TinyViT: Fast Pretraining Distillation for Small Vision Transformers
—— Supplementary Material ——

Kan Wu[1,3,*], Jinnian Zhang[2,4,*], Houwen Peng[3,*,†],
Mengchen Liu[4], Bin Xiao[4], Jianlong Fu[3], Lu Yuan[4]

[1] Sun Yat-sen University, [2] University of Wisconsin-Madison,
[3] Microsoft Research, [4] Microsoft Cloud+AI

This supplementary material presents the details of Section 3.2, 5.1, 5.4. Besides, two extra experiments show how fast the proposed distillation method is and the result of distillation with ground-truth.

- **Model Architectures.** We elaborate the model architectures of TinyViT of Section 3.2.
- **Implementation Details.** We provide the details of ImageNet-21k pretraining, ImageNet-1k finetuning and training from scratch of Section 5.1.
- **Few-shot Learning.** We elaborate few-shot datasets and evaluation protocal of Section 5.4.
- **How fast the distillation method is?** We compare the training cost between our proposed fast pretraining distillation and the conventional method, to show the effectiveness of the proposed method.
- **Distillation with ground-truth.** We show why to use soft labels only to distill student models.

A Model Architectures

Our proposed TinyViT architecture is shown in Tab. 1. It is a hierarchical structure with 4 stages, for the convenience of dense prediction downstream tasks like Swin [10] and LeViT [5]. The attention biases [5] and a 3×3 depthwise convolution between attention and MLP are introduced to capture local information [17,1]. The factors $\{\gamma_{D_{1-4}}, \gamma_{N_{1-4}}, \gamma_{W_{2-4}}, \gamma_R, \gamma_M, \gamma_E\}$ can be contracted to form tiny model families. We start with a 21M model and generate a set of candidate models around the basic model by adjusting the contraction factors. Then we select models that satisfy both constraints on the number of parameters and throughput, and evaluate them on 99% train and 1% val data sampled from ImageNet-1k train set. The models with the best validation accuracy will be utilized for further reduction in the next step until the target is achieved. In TinyViT model family, all models share the same factors: $\{\gamma_{N_1}, \gamma_{N_2}, \gamma_{N_3}, \gamma_{N_4}\} = \{2, 2, 6, 2\}$, $\{\gamma_{W_2}, \gamma_{W_3}, \gamma_{W_4}\} = \{7, 14, 7\}$ and $\{\gamma_R, \gamma_M, \gamma_E\} = \{4, 4, 32\}$. For the embeded dimensions $\{\gamma_{D_1}, \gamma_{D_2}, \gamma_{D_3}, \gamma_{D_4}\}$, TinyViT-21M: {96, 192, 384, 576}, TinyViT-11M: {64, 128, 256, 448} and TinyViT-5M: {64, 128, 160, 320}.

*Equal contribution. Work done when Kan and Jinnian were interns of Microsoft.
†Corresponding author: houwen.peng@microsoft.com.

Table 1: An elastic base architecture of TinyViT.

	Block	Configuration	Output
Patch Embed	Stacked Conv	[kernel size 3×3, stride 2, padding 1] $\times 2$	56×56
Stage 1	MBConv [8]	[embed dim γ_{D_1}, expansion ratio γ_R] $\times \gamma_{N_1}$	56×56
Downsampling	MBConv [8]	[embed dim γ_{D_1}, stride 2, hidden/output dim γ_{D_2}] $\times 1$	28×28
Stage 2	Transformer [15]	[embed dim γ_{D_2}, head γ_{D_2}/γ_E, window size $\gamma_{W_2} \times \gamma_{W_2}$, mlp ratio γ_M] $\times \gamma_{N_2}$	28×28
Downsampling	MBConv [8]	[embed dim γ_{D_2}, stride 2, hidden/output dim γ_{D_3}] $\times 1$	14×14
Stage 3	Transformer [15]	[embed dim γ_{D_3}, head γ_{D_3}/γ_E, window size $\gamma_{W_3} \times \gamma_{W_3}$, mlp ratio γ_M] $\times \gamma_{N_3}$	14×14
Downsampling	MBConv [8]	[embed dim γ_{D_3}, stride 2, hidden/output dim γ_{D_4}] $\times 1$	7×7
Stage 4	Transformer [15]	[embed dim γ_{D_4}, head γ_{D_4}/γ_E, window size $\gamma_{W_4} \times \gamma_{W_4}$, mlp ratio γ_M] $\times \gamma_{N_4}$	7×7
Classifier	AvgPool+LayerNorm+Linear	[output dim: the number of classes]	

Besides, we provide some interesting observations about model contraction. It may help both the manual design and the search space design for efficient small vision transformers.

1) For small vision transformers, it improves the accuracy when replacing the transformer block in the first stage with MBConv [8] blocks. We conjecture that early convolution introduces inductive bias like locality [18,5]. It provides more prior knowledge to help small models converge well.

2) It reduces the number of parameters significantly when decreasing the embeded dimension $\gamma_{D_{1-4}}$, so it is the first step to scale the model down. When the model becomes narrower, its depth (especially in the depth of the third stage γ_{N_3}) is increased to satisfy the constraint of the number of parameters.

3) For the MLP expansion ratio γ_M, the value 4 is better than 3 in our models.

4) Window sizes $\gamma_{W_{2-4}}$ do not affect the model size, but larger windows imrpove the accuracy with more computational cost. Especially for Stage 3, 14×14 window size improves the accuracy with little extra computational cost.

B Implementation Details

ImageNet-21k pretraining. We pretrain TinyViT for 90 epochs on ImageNet-21k [4] with an AdamW [11] optimizer, a weight decay of 0.01, initial learning rate of 0.002 with a cosine scheduler, 5 epochs warm-up, batch size of 4,096 and gradient clipping with a max norm of 5. The stochastic depth [9] rate is set to 0 for TinyViT-5/11M and 0.1 for 21M, respectively. The data augmentation techniques include random resize and crop, horizontal flip, color jittering, random erasing [21], RandAugment [3], Mixup [20] and Cutmix [19].

ImageNet-1k finetuning from the pretrained model. We finetune the pretrained models for 30 epochs on ImageNet-1k, using a batch size of 1,024, a cosine learning rate scheduler with 5-epoch warm-up. The initial learning rate is 5×10^{-4} and weight decay is 10^{-8}. The learning rate of each layer is decayed by the rate 0.8 from the output layer to the input layer. The running statistics of BatchNorm are frozen. We disable Mixup and Cutmix.

ImageNet-1k finetuning on higher resolution. When finetuning TinyViT on higher resolution, the windows of each self-attention layer are enlarged as the increasing of input resolution. The attention biases are bilinear-interpolated to adapt the new window size. For example, the window sizes of the four stages are {7, 7, 14, 7} on 224^2 resolution, {12, 12, 24, 12} on 384^2 resolution and {16, 16, 32, 16} on 512^2 resolution. We finetune the model for 30 epochs, using an accumulated batch size of 1024, a cosine learning rate scheduler with 5-epoch warm up. The initial learning rate is 4×10^{-5} and weight decay is 10^{-8}. The running statistic of BatchNorm are frozen. Mixup and Cutmix are disabled.

ImageNet-1k training from scratch. We train our models for 300 epochs on ImageNet-1k with an AdamW optimizer, a weight decay of 0.05, initial learning rate of 0.001 with a cosine scheduler, 20 warm-up epochs, batch size of 1,024 and gradient clipping with a max norm of 5.0. The stochastic depth rate is set to 0.0/0.1/0.2 for TinyViT-5/11M/21M, respectively.

C Few-shot Learning

The few-shot learning benchmark [6] contains four datasets, namely CropDisease [12] (plant leaf images, 38 disease stages over 14 plant species), EuroSAT [7] (RGB satellite images, 10 categories), ISIC 2018 [2] (dermoscopic images of skin lesions, 7 disease states) and ChestX [16] (Chest X-rays, 16 conditions). The learning and inference settings are the same as in [6]. The evaluation protocol involves 5-way classification across 5-, 20- and 50-shot. The classes and shots are randomly sampled for each episode, for 600 episodes per way and shot. Average accuracy over all episodes is reported. We add a single linear layer in replace of the original classification layer in TinyViT-21M.

D How fast the distillation is?

The proposed fast pretraining distillation is faster than the conventional distillation method by 29.8% when using Florence model as the teacher (682M Params and 97.9 GFLOPs). More concretely, our method takes 92.4 GPU days to store the top-100 logits of Florence and 140.0 GPU days to pretrain TinyViT-21M (4.4 GFLOPs) with the saved logits for 90 epochs on ImageNet-21k, while the conventional distillation uses 330.9 GPU days due to limited batch size. Since the teacher logits per epoch are different and independent, they can be saved in parallel, instead of epoch-by-epoch in the conventional method. Besides, the saved logits can be reused for arbitrary student models, and avoid re-forwarding cost of the large teacher model.

Table 2: Comparison for pretraining distillation w/ and w/o ground truth (GT) labels. The student model is a variant of TinyViT-21M, pretrained for 90 epochs on ImageNet-21k and then finetuned for 30 epochs.

IN-21k Pretrained Teacher	Distillation Type	IN-1k Top-1(%)	IN-Real Top-1(%)	IN-V2 Top-1(%)
CLIP-ViT-L/14	w/ GT	84.3	88.5	73.6
	w/o GT	84.5(+0.2)	88.8(+0.3)	74.4(+0.8)
Florence	w/ GT	84.2	88.5	73.7
	w/o GT	84.9(+0.7)	89.0(+0.5)	74.9(+1.2)

E Distillation with ground-truth

We compare the performance under the distillation with and without ground-truth. The student model is a variant of TinyViT-21M, equipped with talking head [14] and shared blocks in Stage 4. As shown in Tab 2, the distillation with ground-truth would cause slight performance drops. This is probably because that not all the labels in ImageNet-21k [4] are mutually exclusive. For example, it contains labels like "chair" and "furniture", "horse" and "animal" [13], which are correlative pairs. Therefore, the one-hot ground-truth label could not describe an object precisely, and in some cases it suppresses either child classes or parent classes during training. By contrast, the soft labels generated by pretrained foundation models carry a lot of category relation information, that is helpful for distilling a small model, as presented in Fig. 3 of the main paper.

References

1. Chu, X., Tian, Z., Zhang, B., Wang, X., Wei, X., Xia, H., Shen, C.: Conditional positional encodings for vision transformers. arXiv (2021)
2. Codella, N., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., et al.: Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). arXiv (2019)
3. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: CVPR (2020)
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
5. Graham, B., El-Nouby, A., Touvron, H., Stock, P., Joulin, A., Jégou, H., Douze, M.: Levit: a vision transformer in convnet's clothing for faster inference. In: ICCV (2021)
6. Guo, Y., Codella, N.C., Karlinsky, L., Codella, J.V., Smith, J.R., Saenko, K., Rosing, T., Feris, R.: A broader study of cross-domain few-shot learning. In: ECCV (2020)
7. Helber, P., Bischke, B., Dengel, A., Borth, D.: Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (2019)

8. Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al.: Searching for mobilenetv3. In: ICCV (2019)
9. Huang, G., Sun, Y., Liu, Z., Sedra, D., Weinberger, K.Q.: Deep networks with stochastic depth. In: ECCV (2016)
10. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021)
11. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2018)
12. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. Frontiers in plant science (2016)
13. Ridnik, T., Ben-Baruch, E., Noy, A., Zelnik-Manor, L.: Imagenet-21k pretraining for the masses. In: NeurIPS (2021)
14. Shazeer, N., Lan, Z., Cheng, Y., Ding, N., Hou, L.: Talking-heads attention. arXiv preprint arXiv:2003.02436 (2020)
15. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
16. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: CVPR (2017)
17. Wu, K., Peng, H., Chen, M., Fu, J., Chao, H.: Rethinking and improving relative position encoding for vision transformer. In: ICCV (2021)
18. Xiao, T., Dollar, P., Singh, M., Mintun, E., Darrell, T., Girshick, R.: Early convolutions help transformers see better. NeurIPS (2021)
19. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: ICCV (2019)
20. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: ICLR (2018)
21. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. In: AAAI (2020)