Which Tokens to Use? Investigating Token Reduction in Vision Transformers

Joakim Bruslund Haurum[1] Sergio Escalera[2,1] Graham W. Taylor[3] Thomas B. Moeslund[1]

[1] Visual Analysis and Perception (VAP) Laboratory, Aalborg University & Pioneer Centre for AI, Denmark

[2] Universitat de Barcelona & Computer Vision Center, Spain [3] University of Guelph & Vector Institute for AI, Canada

joha@create.aau.dk, sescalera@ub.edu, gwtaylor@uoguelph.ca, tbm@create.aau.dk

Abstract

Since the introduction of the Vision Transformer (ViT), researchers have sought to make ViTs more efficient by removing redundant information in the processed tokens. While different methods have been explored to achieve this goal, we still lack understanding of the resulting reduction patterns and how those patterns differ across token reduction methods and datasets. To close this gap, we set out to understand the reduction patterns of 10 different token reduction methods using four image classification datasets. By systematically comparing these methods on the different classification tasks, we find that the Top-K pruning method is a surprisingly strong baseline. Through in-depth analysis of the different methods, we determine that: the reduction patterns are generally not consistent when varying the capacity of the backbone model, the reduction patterns of pruning-based methods significantly differ from fixed radial patterns, and the reduction patterns of pruning-based methods are correlated across classification datasets. Finally we report that the similarity of reduction patterns is a moderate-to-strong proxy for model performance. Project page at https://vap.aau.dk/tokens.

1. Introduction

The Vision Transformer (ViT) [11] has in record time seen wide spread adoption within computer vision, ousting Convolutional Neural Networks (CNNs). In order to better understand how ViTs function, prior works have investigated whether ViTs process data in a similar way as CNNs [40], and how different types of supervision affect ViT training [53]. In this work we investigate the use of *token reduction methods*, which leverage the fact that ViTs can accommodate variable input sequence lengths. These methods aim to make ViTs more efficient by removing redundant tokens and thereby reduce the computational cost of the self-attention operation, which scales quadratically with the number of tokens [3, 14, 41].

However, with some limited exceptions, little has been done to gain deeper insights into how the token reduction process differs across methods or depends on hyperparameters such as backbone capacity or the number of tokens to be kept. Furthermore, the analysis of methods have primarily been constricted to the ImageNet dataset, and is rarely done with structured comparisons to other methods. Consequently, the community does not have a good understanding of how the methods differ from one another. We set out to rectify this by conducting a systematic comparison of 10 recently proposed methods, leveraging thorough experiments to elucidate the inner workings of reduction processes. In this work we make the following contributions:

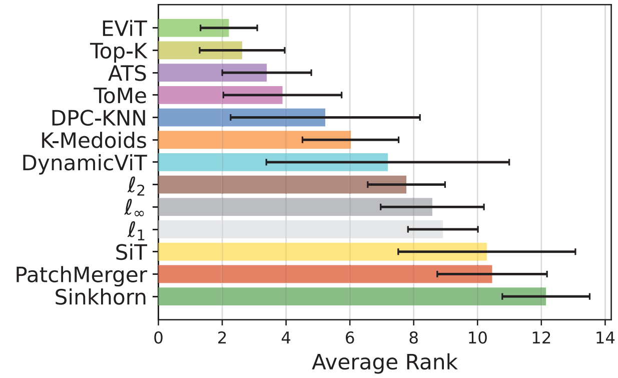


Figure 1: **Average method rank.** The average rank of each tested method plotted with ± 1 standard deviation. The Top-K pruning method and its extension, EViT, are found to be the best performing methods.

- We conduct the first systematic comparison and analysis of 10 state-of-the-art token reduction methods across four image classification datasets, trained using a single codebase and consistent training protocol.
- We find that the Top-K and EViT methods are strong baselines across all datasets.
- Extensive experiments providing deeper insights into the core mechanisms of the token reduction methods.
- We find that the similarity in reduction patterns is a moderate-to-strong proxy for model performance.

2. Related Works

Efficient Vision Transformers. Since the introduction of the Transformer [52] and Vision Transformer [11] a large number of methods have been proposed to make the Transformer model more efficient. Within the computer vision domain several method paradigms have been investigated such as model pruning [5, 7, 37, 47, 62] and quantization [28, 32], structured token downsampling inspired by the pooling layers in CNNs [15, 18, 34], randomly dropping patches [1, 33], part selection modules [17, 20, 21, 54, 57], and dynamic resizing of input patches [2, 6, 30, 55, 56, 63, 65]. Additionally, the Transformer model uniquely allows for variable input sequence lengths, enabling a new type of method, called token reduction, which operate directly on the token sequence. These methods are the focus of this work.

Token Reduction Methods. The prior work on sparsification of the input token sequence can be divided into two primary paradigms: token pruning [12, 14, 23, 27, 29, 35, 39, 41, 49, 60, 61] and token merging [3, 16, 36, 42, 46, 58, 59, 64, 66]. Pruning-based methods aim to reduce the token sequence by removing tokens, whereas merging-based methods reduce the token sequence by combining tokens.

Pruning-based methods can be split into dynamic and static keep rate methods, depending on whether the method can dynamically choose how many tokens to prune. Static keep rate pruning methods select a predetermined number of tokens to keep by using the attention scores between the spatial tokens and the global class (CLS) token [29, 35] or by predicting per-token keep probabilities [23, 41]. In order to not completely discard the information in the pruned tokens, which can contain contextual information regarding *e.g.* location and background, several methods propose merging the pruned tokens into a single token [23, 29, 35, 60]. On the other hand, dynamic keep rate pruning methods select an adaptive number of tokens to keep by using either sampling methods [14, 61], reinforcement learning [39], or alternating training schemes [27, 49].

Similarly, merging-based methods can be split into hard-merging [3, 36, 59, 64] and soft-merging [16, 42, 58, 66], depending on whether the merging operation requires the token assignment to be mutually exclusive. Hard-merging methods typically use commonly known clustering methods such as K-Means [36], K-Medoids [36], and Density-Peak Clustering with K-Nearest Neighbours (DPC-KNN) [64]. Other hard-merging based methods have used bipartite-matching of tokens [3] and cross-attention between spatial tokens and learnable cluster centers, called queries [59]. The soft-merging based approaches have primarily consisted of soft-clustering methods, which lead to a convex combination of spatial tokens derived from similarity with queries or the spatial tokens themselves [16, 42, 66].

To summarize, while many different token reduction methods have been proposed, scant attention has been given to comparing the methods in a systematic way, nor trying to better understand how the reduction process and reduction patterns (*i.e.* constructed clusters and pruned tokens) are affected by the choice of reduction method, datasets, and model settings. In this work we aim to rectify this by conducting a thorough systematic comparison and in-depth analysis of contemporary token reduction methods.

3. Experimental Design

In order to compare the different token reduction methods fairly, we select representative methods from each reduction method paradigm; see Section 3.1. The methods are chosen based on two criteria: 1) the selected methods should cover the key trends within each paradigm, and 2) the methods should be inserted into the backbone with minimal adjustments to the training loop.

Several ways of incorporating the token reduction operation have previously been used, ranging from a single reduction stage in the middle of the ViT [27, 42] to after each stage in the ViT [3, 14, 36, 61]. However, the most common approach is to apply the token reduction operation at three predefined stages dividing the ViT into four sections of equal size [23, 29, 35, 39, 41, 59, 64, 66]. This is the setting which we will follow. At each stage a ratio of the tokens, r, are kept for further processing, where $r \in \{0.25, 0.50, 0.70, 0.90\}$ is denoted the *keep rate*.

3.1. Methods

To ensure diversity of methods we select three representative and top-performing methods from each paradigm. For the Dynamic Keep Rate Pruning paradigm, however, we only select one as the other methods either did not converge during training with the training settings described in Section 3.3 (A-ViT [61]), or required substantial modifications to the training loop (IA-RED^2 [39], DPS-ViT [49], and SaiT [27]). Each method is described in Section 3.1.2–3.1.5. We also introduce a set of baseline pruning methods with a fixed image-centered radial pattern, see Section 3.1.1. These are based on observations made by Yin *et al*. [61] and Rao *et al*. [41] who found that the averaged reduction patterns display a radial pattern focused on the image center. All methods are implemented in a single codebase based on official model implementations when possible.

3.1.1 Fixed Pattern Pruning Baseline

We introduce a set of baseline methods with a fixed reduction pattern based on the distance of each token to the center of the image, measured using the ℓ_p-norm. Specifically, we consider fixed patterns created by using the ℓ_1, ℓ_2, and ℓ_∞ norms. In order to create the fixed patterns such that only r^s tokens are kept at reduction stages $s \in \{1, 2, 3\}$, we prune tokens based on their distance to the image center, setting the threshold such that the absolute difference between the kept tokens and Pr^s is minimized, where P is the initial amount of spatial tokens.

3.1.2 Static Keep Rate Pruning

Top-K is a commonly used pruning baseline, where the attention between the P spatial tokens and the CLS token is used. At reduction stage s the method simply selects the K_s most attended to tokens, where $K_s = Pr^s$.

EViT [29] extends Top-K pruning by creating a single "fused" token at each stage s. The fused token is computed by averaging the $P_s - K_s$ pruned tokens weighted by their CLS token attention scores, where P_s is the number of tokens at stage s before the reduction is applied.

DynamicViT [41] prunes the tokens by constructing a binary decision mask $\mathbf{D}_s$ based on keep probabilities produced by a small prediction module. The Gumbel-Softmax trick [22] is used to ensure training is differentiable, while during inference the K_s most probable tokens are kept. An extra loss is needed to ensure that $\mathbf{D}_s$ only keeps K_s tokens.

3.1.3 Dynamic Keep Rate Pruning

ATS [14] is a sampling-based pruning method which selects a variable amount of tokens at each reduction stage. This is achieved by applying the inverse transform sampling (ITS) on the cumulative distribution function (CDF) of the CLS token attention scores and uniformly sampling the CDF Pr^s times. In case a token is assigned a high attention score by the CLS token it may be sampled multiple times by the ITS operation, in which case only a single copy is kept. Thereby, ATS can sample fewer than Pr^s tokens at stage s.

3.1.4 Hard-Merging

ToMe [3] is a recent token merging method, where the set of tokens are split into a bipartite graph with equal sized partitions A and B, where edges are constructed by drawing a single edge for each node in A to the node in B with the highest cosine similarity. The $P_s(1-r^s)$ highest valued edges are kept and connected nodes are merged by averaging the token features, followed by combining set A and B again. It should be noted that the ToMe method is constrained such that r cannot be below 50% as nodes in set A are only allowed a single edge. In order to align the nomenclature with clustering methods, we denote the output of the ToMe method as cluster centers.

K-Medoids [36] is an iterative hard-clustering baseline where the Pr^s cluster centers are set to be the cluster element which minimizes the ℓ_2 distance to all other elements in the cluster. The method iteratively updates the clusters by assigning tokens to the cluster with the closest cluster center. We initialize the cluster centers based on the CLS token attention scores as proposed by Marin *et al*. [36].

DPC-KNN [13] is a two-step clustering approach, where first the density of each token is computed based on the distance to the k-nearest neighbours, followed by determining the minimum distance to a point with higher density. The two measures are combined and the cluster centers are defined to be the Pr^s tokens with the highest combined scores. The final cluster representation is obtained by averaging the elements assigned to the cluster. Zeng *et al*. [64] proposed to add a small linear layer which predicts the importance for each token, making the cluster representation a weighted average of the cluster elements.

3.1.5 Soft-Merging

SiT [66] is a recent soft-clustering method, where a small network predicts an assignment matrix $\mathbf{A}_s$, representing a convex combination of the Pr^{s-1} input tokens to construct Pr^s clusters. Specifically $\mathbf{X}_s = \mathbf{X}_{s-1}\mathbf{A}_s$, where $\sum_{i=1}^{Pr^s} \mathbf{A}_s[i,j] = 1$ for $j = 1, 2, \dots, Pr^{s-1}$ and $\mathbf{X}$ is the token feature representations.

Sinkhorn [16] is a query-based clustering method, where unlike in SiT, the cluster centers, called queries, are randomly initialized learnable vectors. The assignment matrix is constructed by applying the Sinkhorn-Knopp algorithm on the cosine similarities between the tokens and queries.

PatchMerger [42] is a query-based clustering method, similar to Sinkhorn, where the assignment matrix is constructed by calculating the dot product between the queries and tokens. This is followed by a softmax operation to ensure the assignment matrix results in a convex combination.

3.2. Datasets

Previously, methods have only been tested on the ImageNet [45] classification dataset and primarily against the backbone model with no token reduction methods. In order to gain diverse insights into the methods, we analyse and compare the different token reduction methods using four distinct classification datasets: ImageNet, NABirds [51], COCO [31], and NUS-WIDE [8].

ImageNet and NABirds are used to evaluate the commonly used multi-class classification task. ImageNet is the most commonly used vision classification dataset, consisting of 1000 diverse classes across 1.2 million images. In contrast, the NABirds dataset represents a much more fine-grained classification task, consisting of 48,000 images and 555 bird classes. We also compare methods on the COCO and NUS-WIDE multi-label classification tasks, representing the case where more than one class of interest can be present simultaneously. COCO and NUS-WIDE consists of 80–81 classes of common object and animals across 122k to 220k images, respectively. In contrast to ImageNet, where the object of interest is often in the center of the image, the NABirds, COCO, and NUS-WIDE represent scenarios where the distinguishing attributes are not necessarily in the center of the image, or there may be more than one object of interest, respectively. Example images of each dataset are shown in Fig. 2. Classification performance on ImageNet and NABirds is measured using the standard Top-1 accuracy metric [17, 45], and for COCO and NUS-WIDE we report the mean Average Precision score (mAP) [43].

3.3. Training Details

All methods are inserted into an DeiT backbone pretained on ImageNet without distillation [50] at the 4th, 7th, and 10th transformer blocks. The DeiT backbone was chosen as it is the most commonly used throughout the token reduction literature. We consider both the Tiny, Small, and Base DeiT backbone capacities, denoted DeiT-$\{\mathrm{T}, \mathrm{S}, \mathrm{B}\}$, respectively.

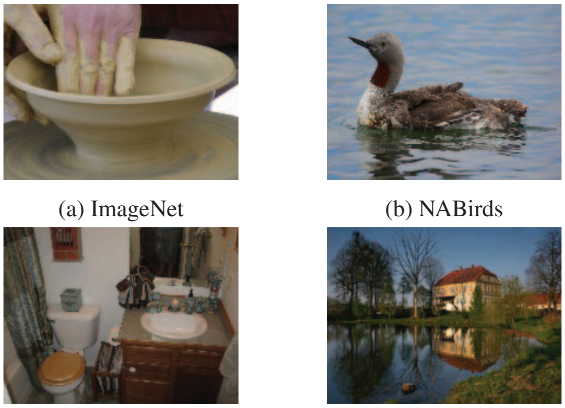
For all methods we based our hyperparameter settings on those presented by Rao *et al*. [41]. A hyperparameter search over the learning rate warmup period, backbone learning rate scaler, and backbone freeze period was initially conducted on the ImageNet dataset using the DeiT-S backbone, training for 30 epochs. The best performing setting for each r was used for training the DeiT-T and DeiT-B variants. It should be noted that for the DynamicViT and SiT methods we do not include the distillation losses used in the original papers, as we find that the effect is minimal and instead choose to keep the training procedure consistent.

For NABirds, COCO, and NUS-WIDE, we fine-tuned the DeiT-S baseline in a similar manner, and for each token reduction method compared the ImageNet hyperparameter setting and fine-tuned DeiT-S hyperparameters. The best set of hyperparameters was used to train the DeiT-T and DeiT-B variants. The NABirds models were trained for 50 epochs with minimal augmentation and no label smoothing, following the guidelines from He *et al*. [17]. COCO and NUS-WIDE models were trained for 40 epochs with the Asymmetric Loss [43] following the multi-label classification guidelines from Ben-Baruch *et al*. [43]. The specific hyperparameter values can be found in the supplementary materials (supp. mat.).

Lastly, it should be noted that for all datasets we trained the models at a resolution of 224×224. This is non-standard for the NABirds, COCO, and NUS-WIDE datasets (normally 448×448). This choice was made to keep reduction patterns comparable, and because the aim was not to push the state-of-the-art in accuracy, but rather to train a set of models from which we can gain deeper insights.

4. Results

We report performance on the four image classification datasets considered with the DeiT-S backbone results in Table 1, results for the DeiT-T and DeiT-B backbones in the supp. mat., and the average rank of the methods in Figure 1. Across all backbone capacities we note two trends: 1) pruning-based methods with learned reduction patterns are consistently among the top-3 methods across all datasets and 2) soft-clustering methods are consistently among the bottom three methods across all datasets.


(a) ImageNet

(b) NABirds

(c) COCO

(d) NUS-WIDE

Figure 2: **Dataset examples.** Randomly selected images from the four considered image classification datasets.

We also find that with the DeiT-T and DeiT-S backbones the hard-merging methods ToMe and DPC-KNN regularly outperform all other methods, especially on the ImageNet dataset and when only 25-50% of the tokens are kept. However, with the DeiT-B backbone, we observe that the pruning-based methods with learned reduction patterns outperform even the hard-merging methods. Looking closer into the pruning-based methods we observe that the fixed-pattern ℓ_p methods are competitive when 90% of tokens are kept, but at lower keep rates the performance drops significantly. For the learned approaches, we find that the DynamicViT method is the most unstable of the tested methods, often being in the bottom three methods when the keep rate is lower than 90%. Similarly, we find that the performance of the ATS method is very dataset dependent, with great performance on the challenging NUS-WIDE dataset, but average performance on all other datasets. It should be noted that the ATS method manages to do so while on average using 50-90 and 10-30 fewer tokens than the other methods when the keep rate is set to 90% and 70%, respectively. This is discussed in the supp. mat.

Comparatively, with a keep rate of 50-90% the Top-K method is the best performing method 36% of the time and in the top-3 methods 83% of the time. This contradicts previous results [14], and indicates that the Top-K method is a very strong baseline. However, at a keep rate of 25% we find that the fused tokens in the EViT method can lead to an improvement of up to 2 percentage points over the Top-K

Table 1: **Performance of Token Reduction methods with DeiT-S backbone.** Model performance is measured across varying keep rates, r, denoted in percentage of tokens kept at each reduction stage. Scores exceeding the DeiT baseline are noted in **bold**, measured in Top-1 accuracy for ImageNet & NABirds and mean Average Precision for COCO & NUS-WIDE. The three best performing methods per keep rate are denoted in descending order with red, orange, and yellow, respectively. Similarly, the three worst performing methods are denoted in descending order with light blue, blue, and dark blue. Results with the DeiT-B and DeiT-T backbones are available in the supp. mat.

	ImageNet				NABirds				COCO				NUS-WIDE			
DeiT-S	79.85				80.57				78.11				63.23			
r (%)	25	50	70	90	25	50	70	90	25	50	70	90	25	50	70	90
ℓ_1	70.05	74.47	77.25	79.17	62.90	70.52	77.10	80.08	61.28	69.49	74.31	77.09	54.44	59.60	61.98	62.91
ℓ_2	70.54	74.86	77.41	79.27	64.28	72.09	77.53	80.11	62.23	70.30	74.66	77.19	55.31	60.22	62.07	62.71
ℓ_∞	70.58	74.03	77.48	79.23	63.36	70.19	77.23	79.96	61.50	69.11	74.73	77.27	55.10	59.34	62.11	62.77
Top-K	72.91	77.82	79.22	**79.87**	76.28	80.38	**80.70**	**80.60**	70.14	75.84	77.50	78.09	59.32	61.98	62.69	**63.26**
EViT	74.17	78.08	79.30	**79.87**	76.74	80.28	**80.73**	**80.64**	71.28	75.78	77.50	78.07	59.69	61.89	62.67	**63.25**
DynamicViT	60.32	77.84	79.17	79.79	70.60	**80.62**	**80.77**	**80.84**	39.18	69.02	75.43	77.69	39.20	57.83	61.96	63.16
ATS	72.95	77.86	79.09	79.63	73.46	78.89	80.36	80.55	70.13	75.66	77.23	77.83	60.20	62.35	62.93	63.18
ToMe	-	78.29	79.63	**79.92**	-	74.99	80.05	**80.68**	-	74.99	77.36	77.88	-	61.51	62.50	62.89
K-Medoids	68.94	76.44	78.74	79.73	65.28	76.95	79.75	80.46	66.26	74.15	76.76	77.94	57.78	61.48	62.47	63.12
DPC-KNN	75.01	77.95	78.85	79.54	68.77	74.14	76.70	78.88	72.15	75.70	77.06	77.74	60.78	62.11	62.67	62.93
SiT	74.65	77.16	77.52	77.71	62.82	62.02	60.72	58.50	57.65	57.33	57.11	57.13	57.95	58.84	59.29	59.59
PatchMerger	69.44	74.17	75.80	76.75	47.26	61.34	65.45	68.24	62.24	68.09	70.75	72.12	55.82	59.27	60.46	61.20
Sinkhorn	64.26	64.07	64.02	64.09	48.89	50.19	51.46	51.22	56.93	56.68	56.85	56.65	50.59	50.67	50.63	50.21

method. This is also evident in Figure 1, where on average the EViT and Top-K methods are the two best ranked methods. Lastly, we note that when 90% of tokens are kept, the Top-K, EViT, DynamicViT, ATS, and ToMe methods outperform the DeiT baselines by up to 0.5 percentage points.

5. In-Depth Analysis of Reduction Patterns

In order to gain deeper insights into the token reduction process, we pose a set of research questions dedicated to uncovering the underlying core mechanisms of the investigated methods. We calculate the defined metrics per dataset and aggregate across all datasets, unless otherwise noted.

Per-dataset results and examples of token reduction patterns can be found in the supp. mat.

5.1. Are Reduction Patterns Consistent when Varying the Keep Rate r?

A common assumption is that the token reduction methods will select tokens from the most representative regions of the image [41]. Assuming this to be true, one would expect that the reduction patterns are consistent (*i.e.* the same set of tokens are merged or pruned) when: 1) reducing the keep rate r and 2) when varying the backbone capacity (see Section 5.2).

In order to evaluate whether the reduction patterns are consistent under varying keep rates, we consider reduction patterns M_1 and M_2 from models trained with keep rates r_1 and r_2, respectively, where $r_1 \neq r_2$. We only compare within the same reduction method and backbone capacity.

For pruning-based methods we evaluate using the Intersection over Area (IoA) between the reduction patterns, *i.e.* how large a ratio of the tokens in M_2 are present in M_1, assuming $r_1 > r_2$. For merging-based methods we evaluate using the Homogeneity of the constructed clusters [44]. Homogeneity is a measure of how consistent the class assignment is within each cluster, *i.e.* whether the elements of each cluster in M_1 are assigned to the same clusters in M_2, assuming $r_1 > r_2$. Further details on IoA and Homogeneity can be found in the supp. mat.

For the hard-clustering methods DPC-KNN and K-Medoids we evaluate using IoA in addition to Homogeneity, by treating the cluster centers as kept tokens. For evaluation of soft-merging methods, we define M by assigning each token to the cluster with the highest assignment score.

We plot our findings in Figure 3. First we find that when lowering the keep rates, the IoA of Top-K, EViT, and DynamicViT (*i.e.* the fixed keep rate pruning methods) are consistently high. However, for the hard-clustering methods and the dynamic keep rate ATS we observe that the IoA quickly drops across all reduction stages, towards the lower bound IoA values, indicating the extracted reduction patterns differ significantly. Secondly, we find that the Homogeneity of the hard-merging methods is consistently high, while it is significantly lower for the soft-merging methods.

From this we can conclude that pruning-based methods, with the exception of ATS, produce consistent reduction patterns when varying r. Similarly, we find that the hard-merging methods select consistent clusters, but with incon-

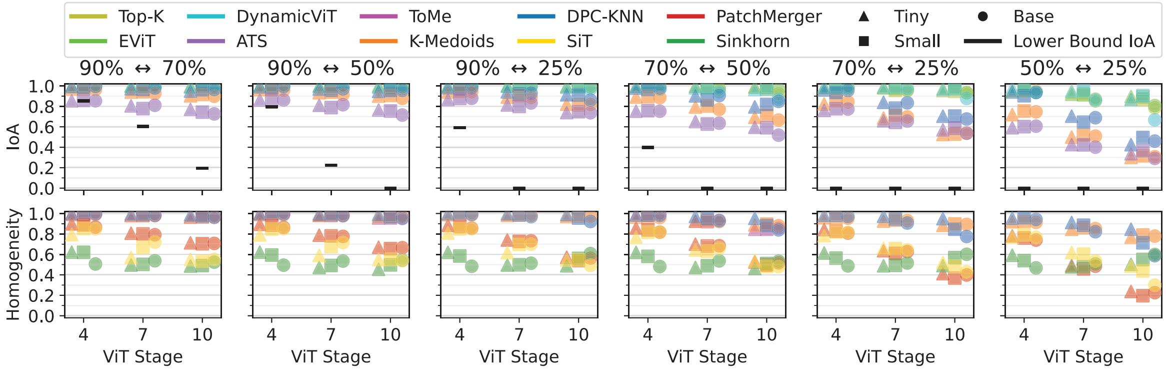


Figure 3: **Effect of varying** r**.** We quantify how similar the reduction patterns are when just the keep rate r is varied. This is quantified with the Intersection over Area (IoA) and Homogeneity for pruning- and merging-methods, respectively. For the IoA metric we can derive the lower bound IoA given different keep rate values; see supp. mat. Note the overlap in pruning methods: Top-K, EViT, and DynamicVit, and merging methods: ToMe, DPC-KNN, and K-Medoids.

sistent cluster centers, while soft-merging methods produce inconsistent clusters when varying r.

5.2. Are Reduction Patterns Consistent when Varying Model Capacity?

In order to evaluate whether the reduction patterns are affected by the backbone model capacity, we consider reduction patterns M_1 and M_2 from the same token reduction method trained with $r_1 = r_2$ but varying backbone capacity.

For pruning-based methods we evaluate using Intersection over Union (IoU) to gauge the similarity of M_1 and M_2. For merging-based methods we evaluate using the Normalized Mutual Information (NMI) [48]. Further details on IoU and NMI can be found in the supp. mat.

As seen in Figure 4, we find that for pruning-based methods the similarity of the reduction patterns is very low for all keep rates. Similar to the observations made in Section 5.1 we observe that the IoU of the DPC-KNN, K-Medoids, and ATS methods is especially low. This indicates that the reduction patterns for pruning-based methods are inconsistent as the backbone model capacity is varied. However, for the hard-merging methods we observed that the clusters are consistent across backbone capacities at all reduction stages when $r > 25\%$. The same can be observed for the soft-merging based PatchMerger method when $r > 50\%$.

From this we can conclude that the reduction patterns of pruning-based methods are inconsistent when varying the backbone capacity. For merging-based methods we find that the ToMe, DPC-KNN, and K-Medoids methods are consistent as long as $r > 25\%$, while PatchMerger is consistent for $r > 50\%$. We can again conclude that the hard-merging methods select consistent clusters, but with varying cluster centers, while soft-merging methods produce inconsistent clusters, as was observed in Section 5.1.

5.3. Do Reduction Patterns Differ Across Datasets?

Little is known about the behaviour of the reduction patterns across different image datasets. One open question is whether there are strong commonalities in the reduction patterns from different datasets, or whether the reduction patterns differ across datasets. In order to do such a comparison, we have to consider the global trends, as per-example comparisons cannot be made. We denote the dataset averaged reduction pattern as $\bar{M}$, which is obtained by computing how many ViT stages each token is passed through, averaged over the validation data splits.

We evaluate the similarity of the averaged reduction patterns across datasets, $\bar{M}_1$ and $\bar{M}_2$, by considering reduction patterns from the same token reduction method trained with keep rate r and backbone capacity, but obtained from different datasets. In order to quantify the similarity we draw inspiration from the saliency detection field, specifically the analysis of different metrics by Bylinskii *et al*. [4]. We use the common Pearson's correlation coefficient, and report results with other common saliency metrics in the supp. mat.

As seen in Figure 5 and following the rule of thumb guidelines by Hinkle *et al*. [19], we find a moderate-to-high correlation of the averaged reduction patterns for nearly all methods across all datasets and keep rates. The exceptions are the DPC-KNN, K-Medoids, and DynamicViT methods, which are found to have spurious lower (but still positive) correlation scores for several dataset pairs, indicating the averaged reduction patterns are less consistent. The lowest correlation scores are obtained by the DPC-KNN method, though this may be attributed to the inconsistent cluster centers as described in Section 5.1-5.2. However, it is not intuitive that the reduction patterns are highly correlated across datasets, as one would expect that due to the significant differences across the investigated datasets imposed by the

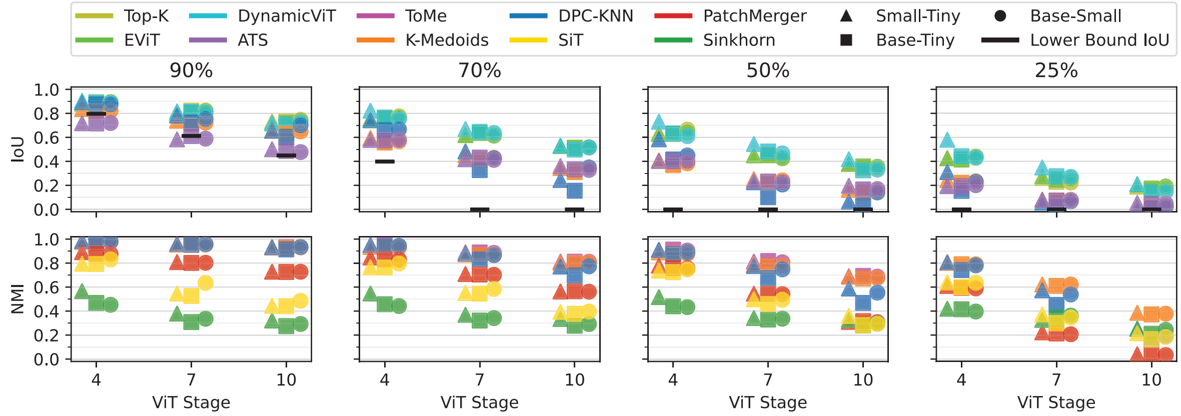


Figure 4: **Effect of varying the backbone capacity.** We quantify similarity of the reduction patterns when the backbone is varied. This is measured with the Intersection over Union (IoU) and Normalized Mutual Information (NMI) for pruning and merging methods, respectively. For the IoU metric we can derive the lower bound IoU given different keep rate values; see supp. mat. Note the IoU for the ATS method can be lower than the lower bound IoU, as it is a dynamic keep rate method.

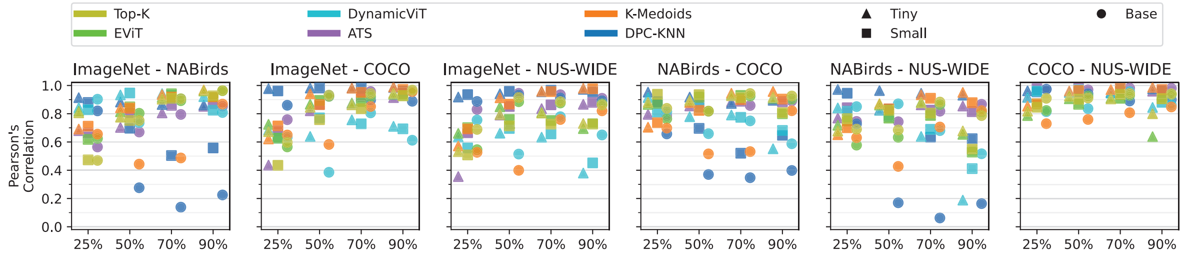


Figure 5: **Cross-Dataset reduction pattern similarity.** We quantify how similar the reduction patterns are across datasets for pruning-based methods, by measuring the correlation between the dataset averaged reduction patterns.

difference in type of classification task and its granularity. Nonetheless, the results indicate that on average the different token positions are used equally often across datasets. This might be due to biases in the image capturing process *e.g.* the sky is always in the top half of the image and the object of interest is in the lower half and center of the image (as seen in Figure 2). We conclude that in general the reduction patterns do not differ significantly across datasets.

5.4. Do Pruning-based Reduction Patterns Differ from Fixed Patterns?

As discovered in prior work [41, 61], when averaging reduction patterns across a dataset the tokens near the image center are kept for longer, resembling a radial selection function. Therefore, it is reasonable to question how similar the per-example reduction patterns are to the fixed patterns from the ℓ_p methods. If they are similar, one could in principle do away with learning the adaptive reduction patterns.

We find that all learned pruning-based reduction patterns have a very low IoU with the fixed ℓ_p patterns at all reduction stages, shown in detail in the supp. mat. As the ℓ_p patterns gradually focus on the tokens close to the image center, this indicates that the learned reduction patterns are not focused on the center. Instead the learned reduction patterns use information from across the entire image at all stages. We can therefore conclude that the learned pruning-based reduction patterns differ from fixed radial patterns.

5.5. Are Reduction Patterns Good Proxies for Model Performance?

A key practical question is whether a pair of reduction patterns can be used to predict the difference in model performance. This is investigated by determining the correlation between $f(A) - f(B)$ and $d(M_A, M_B)$, where f is a performance measure (*i.e.* accuracy or mAP), d is a similarity measure, A and B are two models, and M_A and M_B are their reduction patterns. We use the Spearman's ranked correlation coefficient to measure the correlation. This approach was originally proposed by Ding *et al.* [10]. We set d to be IoU and NMI for pruning- and merging-based methods, respectively, and constrain A to be the Top-K and K-Medoids models as they are both strong baselines. Addi-

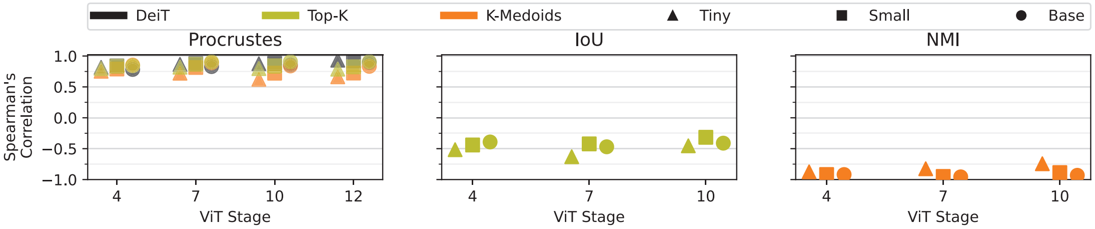


Figure 6: **Reduction patterns as performance proxies.** We determine whether reduction pattern similarity and feature alignment are good proxies for difference in model performance, by measuring the Spearman's ranked correlation between difference in performance and the orthogonal Procrustes distance, IoU, and NMI. Please note that Procrustes is a distance measure, whereas IoU and NMI are similarity measures.

tionally, we measure the feature alignment between A and B by defining d to be the orthogonal Procrustes distance, which Ding *et al*. [10] found to be a better metric of feature alignment than the typically used Centered Kernel Alignment [24] and Projection-Corrected Canonical Correlation Analysis [38]. For the feature alignment test we allow A to be Top-K, K-Medoids, and DeiT, and calculate the alignment using the CLS token after the three reduction stages and the final ViT stage.

We find that for all model capacities the orthogonal Procrustes distance and NMI are highly correlated with the difference in model performance, while the IoU metric is moderately correlated, see Figure 6. The fact that NMI is a better proxy than IoU indicates that the merging-based methods are more sensitive to the construction of the clusters, whereas pruning-based methods are less sensitive to the selection of specific tokens.

Lastly, the reason the Procrustes distance is a good proxy may be grounded in the fact all tested methods use a pretrained DeiT backbone. Therefore, as long as the CLS token does not change during training of the token reduction method, feature alignment should be a good proxy. This has previously been the motivation for distillation losses [41, 66]. However, all tested methods were trained without such losses, indicating that the well-performing models have inherently retained high CLS token alignment.

6. Limitations

We deliberately set certain limitations in order to keep the experiment complexity manageable. First of all, we only consider the image classification task where we extend the analysis from just ImageNet to three additional datasets. Therefore, extending our analysis to additional tasks such as video classification and action recognition is seen as out of scope and we leave this for the future. Secondly, we restricted our training scheme to only consider an ImageNet pre-trained backbone. This is common practice in the literature. Training from scratch on datasets such as ImageNet, OpenImages [26], or Visual Genome [25] would have been prohibitive, and we consider the fine-tuning setting to be more realistic when generalizing to datasets other than ImageNet. Thirdly, we do not investigate how interpretable or robust the different token reduction methods are, though this would be of great interest in the future. Lastly, while the main motivation for the token reduction methods has been to make ViTs more efficient, this work does not evaluate the efficiency of the tested methods. This is a deliberate choice as this work focuses on establishing a systematic comparison of methods with regards to classification performance, as well as gaining deeper insights into the core mechanisms of the tested methods. Furthermore, efficiency is not a simple thing to measure due to confounding factors such as hardware, implementation, and infrastructure as discussed by Dehghani *et al*. [9].

7. Conclusion

In this work we presented the first comprehensive and systematic analysis of 10 state-of-the-art token reduction methods. We find that the simple Top-K pruning approach is a very strong baseline across all tested image classification datasets, only outperformed by the EViT method, a straight-forward extension of Top-K. We conducted the first analysis of how the reduction patterns are affected by choice of dataset, number of tokens to be kept, and model capacity. We observe that varying the backbone has a large effect on the reduction patterns, whereas when the keep rate is varied the reduction patterns are very consistent. We also found a moderate-to-strong correlation of the average reduction patterns across datasets, and that the similarity of reduction patterns between methods is a moderate-to-strong proxy for model performance. We hope these findings will help inform future research in token reduction methods.

Acknowledgements: This work was supported by the Pioneer Centre for AI (DNRF grant number P1), partially supported by the Spanish project PID2022-136436NB-I00 and by ICREA under the ICREA Academia programme, and the Canada CIFAR AI Chairs.

References

[1] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. VATT: Transformers for multimodal self-supervised learning from raw video, audio and text. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 2

[2] Lucas Beyer, Pavel Izmailov, Alexander Kolesnikov, Mathilde Caron, Simon Kornblith, Xiaohua Zhai, Matthias Minderer, Michael Tschannen, Ibrahim Alabdulmohsin, and Filip Pavetic. Flexivit: One model for all patch sizes. *arXiv preprint arXiv:2212.08013*, 2022. 2

[3] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *International Conference on Learning Representations*, 2023. 1, 2, 3

[4] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Frédo Durand. What do different evaluation metrics tell us about saliency models? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(3), 2019. 6

[5] Arnav Chavan, Zhiqiang Shen, Zhuang Liu, Zechun Liu, Kwang-Ting Cheng, and Eric Xing. Vision transformer slimming: Multi-dimension searching in continuous optimization space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[6] Mengzhao Chen, Mingbao Lin, Ke Li, Yunhang Shen, Yongjian Wu, Fei Chao, and Rongrong Ji Cf-vit. A general coarse-to-fine method for vision transformer. *arXiv preprint arXiv:2203.03821*, 2022. 2

[7] Tianlong Chen, Yu Cheng, Zhe Gan, Lu Yuan, Lei Zhang, and Zhangyang Wang. Chasing sparsity in vision transformers: An end-to-end exploration. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 19974–19988. Curran Associates, Inc., 2021. 2

[8] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yan-Tao Zheng. Nus-wide: A real-world web image database from national university of singapore. In *Proc. of ACM Conf. on Image and Video Retrieval (CIVR'09)*, Santorini, Greece., July 8-10, 2009. 3

[9] Mostafa Dehghani, Yi Tay, Anurag Arnab, Lucas Beyer, and Ashish Vaswani. The efficiency misnomer. In *International Conference on Learning Representations*, 2022. 8

[10] Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding representation similarity through statistical testing. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 7, 8

[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 *International Conference on Learning Representations*, 2021. 1, 2

[12] Zhiyang Dou, Qingxuan Wu, Cheng Lin, Zeyu Cao, Qiangqiang Wu, Weilin Wan, Taku Komura, and Wenping Wang. Tore: Token reduction for efficient human mesh recovery with transformer. *arXiv preprint arXiv:2211.10705*, 2022. 2

[13] Mingjing Du, Shifei Ding, and Hongjie Jia. Study on density peaks clustering based on k-nearest neighbors and principal component analysis. *Knowledge-Based Systems*, 99:135–145, 2016. 3

[14] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Juergen Gall. Adaptive token sampling for efficient vision transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 4

[15] Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Herve Jegou, and Matthijs Douze. Levit: A vision transformer in convnet's clothing for faster inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021. 2

[16] Joakim Bruslund Haurum, Meysam Madadi, Sergio Escalera, and Thomas B. Moeslund. Multi-scale hybrid vision transformer and sinkhorn tokenizer for sewer defect classification. *Automation in Construction*, 144:104614, 2022. 2, 3

[17] Ju He, Jie-Neng Chen, Shuai Liu, Adam Kortylewski, Cheng Yang, Yutong Bai, and Changhu Wang. Transfg: A transformer architecture for fine-grained recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(1):852–860, Jun. 2022. 2, 4

[18] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 2

[19] Dennis E Hinkle, William Wiersma, and Stephen G Jurs. *Applied statistics for the behavioral sciences*, volume 663. Houghton Mifflin college division, 2003. 6

[20] Chao Hu, Liqiang Zhu, Weibin Qiu, and Weijie Wu. Data augmentation vision transformer for fine-grained image classification. *arXiv preprint arXiv:2211.12879*, 2022. 2

[21] Yunqing Hu, Xuan Jin, Yin Zhang, Haiwen Hong, Jingfeng Zhang, Yuan He, and Hui Xue. Rams-trans: Recurrent attention multi-scale transformer forfine-grained image recognition. *arXiv preprint arXiv:2107.08192*, 2021. 2

[22] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017. 3

[23] Zhenglun Kong, Peiyan Dong, Xiaolong Ma, Xin Meng, Wei Niu, Mengshu Sun, Xuan Shen, Geng Yuan, Bin Ren, Hao Tang, Minghai Qin, and Yanzhi Wang. Spvit: Enabling faster vision transformers via latency-aware soft token pruning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2

[24] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529. PMLR, 09–

15 Jun 2019. 8

[25] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, Feb. 2017. 8

[26] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, Mar. 2020. 8

[27] Ling Li, David Thorsley, and Joseph Hassoun. Sait: Sparse vision transformers through adaptive token pruning. *arXiv preprint arXiv:2210.05832*, 2022. 2

[28] Zhexin Li, Tong Yang, Peisong Wang, and Jian Cheng. Q-vit: Fully differentiable quantization for vision transformer. *arXiv preprint arXiv:2201.07703*, 2022. 2

[29] Youwei Liang, Chongjian GE, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. EVit: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022. 2, 3

[30] Mingbao Lin, Mengzhao Chen, Yuxin Zhang, Ke Li, Yunhang Shen, Chunhua Shen, and Rongrong Ji. Super vision transformer. *arXiv preprint arXiv:2205.11397*, 2022. 2

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, Cham, 2014. Springer International Publishing. 3

[32] Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. Fq-vit: Post-training quantization for fully quantized vision transformer. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 1173–1179, 2022. 2

[33] Yue Liu, Christos Matsoukas, Fredrik Strand, Hossein Azizpour, and Kevin Smith. Patchdropout: Economizing vision transformers using patch dropout. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3953–3962, January 2023. 2

[34] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[35] Sifan Long, Zhen Zhao, Jimin Pi, Shengsheng Wang, and Jingdong Wang. Beyond attentive tokens: Incorporating token importance and diversity for efficient vision transformers. *arXiv preprint arXiv:2211.11315*, 2022. 2

[36] Dmitrii Marin, Jen-Hao Rick Chang, Anurag Ranjan, Anish Prabhu, Mohammad Rastegari, and Oncel Tuzel. Token pooling in vision transformers for image classification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 12–21, January 2023. 2, 3

[37] Lingchen Meng, Hengduo Li, Bor-Chun Chen, Shiyi Lan, Zuxuan Wu, Yu-Gang Jiang, and Ser-Nam Lim. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12309–12318, June 2022. 2

[38] Ari Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 8

[39] Bowen Pan, Rameswar Panda, Yifan Jiang, Zhangyang Wang, Rogerio Feris, and Aude Oliva. Ia-red^2: Interpretability-aware redundancy reduction for vision transformers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 24898–24911. Curran Associates, Inc., 2021. 2

[40] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 1

[41] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 1, 2, 3, 4, 5, 7, 8

[42] Cedric Renggli, André Susano Pinto, Neil Houlsby, Basil Mustafa, Joan Puigcerver, and Carlos Riquelme. Learning to merge tokens in vision transformers. *arXiv preprint arXiv:2202.12015*, 2022. 2, 3

[43] Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021. 4

[44] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 410–420, Prague, Czech Republic, June 2007. Association for Computational Linguistics. 5

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Apr. 2015. 3, 4

[46] Michael S Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Tokenlearner: Adaptive space-time tokenization for videos. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 2

[47] Zhuoran Song, Yihong Xu, Zhezhi He, Li Jiang, Naifeng Jing, and Xiaoyao Liang. Cp-vit: Cascade vision transformer

pruning via progressive sparsity prediction. *arXiv preprint arXiv:2203.04570*, 2022. 2

[48] Alexander Strehl and Joydeep Ghosh. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.*, 3:583–617, mar 2003. 6

[49] Yehui Tang, Kai Han, Yunhe Wang, Chang Xu, Jianyuan Guo, Chao Xu, and Dacheng Tao. Patch slimming for efficient vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12155–12164, 2022. 2

[50] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers & distillation through attention. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 18–24 Jul 2021. 4

[51] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 3

[52] 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. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 2

[53] Matthew Walmer, Saksham Suri, Kamal Gupta, and Abhinav Shrivastava. Teaching matters: Investigating the role of supervision in vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[54] Jun Wang, Xiaohan Yu, and Yongsheng Gao. Feature fusion vision transformer for fine-grained visual categorization. In *British Machine Vision Conference*, 2021. 2

[55] Yunke Wang, Bo Du, and Chang Xu. Multi-tailed vision transformer for efficient inference. *arXiv preprint arXiv:2203.01587*, 2022. 2

[56] Yulin Wang, Rui Huang, Shiji Song, Zeyi Huang, and Gao Huang. Not all images are worth 16x16 words: Dynamic transformers for efficient image recognition. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 2

[57] Yu Wang, Shuo Ye, Shujian Yu, and Xinge You. R2-trans:fine-grained visual categorization with redundancy reduction. *arXiv preprint arXiv:2204.10095*, 2022. 2

[58] Lemeng Wu, Xingchao Liu, and Qiang Liu. Centroid transformers: Learning to abstract with attention, 2021. 2

[59] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18134–18144, June 2022. 2

[60] Yifan Xu, Zhijie Zhang, Mengdan Zhang, Kekai Sheng, Ke Li, Weiming Dong, Liqing Zhang, Changsheng Xu, and Xing Sun. Evo-vit: Slow-fast token evolution for dynamic vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2964–2972, 2022. 2

[61] Hongxu Yin, Arash Vahdat, Jose Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2, 7

[62] Hao Yu and Jianxin Wu. A unified pruning framework for vision transformers. *arXiv preprint arXiv:2111.15127*, 2021. 2

[63] Xiaoyu Yue, Shuyang Sun, Zhanghui Kuang, Meng Wei, Philip Torr, Wayne Zhang, and Dahua Lin. Vision transformer with progressive sampling. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[64] Wang Zeng, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, Wanli Ouyang, and Xiaogang Wang. Not all tokens are equal: Human-centric visual analysis via token clustering transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11101–11111, 2022. 2, 3

[65] Yichen Zhu, Yuqin Zhu, Jie Du, Yi Wang, Zhicai Ou, Feifei Feng, and Jian Tang. Make a long image short: Adaptive token length for vision transformers. *arXiv preprint arXiv:2112.01686*, 2021. 2

[66] Zhuofan Zong, Kunchang Li, Guanglu Song, Yali Wang, Yu Qiao, Biao Leng, and Yu Liu. Self-slimmed vision transformer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2, 3, 8