Meta-attention for ViT-backed Continual Learning

Mengqi Xue[1], Haofei Zhang[1], Jie Song[1,†], Mingli Song[1,2]
[1]Zhejiang University
[2]Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, Zhejiang University
{mqxue,haofeizhang,sjie,brooksong}@zju.edu.cn

Abstract

Continual learning is a longstanding research topic due to its crucial role in tackling continually arriving tasks. Up to now, the study of continual learning in computer vision is mainly restricted to convolutional neural networks (CNNs). However, recently there is a tendency that the newly emerging vision transformers (ViTs) are gradually dominating the field of computer vision, which leaves CNN-based continual learning lagging behind as they can suffer from severe performance degradation if straightforwardly applied to ViTs. In this paper, we study ViT-backed continual learning to strive for higher performance riding on recent advances of ViTs. Inspired by mask-based continual learning methods in CNNs, where a mask is learned per task to adapt the pre-trained ViT to the new task, we propose ***MEta-ATtention*** *(MEAT), i.e., attention to self-attention, to adapt a pre-trained ViT to new tasks without sacrificing performance on already learned tasks. Unlike prior mask-based methods like Piggyback, where all parameters are associated with corresponding masks, MEAT leverages the characteristics of ViTs and only masks a portion of its parameters. It renders MEAT more efficient and effective with less overhead and higher accuracy. Extensive experiments demonstrate that MEAT exhibits significant superiority to its state-of-the-art CNN counterparts, with* $4.0 \sim 6.0\%$ *absolute boosts in accuracy. Our code has been released at* https://github.com/zju-vipa/MEAT-TIL.

1. Introduction

Being capable of tackling everchanging tasks is a favorable merit in open-world scenarios. Humans excel at solving constantly emerging tasks by associating them with previously learned knowledge. Deep neural networks (DNNs), however, usually suffer from *catastrophic forgetting* [33] if simply adapted to new tasks due to the differences between tasks in data biases.

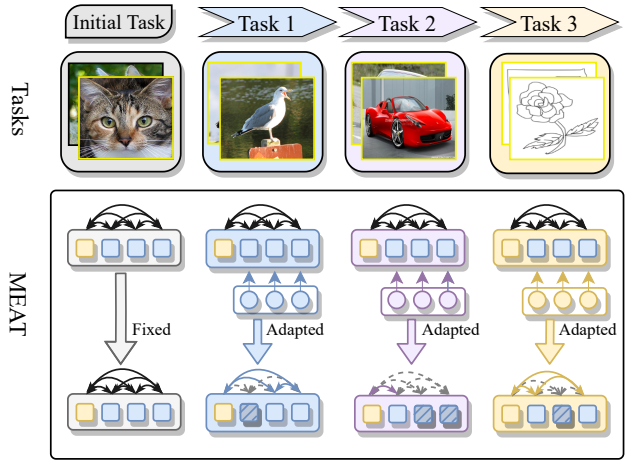

Figure 1. The proposed MEAT for task continual learning in the MHSA block with vision transformers. With the increase of new tasks, MEAT dynamically assigns attention masks to generate task-specific self-attention patterns per task.

Over the past years, large pieces of literature have been devoted to addressing the catastrophic forgetting problem to enable DNNs to master new-arrived tasks in a sequence [23, 27, 30, 36, 38, 41, 43]. Existing continual learning methods can be broadly categorized into three schools: *replay methods* [3, 6, 19, 36, 37], *regularization methods* [21, 23, 26, 27, 34, 50, 52] and *mask methods* [18, 30–32, 41]. Replay methods replay previous task samples, which are stored in raw format or generated with a generative model, to alleviate forgetting while learning a new task. To avoid storing raw inputs, prioritize privacy and alleviate memory requirements, regularization methods introduce a regularization term to consolidate previous knowledge while learning the new task. Mask methods learn a mask per task to adapt the pre-trained model to the new task for preventing any possible forgetting.

Albeit remarkable progress made in computer vision, most of the aforementioned methods are tailored for CNNs for their dominant performance in the field over the past decade. However, the primacy of CNNs in computer vision is

†Corresponding author

recently challenged by vision transformers (ViTs) [10,28,45], due to the more general-purpose architecture (*i.e.*, bridging the architecture gap between natural language processing and computer vision) and superior performance of ViTs. In contrast to the rapid development of ViTs, prior CNN-based continual learning methods appear a bit outdated as straightforwardly applying them to ViTs does not take full advantage of the characteristics of transformers.

In this work, we devote ourselves to ViT-backed continual learning to keep pace with the advancement of ViTs. Specifically, we ground our proposed method on mask method [30–32, 41] for the following three main reasons: (1) mask methods in fact dedicate different parameters to each task, thus perfectly bypassing the catastrophic forgetting problem; (2) mask methods are not sensitive to task order, which is a very favorable merit in continual learning; (3) mask methods avoid expensive data storage and exhibit a larger capacity to handle more tasks, which gives them a prominent edge over replay and regularization methods. Motivated by these attractive advantages, we propose our ViT-backed mask-based continual learning method, dubbed as *MEta-ATtention* (MEAT), to further boost the continual learning performance, as illustrated in Figure 1. MEAT inherits all the aforementioned merits, and meanwhile introduces the following innovations that distinguish it from prior mask methods: (1) MEAT fully leverages the architectural characteristics of ViTs and introduces the *attention to self-attention* (where the meta-attention comes) mechanism, which is tailored for transformer-based architectures and makes MEAT furthermore effective. (2) Prior methods, like Piggyback [30], require manually setting the threshold hyper-parameter for binarizing the mask. MEAT adopts Gumbel-softmax trick [20] to resolve the optimization difficulty of discrete mask values, which relaxes the burden of the hyper-parameter search. (3) Unlike prior mask-based methods where all parameters are assigned to masks, MEAT introduces masks to only a portion of its parameters, which renders it more efficient than prior methods.

To validate the superiority of the proposed method, extensive experiments, including benchmark comparison and ablation study, are conducted on a diverse set of image classification benchmarks (including ImageNet [8], CUB [47], Stanford Cars [24], FGVC-Aircraft [29], CIFAR-100 [25], Sketches [11], WikiArt [40] and Places365 [53]) with various ViT variants (including DeiT-Ti [45], DeiT-S [45] and T2T-ViT-12 [49]). Experimental results demonstrate that MEAT exhibits significant superiority to its state-of-the-art CNN counterparts with $4.0 \sim 6.0\%$ absolute boosts in accuracy, meanwhile consuming much lower storage cost for saving task-specific masks.

In conclusion, the main contributions of our work are summarized as follows:

- We propose MEAT, the first ViT-backed continual learning method to the best of our knowledge, to advance the development of continual learning with ViTs.
- We introduce three innovations into MEAT, including masking partial parameters, avoiding manual hyperparameter setting, and meta-attention mechanism to boost the performance of MEAT.
- Extensive experiments demonstrate that MEAT exhibits significant superiority over its state-of-the-art CNN counterparts, meanwhile consuming much lower storage costs for saving task masks.

2. Related Works

2.1. Vision Transformers

The transformer [46], a prevailing network architecture in nature language processing (NLP) [1, 9, 35], has received growing interest and achieved great accomplishment in the computer vision field, enjoying state-of-the-art performance on many visual tasks, including image classification [10, 28, 45], object detection [2, 5], and object segmentation [4, 51]. Among these works, Vision Transformer [10], the pioneering work in this area, first introduces a complete transformer-based architecture into image classification tasks by splitting an image into 16×16 patches and embedding them into a sequence of tokens as the model input like words in NLP. Inspired by the excellent results achieved by Vision Transformer, many researchers have started to study and improve transformer-based models in computer vision. DeiT [45] improves the training efficiency of Vision Transformer by introducing a new distillation token and some training strategies. Swin Transformer [28] presents a new transformer backbone that constructs a hierarchical representation. Tokens-To-Token Vision Transformer (T2T-ViT) [49] adopts a tokens-to-token (T2T) process to achieve great results trained from scratch on ImageNet [8]. In our experiments we employ three representative vision transformers: DeiT-Ti, DeiT-S and T2T-ViT-12 as backbone networks.

2.2. Continual Learning

Continual learning involves incrementally training a model with a new stream of tasks while preserving its previous knowledge basis, which has attracted much interest in recent years [7]. Generally, there are two kinds of settings for continual learning: (1) *task continual learning* that extends knowledge with new tasks which have clear domain boundaries; (2) *class continual learning* that accumulates knowledge over different sets of categories separated from the same dataset. In this work, we mainly focus on task continual learning. Previous continual learning methods can be broadly categorized into replay methods, regularization methods and mask methods. The replay

methods [3, 6, 19, 36, 37] expect to store a subset of samples of previous tasks and retrain the model on old samples to review knowledge of old tasks. The regularization-based methods [21, 23, 26, 27, 34, 50, 52] utilize the knowledge distillation technique [15], special regularization terms to avoid catastrophic forgetting. The other mask methods of continual learning are devoted to expanding the network capacity via introducing extra masks for each new task [18, 30–32, 41, 43, 48]. The knowledge of previous tasks can be preserved by sequentially increasing new masks (weight masks [18, 30–32, 48] or unit masks [41, 43]) and masking out parameters of old tasks simultaneously. For example, PackNet [31] iteratively performs pruning a well-trained base network and maintains binary sparsity masks to fix necessary parameters for incoming tasks. Piggyback [30] introduces binary masks on all parameters of a base network for each task without the forgetting problem. HAT [41] designs unit masks and keeps the feature embeddings of learned tasks to preserve information of old tasks. Our proposed MEAT builds upon mask methods: given a well-initialized transformer, we assign attention masks to the self-attention mechanism and a portion of parameters to fully leverage the architectural characteristics of ViTs for continual learning.

Besides incremental learning in computer vision mainly designed for CNN structures, a growing body of research in NLP has equipped the transformer with incremental learning. Adaptor-BERT [16] adds two fully connected layers as an adaptor in each layer and freezes old parameters (except for normalization layers) during training. Inspired by this work, B-CL [22] adopts capsules and dynamic routing [39] to transfer old knowledge to new tasks for aspect sentiment classification tasks. [17] presents an information disentanglement-based regularization method to further generalize old task knowledge. We also compare MEAT with Adaptor-BERT to verify the effectiveness of our methods.

3. Method

3.1. Preliminaries

A typical vision transformer is composed of three key components, *i.e.*, the trainable linear projection for embedding patch features, the multi-head self-attention (MHSA) block, and the feed-forward network (FFN) block. An input image is split into n small patches as image tokens, then mapped to a sequence of d-dimension vectors. A trainable class token is concatenated to the image token sequence for the final classification. The input sequence $\mathbf{X} \in \mathbb{R}^{(n+1)\times d}$ is fed into a stack of identical encoder layers. Each encoder layer consists of an MHSA block and an FFN block sequentially with residual connections. Specifically, the MHSA block with H heads can be formulated as

$$\text{MHSA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_H)\mathbf{W}^O, \quad (1)$$

where $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ are the query, key, value embeddings; $\text{head}_h \in \mathbb{R}^{(n+1)\times d_k}$ is the output of attention head h that satisfies $d_k = d/H$, and $\mathbf{W}_h^O \in \mathbb{R}^{d_k \times d}$ is the output projection matrix. The attention head h is calculated by

$$\text{head}_h = \mathbf{\Psi}_h \mathbf{V}_h = \sigma\left(\mathbf{A}_h\right)\mathbf{V}_h = \sigma\left(\frac{\mathbf{Q}_h \mathbf{K}_h^\top}{\sqrt{d_k}}\right)\mathbf{V}_h, \quad (2)$$

where $\mathbf{Q}_h = \mathbf{X}\mathbf{W}_h^Q$, $\mathbf{K}_h = \mathbf{X}\mathbf{W}_h^K$, and $\mathbf{V}_h = \mathbf{X}\mathbf{W}_h^V$ are linear projections of $\mathbf{X}$ by $\mathbf{W}_h^Q$, $\mathbf{W}_h^K$, and $\mathbf{W}_h^V \in \mathbb{R}^{d\times d_k}$ respectively; $\mathbf{A}_h = \mathbf{Q}_h\mathbf{K}_h^\top / \sqrt{d_k}$ is the dot-production attention matrix; $\sigma(\cdot)$ is the softmax activation function; $\mathbf{\Psi}_h \in \mathbb{R}^{(n+1)\times(n+1)}$ is an asymmetrical matrix, measuring the similarity between all the pairs of queries and keys by performing dot-production. For instance, the entry $\Psi_h^{i,j}$ of $\mathbf{\Psi}_h$ denotes the attention score that token i pays to token j.

The FFN block is composed of two linear layers and an activation function $\phi(\cdot)$ (*e.g.*, GELU [14]) and maps the input sequence $\mathbf{X}$ to

$$\text{FFN}(\mathbf{X}; \mathbf{W}_1, \mathbf{W}_2) = \phi(\mathbf{X}\mathbf{W}_1)\mathbf{W}_2, \quad (3)$$

where $\mathbf{W}_1 \in \mathbb{R}^{d\times d'}$, $\mathbf{W}_2 \in \mathbb{R}^{d'\times d}$ are projection matrices. The bias terms of MHSA and FFN are omitted for simplicity.

3.2. Meta-Attention

According to Eqn. 1 and 2, in the MHSA block, the output of each image token is dependent on all the input tokens. Thus the self-attention mechanism can be generally considered as a dense relationship within all the image token pairs. As a result, all image tokens in the same layer are involved for final classification regardless of the assigned tasks. In this paper, we refer to the token interaction pattern like $\mathbf{\Psi}_h$, which performs attention computation between all image token pairs equally and densely, as the *standard token interaction pattern*. The proposed MEta-ATtention (MEAT) aims to dynamically adapt the standard token interaction pattern to the new tasks via putting attention to self-attention. For simplicity and putting the focus on image token interactions, the class token is kept activated, and $\mathbf{\Psi}_h \in \mathbb{R}^{n\times n}$ only measures the relationship between image tokens. Furthermore, we also extend the mechanism of MEAT to the trained neurons of the FFN block by paying attention to each neuron, exploring a suitable sub-network of the initial trained weights to boost the continual learning performance.

3.2.1 Attention to Self-attention

For a well-initialization transformer as the base model, each image token interacts with each other in the standard information interaction form in the MHSA block for the initial old task. MEAT dynamically assigns an attention mask $\mathbf{m} \in \mathbb{R}^n$ with continuous values to modify the standard information interaction between all image tokens to learn adaptive communication patterns when sequentially studying new tasks

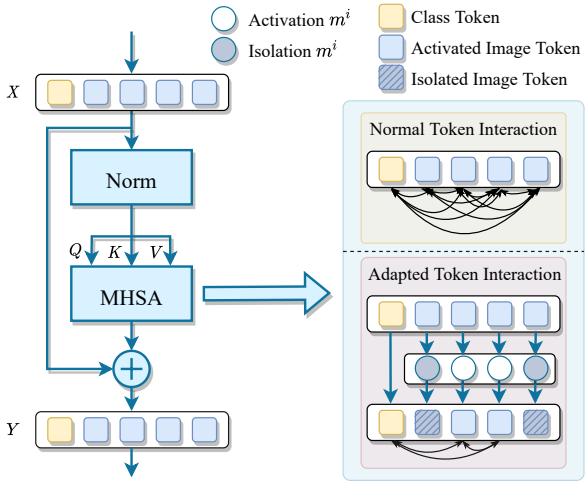

Figure 2. Illustration of the working mechanism of MEAT in the MHSA block of a transformer encoder layer. In the standard token integration, all the tokens interact with each other without limit. MEAT proposes attention masks to modify this communication pattern by dynamically activating and isolating image tokens. X and Y denote the input and the output sequence. Q, K and, V represent query, key, and value for the MHSA module.

with domain shifts from old tasks. In particular, for the i-th input image token, the i-th entry of mask $\mathbf{m}$, m^i, is used to modify the attention values related to the token i in this layer. Consequently, the MEAT mask $\mathbf{m}$ generates the adaptive token interaction pattern based on the standard information interaction $\mathbf{\Psi}_h$ in a token-wise manner. In the forward propagation, the softmax function σ in Eqn. 2 can be modified as an adaptive softmax function σ_A for calculating task specific attention, *i.e.*, $\mathbf{\Psi}_h = \sigma_A(\mathbf{A}_h)$. The i-th row of similarity $\mathbf{\Psi}_h$ can then be written as

$$\Psi_h^i = \left[\Psi_h^{i,j}\right]_{j=1}^n = \left[\frac{m^j \exp\left(A_h^{i,j}\right)}{\sum_{s=1}^n m^s \exp\left(A_h^{i,s}\right)}\right]_{j=1}^n. \quad (4)$$

Accordingly, $\Psi_h^{i,j}$ represents the modified attention that token i pays to token j via the attention mask m. The task-specific relationship provides a task-order invariant solution in modifying token interactions: when inference, each new task only employs the corresponding set of masks and the classifier without interference from other tasks.

With the increasing number of incoming tasks, the proposed MEAT mask with continuous values requires a lot of memory space. As shown in Figure 2, a binary value MEAT mask is adopted to replace the former continuous value mask. Specifically, for the token i, a binary variable, the attention entry $m^i \in \{0, 1\}$ modifies its adapted attention state, where 1 and 0 indicate whether token i is activated or not in the adapted token interaction pattern. With the binarized MEAT mask, $\Psi_h^{i,j}$ in Eqn. 4 can be computed as

$$\tilde{\Psi}_h^{i,j} = \begin{cases} \frac{\exp\left(A_h^{i,j}\right)}{\sum_{s=1}^n m^s \exp\left(A_h^{i,s}\right)}, & \text{if } m^j = 1; \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

Equipped with the attention masks, image tokens can be activated dynamically according to new tasks. However, the binary mask cannot be directly optimized along with the new task classifier when back-propagation, for it belongs to the non-differential categorical distribution, leading to the NP-hard problem [12]. To obtain the binary m^i, we hence introduce a differentiable variable parameterized by trainable $t^i \in \mathbb{R}^2$, then utilize a novel Gumbel-Softmax estimator [20] to approximate the discrete Binomial distribution while enabling gradient descent optimization as

$$m^i = \frac{\exp((\log(t^{i,1}) + g^1)/\tau)}{\sum_{k=1}^2 \exp((\log(t^{i,k}) + g^k)\tau)}, \quad (6)$$

where $\tau > 0$ is the temperature, g^k and g^0 are sampled from Gumbel distribution $g = -\log(-\log(u))$ with $u \sim \text{Uniform}(0, 1)$. The initial weights of t^i are independently sampled from distribution $\text{Uniform}(-\gamma, \gamma)$, where $\gamma > 0$ is a hyperparameter. By employing the Gumbel-Softmax trick, the proposed binary mask can be smoothly optimized along with the classifier of the new task via gradient descent.

For inference, updating weights is not required. The relaxation process is replaced with

$$m^i = \arg\max t^{i,j}. \quad (7)$$

In conclusion, MEAT assigns special-designed attention masks to the self-attention block to create adapted token interaction patterns in a task-specific manner by dynamically activating and isolating corresponding image tokens when incrementally learning new tasks. The binary-value mask substantially reduces additional overheads. The standard token interaction for the initial old task is a special adapted token interaction pattern that the values of each m^i are always set to 1.

3.2.2 Attention To Feed-Forward Network

To further boost the performance of task incremental learning, we extend the mechanism of MEAT to the trained neurons of the FFN block by paying attention to each neuron to explore a suitable sub-network of the initial trained weights. An attention mask designed for the FFN block is also proposed to dynamically activate and isolate neurons in each linear layer in the FFN block. A sub-network of the well-trained FFN block will be generated automatically on the data biases of incoming tasks. Let $\mathbf{W} \in \mathbb{R}^{d_1 \times d_2}$ represent the weight matrix trained on the initial old task of an arbitrary inner-layer in the FFN block, where d_1 and d_2 are the input

and output feature dimensions, respectively. Note that $\mathbf{W}$ has been optimized on the initial task. Instead of retraining $\mathbf{W}$ for new tasks, we customize an activation map to $\mathbf{W}$ to prevent catastrophic forgetting. Similar to the TI adaptor, for each neuron $w^{i,j}$ in the weight matrix W, the entry of the binary MEAT mask, $m^{i,j} \in \{0, 1\}$ stands for its activation state. When introducing a new task, the binary-valued adaptor is adopted to multiply with $w^{i,j}$ as

$$\tilde{w}^{i,j} = m^{i,j} w^{i,j} = \begin{cases} w^{i,j}, & \text{if } m^{i,j} = 1; \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

With the help of the binary adaptor, the weight can be preserved or discarded according to the new task. Similar to the optimization procedure of TI adaptors, the Gumbel-Softmax trick is also adopted to tackle the non-differential problem of the attention mask m like Eqn. 6.

3.3. Optimization Objective

The final optimization objective consists of two loss functions. The first one is the conventional cross-entropy loss $\mathcal{L}_{ce}(\hat{p}, p)$, where $\hat{p}$ and p are the predicted category distribution and the ground-truth label. Isolating image tokens may cause accuracy drops when isolated token numbers are beyond normal limits. A new drop-control loss is therefore introduced to prevent excessive patch dropping as

$$\mathcal{L}_{dc}(m) = \frac{1}{L} \sum_{l=1}^{L} \left(\lambda - \frac{1}{n} \sum_{i=1}^{n} m_l^i \right)^2, \tag{9}$$

where λ is a coefficient to adjust the expected activated token numbers. Eqn. 9 regulates the mask by λ, avoiding isolating too many image tokens at the early training stage, which tends to degrade performance. Let α indicate a weighting factor, the final optimization objective is summed as

$$\mathcal{L} = \mathcal{L}_{ce}(\hat{p}, p) + \alpha \mathcal{L}_{dc}(m). \tag{10}$$

4. Experiments

4.1. Experimental Settings

Datasets ImageNet [8] is set as the initial task. Six widely-used classification benchmarks are adopted as new visual tasks to be added on the ImageNet initialized vision transformers. Three fine-grained classification datasets are involved in demonstrating the performance of our method on images with finer granularity than ImageNet, including CUB [47], Stanford Cars [24] and FGVC-Aircraft [29]. CIFAR-100 [25], which has a category hierarchy like Imagenet, is also adopted. Sketches [11], WikiArt [40] serve as two art-related datasets which contain pictures of different domains from initial tasks. Besides, we further introduce a large-scale dataset, Places365 [53], to investigate the continually learning ability on large domain shifts data of MEAT. All input images are resized to 224×224 pixels.

Backbones In principle, the specially designed meta-attention mechanism can be applied to any vision transformers. In this paper, three popular transformer-based models are used as the backbone: DeiT-Ti [45], DeiT-S [45], T2T-ViT-12 [49]. We follow the official implementations of adopted three ViTs and insert our proposed attention masks to self-attention (only added in the MHSA block) and attention masks to FFN (only added in the FFN block) in each encoder layer. The official pre-trained weights on ImageNet serve as the well-initialization weights.

Parameters In the training stage, we basically follow the training strategies used in the official code of DeiT [45]. The initial learning rates of new classifiers and the binary MEAT masks are $\frac{batchsize}{1024} \times 5e^{-4}$ and $\frac{batchsize}{1024} \times 0.1$. All ViTs are trained for 30 epochs with a batch size of 256. γ, α and λ are set to 4, 2, and 0.9. All experimental results are averaged over 5 runs on 6 random sequences of new tasks.

Competitors We compare our proposed MEAT on three transformer-based models with the following competitors: (1) *Individual*; (2) *Classifier*; (3) *LwF* [27]; (4) *Piggyback* [30]; (5) *HAT* [41]; (6) *Adaptor-Bert* [16]. Among these baselines, *Individual* denotes that an independent ViT is trained for each task. As this method duplicates the model by the number of the tasks, its performance can serve as the upper bound of continual learning. *Classifier* is the simple continual learning strategy that finetunes only the classifier layer, *i.e.*, the last fully connected layer, for the new task. LwF, Piggyback, HAT and Adaptor-Bert are four representative existing methods from both computer vision and NLP for continual learing. For the initial ImageNet task, we directly utilize the official open-source pre-trained weights. For more details, please refer to the supplementary material.

4.2. Benchmark Comparison

Table 1 summarizes the main experimental results. Broadly speaking, our MEAT enjoys superior performance with the three ViTs on all the tasks compared to every competitor except *individual*. While *Individual* is ideal from the perspective of accuracy, it increases the model parameters by about 6 times as 6 more independent models are trained for the new tasks, which actually violates the setting of continual learning. Compared with other competitors of continual learning, MEAT adds and retrains only a small number of parameters (*i.e.*, binary masks). Specifically, *Classifier* does not introduce many extra parameters but shows poor results, especially on data with massive domain shifts from the old task. LwF and HAT require overall retraining of model parameters and suffer from the forgetting problem. Similar to our proposed method, Piggyback and Adaptor-Bert introduce some additional parameters (*i.e.*, masks or layers) for continual learning. However, Piggyback applies masks on all parameters. Adaptor-Bert applies two linear layers in each encoder layer leading to excessive extra parameters af-

	Dataset	Method						
		Individual	Classifier	LwF [27]	Piggyback [30]	HAT [41]	Adaptor-B [16]	MEAT
DeiT-Ti	CUB	75.13	46.05	59.03	60.65	68.34	66.03	**71.16**
	Cars	69.82	16.27	39.39	44.87	50.57	45.50	**53.42**
	FGVC	70.00	14.35	38.87	45.58	46.71	41.28	**52.69**
	WikiArt	72.13	38.64	46.88	62.42	61.84	57.04	**64.63**
	Sketches	73.50	30.64	53.17	69.07	65.49	69.21	**70.73**
	CIFAR-100	83.85	66.05	69.79	71.18	70.67	75.21	**78.13**
	ImageNet	30.82 (0.00)	72.20 (0.00)	26.24 (↓ 45.96)	72.20 (0.00)	N/A N/A	72.20 (0.00)	72.20 (0.00)
	Model Size	149 MB (6.49x)	23 MB (0.06x)	23 MB (1.00x)	26 MB (0.21x)	23 MB (1.01x)	29 MB (0.28x)	25 MB (0.16x)
DeiT-S	CUB	82.69	49.10	69.34	72.89	79.67	77.20	**81.53**
	Cars	84.74	18.29	74.00	74.72	73.22	67.23	**77.20**
	FGVC	82.69	15.51	55.99	60.04	62.99	57.04	**65.69**
	WikiArt	79.48	43.85	65.64	68.09	70.43	71.33	**73.43**
	Sketches	80.68	39.80	70.74	75.03	74.97	72.87	**76.68**
	CIFAR-100	89.03	72.71	75.67	79.76	79.52	84.00	**85.93**
	ImageNet	49.78 (0.00)	79.84 (0.00)	23.01 (↓ 56.83)	79.84 (0.00)	N/A N/A	79.84 (0.00)	79.84 (0.00)
	Model Size	582 MB (6.77x)	86 MB (0.03x)	86 MB (1.00x)	101 MB (0.15x)	86 MB (1.01x)	99 MB (0.17x)	96 MB (0.14x)
T2T-ViT-12	CUB	74.47	26.15	45.33	63.57	66.57	64.31	**69.90**
	Cars	72.67	11.52	59.01	58.22	54.63	53.79	**61.90**
	FGVC	64.09	12.46	42.07	51.47	52.69	48.02	**53.55**
	WikiArt	73.51	35.57	51.24	60.34	58.53	59.01	**61.20**
	Sketches	76.60	18.79	61.98	73.07	71.29	74.02	**74.75**
	CIFAR-100	85.03	33.10	66.34	70.98	74.86	73.58	**77.42**
	ImageNet	32.62 (0.00)	55.42 (0.00)	28.54 (↓ 26.88)	55.42 (0.00)	N/A N/A	55.42 (0.00)	55.42 (0.00)
	Model Size	179 MB (6.63x)	27 MB (0.07x)	27 MB (1.00x)	32 MB (0.20x)	28 MB (1.02x)	36 MB (0.30x)	30 MB (0.14x)

Table 1. Comparison of performance on six new tasks added on the initial ImageNet task with three vision transformers. With new tasks sequentially fed into the network, the results are averaged over 6 random orders according to 5 preset seeds. Note that Adaptor-B is the used Adaptor-Bert baseline. Bold fonts and blue fonts represent the best and second-best performance on each new task except Individual (the ideal setting), respectively. Red values marked with ↓ in parentheses denote the average performance deterioration on the ImageNet task after continually learning new tasks. Gray values in parentheses refer to the times (×) of retrained model sizes compared to Classifier.

New Task	Method	DeiT-Ti	DeiT-S	T2T-ViT-12
Places365	Individual	52.67	55.22	51.53
	Classifier	40.36	44.87	37.06
	LwF	42.17	45.13	39.77
	Piggyback	46.32	50.71	46.38
	MEAT	48.15	52.98	47.25

Table 2. Classification results (%) on Places365 dataset, which is added to the ImageNet pretrained transformers.

ter its average results. In conclusion, our approach balances well in transferring knowledge from the initial task, avoiding catastrophic forgetting, and economizing on parameters.

Another large-scale dataset, Places365 [53], is also introduced as a new task with significant domain shifts from the initial ImageNet task, as shown in Table 2. The experimental results keep nearly the same as those on small datasets. Our proposed MEAT boosts the performance by a large margin, meanwhile averting increasing too many model parameters.

4.3. Ablation Study

4.3.1 Effectiveness of Components

In Figure 3, we demonstrate how our method benefits from each designed component. Five model variants are listed in each sub-graph to denote three vision transformers equipped with different proposed modules or baselines used for com-

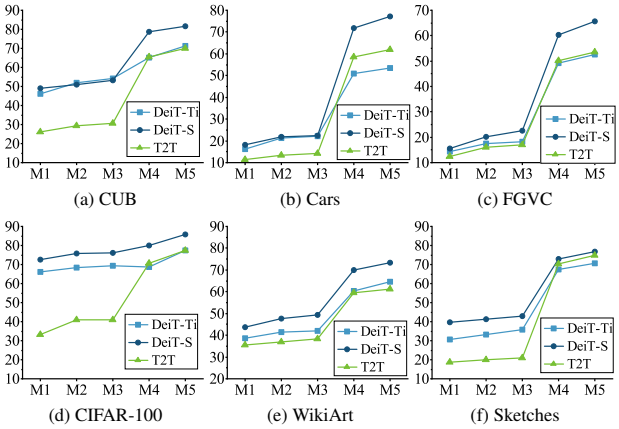

Figure 3. The effectiveness of each component in our method on six new tasks. In each sub-figure, accuracy (%) of fix model variants on the same dataset are plotted with three ViTs.

parison. Concretely, (M1) the Classifier baseline, which is the same as Table 1; (M2) transformers with MEAT masks on MHSA, without drop-control loss $\mathcal{L}_{dc}$ in Eqn. 9; (M3) transformers with MEAT masks on MHSA, with drop-control loss $\mathcal{L}_{dc}$; (M4) transformers with MEAT masks on neurons of the FFN block; (M5) the proposed MEAT. In three vision transformers, both MEAT masks on tokens and neurons, and the loss function efficiently improve the classification accuracy when adding new tasks compared to Classifier baseline (M1). The MEAT mask on image tokens with the proposed loss function (M3) achieves $2.67\% \sim 7.02\%$ performance boosts without modifying the trained weights, demonstrating the effectiveness of paying attention to the self-attention strategy. The attention mask on neurons (M4) promotes the results considerably by dynamically activating and deactivating pre-trained weights, customizing a sub-network of the complete transformer for each new task. In conclusion, each adopted component of our method consistently promotes continual learning performance.

4.3.2 Comparing to CNNs

As an emerging model used in computer vision, the ViTs have quite distinct architecture from CNNs. Given that most existing works on incremental learning are based on CNNs, we conduct the experiments on the CIFAR-100 dataset comparing CNNs (*i.e.*, VGG16-BN [42], ResNet50 [13] and EfficientNet-B4 [44]) and vision transformers using LwF, Piggyback, and our proposed MEAT as shown in Figure 4a. Since the MHSA block only exists in ViTs, in MEAT experiments with CNNs, we only apply the MEAT attention mask to all parameters pre-trained on ImageNet. It can be observed that CNN-based models and transformer-based models show better performance using these three approaches. Nevertheless, due to the absence of the attention mask on

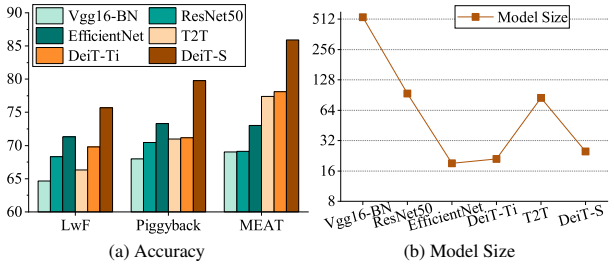

Figure 4. (a) Comparing results (%) over vision transformers and CNNs on CIFAR-100. (b) Model size (MB) comparison.

self-attention, CNNs under MEAT only have slightly better results than the CNNs under Piggyback. Significantly, the EfficientNet is an excellent network architecture with relatively small model sizes and better performance than DeiT-Ti and T2T in LwF and Piggyback experiments. Nevertheless, using the MEAT masks only achieves similar results with Piggyback. In contrast, three ViTs behave much better using MEAT than LwF and Piggyback, because MEAT not only introduces attention masks on a portion of parameters, but also assigns the special attention masks to self-attention and generates unique token interaction patterns for new tasks. We attribute this phenomenon to the fundamental idea of our work: paying attention to self-attention. And the mechanism of self-attention in ViTs builds dense and long-distance dependencies between all image patches. Applied with the MEAT masks to assign different attention values to image tokens, each new task can construct unique token interaction pattern with little overhead. However, the masks for CNNs only modifies the activation state of trained neurons, which lacks of modeling the long-distance relationship and brings less performance boosts than MEAT with ViTs. Please refer to the supplementary material for more ablation experimental results and analysis.

4.4. Analysis and Discussion

We visualize the trained binary masks of the 2-th layer and the 11-th encoder layer in Figure 5 on involved six datasets with three adopted ViTs. It can be observed that the activated and isolation states of the same token in the shallow and deep layer present clearly different patterns between datasets and the backbones . First, all the ViTs tend to activate more image tokens at shallow layers and isolate more tokens with the deepening of layers on all new tasks, which is reasonable that isolating too many tokens at shallow layers will cause severe information loss. The shallow layer mainly isolates the edge and the background tokens of input images, while the deep layer further isolates more central tokens and focuses on the target object region. For example, on CUB, only some edge tokens are isolated at the shallow layer; at the deep layer more background tokens are dropped and the body tokens of birds are more likely to be activated.

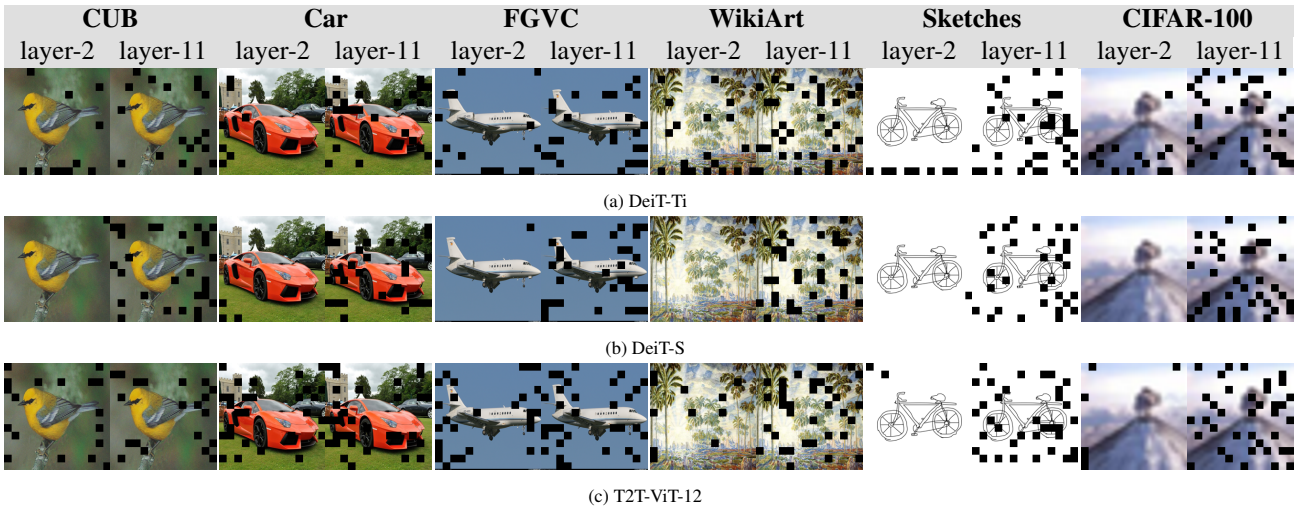

Figure 5. Visualization of the trained MEAT masks on the example images of the adopted six datasets at the 2-th layer and the 11-th layer of ViTs. The unchanged and black patches are the corresponding locationsfi of the activated and isolated (1 and 0 in masks) image tokens.

Another observation is that the trained masks reflect the features of the corresponding datasets. CUB, Car, and FGVC are fine-grained datasets, prone to isolate the background tokens and activate the central tokens containing the target (*e.g.*, the body of the bird, car, and plane). In contrast, the image of Sketches only contains simple lines and large blank space and it is prone to isolate more central tokens where the blank space frequently appears. Finally, the bigger model, DeiT-S, tends to retain more tokens than two smaller models at the shallow layer, like on CUB, it only isolates the tokens at the top right-hand corner. This indicates that big models preserve more token interaction at the shallow layers, benefiting the performances on new tasks in Table 1.

We also compare the activated ratios of image tokens and trained neurons averaged on all continual tasks in Figure 6. The attention masks on image tokens in Figure 6a tend to isolate more tokens at 1-th layer, activate more tokens at mid-layers, and gradually isolate more tokens at deep layers, which matches with observations of the visualization results. Moreover, it is noticing that the big model, DeiT-S, activates more image tokens than two smaller models at shallow layers, contributing to better results. Figure 6b gives the activated states of the MEAT masks on FFN neurons. Similar to the token mask, deep layers favor activating fewer neurons than shallow layers. Refer to supplementary material for more visualization results.

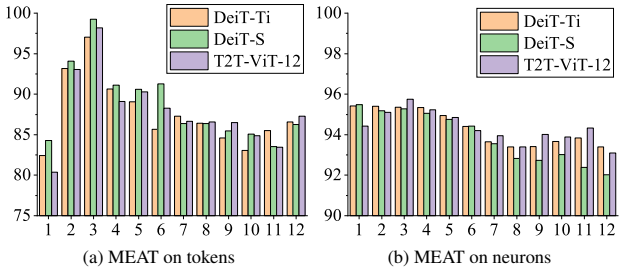

Figure 6. The activated rates of (a) image tokens in the MHSA block (b) neurons of the FFN block in each layer averaged on six new tasks of three ViTs using the MEAT attention mask.

5. Conclusion and Future Work

This paper presents MEAT, a novel task-continual learning method tailored for ViTs, to adapt a pre-trained ViT to new tasks. MEAT applies attention masks to image tokens in the MHSA block to adaptively generate unique token interaction patterns for new tasks. We further extend the MEAT mechanism to pay attention to neurons for exploring the suitable sub-networks in the FFN block per task. Thus MEAT fully leverages the architecture characteristics of ViTs with task-specific attention masks on self-attention and a portion of parameters without manual hyperparameter setting. Experiment results show that MEAT effectively improves the performance of continual tasks with little overhead of parameter storage and retraining. In future work, we will extend the proposed MEAT beyond the task-continual learning to make further improvements of ViTs in continual learning.

Acknowledgements. This work is funded by the National Key R&D Program of China (Grant No: 2018AAA0101503) and the Science and technology project of SGCC (State Grid Corporation of China): fundamental theory of human-in-the-loop hybrid-augmented intelligence for power grid dispatch and control.

References

[1] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 2

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229. Springer, 2020. 2

[3] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and M Ranzato. Continual learning with tiny episodic memories. 2019. 1, 3

[4] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *CVPR*, pages 12299–12310, 2021. 2

[5] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *CVPR*, pages 1601–1610, 2021. 2

[6] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from non-stationary data streams. In *ICCV*, pages 8250–8259, 2021. 1, 3

[7] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *PAMI*, 2021. 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 2, 5

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2

[11] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *TOG*, 31(4):1–10, 2012. 2, 5

[12] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM (JACM)*, 48(4):798–859, 2001. 4

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 7

[14] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 3

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[16] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, pages 2790–2799. PMLR, 2019. 3, 5, 6

[17] Yufan Huang, Yanzhe Zhang, Jiaao Chen, Xuezhi Wang, and Diyi Yang. Continual learning for text classification with information disentanglement based regularization. *arXiv preprint arXiv:2104.05489*, 2021. 3

[18] Steven CY Hung, Cheng-Hao Tu, Cheng-En Wu, Chien-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. Compacting, picking and growing for unforgetting continual learning. *arXiv preprint arXiv:1910.06562*, 2019. 1, 3

[19] David Isele and Akansel Cosgun. Selective experience replay for lifelong learning. In *AAAI*, volume 32, 2018. 1, 3

[20] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 2, 4

[21] Heechul Jung, Jeongwoo Ju, Minju Jung, and Junmo Kim. Less-forgetting learning in deep neural networks. *arXiv preprint arXiv:1607.00122*, 2016. 1, 3

[22] Zixuan Ke, Hu Xu, and Bing Liu. Adapting bert for continual learning of a sequence of aspect sentiment classification tasks. In *NAACL*, pages 4746–4755, 2021. 3

[23] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. 1, 3

[24] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV Workshops*, pages 554–561, 2013. 2, 5

[25] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2, 5

[26] Sang-Woo Lee, Jin-Hwa Kim, Jaehyun Jun, Jung-Woo Ha, and Byoung-Tak Zhang. Overcoming catastrophic forgetting by incremental moment matching. *arXiv preprint arXiv:1703.08475*, 2017. 1, 3

[27] Zhizhong Li and Derek Hoiem. Learning without forgetting. *PAMI*, 40(12):2935–2947, 2017. 1, 3, 5, 6

[28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *ICCV*, 2021. 2

[29] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 2, 5

[30] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *ECCV*, pages 67–82, 2018. 1, 2, 3, 5, 6

[31] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *CVPR*, pages 7765–7773, 2018. 1, 2, 3

[32] Marc Masana, Tinne Tuytelaars, and Joost van de Weijer. Ternary feature masks: zero-forgetting for task-incremental learning. In *CVPR*, pages 3570–3579, 2021. 1, 2, 3

[33] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 1

[34] Inyoung Paik, Sangjun Oh, Taeyeong Kwak, and Injung Kim. Overcoming catastrophic forgetting by neuron-level plasticity control. In *AAAI*, volume 34, pages 5339–5346, 2020. 1, 3

[35] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. 2

[36] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, pages 2001–2010, 2017. 1, 3

[37] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P Lillicrap, and Greg Wayne. Experience replay for continual learning. *arXiv preprint arXiv:1811.11682*, 2018. 1, 3

[38] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 1

[39] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. *arXiv preprint arXiv:1710.09829*, 2017. 3

[40] Babak Saleh and Ahmed Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *arXiv preprint arXiv:1505.00855*, 2015. 2, 5

[41] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *ICML*, pages 4548–4557. PMLR, 2018. 1, 2, 3, 5, 6

[42] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014. 7

[43] Pravendra Singh, Vinay Kumar Verma, Pratik Mazumder, Lawrence Carin, and Piyush Rai. Calibrating cnns for lifelong learning. In *NeurIPS*, 2020. 1, 3

[44] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, pages 6105–6114. PMLR, 2019. 7

[45] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, pages 10347–10357. PMLR, 2021. 2, 5

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 2

[47] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. 2, 5

[48] Jaehong Yoon, Saehoon Kim, Eunho Yang, and Sung Ju Hwang. Scalable and order-robust continual learning with additive parameter decomposition. *arXiv preprint arXiv:1902.09432*, 2019. 3

[49] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021. 2, 5

[50] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *ICML*, pages 3987–3995. PMLR, 2017. 1, 3

[51] Dong Zhang, Hanwang Zhang, Jinhui Tang, Meng Wang, Xiansheng Hua, and Qianru Sun. Feature pyramid transformer. In *ECCV*, pages 323–339. Springer, 2020. 2

[52] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C-C Jay Kuo. Class-incremental learning via deep model consolidation. In *WACV*, pages 1131–1140, 2020. 1, 3

[53] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *PAMI*, 40(6):1452–1464, 2017. 2, 5, 6