Which Transformer to Favor: A Comparative Analysis of Efficiency in Vision Transformers

Tobias Christian Nauen*,[1,2] **Sebastian Palacio,**[2] **Andreas Dengel**[1,2]

[1]University of Kaiserslautern-Landau (RPTU), Gottlieb-Daimler-Straße, 67663 Kaiserslautern, Germany
[2]German Research Center for Artificial Intelligence (DFKI), Trippstadter Str. 122, 67663 Kaiserslautern, Germany
{tobias_christian.nauen, sebastian.palacio, andreas.dengel}@dfki.de

Abstract

The growing popularity of Vision Transformers as the go-to models for image classification has led to an explosion of architectural modifications claiming to be more efficient than the original ViT. However, a wide diversity of experimental conditions prevents a fair comparison between all of them, based solely on their reported results. To address this gap in comparability, we conduct a comprehensive analysis of more than 30 models to evaluate the efficiency of vision transformers and related architectures, considering various performance metrics. Our benchmark provides a comparable baseline across the landscape of efficiency-oriented transformers, unveiling a plethora of surprising insights. For example, we discover that ViT is still Pareto optimal across multiple efficiency metrics, despite the existence of several alternative approaches claiming to be more efficient. Results also indicate that hybrid attention-CNN models fare particularly well when it comes to low inference memory and number of parameters, and also that it is better to scale the model size, than the image size. Furthermore, we uncover a strong positive correlation between the number of FLOPS and the training memory, which enables the estimation of required VRAM from theoretical measurements alone.

Thanks to our holistic evaluation, this study offers valuable insights for practitioners and researchers, facilitating informed decisions when selecting models for specific applications. We publicly release our code and data.[1]

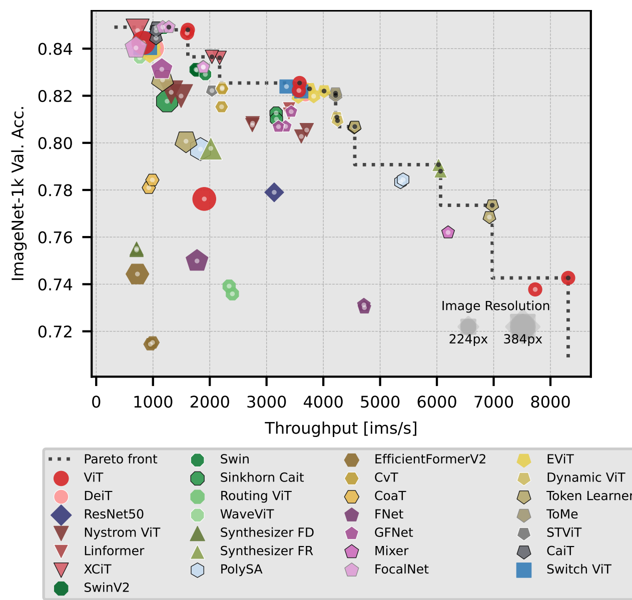

Figure 1: Pareto front of throughput and accuracy with unified legend for the plots of this paper. Markers and hues signify the different taxonomy classes.

Introduction

Recent interest in transformer architectures has led to state-of-the-art solutions for natural language processing (NLP) and computer vision (CV). In the past, specialized architectures existed for different problems in AI. Language tasks usually relied on recurrent neural networks (RNNs) whereas vision problems typically used convolutional neural networks (CNNs). However, the introduction of the transformer (Vaswani et al. 2017) triggered both research communities to adapt its principles to solve all kinds of problems in language and vision. In particular for image classification, the Vision Transformer (ViT) (Dosovitskiy et al. 2021) has positioned itself as one of the best known applications of the original architecture. ViT has achieved state-of-the-art performance on benchmarks like ImageNet, surpassing more traditional CNN architectures (Zhai et al. 2022; Yu et al. 2022).

However, a major challenge in working with transformer models is dealing with the computational complexity of the self-attention mechanism. This mechanism enables the transformer to capture global dependencies between all pairs of sequence positions, but it has a computational complexity of $\mathcal{O}(N^2)$ in the input length, which makes it impractical for long sequences and high-resolution images.

Efforts have been made to reduce the computational complexity in transformer models, particularly in resource-constrained settings, such as embedded systems. Researchers from CV and NLP have explored numerous strategies, like the implementation of sparse, local, or kernelized attention mechanisms, as well as token removal criteria to decrease the sequence length. However, selecting the most efficient model that meets certain performance standards remains a challenging task. This objective is particularly dif-

*corresponding author: tobias_christian.nauen@dfki.de
[1]https://github.com/tobna/WhatTransformerToFavor

ficult, knowing that "*efficiency*" can refer to different concepts, like training resources, inference speed, memory requirements, or number of model parameters. Moreover, difficulties identifying the most efficient, yet highly performant vision transformer are accentuated by the different training and evaluation conditions reported in the literature. It is even more unclear how tradeoffs in efficiency from NLP carry over to vision tasks.

In order to address these problems, we devise a taxonomy to classify changes made to ViT to increase its efficiency. This helps us attribute efficiency gains to general strategies. We then design a fair testbed to compare some of the most recent advances in efficient ViTs using a wide range of metrics. Through this testbed, we provide comprehensive baselines for efficient ViTs in image classification, conducting a comprehensive review and benchmark of the current state of research on efficient transformer models for vision.

To ensure a fair empirical comparison, we adopt the training pipeline by (Touvron, Cord, and Jégou 2022) and train every model architecture on this pipeline. It is an update to the widely popular pipeline by (Touvron et al. 2021a), that has been used in several papers to train efficient vision transformers. Our results, as shown in Figure 1, reveal that, even with the emergence of alternative architectures claiming greater efficiency, a well-trained ViT remains Pareto optimal, showcasing remarkable efficiency in terms of throughput while preserving its high accuracy. Additionally, we gain valuable insights regarding the Pareto optimality of sequence reduction techniques and hybrid attention models, as well as the inefficiency of fine-tuning at higher resolutions. Our approach enables us to evaluate the inherent strengths and weaknesses of different model architectures and to measure real-world performance metrics under consistent conditions, providing researchers and practitioners with a valuable resource for selecting the most efficient and effective model architecture for their specific use case.

Contributions

- A taxonomy that highlights the different ways in which transformer-based architectures can be made more efficient, and an overview of efficient transformers used in both NLP and CV.
- A comprehensive benchmark of different transformer-based models. Experiments are run under similar conditions, making the results comparable to one another. In particular, we provide image classification baselines for models that were proposed for NLP.
- A comparative analysis of model efficiency under different criteria: Number of parameters, speed, and memory, with identification of Pareto optimal models, as well as a correlation analysis between these metrics.

Related Work

Previous studies have surveyed transformers across various problem domains. Additionally, researchers have evaluated the efficiency of different deep learning models, both theoretically and empirically. This section provides an overview of relevant work on transformer surveys and approaches to measure efficiency, while the taxonomy section delves into specific efficient transformer-like architectures.

Surveys on Transformers Efficiency has become a critical aspect of transformers, leading to the exploration of various strategies and evaluations across domains. Surveys on efficient transformers provide valuable taxonomies and insights. (Tay et al. 2022) focus on efficiency gains primarily in NLP, while (Fournier, Caron, and Aloise 2023) and (Zhuang et al. 2023) collect general approaches for enhancing model efficiency, including those applicable to transformers. For ViTs, (Patro and Agneeswaran 2023) presents an extensive list of efficient versions, classified based on different aspects such as computational complexity, robustness, and transparency. They, along with (Liu et al. 2023), compare the efficiency of ViTs in terms of parameters, ImageNet accuracy, and other performance metrics using data from the original papers. Surveys on ViT-like models, conducted by (Han et al. 2023), (Yang et al. 2022b), and (Khan et al. 2022), focus on categorizing models according to different vision tasks, while (Zuo et al. 2022) specializes in dense prediction. Additionally, (Islam 2022) examines the development of ViT-like models across various vision tasks over time, and (Xu et al. 2021b) focuses on different levels of usage, such as high-level, low-level, and backbone models. Specialized surveys investigate the application of transformers in specific domains, such as action recognition (Ulhaq et al. 2022), image restoration (Ali et al. 2023), medical imaging (Shamshad et al. 2023; Li et al. 2023; He et al. 2023; Parvaiz et al. 2023), or remote sensing (Aleissaee et al. 2023). Additional surveys explore transformers in other modalities, like speech recognition (Latif et al. 2023), language processing (Casola, Lauriola, and Lavelli 2020), time series analysis (Wen et al. 2022), or multimodal tasks (Xu, Zhu, and Clifton 2022).

Efficiency in Deep Learning Efficiency evaluation in deep learning models is another area of investigation. (Bartoldson, Kailkhura, and Blalock 2022) provides a theoretical overview of efficiency aspects and metrics, along with measurement methodologies. (Dehghani et al. 2022) offers a detailed discussion on efficiency metrics, highlighting the potential pitfalls of relying solely on theoretical metrics. (Canziani, Paszke, and Culurciello 2016) conducts a highly regarded survey on the efficiency of CNNs, and (Liang et al. 2022) compares the efficiency of their novel architectures with older models using the Pareto front of throughput and accuracy. Finally, (Tay et al. 2021b) constructs a comprehensive benchmark to quantify various aspects of transformer model performance, facilitating efficiency evaluations.

In contrast, this paper focuses on different overarching strategies of making transformers in vision more efficient. We contribute by empirically evaluating efficient transformers across multiple metrics, providing insights and comparisons among these different approaches.

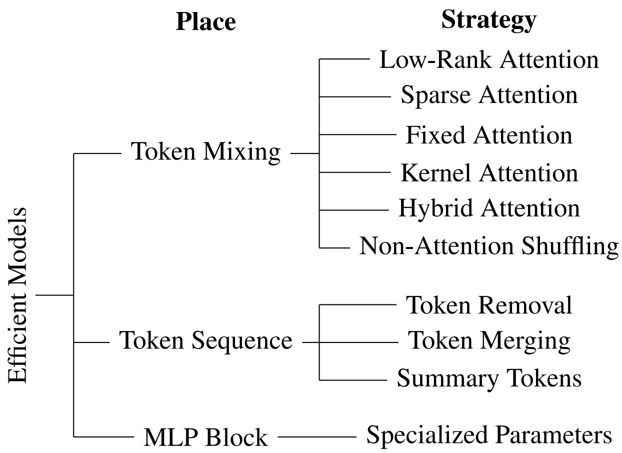

Figure 2: Taxonomy for categorizing strategies for efficient ViT-like models.

Taxonomy

Bottleneck of the Vision Transformer

First, we present the key elements of ViTs, that have been studied to make it more efficient, as well as its key bottleneck: the $\mathcal{O}(N^2)$ computational complexity. By identifying these components, we can establish a taxonomy based on the main strategies that have been proposed to enhance transformer efficiency in the literature.

ViT is an adaptation of the original transformer model for image processing tasks. Instead of text, ViT takes an image as input and converts it into a sequence of non-overlapping patches. Each patch is linearly embedded into a token of size d, with positional encoding and a classification token $[\text{CLS}]$ added to the sequence , which is then fed through a transformer encoder.

There, the self-attention mechanism computes the attention weights A between tokens, utilizing the query ($Q \in \mathbb{R}^{N\times d}$) and key ($K \in \mathbb{R}^{N\times d}$) matrices:

$$A = \operatorname{softmax}\left(\frac{QK^\top}{\sqrt{d_\text{head}}}\right) \in \mathbb{R}^{N\times N},$$

where the softmax is calculated row-wise. This matrix encodes the global interactions between every possible pair of tokens, but it's also the reason why the attention mechanism has an inherent computational (space & time) complexity of $\mathcal{O}(N^2)$. The output sequence of the attention-mechanism is a weighted sum of the input sequence, using the value matrix (V):

$$X_\text{out} = AV = \operatorname{softmax}\left(\frac{QK^\top}{\sqrt{d_\text{head}}}\right) V. \tag{1}$$

After self-attention, the sequence is passed through a feedforward network with MLP layers, and in the end, only the $[\text{CLS}]$ token is used for the classification decision.

Efficiency-Improving Changes

Now that we have gone over the backbone that constitutes a ViT, we can discuss the most important modifications, which have been proposed to make it more efficient. To better understand the different approaches, we propose a taxonomy based on where in the model a change is made (Figure 2). Here, we identify three main areas: The token mixing mechanism, the token sequence, and the MLP block. Each of which we will describe in more detail, below. While this taxonomy is not meant to be a comprehensive overview of ViT-like models, it is proposed as a tool for identifying the most popular strategies to make ViTs more efficient.

Token Mixing

The first and most popular approach is to change the *token mixing* mechanism, which directly tackles the $\mathcal{O}(N^2)$ computational complexity of self-attention. There are multiple strategies through which this can be accomplished: Some methods approximate the attention mechanism with reduced computation, which can be achieved by matrix decomposition, changing the order of operations, or fixing attention values. Other approaches combine attention with CNNs to perform sub-sampling in the attention mechanism or reduce the number of uses of the attention mechanism. Finally, some methods discard the attention mechanism and instead introduce entirely new token mixing strategies.

Low-rank attention leverages the fact, that Q, and K in Equation (1) are matrices of shape $N \times d$, which makes $QK^\top \in \mathbb{R}^{N\times N}$ a matrix of rank $r \leq d \ll N$. The *Linformer* (Wang et al. 2020) utilizes this to project the sequence direction of K and V down to dimension $k \ll N$, without reducing the informational content of the attention matrix too much. Similarly, the *Nyströmformer* (Xiong et al. 2021) uses the Nyström method of matrix decomposition to approximate the matrix $QK^\top$. The approximate attention mechanism's output is then computed with linear complexity by applying the $\operatorname{softmax}$ to each part individually. The approach of *XCiT* (El-Nouby et al. 2021) utilizes a transposed attention mechanism:

$$Y^\top = \operatorname{softmax}\left(\frac{1}{\sqrt{d}}Q^\top K\right) V^\top.$$

Here, $Q^\top K \in \mathbb{R}^{d\times d}$ is used to replace the low-rank matrix $QK^\top$, to define a globally informed filter $A = \operatorname{softmax}\left(\frac{1}{\sqrt{d}}Q^\top K\right) \in \mathbb{R}^{d\times d}$ which is applied to each token individually. Since both $Q \in \mathbb{R}^{N\times d}$ and $K \in \mathbb{R}^{N\times d}$ are likely to be of rank d, the former most likely has full rank, which enables more efficient information encoding. In contrast to these dynamic approximations of the global interactions, there is **sparse attention**. Normal attention tends to focus on only a few input tokens, giving low attention scores to most other tokens (Kim et al. 2021). Sparse attention takes advantage of this by setting most of the attention weights to zero and only calculating the most important ones. The parts of the attention matrix, which are determined dynamically, can be based on a fixed pattern, like in the widely used *Swin Transformer* (Liu et al. 2021), and *SwinV2* (Liu et al. 2022), where attention is performed only inside local sets of image tokens, or in *HaloNet* (Vaswani et al. 2021), where each token can only attend to its local neighborhood. Alternatively, *Routing Transformer* (Roy et al. 2021) determines sets of tokens that exchange information by grouping them based on their content. In contrast, *Sinkhorn Transformer* (Tay et al. 2020) fixes local groups and lets tokens only attend to a different group using a dynamic permutation matrix. Another way of keeping the attention matrix sparse, is by sub-sampling the keys and values. For example, *Wave-ViT* (Yao et al. 2022) uses a discrete wavelet transform for sub-sampling, while *CvT* (Wu et al. 2021) utilizes a convolution with stride two.

An extreme example of setting attention values beforehand is learning a **fixed attention**-matrix A. In this case, attention is only dependent on a tokens position. This is explored in the *Synthesizer* (Tay et al. 2021a).

Kernel attention is an approach that changes the order of computations in Equation (1). Instead of using the softmax on the product $QK^\top$, a kernel $\varphi : \mathbb{R}^d \to \mathbb{R}^d$ is applied to the queries Q and keys K individually:

$$Y = \varphi(Q)\varphi(K)^\top V,$$

which can be calculated with linear complexity $\mathcal{O}(N)$ in the number of tokens. Various kernels have been proposed, including the random Gaussian kernel, introduced with the *Performer* (Choromanski et al. 2021) and expanded on in the *FourierLearner-Transformer* (Choromanski et al. 2023) by adding a relative positional encoding in phase space. *Scatterbrain* (Chen et al. 2021) combines the Performers' attention mechanism with sparse attention. *Poly-SA* (Babiloni et al. 2023) uses the identity function $\varphi \equiv \mathrm{id}$, *Linear Transformer* (Katharopoulos et al. 2020) uses an ELU, and another variant of the *Performer* uses a ReLU kernel.

Hybrid attention approaches combine convolutions with the attention mechanism. *EfficientFormer(-V2)* (Li et al. 2022) uses convolutions initially to focus on local interactions, and then employs the attention mechanism to capture global interactions. Another approach is to use convolutions inside the attention mechanism to create locally informed queries, keys, and values, as in *CvT* (Wu et al. 2021), where it is used to subsample keys and values, and in *CoaT* (Xu et al. 2021a).

In addition to these attention-based methods, recent works have explored **non-attention shuffling** techniques to capture global interactions. *MLP-Mixer* (Tolstikhin et al. 2021) applies a fully connected layer along the token sequence for global interactions, while *FocalNet* (Yang et al. 2022a) extracts a hierarchy of contexts using convolutions. Moreover, *FNet* (Lee-Thorp et al. 2022; Sevim et al. 2022) directly exploits the Fast Fourier Transform (FFT) for $\mathcal{O}(N \log N)$ complexity token mixing, while *GFNet* (Rao et al. 2021b) utilizes it to implement a global convolution.

Token Sequence

The second category, *token sequence*, is more prevalent in efficient transformers used in computer vision compared to natural language processing. The idea here is to remove redundant information typically contained in images, and in doing so, reducing computational costs without significantly affecting the model's performance. Methods aim to reduce the token sequence by removing unnecessary image patches, such as those associated with the background, by merging similar patches to minimize redundant information, or by summarizing the information into a smaller number of abstract tokens to represent higher-level information. These models leverage the $\mathcal{O}(N^2)$ complexity of the self-attention mechanism to attain a large reduction in computational cost, as removing 30% of the tokens reduces the operations needed by around 50%.

For **token removal**, it is essential to determine which tokens to remove without losing critical information. To that end, *Dynamic ViT* (Rao et al. 2021a) uses the Gumbel-softmax for token retention probabilities, and *A-ViT* (Yin et al. 2022) learns halting probabilities that weight the token outputs of different depths. In contrast, *EViT* (Liang et al. 2022) avoids introducing extra parameters by utilizing the previous layer's attention matrix to remove tokens.

Another strategy is to remove redundant information through **token merging**. While a version of *EViT* (Liang et al. 2022) merges the unimportant tokens, *ToMe* (Bolya et al. 2023) merges tokens based on their similarity, using a fast bipartite matching algorithm.

The approaches discussed so far modify the existing token sequence. However, another strategy is to condense it into a few **summary tokens**. *CaiT* (Touvron et al. 2021b) uses token summary via cross-attention on a single token to gather global information on the classification decision in the last layers. *Token Learner* (Ryoo et al. 2021) creates a set number of summary tokens by employing dynamic sums over the image tokens, and *STViT* (Chang et al. 2023) initializes summary tokens with local information by using strided convolutions, then injecting global information into those using cross-attention.

MLP Block

The final way in which proposed methods change the architecture of transformers is by moving compute to the MLP block, which has linear complexity with respect to the sequence length. Despite room for efficiency gains, we have identified just one model taking this approach. *Switch Transformer* (Fedus, Zoph, and Shazeer 2022) focuses on improving performance while maintaining a similar computational cost. This is tackled by introducing multiple sets of **specialized parameters** for each MLP block and passing different tokens through blocks with different parameter sets. Therefore, Switch Transformer can have multiple times the number of parameters of the ViT without introducing lots of extra computations.

To recap, the taxonomy we presented offers a structured framework for understanding the diverse approaches to improve the efficiency of ViT-like models. Consequently, researchers can gain a deeper understanding of the key changes employed to make ViTs more efficient, while it also enables us to compare broad trends across strategies.

Benchmarking Efficiency

We conduct an extensive series of experiments on the models we used for defining our taxonomy, to be able to evaluate which models are ideal under a given set of constraints.

Baselines

In order to find the most efficient models, we need to quantify their efficiency gains by comparing them to the original ViT (Dosovitskiy et al. 2021), as well as its follow-up version (Touvron et al. 2021a; Touvron, Cord, and Jégou 2022). We also include metrics for ResNet50 (He et al. 2016) as a representative baseline for CNN architectures and a point of comparison across papers. We evaluate on the ImageNet-1k dataset (Deng et al. 2009), since it's one of the best known benchmarks in CV.

Training Pipeline

The models were trained for a total of 140 epochs using the training pipeline introduced by (Touvron, Cord, and Jégou 2022). This is an updated version of the pipeline used by the DeiT model (Touvron et al. 2021a), which has been used successfully by many authors in CV for training their efficient ViTs (see Table 1), which is why we consider the updated version a fair point of comparison for all models.

Pretraining was conducted on a cleaned up version of ImageNet-21k (Ridnik et al. 2021) for 90 epochs at resolutions of 224×224 and 192×192 pixels, followed by fine-tuning on ImageNet-1k at 224×224 and 384×384 pixels for 50 epochs. In cases where training was unstable, we adjusted some hyperparameters based on values from the corresponding publications. All training was conducted using 4 or 8 NVIDIA A100 GPUs. While most of the models worked well with our training pipeline, we could not get Performer, Linear Transformer, and HaloNet to converge. See the supplementary material for more details.

Efficiency Metrics

The term *efficiency* can take on different meanings; it is therefore crucial to consider multiple dimensions when evaluating a models efficiency. In this paper, we focus on three dimensions: memory efficiency, inference efficiency, and training efficiency. Memory efficiency refers to training or deploying models with limited VRAM, inference efficiency evaluates the speed of predictions, and training efficiency measures the model's ability to learn in limited time.

To assess efficiency, we use the theoretical metrics of number of parameters and FLOPS, which provide estimates of the model's representational capacity and computational requirements. However, theoretical metrics do not always correlate with real-world performance, especially when comparing different architectures (Dehghani et al. 2022). Instead, empirical metrics, obtained by running models on hardware, offer more accurate evaluations, which is why we also track GPU time for training, inference throughput at optimal batch sizes, and VRAM requirements. While empirical metrics can be sensitive to hardware and software configurations, we ensure consistent evaluations by employing the same setup for all models.

Due to the complex trade-offs, efficiency cannot be captured by a single number. Hence, we focus on the Pareto front to identify models that achieve the best trade-offs between two metrics. The Pareto front represents the most efficient compromises, where a point on the front outperforms every other point in at least one metric.

Model	DeiT Based	Orig. Acc.	New Acc.	Δ
ViT-S (DeiT)	✓	79.8	82.54	+ 2.74
ViT-S (DeiT III)		82.6	82.54	- 0.06
XCiT-S	✓	82.0	83.65	+ 1.65
Swin-S	✓	83.0	84.87	+ 1.87
SwinV2-Ti		81.7	83.09	+ 1.39
Wave-ViT-S		82.7	83.61	+ 0.91
Poly-SA-ViT-S		71.48	78.34	+ 6.86
EfficientFormer-V2-S0		$75.7^{\star}$	71.53	- 4.17
CvT-13		83.3↑	82.35	- 0.95
CoaT-Ti	✓	78.37	78.42	+ 0.05
GFNet-S		80.0	81.33	+ 1.33
FocalNet-S		83.4	84.91	+ 1.51
DynamicViT-S		$83.0^{\star}$	81.09	- 1.91
EViT (delete)	✓	79.4	82.29	+ 2.89
EViT (fuse)	✓	79.5	81.96	+ 2.46
ToMe-ViT-S	✓	79.42	82.11	+ 2.69
CaiT-S24	✓	82.7	84.91	+ 2.21
TokenLearner-ViT-8		77.87↓	80.66	+ 2.79
STViT-Swin-Ti	✓	80.8	82.22	+ 1.42

Table 1: ImageNet-1k accuracy differences of the original papers and the new training pipeline. Models are trained on 224×224 pixel images, unless marked with ↑ or ↓, in which case they are trained on 384×384 and 112×112 pixels, respectively. Results marked with * are obtained using knowledge distillation.

Results

ImageNet-1k Accuracy

To ensure a fair evaluation and comparable results across models, we first compare the ImageNet accuracy reported in the original papers and the one obtained after training with our pipeline in Table 1. We can see that a substantial fraction of these pipelines is based on the one from (Touvron et al. 2021a), making them a good fit for training with the updated version of that pipeline.

Overall, we observe that almost all the models trained with our pipeline achieve a higher accuracy; 1.35% on average. Additionally, all the models for which the original pipeline significantly outperformed ours were trained using knowledge distillation or fine-tuned at a higher resolution. The highest improvement is +6.86% for Poly-SA, indicating possible training instabilities in the original paper, similar to ones we faced with the other Kernel Attention models.

Most taxonomy classes contain models that achieve peak accuracies of approximately 85% (see the supplementary material). However, two notable exceptions are observed: the *kernel attention* class, which poses challenges in optimizing models, and the *fixed attention* class, where the use of a constant attention matrix inherently results in a lower accuracy. In summary, the pipeline provides a strong and comparable baseline for all models, regardless of architectural differences.

Number of Parameters

The number of parameters is used in the literature as a proxy metric for tracking the model complexity, and the

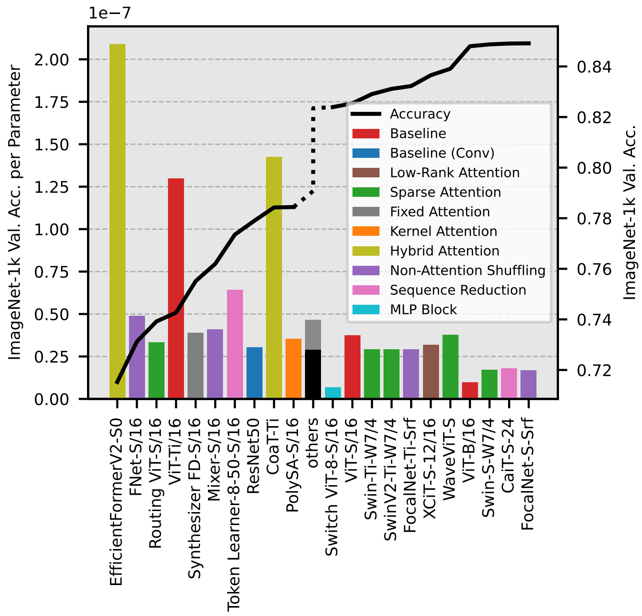

Figure 3: Accuracy and accuracy per parameter of different models at a resolution of 224×224 pixels ordered by model accuracy. We grouped 14 models of intermediate accuracies into the ‘others‘ column, for which the light gray bar shows the range.

overall computational cost of using a model. When analyzing the parameter efficiency of different models (Figure 3), it is evident that for most smaller models, the accuracy per parameter remains relatively constant at about $4 \times 10^{-8} \frac{\%}{\text{param.}}$. However, this value is approximately halved for larger models, which indicates diminishing returns on scaling the model size. The smaller hybrid attention models, EfficientFormerV2-S0 and CoaT-Ti, exhibit the highest accuracy per parameter, significantly outperforming other attention-based models as well as ResNet50, a CNN. This suggests that the combination of attention and convolutions allows for the development of very parameter-efficient models. For the baseline ViT, there is a noticeable drop in accuracy per parameter across the model sizes. While ViT-Ti outperforms models of comparable accuracy, and ViT-S performs on par with similar models, ViT-B slightly underperforms when compared to other, larger models.

Figure 4 highlights the Pareto boundary between accuracy and the number of parameters, showing models with fewer than 30 million parameters. It is noteworthy that the majority of Pareto optimal models are fine-tuned at a higher resolution of 384 pixels, which contributes to an increase in accuracy without a any growth in the parameter count. At the smallest model sizes, we observe once again, that EfficientFormerV2-S0 and CoaT-Ti are Pareto optimal models, while the TokenLearner is a Pareto optimal choice between ViT-Ti and the also Pareto optimal ViT-S.

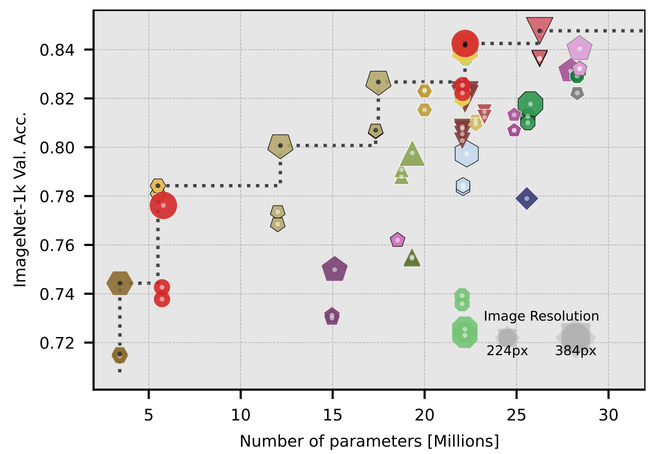

Figure 4: Pareto front of number of parameters vs. accuracy for models with at most 30 million parameters. Legend in Figure 1.

Speed

The inference speed is a critical metric of significant importance for practitioners when making decisions regarding model deployment. Whether driven by strict requirements for real-time processing or the desire to obtain model outputs within reasonable timeframes, inference speed directly impacts the usability and effectiveness of deployed models. The models we evaluated often claim a superior throughput vs. accuracy trade-off compared to the original ViT. However, our comprehensive evaluation (see Figure 1) reveals that ViT remains Pareto optimal at all sizes. Additionally, it becomes evident that only a subset of models, namely Synthesizer-FR and some sequence reduction models demonstrate improvements in the Pareto front when compared to a ViT of the corresponding size.

Moreover, our observations indicate that fine-tuning at a higher resolution of 384 pixels is not an efficient strategy. While it may result in improved model accuracy, it entails a significant increase in computational cost, leading to a substantial reduction in throughput. Consequently, opting for the next larger model turns out to be more efficient. Although a larger model may involve more floating-point operations, these are parallelized more effectively, resulting in higher overall throughput as well as a higher accuracy.

We observe a substantial correlation of 0.81 between inference time and fine-tuning time, which is supported by the overall similarities we see in the Pareto fronts. Additionally, in our analysis of fine-tuning time, the TokenLearner model emerges as a standout performer, demonstrating the fastest fine-tuning speed while achieving a commendable accuracy of 77.35%. For more details, see the supplementary material.

Memory

VRAM imposes a significant constraint in deep learning research and practice, as it enforces hard limitations on the usable models. When optimizing for VRAM usage during inference, the analysis presented in Figure 5 reveals the remarkable performance of the *hybrid attention* models CoaT

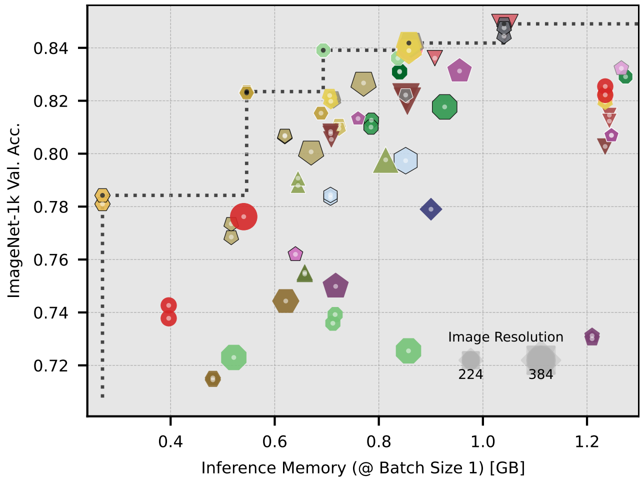

Figure 5: Excerpt of the Pareto front of inference memory and accuracy for models which need less then 1.25GB of VRAM. Legend in Figure 1.

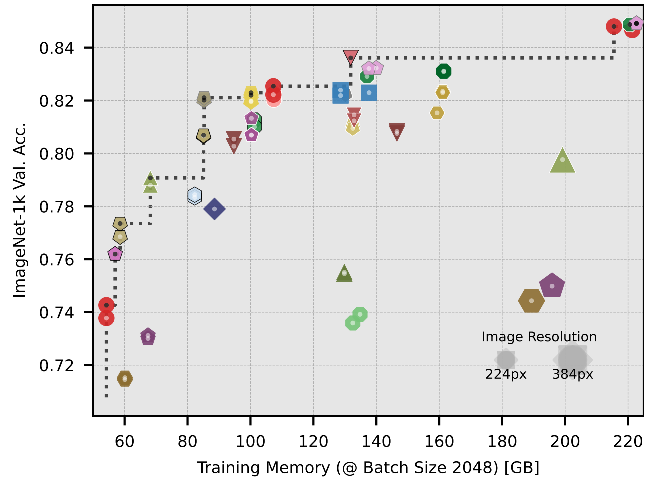

Figure 6: Excerpt of the Pareto front of training memory and accuracy for models which need less than 225GB of VRAM for training. Legend in Figure 1.

corr(x, y)	θ	*FT Time*	*FFwd Time*	*FFwd Mem*	*Train Mem*
FLOPS	0.30	**0.72**	0.48	0.42	**0.85**
θ		0.05	0.02	0.40	0.18
FT Time			**0.81**	0.17	**0.89**
FFwd Time				0.13	**0.71**
FFwd Memory					0.48

Table 2: Correlation between the number of floating point operations (FLOPS), number of parameters (θ), fine-tuning time (*FT Time*), inference time (*FFwd Time*), inference memory (*FFwd Mem*) and training memory (*Train Mem*).

and CvT. Notably, Wave-ViT, EViT and ToMe, fine-tuned at 384 px, and CaiT form the Pareto front for larger sizes. Importantly, ViT fails to achieve Pareto optimality in this metric. This observation suggests that, similar to the number of parameters, hybrid attention models excel in low memory environments. In contrast, the aspect of training memory (Figure 6) exhibits a similar pattern as observed in throughput, and consequently we measured a relatively high correlation of 0.71 between training memory and inference time (see the supplementary material). Here, CoaT and CvT need more memory than other models of comparable accuracy, and again ViT maintains its Pareto optimality, along with the sequence reduction models EViT, ToMe, TokenLearner, and Synthesizer-FR, and XCiT.

Correlation of Metrics

We discovered the highest correlation coefficient of 0.89 between fine-tuning time and training memory. This strong correlation suggests a common underlying factor or bottleneck, possibly related to the necessity of memory reads during training. Understanding this relationship can provide valuable insights into the factors influencing training efficiency. Intriguingly, the highest correlation coefficient of a theoretical metric is 0.85, found between FLOPS and training memory, suggesting that the VRAM required for training can be roughly estimated based on the theoretical FLOPS using the following approximation (plot in the supplementary material):

$$\text{VRAM [GB]} \approx 25.43 \cdot \text{GFlops} + 25.50.$$

Surprisingly, the other evaluated metrics exhibit relatively weak correlations when considering different model architectures as seen in Table 2, which highlights the limited reliability of estimating computational costs solely based on theoretical metrics. Consequently, assessing model efficiency in practical scenarios requires the measurement of throughput and memory requirements for novel architectures.

Conclusion

We provide an extensive and empirical analysis of transformer-like architectures for vision, providing valuable insights into their performance and efficiency across multiple dimensions. Through rigorous experiments, we have demonstrated that a well-trained ViT remains Pareto optimal along various dimensions, emphasizing its effectiveness as a baseline model. Furthermore, our findings highlight the efficiency of sequence reducing techniques. We have also established that fine-tuning at a higher resolution is not an efficient strategy, as scaling the model's size turns out to be more effective. When comparing different efficiency metrics, we have uncovered the limitations of estimating computational costs solely based on theoretical metrics. However, VRAM requirements for training can be roughly estimated based on the theoretical FLOPS of a model.

Importantly, we hope that our research will be used as a reference baseline, providing a solid foundation for further research in the ongoing efforts of developing efficient deep learning models. The insights gained from this study can inform the design and optimization of models for various practical applications.

References

Aleissaee, A. A.; Kumar, A.; Anwer, R. M.; Khan, S.; Cholakkal, H.; Xia, G.-S.; and Khan, F. S. 2023. Transformers in Remote Sensing: A Survey. *Remote Sensing*, 15(7): 1860.

Ali, A. M.; Benjdira, B.; Koubaa, A.; El-Shafai, W.; Khan, Z.; and Boulila, W. 2023. Vision Transformers in Image Restoration: A Survey. *Sensors*, 23(5).

Babiloni, F.; Marras, I.; Deng, J.; Kokkinos, F.; Maggioni, M.; Chrysos, G.; Torr, P.; and Zafeiriou, S. 2023. Linear Complexity Self-Attention with 3rd Order Polynomials. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–12.

Bartoldson, B. R.; Kailkhura, B.; and Blalock, D. 2022. Compute-Efficient Deep Learning: Algorithmic Trends and Opportunities.

Bolya, D.; Fu, C.-Y.; Dai, X.; Zhang, P.; Feichtenhofer, C.; and Hoffman, J. 2023. Token Merging: Your ViT But Faster. In *International Conference on Learning Representations*.

Canziani, A.; Paszke, A.; and Culurciello, E. 2016. An Analysis of Deep Neural Network Models for Practical Applications. arXiv:1605.07678.

Casola, S.; Lauriola, I.; and Lavelli, A. 2020. Pre-trained transformers: an empirical comparison. *Machine Learning with Applications*, 9: 100334.

Chang, S.; Wang, P.; Lin, M.; Wang, F.; Zhang, D. J.; Jin, R.; and Shou, M. Z. 2023. Making Vision Transformers Efficient from A Token Sparsification View. arXiv:2303.08685.

Chen, B.; Dao, T.; Winsor, E.; Song, Z.; Rudra, A.; and Ré, C. 2021. Scatterbrain: Unifying Sparse and Low-rank Attention Approximation. In Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*.

Choromanski, K. M.; Li, S.; Likhosherstov, V.; Dubey, K. A.; Luo, S.; He, D.; Yang, Y.; Sarlos, T.; Weingarten, T.; and Weller, A. 2023. Learning a Fourier Transform for Linear Relative Positional Encodings in Transformers. arXiv:2302.01925.

Choromanski, K. M.; Likhosherstov, V.; Dohan, D.; Song, X.; Gane, A.; Sarlos, T.; Hawkins, P.; Davis, J. Q.; Mohiuddin, A.; Kaiser, L.; Belanger, D. B.; Colwell, L. J.; and Weller, A. 2021. Rethinking Attention with Performers. In *International Conference on Learning Representations*.

Dehghani, M.; Tay, Y.; Arnab, A.; Beyer, L.; and Vaswani, A. 2022. The Efficiency Misnomer. In *International Conference on Learning Representations*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

El-Nouby, A.; Touvron, H.; Caron, M.; Bojanowski, P.; Douze, M.; Joulin, A.; Laptev, I.; Neverova, N.; Synnaeve, G.; Verbeek, J.; and Jegou, H. 2021. XCiT: Cross-Covariance Image Transformers. In Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*. arXiv.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *The Journal of Machine Learning Research*, 23(1): 5232–5270.

Fournier, Q.; Caron, G. M.; and Aloise, D. 2023. A Practical Survey on Faster and Lighter Transformers. *ACM Comput. Surv.*

Han, K.; Wang, Y.; Chen, H.; Chen, X.; Guo, J.; Liu, Z.; Tang, Y.; Xiao, A.; Xu, C.; Xu, Y.; Yang, Z.; Zhang, Y.; and Tao, D. 2023. A Survey on Vision Transformer. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 45(01): 87–110.

He, K.; Gan, C.; Li, Z.; Rekik, I.; Yin, Z.; Ji, W.; Gao, Y.; Wang, Q.; Zhang, J.; and Shen, D. 2023. Transformers in medical image analysis. *Intelligent Medicine*, 3(1): 59–78.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Islam, K. 2022. Recent Advances in Vision Transformer: A Survey and Outlook of Recent Work. arXiv:2203.01536.

Katharopoulos, A.; Vyas, A.; Pappas, N.; and Fleuret, F. 2020. Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention. *International Conference on Machine Learning*, 5156–5165.

Khan, S.; Naseer, M.; Hayat, M.; Zamir, S. W.; Khan, F. S.; and Shah, M. 2022. Transformers in Vision: A Survey. *ACM Computing Surveys*, 54(10s): 1–41.

Kim, K.; Wu, B.; Dai, X.; Zhang, P.; Yan, Z.; Vajda, P.; and Kim, S. 2021. Rethinking the Self-Attention in Vision Transformers. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE.

Latif, S.; Zaidi, A.; Cuayahuitl, H.; Shamshad, F.; Shoukat, M.; and Qadir, J. 2023. Transformers in Speech Processing: A Survey. arXiv:2303.11607.

Lee-Thorp, J. P.; Ainslie, J.; Eckstein, I.; and Ontañón, S. 2022. FNet: Mixing Tokens with Fourier Transforms. In *NAACL*.

Li, J.; Chen, J.; Tang, Y.; Wang, C.; Landman, B. A.; and Zhou, S. K. 2023. Transforming medical imaging with Transformers? A comparative review of key properties, current progresses, and future perspectives. *Medical Image Analysis*, 85: 102762.

Li, Y.; Hu, J.; Wen, Y.; Evangelidis, G.; Salahi, K.; Wang, Y.; Tulyakov, S.; and Ren, J. 2022. Rethinking Vision Transformers for MobileNet Size and Speed. arXiv:2212.08059.

Liang, Y.; Ge, C.; Tong, Z.; Song, Y.; Wang, J.; and Xie, P. 2022. Not All Patches are What You Need: Expediting Vision Transformers via Token Reorganizations. In *International Conference on Learning Representations*.

Liu, Y.; Zhang, Y.; Wang, Y.; Hou, F.; Yuan, J.; Tian, J.; Zhang, Y.; Shi, Z.; Fan, J.; and He, Z. 2023. A Survey of Visual Transformers. *IEEE Transactions on Neural Networks and Learning Systems*, 1–21.

Liu, Z.; Hu, H.; Lin, Y.; Yao, Z.; Xie, Z.; Wei, Y.; Ning, J.; Cao, Y.; Zhang, Z.; Dong, L.; Wei, F.; and Guo, B. 2022. Swin Transformer V2: Scaling Up Capacity and Resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11999–12009. arXiv.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 9992–10002. Los Alamitos, CA, USA: IEEE Computer Society.

Parvaiz, A.; Khalid, M. A.; Zafar, R.; Ameer, H.; Ali, M.; and Fraz, M. M. 2023. Vision Transformers in medical computer vision—A contemplative retrospection. *Engineering Applications of Artificial Intelligence*, 122: 106126.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, 8024–8035. Curran Associates, Inc.

Patro, B. N.; and Agneeswaran, V. S. 2023. Efficiency 360: Efficient Vision Transformers. arXiv:2302.08374.

Rao, Y.; Zhao, W.; Liu, B.; Lu, J.; Zhou, J.; and Hsieh, C.-J. 2021a. DynamicViT: Efficient Vision Transformers with Dynamic Token Sparsification. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 13937–13949. Curran Associates, Inc.

Rao, Y.; Zhao, W.; Zhu, Z.; Lu, J.; and Zhou, J. 2021b. Global Filter Networks for Image Classification. *Advances in Neural Information Processing Systems*, 34.

Ridnik, T.; Ben-Baruch, E.; Noy, A.; and Zelnik-Manor, L. 2021. ImageNet-21K Pretraining for the Masses. arXiv:2104.10972.

Roy, A.; Saffar, M.; Vaswani, A.; and Grangier, D. 2021. Efficient Content-Based Sparse Attention with Routing Transformers. *Transactions of the Association for Computational Linguistics*, 9: 53–68.

Ryoo, M. S.; Piergiovanni, A.; Arnab, A.; Dehghani, M.; and Angelova, A. 2021. TokenLearner: What Can 8 Learned Tokens Do for Images and Videos? *NeurIPS 2021*.

Sevim, N.; Özyedek, E. O.; Şahinuç, F.; and Koç, A. 2022. Fast-FNet: Accelerating Transformer Encoder Models via Efficient Fourier Layers. arXiv:2209.12816.

Shamshad, F.; Khan, S.; Zamir, S. W.; Khan, M. H.; Hayat, M.; Khan, F. S.; and Fu, H. 2023. Transformers in medical imaging: A survey. *Medical Image Analysis*, 102802.

Tay, Y.; Bahri, D.; Metzler, D.; Juan, D.-C.; Zhao, Z.; Zheng, C.; Tay, Y.; Bahri, D.; Metzler, D.; Juan, D.-C.; Zhao, Z.; and Zheng, C. 2021a. Synthesizer: Rethinking Self-Attention in Transformer Models. In *International conference on machine learning*, 10183–10192. PMLR.

Tay, Y.; Bahri, D.; Yang, L.; Metzler, D.; and Juan, D.-C. 2020. Sparse Sinkhorn Attention. In *International Conference on Machine Learning*, 9438–9447. PMLR, arXiv.

Tay, Y.; Dehghani, M.; Abnar, S.; Shen, Y.; Bahri, D.; Pham, P.; Rao, J.; Yang, L.; Ruder, S.; and Metzler, D. 2021b. Long Range Arena: A Benchmark for Efficient Transformers. In *International Conference on Learning Representations*.

Tay, Y.; Dehghani, M.; Bahri, D.; and Metzler, D. 2022. Efficient Transformers: A Survey. *ACM Comput. Surv.*

Tolstikhin, I.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Keysers, D.; Uszkoreit, J.; Lucic, M.; and Dosovitskiy, A. 2021. MLP-Mixer: An all-MLP Architecture for Vision. arXiv:2105.01601.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jegou, H. 2021a. Training data-efficient image transformers & distillation through attention. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 10347–10357. PMLR.

Touvron, H.; Cord, M.; and Jégou, H. 2022. DeiT III: Revenge of the ViT. In Avidan, S.; Brostow, G.; Cissé, M.; Farinella, G. M.; and Hassner, T., eds., *Computer Vision – ECCV 2022*, 516–533. Cham: Springer Nature Switzerland.

Touvron, H.; Cord, M.; Sablayrolles, A.; Synnaeve, G.; and Jegou, H. 2021b. Going deeper with Image Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE.

Ulhaq, A.; Akhtar, N.; Pogrebna, G.; and Mian, A. 2022. Vision Transformers for Action Recognition: A Survey. arXiv:2209.05700.

Vaswani, A.; Ramachandran, P.; Srinivas, A.; Parmar, N.; Hechtman, B.; and Shlens, J. 2021. Scaling Local Self-Attention for Parameter Efficient Visual Backbones. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12889–12899. Los Alamitos, CA, USA: IEEE Computer Society.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wang, S.; Li, B. Z.; Khabsa, M.; Fang, H.; and Ma, H. 2020. Linformer: Self-Attention with Linear Complexity. arXiv:2006.04768.

Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; and Sun, L. 2022. Transformers in Time Series: A Survey. *In the 32nd International Joint Conference on Artificial Intelligence (IJCAI 2023)*.

Wightman, R. 2019. PyTorch Image Models. https://github.com/rwightman/pytorch-image-models.

Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; and Zhang, L. 2021. CvT: Introducing Convolutions to Vision Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 22–31.

Xiong, Y.; Zeng, Z.; Chakraborty, R.; Tan, M.; Fung, G.; Li, Y.; and Singh, V. 2021. Nyströmformer: A Nyström-Based Algorithm for Approximating Self-Attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Xu, P.; Zhu, X.; and Clifton, D. A. 2022. Multimodal Learning with Transformers: A Survey.

Xu, W.; Xu, Y.; Chang, T.; and Tu, Z. 2021a. Co-Scale Conv-Attentional Image Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9981–9990.

Xu, Y.; Wei, H.; Lin, M.; Deng, Y.; Sheng, K.; Zhang, M.; Tang, F.; Dong, W.; Huang, F.; and Xu, C. 2021b. Transformers in computational visual media: A survey. *Computational Visual Media*, 8(1): 33–62.

Yang, J.; Li, C.; Dai, X.; Yuan, L.; and Gao, J. 2022a. Focal Modulation Networks. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

Yang, Y.; Jiao, L.; Liu, X.; Liu, F.; Yang, S.; Feng, Z.; and Tang, X. 2022b. Transformers Meet Visual Learning Understanding: A Comprehensive Review. arXiv:2203.12944.

Yao, T.; Pan, Y.; Li, Y.; Ngo, C.-W.; and Mei, T. 2022. Wave-ViT: Unifying Wavelet and Transformers for Visual Representation Learning. In Avidan, S.; Brostow, G.; Cissé, M.; Farinella, G. M.; and Hassner, T., eds., *Computer Vision – ECCV 2022*, 328–345. Cham: Springer Nature Switzerland.

Yin, H.; Vahdat, A.; Alvarez, J. M.; Mallya, A.; Kautz, J.; and Molchanov, P. 2022. A-ViT: Adaptive Tokens for Efficient Vision Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10809–10818.

You, Y.; Li, J.; Reddi, S.; Hseu, J.; Kumar, S.; Bhojanapalli, S.; Song, X.; Demmel, J.; Keutzer, K.; and Hsieh, C.-J. 2020. Large Batch Optimization for Deep Learning: Training BERT in 76 minutes. In *International Conference on Learning Representations*.

Yu, J.; Wang, Z.; Vasudevan, V.; Yeung, L.; Seyedhosseini, M.; and Wu, Y. 2022. CoCa: Contrastive Captioners are Image-Text Foundation Models. *Transactions on Machine Learning Research*.

Zhai, X.; Kolesnikov, A.; Houlsby, N.; and Beyer, L. 2022. Scaling Vision Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12104–12113.

Zhuang, B.; Liu, J.; Pan, Z.; He, H.; Weng, Y.; and Shen, C. 2023. A Survey on Efficient Training of Transformers. arXiv:2302.01107.

Zuo, S.; Xiao, Y.; Chang, X.; and Wang, X. 2022. Vision transformers for dense prediction: A survey. *Knowledge-Based Systems*, 253: 109552.