


Learnable Masked Tokens for Improved Transferability of Self-Supervised Vision Transformers^{*}

Hao Hu¹, Federico Baldassarre¹, and Hossein Azizpour¹

KTH Royal Institute of Technology, Stockholm, Sweden
{haohu, fedbal, azizpour}@kth.se

Abstract. Vision transformers have recently shown remarkable performance in various visual recognition tasks specifically for self-supervised representation learning. The key advantage of transformers for self supervised learning, compared to their convolutional counterparts, is the reduced inductive biases that makes transformers amenable to learning rich representations from massive amounts of unlabelled data. On the other hand, this flexibility makes self-supervised vision transformers susceptible to overfitting when fine-tuning them on small labeled target datasets. Therefore, in this work, we make a simple yet effective architectural change by introducing new learnable masked tokens to vision transformers whereby we reduce the effect of overfitting in transfer learning while retaining the desirable flexibility of vision transformers. Through several experiments based on two seminal self-supervised vision transformers, SiT and DINO, and several small target visual recognition tasks, we show consistent and significant improvements in the accuracy of the fine-tuned models across all target tasks.

Keywords: Vision Transformer · Transfer Learning · Computer Vision.

1 Introduction

Deep learning on small datasets usually relies on transferring a model that is pretrained on a large-scale source task [25]. Recent concurrent advancements in transformers [1] and self-supervised pretraining [10, 12, 11, 9] have made self-supervised Vision Transformers (ViTs) a viable alternative to supervised pretraining of Convolutional Networks (ConvNets) [5–7]. Mainly based on self-attention [1, 4] and multi-layer perceptron, ViTs have shown improved performance over the state-of-the-art ConvNets on large datasets [2, 62, 3, 64] while retaining computational efficiency [23, 24]. Considering that collecting large volumes of unlabeled data is becoming increasingly easier, a practical approach for transfer learning would be to pretrain ViTs with self-supervision and then fine-tune them on the downstream task with a small amount of labeled data.

^{*} This work is partially supported by KTH Digital Futures and Wallenberg AI, Autonomous Systems and Software Program (WASP).

The supremacy of ViTs for self-supervised learning over ConvNets can be attributed to the reduced inductive biases of ViTs which facilitates learning from the abundance of unlabelled data that is commonly available for self-supervised learning. However, this comes at a cost. That is, such flexibility of ViTs makes a fully-fledged fine-tuning of them on small target datasets susceptible to overfitting. This is due to the fact that the dense self-attention among image patches in ViTs is more likely to find spurious patterns in small datasets. This makes the locality and sparsity inductive biases of ConvNets, in contrast to ViTs, crucial for fine-tuning on small amount of labelled data.

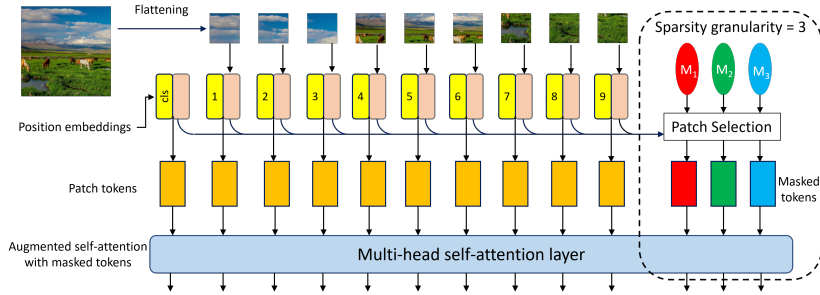


Fig. 1: An overview of the vision transformer with masked tokens. The part in the dashed rectangle is the proposed structural augmentation.

Therefore, in this paper, we aim at alleviating overfitting of fine-tuning self-supervised ViTs on small, domain-specific target sets while preserving their flexibility when learning from large unlabeled data. To this purpose, we propose *masked tokens*, a simple and flexible structural augmentation for self-attention layers. Each masked token aggregates a selected *subset* of patches to draw out sparse informative patterns. By varying the subset size from small to large, masked tokens encode the spatial information at different sparsity levels. We augment a self-attention layer by adding all the masked tokens to regular ones such that its output contains not only dependencies between patches, but also among different sparsity levels. Furthermore, we employ a data-driven method and two regularization techniques to *learn* the patch selection function for each masked token that can select patches with the most informative sparse patterns. The introduced sparsity makes the fine-tuning less prone to overfitting while the learnt selectivity retains the benefits of ViTs. Importantly, the proposed masked tokens are trained to encode details from local regions, reminiscent of the locality bias in the convolutional layers but with two key differences that the locality (i) can be learnt and (ii) can happen at various levels.

We summarize our contributions as below.

- We mitigate the overfitting of fine-tuned self-supervised ViTs by integrating sparsity and locality biases of ConvNets through masked tokens.

- We propose data-driven mechanisms to dynamically select the local region individually for each masked token and at different sparsity levels.
- We conduct extensive experiments on two self-supervised ViTs and various target tasks which show effectiveness of learnable masked tokens for ViTs.

2 Vision Transformers with Learnable Masked Tokens

2.1 Background: Vision Transformers

Given an input image $\mathbf{I} \in \mathbb{R}^{H \times W}$, it is divided into T non-overlapping patches $\{\mathbf{p}^i \in \mathbb{R}^{P \times P}\}_T$ and flattened into a sequence, where $T = \lceil \frac{HW}{P^2} \rceil$. A transformer [1] consists of L identical blocks with residual connections [33, 34]. Each block processes the input patch sequence $\{\mathbf{p}^i\}$ as

$$\mathbf{Z}_0 = [\mathbf{h}_{cls}; \mathcal{F}(\mathbf{p}^1) + \mathbf{e}^1; \dots; \mathcal{F}(\mathbf{p}^T) + \mathbf{e}^T], \quad (1)$$

$$\mathbf{Z}'_l = \text{MSA}(\text{LN}(\mathbf{Z}_{l-1})) + \mathbf{Z}_{l-1}, \quad (2)$$

$$\mathbf{Z}_l = \text{MLP}(\text{LN}(\mathbf{Z}'_l)) + \mathbf{Z}'_l, \quad (3)$$

$$\mathbf{y} = \text{softmax}(\text{MLP}(\mathbf{Z}_L^0)), \quad (4)$$

where MSA, MLP, LN and $\text{softmax}(\cdot)$ respectively indicate multi-head self-attentions, MLP with GELU, layer normalization and softmax. $\mathcal{F}(\cdot)$ is a convolutional feature extractor and $\{\mathbf{e}^i\}_T$ are position embeddings. Further, a learnable class token $\mathbf{h}_{cls} \in \mathbb{R}^d$ is used at the beginning of the sequence to globally represent entire image by taking an attention-weighted sum of every patch.

We augment ViT blocks by introducing *masked tokens*, which can alleviate overfitting by masking out redundant regions of input images. Given the sequence $\mathbf{Z}_l \setminus \mathbf{h}_{cls} = [\mathbf{z}_l^1; \dots; \mathbf{z}_l^T]$ of input tokens for layer $l + 1$, we construct N masked tokens $\{\mathbf{s}_l^j\}_N$ via selecting and aggregating a subset of patch tokens for each \mathbf{s}_l^j using a selection function $\mathcal{G}(\cdot, \cdot)$. More specifically, the subset $\mathbf{S}_l^j \subset \mathbf{Z}_l$ of tokens selected for \mathbf{s}_l^j can be presented as

$$\mathbf{S}_l^j = \{\mathbf{z}_l^{i_1}, \dots, \mathbf{z}_l^{i_{M_j}}\}, M_j = \lceil j \cdot \frac{T}{N} \rceil, \quad (5)$$

where M_j defines the sparsity level of \mathbf{s}_l^j indicating the size of the informative sub regions for encoding. Then \mathbf{s}_l^j can be produced by aggregating \mathbf{S}_l^j using any permutation-invariant pooling. We use mean pooling in this work. Finally, the generated masked tokens for this layer are appended at the end of \mathbf{Z}_l :

$$\tilde{\mathbf{Z}}_l = [\mathbf{h}_{cls}; \mathbf{z}_l^1; \dots; \mathbf{z}_l^T; \mathbf{s}_l^1; \dots; \mathbf{s}_l^N]. \quad (6)$$

This augmented $\tilde{\mathbf{Z}}_l$ is then fed to the MSA layer instead of the original \mathbf{Z}_l . For two consecutive augmented MSA layers¹ l and $l + 1$, where masked tokens \mathbf{s}_l^j

¹ We omit LN and MLP layers in between for convenience

and self-attention output \mathbf{z}_l^{T+j} are both there, we combine these two parts using a weighted summation as

$$\tilde{\mathbf{s}}_l^j = \alpha \mathbf{s}_l^j + (1 - \alpha) \mathbf{z}_l^{T+j}, \quad (7)$$

where α is a hyper-parameter set to 0.2 as default. In such cases, $\tilde{\mathbf{s}}_l^j$ will replace \mathbf{s}_l^j in equation (6) as masked tokens.

Fig. 1 illustrates the workflow of an augmented self-attention. We name the number of masked tokens N as the *sparsity granularity* since it spans the sparsity level, from sparse to dense, that masked tokens cover. It is also worth noting that despite the fact that a single masked token can only encode information for a region, with multi-head MSA layers, it can be extended to multiple regions.

2.2 Learning the Selection Function

We propose a data-driven approach to learn the patch selection function $\mathcal{G}(\cdot, \cdot)$ such that it can choose the most informative patches for masked tokens. We reformulate it as a corresponding ranking problem, where each masked token \mathbf{s}_l^j takes the top M_j patch tokens based on ranking scores $\mathbf{o}_l^j = \{o_l^{i,j}\}_T$. To obtain \mathbf{o}_l^j , we define a set of new parameters $\{\mathbf{w}_l^j \in \mathbb{R}^d\}_N$ to dot-product with each \mathbf{z}_l^i , whose score $o_l^{i,j}$ can be computed as

$$o_l^{i,j} = (\mathbf{z}_l^i)^\top \mathbf{w}_l^j. \quad (8)$$

We name \mathbf{w}_l^j as *Masked Query Embedding* (MQE), it can be seen as a learned query that selects the (masked) tokens. It is worth mentioning that similar to position embedding, when N is changed, \mathbf{w}_l^j can be interpolated to match the new sparsity granularity. Now the selection function $\mathcal{G}(\cdot, \cdot)$ can be further defined as

$$\mathcal{G}(\mathbf{z}_l^{1:T}, M_j; \mathbf{w}_l^j) = \text{argsort}(\mathbf{o}_l^j) \mathbf{z}_l^{1:T} |_{1:M_j}, \quad (9)$$

where $\text{argsort}(\cdot)$ returns a $T \times T$ matrix whose rows are one-hot vectors, indicating the location of i -th largest value at the i -th row. $\cdot |_{1:M_j}$ means it takes only the top M_j rows as the output.

To overcome the discrete nature of $\text{argsort}(\cdot)$, we approximate it by a differentiable relaxation named SoftSort [14], denoted by $\text{SS}(\cdot)$:

$$\text{SS}(\mathbf{o}_l^j) = \text{softmax} \left(\frac{|\text{sort}(\mathbf{o}_l^j) \mathbf{1}^\top - \mathbf{1}(\mathbf{o}_l^j)^\top|}{\tau} \right), \quad (10)$$

where $\text{softmax}(\cdot)$ is applied row-wise. $\text{sort}(\cdot)$ returns a sorted input. $|\cdot|$ takes element-wise absolute value and τ is a temperature set to 0.1 by default. By replacing argsort with SS , \mathbf{w}_l^j can be learnt jointly with other network weights.

2.3 Regularizations on Masked Tokens

We additionally introduce two regularizations that directly work on masked tokens to stabilize the training. First, to avoid masked tokens collapsing due to overlapping patches [11, 52], we add a linear classifier $\mathcal{C}(\cdot)$ at the top of the last layer to identify the sparsity index j associated to each masked token features. This is trained with the cross-entropy loss $\mathcal{L}_{\text{sparse}}$:

$$\mathcal{L}_{\text{sparse}} = -\frac{1}{N} \sum_{j=1}^N \delta_j^T \log(\mathcal{C}(\mathbf{z}_L^{T+j})), \quad (11)$$

where δ_j is the one-hot vector with one at element j . Conversely, we use contrastive loss [13], to have masked tokens of the same image with maximal similarities to each other. Specifically, given a batch of images $\{\mathbf{I}_k\}_K$, we consider any pair of the form $(\mathbf{z}_{k_1}^{T+j_1}, \mathbf{z}_{k_1}^{T+j_2})$ as a positive pair², and the rest as negative pairs. We compute the contrastive loss \mathcal{L}_{con} between the positive pairs like

$$\mathcal{L}_{\text{con}}(\mathbf{z}_{k_1}^{T+j_1}, \mathbf{z}_{k_1}^{T+j_2}) = -\frac{\exp(\text{cs}(\mathbf{z}_{k_1}^{T+j_1}, \mathbf{z}_{k_1}^{T+j_2}))}{\exp(\text{cs}(\mathbf{z}_{k_1}^{T+j_1}, \mathbf{z}_{k_1}^{T+j_2})) + \sum_{j,k \neq k_1}^{N \cdot (K-1)} \exp(\text{cs}(\mathbf{z}_{k_1}^{T+j_1}, \mathbf{z}_k^{T+j}))}, \quad (12)$$

and the total contrastive loss $\mathcal{L}_{\text{total_con}}$ as

$$\mathcal{L}_{\text{total_con}} = \frac{1}{K \cdot N \cdot (N-1)} \sum_{k=1}^K \sum_{j_1=1}^N \sum_{j_2 \neq j_1}^N \log \mathcal{L}_{\text{con}}(\mathbf{z}_k^{T+j_1}, \mathbf{z}_k^{T+j_2}), \quad (13)$$

here $\text{cs}(\cdot)$ is a cosine-similarity function.

3 Experiments

In this section we evaluate the effectiveness of learning masked tokens on various image benchmarks. This aim of the proposed modifications is to improve the transferability of self-supervised ViTs. Thus, we mainly focus on fine-tuning pretrained models on small datasets. We only consider image-level classification tasks to simplify the architectural choices of the backbone.

3.1 Configurations

Baselines. We apply two state-of-the-art self-supervised ViTs as our pretraining schemes and baselines: SiT [8] and DINO [9].

- **SiT** [8] replaces the class token \mathbf{h}_{cls} with two tokens, namely a rotation token \mathbf{h}_{rot} and a contrastive token \mathbf{h}_{contr} , such that it can be trained by predicting image rotations [15] and maximizing the similarity between positive pairs [13]. Furthermore, it features another regularization task where corrupted inputs are reconstructed via inpainting.

² We remove the layer index l , and replace it with the image index k for convenience

Table 1: Top-1 accuracy (%) for linear evaluations on CIFAR datasets. All the baseline performance are reported from [8].

Method	Backbone	CIFAR-10	CIFAR-100
DeepCluster [41]	ResNet-32	43.31	20.44
RotationNet [15]	ResNet-32	62.00	29.02
Deep InfoMax [42]	ResNet-32	47.13	24.07
SimCLR [12]	ResNet-32	77.02	42.13
Rel. Reasoning[43]	ResNet-32	74.99	46.17
Rel. Reasoning[43]	ResNet-56	77.51	47.90
SiT [8]	ViT-B/16	81.20	55.97
MT SiT (ours)	ViT-B/16	81.98	57.18

- **DINO** [9] takes a self-distillation paradigm by simultaneously updating the teacher with an exponential moving average and encouraging the student to have similar outputs as the teacher. Such objective is further optimised using multi-crops augmentation to ensure consistency between different scales.

Implementations. We implement our proposed augmentations based on their officially released codes in PyTorch. Our pilot studies show that augmenting many layers with masked tokens will show diminishing return. Thus, unless specified otherwise, we use a single ViT variant by replacing the MSA layers in the last four blocks with the augmented ones, and set the default sparsity granularity to 4. To reduce the computational cost we only do the token selection for the first of the four augmented blocks. For both baselines, we refer to their augmented ones with the prefix “MT”. All experiments are done with 8 Nvidia A100 GPUs.

3.2 In-domain Transfer Learning

We first present the results using the SiT-based [8] pretraining on three datasets: CIFAR-10/100 [17] and STL-10 [19]. For a fair comparison, we follow the same experimental protocols as [8] including random seeds, hyper-parameters and data augmentations. In this way, we first train the model on the entire dataset using SiT losses, then fine-tune the model on a fully labeled subset. Since both source and target are from the same domain, we refer it as **In-domain Transfer Learning** (IdTL) in the rest of the paper. ViT-B/16 will be our default backbone.

IdTL for CIFAR-10 and CIFAR-100

Linear evaluation. We first report the linear evaluation results in Tab. 1 to make sure that masked tokens won’t degrade the pretrained features due to additional model complexities. As we can see, MT SiT can outperform ConvNet-based methods by significant margins of 4.47 percentage points on CIFAR-10 and 9.28 on CIFAR-100. However, such gains become smaller when compared with SiT,

Table 2: Top-1 accuracy (%) of IdTL on CIFAR datasets. Referred to as 'few-shot' in [8].

Method	1%	10%	25%
CIFAR-10			
[43]	76.55	80.14	85.30
SiT [8]	74.78	87.16	92.90
MT SiT (ours)	82.52	92.23	95.60
CIFAR-100			
[43]	46.10	49.55	54.44
SiT [8]	27.50	53.72	67.58
MT SiT (ours)	24.51	61.39	72.69

with only 0.78 for CIFAR-10 and 1.21 percentage point for CIFAR-100. This supports the assumption that the architectural change of introducing masked tokens would not make overfitting worse.

Fine-tuning. Following [8], we fine-tune the MT SiT on subsets with different percentage of available labels. From Tab. 2, we can observe significant improvements over the SiT baseline in most cases. More specifically, we achieve 7.74, 5.07 and 2.70 percentage point improvements on CIFAR-10 with only 1%, 10% and 25% percent of labels. Moreover, in most cases MT SiT can achieve higher performance gain over the vanilla SiT when fine-tuning labels become much less, indicating the positive effects for reducing overfitting brought by masked tokens. On the other hand, while we can find similar improvements on CIFAR-100 with 10% and 25% labels, MT SiT performs worse than the SiT baseline and has a nearly 20 percentage point gap with the ConvNet baseline [43]. We argue that this is due to too few training samples to learn meaningful patterns on the target set. In such cases, fine-tuning can have a high variance and furthermore attentions between masked tokens may put an overly strong emphasis on localities, causing the drop in transferability.

IdTL for STL-10 Now we consider the STL-10 [19] dataset, which contains 100,000 unlabeled and 5,000 labeled training images. Thus, compared to CIFAR, it almost doubles the pretraining size while keeping the target set small. We directly fine-tune our models with all training labels without further dividing them into various subsets.

Fine-tuning. Tab. 3 summarizes the fine-tuning results for STL-10. Similar to the CIFAR, MT SiT consistently outperforms the SiT and other ConvNet baselines with a small 1.84 percentage points margin, showing the relative effectiveness of involving masked tokens for fine-tuning. Moreover, the experiments on three popular benchmarks, so far, suggest that ViTs could benefit from masked tokens for IdTL on small datasets.

Table 3: IdTL comparisons with SOTAs on STL-10 dataset.

Method	Backbone	Fine-tuning (%)
Exemplars [37]	Conv-3	72.80
Artifacts [38]	Custom	80.10
ADC [39]	ResNet-34	56.70
Invariant Info Clustering [40]	ResNet-34	88.80
DeepCluster [41]	ResNet-34	73.37
RotationNet [15]	ResNet-34	83.22
Deep InfoMax [42]	AlexNet	77.00
Deep InfoMax [42]	ResNet-34	76.03
SimCLR [12]	ResNet-34	89.31
Relational Reasoning [43]	ResNet-34	89.67
SiT [8]	ViT-B/16	93.02
MT SiT (ours)	ViT-B/16	94.86

Table 4: Ablation studies of pretraining the MT SiT with different components on the STL-10 dataset.

Method	MT	$\mathcal{L}_{\text{sparse}}$	\mathcal{L}_{con}	$\mathcal{G}(\cdot, \cdot)$	Linear	Fine Tuning
SiT [8]	-	-	-	-	78.58	93.02
MT SiT (ours)	✓				71.95	94.22
	✓	✓			68.77	93.89
	✓		✓		69.47	94.00
	✓			✓	77.71	94.44
	✓	✓	✓		78.75	94.78
	✓	✓	✓	✓	78.99	94.86

Ablation study. We further perform ablation studies on STL-10 to understand how each component affects the performance. The corresponding results are listed in Tab. 4. Although the fine-tuning accuracy can be boosted by any of the individual components, randomly selecting patches (as opposed to \mathcal{G}), pretraining with no or partial regularizers has produced worse self-supervised features than the complete model. Therefore, all proposed components seem to be important for achieving the best performance.

Visualization. This additional qualitative study investigates how the learnt masked regions are spatially distributed by visualizing selected patches of each masked token at the first augmented block. We randomly sample 10 examples from STL-10 and highlight the positions of selected patches using different colors for each masked token in Fig. 2. Overall, in most cases, the majority of patches in the same sparsity level are spatially close to each other, forming local clusters that cover multiple small regions. This lends evidence that masked tokens can indeed

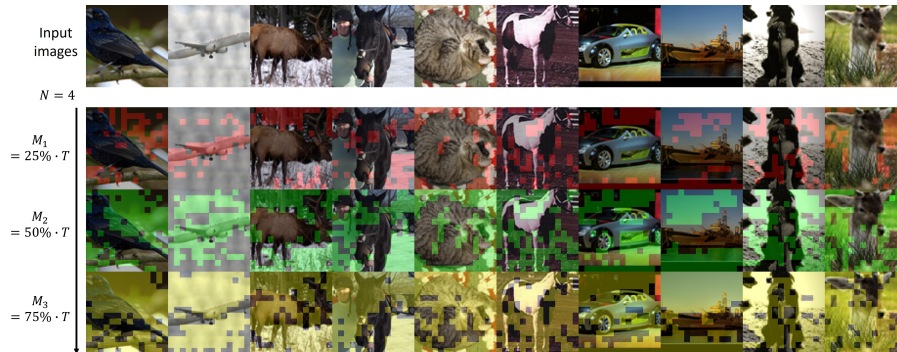


Fig. 2: Visualized examples from STL-10 datasets showing selected patch tokens at the first augmented MSA layer for each sparsity level using the learnt selection.

Table 5: Accuracy on ImageNet-1K. We list both results that reported from [9] (top) and trained by ourselves (bottom) to assure comparison is fair. ‘FT’ means ‘fine-tune’.

Method	Arch	KNN	Linear	FT-10%	FT-20%
BYOL [11]	ViT-S	66.6	71.4	N/A	N/A
MoCov2 [10]	ViT-S	64.4	72.7	N/A	N/A
SwAV [16]	ViT-S	66.3	73.5	N/A	N/A
DINO [9]	ViT-S	73.3	76.0	N/A	N/A
DINO (300 epochs)	ViT-S/16	73.06	75.83	58.48	68.46
MT DINO (4 masked tokens)	ViT-S/16	73.10	75.93	57.71	68.69
MT DINO (28 masked tokens)	ViT-S/16	73.08	75.90	60.26	70.02

encode local information from the informative sub-regions at various sparsity levels in the image. On the other hand, it is also surprising to see that low-level tokens tend to select the patches lying outside of the main interested object in many cases, which is slightly counter-intuitive. We conjecture about this observation as the usefulness of the associated contexts for reducing overfitting. While global attentions are pretrained to focus on the interested object due to large pretraining samples, the secondary contents also becomes informative and complementary when the training size decreases during the fine-tuning. Low-level masked tokens are flexible enough to be tuned to capture such information.

IdTL for ImageNet-1K Here we consider a larger pretraining source i.e., ImageNet-1K [18], which serves as a fundamental pretraining source for many

small datasets. Since SiT [8] doesn’t report ImageNet-pretrained results, we switch to another state-of-the-art baseline DINO [9] to avoid any setting inconsistencies. Besides, due to our hardware limitations, we can only afford to train DINO and MT DINO using ViT-small (ViT-S) [2]. Similar to previous experiments, we follow the same protocol provided by the official baseline implementation.

KNN and linear classification. Like [9], we do KNN ($K = 20$) and linear classification for self-supervised features first, whose results are in the middle two columns in Tab. 5. Compared with the baseline, MT DINO does not show clear improvements for either KNN or linear evaluation, implying that masked tokens may not be necessarily helpful when the pretraining size is large enough, as even the most informative localities are likely to be modelled by global attentions.

Fine-tuning with increased number of masked tokens. We further inspect the fine-tuning on two subsets of ImageNet-1K with only 10% and 20% labels, and report their accuracy in the last two columns of Tab. 5. Surprisingly, we initially find MT DINO with default number of scale tokens are outperformed by the baseline on 10% labeled subset with a 0.77 percentage point margin. We then increase the sparsity granularity up to 28 and find the performances are boosted by 2.55 and 1.33 for each subset. We speculate that as the dataset size grows, there are enough samples for the global patch tokens to model some sparse and local patterns, therefore, more masked tokens are needed to become complementary in addition to the standard tokens. Thus, a proper sparsity granularity is also important. Moreover, comparing with Sec. 3.2, the performance gain significantly drops, implying masked tokens may become less effective as dataset size increases. This also coincides with [2] that ViTs may beat ConvNets as training set size grows.

Costs for introducing masked tokens. Here we briefly discuss the additional model complexity and time consumption added by masked tokens. It is easy to see that the only new model weights are MQE for the first augmented block, bringing around $1\%(4/384)$ more parameters than any projections of a self-attention layer in our implementation. Thus, the computational overhead of masked token is quite negligible. Meanwhile, the inference time using 4 and 28 tokens increases 1 and 5 seconds respectively on the entire ImageNet validation set, showing that the extra computational costs don’t affect the ViT’s efficiency too much.

3.3 Cross-domain Transfer Learning

We now conduct experiments of transferring ImageNet-pretrained ViTs to various domain-specific datasets, which is closer to the mainstream transfer learning applications. Compared to Sec. 3.2, the target datasets exhibit a significant domain shift from the source, making them more challenging. Thus, we refer to such tasks as **Cross-domain Transfer Learning** (CdTL) in contrast to IdTL. We continue using ViT-S/16 [2] as the backbone and DINO [9] for self-supervised pretraining on ImageNet-1K.

Table 6: CdTL performance comparing with SOTA baselines for fine-grained recognition on CUB200-2011 dataset. The input size is 448. Baselines that outperform MT DINO are underlined.

Method	Backbone	supervised pretraining?	Accuracy (%)
RA-CNN [30]	VGG-19	✓	85.30
ResNet-50 [6]	ResNet-50	✓	85.50
M-CNN [31]	VGG-16	✓	85.70
GP-256 [45]	VGG-16	✓	85.80
MaxEnt [46]	DenseNet161	✓	86.60
DFL-CNN [47]	ResNet-50	✓	87.40
Nts-Net [32]	ResNet-50	✓	<u>87.50</u>
Cross-X [50]	ResNet-50	✓	<u>87.70</u>
DCL [49]	ResNet-50	✓	<u>87.80</u>
CIN [48]	ResNet-101	✓	<u>88.10</u>
ViT [2]	ViT-B/16	✓	<u>90.80</u>
TransFG [51]	ViT-B/16	✓	<u>91.70</u>
DINO[9]	ViT-S/16	✗	86.47
MT DINO	ViT-S/16	✗	86.68
MT DINO (28 masked tokens)	ViT-S/16	✗	87.38

Comparison with the State-of-The-Art We compare the MT DINO with DINO and other related baselines on four small datasets from three different domains, CUB-200-2011 birds [20] for fine-grained recognition, SoybeanLocal and Cotton80 [21] for ultra fine-grained recognition, and COVID-CT [22] for medical imagery-based diagnosis.

Fine-Grained classifications (FG). Tab. 6 lists the accuracy for MT DINO and other state-of-the-art baselines on CUB birds dataset. MT DINO can achieve ~ 1 percentage point improvement over vanilla DINO with 28 masked tokens and a comparable result with most ConvNet baselines. It is worth emphasizing that computational cost prevents us from getting higher performance by either pretraining on larger datasets or using larger backbones. Besides, those outperforming baselines (underlined in the table) are achieved by extra mechanisms such as *fully-supervised* pretraining on larger datasets using more powerful backbones [2, 51], or fine-tuned with FG-specific losses [32, 50, 49, 48]. We believe this does not undermine the effectiveness of masked tokens.

Ultra Fine-Grained classifications (UFG). Comparing to the FG, UFG requires more subtle details to distinguish its categories, effectively rendering the available data even smaller. Tab. 7 shows the comparison with multiple ViT and ConvNet baselines on SoybeanLocal and Cotton80 datasets, which only have 600 and 240 fine-tuning samples for each. It is encouraging to see that MT DINO performs

Table 7: CdTL performances for UFG datasets. Same as [21], the input size is set to 384.

Method	backbone	supervised pretraining?	Soybean Local	Cotton80
Nts-Net [32]	ResNet-50	✓	42.67	51.67
ADL [35]	ResNet-50	✓	34.67	43.75
Cutmix [36]	ResNet-50	✓	26.33	45.00
MoCov2 [10]	ResNet-50	✗	32.67	45.00
BYOL [11]	ResNet-50	✗	33.17	52.92
SimCLR [12]	ResNet-50	✗	37.33	51.67
ViT[2]	ViT-B/16	✓	39.33	51.25
BeiT[3]	ViT-B/16	✓	38.67	53.75
TransFG[51]	ViT-B/16	✓	38.67	45.84
DINO[9]	ViT-S/16	✗	<u>41.33</u>	49.58
MT DINO	ViT-S/16	✗	41.17	51.67
MT DINO (28 masked tokens)	ViT-S/16	✗	43.33	53.75

Table 8: CdTL performances for COVID-CT dataset.

Method	backbone	supervised pretraining?	Accuracy (%)
DenseNet [44]	DenseNet-169	✓	84.65
DINO	ViT-S/16	✗	83.25
MT DINO	ViT-S/16	✗	82.76
MT DINO (28 masked tokens)	ViT-S/16	✗	85.22

2.00 and 4.17 percentage points better than DINO with 28 masked tokens on the two datasets respectively, and outperforms most baselines. Especially, MT DINO can improve over ViT baselines [2, 3, 51] that use more powerful backbones and supervised pretraining, demonstrating the usefulness of masked tokens for reducing overfitting. Similar to ImageNet results, more scale tokens help improve the fine-tuning performance.

Medical imagery-based diagnosis. We conduct domain-specific transferability experiments on COVID-CT dataset [22], which contains only 425 samples for fine-tuning. The results are shown in Tab. 8. Similar to the UFG, we achieve a better performance than baselines with increased masked token number than the default case, which provides corroborates to our assumption that higher performance can be achieved with more masked tokens.

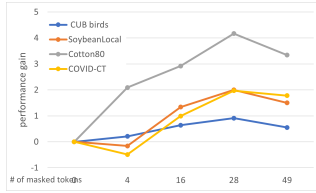


Fig. 3: Performance variations versus masked tokens numbers (sparsity granularity) on CdTL tasks.

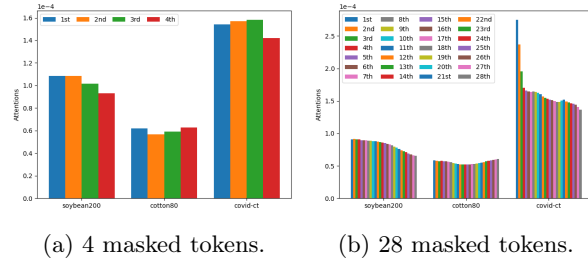


Fig. 4: Average attention values of classification head and each masked tokens across different datasets and token numbers.

The Impact of Sparsity Granularity Inspired by previous observations *w.r.t.* the number of masked tokens, we further study the relationship between the performance and the masked tokens in two more experiments.

Granularity vs. performance. We plot the line chart in Fig. 3 to show the performance gains with different masked token numbers on all four CdTL datasets. Overall, despite a few exceptions, the performance increases as masked token number grows for all datasets, confirming that more sparsity levels can yield better results. However, after a certain point, the performance begins to drop as number continue growing. This is expected as too many masked tokens can carry many overlapped patches, leading to a higher chance of overfitting on smaller sets. Thus, it is not always good to keep a large masked token number.

Attentions for masked tokens. We additionally compute the attention values between the class token \mathbf{h}_{cls} and each masked tokens, and visualize their means across multi-heads and samples for UFG and COVID-CT datasets in Fig. 4. Basically, the patterns of attention are similar when masked token number is small, where attentions are uniformly distributed across each masked token. As token number increases, these patterns act differently for each dataset. For Cotton80, the class token has more dependencies with both the low and high sparsity levels than the mid level, while such dependencies tend to decrease from low to high for SoybeanLocal and COVID-CT as their attention values drop when the sparsity level goes higher. Especially on COVID-CT, tokens with the lower levels have significantly higher attention than others, indicating the class token relies more on the lower sparsity level information.

4 Related works

Vision transformers. Inspired by works in NLP [56, 57], transformers are introduced into computer vision by iGPT [55]. Later, ViT [2] introduced the class token for supervised classification and demonstrated its superiority over traditional

CovNets on large-scale datasets. Since it may yield suboptimal performance due to vulnerability of overfitting, works such as [58–64] are proposed to moderate the effect by strengthening the inductive bias of locality. Some of which try to aggregate spatial information in smaller regions [58], where others focus on removing redundant patches to highlight informative ones [62–64]. Other methods like [59, 61, 60] introduce localities by reshaping tokens back to 2D grids and forward them to a convolution kernel just like ConvNets.

Self-supervised learning. Numerous techniques are introduced to train a visual model in a self-supervised fashion. Some earlier works do this by predicting patch orders [26], image rotations [15], or colorization [27]. Recently, contrastive-based methods have become increasingly popular [28, 29, 10, 52, 12], which augment the input image into multiple views and optimise the model by maximizing the similarity between positive pairs. To prevent from collapsing, [28, 10, 12] propose to increase the number of informative negative pairs by constructing large memory banks or batches, while other works [11, 52] build non-gradient-based targets without explicitly involving negative pairs. Besides, a few methods focus on clustering-based training [53, 16, 54], or using transformers as backbones [8, 9].

5 Conclusions

We tackle the problem of alleviating overfitting for fine-tuned self-supervised ViTs on small, domain-specific datasets. We introduce masked tokens, which mask out redundant regions by aggregating a subset of informative patch tokens. Defined by their sparsity levels, multiple masked tokens encode different sub regions of input images with sizes from small to large. With the proposed patch selection and regularizations, masked tokens can be trained to determine most interesting encoding regions in a data-driven manner. Via integrating masked tokens with self-attentions, we augment ViTs with sparsity and locality biases without altering their core structures. We conduct extensive experiments on various datasets and have found that masked tokens can more effectively capture local secondary contents, which can be complimentary to the standard global attention. Thus with a proper number of masked tokens, an augmented ViT is more amenable to small sets, and retains capabilities of learning rich representations when training sets grow larger.

Acknowledgement. The project was partially funded by KTH Digital Futures and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł. & Polosukhin, I. Attention is all you need. *Advances In Neural Information Processing Systems*. pp. 5998-6008 (2017)

2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. & Others An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv Preprint ArXiv:2010.11929*. (2020)
3. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A. & Jégou, H. Training data-efficient image transformers & distillation through attention. *International Conference On Machine Learning*. pp. 10347-10357 (2021)
4. Bahdanau, D., Cho, K. & Bengio, Y. Neural machine translation by jointly learning to align and translate. *ArXiv Preprint ArXiv:1409.0473*. (2014)
5. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. *ArXiv Preprint ArXiv:1409.1556*. (2014)
6. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 770-778 (2016)
7. Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Densely connected convolutional networks. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 4700-4708 (2017)
8. Atito, S., Awais, M. & Kittler, J. Sit: Self-supervised vision transformer. *ArXiv Preprint ArXiv:2104.03602*. (2021)
9. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P. & Joulin, A. Emerging properties in self-supervised vision transformers. *ArXiv Preprint ArXiv:2104.14294*. (2021)
10. Chen, X., Fan, H., Girshick, R. & He, K. Improved baselines with momentum contrastive learning. *ArXiv Preprint ArXiv:2003.04297*. (2020)
11. Grill, J., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Pires, B., Guo, Z., Azar, M. & Others Bootstrap your own latent: A new approach to self-supervised learning. *ArXiv Preprint ArXiv:2006.07733*. (2020)
12. Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. A simple framework for contrastive learning of visual representations. *International Conference On Machine Learning*. pp. 1597-1607 (2020)
13. Park, T., Efros, A., Zhang, R. & Zhu, J. Contrastive learning for unpaired image-to-image translation. *European Conference On Computer Vision*. pp. 319-345 (2020)
14. Prillo, S. & Eisenschlos, J. SoftSort: A continuous relaxation for the argsort operator. *International Conference On Machine Learning*. pp. 7793-7802 (2020)
15. Gidaris, S., Singh, P. & Komodakis, N. Unsupervised representation learning by predicting image rotations. *ArXiv Preprint ArXiv:1803.07728*. (2018)
16. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P. & Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *ArXiv Preprint ArXiv:2006.09882*. (2020)
17. Krizhevsky, A., Hinton, G. & Others Learning multiple layers of features from tiny images. (Citeseer,2009)
18. Deng, J., Dong, W., Socher, R., Li, L., Li, K. & Fei-Fei, L. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference On Computer Vision And Pattern Recognition*. pp. 248-255 (2009)
19. Coates, A., Ng, A. & Lee, H. An analysis of single-layer networks in unsupervised feature learning. *Proceedings Of The Fourteenth International Conference On Artificial Intelligence And Statistics*. pp. 215-223 (2011)
20. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S. & Perona, P. Caltech-UCSD birds 200. (California Institute of Technology,2010)

21. Yu, X., Zhao, Y., Gao, Y., Yuan, X. & Xiong, S. Benchmark platform for ultra-fine-grained visual categorization beyond human performance. *Proceedings Of The IEEE/CVF International Conference On Computer Vision*. pp. 10285-10295 (2021)
22. Zhao, J., Zhang, Y., He, X. & Xie, P. Covid-ct-dataset: a ct scan dataset about covid-19. *ArXiv Preprint ArXiv:2003.13865*. **490** (2020)
23. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. & Others Language models are few-shot learners. *ArXiv Preprint ArXiv:2005.14165*. (2020)
24. Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N. & Chen, Z. Gshard: Scaling giant models with conditional computation and automatic sharding. *ArXiv Preprint ArXiv:2006.16668*. (2020)
25. Pan, S. & Yang, Q. A survey on transfer learning. *IEEE Transactions On Knowledge And Data Engineering*. **22**, 1345-1359 (2009)
26. Noroozi, M. & Favaro, P. Unsupervised learning of visual representations by solving jigsaw puzzles. *European Conference On Computer Vision*. pp. 69-84 (2016)
27. Zhang, R., Isola, P. & Efros, A. Colorful image colorization. *European Conference On Computer Vision*. pp. 649-666 (2016)
28. Wu, Z., Xiong, Y., Yu, S. & Lin, D. Unsupervised feature learning via non-parametric instance discrimination. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 3733-3742 (2018)
29. Oord, A., Li, Y. & Vinyals, O. Representation learning with contrastive predictive coding. *ArXiv Preprint ArXiv:1807.03748*. (2018)
30. Zheng, H., Fu, J., Mei, T. & Luo, J. Learning multi-attention convolutional neural network for fine-grained image recognition. *Proceedings Of The IEEE International Conference On Computer Vision*. pp. 5209-5217 (2017)
31. Wei, X., Xie, C., Wu, J. & Shen, C. Mask-CNN: Localizing parts and selecting descriptors for fine-grained bird species categorization. *Pattern Recognition*. **76** pp. 704-714 (2018)
32. Nawaz, S., Calefati, A., Caraffini, M., Landro, N. & Gallo, I. Are these birds similar: Learning branched networks for fine-grained representations. *2019 International Conference On Image And Vision Computing New Zealand (IVCNZ)*. pp. 1-5 (2019)
33. Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. & Chao, L. Learning deep transformer models for machine translation. *ArXiv Preprint ArXiv:1906.01787*. (2019)
34. Baevski, A. & Auli, M. Adaptive input representations for neural language modeling. *ArXiv Preprint ArXiv:1809.10853*. (2018)
35. Choe, J. & Shim, H. Attention-based dropout layer for weakly supervised object localization. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 2219-2228 (2019)
36. Yun, S., Han, D., Oh, S., Chun, S., Choe, J. & Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. *Proceedings Of The IEEE/CVF International Conference On Computer Vision*. pp. 6023-6032 (2019)
37. Dosovitskiy, A., Springenberg, J., Riedmiller, M. & Brox, T. Discriminative unsupervised feature learning with convolutional neural networks. *Advances In Neural Information Processing Systems*. **27** pp. 766-774 (2014)
38. Jenni, S. & Favaro, P. Self-supervised feature learning by learning to spot artifacts. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 2733-2742 (2018)

39. Haeusser, P., Plapp, J., Golkov, V., Aljalbout, E. & Cremers, D. Associative deep clustering: Training a classification network with no labels. *German Conference On Pattern Recognition*. pp. 18-32 (2018)
40. Ji, X., Henriques, J. & Vedaldi, A. Invariant information clustering for unsupervised image classification and segmentation. *Proceedings Of The IEEE/CVF International Conference On Computer Vision*. pp. 9865-9874 (2019)
41. Caron, M., Bojanowski, P., Joulin, A. & Douze, M. Deep clustering for unsupervised learning of visual features. *Proceedings Of The European Conference On Computer Vision (ECCV)*. pp. 132-149 (2018)
42. Hjelm, R., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A. & Bengio, Y. Learning deep representations by mutual information estimation and maximization. *ArXiv Preprint ArXiv:1808.06670*. (2018)
43. Patacchiola, M. & Storkey, A. Self-supervised relational reasoning for representation learning. *ArXiv Preprint ArXiv:2006.05849*. (2020)
44. He, X., Yang, X., Zhang, S., Zhao, J., Zhang, Y., Xing, E. & Xie, P. Sample-efficient deep learning for COVID-19 diagnosis based on CT scans. *Medrxiv*. (2020)
45. Wei, X., Zhang, Y., Gong, Y., Zhang, J. & Zheng, N. Grassmann pooling as compact homogeneous bilinear pooling for fine-grained visual classification. *Proceedings Of The European Conference On Computer Vision (ECCV)*. pp. 355-370 (2018)
46. Dubey, A., Gupta, O., Raskar, R. & Naik, N. Maximum-entropy fine-grained classification. *ArXiv Preprint ArXiv:1809.05934*. (2018)
47. Wang, Y., Morariu, V. & Davis, L. Learning a discriminative filter bank within a cnn for fine-grained recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 4148-4157 (2018)
48. Gao, Y., Han, X., Wang, X., Huang, W. & Scott, M. Channel interaction networks for fine-grained image categorization. *Proceedings Of The AAAI Conference On Artificial Intelligence*. **34**, 10818-10825 (2020)
49. Chen, Y., Bai, Y., Zhang, W. & Mei, T. Destruction and construction learning for fine-grained image recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 5157-5166 (2019)
50. Luo, W., Yang, X., Mo, X., Lu, Y., Davis, L., Li, J., Yang, J. & Lim, S. Cross-X learning for fine-grained visual categorization. *Proceedings Of The IEEE/CVF International Conference On Computer Vision*. pp. 8242-8251 (2019)
51. He, J., Chen, J., Liu, S., Kortylewski, A., Yang, C., Bai, Y., Wang, C. & Yuille, A. TransFG: A Transformer Architecture for Fine-grained Recognition. *ArXiv Preprint ArXiv:2103.07976*. (2021)
52. Chen, X. & He, K. Exploring simple siamese representation learning. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 15750-15758 (2021)
53. Asano, Y., Rupprecht, C. & Vedaldi, A. Self-labelling via simultaneous clustering and representation learning. *ArXiv Preprint ArXiv:1911.05371*. (2019)
54. Li, J., Zhou, P., Xiong, C. & Hoi, S. Prototypical contrastive learning of unsupervised representations. *ArXiv Preprint ArXiv:2005.04966*. (2020)
55. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D. & Sutskever, I. Generative pretraining from pixels. *International Conference On Machine Learning*. pp. 1691-1703 (2020)
56. Devlin, J., Chang, M., Lee, K. & Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv Preprint ArXiv:1810.04805*. (2018)
57. Radford, A., Narasimhan, K., Salimans, T. & Sutskever, I. Improving language understanding by generative pre-training. (2018)

58. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z., Tay, F., Feng, J. & Yan, S. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *ArXiv Preprint ArXiv:2101.11986*. (2021)
59. Yuan, K., Guo, S., Liu, Z., Zhou, A., Yu, F. & Wu, W. Incorporating convolution designs into visual transformers. *ArXiv Preprint ArXiv:2103.11816*. (2021)
60. Li, Y., Zhang, K., Cao, J., Timofte, R. & Van Gool, L. Localvit: Bringing locality to vision transformers. *ArXiv Preprint ArXiv:2104.05707*. (2021)
61. Hudson, D. & Zitnick, C. Generative adversarial transformers. *ArXiv Preprint ArXiv:2103.01209*. (2021)
62. Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J. & Hsieh, C. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances In Neural Information Processing Systems*. **34** pp. 13937-13949 (2021)
63. Liang, Y., Ge, C., Tong, Z., Song, Y., Wang, J. & Xie, P. Not all patches are what you need: Expediting vision transformers via token reorganizations. *ArXiv Preprint ArXiv:2202.07800*. (2022)
64. Tang, Y., Han, K., Wang, Y., Xu, C., Guo, J., Xu, C. & Tao, D. Patch slimming for efficient vision transformers. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 12165-12174 (2022)