MADTP: Multimodal Alignment-Guided Dynamic Token Pruning for Accelerating Vision-Language Transformer

Jianjian Cao[1] Peng Ye[1] Shengze Li[1] Chong Yu[2] Yansong Tang[3]
Jiwen Lu[4] Tao Chen[1†]
[1]School of Information Science and Technology, Fudan University
[2]Academy for Engineering and Technology, Fudan University
[3]Tsinghua Shenzhen International Graduate School, Tsinghua University
[4]Department of Automation, Tsinghua University

Abstract

*Vision-Language Transformers (VLTs) have shown great success recently, but are meanwhile accompanied by heavy computation costs, where a major reason can be attributed to the large number of visual and language tokens. Existing token pruning research for compressing VLTs mainly follows a single-modality-based scheme yet ignores the critical role of aligning different modalities for guiding the token pruning process, causing the important tokens for one modality to be falsely pruned in another modality branch. Meanwhile, existing VLT pruning works also lack the flexibility to dynamically compress each layer based on different input samples. To this end, we propose a novel framework named **M**ultimodal **A**lignment-Guided **D**ynamic **T**oken **P**runing (**MADTP**) for accelerating various VLTs. Specifically, we first introduce a well-designed Multi-modality Alignment Guidance (MAG) module that can align features of the same semantic concept from different modalities, to ensure the pruned tokens are less important for all modalities. We further design a novel Dynamic Token Pruning (DTP) module, which can adaptively adjust the token compression ratio in each layer based on different input instances. Extensive experiments on various benchmarks demonstrate that MADTP significantly reduces the computational complexity of kinds of multimodal models while preserving competitive performance. Notably, when applied to the BLIP model in the NLVR2 dataset, MADTP can reduce the GFLOPs by 80% with less than 4% performance degradation. The code is available at https://github.com/double125/MADTP.*

1. Introduction

Vision-Language Transformers (VLTs) have taken multimodal learning domain by storm due to their superior performance on various multimodal tasks, including Visual Reasoning [23], Image Captioning [28], Image-Text Retrieval [22], and Visual Question Answering (VQA) [1]. However, these models [6, 24–26, 33], such as CLIP [33] and BLIP [25], inevitably suffer from expensive computational costs due to their complex architecture, large parameters, and numerous tokens, which restrict their real-world applications and deployments.

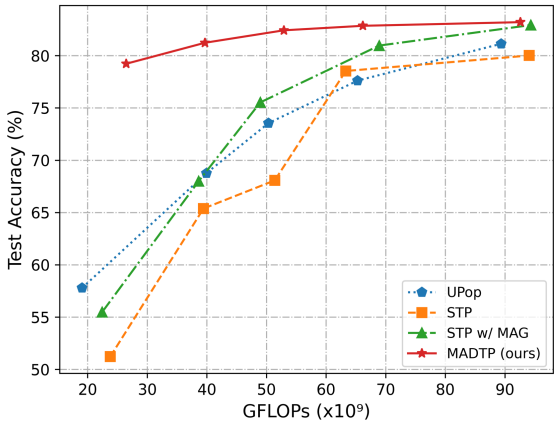

Figure 1. Comparison between our MADTP and other compression methods for the BLIP model tested on the NLVR2 dataset. STP represents the Static Token Pruning method, and MAG denotes our Multi-modality Alignment Guidance module.

To release this limitation, a few works have attempted to accelerate the VLT models. As a pioneer, Upop [35] suggests a unified parameter pruning strategy for compressing VLTs, allowing for simultaneous pruning of submodules across diverse modalities. Recently, considering the token number plays a dominant role in the total computation cost, several studies have put more effort into accelerating VLTs via pruning tokens. ELIP [17] introduces a vision token pruning method to remove less influential tokens based on the supervision of language outputs. CrossGET [36] implements token pruning by selectively eliminating redundant

†Corresponding authors. Email: eetchen@fudan.edu.cn

Methods	Layer-wise Dynamic	Instance-wise Dynamic	Modality Guidance	Modality Alignment
Upop [35]	✓	✗	✗	✗
ELIP [17]	✗	✗	✓	✗
CrossGET [36]	✗	✗	✓	✗
MADTP	✓	✓	✓	✓

Table 1. Characteristics of existing compression methods for VLTs. The proposed MADTP first conducts visual-language modality alignment and then utilizes the aligned features to guide layer-wise and instance-wise dynamic token pruning.

tokens at each layer of the VLTs. Despite some progress achieved by these works, there still exist two unresolved issues. As depicted in Table 1, all these methods face challenges in exploring multi-modality alignment and different inputs to dynamically compress VLT, details are as follows.

Firstly, existing popular VLT models [24, 25, 33] usually consist of multiple modality-specific sub-modules for better capturing the representative knowledge for each modality, which often leads to imbalanced distributions of parameters and features between different modalities. Such imbalances have been extensively analyzed in studies [11, 32]. In other words, different modality branches in VLT generally produce tokens with different representation capabilities for the same semantic concept. As a result, directly applying existing unimodal pruning methods [14, 40, 48] to prune the VLT without considering each token's cross-modality semantic relevance, may falsely remove tokens that are less important in one modality but may be crucial in another. This will further worsen the representation capability imbalance between different modality branches in the compressed VLT. Thus, introducing cross-modality alignment can explicitly align the joint representation of different modalities for the same semantic concept, and increase the chances of eliminating less important tokens for all modalities, resulting in more effective compression of VLTs.

Secondly, different input samples often require different levels of computation complexity [19, 41] for inference. Hence, some research on unimodal dynamic token pruning [7, 29, 31, 44] have emerged recently. These works offer flexibility in removing redundant tokens across different layers of the network by considering the complexity of input instances. However, one disadvantage is that these dynamic pruning works focus on single-modality compression, lacking the consideration of how to dynamically determine one token's importance across multi-modalities for different inputs. Another challenge is that, although promising, the exploration of dynamic token pruning for multimodal models is rarely studied. Thus, based on the aligned multi-modalities representations mentioned above, we further introduce dynamic token pruning modules at different layers of the Vision-Language Transformers, to achieve both input instance- and layer-wise VLT compression.

In this work, we introduce a novel framework called Multimodal Alignment-Guided Dynamic Token Pruning (MADTP) to accelerate VLTs. The MADTP framework accepts image and text inputs, which are fed into a vision branch and a language branch to extract visual and language tokens, respectively. Then, the Multi-modality Alignment Guidance (MAG) module is designed to learn the semantic relevance between tokens from two modalities. Specifically, MAG utilizes learnable tokens to facilitate cross-modal feature alignment and guide the multimodal token pruning. Furthermore, the Dynamic Token Pruning (DTP) module is presented within the Transformer blocks, enabling dynamic adjustment of the compression ratio for each layer based on the complexity of different input instances and the learned alignment guidance. Fig. 1 illustrates the substantial performance improvement achieved by our MADTP framework. Our main contributions can be summarized as follows:

- We reveal the vital role of aligning multi-modalities for guiding VLT compression, and further propose a novel multimodal alignment-guided dynamic token pruning framework called MADTP, to effectively accelerate various Vision-Language Transformers.
- To relieve the unaligned modalities issue, we propose the Multi-modality Alignment Guidance (MAG) module, explicitly aligning the joint representations from different modalities and providing guidance during the multimodal token pruning process.
- To achieve adaptive VLT acceleration based on different inputs, we present the Dynamic Token Pruning (DTP) module, which dynamically adjusts the compression ratio for each layer of VLT models based on the complexity of input instance.
- Extensive experiments across diverse datasets and models consistently verify that MADTP can achieve new state-of-the-art performance. Notably, MADTP achieves outstanding compression on the BLIP model in the NLVR2 dataset, reducing GFLOPs by 80% while experiencing a performance decrease of less than 4%.

2. Related Work

2.1. Vision-Language Transformer

Vision-Language Transformer(VLT) models aim to make full use of information from different modalities and have been proven to be effective in various fields. CLIP [33] and BLIP [25] are two representative VLT models. CLIP performs well on many downstream tasks by pretraining with images and texts matching. Further, BLIP uses a cross-attention layer to interact visual information with text information during the matching process of images and texts. Although VLT models show the powerful ability, they generally suffer high computation costs due to the need to process different modalities of information. Thus, it is necessary and of practical value to compress VLT models.

2.2. Multimodal Compression

The dominant techniques for model compression [8, 18, 37] encompass pruning [2, 4, 40, 42, 46], quantization [13], knowledge distillation [15] and low-rank decomposition [47], among others [20, 21]. However, these methods mainly focus on single-modality model compression, such as ViTs, while multimodal compression such as VLTs remain challenges. To this end, a few works have attempted to compress the VLT models recently. As the pioneering work, DistillVLM [12] leverages knowledge distillation to transfer the knowledge from larger VLTs to smaller VLTs. Upop [35] adopts a layer-wise dynamic parameter pruning approach, which uniformly searches subnets and adaptively adjusts the pruning ratio of each layer. ELIP [17] presents a vision token pruning technique that eliminates less important tokens by leveraging language outputs as supervision. CrossGET [36] introduces the cross tokens to facilitate multimodal token pruning. However, all these methods overlook the significance of multi-modality alignment guidance for VLT compression, leading to a decrease in the performance of the compressed models. Although some works [17, 36] attempt to utilize modality guidance to assist token pruning, this problem still exists. Our proposed MAG module explicitly aligns the feature representations of the two modalities using learnable tokens. It provides comprehensive guidance for subsequent dynamic token pruning process, enabling effective resolution of this challenge.

2.3. Token Merging and Pruning

Token merging and pruning [3, 5] are proven effective for model compression. ToMe [5] designed a token merging strategy for ViTs, merging similar parts in each block. Further, [3] merges non-critical tokens into crucial tokens, which not only reduces the number of tokens but also retains more information. Most of these methods reduce a fixed number of tokens at each step. However, according to [34, 39, 43], the number of tokens retained by the current block should be related to its importance to the final task. DynamicViT [34] uses a prediction module to measure the importance of each patch embedding in the current input to decide whether to discard the patch. AdaViT [43] adaptively stop some tokens from participating in subsequent calculations. MuE [39] design an early exiting strategy based on input similarity for ViT models. Unlike these works processing unimodal ViT models, we focus on reducing the computation cost of various VLT models, by designing a multimodal dynamic token pruning strategy based on the complexity of the input image and text pairs.

3. Methodology

The MADTP architecture overview is depicted in Fig. 2. In this following, we first give a brief introduction of the Vision-Language Transformers in Sec. 3.1. We then present our Multi-modality Alignment Guidance module and Dynamic Token Pruning module in Sec. 3.2 and Sec. 3.3, respectively. Finally, we elaborate on the optimization function of the framework in Sec. 3.4.

3.1. Preliminaries

Vision-Language Transformers have emerged as the prominent architectures [25, 26, 33] in multimodal learning, comprising two branches: the vision branch and the language branch. The vision branch usually employs the ViT [10] as the visual encoder, while the language branch utilizes BERT [9] as the language encoder, extracting visual and language tokens from their respective modalities. In detail, given an image and a text as inputs, the visual encoder performs patch embedding on the image to generate the visual tokens $V = \{V_1, V_2, ...V_N\}$, where N is the patch number, and the language encoder processes the words in the text using token embedding, converting them into language tokens $L = \{L_1, L_2, ..., L_M\}$, where M is the number of words. Furthermore, two learnable tokens, V_{cls} and L_{eos}, are added to the visual tokens and language tokens, respectively. These token embeddings provide comprehensive representations for the image and text inputs, which are then passed through transformer blocks for feature encoding. In VLTs, both the vision and language branches consist of L layers of transformer blocks. Each block comprises a Multi-Head Self Attention (MHSA) layer and a Feed Forward Network (FFN) layer, enabling the model to capture contextual relationships within each modality. In addition, some VLT models like BLIP [25], incorporate several Cross Attention layers to capture inter-modal interactions and enhance information fusion between two modalities.

3.2. Multi-modality Alignment Guidance

As discussed in Sec. 1, the unaligned modalities issue highlights the challenge of directly applying unimodal token pruning methods to VLTs. To alleviate this problem, the Multi-modality Alignment Guidance (MAG) module is designed to explicitly align the feature representations between two modalities, and provide sufficient guidance for the multimodal token pruning process. As shown in Fig. 2, we insert the MAG module between the transformer blocks of two modal branches in the VLT architecture.

Specifically, we first apply two linear layers to map the visual tokens V and language tokens L from each layer of VLTs into the same feature dimension. The linear layers and mapping process can be represented as follows:

$$\begin{aligned} V' &= W_v V + B_v, \\ L' &= W_t L + B_t, \end{aligned} \tag{1}$$

where V' and L' are the mapped visual and language tokens, respectively. The W_v, W_t, B_v, and B_t are layer-specific trainable weight matrices and biases.

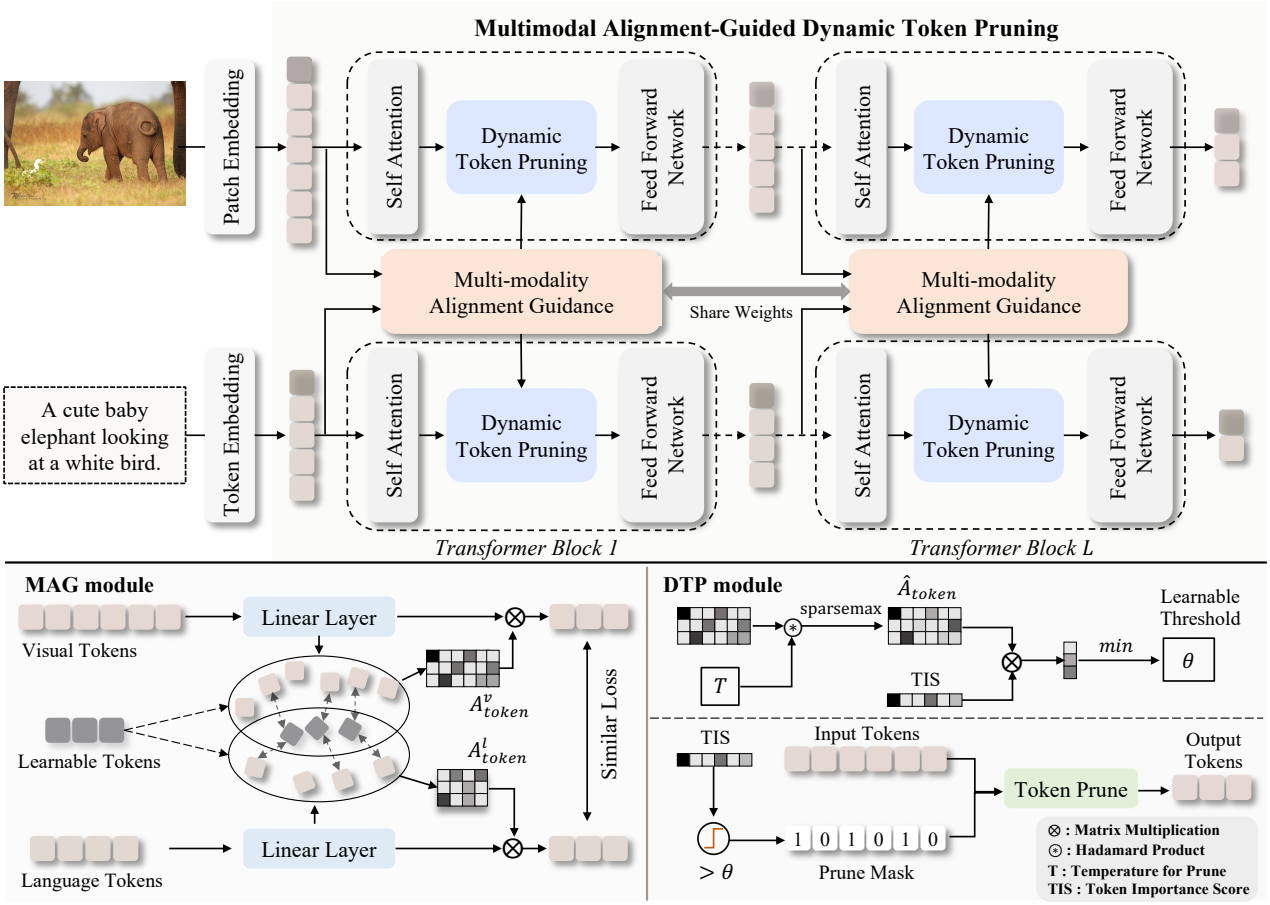

Figure 2. Overview of the proposed MADTP framework. It comprises two main components: the Multi-modality Alignment Guidance (MAG) module and the Dynamic Token Pruning (DTP) module. The MAG module is placed between the vision and language branches in VLTs, facilitating explicit alignment of representations across modalities and offering guidance for token pruning. Meanwhile, the DTP module is incorporated within each transformer block, allowing for dynamic token pruning based on the complexity of input instances.

Next, we utilize learnable tokens $E = \{E_1, E_2, ...E_K\}$ as common feature space to establish associations between the visual and language modalities, where K is the number of learnable tokens. In detail, we employ a scaled dot-product attention layer to calculate the correlation between the learnable tokens E and the mapped visual tokens V', resulting in token attention maps $A^v_{token} \in \mathbb{R}^{K \times N}$ and visual features E^v. This process can be expressed as:

$$A^v_{token} = softmax(\frac{EV'^T}{\sqrt{d_k}}), \tag{2}$$

$$E^v = A^v_{token} * V', \tag{3}$$

where d_k is a scaling factor. Similarly, we can also obtain the token attention maps $A^l_{token} \in \mathbb{R}^{K \times M}$ between the mapped language tokens and learnable tokens, and extract the language features E^l.

Further, we calculate the similarity between these two features and incorporate it into the final loss constraint to assist the model during training. We believe that the visual and language features learned by the same learnable tokens should exhibit strong semantic relevance. Through the above operations, we explicitly align the representations between two modalities and obtain token attention maps representing the modality alignment achieved by the learnable tokens. Afterward, these maps are fed into the Dynamic Token Pruning module to guide the token pruning process of the VLTs, ensuring that the pruned tokens are redundant in both modalities and enhancing the compression effectiveness of the multimodal model, which is exemplified in Fig. 3. Note that the MAG modules share weights in the MADTP framework.

3.3. Dynamic Token Pruning

Dynamic token pruning in single-modality compression has been proven to be more efficient than static token pruning, as it enables adaptive adjustment of the model's compression rate based on the complexity of the input instance. Motivated by this, we have also designed a Dynamic Token Pruning (DTP) module in the MADTP framework. As illustrated in Fig. 2, we insert the DTP module between the Self Attention layer and the Feed Forward Network in each Transformer block, allowing it to dynamically reduce the

number of input tokens at each layer of VLTs. Following a similar procedure as in the single-modality token pruning, we first calculate the importance score for each token. Then, a learnable threshold is employed to dynamically prune tokens at both the input instance-wise and layer-wise levels.

Token Importance Score. Apart from considering token importance based on the class attention map [29, 44], as commonly done in traditional token pruning for ViTs, our approach extends to incorporate the importance of tokens within the same modality and the guidance of token alignment across different modalities. The Token Importance Score (TIS) is obtained by averaging three types of scores:

$$\text{TIS} = (S_{\text{cls}} + S_{\text{self}} + S_{\text{token}})/3, \tag{4}$$

where S_{cls} represents the class attention score as implemented by [29]. S_{self} and S_{token} denote the self-attention score and token attention score, respectively. Taking the visual modality as an example, we utilize the self-attention maps $A^v_{self} \in \mathbb{R}^{N\times N}$ from the MHSA layer and the token attention maps $A^v_{token} \in \mathbb{R}^{K\times N}$ obtained from the MAG module to calculate the attention scores S^v_{self} and S^v_{token} through the following steps:

$$S^{v,k}_{\text{self}} = \frac{\max(A^{v,k}_{self})}{\sum_{k=1}^{N}\max(A^{v,k}_{self})}, \tag{5}$$

$$S^{v,k}_{\text{token}} = \frac{\max(A^{v,k}_{token})}{\sum_{k=1}^{N}\max(A^{v,k}_{token})}. \tag{6}$$

Here, N refers to the total number of visual tokens. $\max(A^{v,k}_{self})$ and $\max(A^{v,k}_{token})$ represent the maximum value for the k-th token in the self-attention maps and token attention maps, respectively. To ensure the scores are within the range of $[0, 1]$, the attention scores (S^v_{self} and S^v_{token}) are normalized by dividing them by the sum of their corresponding values. Note that by incorporating these three attention scores, our TIS can effectively avoid discarding crucial tokens by considering their relevance to the task, as well as their importance within and across modalities.

Learnable Threshold. To achieve instance-wise adaptive token pruning while minimizing operational costs, we propose the use of learnable thresholds for dynamic token pruning within MADTP. Specifically, we utilize the token attention maps A_{token} learned from the MAG module to compute these thresholds. Firstly, we multiply A_{token} by a temperature parameter T and apply sparsemax function [30] to obtain sparse token attention maps, denoted as $\hat{A}_{token}$,

$$\hat{A}_{token} = \text{sparsemax}(T * A_{token}). \tag{7}$$

The role of the sparsemax function is to produce sparse distributions by minimizing the squared Euclidean distance between the output distribution and the input values.

$$\text{sparsemax}(\boldsymbol{z}) := \underset{\boldsymbol{p}\in\Delta^{K-1}}{\arg\min} \ \|\boldsymbol{p}-\boldsymbol{z}\|^2, \tag{8}$$

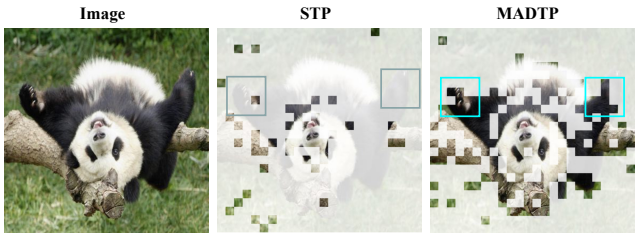

Figure 3. Visualization of token pruning results between STP and MADTP, providing strong evidence that our approach emphasizes modality correlation and effectively avoids pruning crucial tokens.

where $\Delta^{K-1} := \{\boldsymbol{p} \in \mathbb{R}^K | \mathbf{1}^T\boldsymbol{p} = 1, \boldsymbol{p} \geq 0\}$. Next, we perform matrix multiplication between $\hat{A}_{token}$ and TIS to obtain K thresholds, and take the minimum value among these thresholds as the final threshold θ, used for the following token pruning procedure for this DTP module.

$$\theta = \min(\hat{A}_{token} \otimes \text{TIS}). \tag{9}$$

Token Pruning. Based on the token importance scores and learnable threshold mentioned above, we can proceed with the designed token pruning scheme to reduce the number of input tokens. Firstly, we compare the TIS score of each token with the threshold θ to obtain the prune mask M_p, which can be formulated in Equation 10:

$$M_p(x_i) = \begin{cases} 1, & if \ \text{TIS}(x_i) > \theta, \\ 0, & otherwise. \end{cases} \tag{10}$$

Where x_i represents the i-th input tokens. Then we keep the tokens with scores greater than the threshold and eliminate the other tokens according to the pruning mask. However, directly discarding tokens may result in information loss. To address this, we adopt a similar approach as EVit [27], weighting the pruned tokens based on their TIS to generate a new token, which is then added to the retained tokens.

3.4. Objective Function

Due to VLTs having different loss functions for various multimodal tasks, we represent the specific task loss function as L_{task} during training. Additionally, as explained in Section 3.2, we incorporate a similar loss denoted as L_{sim} to capture the alignment relationship between the visual features E^v and language features E^l obtained from the MAG modules for optimizing the model pruning process. Consequently, the overall loss function L of the proposed MADTP framework can be expressed as:

$$L = L_{task} + \alpha L_{sim}, \tag{11}$$

where α denotes the balance coefficient. The computation for L_{sim} is defined as follows:

$$L_{sim} = \frac{1}{K}\sum_{i=1}^{K}(1 - cos(E^v_i, E^l_i)). \tag{12}$$

Where K is the number of visual and language features.

4. Experiments

4.1. Experimental Setup

Dataset and evaluation metrics. To evaluate our method comprehensively, four multimodal datasets are used, including NLVR2 [38], COCO [28], Flickr30k [45] and VQA v2.0 [16]. NLVR2 [38] contains 107,292 pairs of images and text descriptions. COCO [28] comprises around 330,000 images, each accompanied by five text descriptions. Flickr30k [45] is mainly used for image and text retrieval tasks, and consists of 31,783 images, and each image has a descriptive title. VQA v2.0 [16] is a human-annotated, open-ended question-and-answer dataset about images. Performance evaluation metrics are task-specific, while model complexity is measured in GFLOPs (Giga-Floating-Operations per image-text pair).

Implementation details. We use the MADTP framework to compress the CLIP [33] and BLIP [25] models, which are initialized with pretrained weights from the official implementation of [35]. During the compressing process, we utilize 8 A100 GPUs with a batch size of 32, and the hyper-parameter α in the loss function is set to 0.1. The temperature T in the DTP module is dynamically adjusted at each epoch, based on the GFLOPs of the pruned model. Due to space limitations and the variability of training configurations across different models, more detailed experiment settings can be found in Appendix B.

4.2. Experiments on the Visual Reasoning Task

In this section, we conduct experiments utilizing our MADTP framework to compress the BLIP model on the NLVR2 dataset. In Table 2, we compare our approach with the state-of-the-art method [35] to demonstrate its effectiveness. Additionally, we perform ablation studies to analyze the impact of different components and hyperparameters of the MADTP framework, presenting the results in Table 3 and Table 4, respectively. Moreover, we visualize the token pruning results for the compressed model in Fig. 4.

Comparison to State-of-the-art Approaches. We report the performance of the MADTP framework for compressing the BLIP model at reduce ratios of 0.3, 0.5, 0.6, 0.7, and 0.8. The reduce ratio represents the proportion of the model's GFLOPs targeted for compression. In order to assess the efficiency of our dynamic compression approach, we implement a baseline approach called Static Token Pruning (STP) which prunes a fixed number k of redundant tokens at each layer of the VLTs based on their importance scores computed in equation 4. In Table 2, under a reduce ratio of 0.3, MADTP achieved a 2.17% increase in accuracy on the dev set and a 2.07% increase on the test set compared to Upop [35]. Notably, at a reduce ratio of 0.5, these improvements extended to 5.08% and 5.24%, respectively. Even at higher reduce ratios of 0.6, 0.7, and 0.8, MADTP demonstrated its ability to further compress the model while maintaining performance within an acceptable range. Remarkably, at a reduce ratio of 0.8, our method only experienced a 3.86% drop on the test set compared to the uncompressed model. These results highlight the effectiveness and superiority of our MADTP in achieving substantial model compression while preserving task performance across different reduce ratios.

Approach	Reduce Ratio	Dev Acc	Test Acc	GFLOPs
Uncompressed	/	82.48	83.08	132.54
STP	0.3	79.50	80.01	94.08
	0.5	78.08	77.61	68.31
UPop [35]	0.3	80.33	81.13	89.36
	0.5	76.89	77.61	65.29
	0.6	72.85	73.55	50.35
	0.7	68.71	68.76	39.93
	0.8	57.17	57.79	19.08
MADTP (Ours)	0.3	**82.50**	**83.20**	92.60↓30%
	0.5	**81.97**	**82.85**	66.16↓50%
	0.6	**81.92**	**82.42**	52.92↓60%
	0.7	**80.67**	**81.23**	39.69↓70%
	0.8	**78.28**	**79.22**	26.46↓80%

Table 2. Comparison of compression results for BLIP model on the NLVR2 dataset. Bold indicates the best results. Reduce Ratio indicates the desired compression ratio of GFLOPs.

Components of MADTP		Dev Acc	Test Acc	GFLOPS
TIS	only w/ S_{self}	81.49	82.13	70.46
	only w/ S_{token}	80.68	81.00	66.74
	only w/ S_{cls}	81.62	82.25	69.67
Module	w/o MAG	79.65	80.96	68.91
	w/o DTP	80.83	81.44	68.70
MADTP (Ours)		**81.97**	**82.85**	**66.16**

Table 3. Ablation study of different components in MADTP framework for compressing BLIP on NLVR2 at 0.5 reduce ratio.

Hyperparameters		Dev Acc	Test Acc	GFLOPS
K	50	81.44	82.03	67.70
	100	**81.97**	**82.85**	**66.16**
	150	81.49	82.19	66.79
	200	81.74	81.96	66.99
d_k	256	81.79	82.28	66.94
	512	81.79	82.46	68.63
	768	**81.97**	**82.85**	**66.16**
	1024	81.60	81.95	66.55
Operation	mean-keep	81.34	81.70	67.10
	max-keep	**81.97**	**82.85**	**66.16**

Table 4. Hyperparameters for compressing BLIP on NLVR2 at 0.5 reduce ratio. K and d_k donets the number and the channel dimension of learnable tokens. The "mean-keep" and "max-keep" operations are utilized for parallel training within each mini-batch.

Dataset	Approach	Reduce Ratio	Image→Text			Text→Image			GFLOPS
			R@1	R@5	R@10	R@1	R@5	R@10	
Flickr30K (1K test set)	Uncompressed	/	96.8	100.0	100.0	86.6	97.8	99.1	395.7
	UPop [35]	0.5	93.2	99.4	99.8	80.5	95.4	97.6	201.1
		0.75	82.9	95.7	97.8	67.3	89.5	93.5	102.6
	MADTP (Ours)	0.5	**93.9**	**99.5**	**99.8**	**83.3**	**97.0**	**98.5**	178.8↓55%
		0.75	**88.4**	**97.3**	**99.0**	**76.9**	**94.2**	**97.0**	99.5↓75%
COCO (5K test set)	Uncompressed	/	71.5	90.8	95.4	56.8	80.7	87.6	395.7
	UPop [35]	0.5	70.8	90.8	95.2	53.1	79.9	87.3	196.3
		0.75	56.1	82.4	90.2	41.1	71.0	81.4	105.9
	MADTP (Ours)	0.5	**72.7**	**91.8**	**96.1**	**55.0**	**79.9**	**87.5**	190.2↓52%
		0.75	**66.2**	**88.4**	**93.7**	**49.9**	**76.3**	**85.1**	92.4↓77%

Table 5. Compress CLIP on the Flickr30K and COCO datasets of the Image-Text Retrieval task. The R@1, R@5, and R@10 are the higher the better. The best results are in bold.

Dataset	Approach	Reduce Ratio	Image→Text			Text→Image			GFLOPS
			R@1	R@5	R@10	R@1	R@5	R@10	
Flickr30K (1K test set)	Uncompressed	/	96.8	99.9	100.0	86.9	97.3	98.7	153.2
	UPop [35]	0.5	94.0	99.5	99.7	82.0	95.8	97.6	91.0
		0.75	85.8	97.4	98.4	71.3	91.0	94.9	51.0
	MADTP (Ours)	0.5	**95.1**	**99.5**	**99.7**	**82.3**	**96.2**	**98.0**	74.5↓51%
		0.75	**91.8**	**98.5**	**99.6**	**77.1**	**93.2**	**96.1**	58.7↓62%
COCO (5K test set)	Uncompressed	/	81.9	95.4	97.8	64.3	85.7	91.5	153.2
	UPop [35]	0.5	77.4	93.4	97.0	59.8	83.1	89.8	88.3
		0.75	62.9	86.2	92.3	47.4	74.8	83.9	50.2
	MADTP (Ours)	0.5	**79.1**	**94.2**	**97.2**	**60.3**	**83.6**	**89.9**	87.4↓43%
		0.75	**71.2**	**90.0**	**94.0**	**53.4**	**78.4**	**86.2**	50.2↓67%

Table 6. Compress BLIP on the Flickr30K and COCO datasets of the Image-Text Retrieval task. The R@1, R@5, and R@10 are the higher the better. The best results are in bold.

Effect of Components. Table 3 illustrates the contributions of different components in the proposed MADTP framework. We evaluate the impact of Token Importance Scores (TIS) and observe that combining scores from three sources yields the best results for token pruning. Additionally, we assess the individual effects of the two modules introduced in the MADTP framework. The MAG module improves performance by 2.32% on the dev set and 1.89% on the test set. Similarly, the DTP module leads to performance improvements of 1.14% and 1.41% on the respective sets. These experiments confirm the effectiveness of our proposed module within the MADTP framework.

Effect of Hyperparameters. To illustrate the influence of various hyperparameters in the proposed MADTP framework, we compare the performance of the pruned model under different hyperparameter settings. Table 4 showcases how the compression results are influenced by the number and channel dimensions of learnable tokens in the MAG module. The best performance is achieved when K is set to 100 and d_k is set to 768. Additionally, we discuss the pruning strategy used in the dynamic token pruning process. The results indicate that the "max-keep" operation yields the best results, which determine the number of tokens to prune for a mini-batch based on the instance with the highest inference complexity.

4.3. Experiments on the Retrieval Task

We compress the CLIP [33] and BLIP [25] models on the Flickr30K and COCO datasets with reduce ratios of 0.5 and 0.75, respectively. Tables 5 and 6 demonstrate the superior performance of our MADTP framework in image-text retrieval tasks across different model architectures. It can be observed that when compressing the CLIP model on COCO dataset using our MADTP, there is a significant improvement in various metrics compared to the Upop [35]. Particularly, for high reduce ratio such as 0.75, we achieved improvements of up to 10% in certain metrics (e.g., image-to-text recall@1 increased from 56.1% to 66.2%), and our GFLOPS metric is lower. Similarly, our MADTP compression experiments on the BLIP model also achieve impressive results compared to the Upop [35] method.

4.4. Experiments on the Image Caption Task

To assess the generalization capability of our proposed MADTP, we conducted additional experiments on the Image Caption task. Specifically, we compressed the BLIP model using reduce ratios of 0.5 and 0.75 on the COCO

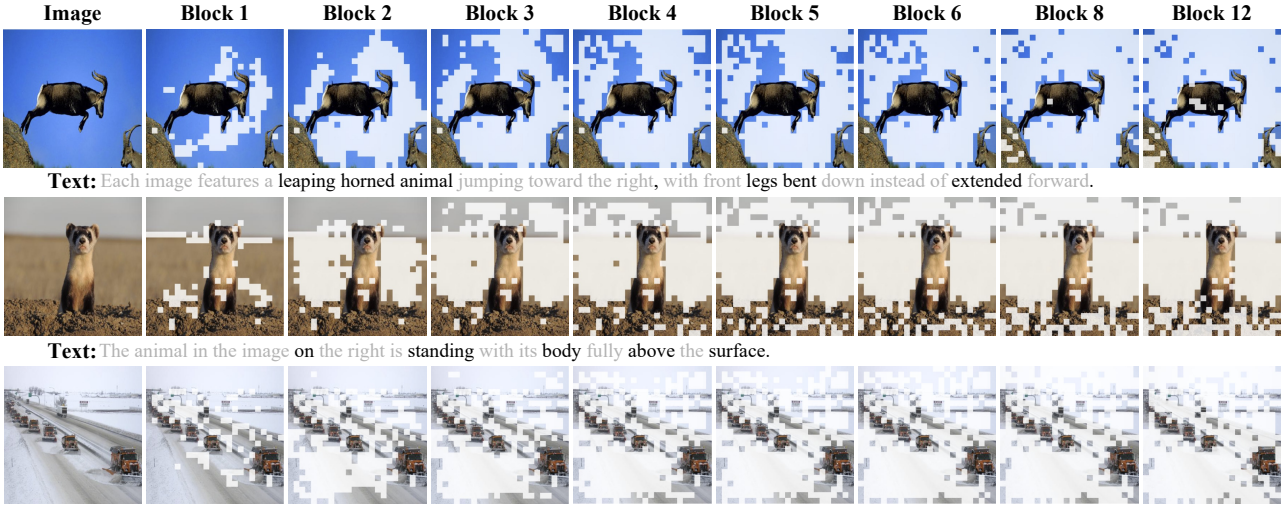

Figure 4. Visualization of our MADTP's compressed BLIP results on NLVR2 dataset at each transformer block. The white mask in the image represents the pruned visual tokens, while the gray words in the text indicate the discarded language tokens. Our method effectively learns semantic relevance between modalities and effectively prunes tokens that are unimportant in both modalities.

Approach	Reduce Ratio	Image Caption			Visual Question Answering		
		CIDEr	SPICE	GFLOPs	Test-dev	Test-std	GFLOPs
Uncompressed	/	133.3	23.8	65.7	77.4	77.5	186.1
UPop [35]	0.5	128.9	23.3	39.8	76.3	76.3	109.4
	0.75	117.4	21.7	22.2	74.5	74.6	62.3
MADTP (Ours)	0.5	**131.0**	**23.5**	39.7↓39%	**76.8**	**76.8**	79.4↓57%
	0.75	**120.1**	**22.0**	22.1↓66%	**76.3**	**76.2**	61.6↓67%

Table 7. Compress BLIP on the Image Caption task and the Visual Question Answering task. The CIDEr, SPICE, test-dev, and test-std are the higher the better. The best results are in bold.

caption dataset. The results in Table 7 demonstrate the superior performance of our MADTP in the Image Caption task. Specifically, our MADTP method surpasses Upop [35] in terms of the CIDEr metric, achieving a 2.1% improvement at a reduce ratio of 0.5 and a 2.7% improvement at a reduce ratio of 0.75. These results emphasize the potential of MADTP in finding a balance between the computational cost of Vision-Language Transformers (VLTs) and maintaining high-quality image captioning capabilities.

4.5. Experiments on the Visual QA Task

In order to further validate the effectiveness of our MADTP method, we conducted compression experiments on the BLIP model using the VQA v2.0 dataset with reduce ratios of 0.5 and 0.75. The results, as depicted in Table 7, provide clear evidence that MADTP outperforms Upop [35] in terms of compression performance on the Visual QA task, particularly at higher reduce ratios. It is worth noting that our MADTP method achieves a remarkable 57% reduction in the GFLOPs of the BLIP model while maintaining a performance degradation of less than 1%. These experimental findings serve as strong validation for the capability of our MADTP method to effectively accelerate VLTs while preserving model performance.

5. Conclusion

We present the Multi-modality Alignment-Guided Dynamic Token Pruning (MADTP) framework to tackle the heavy computation costs of VLTs. Our MADTP integrates the MAG module, which aligns features across modalities and guides the token pruning process to eliminate less important tokens in both modalities. Additionally, the DTP module is introduced to dynamically adjust the token compression ratio based on complexity of input instance. Through extensive experiments, we show that MADTP is a promising approach for accelerating VLTs by reducing computational costs without sacrificing performance.

6. Acknowledgments

This work is supported by National Natural Science Foundation of China (No. 62071127, and 62101137), National Key Research and Development Program of China (No. 2022ZD0160100), Shanghai Natural Science Foundation (No. 23ZR1402900), Shanghai Municipal Science and Technology Major Project (No.2021SHZDZX0103). The computations in this research were performed using the CFFF platform of Fudan University.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 1

[2] Sajid Anwar, Kyuyeon Hwang, and Wonyong Sung. Structured pruning of deep convolutional neural networks. *ACM Journal on Emerging Technologies in Computing Systems*, 13(3), 2015. 3

[3] Zhe Bian, Zhe Wang, Wenqiang Han, and Kangping Wang. Muti-scale and token mergence: Make your vit more efficient. *arXiv preprint arXiv:2306.04897*, 2023. 3

[4] Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? *Proceedings of machine learning and systems*, 2: 129–146, 2020. 3

[5] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022. 3

[6] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin-Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, MarcoTulio Ribeiro, and Yi Zhang. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023. 1

[7] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[8] Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*, 2017. 3

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North*, 2019. 3

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[11] Yunfeng Fan, Wenchao Xu, Haozhao Wang, Junxiao Wang, and Song Guo. Pmr: Prototypical modal rebalance for multimodal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20029–20038, 2023. 2

[12] Zhiyuan Fang, Jianfeng Wang, Xiaowei Hu, Lijuan Wang, Yezhou Yang, and Zicheng Liu. Compressing visual-linguistic model via knowledge distillation. *arXiv: Computer Vision and Pattern Recognition,arXiv: Computer Vision and Pattern Recognition*, 2021. 3

[13] Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. In *Low-Power Computer Vision*, pages 291–326. Chapman and Hall/CRC, 2022. 3

[14] Mitchell A Gordon, Kevin Duh, and Nicholas Andrews. Compressing bert: Studying the effects of weight pruning on transfer learning. *arXiv preprint arXiv:2002.08307*, 2020. 2

[15] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021. 3

[16] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 6

[17] Yangyang Guo, Haoyu Zhang, Liqiang Nie, Yongkang Wong, and Mohan Kankanhalli. Elip: Efficient language-image pre-training with fewer vision tokens. *arXiv preprint arXiv:2309.16738*, 2023. 1, 2, 3

[18] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. 3

[19] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7436–7456, 2021. 2

[20] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3

[21] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016. 3

[22] X. Jia, E. Gavves, B. Fernando, and T. Tuytelaars. Guiding the long-short term memory model for image caption generation. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2407–2415. IEEE Computer Society, 2015. 1

[23] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[24] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021. 1, 2

[25] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 1, 2, 3, 6, 7

[26] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 1, 3

[27] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *ICLR*, 2022. 5

[28] 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1, 6

[29] Xiangcheng Liu, Tianyi Wu, and Guodong Guo. Adaptive sparse vit: Towards learnable adaptive token pruning by fully exploiting self-attention. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 1222–1230. International Joint Conferences on Artificial Intelligence Organization, 2023. Main Track. 2, 5

[30] Andre Martins and Ramon Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *International conference on machine learning*, pages 1614–1623. PMLR, 2016. 5

[31] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[32] Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8238–8247, 2022. 2

[33] Alec Radford, JongWook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *Cornell University - arXiv,Cornell University - arXiv*, 2021. 1, 2, 3, 6, 7

[34] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Neural Information Processing Systems,Neural Information Processing Systems*, 2021. 3

[35] Dachuan Shi, Chaofan Tao, Ying Jin, Zhendong Yang, Chun Yuan, and Jiaqi Wang. UPop: Unified and progressive pruning for compressing vision-language transformers. In *Proceedings of the 40th International Conference on Machine Learning*, pages 31292–31311. PMLR, 2023. 1, 2, 3, 6, 7, 8

[36] Dachuan Shi, Chaofan Tao, Anyi Rao, Zhendong Yang, Chun Yuan, and Jiaqi Wang. Crossget: Cross-guided ensemble of tokens for accelerating vision-language transformers. *arXiv preprint arXiv:2305.17455*, 2023. 1, 2, 3

[37] Pablo Sprechmann, Alexander M Bronstein, and Guillermo Sapiro. Learning efficient sparse and low rank models. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1821–1833, 2015. 3

[38] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. *arXiv preprint arXiv:1811.00491*, 2018. 6

[39] Shengkun Tang, Yaqing Wang, Zhenglun Kong, Tianchi Zhang, Yao Li, Caiwen Ding, Yanzhi Wang, Yi Liang, and Dongkuan Xu. You need multiple exiting: Dynamic early exiting for accelerating unified vision language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10781–10791, 2023. 3

[40] Yehui Tang, Kai Han, Yunhe Wang, Chang Xu, Jianyuan Guo, Chao Xu, and Dacheng Tao. Patch slimming for efficient vision transformers. *Cornell University - arXiv,Cornell University - arXiv*, 2021. 2, 3

[41] Wenhan Xia, Hongxu Yin, Xiaoliang Dai, and N.K. Jha. Fully dynamic inference with deep neural networks. *IEEE Transactions on Emerging Topics in Computing*, PP:1–1, 2021. 2

[42] Huanrui Yang, Hongxu Yin, Maying Shen, Pavlo Molchanov, Hai Li, and Jan Kautz. Global vision transformer pruning with hessian-aware saliency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18547–18557, 2023. 3

[43] 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. 3

[44] Hongxu Yin, Arash Vahdat, Jose M. 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 (CVPR)*, pages 10809–10818, 2022. 2, 5

[45] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 6

[46] Lu Yu and Wei Xiang. X-pruner: explainable pruning for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24355–24363, 2023. 3

[47] Xiyu Yu, Tongliang Liu, Xinchao Wang, and Dacheng Tao. On compressing deep models by low rank and sparse decomposition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7370–7379, 2017. 3

[48] Chuanyang Zheng, Kai Zhang, Zhi Yang, Wenming Tan, Jun Xiao, Ye Ren, Shiliang Pu, et al. Savit: Structure-aware vision transformer pruning via collaborative optimization. *Advances in Neural Information Processing Systems*, 35:9010–9023, 2022. 2