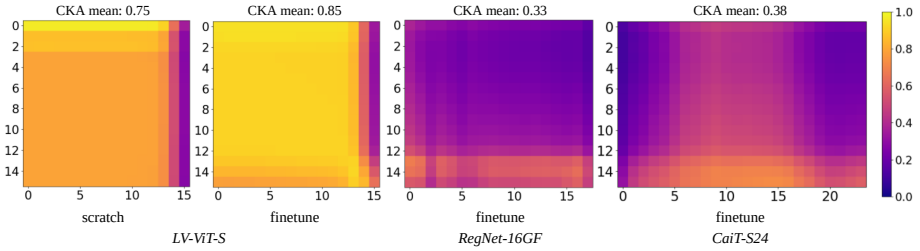

Fig. 1: **Cross CKA heatmap between different student models and the teacher models.** We adopt LV-ViT-S [1] as student. Transfering knowledge densely from same structure yields the largest similarity.

Qualitative FRD visualization. In Fig. 1, we compute the CKA [2] heatmap by comparing all layers of the student models (LV-ViT-S) with all layers of their teacher models. It shows that the CKA similarities between the similar structures are generally higher than those between different structures (0.75/0.85 *vs.* 0.33/0.38). Interestingly, we find the pre-trained weights inherited by the student force itself to be similar to its teacher. Besides, for similar structures, the CKA similarities in the shallow layers are higher than those in deep layers. It is mainly because we slim a large number of tokens after the third layer, leading to an inevitable information loss. As for different structures, the CKA similarities in the deep layers are higher than those in shallow layers, which is mainly because the logits distillation provides direct supervision for features in the deeper layers. Note that the above observations are consistent with the results in our experiments, which reveals that teachers with similar structures can transfer structure knowledge better for higher performance.

References

1. Jiang, Z.H., Hou, Q., Yuan, L., Zhou, D., Shi, Y., Jin, X., Wang, A., Feng, J.: All tokens matter: Token labeling for training better vision transformers. Advances in Neural Information Processing Systems (2021)
2. Kornblith, S., Norouzi, M., Lee, H., Hinton, G.: Similarity of neural network representations revisited. In: International Conference on Machine Learning (2019)
3. Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., Hsieh, C.J.: Dynamicvit: Efficient vision transformers with dynamic token sparsification. Advances in neural information processing systems (2021)
4. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning (2021)

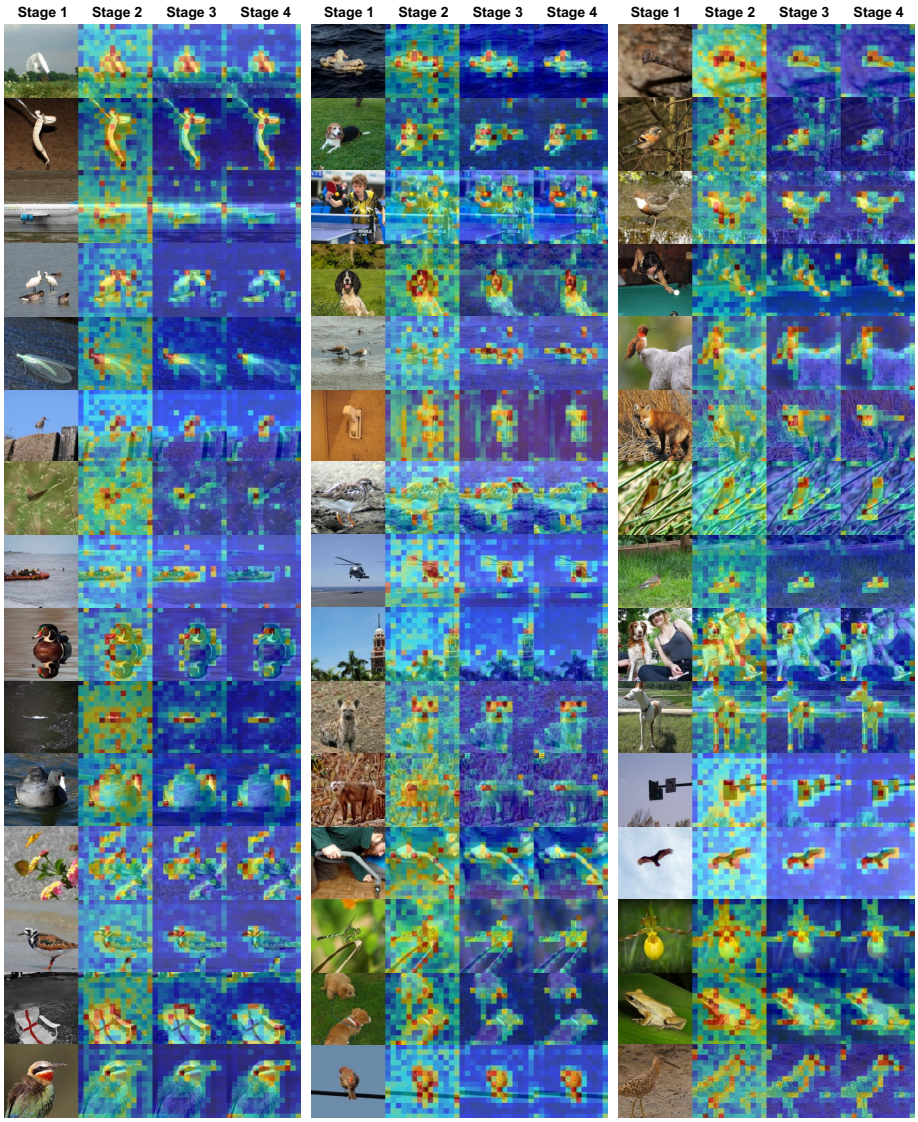

Fig. 2: More visualizations of our SiT.