Not All Images are Worth 16x16 Words: Dynamic Transformers for Efficient Image Recognition

Yulin Wang[1*] **Rui Huang**[1*] **Shiji Song**[1] **Zeyi Huang**[2] **Gao Huang**[1,3†]
[1]Department of Automation, BNRist, Tsinghua University, Beijing, China
[2]Huawei Technologies Ltd., China
[3]Beijing Academy of Artificial Intelligence, Beijing, China
{wang-yl19, hr20}@mails.tsinghua.edu.cn, huangzeyi2@huawei.com.
{shijis, gaohuang}@tsinghua.edu.cn

Abstract

Vision Transformers (ViT) have achieved remarkable success in large-scale image recognition. They split every 2D image into a fixed number of patches, each of which is treated as a token. Generally, representing an image with more tokens would lead to higher prediction accuracy, while it also results in drastically increased computational cost. To achieve a decent trade-off between accuracy and speed, the number of tokens is empirically set to 16x16 or 14x14. In this paper, we argue that every image has its own characteristics, and ideally the token number should be conditioned on each individual input. In fact, we have observed that there exist a considerable number of "easy" images which can be accurately predicted with a mere number of 4x4 tokens, while only a small fraction of "hard" ones need a finer representation. Inspired by this phenomenon, we propose a Dynamic Transformer to automatically configure a proper number of tokens for each input image. This is achieved by cascading multiple Transformers with increasing numbers of tokens, which are sequentially activated in an adaptive fashion at test time, i.e., the inference is terminated once a sufficiently confident prediction is produced. We further design efficient feature reuse and relationship reuse mechanisms across different components of the Dynamic Transformer to reduce redundant computations. Extensive empirical results on ImageNet, CIFAR-10, and CIFAR-100 demonstrate that our method significantly outperforms the competitive baselines in terms of both theoretical computational efficiency and practical inference speed. Code and pre-trained models (based on PyTorch and MindSpore) are available at `https://github.com/blackfeather-wang/Dynamic-Vision-Transformer` and `https://github.com/blackfeather-wang/Dynamic-Vision-Transformer-MindSpore`.

1 Introduction

Transformers, the dominant self-attention-based models in natural language processing (NLP) [10, 40, 3], have been successfully adapted to image recognition problems [11, 55, 38, 17] recently. In particular, vision Transformers achieve state-of-the-art performance on the large scale ImageNet benchmark [9], while exhibit excellent scalability with the further growing dataset size (e.g., on JFT-300M [11]). These models split each image into a fixed number of patches and embed them into 1D tokens as inputs. Typically, representing the data using more tokens contributes to higher prediction accuracy, but leads to intensive computational cost, which grows quadratically with respect to the

*Equal contribution.
†Corresponding author.

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

arXiv:2105.15075v2 [cs.CV] 26 Oct 2021

并非所有图像都值16x16字：动态用于EFICIEN识别的变压器

Yulin Wang[1*] **Rui Huang**[1*] **Shiji Song**[1] **Zeyi Huang**[2] **Gao Huang**[1,3†]
1北京清华大学BNRIST自动化，中国
[2]Huawei Technologies Ltd., China
3中国北京北京艺术学院
{wang-yl19, hr20}@mails.tsinghua.edu.cn, huangzeyi2@huawei.com.
{shijis, gaohuang}@tsinghua.edu.cn

Abstract

视觉变压器（VIV）在大规模的图像识别方面取得了显着的成功。它们将每个2D图像分开到一个固定的补丁数量中，每个图像都被视为令牌。通常，表示具有更多令牌的图像将导致更高的预测精度，而它也导致急剧增加的计算成本。为了在准确性和速度之间实现体面的权衡，令牌的数量是经验组成的16x16或14x14。在本文中，我们认为每个图像都有自己的特征，理想情况下，令牌号码应在每个单独输入上调节。事实上，我们已经观察到存在相当数量的“简单”图像，该图像可以被精确地预测，只有4x4令牌的数量，而只有一小部分“硬”需要一个杂志表示。灵感来自这种现象，我们提出了一种动态变压器，以自动对每个输入图像进行适当数量的令牌。这是通过将多个变压器级联随着数量的令牌级联的多个变压器来实现，这在测试时间以自适应方式依次激活，即，一旦产生了SUF展示的预测，就终止了推断。我们进一步设计了跨动态变压器的不同组件的EFICE
CIET功能重用和关系重用机制，以减少冗余计算。在想象中，CiFar−10和CiFar−100上的广泛经验结果表明，我们的方法在理论上计算EF效率和实际推理速度方面，我们的方法显着优于竞争力的基线。代码和预先训练的型号（基于Pytorch和Mindspore）都可以在https://github.com/blackfeater−wang /动态视觉变压器和https：//github上获得。 COM / Blackfeather−Wang /动态视觉变压器 − Mindspore。

1 Introduction

变压器，自然语言处理中的主要自我关注模型（NLP）[10,40,3]已成功适应最近的图像识别问题[11,55,38,17]。特别是，视觉变压器在大规模的想象基准[9]上实现最先进的性能，虽然具有进一步生长的数据集尺寸（例如，在JFT300M [11]上）具有出色的可扩展性。这些模型将每个图像拆分为FIXED编号，并将其嵌入到1D令牌中作为输入。通常，表示使用更多令牌的数据有助于更高的预测精度，但导致强化计算成本，这相对于

*Equal contribution.
†Corresponding author.

第35次神经信息处理系统（Neurips 2021），澳大利亚悉尼。

token number in self-attention blocks. For a proper trade-off between efficiency and effectiveness, existing works empirically adopt 14x14 or 16x16 tokens [11, 55].

In this paper, we argue that it may not be optimal to treat all samples with the same number of tokens. In fact, there exist considerable variations among different images (e.g., contents, scales of objects, backgrounds, etc.). Therefore, the number of representative tokens should ideally be configured specifically for each input. This issue is critical for the computational efficiency of the models. For example, we train a T2T-ViT-12 [55] with varying token numbers, and report the corresponding accuracy and FLOPs in Table 1. One can observe that adopting the officially recommended 14x14 tokens only correctly recognizes ∼15.9% (76.7% v.s. 60.8%) more test samples compared to that of using 4x4 tokens, while increases the computational cost by 8.5x (1.78G v.s. 0.21G). In other words, computational resources are wasted on applying the unnecessary 14x14 tokens to many "easy" images for which 4x4 tokens are sufficient.

Table 1: Accuracy and computational cost of T2T-ViT-12 with different token numbers on ImageNet.

# of Tokens	14x14	7x7	4x4
Accuracy	76.7%	70.3%	60.8%
FLOPs	1.78G	0.47G	0.21G

Motivated by this observation, we propose a novel *Dynamic Vision Transformer* (DVT) framework, aiming to automatically configure a decent token number conditioned on each image for high computational efficiency. In specific, a cascade of Transformers are trained using increasing number of tokens. At test time, these models are sequentially activated starting with less tokens. Once a prediction with sufficient confidence has been produced, the inference procedure will be terminated immediately. As a consequence, the computation is unevenly allocated among "easy" and "hard" samples by adjusting the token number, yielding a considerable improvement in efficiency. Importantly, we further develop *feature-wise* and *relationship-wise* reuse mechanisms to reduce redundant computations. The former allows the downstream models to be trained on the basis of previously extracted deep features, while the later enables leveraging existing upstream self-attention relationships to learn more accurate attention maps. Illustrative examples of our method are given in Figure 1.

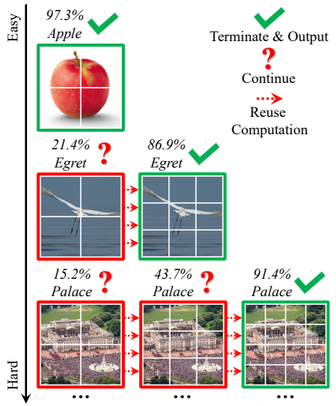


Figure 1: Examples for DVT.

Notably, DVT is designed as a general framework. Most of the state-of-the-art image recognition Transformers, such as ViT [11], DeiT [38], and T2T-ViT [55], can be straightforwardly deployed as its backbones for higher efficiency. Our method is also appealing in its flexibility. The computational cost of DVT is able to be adjusted online by simply adapting the early-termination criterion. This characteristic makes DVT suitable for the cases where the available computational resources fluctuate dynamically or a minimal power consumption is required to achieve a given performance. Both situations are ubiquitous in real-world applications (e.g., searching engines and mobile apps).

The performance of DVT is evaluated on ImageNet [9] and CIFAR [26] with T2T-ViT [55] and DeiT [38]. Experimental results show that DVT significantly improves the efficiency of the backbones. For examples, DVT reduces the computational cost of T2T-ViT by 1.6-3.6x without sacrificing accuracy. The real inference speed on a NVIDIA 2080Ti GPU is consistent with our theoretical results.

2 Related Work

Vision Transformers. Inspired by the success of Transformers on NLP tasks [10, 40, 3, 43], vision Transformers (ViT) have recently been developed for image recognition [11]. Although ViT by itself is not comparable with state-of-the-art convolutional networks (CNN) on the standard ImageNet benchmark, it attains excellent results when pre-trained on the larger JFT-300M dataset. DeiT [38] studies the training strategy of ViT and proposes a knowledge distilling-based approach, surpassing the performance of ResNet [19]. Some following works such as T2T-ViT [55], TNT [17], CaiT [39], DeepViT [62], CPVT [6], LocalViT [28] and CrossViT [5] focus on improving the architecture design of ViT. Another line of research proposes to integrate the inductive bias of CNN into Transformers [49, 8, 54, 16]. There are also attempts to adapt ViT for other vision tasks (e.g., object detection, semantic segmentation, etc.) [30, 44, 12, 20, 57, 60, 14]. The most majority of these concurrent

自我关注块中的令牌号码。对于EF效率和有效性之间的适当权衡，现有工程经验采用14x14或16x16令牌[11,55]。

In this paper, we argue that it may not be optimal to treat all samples with the same number of tokens. In fact, there exist considerable variations among different images (e.g., contents, scales of objects, backgrounds, etc.). Therefore, the number of representative tokens should ideally be configured specifically for each input. This issue is critical for the computational efficiency of the models. For example, we train a T2T-ViT-12 [55] with varying token numbers, and report the corresponding accuracy and FLOPs in Table 1. One can observe that adopting the officially recommended 14x14 tokens only correctly recognizes ∼15.9% (76.7% v.s. 60.8%) more test samples compared to that of using 4x4 tokens, while increases the computational cost by 8.5x (1.78G v.s. 0.21G). In other words, computational resources are wasted on applying the unnecessary 14x14 tokens to many "easy" images for which 4x4 tokens are sufficient.

表1：T2T-VT-12的精度和计算成本，在想象中的不同令牌数字。

# of Tokens	14x14	7x7	4x4
Accuracy	76.7%	70.3%	60.8%
FLOPs	1.78G	0.47G	0.21G

通过这种观察，我们提出了一种新颖的动态视觉变压器（DVT）框架，旨在自动配置在每个图像上调节的体面令牌编号，以实现高计算EF效率。在规范中，使用越来越多的令牌训练级联的变压器。在测试时间时，这些模型按照较少的令牌开始顺序激活。一旦产生了对来自SUFIEN CONINE的预测，就会立即终止推理过程。因此，通过调整令牌数量，计算在“易”和“硬”样本中不均匀地分配，从而产生了相当大的效率。重要的是，我们进一步开发了功能明智的和关系重复使用机制，以减少冗余计算。前者允许在先前提取的深度特征的基础上培训下游型号，而后来可以利用现有的上游自我关注关系来学习更准确的注意图。我们方法的说明性实例在图1中给出。

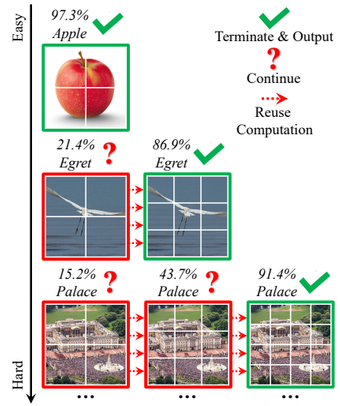


Figure 1: Examples for DVT.

值得注意的是，DVT被设计为一般框架。大多数最先进的图像识别变压器，例如VIT [11]，Deit [38]和T2T-VIT [55]，可以直接地部署为其骨干，以便更高的EF效率。我们的方法在其流动中也有吸引力。DVT的计算成本可以通过简单地调整早期终止标准在线进行调整。这种特性使DVT适用于可用计算资源动态流出的情况或需要最小的功耗来实现给定的性能。两种情况都在现实世界应用中普遍存在（例如，搜索引擎和移动应用程序）。

DVT的性能在Imagenet [9]和CiFar [26]上评估了T2T-VIT [55]和Deit [38]。实验结果表明，DVT显着改善了骨干的EF效率。例如，DVT将T2T-VT-VIT的计算成本降低1.6-3.6倍而没有牺牲精度。NVIDIA 2080TI GPU上的真正推理速度与我们的理论结果一致。

2 Related Work

视觉变形金刚。灵感来自于NLP任务[10,40,3,43]的变压器的成功，最近已经开发了用于图像识别的视觉变压器（VIT）[11]。虽然vit本身与标准想象中心基准上的最先进的卷积网络（CNN）相当，但在较大的JFT-300M数据集上预先培训时，它可以获得出色的结果。Deit [38]研究Vit的培训策略，提出了一种基于知识的蒸馏方法，超越了Reset [19]的表现。一些以下作品如T2T-VIT [55]，TNT [17]，Cait [39]，DeepVit [62]，CPVT [6]，LocalVit [28]和Crossvit [5]注重改善Vit的架构设计。另一项研究建议将CNN的电感偏压集成到变压器[49,8,54,16]中。还试图适应其他视觉任务（例如，物体检测，语义分割等）[30,44,12,20,57,60,14]。这些并发最多的

works represent each image with a fix number of tokens. To the best of our knowledge, we are the first to consider configuring token numbers conditioned on the inputs.

Efficient deep networks. Computational efficiency plays a critical role in real-world scenarios, where the executed computation translates into power consumption, carbon emission or latency. A number of works have been done on reducing the computational cost of CNNs [22, 33, 21, 53, 59, 31, 35]. However, designing efficient vision Transformers is still an under-explored topic. T2T-ViT [55] proposes a light-weighted tokens-to-token module and obtains a competitive accuracy-parameter trade-off compared to MobileNetV2 [33]. LeViT [16] accelerates the inference of Transformer models by involving convolutional layers. Swin Transformer [30] introduces an efficient shifted window-based approach in multi-stage vision Transformers. Compared to these models with fixed computational graphs, the proposed DVT framework improves the efficiency by adaptively changing the architecture of the network on a per-sample basis.

Dynamic models. Designing dynamic architectures is an effective approach for efficient deep learning [18]. In the context of recognition tasks, MSDNet and its variants [23, 52, 27] develop a multi-classifier CNN architecture to perform early exiting for easy samples. Another type of dynamic CNNs skips redundant layers [41, 45, 51] or channels [29] conditioned on the inputs. Besides, the spatial adaptive paradigm [15, 4, 47, 42, 46] has been proposed for efficient image and video recognition. Although these works are related to DVT on the spirit of adaptive computation, they are developed based on CNN, while DVT is tailored for vision Transformers.

Notably, our work may seem similar to the multi-exit networks [23, 52, 27, 47] from the lens of early-exiting. However, DVT differentiates itself from these existing works in several important aspects. For example, a major efficiency gain in both MSDNet [23] and RANet [52] comes from the adaptive depth, namely processing "easier" samples using fewer layers within a single network. In comparison, DVT cascades multiple Transformers to infer several entire networks at test time. Besides, DVT does not leverage the multi-scale architecture or dense connection in MSDNet [23]. Compared with RANet [52], DVT introduces the adaptive token number and the relationship reuse mechanism tailored for ViT. In addition, IMTA [27] studies the training techniques of multi-exit models, which are actually complementary to DVT. Another difference is that all these networks [23, 52, 27, 47] adopt pure convolutional models.

3 Dynamic Vision Transformer

Vision Transformers [11, 17, 38, 55] split each 2D image into 1D tokens, while model their long range interaction with the self-attention mechanism [40]. As aforementioned, to correctly recognize some "hard" images and achieve high accuracy, the number of tokens usually needs to be large, leading to the quadratically grown computational cost. However, "easier" images that make up the bulk of the datasets typically require far fewer tokens and much less costs (as shown in Table 1). Inspired by this observation, we propose a *Dynamic Vision Transformers* (DVT), aiming to improve the computational efficiency of Transformers via adaptively reducing the number of representative tokens for each input.

In specific, we propose to deploy multiple Transformers trained with increasing number of tokens, such that one can sequentially activate them for each test image until obtaining a convincing prediction (e.g., with sufficient confidence). The computation is allocated unevenly across different samples for improving the overall efficiency. It is worth noting that, if all the Transformers are learned separately, the computation performed by upstream models will simply be abandoned once a downstream Transformer is activated, resulting in considerable inefficiency. To alleviate this problem, we introduce the efficient feature and relationship reuse mechanisms.

3.1 Overview

Inference. We start by describing the inference procedure of DVT, which is shown in Figure 2. For each test sample, we first coarsely represent it using a small number of 1D token embeddings. This can be achieved by either straightforwardly flattening the split image patches [11, 17] or leveraging techniques like the tokens-to-token module [55]. We infer a vision Transformer with these few tokens to obtain a quick prediction. This process enjoys high efficiency since the computational cost of Transformers grows quadratically with respect to token number. Then the prediction will be evaluated with certain criterion to determine whether it is reliable enough to be retrieved immediately. In this paper, early-termination is performed when the model is sufficiently confident (details in Section 3.3).

作品代表每个图像使用fi x令牌。据我们所知，我们是第一次考虑在输入上调节的配置令牌编号。

EF
CIEN深度网络。计算EF型效率在现实世界场景中发挥着关键作用，其中执行的计算转化为功耗，碳发射或延迟。已经在降低CNNS的计算成本[22,33,21,53,59,31,35]上进行了许多作品。然而，设计EFIC EIEN VISION变形金刚仍然是一个探索的主题。T2T−VIT
[55]提出了一种光加权令牌到令牌模块，与MobileNetv2 [33]相比，获得了竞争精度参数折衷。Levit [16]通过涉及卷积层加速变压器模型的推动。SWIN变压器[30]在多级视觉变压器中介绍了基于EF的基于窗口的方法。与这些模型相比，通过与计算图表进行了相比，所提出的DVT框架通过自适应地改变网络的基础上的网络架构来改善EF效率。

动态模型。设计动态架构是有效的EFICE深度学习方法[18]。在识别任务的背景下，MSDNet及其变体[23,52,27]开发多种分类的CNN架构，以便为易于样本进行早期退出。另一种类型的动态CNN跳过冗余层[41,45,51]或通道[29]在输入上调节。此外，已经提出了空间自适应范式[15,4,47,42,46]，用于EF展示图像和视频识别。虽然这些作品与自适应计算精神有关DVT，但它们是基于CNN开发的，而DVT则为视觉变压器量身定制。

值得注意的是，我们的工作似乎类似于从早期退出镜头的多出口网络[23,52,27,47]。然而，DVT在几个重要方面的这些现有工作中区分了本现有的工作。例如，MSDNet [23]和RANET
[52]中的主要EF效率增益来自自适应深度，即使用单个网络中的较少层处理“更轻松”的样本。相比之下，DVT级联多个变压器以在测试时间推断出几个整个网络。此外，DVT不利用MSDNet中的多尺度架构或密集连接[23]。与Ranet
[52]相比，DVT介绍了自适应令牌编号和适用于VIT的关系重复使用机制。此外，IMTA
[27]研究了多出口模型的训练技术，其实际上与DVT互补。另一个不同之处在于所有这些网络[23,52,27,47]采用纯卷积模型。

3 Dynamic Vision Transformer

视觉变压器[11,17,38,55]将每个2D图像分成1D令牌，同时模拟它们与自我关注机制的长距离相互作用[40]。如上所述，要正确认识到一些“硬”图像并实现高精度，令牌的数量通常需要大，导致二次种植的计算成本。然而，构成大部分数据集的“更容易”的图像通常需要较少的令牌和更少的成本（如表1所示）。灵感来自这种观察，我们提出了一种动态视觉变压器（DVT），旨在通过自适应地减少每个输入的代表性令牌的数量来改善变压器的计算EF效率。

在规范中，我们建议部署随着越来越多的令牌越来越多的训练的多个变压器，使得可以为每个测试图像顺序地激活它们，直到获得令人信服的预测（例如，使用SUF CIEN
CONEDENDENCE）。计算在不同的样本中不均匀地分配，以改善整体EF效率。值得注意的是，如果分别学习所有变压器，则在激活下游变压器时，通过上游型号执行的计算将简单地放弃，导致相当大的终效率。为了缓解这个问题，我们介绍了EFICE CIET功能和关系重复使用机制。

3.1 Overview

推理。我们首先描述了DVT的推理过程，如图2所示。对于每个测试样本，我们使用少量的1D令牌嵌入式粗略代表它。这可以通过直接呈现分割图像贴片[11,17]或利用像令牌到令牌模块的杠杆技术[55]来实现。我们推断使用这几个令牌的视觉变压器以获得快速预测。该过程享有高效效率，因为变形金刚的计算成本相对于令牌编号逐步发展。然后，将使用某些标准进行评估预测以确定是否可以立即检索它是可靠的。在本文中，当模型SUFIESIELY CON
FIENT（第3.3节中的细节）时，执行早期终止。

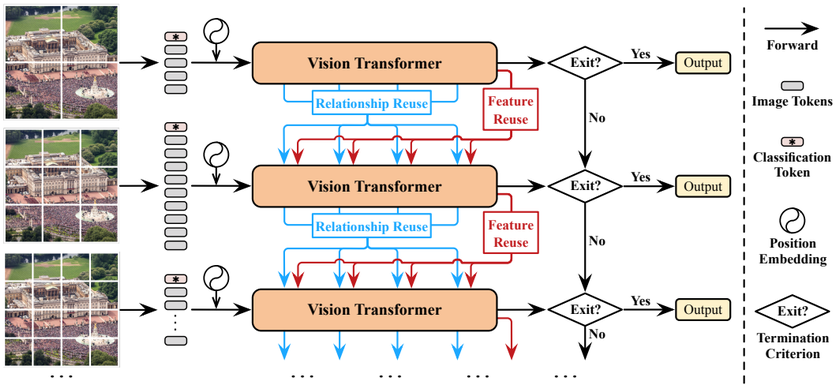


Figure 2: An overview of *Dynamic Vision Transformers* (DVT). Under the objective of configuring proper token numbers conditioned on the inputs, we cascade multiple Transformers with increasing number of tokens. At test time, they are sequentially activated until a convincing prediction (e.g. sufficiently confident) has been obtained or the final model has been inferred. The feature and relationship reuse mechanisms allow reusing computation across different Transformers.

Once the prediction fails to meet the termination criterion, the original input image will be split into more tokens for more accurate but computationally more expensive inference. Note that, here the dimension of each token embedding remains unchanged, while the number of tokens increases, enabling more fine-grained representation. An additional Transformer with the same architecture as the previous one but different parameters will be activated. By design, this stage trades off computation for higher accuracy on some "difficult" test samples. To improve the efficiency, the new model can reuse the previously learned features and relationships, which will be introduced in Section 3.2. Similarly, after obtaining a new prediction, the termination criterion will be applied, and the above procedure will proceed until the sample exits or the final Transformer has been inferred.

Training. For training, we simply train DVT to produce correct predictions at all exits (i.e., each with the corresponding number of tokens). Formally, the optimization objective is

$$\text{minimize} \ \frac{1}{|\mathcal{D}_{\text{train}}|} \sum\nolimits_{(\boldsymbol{x},y)\in\mathcal{D}_{\text{train}}} \left[\sum\nolimits_i L_{\text{CE}}(\boldsymbol{p}_i, y)\right], \tag{1}$$

where $(\boldsymbol{x}, y)$ denote a sample in the training set $\mathcal{D}_{\text{train}}$ and its corresponding label. We adopt the standard cross-entropy loss function $L_{\text{CE}}(\cdot)$, while $\boldsymbol{p}_i$ denotes the softmax prediction probability output by the i^{th} exit. We find that such a simple training objective works well in practice.

Transformer backbone. DVT is proposed as a general and flexible framework. It can be built on top of most existing vision Transformers like ViT [11], DeiT [38] and T2T-ViT [55] to improve their efficiency. The architecture of Transformers simply follows the implementation of these backbones.

3.2 Feature and Relationship Reuse

An important challenge to develop our DVT approach is how to facilitate the *reuse* of computation. That is, once a downstream Transformer with more tokens is inferred, it is obviously inefficient if the computation performed in previous models is abandoned. The upstream models, although being based on smaller number of input tokens, are trained with the same objective, and have extracted valuable information for fulfilling the task. Therefore, we propose two mechanisms to reuse the learned deep features and self-attention relationships. Both of them are able to improve the test accuracy significantly by involving minimal extra computational cost.

Background. For the ease of introduction, we first revisit the basic formulation of vision Transformers. The Transformer encoders consist of alternatively stacked multi-head self-attention (MSA) and multi-layer perceptron (MLP) blocks [40, 11]. The layer normalization (LN) [2] and residual

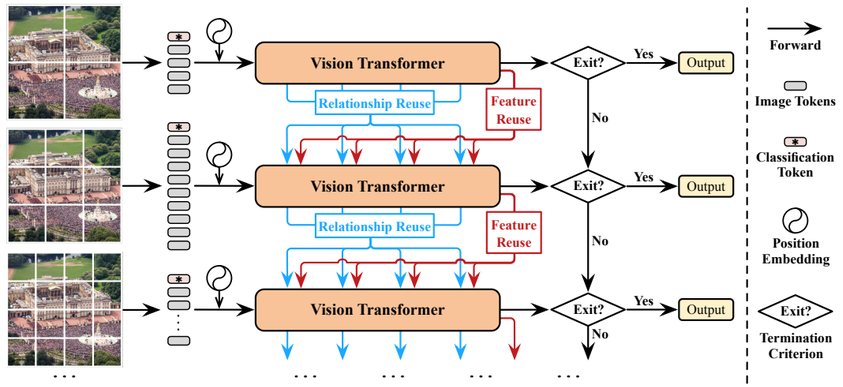


图2：动态视觉变压器（DVT）的概述。在配置对输入上的适当令牌编号的目的下，我们将多个变压器倒入越来越多的令牌。在测试时间时，它们被依次激活直到获得了令人信服的预测（例如，SUF FI CONIES CON
FIENT）或者已经推断出限定模型。该特征和关系重用机制允许在不同的变压器上重用计算。

一旦预测无法满足终止标准，就会将原始输入图像分成更多的令牌以用于更准确但计算更昂贵的推断。请注意，这里每个令牌嵌入的维度保持不变，而令牌的数量增加，启用更有内容的表示。将激活具有与前一个但不同参数相同的架构相同的变压器。通过设计，该阶段对一些“难点”测试样品的更高准确性交易计算。为了提高EF效率，新模型可以重用先前学习的功能和关系，这将在第3.2节中引入。类似地，在获得新的预测之后，将应用终止标准，并且上述过程将继续，直到样品离开或者已经推断出的变压器。

训练。对于培训，我们只是培训DVT以产生正确的退出（即，每个都有相应数量的令牌）的正确预测。正式，优化目标是

$$\text{minimize} \ \frac{1}{|\mathcal{D}_{\text{train}}|} \quad _{(\boldsymbol{x},y)\in\mathcal{D}_{\text{train}}} \left[\quad _i L_{\text{CE}}(\boldsymbol{p}_i, y)\right], \tag{1}$$

其中（x，y）表示训练集DTRAIN及其相应标签中的样本。我们采用标准跨熵损失函数LCE（·），而PI表示ITH出口输出的Softmax预测概率。我们知道这种简单的训练目标在实践中很好地运作。

变压器骨干。DVT被提出为一般和流行的框架。它可以建立在大多数现有视觉变压器的顶部，如vit [11]，deit [38]和T2T−VIT [55]，以改善其EF效率。变形金刚的架构只是遵循这些骨干的实现。

3.2功能和关系重用

开发DVT方法的一个重要挑战是如何促进重用计算。也就是说，一旦推断出具有更多令牌的下游变压器，如果在以前的模型中执行的计算被抛弃，则显然是终点。上游型号虽然基于较少数量的输入令牌，但培训具有相同的目标，并提取了有价值的信息来实现任务。因此，我们提出了两种机制来重用学习的深度特征和自我关注关系。它们都能够通过涉及最小的额外计算成本来提高测试精度显着。

背景。为了便于介绍，我们首先重新审视视觉变压器的基本配方。变压器编码器包括替代地堆叠的多头自我关注（MSA）和多层Perceptron（MLP）块[40,11]。层归一化（LN）[2]和残差

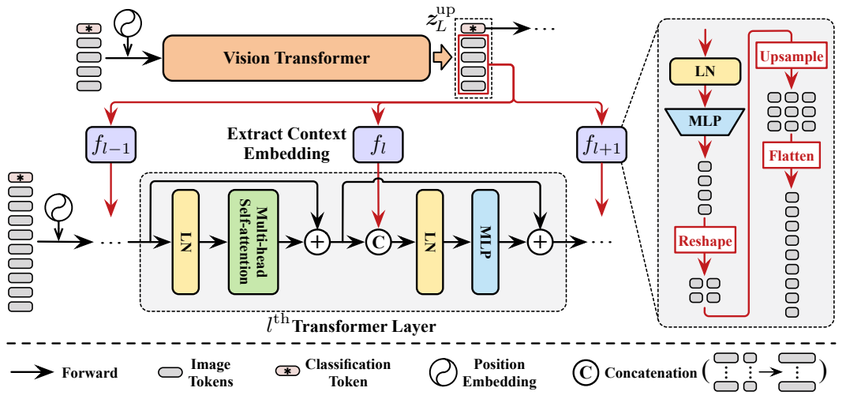


Figure 3: Illustration of the feature reuse mechanism. A layer-wise context embedding is learned based on the final representations output by the upstream model, i.e., $\boldsymbol{z}_L^{\text{up}}$, and integrated into the MLP block of each downstream Transformer layer.

connection [19] are applied before and after each block, respectively. Let $\boldsymbol{z}_l \in \mathbb{R}^{N\times D}$ denote the output of the l^{th} Transformer layer, where N is the number of tokens for each sample, and D is the dimension of each token. Note that $N = HW + 1$, which corresponds to $H \times W$ patches of the original image and a single learnable classification token. Formally, we have

$$\boldsymbol{z}'_l = \text{MSA}(\text{LN}(\boldsymbol{z}_{l-1})) + \boldsymbol{z}_{l-1}, \qquad l \in \{1, \dots, L\}, \tag{2}$$

$$\boldsymbol{z}_l = \text{MLP}(\text{LN}(\boldsymbol{z}'_l)) + \boldsymbol{z}'_l, \qquad l \in \{1, \dots, L\}, \tag{3}$$

where L is the total number of layers in the Transformer. The classification token in $\boldsymbol{z}_L$ will be fed into a LN layer followed by a fully-connected layer for the final prediction. For simplicity, here we omit the details on the position embedding, which is unrelated to our main idea. No modification is performed on it in addition to the configurations of backbones.

Feature reuse. All the Transformers in DVT share the same goal of extracting discriminative representations for accurate recognition. Therefore, it is straightforward that downstream models should be learned on the basis of previously obtained deep features, rather than extracting features from scratch. The former is more efficient since the computation performed in an upstream model contributes to both itself and the successive models. To implement this idea, we propose a feature reuse mechanism (see: Figure 3). In specific, we leverage the image tokens output by the final layer of the upstream Transformer, i.e., $\boldsymbol{z}_L^{\text{up}}$, to learn a layer-wise embedding $\mathbf{E}_l$ for the downstream model:

$$\mathbf{E}_l = f_l(\boldsymbol{z}_L^{\text{up}}) \in \mathbb{R}^{N\times D'}. \tag{4}$$

Herein, f_l: $\mathbb{R}^{N\times D} \to \mathbb{R}^{N\times D'}$ consists of a sequence of operations starting with a LN-MLP ($\mathbb{R}^D \to \mathbb{R}^{D'}$), which introduces nonlinearity and allows more flexible transformations. Then the image tokens are reshaped to the corresponding locations in the original image, upsampled and flattened to match the token number of the downstream model. Typically, we use a small D' for an efficient f_l.

Consequently, the embedding $\mathbf{E}_l$ is injected into the downstream model, providing prior knowledge on recognizing the input image. Formally, we replace Eq. (3) by:

$$\boldsymbol{z}_l = \text{MLP}(\text{LN}(\text{Concat}(\boldsymbol{z}'_l, \mathbf{E}_l))) + \boldsymbol{z}'_l, \qquad l \in \{1, \dots, L\}, \tag{5}$$

where $\mathbf{E}_l$ is concatenated with the intermediate tokens $\boldsymbol{z}'_l$. We simply increase the dimension of LN and the first layer of MLP from D to $D+D'$. Since $\mathbf{E}_l$ is based on the upstream outputs $\boldsymbol{z}_L^{\text{up}}$ that have less tokens than $\boldsymbol{z}'_l$, it actually concludes the context information of the input image for each token in $\boldsymbol{z}'_l$. Therefore, we name $\mathbf{E}_l$ as the *context embedding*. Besides, we do not reuse the classification token and pad zero for it in Eq. (5), which we empirically find beneficial for the performance. Intuitively, Eqs. (4) and (5) allow training the downstream model to flexibly exploit the information within $\boldsymbol{z}_L^{\text{up}}$ on a per-layer basis, under the objective of minimizing the final recognition loss (Eq. (1)). This feature reuse formulation can also be interpreted as implicitly enlarging the depth of the model.

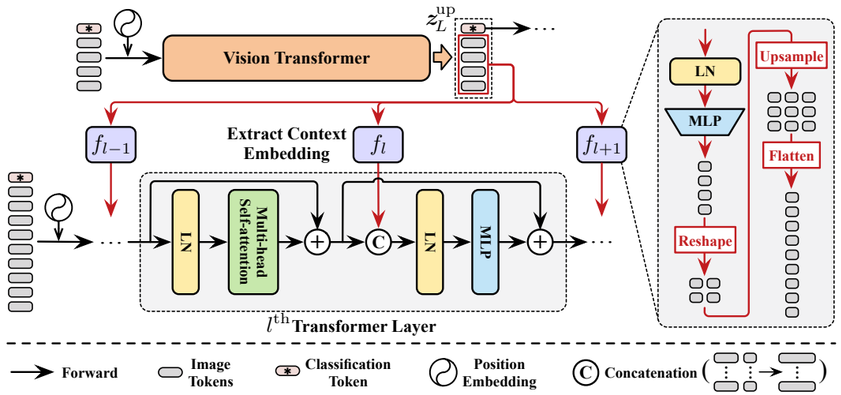


图3：功能重用机制的插图。基于由上游模型输出的FI NAL表示，即，ZUP来学习层面上下文嵌入
1，并融入了

每个下游变压器层的MLP块。

连接[19]分别在每个块之前和之后应用。让ZL ∈ rn × d表示朗格变压器层的输出，其中n是每个样本的令牌数，d是每个令牌的尺寸。请注意，n = HW +
1，其对应于原始图像的H × W曲线，以及单个被动分类令牌。正式，我们有

$$\boldsymbol{z}'_l = \text{MSA}(\text{LN}(\boldsymbol{z}_{l-1})) + \boldsymbol{z}_{l-1}, \qquad l \in \{1, \dots, L\}, \tag{2}$$

$$\boldsymbol{z}_l = \text{MLP}(\text{LN}(\boldsymbol{z}'_l)) + \boldsymbol{z}'_l, \qquad l \quad 1, \dots, L\ , \tag{3}$$

其中L是变压器中的图层总数。ZL中的分类令牌将被送入LN层，然后是用于固定的层，用于FI
NAL预测。为简单起见，我们在这里省略了嵌入位置的细节，这与我们的主要思想无关。除了骨干网的配置外，还没有对其进行修改。

功能重用。DVT中的所有变压器都分享了提取歧视性表示以准确识别的相同目标。因此，它很简单地，应该基于先前获得的深度特征来学习下游模型，而不是从划痕中提取特征。由于在上游模型中执行的计算有助于自身和连续模型，因此前者更加效率。要实现这个想法，我们提出了一个功能重用机制（参见：图3）。在规范中，我们利用上游变压器的Fi NAL层输出的图像令牌，即，zup

L，学习嵌入嵌入EL的下游型号：

up N D'

这里，FL：RN × D→RN × D'由从LN−MLP（RD→RD'）开始的一系列操作，该操作引入非线性并允许更多的流动变换。然后，图像令牌被重新装入原始图像中的相应位置，所以upsampled和fl，以匹配下游模型的令牌编号。通常，我们使用小D'来获得EFICIIT FL。

因此，将嵌入EL注入下游模型，提供关于识别输入图像的先验知识。正式，我们取代了eq。（3）通过：

$$\boldsymbol{z}_l = \text{MLP}(\text{LN}(\text{Concat}(\boldsymbol{z}'_l, \mathbf{E}_l))) + \boldsymbol{z}'_l, \qquad l \quad 1, \dots, L\ , \tag{5}$$

其中EL与中间令牌Z'连接 湖我们只是增加了LN的尺寸
和从d到d + d'的MLP的第一层。由于EL基于上游输出zup

zl，它实际上是总结每个令牌的输入图像的上下文信息
z湖因此，我们将EL命名为上下文嵌入。此外，我们不重复使用分类令牌
并在eq中填充零。（5），我们凭经验的绩效。直观地，eqs。（4）和（5）允许培训下游模型进行zup

l在每层的基础上，在最大限度地减少最小化的识别损失（方程式（1））。该特征重用配方也可以解释为隐式放大模型的深度。

Relationship reuse. A prominent advantage of vision Transformers is that their self-attention blocks enable integrating information across the entire image, which effectively models the long-range dependencies in the data. Typically, the models need to learn a group of attention maps at each layer to describe the relationships among tokens. Apart from the deep features mentioned above, the downstream models also have access to the self-attention maps produced in previous models. We argue that these learned relationships are also capable of being reused to facilitate the learning of downstream Transformers.

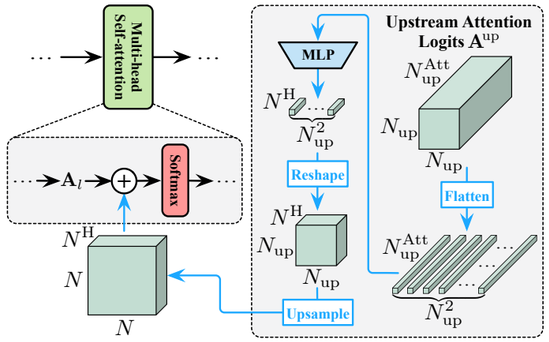


Figure 4: Illustration of the relationship reuse mechanism. We leverage the learned self-attention relationships from all upstream layers and attention heads, i.e., $\mathbf{A}^{\mathrm{up}}$, to refine the downstream attention maps. The addition operation of logits is adopted. Note that N^{H} denotes the head number for multi-head self-attention.

Given the input representation $\boldsymbol{z}_l$, the self-attention is performed as follows. First, the query, key and value matrices $\mathbf{Q}_l$, $\mathbf{K}_l$ and $\mathbf{V}_l$ are computed via linear projections:

$$\mathbf{Q}_l = \boldsymbol{z}_l \mathbf{W}_l^{\mathrm{Q}}, \quad \mathbf{K}_l = \boldsymbol{z}_l \mathbf{W}_l^{\mathrm{K}}, \quad \mathbf{V}_l = \boldsymbol{z}_l \mathbf{W}_l^{\mathrm{V}}, \tag{6}$$

where $\mathbf{W}_l^{\mathrm{Q}}$, $\mathbf{W}_l^{\mathrm{K}}$ and $\mathbf{W}_l^{\mathrm{V}}$ are weight matrices. Then the attention map is calculated by a scaled dot-product operation with softmax to aggregate the values of all tokens, namely

$$\mathrm{Attention}(\boldsymbol{z}_l) = \mathrm{Softmax}(\mathbf{A}_l)\mathbf{V}_l, \quad \mathbf{A}_l = \mathbf{Q}_l \mathbf{K}_l^{\top} / \sqrt{d}. \tag{7}$$

Here d is the hidden dimension of $\mathbf{Q}$ or $\mathbf{K}$, and $\mathbf{A}_l \in \mathbb{R}^{N \times N}$ denotes the logits of the attention map. Note that we omit the details on the multi-head attention mechanism for clarity, where $\mathbf{A}_l$ may include multiple attention maps. Such a simplification does not affect the description of our method.

For relationship reuse, we first concatenate the attention logits produced by all layers of the upstream model (i.e., $\mathbf{A}_l^{\mathrm{up}}, l \in \{1, \ldots, L\}$):

$$\mathbf{A}^{\mathrm{up}} = \mathrm{Concat}(\mathbf{A}_1^{\mathrm{up}}, \mathbf{A}_2^{\mathrm{up}}, \ldots, \mathbf{A}_L^{\mathrm{up}}) \in \mathbb{R}^{N_{\mathrm{up}} \times N_{\mathrm{up}} \times N_{\mathrm{up}}^{\mathrm{Att}}}, \tag{8}$$

where N_{up} and $N_{\mathrm{up}}^{\mathrm{Att}}$ denote the number of tokens and all attention maps in the upstream model, respectively. Typically, we have $N_{\mathrm{up}}^{\mathrm{Att}} = N^{\mathrm{H}} L$, where N^{H} is the number of heads for the multi-head attention and L is the number of layers. Then the downstream Transformer learns attention maps by leveraging both its own tokens and $\mathbf{A}^{\mathrm{up}}$ simultaneously. Formally, we replace Eq. (7) by

$$\mathrm{Attention}(\boldsymbol{z}_l) = \mathrm{Softmax}(\mathbf{A}_l + r_l(\mathbf{A}^{\mathrm{up}}))\mathbf{V}_l, \quad \mathbf{A}_l = \mathbf{Q}_l \mathbf{K}_l^{\top} / \sqrt{d}, \tag{9}$$

where $r_l(\cdot)$ is a transformation network that integrates the information provided by $\mathbf{A}^{\mathrm{up}}$ to refine the downstream attention logits $\mathbf{A}_l$. The architecture of $r_l(\cdot)$ is presented in Figure 4, which includes a MLP for nonlinearity followed by an upsample operation to match the size of attention maps. For multi-head attention, the output dimension of the MLP will be set to the number of heads.

Notably, Eq. (9) is a simple but flexible formulation. For one thing, each self-attention block in the downstream model has access to all the upstream attention heads in both shallow and deep layers, and hence can be trained to leverage multi-level relationship information on its own basis. For another, as the newly generated attention maps and the reused relationships are combined in logits, their relative importance can be automatically learned by adjusting the magnitude of logits. It is also worth noting that the regular upsample operation cannot be directly applied in $r_l(\cdot)$. To illustrate this issue, we take upsampling a $HW \times HW$ ($H = W = 2$) attention map to $H'W' \times H'W'$ ($H' = W' = 3$) for example in Figure 5. Since each of its rows and columns corresponds to $H \times W$ image tokens, we reshape the rows or columns back to $H \times W$, scale them to $H' \times W'$, and then flatten them to $H'W'$ vectors.

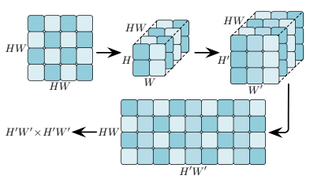
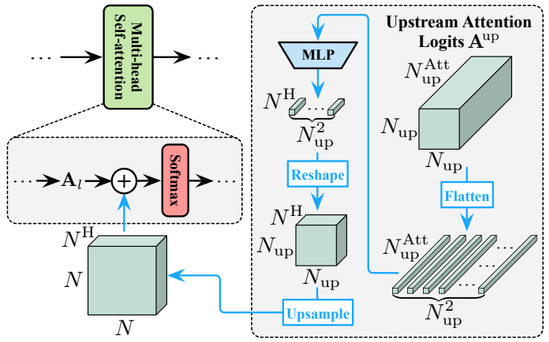


Figure 5: An example for the upsample operation in $r_l(\cdot)$.

A promi-
视觉变形金刚的很多优点是它们的自我关注块使得整合整个图像的信息，这有效地模拟了数据中的远程依赖性。通常，模型需要在每个层中学习一组注意力映射以描述令牌之间的关系。除了上述深度特征外，下游型号还可以访问以前型号生产的自我关注图。我们争吵



这些学众的关系也能够重复使用，以促进下游变形金刚的学习。

图4：关系重用机制的插图。我们利用来自所有上游层和关注头的学习的自我关注关系，即，AUP，重新上游注意力地图。采用Logits的添加操作。请注意，n h表示多头自我关注的头号。

给定输入表示ZL，如下进行自我关注。首先，通过线性投影计算查询，键和值矩阵QL，KL和VL：

$$\mathbf{Q}_l = \boldsymbol{z}_l \mathbf{W}_l^{\mathrm{Q}}, \quad \mathbf{K}_l = \boldsymbol{z}_l \mathbf{W}_l^{\mathrm{K}}, \quad \mathbf{V}_l = \boldsymbol{z}_l \mathbf{W}_l^{\mathrm{V}}, \tag{6}$$

where $\mathbf{W}^{\mathrm{Q}}$, $\mathbf{W}^{\mathrm{K}}$ and $\mathbf{W}$l是重量矩阵。然后注意地图由缩放计算
Dot−Product操作使用Softmax汇集所有令牌的值，即

$$(\boldsymbol{z}) = \quad (\mathbf{A})\mathbf{V}, \quad \mathbf{A} = \mathbf{Q}\,\mathbf{K}^{\top} / \quad \bar{d}.$$

这里D是Q或K的隐藏尺寸，Al∈rn×n表示注意图的注册。请注意，我们省略了用于清楚起见的多主题注意机制的细节，其中AL可能包括多个注意图。这种简单阳离子不会影响我们的方法的描述。

对于关系重用，我们首先串联由上游模型的所有层产生的注意页面（即，AUP

$$\mathbf{A}^{\mathrm{up}} = \mathrm{Concat}(\mathbf{A}_1^{\mathrm{up}}, \mathbf{A}_2^{\mathrm{up}}, \ldots, \mathbf{A}_L^{\mathrm{up}}) \quad \mathbb{R}^{N_{\mathrm{up}} \times N_{\mathrm{up}} \times N_{\mathrm{up}}^{\mathrm{Att}}}, \tag{8}$$

where N_{up} and N向上表示上游模型中的令牌和所有注意图的数量，
分别。通常，我们有n att　　UP = n HL，其中n h是多头的头部数
注意，L是层数。然后，下游变换器通过同时利用其自己的令牌和AUP来了解注意力地图。正式，我们取代了eq。（7）

$$(\boldsymbol{z}) = \quad (\mathbf{A} + r\,(\mathbf{A}^{\mathrm{up}}))\mathbf{V}, \quad \mathbf{A} = \mathbf{Q}\,\mathbf{K}^{\top} / \quad \bar{d},$$

其中R1（·）是一个转换网络，它集成了AUP提供的信息以重新编辑下游注意Logits
al。RL（·）的架构如图4所示，该架构包括用于非线性的MLP，然后是upsample操作以匹配注意力映射的大小。对于多主头注意，MLP的输出维度将设置为头部。

Notably, Eq. (9) is a simple but flexible formulation. For one thing, each self-attention block in the downstream model has access to all the upstream attention heads in both shallow and deep layers, and hence can be trained to leverage multi-level relationship information on its own basis. For another, as the newly generated attention maps and the reused relationships are combined in logits, their relative importance can be automatically learned by adjusting the magnitude of logits. It is also worth noting that the regular upsample operation cannot be directly applied in $r_l(\cdot)$. To illustrate this issue, we take upsampling a $HW \times HW$ ($H = W = 2$) attention map to $H'W' \times H\ W \quad H = W = 3$ in Figure 5. Since each of its rows and columns corresponds to $H \times W$ image tokens, we reshape the rows or columns back to $H \times W$, scale them to $H' \times W'$, and then flatten them to $H'W'$ vectors.

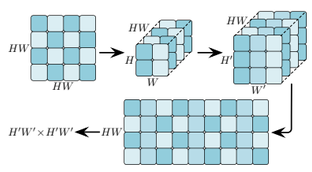


图5：RL（·）中的上置操作的示例。

3.3 Adaptive Inference

As aforementioned, the proposed DVT framework progressively increases the number of tokens for each test sample and performs early-termination, such that "easy" and "hard" images can be processed using varying tokens with uneven computational cost, improving the overall efficiency. Specifically, at the i^{th} exit that produces the softmax prediction $\boldsymbol{p}_i$, the largest entry of $\boldsymbol{p}_i$, i.e., $\max_j p_{ij}$ (defined as confidence [23, 52, 47]), is compared with a threshold η_i. If $\max_j p_{ij} \geq \eta_i$, the inference will stop by adopting $\boldsymbol{p}_i$ as the output. Otherwise, the image will be represented using more tokens to activate the downstream Transformer. We always adopt a zero-threshold for the final Transformer.

The values of $\{\eta_1, \eta_2, \ldots\}$ are solved on the validation set. We assume a *budgeted batch classification* [23] setting, where DVT needs to recognize a set of samples $\mathcal{D}_{\text{val}}$ within a given computational budget $B > 0$. Let $\text{Acc}(\mathcal{D}_{\text{val}}, \{\eta_1, \eta_2, \ldots\})$ and $\text{FLOPs}(\mathcal{D}_{\text{val}}, \{\eta_1, \eta_2, \ldots\})$ denote the accuracy and computational cost on $\mathcal{D}_{\text{val}}$ when using the thresholds $\{\eta_1, \eta_2, \ldots\}$. The optimal thresholds can be obtained by solving the following optimization problem:

$$\underset{\eta_1, \eta_2, \ldots}{\text{maximize}} \quad \text{Acc}(\mathcal{D}_{\text{val}}, \{\eta_1, \eta_2, \ldots\}) \qquad \text{subject to} \quad \text{FLOPs}(\mathcal{D}_{\text{val}}, \{\eta_1, \eta_2, \ldots\}) \leq B. \tag{10}$$

Due to the non-differentiability, we solve this problem with the genetic algorithm [48] in this paper.

4 Experiments

In this section, we empirically validate the proposed DVT on ImageNet [9] and CIFAR-10/100 [26]. Ablation studies and visualization are presented on ImageNet to give a deeper understanding of our method. Code and pre-trained models based on PyTorch are available at `https://github.com/blackfeather-wang/Dynamic-Vision-Transformer`. We also provide the implementation using the MindSpore framework and the models trained on a cluster of Ascend AI processors at `https://github.com/blackfeather-wang/Dynamic-Vision-Transformer-MindSpore`.

Datasets. (1) ImageNet is a 1,000-class dataset from ILSVRC2012 [9], containing 1.2 million images for training and 50,000 images for validation. (2) CIFAR-10/100 datasets [26] contain 32x32 colored images in 10/100 classes. Both of them consist of 50,000 images for training and 10,000 images for testing. For all the three datasets, we adopt the same data pre-processing and data augmentation policy as [19, 24, 23]. In addition, we solve the confidence thresholds stated in Section 3.3 on the training set, which we find achieves similar performance to adopting cross-validation.

Backbones. Our experiments are based on several state-of-the-art vision Transformers, namely T2T-ViT-12 [55], T2T-ViT-14 [55], and DeiT-small (w/o distillation) [38]. Unless otherwise specified, we deploy DVT with three exits, corresponding to representing the images as 7x7, 10x10 and 14x14 tokens[3]. For fair comparisons, our implementation exploits the official code of the backbones, and adopts exactly the same training hyper-parameters. More training details can be found in Appendix A. The number of FLOPs is calculated using the *fvcore* toolkit provided by Facebook AI Research, which is also used in Detectron2 [50], PySlowFast [13], and ClassyVision [1].

Implementation details. For feature reuse, the hidden size and output size of the MLP in $f_l(\cdot)$ are set to 128 and 48. In relationship reuse, for implementation efficiency, we share the same hidden state across the MLPs of all $r_l(\cdot)$, such that $r_l(\mathbf{A}^{\text{up}}), l \in \{1, \ldots, L\}$ can be obtained at one time in concatenation by implementing a single large MLP, whose hidden size and output size are $3N^{\text{H}}L$ and $N^{\text{H}}L$. Note that N^{H} is the head number of multi-head attention and L is the layer number.

4.1 Main Results

Results on ImageNet are shown in Figures 6 and 7, where T2T-ViT [55] and DeiT [38] are implemented as backbones respectively. As stated in Section 3.3, we vary the average computational budget, solve the confidence thresholds, and evaluate the corresponding validation accuracy. The performance of DVT is plotted in gray curves, with the best accuracy under each budget plotted in black curves. We also compare our method with several highly competitive baselines, i.e., TNT [17], LocalViT [28], CrossViT [5], PVT [44], ViT [11] and ResNet [19]. It can be observed that DVT consistently reduces the computational cost of the backbones. For example, DVT achieves the 82.3% accuracy with 3.6x less FLOPs compared with the vanilla T2T-ViT. When the budget ranges among 0.5-2 GFLOPs, DVT has ∼1.7-1.9x less computation than T2T-ViT with the same performance. Notably, our method can flexibly attain all the points on each curve by simply adjusting the values of confidence thresholds with a single DVT.

[3] Although 4x4 tokens are also used as an example in Section 1, we find starting with 7x7 is more efficient.

3.3 Adaptive Inference

如上所述，所提出的DVT框架逐渐增加每个测试样本的令牌数量并执行早期终止，使得可以使用不同的令牌具有不均匀的计算成本，提高整体EF效率来处理“容易”和“硬”图像。在产生Softmax预测PI的ITH出口处，与阈值ηi相比，在产生Softmax预测PI的ITH Exit中，PI的最大条目，即MAXJ PIJ（定义为COFEDENCE [23,52,47]）。如果MAXJPIJ≥0，则推断将通过采用PI作为输出来停止。否则，将使用更多令牌来表示图像以激活下游变压器。我们始终为FI NAL变压器采用零阈值。

{η1，η2的值。。。}在验证集上得到解决。我们假设预算批量分类[23]设置，其中DVT需要在给定的计算预算B> 0中识别一组样本DVAL B> 0.让ACC（DVAL，{η1，η2，.....}）和拖鞋（DVAL，{η1，η2，...。。}）在使用阈值时，表示DVAL上的准确性和计算成本{η1，η2。。。}。可以通过解决以下优化问题来获得最佳阈值：

$$\underset{\eta_1, \eta_2, \ldots}{\text{maximize}} \quad \text{Acc}(\mathcal{D}_{\text{val}}, \{\eta_1, \eta_2, \ldots\}) \qquad \text{subject to} \quad \text{FLOPs}(\mathcal{D}_{\text{val}}, \{\eta_1, \eta_2, \ldots\}) \leq B. \tag{10}$$

由于非差异性，我们用本文的遗传算法解决了这个问题[48]。

4 Experiments

在本节中，我们经验验证了Imagenet [9]和CiFar−10/100上所提出的DVT [26]。在想象中展示了消融研究和可视化，以更深入地了解我们的方法。基于Pytorch的代码和预训练模型可在HTTPS：// Github上获得。COM / Blackfeather−Wang /动态视觉变压器。我们还通过MindSpore框架和在Https://github.com/blackfeater−transformer−mindspore上使用Mindspore框架和培训的模型提供了培训的模型。

数据集。（1）Imagenet是ILSVRC2012 [9]的1,000类数据集，其中包含120万图像进行培训和50,000个图像进行验证。（2）CIFAR−10/100数据集[26]包含32x32彩色图像10/100级。其中两者都包括50,000张图像进行培训和10,000张图像进行测试。对于所有三个数据集，我们采用与[19,24,23]相同的数据预处理和数据增强策略。此外，我们解决了第3.3节中所述的Con Fi Dence阈值，我们认为我们可以实现类似的交叉验证的性能。

骨干。我们的实验基于几种最先进的视觉变压器，即T2T−VT−12 [55]，T2T−VT−14 [55]和Deit−might（W / O蒸馏）[38]。除非另有规定，否则我们将DVT部署有三个出口，对应于将图像表示为7x7，10x10和14x14令牌3。为了进行公平比较，我们的实现利用了底部的Cial代码，并采用完全相同的培训超参数。可以在附录A中找到更多培训细节。使用Facebook AI Research提供的FVCore工具包计算拖鞋的数量，该研究也用于Detectron2 [50]，Pyslowfast [13]和ClassyVision [1]。

实施细节。对于特征重用，FL（·）中MLP的隐藏大小和输出大小被设置为128和48.在关系重用中，为了实现EF效率，我们在所有RL（·）的MLP中共享相同的隐藏状态，例如该RL（AUP），L∈{1，。。。通过实现单个大MLP可以在串联一次，其隐藏尺寸和输出大小是3N HL和N HL的一次。请注意，N H是多主题的头号，L是图层编号。

4.1 Main Results

图6和图7中示出了图6和7的结果，其中T2T−VIT [55]和Deit [38]分别作为骨干。如第3.3节所述，我们改变了平均计算预算，解决了确认阈值，并评估了相应的验证准确性。
DVT的性能绘制在灰色曲线中，在每个预算下具有最佳精度，在黑色曲线上绘制。我们还将我们的方法与几个高竞争力的基线进行了比较，即TNT [17]，LocalVit [28]，Crossvit [5]，PVT [44]，Vit [11]和Reset [19]。可以观察到DVT一致地降低了骨干的计算成本。例如，与香草T2T−VT−VIT相比，DVT通过3.6倍的精度实现了82.3％的精度。当预算范围在0.5−2 GFLOP之间的范围内时，DVT的计算比T2T−VIT具有相同的性能。值得注意的是，我们的方法可以通过简单地调整用单个DVT的配置值的值来实现每个曲线上的所有点。

3虽然4x4令牌也用作第1节中的示例，但我们从7x7开始的我们是更加效率的。

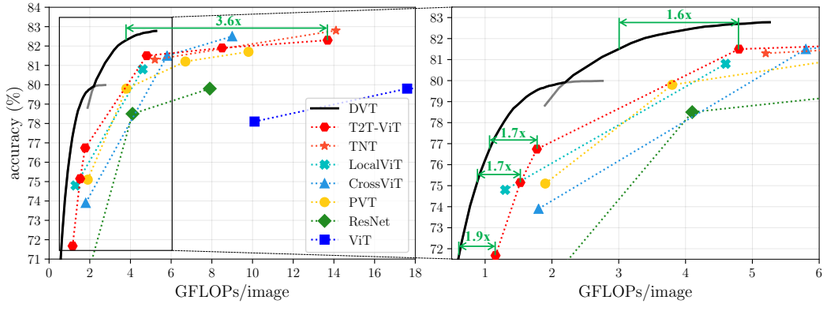


Figure 6: Top-1 accuracy v.s. GFLOPs on ImageNet. DVT is implemented on top of T2T-ViT-12/14.

Table 2: The practical speed of DVT.

Models	ImageNet (NVIDIA 2080Ti, bs=128) Top-1 Acc.	Throughput
T2T-ViT-7	71.68%	1574 img/s
DVT	**78.48%** (↑6.80%)	1574 img/s
T2T-ViT-10	75.15%	1286 img/s
DVT	**79.74%** (↑4.59%)	1286 img/s
T2T-ViT-12	76.74%	1121 img/s
DVT	**80.43%** (↑3.69%)	1128 img/s
T2T-ViT-14	81.50%	619 img/s
DVT	81.50%	**877 img/s** (↑1.42x)
T2T-ViT-19	81.93%	382 img/s
DVT	81.93%	**666 img/s** (↑1.74x)

Table 3: Performance of DVT on CIFAR-10/100.

Models	CIFAR-10 Top-1 Acc.	CIFAR-10 GFLOPs	CIFAR-100 Top-1 Acc.	CIFAR-100 GFLOPs
T2T-ViT-10	97.21%	1.53	85.44%	1.53
DVT	97.21%	**0.50** (↓3.1x)	85.45%	**0.54** (↓2.8x)
T2T-ViT-12	97.45%	1.78	86.23%	1.78
DVT	97.46%	**0.52** (↓3.4x)	86.26%	**0.61** (↓2.9x)
T2T-ViT-14	98.19%	4.80	89.10%	4.80
DVT	98.19%	**0.77** (↓6.2x)	89.11%	**1.62** (↓3.0x)
T2T-ViT-19	98.43%	8.50	89.37%	8.50
DVT	98.43%	**1.44** (↓5.9x)	89.38%	**1.74** (↓4.9x)
T2T-ViT-24	98.53%	13.69	89.62%	13.69
DVT	98.53%	**1.49** (↓9.2x)	89.63%	**1.86** (↓7.4x)

Practical efficiency of DVT. We test the actual speed of DVT on a NVIDIA 2080Ti GPU under a batch inference setting, where a mini-batch of data is fed into the model at a time. After inferring each Transformer, the samples that meet the early-termination criterion will exit, with the remaining images fed into the downstream Transformer. The results are presented in Table 2. Here we adopt a two-exit DVT based on T2T-ViT-12 using 7x7 and 14x14 tokens, which we find more efficient in practice. All other implementation details remain unchanged. One can observe that DVT improves the accuracy of small models (T2T-ViT-7/10/12) by 3.7-6.8% with the same inference speed, while accelerates the inference of the large T2T-ViT-14/19 models by 1.4-1.7x without sacrificing performance.

Results on CIFAR are presented in Table 3. Following the common practice [11, 55, 17, 38], we resize the CIFAR images to 224x224, and fine-tune the T2T-ViT and DVT models in Figure 6. The official code and training configurations provided by [55] are utilized. We report the computational cost of DVT when it achieves the competitive performance with baselines. Our proposed method is shown to consume ∼3-9x less computation compared with T2T-ViT.

Table 4: Comparisons between DVT and three state-of-the-art methods for improving the computational efficiency of vision Transformers, i.e., Data distillation [38], DynamicViT [32] and Patch slimming [36]. Both DVT and the baselines are implemented on top of DeiT-small. Since the computational cost of DVT can be adjusted online, we match the FLOPs or the accuracy of DVT with the baselines, respectively, to see the difference.

	DeiT-small	Data distillation	DVT		DynamicViT	DVT		Patch slimming	DVT	
Top-1 Acc.	79.80%	81.20%	81.20%	**81.67%**	79.30%	79.30%	**80.40%**	79.40%	79.40%	**79.81%**
GFLOPs/image	4.61	4.63	**3.61**	4.63	2.90	**2.37**	2.90	2.60	**2.41**	2.60

Comparisons with SOTA baselines. In Table 4, we compare DVT with several recently proposed approaches for facilitating efficient vision Transformers on ImageNet. Both knowledge distillation (KD) [38] and patch pruning based methods [32, 36] are considered. One can observe that DVT outperforms the baselines with similar FLOPs, or has significantly lower computational cost with the same accuracy. Besides, one can expect that DVT is compatible with these techniques, i.e., DVT can also be trained with KD to further boost the accuracy or leverage patch pruning to reduce the FLOPs.

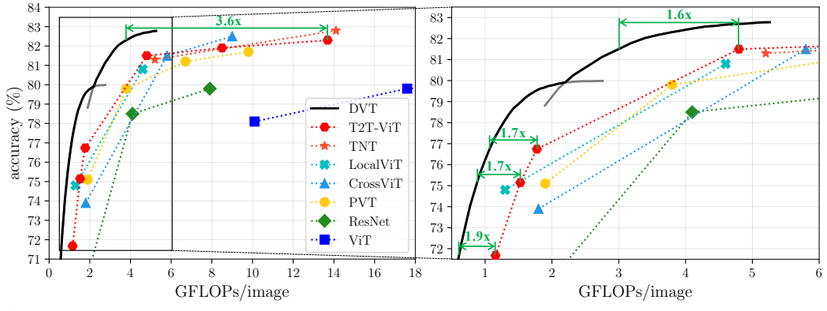


图6：前1个精度V.S.想象成的gflops。DVT在T2T−VT−12/14顶部实现。

表2：DVT的实用速度。

Models	ImageNet (NVIDIA 2080Ti, bs=128) Top-1 Acc.	Throughput
T2T-ViT-7	71.68%	1574 img/s
DVT	**78.48%** (↑6.80%)	1574 img/s
T2T-ViT-10	75.15%	1286 img/s
DVT	**79.74%** (↑4.59%)	1286 img/s
T2T-ViT-12	76.74%	1121 img/s
DVT	**80.43%** (↑3.69%)	1128 img/s
T2T-ViT-14	81.50%	619 img/s
DVT	81.50%	**877 img/s** (↑1.42x)
T2T-ViT-19	81.93%	382 img/s
DVT	81.93%	**666 img/s** (↑1.74x)

表3：CIFAR−10/100上的DVT的性能。

Models	CIFAR-10 Top-1 Acc.	CIFAR-10 GFLOPs	CIFAR-100 Top-1 Acc.	CIFAR-100 GFLOPs
T2T-ViT-10	97.21%	1.53	85.44%	1.53
DVT	97.21%	**0.50** (↓3.1x)	85.45%	**0.54** (↓2.8x)
T2T-ViT-12	97.45%	1.78	86.23%	1.78
DVT	97.46%	**0.52** (↓3.4x)	86.26%	**0.61** (↓2.9x)
T2T-ViT-14	98.19%	4.80	89.10%	4.80
DVT	98.19%	**0.77** (↓6.2x)	89.11%	**1.62** (↓3.0x)
T2T-ViT-19	98.43%	8.50	89.37%	8.50
DVT	98.43%	**1.44** (↓5.9x)	89.38%	**1.74** (↓4.9x)
T2T-ViT-24	98.53%	13.69	89.62%	13.69
DVT	98.53%	**1.49** (↓9.2x)	89.63%	**1.86** (↓7.4x)

DVT的实用效率。在批处理推断设置下，我们在NVIDIA 2080TI
GPU上测试DVT的实际速度，其中迷你批次数据一次进入模型。在推断每个变压器之后，符合早期标准的样本将离开，其余图像馈入下游变压器。结果如表2所示。这里，我们采用了一种基于T2T−VIT−12的双出口DVT，使用7x7和14x14令牌，我们在实践中更加效用。所有其他实施细节都保持不变。人们可以观察到DVT以相同的推断速度提高小型模型（T2T−VIT−7 / 10/12）的精度3.7−6.8%，同时加速大T2T−VT−14/19型号的推理1.4−1.7x没有牺牲性能。

Cifar的结果列于表3中。在常见的做法[11,55,17,38]之后，我们将CiFar图像调整为224x224，以及图6中的T2T−VIT和DVT型号。利用[55]提供的培训确认刺激。我们在使用基线实现竞争性能时，我们报告了DVT的计算成本。与T2T−VIT相比，我们所提出的方法显示少量计算较少。

表4：DVT和三种最先进的方法的比较，用于改善视觉变压器的计算EF效率，即数据蒸馏[38]，DynamicVit
[32]和跳线[36]。DVT和基线都在Deit−might的顶部实施。由于DVT的计算成本可以在线调整，因此我们将拖鞋或DVT的准确性分别与基线相匹配，以查看差异。

	DeiT-small	Data distillation	DVT		DynamicViT	DVT		Patch slimming	DVT	
Top-1 Acc.	79.80%	81.20%	81.20%	**81.67%**	79.30%	79.30%	**80.40%**	79.40%	79.40%	**79.81%**
GFLOPs/image	4.61	4.63	**3.61**	4.63	2.90	**2.37**	2.90	2.60	**2.41**	2.60

与SOTA基线的比较。在表4中，我们将DVT与几种最近提出的方法进行比较，以便于在想象中促进EF牌视觉变压器。考虑知识蒸馏（KD）[38]和贴剂预剪的方法[32,36]。人们可以观察到DVT优于具有相似拖鞋的基线，或者具有相同的准确度的显着降低计算成本。此外，可以预期DVT与这些技术兼容，即，DVT也可以用KD培训，以进一步提高精度或利用补丁修剪以减少拖鞋。

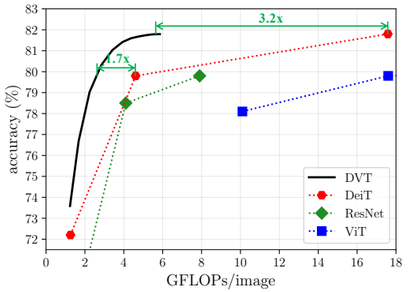


Figure 7: Performance of DeiT-based DVT on ImageNet. DeiT-small is used as the backbone.

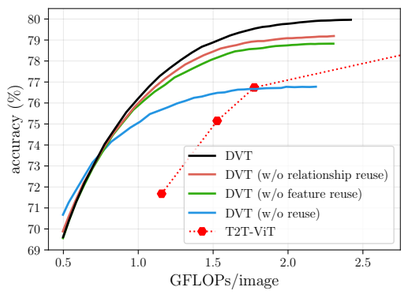


Figure 8: Performance of the DVT based on T2T-ViT-12 with and without the reuse mechanisms.

Comparisons with existing early-exiting networks are presented in Table 5. We adopt the same adaptive strategy (see: Section 3.3) for all the baselines. The Top-1 accuracy on ImageNet under different computational budgets is reported. One can observe that DVT significantly outperforms these baselines.

Table 5: DVT v.s. existing multi-exit models with the same adaptive inference strategy. The accuracy under each budget is reported.

Networks	Top-1 Acc.					
	0.75G	1.00G	1.25G	1.50G	1.75G	2.00G
MSDNet [23]	69.82%	71.24%	72.73%	73.66%	73.99%	74.20%
IMTA-MSDNet [27]	70.84%	71.92%	73.41%	74.31%	74.64%	74.94%
RANet [52]	70.48%	72.24%	73.57%	74.56%	75.02%	75.10%
DVT (T2T-ViT-12)	**73.70%**	**76.22%**	**77.89%**	**78.89%**	**79.47%**	**79.75%**

4.2 Ablation Study

Effectiveness of feature and relationship reuse. We conduct experiments by ablating one or both of the reuse mechanisms. For a clear comparison, we first deactivate the early-termination, and report the accuracy and GFLOPs corresponding to each exit in Table 6. The three-exit DVT based on T2T-ViT-12 is considered. One can observe that both the two reuse mechanisms are able to significantly boost the accuracy of DVT at the 2nd and 3rd exits with at most 6% additional computation, while they are compatible with each other to further improve the performance. We also find that involving computation reusing slightly hurts the accuracy at the 1st exit, which may be attributed to the compromise made by the first Transformer for downstream models. However, once the early-termination is adopted, this difference only results in trivial disadvantage when the computational budget is very small, as shown in Figure 8. DVT outperforms the baseline significantly in most cases.

Table 6: Effects of feature (F) and relationship (R) reuse. The percentages in brackets denote the additional computation compared to baselines involved by the reuse mechanisms.

Reuse		1st Exit (7x7)		2nd Exit (10x10)		3rd Exit (14x14)	
F	R	Top-1 Acc.	GFLOPs	Top-1 Acc.	GFLOPs	Top-1 Acc.	GFLOPs
		70.33%	0.47	73.54%	1.37	76.74%	3.15
✓		69.42%	0.47	75.31%	1.43$_{(4.4\%)}$	79.21%	3.31$_{(5.1\%)}$
	✓	69.03%	0.47	75.34%	1.41$_{(2.9\%)}$	78.86%	3.34$_{(6.0\%)}$
✓	✓	69.04%	0.47	**75.65%**	1.46$_{(6.6\%)}$	**80.00%**	3.50$_{(11.1\%)}$

Design choices for the reuse mechanisms. Here we study the design of the feature and relationship reuse mechanisms. For experimental efficiency, we consider a two-exit DVT based on T2T-ViT-12 using 7x7 and 10x10 tokens, while enlarge the batch size and the initial learning rate by 4 times. Such a training setting slightly degrades the accuracy of DVT, but it is still reliable to reflect the difference between different design variants. We deactivate early-termination, and report the performance of each exit. Notably, as the FLOPs of 1st exit remain unchanged (i.e., 0.47G), we do not present it.

We consider four variants of feature reuse in Table 7: (1) reusing features from the corresponding upstream layer instead of the final layer; (2) reusing classification token; (3) only performing feature reuse in the first layer of the downstream model; (4) removing the LN in $f_l(\cdot)$. One can see that taking final tokens of the upstream model and reusing them in each downstream layer are both important.

Ablation results for relationship reuse are presented in Table 8. We consider four variants as well: (1) only reusing the attention logits from the corresponding upstream layer; (2) only reusing the attention logits from the final upstream layer; (3) replacing the MLP by a linear layer in $r_l(\cdot)$; (4) adopting naive upsample operation instead of what is shown in Figure 5; The results indicate that it is beneficial to enable each downstream layer to flexibly reuse all upstream attention logits. Besides, naive upsampling significantly hurts the performance.

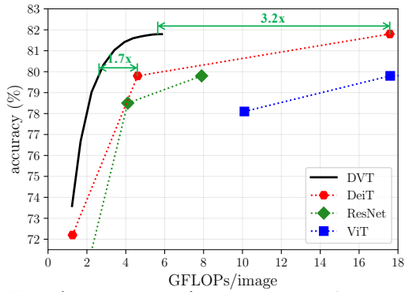


图7：基于Deit的DVT在Imagenet上的性能。deit-mand用作骨干。

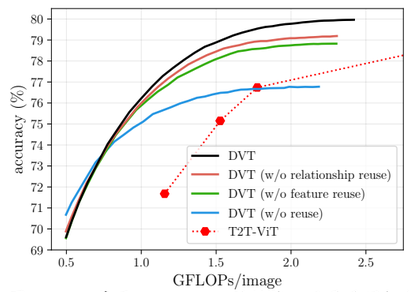


图8：DVT基于T2TVIT−12的性能，无需重用机制。

早期退出的网络呈现在表5中。我们采用

表5：DVT
V.S.具有相同自适应推理策略的现有多退出模型。报告了每次预算下的准确性。

the same adaptive strategy (see: Section 3.3) for all the baselines. The Top-1 accuracy on ImageNet under different computational budgets is reported. One can observe that DVT significantly outperforms these baselines.

Networks	Top-1 Acc.					
	0.75G	1.00G	1.25G	1.50G	1.75G	2.00G
MSDNet [23]	69.82%	71.24%	72.73%	73.66%	73.99%	74.20%
IMTA-MSDNet [27]	70.84%	71.92%	73.41%	74.31%	74.64%	74.94%
RANet [52]	70.48%	72.24%	73.57%	74.56%	75.02%	75.10%
DVT (T2T-ViT-12)	**73.70%**	**76.22%**	**77.89%**	**78.89%**	**79.47%**	**79.75%**

4.2 Ablation Study

Effectiveness of feature and relationship reuse. We conduct experiments by ablating one or both of the reuse mechanisms. For a clear comparison, we first deactivate the early-termination, and report the accuracy and GFLOPs corresponding to each exit in Table 6. The three-exit DVT based on T2T-ViT-12 is considered. One can observe that both the two reuse mechanisms are able to significantly boost the accuracy of DVT at the 2nd and 3rd exits with at most 6% additional computation, while they are compatible with each other to further improve the performance. We also find that involving computation reusing slightly hurts the accuracy at the 1st exit, which may be attributed to the compromise made by the first Transformer for downstream models. However, once the early-termination is adopted, this difference only results in trivial disadvantage when the computational budget is very small, as shown in Figure 8. DVT outperforms the baseline significantly in most cases.

表6：特征（f）和关系（r）重用的影响。与由重复使用机制涉及的基线相比，括号中的百分比表示附加计算。

Reuse		1st Exit (7x7)		2nd Exit (10x10)		3rd Exit (14x14)	
F	R	Top-1 Acc.	GFLOPs	Top-1 Acc.	GFLOPs	Top-1 Acc.	GFLOPs
		70.33%	0.47	73.54%	1.37	76.74%	3.15
✓		69.42%	0.47	75.31%	1.43$_{(4.4\%)}$	79.21%	3.31$_{(5.1\%)}$
	✓	69.03%	0.47	75.34%	1.41$_{(2.9\%)}$	78.86%	3.34$_{(6.0\%)}$
✓	✓	69.04%	0.47	**75.65%**	1.46$_{(6.6\%)}$	**80.00%**	3.50$_{(11.1\%)}$

重复使用机制的设计选择。在这里，我们研究了特征和关系重用机制的设计。对于实验性EF效率，我们考虑使用7x7和10x10令牌的T2T−VT−12基于T2T−VT−12的双出口DVT，同时将批量尺寸和初始学习率扩大4次。这种训练设置略微降低了DVT的准确性，但仍然可靠地重复不同设计变体之间的差异。我们停用早期终止，并报告每个出口的表现。值得注意的是，由于第一个出口的拖鞋保持不变（即0.47g），我们不呈现它。

我们考虑表7中的特征重用的四个变体：（1）从相应的上游层的重用特征而不是固定层;（2）重用分类令牌;（3）仅在下游模型的第一个层中执行功能重用;（4）在FL（·）中除去LN。可以看到上游模型的拍摄令牌并在每个下游层中重用它们都很重要。

关系重用的消融结果如表8所示。我们也考虑了四个变体：（1）仅从相应的上游层重用注意值;（2）只能从FI NAL上游层重用注意值;（3）通过RL（·）中的线性层替换MLP;（4）采用Naive Upsample操作而不是图5中所示的操作;结果表明，使每个下游层能够重复使用所有上游注意值。此外，天真的上采样的角度显得伤害了性能。

Table 7: Ablation studies for feature reuse.

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	2nd Exit (10x10) GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Layer-wise feature reuse	69.84%	74.31%	1.43
Reuse classification token	69.79%	74.70%	1.43
Remove $f_l(\cdot), l \geq 2$	69.33%	74.73%	1.38
Remove LN in $f_l(\cdot)$	69.63%	75.05%	1.42
Ours	69.44%	**75.23%**	1.43

Table 8: Ablation studies for relationship reuse.

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	2nd Exit (10x10) GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Layer-wise relationship reuse	69.63%	73.89%	1.38
Reuse final-layer relationships	69.25%	74.31%	1.39
MLP→Linear	69.20%	73.84%	1.38
Naive upsample	69.60%	73.34%	1.41
Ours	69.50%	**74.91%**	1.41

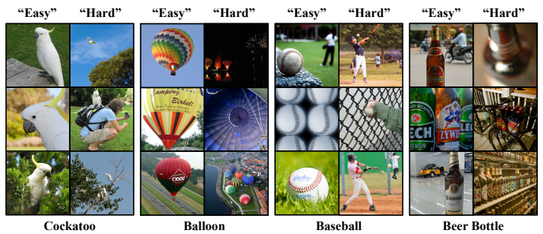


Figure 9: Visualization of the "easy" and "hard" samples in DVT.

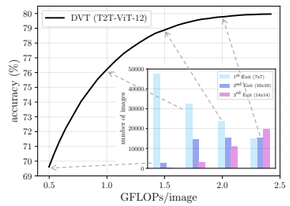


Figure 10: Numbers of images exiting at different exits with varying computational budgets.

Early-termination criterion. We vary the criterion for adaptive inference based on DVT (T2T-ViT-12) and report the accuracy under several computational budgets in Table 9. Two variants are considered: (1) adopting the entropy of softmax prediction to determine whether to exit [37]; (2) performing random exiting with the same exit proportion as DVT. The simple but effective confidence-based criterion achieves better performance than both of them.

Table 9: Comparisons of early-termination criterions. The accuracy under each budget is reported.

Ablation	Top-1 Acc. 0.75G	1.00G	1.25G	1.50G
Randomly Exit	70.19%	71.66%	72.61%	73.59%
Entropy-based	73.41%	75.21%	77.08%	78.40%
Confidence-based (ours)	**73.70%**	**76.22%**	**77.89%**	**78.89%**

Vision Transformers with adaptive depth. We test applying the adaptive depth paradigm [23, 52] to ViT in Table 10. Two classifiers are uniformly added to the intermediate layers of T2T-ViT-14, and the early-exiting is performed with the three exits in the same way as DVT. One can observe that the layer-adaptive T2T-ViT achieves significantly lower accuracy than DVT using identical computational budgets. This inferior performance may be alleviated by improving the network architecture, as shown in [23].

Table 10: Effects of applying the adaptive depth mechanism to vision Transformers. The accuracy under each budget is reported.

Ablation	Top-1 Acc. 2.00G	2.50G	3.00G	3.50G	4.00G
T2T-ViT-14 with adaptive depth	76.09%	78.88%	79.46%	79.54%	79.55%
DVT (T2T-ViT-14)	**79.24%**	**80.61%**	**81.47%**	**82.10%**	**82.42%**

4.3 Visualization

Figure 9 shows the images that are first correctly classified at the 1st and 3rd exits of the DVT (T2T-ViT-12). The former are recognized as "easy" samples, while the later are considered to be "hard". One can observe that "easy" samples usually depict the recognition objectives in clear and canonical poses and sufficiently large resolution. On the contrary, "hard" samples may contain complex scenes and non-typical poses or only include a small part of the objects, and require a finer representation using more tokens. Figure 10 presents the numbers of images that exit at different exits when the computational budget increases. The plot shows that the accuracy of DVT is significantly improved with more images exiting later, which is achieved by changing the confidence thresholds online.

5 Conclusion

In this paper, we sought to optimally configure a proper number of tokens for each individual image in vision Transformers, and hence proposed the *Dynamic Vision Transformer* (DVT) framework. DVT processes each test input by sequentially activating a cascade of Transformers using increasing numbers of tokens, until an appropriate token number is reached (measured by the prediction confidence). We further introduce the feature and relationship reuse mechanisms to facilitate efficient computation reuse. Extensive experiments indicate that DVT significantly improves the computational efficiency on top of state-of-the-art vision Transformers, both theoretically and empirically.

表7：用于特征重用的消融研究。

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	2nd Exit (10x10) GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Layer-wise feature reuse	69.84%	74.31%	1.43
Reuse classification token	69.79%	74.70%	1.43
Remove $f_l(\cdot), l \geq 2$	69.33%	74.73%	1.38
Remove LN in $f_l(\cdot)$	69.63%	75.05%	1.42
Ours	69.44%	**75.23%**	1.43

表8：关系重用的消融研究。

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	2nd Exit (10x10) GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Layer-wise relationship reuse	69.63%	73.89%	1.38
Reuse final-layer relationships	69.25%	74.31%	1.39
MLP→Linear	69.20%	73.84%	1.38
Naive upsample	69.60%	73.34%	1.41
Ours	69.50%	**74.91%**	1.41

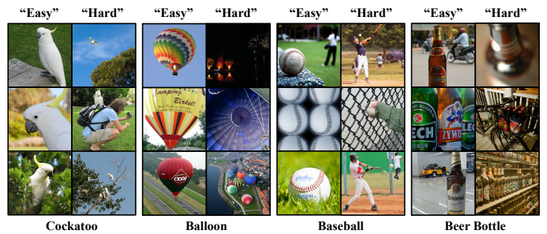


图9：DVT中的“轻松”和“硬”样本的可视化。

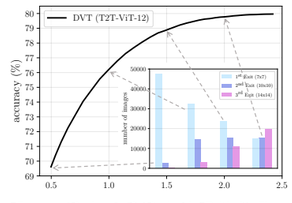


图10：退出不同退出的图像数量，具有不同的计算预算。

Early-termination criterion. We vary the criterion for adaptive inference based on DVT (T2T-ViT-12) and report the accuracy under several computational budgets in Table 9. Two variants are considered: (1) adopting the entropy of softmax prediction to determine whether to exit [37]; (2) performing random exiting with the same exit proportion as DVT. The simple but effective confidence-based criterion achieves better performance than both of them.

表9：提前终止标准的比较。报告了每次预算下的准确性。

Ablation	Top-1 Acc. 0.75G	1.00G	1.25G	1.50G
Randomly Exit	70.19%	71.66%	72.61%	73.59%
Entropy-based	73.41%	75.21%	77.08%	78.40%
Confidence-based (ours)	**73.70%**	**76.22%**	**77.89%**	**78.89%**

Vision Transformers with adaptive depth. We test applying the adaptive depth paradigm [23, 52] to ViT in Table 10. Two classifiers are uniformly added to the intermediate layers of T2T-ViT-14, and the early-exiting is performed with the three exits in the same way as DVT. One can observe that the layer-adaptive T2T-ViT achieves significantly lower accuracy than DVT using identical computational budgets. This inferior performance may be alleviated by improving the network architecture, as shown in [23].

表10：将自适应深度机制应用于视觉变压器的效果。报告了每次预算下的准确性。

Ablation	Top-1 Acc. 2.00G	2.50G	3.00G	3.50G	4.00G
T2T-ViT-14 with adaptive depth	76.09%	78.88%	79.46%	79.54%	79.55%
DVT (T2T-ViT-14)	**79.24%**	**80.61%**	**81.47%**	**82.10%**	**82.42%**

图9示出了在DVT（T2TVIT-12）的第1和第3次出口中正确分类的图像。前者被认为是“简单”样本，而后续被认为是“硬”。人们可以观察到“简单”样本通常描绘清晰，规范的姿势和大量分辨率的识别目标。相反，“硬”样本可以包含复杂的场景和非典型姿势，或者仅包括对象的一小部分，并且需要使用更多令牌的文件表示。图10显示了在计算预算增加时在不同退出时退出的图像的数量。该曲线表明，DVT的准确性是通过稍后退出的更多图像来显着提高，这是通过在线改变配置阈值来实现的更多图像。

5 Conclusion

在本文中，我们寻求最佳地对视觉变压器中的每个图像进行适当数量的令牌，因此提出了动态视觉变压器（DVT）框架。DVT通过顺序地激活每个测试输入，使用越来越多的令牌顺序激活变压器，直到达到适当的令牌编号（通过预测控制来测量）。我们进一步介绍了功能和关系重用机制，以便于EF展计算重用。广泛的实验表明，DVT标志在理论上和经验上，可以在最先进的视觉变压器顶部提高计算EF效率。

Acknowledgements

This work is supported in part by the National Science and Technology Major Project of the Ministry of Science and Technology of China under Grants 2018AAA0100701, the National Natural Science Foundation of China under Grants 61906106 and 62022048, and Huawei Technologies Ltd. In particular, we appreciate the valuable discussions with Chenghao Yang.

References

[1] A. Adcock, V. Reis, M. Singh, Z. Yan, L. van der Maaten, K. Zhang, S. Motwani, J. Guerin, N. Goyal, I. Misra, L. Gustafson, C. Changhan, and P. Goyal. Classy vision. `https://github.com/facebookresearch/ClassyVision`, 2019.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *NeurIPS*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[4] Shijie Cao, Lingxiao Ma, Wencong Xiao, Chen Zhang, Yunxin Liu, Lintao Zhang, Lanshun Nie, and Zhi Yang. Seernet: Predicting convolutional neural network feature-map sparsity through low-bit quantization. In *CVPR*, pages 11216–11225, 2019.

[5] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. *arXiv preprint arXiv:2103.14899*, 2021.

[6] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021.

[7] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPRW*, pages 702–703, 2020.

[8] Stéphane d'Ascoli, Hugo Touvron, Matthew Leavitt, Ari Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. *arXiv preprint arXiv:2103.10697*, 2021.

[9] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *ICML*, pages 248–255, 2009.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

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

[12] Alaaeldin El-Nouby, Natalia Neverova, Ivan Laptev, and Hervé Jégou. Training vision transformers for image retrieval. *arXiv preprint arXiv:2102.05644*, 2021.

[13] Haoqi Fan, Yanghao Li, Bo Xiong, Wan-Yen Lo, and Christoph Feichtenhofer. Pyslowfast. `https://github.com/facebookresearch/slowfast`, 2020.

[14] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. *arXiv preprint arXiv:2104.11227*, 2021.

[15] Michael Figurnov, Maxwell D Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov. Spatially adaptive computation time for residual networks. In *CVPR*, pages 1039–1048, 2017.

Acknowledgements

这项工作是由中国科学技术部的国家科学和技术主要项目支持，中国自然科学基金2018AAA0100701，中国国家自然科学基金会授予61906106和62022048，以及华为技术有限公司，我们欣赏与澄瑶杨有价值的讨论。

References

[1] A. Adcock，V.Reis，M. Singh，Z. Yan，L.Van der Maaten，K. Zhang，S. Motwani，J. Guerin，N. Goyal, I. Misra, L. Gustafson, C. Changhan, and P. Goyal. Classy vision. `https://github.com/facebookresearch/ClassyVision`, 2019.

[2] Jimmy Lei Ba，Jamie Ryan Kiros和Geoffrey E Hinton。层标准化。arxiv预印迹 *arXiv:1607.06450*, 2016.

[3]汤姆金龙，本杰明曼，尼克莱德，梅兰妮萨莱希，贾德德理解，傲慢的印刷品，Arvind Neelakantan，Pranav Shyam，Girish Sastry，Amanda Askell，Sandhini Agarwal，Ariel Herbert-Voss，Gretchen Krueger，Tom Henighan，Rewon Child，Aditya Ramesh，Daniel Ziegler，Jeffrey Wu，Clemens冬天，Chris Hesse，Mark Chen，Eric Sigler，Mateusz Litwin，Scott Gray，Benjamin棋，杰克克拉克，克里斯托弗伯尔伯勒，Sam McCandlish，Alec Radford，Ilya Sutskever和Dario Amodei。语言模型是几秒钟的学习者。在H. Larochelle，M.Ranzato，R. Hadsell，M.F.Balcan和H. Lin，编辑，Neurips，第33卷第1877-18-1810卷.Curran Associates，Inc.，2020。

[4] shi姐CA哦, l ing小MA, wen从ξ臭, Chen Zhang, Y UN心l IU, Lin套Zhang, l安顺n IE, 和芝阳。SENEET：通过低比特量化预测卷积神经网络功能映射稀疏性。在CVPR中，2019年第11216-11225页，2019年。

[5] C混-F UC很, Q U安抚fan, and RAM ES war panda. cross V IT: cross-attention multi-scale vision transformer for image classification. *arXiv preprint arXiv:2103.14899*, 2021.

[6] ξ按G向chu, zh IT Ian, BO Zhang, ξ尼龙Wang, ξ奥林Wei, hu A下X IA, 安定春华沉。视觉变压器的条件位置编码。Arxiv预印迹arxiv：2102.10882,2021。

[7] ekin d subuk，barit zoff，jonatha shlens和qauRangment：实用使用缩小的搜索空间自动化数据增强。在CVPRW中，页面702-703,2020。

[8]St é phaned'ascoli，Hugo Touvron，Matthew Leavitt，Ari Morcos，Giulio Biroli和Levent萨格。赛事：改善具有软卷积诱导偏差的视觉变压器。Arxiv预印迹arxiv：2103.10697,2021。

[9] J. Deng, W. dong, R. Soc her, l. l i, Kai l i, and life i-Fe i. image net: A large-scale hierarchical图像数据库。在ICML中，第248-255页，2009年。

[10] Jacob Devlin, Ming-Wei Chang, Kent on Lee, and Kristina to U谭OVA. Bert: pre-training关于语言理解的深双侧变压器。在Naacl-HLT（1），2019年6月的明尼苏达州Minneapolis页面4171-4186.计算语言学协会。

[11] Alex恶意DOS OVI T ski Y, Lucas be也如, Alexander KO了S你KO V, Dir看Weiss en born, ξ澳华Z还, Thomas Unterthiner，Mostafa dehghani，Matthias Minderer，Georg Heigold，Sylvain Gelly，Jakob Uszkoreit和Neil Houlsby。图像值为16x16字：变形金刚以缩放图像识别。在ICLR，2021年。

[12] Alaaeldin El-Noby，NataliaVellova，Ivan Laptev和Herv é Jegou。Transing Trans Vision.用于图像检索的卷轴。ARXIV预印迹arxiv：2102.05644,2021。

[13] H傲气fan, yang好l i, box ion G, wan-yen lo, and Christoph Fe IC惠特尼ho反而. py slow fast. `https://github.com/facebookresearch/slowfast`, 2020.

[14] H傲气fan, box ion G, kart提科亚man gala迷, yang好l i, Z Hi称y按, JIT恩drama里K, 安定Christoph Feichtenhofer. Multiscale vision transformers. *arXiv preprint arXiv:2104.11227*, 2021.

[15] Michael FI固然Nov, Maxwell D Collins, Y U困Z虎, Liz航, Jonathan Huang, Dmitry VE涂RO V, 和Ruslan Salakhutdinov。残差网络的空间自适应计算时间。在CVPR，2017年第1039-1048页，2017年。

[16] Ben Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. *arXiv preprint arXiv:2104.01136*, 2021.

[17] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *arXiv preprint arXiv:2103.00112*, 2021.

[18] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *arXiv preprint arXiv:2102.04906*, 2021.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[20] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. *arXiv preprint arXiv:2102.04378*, 2021.

[21] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *ICCV*, pages 1314–1324, 2019.

[22] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[23] Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Q Weinberger. Multi-scale dense networks for resource efficient image classification. In *ICLR*, 2018.

[24] Gao Huang, Zhuang Liu, Geoff Pleiss, Laurens Van Der Maaten, and Kilian Weinberger. Convolutional networks with dense connectivity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019.

[25] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *ECCV*, pages 646–661. Springer, 2016.

[26] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[27] Hao Li, Hong Zhang, Xiaojuan Qi, Ruigang Yang, and Gao Huang. Improved techniques for training adaptive deep networks. In *ICCV*, pages 1891–1900, 2019.

[28] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

[29] Ji Lin, Yongming Rao, Jiwen Lu, and Jie Zhou. Runtime neural pruning. In *NeurIPS*, pages 2181–2191, 2017.

[30] 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. *arXiv preprint arXiv:2103.14030*, 2021.

[31] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *ECCV*, pages 116–131, 2018.

[32] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *arXiv preprint arXiv:2106.02034*, 2021.

[33] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, pages 4510–4520, 2018.

[34] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

[35] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, volume 97, pages 6105–6114, 2019.

[36] Yehui Tang, Kai Han, Yunhe Wang, Chang Xu, Jianyuan Guo, Chao Xu, and Dacheng Tao. Patch slimming for efficient vision transformers. *arXiv preprint arXiv:2106.02852*, 2021.

[37] Surat Teerapittayanon, Bradley McDanel, and Hsiang-Tsung Kung. Branchynet: Fast inference via early exiting from deep neural networks. In *ICPR*, pages 2464–2469. IEEE, 2016.

[16]本格雷厄姆，阿拉塞林州埃尔·努比，雨果Touvron，石头股票，Armand Joulin，Herv é J é gou，和马蒂斯杜泽。Levit：Convnet的衣物的视觉变压器，用于更快推理。Arxiv预印迹arxiv：2104.01136,2021。

[17] Kai Han, an ξ 奥, en花W U, J Ian元g UO, C混进GX U, Andy UN和Wang. transformer in transformer. *arXiv preprint arXiv:2103.00112*, 2021.

[18] Y i增Han, GA o Huang, shi记song, LE yang, hong会Wang, Andy U林Wang. dynamic neural networks: A survey. *arXiv preprint arXiv:2102.04906*, 2021.

[19] Kai名he, ξ按G与Zhang, S豪情r嗯, and J Ian sun. deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[20] Shu听he, H AO luo, pi超Wang, fan Wang, H Aoli, and Wei Jiang. trans Reid: transformer based object re-identification. *arXiv preprint arXiv:2102.04378*, 2021.

[21] Andrew Howard, mark Sandler, Grace chu, l i昂−CH IE和Chen, BO Chen, Ming姓tan, Wei君Wang, Y U图Z虎, Ru o名pang, vi Jay VA速度Evan, ETA了. searching for Mobile net V3. INI cc V, pages 1314 1324, 2019.

[22] Andrew G Howard, me ng龙Z虎, BO Chen, Dmitry Kale你趁空哦, Wei君Wang, Tobias Weyand，Marco Andreetto和Hartwig Adam。MobileNets：用于移动视觉应用的EF卷积神经网络。Arxiv预印亚克日期：1704.04861,2017。

[23] GA o Huang, Dan路Chen, TI安红l i, Felix W U, Lauren是van的rm AA ten, and K I连Q Weinberger。用于资源的多尺寸密度网络，用于资源EF键图像分类。在ICLR，2018年。

[24] GA o Huang, Z黄l IU, Geoff P类赛事, Lauren S vander MA阿ten, and K I连Weinberg而. 具有密集连接的卷积网络。IEEE关于模式分析和机器智能的交易，2019年第1−1页。

[25] GA o Huang, Y u sun, Z黄l IU, Daniel sed RA, and K I连Q Weinberg而. deep networks with 随机深度。在ECCV中，第646−661页。Springer，2016年。

[26] Alex Krizhevsky和Geoffrey Hinton。从小图像学习多层特征。

Technical report, Citeseer, 2009.

[27] H Aoli, hong Zhang, ξ AO卷Q i, Ru I缸yang, and GA o Huang. improved techniques for 培训自适应深网络。在ICCV，2019年第1891−1900页第1891−1900页。

[28] ya为l i, Kai Zhang, J IE张C拗, RA的UT IMO福特, and l UC VA NGO OL. local V IT: bringing 视觉变形金刚的地方。Arxiv预印迹arxiv：2104.05707,2021。

[29] J i Lin, yo闹革命RA O, J i问l U, and J IE Zhou. runtime neural pruning. inn Eur IPS, pages 2181–2191, 2017.

[30] Z Eli U, Y u通Lin, Y UE CA哦, Han hu, Y i宣Wei, Z横Zhang, Stephen Lin, and b爱宁郭。Swin变压器：使用Shifted Windows的分层视觉变压器。ARXIV预印迹ARXIV：2103.14030,2021。

[31] Ningning Ma, Xiangyu Zhang, Hai−Tao Zheng, and Jian Sun. Shuf enet v2: Practical guidelines 用于EFICE CNN架构设计。在ECCV，2018年第116−131页第116−131页。

[32] yo闹革命RA O, wen两Zhao, Ben林L IU, J i问l U, J IE Zhou, and CH O−J UI HS IE和. D有namicvit：具有动态令牌的efiecient视觉变压器，具有动态令牌稀疏。Arxiv预印迹arxiv：2106.02034,2021。

[33] mark Sandler, Andrew Howard, me ng龙Z虎, Andre YZ红魔GI Nov, 按DLI昂−CH IE和Chen. MobileNetv2：倒置残留和线性瓶颈。在CVPR中，2018年第4510−4520页，2018年。

[34] Christian Szegedy，Vincent Vanhoucke，Sergey Iroffe，Jon Shlens和Zbigniew War。回覆思考计算机愿景的初始架构。在CVPR，2016年第2818−2826,2016页。

[35]明兆革棕褐色和呼号。ef fi cientnet：卷积神经网络的重新思考模型缩放网络。在ICML，第97卷，第6105−6114页，2019。

[36] Y E会tang, Kai Han, Y UN和Wang, Chang X U, J Ian元g UO, C好X U, 按DD A城tao. 用于EFICEIIE视觉变压器的贴片纤细。Arxiv预印迹arxiv：2106.02852,2021。

[37] Surat Teerapittayanon，Bradley McDanel和Hsiang−Tsung Kung。Branchynet：快速推断通过早期退出深神经网络。在ICPR，第2464−2469页。IEEE，2016年。

[38] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020.

[39] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. *arXiv preprint arXiv:2103.17239*, 2021.

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *NeurIPS*, volume 30. Curran Associates, Inc., 2017.

[41] Andreas Veit and Serge Belongie. Convolutional networks with adaptive inference graphs. In *ECCV*, pages 3–18, 2018.

[42] Thomas Verelst and Tinne Tuytelaars. Dynamic convolutions: Exploiting spatial sparsity for faster inference. In *CVPR*, pages 2320–2329, 2020.

[43] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*, 2019.

[44] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. *arXiv preprint arXiv:2102.12122*, 2021.

[45] Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E Gonzalez. Skipnet: Learning dynamic routing in convolutional networks. In *ECCV*, pages 409–424, 2018.

[46] Yulin Wang, Zhaoxi Chen, Haojun Jiang, Shiji Song, Yizeng Han, and Gao Huang. Adaptive focus for efficient video recognition. *arXiv preprint arXiv:2105.03245*, 2021.

[47] Yulin Wang, Kangchen Lv, Rui Huang, Shiji Song, Le Yang, and Gao Huang. Glance and focus: a dynamic approach to reducing spatial redundancy in image classification. In *NeurIPS*, 2020.

[48] Darrell Whitley. A genetic algorithm tutorial. *Statistics and computing*, 4(2):65–85, 1994.

[49] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021.

[50] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[51] Zuxuan Wu, Tushar Nagarajan, Abhishek Kumar, Steven Rennie, Larry S Davis, Kristen Grauman, and Rogerio Feris. Blockdrop: Dynamic inference paths in residual networks. In *CVPR*, pages 8817–8826, 2018.

[52] Le Yang, Yizeng Han, Xi Chen, Shiji Song, Jifeng Dai, and Gao Huang. Resolution adaptive networks for efficient inference. In *CVPR*, pages 2369–2378, 2020.

[53] Le Yang, Haojun Jiang, Ruojin Cai, Yulin Wang, Shiji Song, Gao Huang, and Qi Tian. Condensenet v2: Sparse feature reactivation for deep networks. In *CVPR*, 2021.

[54] Kun Yuan, Shaopeng Guo, Ziwei Liu, Aojun Zhou, Fengwei Yu, and Wei Wu. Incorporating convolution designs into visual transformers. *arXiv preprint arXiv:2103.11816*, 2021.

[55] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, 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.

[56] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *PICCV*, pages 6023–6032, 2019.

[57] Fangao Zeng, Bin Dong, Tiancai Wang, Cheng Chen, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. *arXiv preprint arXiv:2105.03247*, 2021.

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

[59] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *CVPR*, pages 6848–6856, 2018.

[38] Hugo Touvron，Matthieu绳子，Matthijs十二，Francisco Massa，Alexandre Sablayrolles，以及Herv é J é gou。通过注意培训数据 - EFICEICE图像变压器和蒸馏。Arxiv预印迹arxiv：2012.12877,2020。

[39] Hugo Touvron，Matthieu Cord，Alexandre Salabayrolles，Gabriel Synnaeve和Herv é J é gou。与图像变形金刚更深入。Arxiv预印迹ARXIV：2103.17239,2021。

[40] ashish打进了名称，nicki perma，jacob，jacob，leon琼斯，艾登没有戈麦斯，Quasz Kaiser和Illia Polosukhin。关注你所需要的。在I. Guyon，U.V. Luxburg，S.Bengio，H. Wallach，R.Fergus，S. Vishwanathan，以及R.Garnett，编辑，神经潜望者，第30卷。

[41] Andreas Veit和Serge Ipplie。具有自适应推理图的卷积网络。在*ECCV*, pages 3–18, 2018.

[42] Thomas Verelst和Tinne Tuytelaars。动态互联网：利用空间稀疏性更快推断。在CVPR中，页面2320-2329,2020。

[43] Alex Wang，Amanpreet Singh，Julian Michael，Felix Hill，Omer Levy和Samuel R. Bowman。胶水：自然语言理解的多任务基准和分析平台。在ICLR，2019年。

[44] wen还Wang, E能在ex IE, X Ian GL i, Deng-ping fan, Kai套song, ding Lian G, ton Glu, ping罗，凌邵。金字塔视觉变压器：无卷积的密集预测多功能骨干。Arxiv预印迹arxiv：2102.12122,2021。

[45]鑫王，渔业玉，紫益斗，特雷维尔·达雷尔和约瑟夫·埃格祖兹。Skipnet：学习卷积网络中的动态路由。在ECCV，2018年409-424页。

[46] Y U林Wang, Zhao系Chen, ha O君Jiang, shi记song, Y i增Han, and GA o Huang. adaptive专注于EF牌视频识别。Arxiv预印迹arxiv：2105.03245,2021。

[47] Y U林Wang, KA ng陈LV, R UI Huang, shi记song, LE yang, and GA o Huang. glance and focus:减少图像分类中空间冗余的动态方法。在Neurips，2020年。

[48] Darrell Whitley。一种遗传算法教程。统计和计算，4（2）：65-85,1994。

[49] H爱萍W U, bin ξ奥, Noel cod Ella, men g尘l IU, ξ样D爱, l u yuan, and lei Zhang. CVT：向视觉变压器引入卷曲。Arxiv预印迹arxiv：2103.15808,2021。

[50] Yuxin Wu，Alexander Kirillov，Francisco Massa，Wan-Yen Lo和Ross Girshick。Detectron2。`https://github.com/facebookresearch/detectron2`, 2019.

[51]吴，瓜尔诺伊，Grauman和Rogerio Feris。BlockDrop：残差网络中的动态推理路径。在CVPR，2018年第8817-8826页，2018年。

[52] LE yang, Y i增Han, ξ Chen, shi记song, J i疯D爱, and GA o Huang. resolution adaptive用于EFICIEN推断的网络。在CVPR中，页面2369-2378,2020。

[53] LE yang, ha O君Jiang, Ru o谨CAI, Y U林Wang, shi记song, GA o Huang, and Q IT Ian. con DENSENET V2：深网络的稀疏功能重新激活。在CVPR，2021中。

[54] kun yuan, SHA open GG UO, Z I为l IU, AO君Zhou, Fe ng为y U, and Wei W U. incorporating卷积设计到视觉变压器中。Arxiv预印迹arxiv：2103.11816,2021。

[55] l i yuan, Y UN碰Chen, tao Wang, Wei好y U, Y U君shi, Francis eh ta有, J IA是Fe ng, 安定水城燕。令牌到令牌VIT：从想象中的划痕训练视觉变压器。Arxiv预印迹arxiv：2101.11986,2021。

[56]发票，Dongon Hana，Seong六月哦，Sanguuk，6月，6月，6月和瑜伽yoo。Cutmix：正规化策略培训具有可定位功能的强大分类。在Piccv，页面6023-6032,2019。

[57] fan高Zen G, bin dong, TI安彩Wang, Cheng Chen, ξ按G与Zhang, Andy i陈Wei. MO TR: End-to-end multiple-object tracking with transformer. *arXiv preprint arXiv:2105.03247*, 2021.

[58]洪义张，穆斯图Cisse，Yann N. Dauphin和David Lopez-Paz。混合：超越经验风险最小化。在ICLR，2018年。

[59] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shuf enet: An extremely ef cient用于移动设备的卷积神经网络。在CVPR中，第6848-6856,2018页。

[60] Moju Zhao, Kei Okada, and Masayuki Inaba. Trtr: Visual tracking with transformer. *arXiv preprint arXiv:2105.03817*, 2021.

[61] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, volume 34, pages 13001–13008, 2020.

[62] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021.

[60] Moju Zhao，Kei Okada和Masayuki Inaba。TRTR：使用变压器的视觉跟踪。阿克西夫 *preprint arXiv:2105.03817*, 2021.

[61] Z混z洪, l i昂Z恒, GU O两K昂, S耗子l i, Andy i yang. random erasing data 增强。Inaai，Vun 2：，01008,2000。

[62] DA全Zhou, Bing以K昂，ξ AO姐jin, Lin姐yang, ξ奥称Lian, Z i行Jiang, Q i比NH偶, 和贾西冯。DeepVit：走向更深的视觉变压器。Arxiv预印迹arxiv：2103.11886,2021。

14

Appendix

A Training Details

The proposed DVT framework is implemented based on the official code of T2T-ViT[4] [55] and DeiT[5] [38] when these two models are used as backbones, respectively. The training of DVT follows exactly the same configurations as the backbones, which are also recommended by their official implementations. The details on optimizer, learning rate schedule, batch size and other hyper-parameters can be easily found in their papers or code. Note that a number of regularization and data augmentation techniques are exploited, including RandAugment [7], Random Erasing [61], Label Smoothing [34], Mixup [58], Cutmix [56], and stochastic depth [25]. We train all the models with 8 NVIDIA V100 GPUs.

Table 11: Effects when the location for performing feature reuse varies.

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Reuse on shallow half of downstream layers	69.67%	75.02%	1.40
Reuse on deep half of downstream layers	69.94%	74.57%	1.40
Reuse on all downstream layers	69.44%	**75.23%**	1.43

Table 12: Effects when the location for performing relationship reuse varies.

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Reuse on shallow half of downstream layers	69.72%	74.89%	1.40
Reuse on deep half of downstream layers	70.00%	73.68%	1.40
Reuse on all downstream layers	69.50%	**74.91%**	1.41

Table 13: Effects of taking attention logits from varying upstream layers.

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Reuse relationships from shallow half of upstream layers	69.93%	74.67%	1.40
Reuse relationships from deep half of upstream layers	69.55%	74.58%	1.40
Reuse relationships from all upstream layers	69.50%	**74.91%**	1.41

B Additional Results

Which downstream layers benefit more from reuse? To shed light on the layer-wise reuse paradigm in the downstream model, we test performing feature or relationship reuse only in the shallow/deep half of downstream layers. The same experimental protocol as ablating the design of reuse mechanisms is adopted. The results are shown in Tables 11 and 12. Obviously, it is more important to reuse upstream features/relationships at the shallow layers. This phenomenon indicates that the main effects of the proposed reuse mechanisms lie in helping the first several transformer layers to rapidly extract discriminative representations or to learn accurate attention maps. The successive layers focus more on further improving the shallow features, while less on leveraging upstream information, which may have been effectively integrated into the shallow layers.

Which upstream layers contribute more to relationship reuse? In Table 13, we further test only taking the attention logits from the shallow/deep half of upstream layers in relationship reuse. One can observe that both shallow and deep relationships are important for boosting the accuracy of the

[4] https://github.com/yitu-opensource/T2T-ViT
[5] https://github.com/facebookresearch/deit

Appendix

A Training Details

当这两个模型用作骨架时，基于T2T−VIT4 [55]和DEIT5
[38]的FIIAL代码来实现所提出的DVT框架。DVT的训练遵循与骨干的完全相同的配置，也是由其唯一的实施建议的。在他们的论文或代码中可以轻松找到关于优化器，学习率计划，批量大小和其他超参数的详细信息。注意，利用了许多正则化和数据增强技术，包括randaugment
[7]，随机擦除[61]，标记平滑[34]，混合[58]，切割[56]和随机深度[25]。我们用8个NVIDIA V100
GPU训练所有型号。

表11：执行功能重用的位置时的效果变化。

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Reuse on shallow half of downstream layers	69.67%	75.02%	1.40
Reuse on deep half of downstream layers	69.94%	74.57%	1.40
Reuse on all downstream layers	69.44%	**75.23%**	1.43

表12：执行关系重用的位置时的效果变化。

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Reuse on shallow half of downstream layers	69.72%	74.89%	1.40
Reuse on deep half of downstream layers	70.00%	73.68%	1.40
Reuse on all downstream layers	69.50%	**74.91%**	1.41

表13：注意注意标志从不同上游层的影响。

Ablation	1st Exit (7x7) Top-1 Acc.	2nd Exit (10x10) Top-1 Acc.	GFLOPs
w/o reuse	**70.08%**	73.61%	1.37
Reuse relationships from shallow half of upstream layers	69.93%	74.67%	1.40
Reuse relationships from deep half of upstream layers	69.55%	74.58%	1.40
Reuse relationships from all upstream layers	69.50%	**74.91%**	1.41

B Additional Results

哪个下游层面越过重用？在下游模型中的层展重用范例上脱光，我们只测试在下游层的浅/深层中的表演功能或关系重用。采用相同的实验协议作为烧蚀重用机构的设计。结果显示在表11和12中。显然，在浅层处重用上游特征/关系更为重要。这种现象表明，所提出的重用机制的主要效果在于帮助第一款变压器层快速提取歧视性表示或学习准确的注意图。连续的层更加侧重于进一步改善浅功能，而在利用上游信息的情况下较少，这可能已经有效地集成到浅层中。

哪个上游层有助于重复使用更多？在表13中，我们进一步测试仅从关系重用中从浅/深层的浅/深层次的注意值进行进一步测试。人们可以观察到浅层和深层的关系都很重要，对提高准确性

[4] https://github.com/yitu-opensource/T2T-ViT
[5] https://github.com/facebookresearch/deit

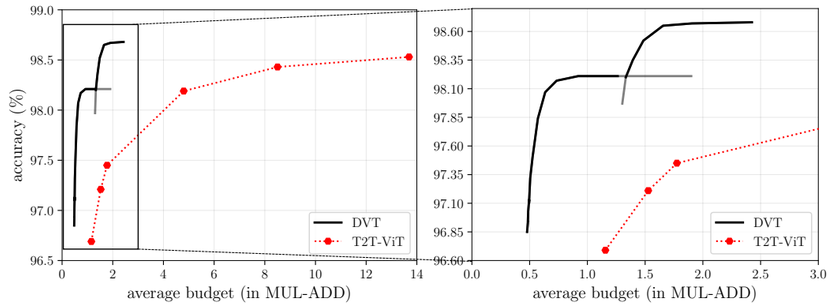


Figure 11: Top-1 accuracy v.s. GFLOPs on CIFAR-10. DVT is implemented on top of T2T-ViT-12/14.

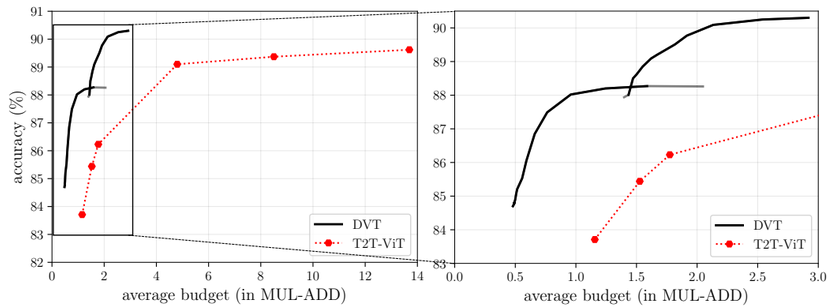


Figure 12: Top-1 accuracy v.s. GFLOPs on CIFAR-100. DVT is implemented on top of T2T-ViT-12/14.

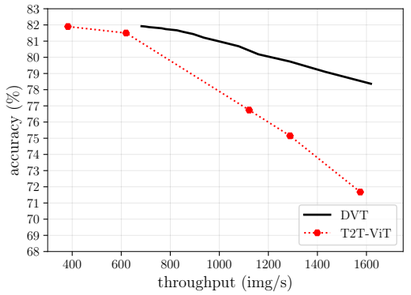


Figure 13: Top-1 accuracy v.s. throughput on ImageNet. The results are obtained on NVIDIA 2080Ti GPU with a batch size of 128.

downstream model. In addition, it is interesting that reusing more relationships only slightly improves the performance. We attribute this to the redundancy within the learned attention logits from different layers of the upstream model.

Top-1 accuracy v.s. GFLOPs curves on CIFAR-10/100 are presented in Figures 11 and 12, respectively, corresponding to the results reported in Table 3 of the paper.

Top-1 accuracy v.s. throughput curves on ImageNet are presented in Figures 13 corresponding to the results reported in Table 2 of the paper.

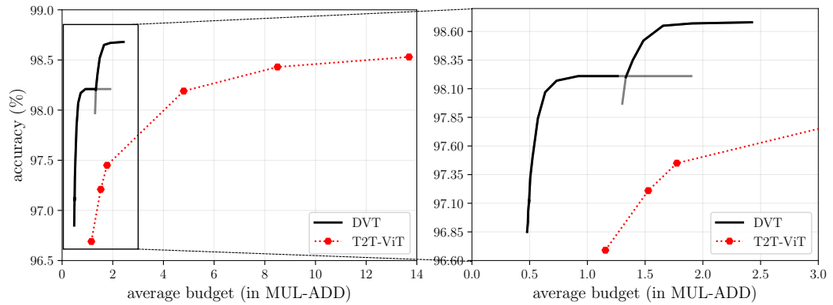


图11：前1个精度V.S.CiFar-10上的GFLOPS。DVT在T2T-VT-12/14顶部实现。

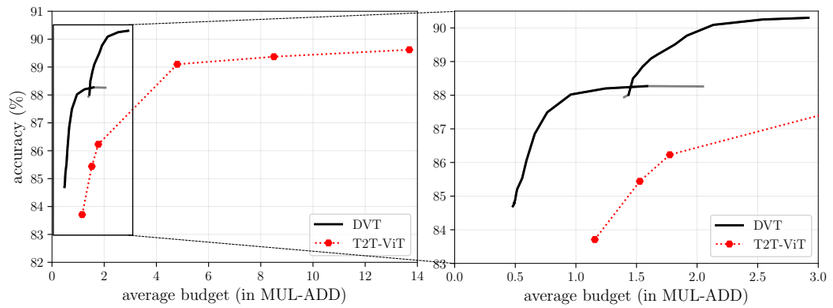


图12：前1个精度V.S.Cifar-100上的GFLOPS。DVT在T2T-VIT12 / 14的顶部实现。

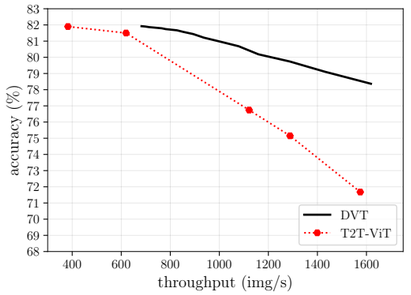


图13：前1个精度V.S.想象成的吞吐量。在NVIDIA 2080TI GPU上获得的结果，批量为128。

下游模型。此外，有趣的是重用更多关系只会略微提高性能。我们将此归因于来自上游模型的不同层的知名度标志中的冗余。

前1个精度V.S.CIFAR-10/100上的GFLOPS曲线分别在图11和12中呈现，对应于纸张的表3中报告的结果。

前1个精度V.S.想象成的吞吐量曲线在图13中呈现，对应于纸张的表2中的结果。