LTP-ViT: Lightweight Token Pre-Pruning for Low-Latency Vision Transformer Inference

Jun-Sung Kim
Department of Electronic Engineering
Hanyang University
Seoul, Korea
cooljun0108@hanyang.ac.kr

Soomin Rho
Department of Electronic Engineering
Hanyang University
Seoul, Korea
smrho@hanyang.ac.kr

Ki-Seok Chung*
Department of Electronic Engineering
Hanyang University
Seoul, Korea
kchung@hanyang.ac.kr

***Abstract*—Vision Transformer models (ViTs) achieve state-of-the-art performance but suffer from quadratic computation complexity, hindering widespread deployment, especially in real-time systems and resource-constrained edge devices. Many existing token-reduction methods introduce significant algorithmic overhead, often failing to align with theoretically predicted reductions in practice. We propose a lightweight token pre-pruning method for Vision Transformer models called LTP-ViT, designed to minimize inference latency and memory footprint. By leveraging semantic redundancy and spatial importance between tokens, LTP-ViT identifies and prunes unimportant tokens in a single step before processing the first Transformer block, eliminating computational bottlenecks. Experimental results on ImageNet-1K demonstrate that LTP-ViT achieves a 1.4x–2.6x end-to-end speedup over widely known methods such as PaPr and ToMe in a server platform with 3.6 percent higher accuracy at similar throughput. On an NVIDIA Jetson Xavier NX, it reduces peak memory usage by 30 percent (520 MB) and accelerates pre-pruning latency by up to 8.0x compared to PaPr. These experimental results confirm that our hardware-friendly, front-loaded LTP-ViT provides a practical solution for real-time ViT deployment on both server and edge platforms.**

***Index Terms*—Vision Transformer, Token pruning, Image classification, Computer vision, Edge computing**

I. Introduction

Vision Transformers (ViTs) [1] have demonstrated state-of-the-art performance across various computer vision tasks. However, their high computational cost, which scales quadratically with the number of tokens, remains a significant barrier to practical deployment. This issue is serious not only to resource-constrained edge devices but also to large-scale servers. To mitigate these issues across diverse computing platforms, intra-layer token pruning and merging techniques have been proposed to reduce the number of tokens in self-attention layers, as shown in Fig. 1 (a) [2]–[9]. Existing approaches can be broadly categorized into three types:

- Merging and fusion-based methods: ToMe [2], Token Fusion [3], GTP-ViT [8], and EViT [5] merge similar tokens to reduce the total count.
- Pruning-based methods: DynamicViT [4], A-ViT [6], and Zero-TPrune [9] utilize predictive modules or adaptive thresholds to hierarchically skip redundant processing.
- Attention-based method: ATS [7] leverages self-attention weights for token sampling.

Intra-layer pruning and merging methods often fail to achieve substantial reduction and execution speedups in practice. In fact, these methods often suffer from high latency due to the additional computational overhead of pruning and merging operations within a layer. More recently, one-step pre-pruning methods such as PaPr have been proposed to overcome these issues, achieving higher accuracy and throughput than intra-layer methods [10]. One-step pre-pruning refers to completing the entire token pruning process before processing the first Transformer block. Fig. 1 (b) shows the pre-pruning flow.

Although PaPr achieves better trade-offs between accuracy and efficiency than intra-layer pruning and merging methods, it often falls short of meeting the requirements for real-time applications. The latency remains a bottleneck because the underlying pruning mechanism introduces significant operational complexity, preventing the model from achieving its intended speedup in practice. To overcome this problem, in this paper, we propose a lightweight token pre-pruning method for Vision Transformer models (LTP-ViT). Unlike PaPr, which incurs significant computational overhead due to its CNN-based pre-pruning method, LTP-ViT leverages the benefits of a pre-pruning strategy while overcoming the limitation by employing a far simpler, yet effective pre-pruning method.

LTP-ViT determines token importance by integrating semantic redundancy and spatial importance perspectives into a unified metric called the Spatial-Redundancy Balance (SRB) score. In the SRB score, the semantic redundancy is estimated via cosine similarity and the spatial importance is estimated via the L_2 norm of positional embedding vectors. By evaluating tokens based on this SRB score in a single step, our method can identify and discard unimportant tokens with minimal computational cost. This ”front-loading” of the pruning process ensures that the subsequent self-attention operations, which are the most computationally expensive parts of the ViT, are performed only on a small set of important tokens.

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) (RS-2024-00409492)

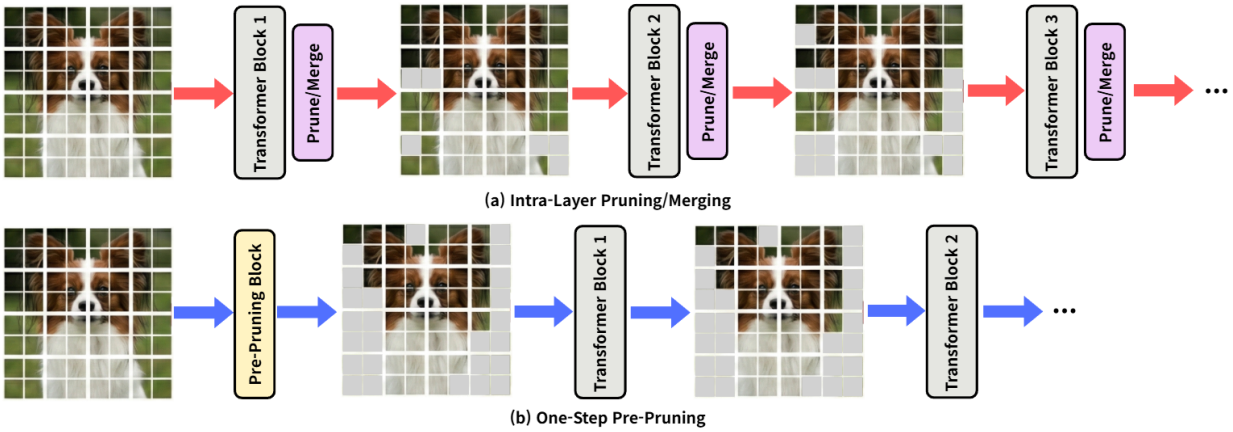

Fig. 1. The pipeline contrast: Intra-layer pruning/merging vs One-step pre-pruning

LTP-ViT is specifically tailored for real-time applications where user experience hinges on sub-millisecond response, or resource-constrained edge devices where operating with a small batch size is practically required. Unlike conventional pruning methods that incur significant overhead, LTP-ViT is engineered to be highly efficient even under minimal batch configurations, maximizing throughput where it matters most. As a training-free, one-step solution, LTP-ViT can be integrated seamlessly with diverse ViT models, providing a plug-and-play framework for real-world deployment without requiring any fine-tuning.

Experimental results on ImageNet-1K demonstrate that LTP-ViT not only reduces the number of operations but also improves inference speed. Specifically, LTP-ViT outperforms the existing ToMe and PaPr in terms of latency by 1.4× to 2.6×, respectively, on an NVIDIA RTX 3090, with negligible accuracy drop, demonstrating that high efficiency can be achieved without having to have complex pre-pruning modules. Furthermore, LTP-ViT achieves an 8.0× reduction in pre-pruning latency and 30% memory savings on edge devices such as NVIDIA Jetson Xavier NX, demonstrating its suitability for resource-constrained devices. The key contributions of this work are summarized as follows.

- A training-free, one-step pre-pruning mechanism: We propose the SRB score that integrates semantic redundancy and spatial importance perspectives, thereby eliminating the need for additional training or fine-tuning (Section III).
- Superior inference efficiency: We demonstrate significant practical gains across diverse platforms, including a 2.6× end-to-end speedup in servers and an 8.0× reduction in pre-pruning latency on edge devices compared to existing baselines (Section IV).
- A practical solution for memory optimization: Through extensive evaluation, we verify that LTP-ViT effectively reduces peak memory usage by 30% on edge devices, providing a robust ”plug-and-play” module for resource-constrained edge applications (Section IV).

Consequently, LTP-ViT provides a highly practical solution for deploying ViTs in real-time applications where every millisecond of the inference time is critical.

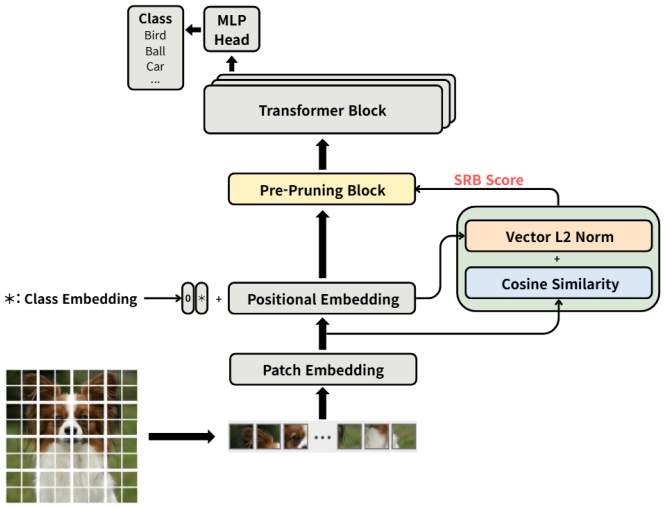

Fig. 2. Overview of the LTP-ViT framework within the Vision Transformer (ViT) pipeline.

II. Related Work

A. Vision Transformers

The landscape of computer vision has undergone a significant paradigm shift, transitioning from the long-standing dominance of Convolutional Neural Networks (CNNs) to the emergence of Vision Transformers (ViTs). For decades, CNNs were the standard due to their inductive biases [11], [12], such as translation invariance and locality, which are effective for processing grid-like image data. However, the fixed receptive fields of CNNs often limit their ability to capture long-range dependencies across an entire image.

To overcome these limitations, Dosovitskiy et al. introduced the Vision Transformer [1], which treats image patches as tokens, similar to words in Natural Language Processing. Subsequently, DeiT [13] enhanced the practical viability of

ViTs by introducing a teacher-student distillation strategy that allows for data-efficient training on smaller datasets. By leveraging the Self-Attention mechanism, ViTs can model global context from the very first layer, consistently setting new benchmarks across various tasks, including image classification, object detection, and segmentation. Despite their superior representation capability, the quadratic computational complexity of self-attention—relative to the number of input tokens—becomes a critical bottleneck, hindering their deployment in real-time inference scenarios.

B. Intra-Layer Token Pruning and Merging

To mitigate the high computational costs associated with a large number of tokens, several redundancy reduction strategies have been explored, primarily categorized into Token Pruning and Token Merging. Token Pruning focuses on identifying and removing "unimportant" tokens (e.g., background pixels) based on attention scores or importance metrics. While this effectively lowers computational costs, aggressive pruning may lead to irreversible loss of information.

On the other hand, ToMe [2] is a widely known Token Merging method that aims to preserve information by combining similar tokens rather than keeping one and discarding the rest. ToMe utilizes a bipartite matching algorithm to fuse redundant tokens within each Transformer block. However, the practical utility of ToMe is often limited; the matching and the fusion processes introduce significant latency overhead, sometimes negating the speedup gained from token reduction during the actual inference.

C. Pre-Pruning Method

Recent studies for pre-pruning, such as PaPr [10], have attempted to find the best trade-off between accuracy and efficiency by employing a pre-pruning module based on some CNN models. In PaPr, by utilizing feature maps from a CNN model, MobileOne-S0, a map called Patch Significance Map (PSM) is generated to prune unimportant tokens. But employing an additional CNN model at the front end incurs significant operational complexity. For deployment in edge devices or servers, there exists a pressing need for a simpler, one-step approach that eliminates redundant computation without the heavy overhead of token management.

III. Proposed Method: LTP-ViT

We propose LTP-ViT that leverages the cosine similarities of tokens, representing semantic redundancy, and the L_2 norm of positional embedding vectors, which serves as an indicator of spatial importance (See Fig. 2.). LTP-ViT minimizes deployment overhead by eliminating the need for additional training. Furthermore, the entire pruning process is executed in a single step, ensuring high efficiency and seamless integration into the model pipeline. LTP-ViT not only minimizes inference latency but also reduces peak memory usage compared to existing techniques. This makes it exceptionally efficient for both server systems and resource-constrained devices.

A. Identification of Semantically Redundant Tokens

To identify redundant information within an input image, we compute the cosine similarity between embedding tokens. Let $\mathbf{X} \in \mathbb{R}^{N \times D}$ be the patch-embedded tokens, where N is the number of tokens and D is the embedding dimension. We first normalize each patch embedding token to the unit length:

$$\hat{x}_i = x_i / (\|x_i\|_2 + \epsilon). \tag{1}$$

The cosine similarity matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ is computed using the batch matrix multiplication of the normalized embeddings: $\mathbf{S} = \hat{\mathbf{X}}\hat{\mathbf{X}}^\top$. Each element $s_{i,j}$ represents the cosine similarity between the i^{th} and the j^{th} tokens. To evaluate the redundancy of a specific token i, we identify its maximum similarity to any other token in the same image, excluding self-similarity:

$$R_i = \max_{j \neq i}(s_{i,j}); \quad (i = 0, 1, \ldots, N). \tag{2}$$

$R \in \mathbb{R}^N$ and R_i is the token-wise max similarity score. A high R_i value indicates that the i^{th} token is semantically similar to at least one other token, suggesting that its information might be redundant and thus expendable without significant loss of the global context.

B. L_2 Norm of Positional Embedding Vector

To capture the inherent spatial significance of each token, we leverage the trained positional embeddings $\mathbf{P} \in \mathbb{R}^{N \times D}$ as a spatial importance. In Vision Transformers, positional embeddings $\mathbf{p}_i$ are added to patch embeddings to provide the model with information about the coordinates of each token. We hypothesize that the magnitude of these learned vectors reflects the model's sensitivity or priority assigned to specific spatial locations within the image grid. The spatial importance I_i for a token i is defined by the L_2 norm of its corresponding positional embedding vector:

$$I_i = \|\mathbf{p}_i\|_2 = \sqrt{\sum_{d=1}^{D} p_{i,d}^2}; \quad (i = 0, 1, \ldots, N). \tag{3}$$

$I \in \mathbb{R}^N$ and I_i is the token-wise spatial importance score. Essentially, a higher L_2 norm regards a token as a spatially critical component. By utilizing this metric, we can identify important regions without incurring significant computational overhead.

Fig. 3 shows a positional embedding vector heatmap based on the L_2 norm. Although the L_2 norm heatmaps for both models share the commonality of higher values being concentrated in the central region, their overall distribution patterns are not identical. This suggests that these values reflect not only the characteristics of the training dataset but also the inherent structural properties of each specific model architecture.

C. SRB Score: Balancing Spatial Priority and Redundancy

The final token selection process aims to preserve tokens that are both spatially significant and semantically unique (See Fig. 4.). While semantic redundancy (R_i) identifies tokens

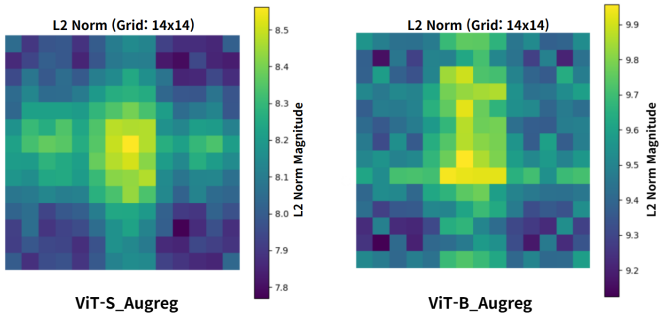

Fig. 3. Positional embedding vector heatmap using the L_2 norm

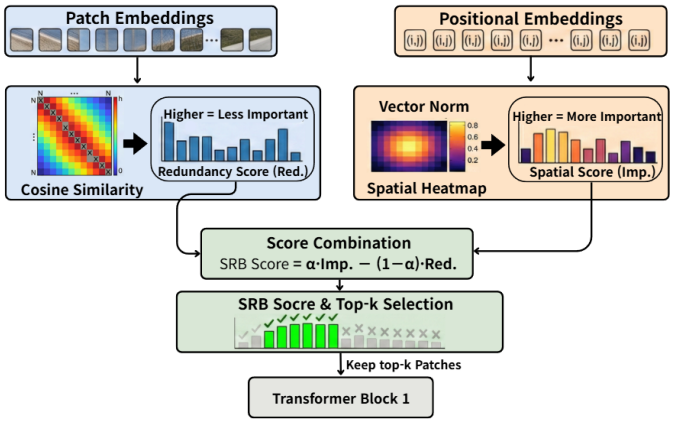

Fig. 4. Architecture of the LTP-ViT pre-pruning block

with overlapping information, the spatial importance (I_i) ensures that the structural integrity of an image is preserved. To combine these two distinct metrics, we first apply min-max normalization (Norm) to align their scales into a unified range $[0, 1]$. The SRB score S_i is then calculated as follows:

$$S_i = \alpha \cdot \text{Norm}(I_i) - (1-\alpha) \cdot \text{Norm}(R_i); (i = 0, 1, \ldots, N). \quad (4)$$

$S \in \mathbb{R}^N$ and S_i is the token-wise SRB score and α is a balancing hyperparameter. Through empirical evaluation, we set $\alpha = 0.5$ to provide the best balance between these two factors.

The interpretation of this score is two-fold: (i) Positive component ($\alpha \cdot I_i$) prioritizes tokens located in regions the model deems spatially important based on its learned positional weights, (ii) Negative component ($(1-\alpha) \cdot R_i$) penalizes tokens that are highly redundant and can be represented by other tokens. This synergy addresses the limitations of using each metric alone: I_i preserves the global structure but may retain redundant background, whereas R_i eliminates redundancy but risks losing essential isolated-tokens. This combination allows for mutual compensation, leading to superior pruning accuracy. Subsequently, LTP-ViT prunes a predefined ratio of tokens with the lowest S_i values. This one-step, training-free pre-pruning mechanism allows the model to significantly reduce the number of tokens before processing the first Transformer block, thereby minimizing the computational overhead of the self-attention layers.

IV. Experiments

We conducted inference experiments on the ImageNet-1K validation set using ViT-Augreg and ViT-MAE models. For the experiments, we used a single NVIDIA GeForce RTX 3090 GPU for server evaluations and an NVIDIA Jetson Xavier NX 8GB board for edge device evaluations.

A. Performance on Server Environment

Table I summarizes the performance of LTP-ViT compared to existing token reduction techniques, ToMe and PaPr, on the server environment. We evaluated each method using ViT-S-Augreg, ViT-B-Augreg, and ViT-B-MAE models when processing similar GFLOPs (Giga Floating-Point Operations)

TABLE I
Comparison of accuracy and latency at a similar GFLOPs for ViT models (batch-size=1, server environment)

Models	Methods	Accuracy	GFLOPs	Latency(ms)
ViT-S-Augreg	Baseline	81.39	4.24	3.73
	ToMe	77.0	2.29	9.86
	PaPr	77.21	2.308	5.35
	LTP-ViT (Ours)	76.84	2.34	3.8
ViT-B-Augreg	Baseline	84.54	16.85	4.29
	ToMe	80.44	8.78	9.85
	PaPr	82.09	8.8	5.33
	LTP-ViT (Ours)	81.01	8.89	3.82
ViT-B-MAE	Baseline	83.74	16.85	4.16
	ToMe	78.88	8.78	9.26
	PaPr	82.07	8.81	5.28
	LTP-ViT (Ours)	81.76	8.9	3.86

to ensure a fair comparison. The results demonstrate that LTP-ViT achieves a superior trade-off between accuracy and end-to-end latency than the compared methods. In the case of ViT-S-Augreg, while all three methods reduced GFLOPs by approximately 46%, LTP-ViT achieved a latency of 3.80 ms, which is $2.6\times$ faster than ToMe (9.86 ms) and $1.4\times$ faster than PaPr (5.35 ms). Notably, ToMe and PaPr showed significantly higher latency than the baseline (3.73 ms) despite the reduction in GFLOPs, mainly due to the computational overhead introduced by the matching-based merging and the CNN-based pre-pruning process, respectively. A similar trend was observed in ViT-B-Augreg. LTP-ViT not only maintained competitive Top-1 accuracy (81.01%) but also recorded the lowest latency (3.82 ms). Remarkably, in the larger ViT-B model, LTP-ViT outperformed the baseline's latency (4.29 ms). These results highlight that LTP-ViT is highly effective for actual hardware acceleration compared to other approaches.

In the case of ViT-B-MAE, LTP-ViT shows a slight decrease in accuracy compared to PaPr (81.76% vs. 82.07%). However, LTP-ViT significantly outperforms ToMe (78.88%), which suffers from a substantial accuracy drop. Most notably, LTP-ViT achieves the lowest latency of 3.86 ms. This is in stark contrast to ToMe and PaPr, which exhibit considerably higher latency than the baseline despite their reduction in

TABLE II
COMPARISON OF ACCURACY WITH SIMILAR THROUGHPUTS FOR THE VIT-S-AUGREG MODEL (BATCH-SIZE=64, SERVER ENVIRONMENT)

Models	Methods	Accuracy	Throughput(img/s)
ViT-S-Augreg	Baseline	81.39	1355
	ToMe	75.33	2232
	PaPr	75.38	2219
	LTP-ViT (Ours)	78.99	2234

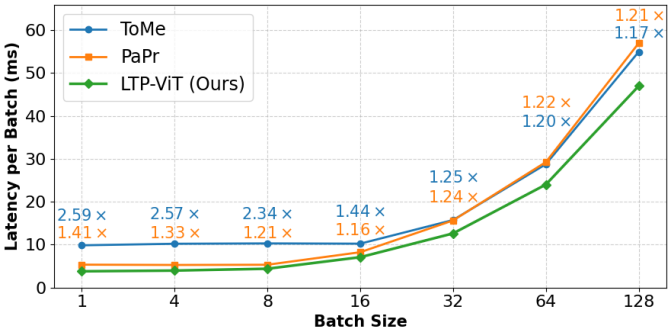

Fig. 5. Latency comparison across various batch sizes (speedup of LTP-ViT over ToMe and PaPr)

GFLOPs. These results demonstrate that LTP-ViT provides the most effective balance between computational complexity and actual inference speed.

As shown in Table II, LTP-ViT outperforms the other pruning techniques in terms of accuracy-throughput balance in a high-throughput server environment (batch size 64). While ToMe and PaPr achieve throughputs of 2232 and 2219 img/s, respectively, they suffer from significant accuracy drops. In contrast, LTP-ViT maintains a much higher accuracy of 78.99%, marking a 3.6% improvement over the compared methods while achieving a comparable throughput of 2234 img/s. This performance gap underscores the superiority of LTP-ViT as an effective token pruning solution, proving its capability to preserve critical semantic information more effectively than the others.

B. Latency with Various Batch Sizes on Server Environment

As illustrated in Fig. 5, on the ViT-S-Augreg model with a computational amount of 2.3 GFLOPs (as in the setting of Table I), LTP-ViT consistently outperforms both ToMe and PaPr in terms of per-batch latency across all tested batch sizes. Notably, the performance gap is most pronounced at smaller batch sizes; for instance, at a batch size of 1, LTP-ViT achieves a significant speedup, being $2.59\times$ and $1.41\times$ faster than ToMe and PaPr, respectively.

C. Latency and Peak Memory Usage on Edge Environment

To further evaluate the accuracy and the latency of LTP-ViT on a resource-constrained edge device, we conducted similar experiments on an NVIDIA Jetson Xavier NX with 8GB. The latency evaluation was performed using the ViT-S-Augreg model with a batch size of 1 to mimic real-time inference scenarios. For a fair comparison, the accuracy and

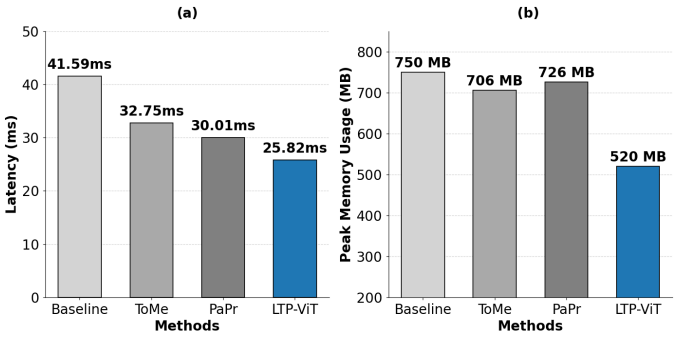

Fig. 6. (a) Latency Comparison on Jetson Xavier NX (b) Peak Memory Usage Comparison on Jetson Xavier NX

the latency of all evaluated methods—ToMe, PaPr, and LTP-ViT—were measured with similar computational amounts of approximately 2.3 GFLOPs, as in the setting of Table I.

As illustrated in Fig. 6 (a), LTP-ViT consistently outperforms the existing token pruning techniques in the edge platforms. While the relative latency gap is slightly narrower than that observed in the server environment—as Transformer block computations dominate the bottleneck on edge devices—our approach still achieves significant speedups. Specifically, LTP-ViT recorded a latency of 25.82 ms, which corresponds to a 21% and a 14% latency reduction compared to ToMe (32.75 ms) and PaPr (30.01 ms), respectively. These results confirm that LTP-ViT achieves significant latency improvements even in resource-limited edge devices, proving its robustness across diverse computing platforms.

In addition to latency improvements, memory efficiency is a critical requirement for deploying deep learning models on resource-constrained edge hardware. To evaluate this, we measured the peak memory usage of the ViT-S-Augreg model on the NVIDIA Jetson Xavier NX 8GB with a batch size of 64. To ensure a fair comparison, all models were evaluated with a similar computational amount of 2.3 GFLOPs. High batch sizes often lead to out-of-memory (OOM) errors in edge environments, making memory optimization essential for stable inference.

As illustrated in Fig. 6 (b), LTP-ViT demonstrates significantly better results in terms of memory requirement than the compared methods. While the baseline requires 750 MB of peak memory, and the compared other methods, ToMe and PaPr achieve only modest reductions (recording 706 MB and 726 MB, respectively), LTP-ViT substantially reduces the memory requirement to 520 MB. This represents a reduction of approximately 30% relative to the baseline. These results verify that LTP-ViT is exceptionally well-suited for edge devices where memory capacity is a primary constraint. By significantly lowering the peak memory usage, LTP-ViT not only avoids OOM issues during high-load processing but also allows for more efficient resource allocation, facilitating the deployment of robust Vision Transformers in real-world edge applications.

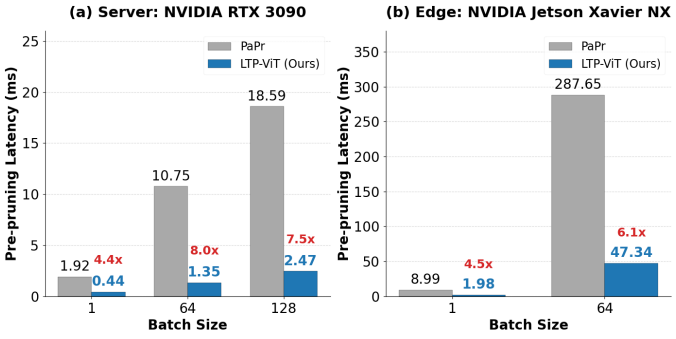

Fig. 7. Pre-pruning latency comparison on both the server and the edge environments

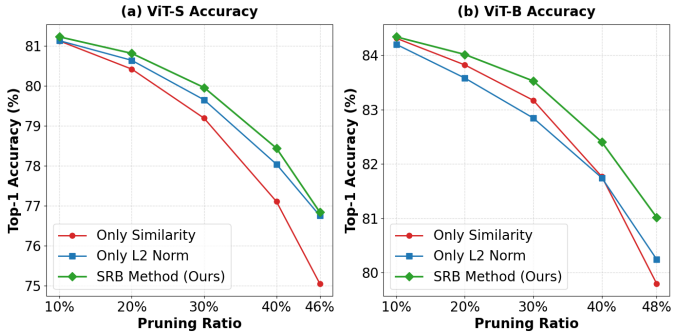

Fig. 8. Synergistic effect of combining token similarity and the L_2 norm on the model accuracy.

D. Comparison of the Pre-pruning Latency

To investigate the main reason for LTP-ViT's superior speed, we analyzed pre-pruning latency on ViT-S-Augreg with a computational amount of 2.3 GFLOPs for both the server and the edge environments with diverse batch sizes. As shown in Fig. 7, LTP-ViT consistently demonstrates significantly lower computational overhead and better scalability than PaPr. On the server, LTP-ViT achieves speedups of 4.4× to 8.0×, effectively mitigating the pre-processing bottleneck. This efficiency is also pronounced on the NVIDIA Jetson Xavier NX, where LTP-ViT maintains a manageable 47.34 ms with a batch size of 64 (6.1× faster than PaPr's 287.65 ms). These results confirm that LTP-ViT's theoretical GFLOPs reduction successfully translates into actual hardware acceleration, making it highly suitable for both high-throughput servers and resource-constrained edge devices.

E. Ablation Study

To evaluate the synergistic effects of the redundancy score and the spatial score, we conducted an ablation study that compares the use of (i) only the semantic redundancy, (ii) only the spatial importance, and (iii) our SRB score that considers both. As shown in Fig. 8, the SRB score consistently achieves the highest Top-1 accuracy across all pruning ratios for both ViT-S and ViT-B. By integrating these two indicators, the SRB score makes it possible to compensate for each other's shortcomings, creating synergistic effects.

V. Discussion

While our method shows strong performance on ImageNet-1K, its effectiveness on dense datasets like COCO, which feature multiple objects and scale variations, remains to be explored. Future research will focus on these scenarios.

VI. Conclusion

In this paper, we propose a novel, training-free one-step lightweight token pre-pruning strategy for Vision Transformers called LTP-ViT to overcome the potential latency bottlenecks of the existing pre-pruning methods. By utilizing a new metric called the SRB score that integrates semantic redundancy and spatial importance, LTP-ViT can identify and remove redundant tokens before the first Transformer block is processed. Our experiments on ImageNet-1K, LTP-ViT offers a superior accuracy-latency trade-off compared to existing methods like ToMe and PaPr. In server environments, LTP-ViT achieves a 3.6% accuracy improvement with significantly lower latency at similar GFLOPs compared to existing methods. On an NVIDIA Jetson Xavier NX, it reduces peak memory usage by 30% (520 MB), effectively mitigating out-of-memory risks on edge devices. Crucially, LTP-ViT's pre-pruning is up to 8.0× and 6.1× faster than PaPr on server and edge platforms, respectively, proving its exceptional scalability and hardware friendliness. These findings underscore that LTP-ViT is highly effective for the real-world deployment of ViTs, where every millisecond of the inference time and every megabyte of the memory usage are crucial.

References

[1] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021.
[2] D. Bolya *et al.*, "Token merging: Your ViT but faster," in *ICLR*, 2023.
[3] M. Kim, S. Gao, Y.-C. Hsu, Y. Shen, and H. Jin, "Token fusion: Bridging the gap between token pruning and token merging," *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1372–1381, 2023.
[4] Y. Rao *et al.*, "Dynamicvit: Efficient vision transformers with dynamic token sparsification," in *NeurIPS*, pp. 13937–13949, 2021.
[5] Y. Liang *et al.*, "Evit: Expediting vision transformers via token reorganizations," in *ICLR*, 2022.
[6] H. Yin *et al.*, "A-vit: Adaptive tokens for efficient vision transformer," in *CVPR*, pp. 10799–10808, IEEE, 2022.
[7] Fayyaz *et al.*, "ATS: Adaptive Token Sampling For Efficient Vision Transformers," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 396–414, 2022.
[8] X. Xu *et al.*, "Gtp-vit: Eficient vision transformers via graph-based token propagation," *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 86–95, 2023.
[9] H. Wang, B. Dedhia, and N. K. Jha, "Zero-tprune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers," in *CVPR*, pp. 16070–16079, IEEE, 2024.
[10] T. Mahmud, B. Yaman, C. Liu, and D. Marculescu, "PaPr: training-free one-step patch pruning with lightweight ConvNets for faster inference," in *ECCV (23)*, vol. 15081 of *Lecture Notes in Computer Science*, pp. 110–128, Springer, 2024.
[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, pp. 1106–1114, 2012.
[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, pp. 770–778, IEEE Computer Society, 2016.
[13] H. Touvron *et al.*, "Training data-efficient image transformers & distillation through attention," in *ICML*, vol. 139 of *Proceedings of Machine Learning Research*, pp. 10347–10357, PMLR, 2021.