Optimal patch partitioning in Vision Transformers: Reassessing the notion of model size superiority

Anonymous submission

Abstract

Recent advances in Vision Transformers (ViT) have demonstrated notable efficacy in large-scale image recognition by splitting 2D images into a fixed quantity of patches, treating each patch as a distinct token. Generally, augmenting the number of tokens used for image representation enhances prediction accuracy, albeit at the expense of heightened computational demands due to the quadratic complexity of these models. Therefore, to strike a judicious balance between accuracy and computational efficiency, conventional practice involves empirically setting the token count to values such as 14×14. This study contends that the optimal token count depends on the inherent characteristics of each individual image. Our empirical investigations show that adapting the patch partitioning to each image "hardness" leads to an operating point in accuracy vs.complexity tradeoff. Consequently, we advocate a dynamic adjustment of the token count based on the unique attributes of each input image. In our experimental study on the ImageNet-1K dataset, we observed a notable phenomenon: a smaller 7×7 Transformer model outperforms a larger 14×14 counterpart, excelling not only in computational efficiency ($FLOPs$) but also in top-1 accuracy. This result challenges conventional assumptions regarding the relationship between model size and performance, prompting a reconsideration of scalability in image recognition tasks.

Introduction

Recent accomplishments of Transformer models within natural language processing (NLP) (Vaswani et al. 2017) have instigated further investigations in computer vision (Dosovitskiy et al. 2020). As a result, there is a growing interest in vision Transformers (ViT) across diverse visual tasks, such as image classification (Dosovitskiy et al. 2020; Jiang et al. 2021), object detection (Liu et al. 2021; Wang et al. 2021a), and semantic segmentation (Zheng et al. 2021; Xie et al. 2021). ViT accomplishes this by splitting a 2D image into a sequence of patches and using linear projection to embed these patches into 1D tokens, facilitating the modeling of extensive dependencies among tokens.

In essence, the performance of a ViT model is closely tied to the number of input tokens (Dosovitskiy et al. 2020; Wang et al. 2021b), which quadratically increases the computational cost of ViT. Lately, there have been efforts to remedy this issue by proposing numerous compression approaches

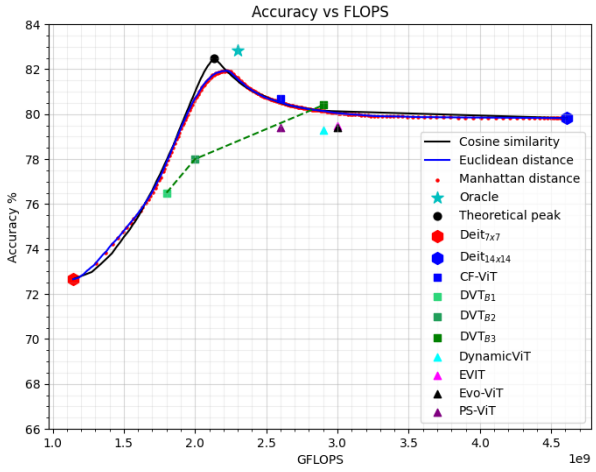
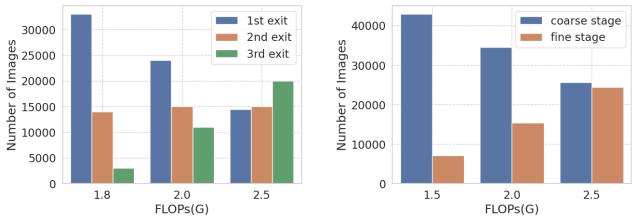


Figure 1: Accuracy vs. complexity diagram of our experiment. Cascading models are depicted with a square and token pruning models with a triangle. The "Oracle" depicted with a star is an upper bound for the theoretical peak, achievable only if we assign the optimal partitioning for all images.

to unearth redundant tokens for ViTs. these include PS-ViT (Tang et al. 2022), which enhances Transformer efficiency through a top-down token pruning paradigm, DynamicViT (Rao et al. 2021) and IA-RED^2 (Pan et al. 2021) that both introduce a lightweight prediction module to estimate the importance score of each token and discard low-score tokens. Additionally, there also exist approaches that prune tokens according to their importance score like EViT (Liang et al. 2022). DGE (Song et al. 2021), in contrast to discarding tokens directly, introduces sparse queries to reduce the output token number.

Finally, approaches like DVT (Wang et al. 2021b) and CF-ViT (Chen et al. 2023) focus on cascading multiple ViTs with increasing number of tokens and then uses an early exiting policy at each stage based on the confidence of the model. If the confidence is higher than a predefined threshold, the model classifies the input sample at the present stage, Conversely, if confidence is bellow the threshold, the input is forwarded to a downstream model that requires a finer partitioning of the image, thereby requiring additional



(a) DVT (Wang et al. 2021b) on DeiT-S (Touvron et al. 2021) backbone.

(b) CF-ViT (Chen et al. 2023) on DeiT-S (Touvron et al. 2021) backbone.

Figure 2: Number of samples exiting at different exits with varying computational budgets for both DVT (Wang et al. 2021b) (left) and CF-ViT. (Chen et al. 2023) (right)

computational resources. However, the primary challenge associated with cascading models lies in the redundancy of sample processing. In simpler terms, "hard" samples are often classified in the final stages, requiring both the computational resources of the last stage and those of the upstream stages which lead to a processing redundancy.

To remedy this limitation, we raised the following questions in our study: What is the percentage of samples classified by late stages? If this percentage is significant, is there a way to get an optimal patch partitioning (i.e. token number) depending on each instance "hardness"?

Both DVT (Wang et al. 2021b) and CF-ViT (Chen et al. 2023) answer the first question and they clearly show that the percentages are high. In figure 2, we have reproduced the experiments for the sake of visualization using their open-source code.

Our work tackles the second question. Typically, there is a common belief that Transformers with greater complexity, indicated by a higher number of tokens, generally exhibit better performance in terms of accuracy compared to less complex ones. Through our experimental study, we came across a counter intuitive but rather interesting observation: When carefully selecting instances to be processed by one of both Transformer stages (i.e. 7×7 or 14×14) based on distance metrics between each model's input and the true class vector, we get theoretical peak performance of 82.50% for 2.2 GFLOPs on DeiT (Touvron et al. 2021) as depicted in figure 1, surpassing the DeiT-S$_{14\times14}$ baseline and state-of-the-art in both accuracy and computational complexity. The main contribution of this paper unfolds as follows:

- Transformers exhibit enhanced efficiency by opting for an adaptive patch partitioning strategy for individual input samples. This leads a theoretical peak performance on ImageNet-1K that outperforms DeiT-S$_{14\times14}$ baseline.

Related work

This section reviews related work about vision transformers and dynamic compression.

Vision Transformers

Transformers (Vaswani et al. 2017; Dosovitskiy et al. 2020) have emerged as a reliable alternative to convolutional networks (CNN) for image recognition and are now competitive on the standard ImageNet benchmark (Deng et al. 2009). DeiT (Touvron et al. 2021) explores ViT's training strategy and suggests a knowledge distillation-based approach, surpassing the performance of ResNet (He et al. 2016). T2T-ViT (Yuan et al. 2021) repeatedly merges adjacent tokens into a single token, aiming to decrease the token length and gather spatial context. LocalViT (Li et al. 2021) incorporates depthwise convolutions to improve the ViTs' ability to model local features.

The main drawback of these approaches is the complexity of the multi-head attention (MHA) that constitutes the elementary building block of Transformers. Indeed, when considering two image embeddings, $(X_1, X_2) \in \mathbb{R}^{N\times d}$, where N represents the number of tokens and d the embedding dimension, the attention module facilitates the exchange of information between them, this process is initiated by generating a query (Q), a key (K), and a value (V) using the following equations:

$$Q = W_Q X_1,$$
$$K = W_K X_2,$$
$$V = W_V X_2,$$

Here, $Q, K, V \in \mathbb{R}^{N\times d}$. $W_Q, W_K, W_V \in \mathbb{R}^{N\times N}$ represent the parameters that the model learns. Subsequently, the process involves message aggregation, achieved by calculating attention scores between the query and key as follows:

$$Attention(Q, K, V) = \text{softmax}\left[\frac{QK^T}{\sqrt{d}}\right]V$$

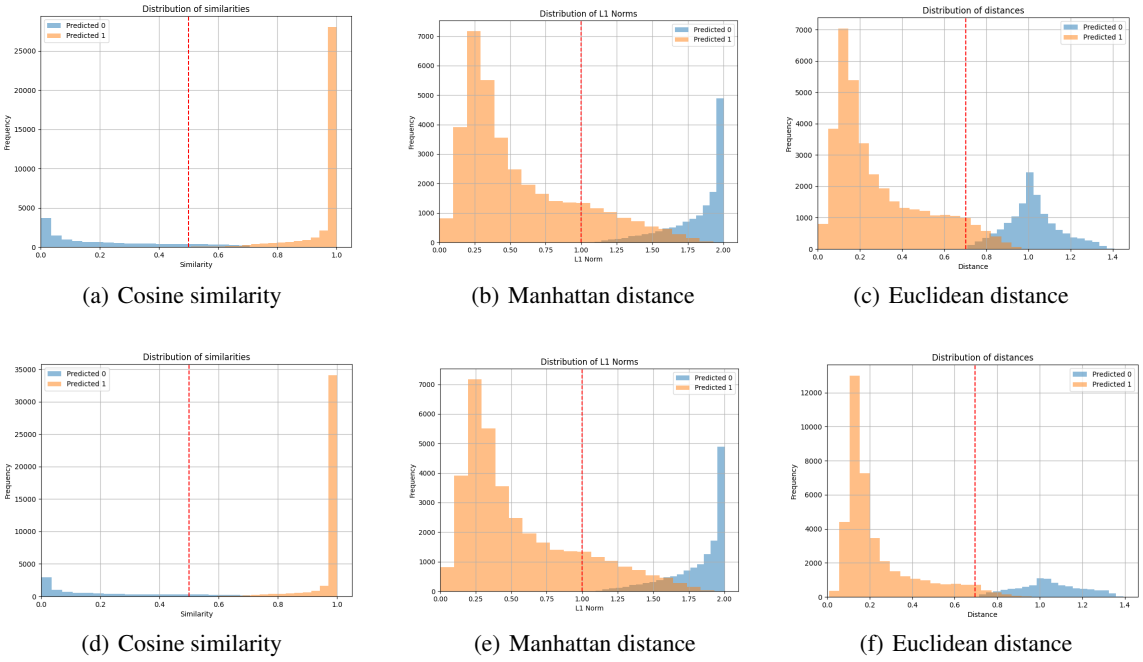
The outlined procedure manifests a computational complexity of $O(d \times N^2)$. This quadratic complexity hinders the integration and the embedding of theses models on edge devices due to high computational demands. Given that most of the works outlined above represent each image with a fixed number of tokens, this leads to allocating resources regardless of wheather input images are "easy" or "hard" to classify. In our work, we advocate for dynamic partitioning based on image "hardness", hence adaptively allocating computational resources according to input sample's "hardness" level.

Dynamic compression

Dynamic compression adjusts the computational graph based on input images. Token pruning methods (Liang et al. 2022; Long et al. 2023) dynamically discard tokens considered unimportant during inference by applying a top-k operation on the classification token [CLS] to select the K tokens that receive the highest attention. In contrast, Evo-ViT (Xu et al. 2022) retains unimportant tokens, albeit with a lower computational budget for updates.

Another family of methods named cascading methods can act on the model's depth by ending inference based on its confidence (Huang et al. 2017; Wang et al. 2021b; Chen et al. 2023).

We show this in figure 2 which depicts the number of samples exiting at different exits with varying computational



(a) Cosine similarity (b) Manhattan distance (c) Euclidean distance

(d) Cosine similarity (e) Manhattan distance (f) Euclidean distance

Figure 3: Distribution of ImageNet-1K 50K validation instances for the three metrics. (a) (b) and (c) are $P_{7\times7}$ predictions and (d) (e) (f) are $P_{14\times14}$ predictions. Orange bars represent the number of images correctly predicted for a given value of the metric and the blue ones are the number of miss-predicted images. the vetical dotted red line represents a confidence threashold on a given metric to better discriminate the instances

budget for both DVT (Wang et al. 2021b) and CF-ViT (Chen et al. 2023). In DVT (Wang et al. 2021b) a three-stage architecture (7×7, 10×10, 14×14) is proposed. The authors explicitly demonstrate that for their best accuracy/budget trade-off (2.5 GFLOPs), 40% of input samples are processed by the third stage. Essentially, all stages redundantly handle 40% of samples, incurring significant computational expense. This redundancy is also depicted in CF-ViT (Chen et al. 2023), featuring a two-stage architecture. Here, the authors reveal that for their best accuracy/budget trade-off (2.5 GFLOPs), 48% of samples are processed in the second stage, indicating that almost half of these samples are redundantly processed by the current and upstream stage, exacerbating computational inefficiency.

While token pruning has shown interesting results in reducing computational complexity, selecting region of interest via token selection reduces context, which has a large effect on accuracy on small backbones with fewer input tokens (Haurum et al. 2023). The main drawback of these approaches is the computational redundancy when processing inputs in late stages.

Methodology

Instance hardness

Since 40% of instances are classified by the third exit for a budget of 2.5 GFLOPs, processing samples in a sequential multi-stage scheme seems inefficient. Instead, We can design an architecture that considers the complexity of each instance and directs it toward the appropriate target model. In other words, we could predict instance's "hardness" before processing it. We made the following hypothesis:

There is a link between instance hardness and the model's confidence

The true class vector in our case refers to the ground truth for every prediction, i.e. the one-hot encoded vector for each sample. The model's confidence refers to the prediction, i.e the softmax probability vector of dimension N_c classes:

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{N_c} e^{z_j}} \quad \text{for } i = 1, 2, \ldots, N_c \tag{1}$$

where z_i is the output vector of the model and N_c the number of classes, hence, the softmax vector is a 1D dimension such as $\sigma(z)_i \in \mathbb{R}^{1\times N_c}$.

We selected three metrics to measure the model's confidence on each individual sample. The Euclidean distance, the manhattan distance and the cosine similarity respectively:

$$D_{Eucl} = \sqrt{\sum_{i=1}^{N_c} (\sigma(z)_i - c_i)^2}$$

$$D_{Manh} = \sum_{i=1}^{N_c} \| \sigma(z)_i - c_i \|$$

$$S = \frac{\sigma(z)_i \cdot c_i}{\|\sigma(z)_i\| \cdot \|c_i\|}$$

Here, $c_i \in \mathbb{R}^{1 \times N_c}$ is the one-hot encoded vector, i.e. the ground truth of each sample. These metrics are the most used in the literature to compute disparities between model's prediction (i.e softmax vector) and the true class vector.

Experiments

We ran an experiments on the small Transformer $P_{7\times7}$ and $P_{14\times14}$[1] in order to analyze the distribution of well classified and misclassified instances (i.e predicted or not) from the validation set relative to their metrics value. For each inference on validation set, we compute the distance between $P_{7\times7}$ prediction's output (i.e., softmax vector) and its corresponding class vector, as well as the distance between $P_{14\times14}$ prediction's output and its corresponding class vector.

These distributions are illustrated in figure 3 for the three metrics. In orange, the well-classified instances and in blue the misclassified ones. We notice that for the three metrics, the closer the distance to center of class, the more correct the predictions are. In other words, most of the well classified predictions are situated on the left side of the distribution for the euclidean distance and Manhattan and on the right side of the distribution for the similarity[2].

If we focus on the euclidean distance and perform another experiment in which we move the vertical red-dotted threshold line from left to right, each time, we infer all the instances that are situated on the left side of the line with the small model ($P_{7\times7}$) and the remaining ones with the large model ($P_{14\times14}$), the key idea is to process the well classified instances with the small model and the remaining ones with the large model. After each inference, we get the accuracy (in %) and complexity (in GFLOPs), we move the vertical line (i.e, threshold) to the right with a step of 0.01 and repeat the experience until we infer for all the distances. Finally, we get an accuracy vs. complexity diagram depicted by figure 1 where we expect an increasing concave curve.

The two extreme points on the graph represent the performance of the small model (extreme left red point) and the large model (extreme right blue point). In between, the performance of the hybrid model according to the threshold, instead of being concave and increasing, it has an interval where its accuracy (i.e theoretical peak) in figure 1 exceeds the accuracy of the large model, which is intriguing. In other words, $P_{7\times7}$ classifies some instances better than $P_{14\times14}$, this results in an increase of accuracy of 2.5% while significantly reducing computational complexity by 52% in FLOPs. The theoretical result also surpasses both DVT (Wang et al. 2021b) in its most competitive budget configuration and CF-ViT (Chen et al. 2023) in accuracy (+2.1% and +1.8% resp.) while achieving a decent computational complexity reduction (12% and 15.3% resp.) in FLOPs.

[1]From now on, the small model will be denoted $P_{7\times7}$ and the large model $P_{14\times14}$, where 7×7 and 14×14 depict the number of patches in input images. P stands for patch

[2]Cosine similarity is inversely proportional to the distance

At last, compared to other state-of-the-art approaches that rely on token pruning as EViT (Liang et al. 2022) or DynamicViT (Rao et al. 2021), we clearly see that the theoretical result also surpasses them in both accuracy and computational complexity.

These results left us with a two questions, what can explain such a interesting behavior? And how can we achieve the theoretical peak performance?

Future work

Since this is a work-in-progress, our forthcoming efforts will focus on addressing the two preceding questions. Next sections will give a first clue on what could explain this behavior and how we can reproduce the theoretical result.

Peak explanation The validation set may contain samples with varying levels of complexity. The $P_{7\times7}$ model might excels in capturing features and patterns in simpler images where a more coarse-grained representation suffices. Indeed, certain patterns or structures within the images might align more favorably with the receptive field of a $P_{7\times7}$ token grid. On the other hand, the $P_{14\times14}$ model, being more fine-grained could struggle to gain a significant advantage in such cases.

There are efforts in the literature to define instance "hardness" and tries to quantify using various metrics (Paiva et al. 2022; Lorena, Paiva, and Prudêncio 2023; Jiang et al. 2022). A next step will be to use these metrics and establish correlations between them and their model's predictions.

Theoretical peak performance To reproduce the theoretical peak performance, we are currently thinking about an approach based on a lightweight "router" model. This model will have to predict the optimal partitioning for each instance before directing the instance to be processed on its corresponding Transformer target. It will be trained on D_{train} such as: $D_{train} = \{image, P_N\}$ where $P_0 = 0$ is the label of $P_{7\times7}$ and $P_1 = 1$ is the label of $P_{14\times14}$. More precisely, the router's task is to pre-process instances and determine which target model should process them based on inherent visual characteristics. This means that for a "hard to classify" instance that requires more computation, the router predicts P_1, in contrast, for an "easy to classify" instance which requires less computation, the router predicts P_0.

Conclusion

Our investigation criticizes redundancies in multi-stage architectures, particularly due to the sequential processing of samples and the associated memory requirements. We addressed an important question concerning instance hardness and its management by models with varying complexities. Notably, our study showcased an interesting phenomenon: when considering $P_{7\times7}$ and $P_{14\times14}$, a theoretical peak performance score outperformed the baseline $P_{14\times14}$ in accuracy while reducing computational complexity by nearly 52%. Since this is a work-in-progress, our next objective is to design an approach for attaining the theoretical score. This involves training a model capable of distinguishing between hard and easy samples, while efficiently utilizing the appropriate level of computation.

References

Chen, M.; Lin, M.; Li, K.; Shen, Y.; Wu, Y.; Chao, F.; and Ji, R. 2023. Cf-vit: A general coarse-to-fine method for vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 7042–7052.

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*, 248–255. Ieee.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Haurum, J. B.; Escalera, S.; Taylor, G. W.; and Moeslund, T. B. 2023. Which Tokens to Use? Investigating Token Reduction in Vision Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 773–783.

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.

Huang, G.; Chen, D.; Li, T.; Wu, F.; Van Der Maaten, L.; and Weinberger, K. Q. 2017. Multi-scale dense networks for resource efficient image classification. *arXiv preprint arXiv:1703.09844*.

Jiang, S.; Zhu, Y.; Liu, C.; Song, X.; Li, X.; and Min, W. 2022. Dataset bias in few-shot image recognition. *IEEE transactions on pattern analysis and machine intelligence*, 45(1): 229–246.

Jiang, Z.-H.; Hou, Q.; Yuan, L.; Zhou, D.; Shi, Y.; Jin, X.; Wang, A.; and Feng, J. 2021. All tokens matter: Token labeling for training better vision transformers. *Advances in neural information processing systems*, 34: 18590–18602.

Li, Y.; Zhang, K.; Cao, J.; Timofte, R.; and Van Gool, L. 2021. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*.

Liang, Y.; GE, C.; Tong, Z.; Song, Y.; Wang, J.; and Xie, P. 2022. EViT: Expediting Vision Transformers via Token Reorganizations. In *International Conference on Learning Representations*.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, 10012–10022.

Long, S.; Zhao, Z.; Pi, J.; Wang, S.; and Wang, J. 2023. Beyond Attentive Tokens: Incorporating Token Importance and Diversity for Efficient Vision Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10334–10343.

Lorena, A. C.; Paiva, P. Y.; and Prudêncio, R. B. 2023. Trusting my predictions: on the value of Instance-Level analysis. *ACM Computing Surveys*.

Paiva, P. Y. A.; Moreno, C. C.; Smith-Miles, K.; Valeriano, M. G.; and Lorena, A. C. 2022. Relating instance hardness to classification performance in a dataset: a visual approach. *Machine Learning*, 111(8): 3085–3123.

Pan, B.; Panda, R.; Jiang, Y.; Wang, Z.; Feris, R.; and Oliva, A. 2021. IA-RED^2: Interpretability-Aware Redundancy Reduction for Vision Transformers. *Advances in Neural Information Processing Systems*, 34: 24898–24911.

Rao, Y.; Zhao, W.; Liu, B.; Lu, J.; Zhou, J.; and Hsieh, C.-J. 2021. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34: 13937–13949.

Song, L.; Zhang, S.; Liu, S.; Li, Z.; He, X.; Sun, H.; Sun, J.; and Zheng, N. 2021. Dynamic grained encoder for vision transformers. *Advances in Neural Information Processing Systems*, 34: 5770–5783.

Tang, Y.; Han, K.; Wang, Y.; Xu, C.; Guo, J.; Xu, C.; and Tao, D. 2022. Patch slimming for efficient vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12165–12174.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, 10347–10357. PMLR.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; and Shao, L. 2021a. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, 568–578.

Wang, Y.; Huang, R.; Song, S.; Huang, Z.; and Huang, G. 2021b. Not all images are worth 16x16 words: Dynamic transformers for efficient image recognition. *Advances in Neural Information Processing Systems*, 34: 11960–11973.

Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34: 12077–12090.

Xu, Y.; Zhang, Z.; Zhang, M.; Sheng, K.; Li, K.; Dong, W.; Zhang, L.; Xu, C.; and Sun, X. 2022. Evo-vit: Slow-fast token evolution for dynamic vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 2964–2972.

Yuan, L.; Chen, Y.; Wang, T.; Yu, W.; Shi, Y.; Jiang, Z.-H.; Tay, F. E.; Feng, J.; and Yan, S. 2021. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF international conference on computer vision*, 558–567.

Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P. H.; et al. 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6881–6890.