ResFormer: Scaling ViTs with Multi-Resolution Training (Supplementary Material)

A. More Experiments

A.1. More about Resolution Scalability

Scalability of vanilla ViTs. As displayed in Fig. 1 and Fig. 2, in order to provide more comprehensive insights into resolution scalability, we further test tiny and base models of DeiT [11] which are pre-trained on training resolutions of 196,128, 224, 288 and 384, respectively. The evaluation is conducted by generalizing models to different testing resolutions ranging from 80 to 576. We can observe that the trends towards scaling up and scaling down testing resolutions are consistent with ones on DeiT-S.

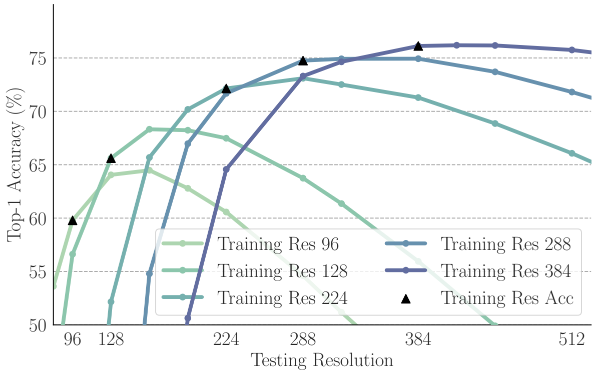

Figure 1. Top-1 accuracy of DeiT-T trained with 5 different resolutions and tested on resolutions varying from 80 to 576.

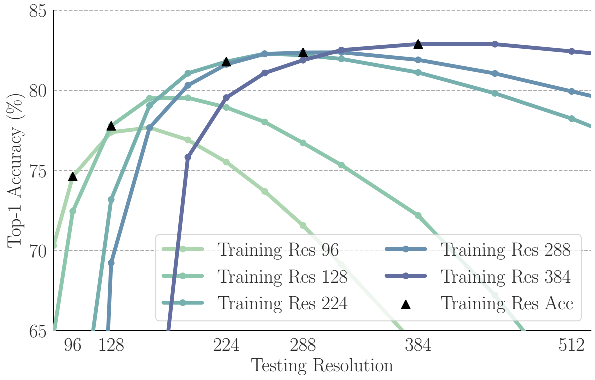

Figure 2. Top-1 accuracy of DeiT-B trained with 5 different resolutions and tested on resolutions varying from 80 to 576.

Table 1. Comparison of Top-1 accuracy between DeiT and ResFormer on ImageNet-1K with high testing resolutions.

Model	Testing resolution			
	512	640	800	1024
DeiT-S-224	72.63	63.86	49.31	31.45
ResFormer-S-MR (224)	82.00	80.72	78.12	72.49
DeiT-S-384	81.09	79.35	75.67	67.61
ResFormer-S-MR (384)	83.86	83.71	83.37	82.58

Extending the range of testing resolutions. To further explore the potential for ResFormer, we extend the range of testing resolutions to 1024. As shown in Tab. 1, compared with DeiT, ResFormer achieves much more decent performance on fairly testing resolutions.

A.2. Evaluation on Robustness Datasets

We also evaluate our models on ImageNet-related robustness datasets, *i.e.*, ImageNet-Rendition (IN-R) [4], ImageNet-A (IN-A) [6], ImageNet-Sketch (IN-SK) [12], ImageNet-C (IN-C) [5] and ImageNetv2 (IN-v2) [10]. As reported in Tab. 2, we observe that ResFormer achieves promising performance on robustness as well. For example, ResFormer-S-224 is superior to DeiT-S on each dataset while ResFormer-S-MR makes further improvements. In particular, on IN-A, ResFormer-S-MR surpasses ResFormer-S-224 by 7.88 % and DeiT-S by 10.07%. This suggests that training with multi-scale inputs facilitates ViTs to cope with hard as well as out-of-distribution inputs.

Table 2. Performance on ImageNet-based robustness benchmarks. mCE [5] is employed for IN-C while Top-1 accuracy is used for IN-R, IN-A and IN-SK.

Model	IN-R↑	IN-A↑	IN-SK↑	IN-C↓	INv2↑
DeiT-S [11]	41.93	19.84	29.09	54.60	68.47
ResFormer-S-224	43.95	22.03	30.91	52.31	69.81
ResFormer-S-MR	45.08	29.91	31.47	51.03	71.68
DeiT-B [11]	44.66	28.15	31.96	48.52	70.91
ResFormer-B-MR	45.38	33.89	33.06	48.83	71.88

A.3. Training Efficiency

During the training of ResFormer, each input sample is replicated by r times, and thus this increases the training time. For efficiency, we reduce the total number of training epochs to 200, 150 and 100 respectively while keeping other hyperparameters unchanged. As shown in Fig. 3, ResFormer demonstrates competitive performance on training efficiency. For instance, ResFormer-S-MR with 200-epoch training surpasses the 300-epoch counterparts ResFormer-S-224 and DeiT-S by 0.83% and 1.83% in Top-1 accuracy despite that they share similar training time.

As depicted in Fig. 4, training with a single lower resolution (*i.e.*, 160) significantly saves time. Nevertheless, ResFormer-S-MR still has an edge on time-performance trade-off, *e.g.*, ResFormer-S-160 with 450-epoch training is more time-consuming than ResFormer-S-MR with 200-epoch training while the accuracy is 0.61% lower.

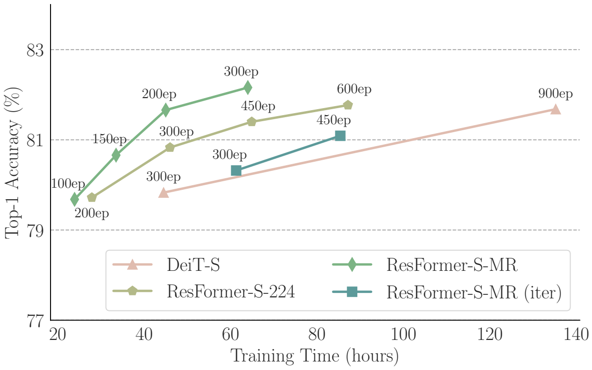

Figure 3. Trade-off between training time and Top-1 Accuracy on ImageNet-1K with a testing resolution of 224. Same hardware and software settings are adopted for all experiments, *i.e.*, we utilize 8× V100-32GB GPUs and set the per-GPU batch size to 128.

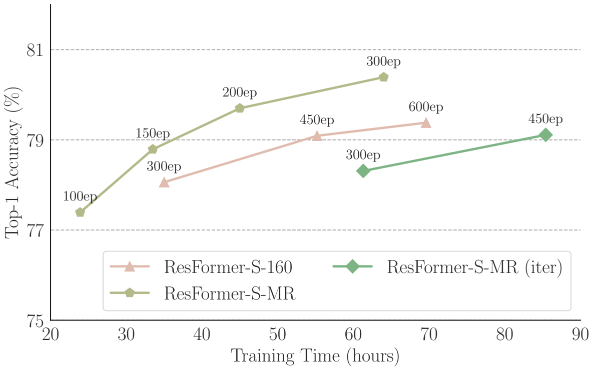

Figure 4. Trade-off between training time and Top-1 Accuracy on ImageNet-1K with a testing resolution of 160. Same hardware and software settings are adopted for all experiments.

B. Implementation Details

B.1. Image Classification

Sine-Cosine positional embedding. We demonstrate the explicit mapping function $\mathcal{F}_{sine}$ for sine-cosine positional embedding p as follows. Firstly, image tokens are placed in a 2D spatial dimension as $x^{img} \in \mathbb{R}^{N_H \times N_W \times D}$. We denote the positional embedding for the token coordinated at (m, n) as $p_{m,n} \in \mathbb{R}^{1\times D}$. Particularly, d-th dimension of $p_{m,n}$ can be mapped with $\mathcal{F}_{sine}(m, n, d)$ as below,

$$\mathcal{F}_{sin}(m,n,d) = \begin{cases} f_{sin}(m,d,N_H,D) & \text{if } d < D/2 \\ f_{sin}(n,d,N_W,D) & \text{otherwise} \end{cases},$$

$$f_{sin}(pos,d,N,D) = \begin{cases} \sin(\frac{pos}{N+\epsilon}/T^{2d/D}) & \text{if } d\%2 = 0 \\ \cos(\frac{pos}{N+\epsilon}/T^{2(d-1)/D}) & \text{otherwise} \end{cases},$$

where the temperature T and ϵ is set to 10000 and $1e^{-6}$ respectively, and a normalization is also used to ensure better continuity among varying resolutions. For simplicity, N_i^H, N_i^W, D are omitted from function parameters.

Detailed hyperparameters. For experiments of image classification on ImageNet-1K, we set the hyperparameters for training ResFormer-T, ResFormer-S, ResFormer-B froom scratch and fine-tuning on DeiT according to Tab. 4.

Augmentation strategy. Motivated by unsupervised learning, we apply separate random augmentation on different scales of inputs. In particular, to ensure the consistency of class tokens between different scales, as an exception, we apply MixUp [15] and CutMix [14] across different scales with same variables. As shown in Tab. 3, separate augmentation slightly outperform its counterpart, especially on the lowest testing resolution.

Table 3. Ablation study of augmentation strategies.

Model	Sep	Testing resolution		
	Aug	128	160	224
ResFormer-S-MR		77.50	80.14	81.93
ResFormer-S-MR	✓	78.24	80.39	82.16

B.2. Semantic Segmentation

We follow the common practice on ADE20K [16] by training on 512 × 512 inputs for 80k iterations for ResFormer-S and for 160k iterations for ResFormer-B, respectively. In addition, we employ the AdamW optimizer with a learning rate of $6e^{-5}$, a weight decay of 0.01 and a batch-size of 16. We base our implementation on MMSegmentation [9] and adopt the corresponding augmentations, *i.e.*, random resizing with the ratio range set to (0.5, 2.0), random horizontal flipping with probability of 0.5 and random photometric distortion. Despite that ResFormer employs a columnar structure, we simply extract features from different layers (*i.e.* the 2nd, 5th, 8th and 11th layers) as inputs of UperNet [13] without FPN-like necks. We report results in two different testing settings. For the first one, inputs are scaled to having a shorter side of 512. In addition, we apply flipping on inputs of multiple scales that are varied in (0.5, 0.75, 1.0, 1.25, 1.5 1.75) × of training resolutions.

B.3. Object Detection

To further validate the efficacy of ResFormer on dense prediction tasks, we evaluate ResFormer on COCO 2017 [8] for object detection and instance segmentation. In particular, we adopt Mask R-CNN [3] as our framework based on MMDetection [2] and train with the 3× schedule. Furthermore, we utilize AdamW optimizer with a learning rate of $1e^{-4}$, weight decay of 0.05 and a batch size of 16. It is worth noting that we follow the common multi-scale training for object detection instead of fine-tuning with the multi-resolution strategy. Therefore, training samples are resized randomly so that the shorter sizes vary from 480 to 800 with step of 32 and the longer sides are within 1333.

B.4. Video Action Recognition

Similar to the implementation for images, we train ResFormer on videos by replicating video clips to get multi-scale copies. Specifically, given a certain sampling rate s of $1/32$, a clip X of $F = 8$ frames is sampled and replicated into r copies. Different cropping sizes are applied on each sequence of frames. Consequently the i-th training copy X_i is sized in $\mathbb{R}^{F \times H_i \times W_i}, i \in \{1, \cdots, r\}$. We also keep the augmentation strategy used for images by applying separate random augmentations [1] on each clip.

We follow the divided attention design adopted in TimeSFormer [1], in which attention computation is conducted along spatial dimension and temporal dimension separately. In order to align with image models, we only incorporate global and local positional embeddings into into spatial dimensions. For training on Kinetics-400 [7], we adopt the same strategy with TimeSFormer [1]. In particular, the training epoch is set to 15 and the initial learning rate is set to $5e^{-3}$. In addition, we employ a SGD optimizer and a multi-step scheduler which divides the learning rate by 10 times at the 11th and the 14th epoch respectively.

In particular, we observe that ResFormer achieves better performance on videos with L_2 scale consistency loss. In order to improve performance by ensuring coherence in pre-training and fine-tuning. We adapt ResFormer-B-MR for L_2 loss for an extended fine-tuning of 100 epochs which matches the common 300-epoch pre-training. For fair comparison, we initiate all ResFormers in Kinetics-400 downstream tasks with same pre-trained weights.

Table 4. Hyperparameters for training on ImageNet-1K.

Hyperparameters	**Tiny / Small**	**Base**	**Fine-tune**
Epochs	300	200	30
Base learning rate	5e-4	8e-4	5e-5
Warmup epochs	5	20	5
Stoch. depth	0.1	0.2	0.1
Gradient clipping	✗	5.0	✗
Batch size	1024		
Weight decay	0.05		
Optimizer	AdamW		
Learning rate schedule	Cosine		
Repeated augmentation	✓		
Random erasing	0.25		
Random augmentation	9/0.5		
Mixup	0.8		
Cutmix	1.0		
Color jitter	0.4		

References

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 3

[2] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 3

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 3

[4] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021. 1

[5] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ICLR*, 2019. 1

[6] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *CVPR*, 2021. 1

[7] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 3

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 3

[9] MMSegmentation Contributors. OpenMMLab Semantic Segmentation Toolbox and Benchmark, 2020. 2

[10] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019. 1

[11] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 1

[12] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *NeurIPS*, 2019. 1

[13] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018. 3

[14] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019. 2

[15] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 2

[16] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *IJCV*, 2019. 2