Edge AI Efficiency: Evaluating Transformer Compression Methods for Limited-Resource Platforms

Cheng Zheng[1], Jian Li[2]

[1] Hunan University, Hunan, China. zhangc@hnu.edu.cn
[2]Peking University, Beijing, China. jianli@pku.edu.cn

Abstract

Recent progress in image analysis has been largely driven by neural networks inspired by language processing systems. These architectures have shown superior performance across various applications compared to traditional methods. However, their substantial processing and storage needs present challenges for use in constrained settings. Our research addresses these issues by investigating four key strategies to streamline these models: data representation reduction, matrix simplification, model emulation, and structural trimming. We conduct a thorough evaluation of these approaches, examining their effects both separately and jointly, to enhance the models' suitability for limited-resource environments. Our extensive tests demonstrate that these techniques effectively balance performance and efficiency, making these advanced neural networks more practical for deployment in edge devices and similar constrained computational contexts.

Keywords: Model Compression, Edge Computing, Vision Transformers

DOI: 10.5281/zenodo.13831494
DOI URL: https://zenodo.org/doi/10.5281/zenodo.13831493

1 Introduction

The emergence of Transformer architectures, pioneered by Vaswani et al. [1], has ushered in a new era of artificial intelligence. These innovative models have redefined the boundaries of machine learning with their unparalleled ability to process complex data structures and capture intricate relationships across vast information landscapes. Initially conceived for natural language tasks, Transformers quickly demonstrated their versatility, achieving remarkable success in areas such as text generation and language understanding [2].

The adaptability of Transformer models soon caught the attention of computer vision researchers. This led to groundbreaking applications in image analysis [3] and scene interpretation [4], challenging the long-standing dominance of convolutional neural networks in visual processing tasks. The fusion of linguistic and visual paradigms opened up exciting new possibilities for AI development.

One of the most intriguing developments in recent years has been the application of Transformers to multimodal sentiment analysis [5]. This cutting-edge field integrates diverse data types - text, images [6], and audio [7] to decipher the subtle nuances of human emotion [8]. By synthesizing information from multiple sensory channels, these systems are pushing the boundaries of machine emotional intelligence.

The impact of Transformers has extended far beyond academic research, making significant inroads into practical applications, particularly in healthcare [9]. These models are revolutionizing medical practices, from enhancing early disease detection [10] to transforming the interpretation of complex medical imagery [11]. The ability of Transformers to process and analyze vast amounts of patient data is paving the way for more accurate diagnoses and personalized treatment plans [12].

While Transformers have shown remarkable potential across various domains, they face unique challenges in different applications. In natural language processing, researchers like Child et al. [13] have grappled with the computational complexity of attention mechanisms for long sequences. Conversely, Vision Transformers (ViTs) deal with different constraints, processing fixed-size image patches and presenting distinct optimization challenges.

The introduction of Vision Transformers [14] marked a paradigm shift in computer vision. By treating images as sequences of patches and applying self-attention mechanisms, ViTs have achieved state-of-the-art performance on numerous visual tasks. This approach has challenged traditional convolutional architectures and opened up new avenues for image understanding.

However, the impressive capabilities of ViTs come at a cost. These models often contain an enormous number of parameters, leading to significant computational and memory requirements. This presents a major hurdle for deployment on resource-constrained devices or in applications requiring real-time processing. As a result, there is a growing need for effective model compression techniques specifically tailored to Vision Transformers.

Unlike more established neural network architectures, the field of ViT compression remains largely unexplored. This paper aims to address this gap by conducting a comprehensive investigation into various compression strategies for Vision Transformers. We focus on four key approaches: quantization, low-rank approximation, knowledge distillation, and pruning. Through extensive experimentation and analysis, we evaluate the effectiveness of these techniques both individually and in combination.

Our research delves into the potential synergies between different compression methods, exploring how they can be optimally combined to maximize efficiency gains while preserving model accuracy. The findings presented here offer valuable insights for researchers and practitioners looking to deploy Vision Transformers in real-world scenarios where computational resources are limited.

This study not only contributes to the growing body of knowledge on Vision Transformer optimization but also provides practical guidelines for implementing these powerful models across a wide range of applications. As AI continues to evolve, the ability to efficiently deploy sophisticated architectures like ViTs will play a crucial role in unlocking their full potential and driving innovation in fields ranging from autonomous systems to advanced medical diagnostics.

2 Related work

2.1 Quantization

The field of neural network optimization has witnessed significant advancements, with quantization emerging as a pivotal strategy for enhancing inference efficiency. This technique involves the transformation of high-precision network parameters into low-bit representations, effectively reducing both computational complexity and memory requirements while striving to maintain model performance. A crucial aspect of quantization is the determination of an optimal clipping range for weights. Various approaches have been proposed to address this challenge, with Krishnamoorthi [15] advocating for a layer-wise evaluation of convolutional filter weights, while Shen et al. [16] introduce a novel groupwise quantization method specifically tailored for Transformer architectures. As researchers delve deeper into quantization techniques, the potential trade-off between model compression and accuracy preservation has become a central focus. To mitigate potential performance degradation, Quantization-Aware Training (QAT) has emerged as a promising solution. This innovative approach integrates quantization directly into the training process, allowing the model to adapt to lower precision representations gradually. During QAT, the network performs forward and backward passes using floating-point simulations of the quantized model, followed by a re-quantization step after each parameter update. This iterative process enables the model to learn robust representations that are resilient to the discretization effects of quantization.

2.2 Low-rank Approximation

The Vision Transformer (ViT) architecture has revolutionized computer vision tasks by leveraging the powerful self-attention mechanism, but this comes at the cost of quadratic computational complexity, posing significant challenges for scalability. Research by Chen et al. [17] has revealed an intriguing characteristic of ViTs: their attention matrices inherently exhibit low-rank properties [18], opening up new avenues for complexity reduction [19]. This discovery has spurred the development of various low-rank matrix approximation techniques aimed at optimizing the computational efficiency of ViTs [20]. Among these, Nyström-based methods [21], Performer [22], and Linformer [23] stand out as innovative approaches, each offering unique advantages and compatibility with pre-trained ViT models during fine-tuning and validation stages. The field has further evolved with the exploration of hybrid techniques, such as the combination of low-rank approximation and sparse attention mechanisms, which has demonstrated enhanced approximation quality and improved overall efficiency of ViTs. These advancements collectively represent a significant step forward in addressing the computational demands of Vision Transformers, paving the way for their wider adoption in resource-constrained environments and real-time applications, while maintaining their impressive performance on complex visual tasks.

2.3 Knowledge Distillation

In the realm of model compression, knowledge distillation has emerged as a sophisticated technique that leverages the expertise of a complex 'teacher' model to train a more compact 'student' model. This process hinges on the use of soft labels generated by the teacher, which contain a wealth of nuanced information that often surpasses the utility of traditional hard labels [24]. The effectiveness of this approach in enhancing student model performance has been substantiated by the works of Yuan et al. [25] and Wei et al. [26]. A significant innovation in this field, particularly for Vision Transformers, is the concept of a distillation token introduced by Touvron et al. [27]. This specialized token, while similar in principle to the class token, is uniquely designed to capture and integrate the teacher's predictions through the self-attention mechanism, thereby refining the distillation process. Such tailored strategies have demonstrated remarkable improvements over conventional distillation methods, highlighting the potential for developing transformer-specific optimization techniques. As research in this area progresses, we can anticipate further advancements in knowledge distillation methodologies, potentially leading to even more efficient and powerful compact models that can be deployed across a wide range of applications, from mobile devices to large-scale AI systems.

2.4 Pruning

In the realm of Vision Transformer optimization, pruning has emerged as a highly effective technique for reducing model complexity while maintaining performance [28]. This approach involves a meticulous process of assigning importance scores to various dimensions within the model, enabling the strategic removal of less crucial components. By carefully balancing the pruning ratio with accuracy preservation, researchers aim to create leaner, more efficient ViT architectures without sacrificing their powerful capabilities. Yang et al. [29] have proposed an innovative extension to this method, introducing the concept of dimensional redistribution, which can be seamlessly integrated into the pruning process to further enhance model efficacy. Interestingly, empirical evidence suggests that pruned models can occasionally outperform their original, unpruned counterparts [30], hinting at the transformative potential of pruning beyond mere simplification. This phenomenon underscores the intricate relationship between model architecture and performance, suggesting that judicious pruning may not only streamline ViTs but also unlock hidden potential, potentially leading to more robust and generalized representations. As the field continues to evolve, pruning techniques are likely to play an increasingly pivotal role in the development of next-generation Vision Transformers, enabling their deployment across a wider range of applications and hardware configurations.

3 Methodology

3.1 Quantization

3.1.1 Basic Concept

The overarching objective of quantization is to reduce the precision of model parameters (ϑ) and intermediate activation maps to a lower precision format, such as 8-bit integers, while minimizing the impact on the model's generalization performance. The initial step in this process involves defining a quantization function capable of mapping weights and activations to a discrete set of values. A commonly utilized function for this purpose is delineated as follows:

$$Q(r) = \mathrm{Int}(r/S) - Z, \tag{1}$$

where Q represents the quantization mapping function, r denotes a real-valued input (e.g., weights, activation), S is a scaling factor, and Z is an integer zero point. This mechanism, known as *uniform quantization*, ensures the equidistant spacing of resultant values. It's noteworthy that alternative *non-uniform quantization* strategies exist. Moreover, the original real value r can be approximated from its quantized counterpart $Q(r)$ through a process known as *dequantization*:

$$r^{\sim} = S(Q(r) + Z), \tag{2}$$

where the approximation $r^{\sim}$ may differ from r due to rounding errors inherent in quantization.

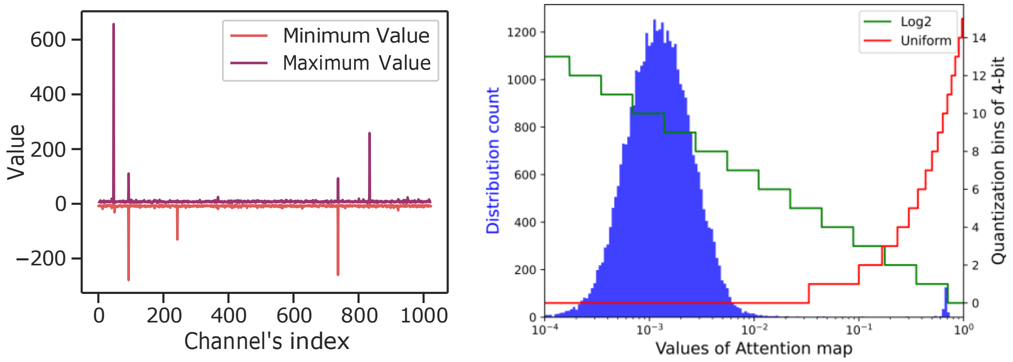
A critical aspect of quantization is determining the optimal scaling factor S, which effectively partitions real values r into discrete segments:

$$S = \frac{\beta - \alpha}{2^b - 1}, \tag{3}$$

with $[\alpha, \beta]$ representing the clipping range and b denoting the bit width of quantization. The selection of the clipping range $[\alpha, \beta]$, a process termed as *calibration*, is pivotal. A straightforward method involves employing the minimum and maximum of the inputs as the clipping range, i.e., $\alpha = r_{\min}$ and $\beta = r_{\max}$, corresponding to an *asymmetric quantization* scheme where $-\alpha \neq \beta$. Alternatively, a *symmetric quantization* approach, where $-\alpha = \beta = \max(|r_{\max}|, |r_{\min}|)$, can be employed. In such cases, the quantization function in Eq. 1 can be simplified by setting $Z = 0$.

3.1.2 Post Training Quantization

Post Training Quantization (PTQ) streamlines the quantization process by adjusting weights directly, without necessitating further fine-tuning. This efficiency, however, may lead to notable accuracy declines due to the inherent precision loss of quantization. Liu et al. [31] observed substantial accuracy reductions when applying quantization to LayerNorm and Softmax layers within Transformer architectures. Lin et al. [32] attributed these discrepancies to the polarized distribution of activation values in LayerNorm layers and attention map values. Specifically, significant inter-channel variability within LayerNorm layer inputs (as illustrated on the left side of Figure 1) induces considerable quantization errors when employing layer-wise quantization approaches. Moreover, a predominance of small-value distributions in attention maps—with only sparse outliers approaching a value of 1—further exacerbates performance declines under uniform quantization strategies. Addressing these challenges, Lin et al. [32] introduced a novel quantization approach employing Powers-of-Two Scale for LayerNorm and Log-Int-Softmax for Softmax layers, aiming to mitigate the adverse effects of traditional quantization methods.



(a) Data Distribution in LayerNorm

(b) Data Distribution in Attention Maps

Figure 1: **Left [32]**: Channel-wise minimum and maximum values of the last LayerNorm inputs in full precision Swin-B. **Right [32]**: Distribution of the attention map values from the first layer of ViT-L, and visualizing the quantized bins using uniform or Log2 quantization with 4-bit.

3.1.3 Quantization Aware Training

Applying quantization directly to a fully trained model can inadvertently perturb model parameters, leading to notable performance declines. An effective strategy to circumvent this issue involves re-training the model with quantized parameters, thereby guiding the model towards a more favorable loss landscape. Quantization Aware Training (QAT) stands out as a prominent technique for this purpose. In QAT, the standard forward and backward processes are executed on a model represented in floating-point, yet parameters are re-quantized following each gradient update, ensuring the model adapts to quantization-induced changes.

Learned Step Size Quantization (LSQ) [33], an advancement in this field, refines the quantizer configuration process and has set new benchmarks in quantization performance by optimizing the quantization intervals. Similarly, DIFFQ [34] introduces a differentiable model compression technique that eschews the need for gradient approximation methods such as the Straight Through Estimator (STE). By employing pseudo quantization noise, DIFFQ achieves an approximation of the quantization process during training that is fully differentiable, thereby facilitating more nuanced adjustments to both the weights and quantization bit-depth.

3.2 Knowledge Distillation

Knowledge distillation techniques, such as soft and hard distillation, facilitate the transfer of knowledge from a complex 'teacher' model to a simpler 'student' model. Soft distillation focuses on minimizing the Kullback-Leibler (KL) divergence between the softened logits (or outputs) of both the teacher and student models. This is formally captured by the distillation objective:

$$L\text{global} = (1-\lambda)L\text{CE}\,(\psi(Z_s), y) + \lambda\tau^2\text{KL}\;\;\psi\;\;\frac{Z_s}{\tau}\;,\psi\;\;\frac{Z_t}{\tau}\;\;\;, \tag{4}$$

where LCE denotes the cross-entropy loss, ψ represents the softmax function, Zt and Z_s are the logits from the teacher and student models, respectively, τ is the temperature parameter enhancing softness of distributions, and λ balances the contributions of the KL divergence and the cross-entropy loss.

Conversely, hard distillation uses the teacher's predictions as definitive labels for training the student, simplifying the process by directly comparing the student's predictions against these labels:

$$L\text{global}^{\text{hardDistill}} = \frac{1}{2}L\text{CE}\,(\psi(Z_s), y) + \frac{1}{2}L\text{CE}\,(\psi(Z\text{s}), y_t)\,, \tag{5}$$

where y_t = argmaxcZt(c) represents the hard label decision by the teacher model.

The DeiT [27] method introduces a novel approach specific to Transformers, incorporating a 'distillation token' into the architecture that functions analogously to the class token but focuses on mimicking the teacher's predictions. This mechanism allows for a direct interaction between the distillation token and other components through the self-attention layers, demonstrating superior performance in distillation. Our experimental setup involves applying the DeiT framework for knowledge distillation on the CIFAR dataset, adjusting for computational resource constraints.

3.3 Pruning

Pruning in Vision Transformers focuses primarily on reducing the model's complexity by decreasing the number of parameters, specifically by adjusting the dimensions of weight kernels between hidden layers. This objective can be formalized as:

$$\min \alpha, \beta$$

$$\text{s.t.} \sum_k loss(I_\beta^{(k)} W_{\alpha,\beta}^{(k)} I_\alpha^{(k+1)}) - loss(I_b^{(k)} W_{a,b}^{(k)} I_a^{(k+1)}) < \delta \tag{6}$$

where a, b represent the original dimensions of $W^{(k)}$, and α, β are the reduced dimensions post-pruning. The goal is to ensure the incremental loss incurred from this reduction remains below a predefined threshold δ, preserving the integrity of the model for subsequent tasks. Determining which dimensions to prune involves the use of importance scores, a concept learned either during pre-training or fine-tuning. Zhu et al. [28] and Yang et al. [29] derive these scores from the gradient magnitude of each weight, proposing the integration of a "soft gate" layer post-pruning which hardens to zero-out less critical dimensions during inference:

$$s_B(W) = \sum_{b \in B} \left(\frac{\partial s}{\partial w_b} w_b \right)^2 . \tag{7}$$

Alternatively, Yu et al. [30] employ KL divergence to calculate importance scores, focusing on the divergence between model performances with and without specific modules across a dataset Ω. This method facilitates both within-layer and across-module pruning:

$$S_B(W) = \sum_{i \in \Omega} D_{KL}(p_i || q_i) \tag{8}$$

where q_i corresponds to the loss with the full model, and p_i to the loss sans the pruned weights. Recent innovations have introduced even more nuanced importance scoring systems. Tang et al. [35] devised a score reflecting the theoretical impact of each patch on the final error, enhancing patch slimming efficiency. Rao et al. [36] combined local and global features for a more holistic assessment of token significance. Similarly, Yi et al. [37] synthesized various scores into a unified loss function, further refining the pruning process.

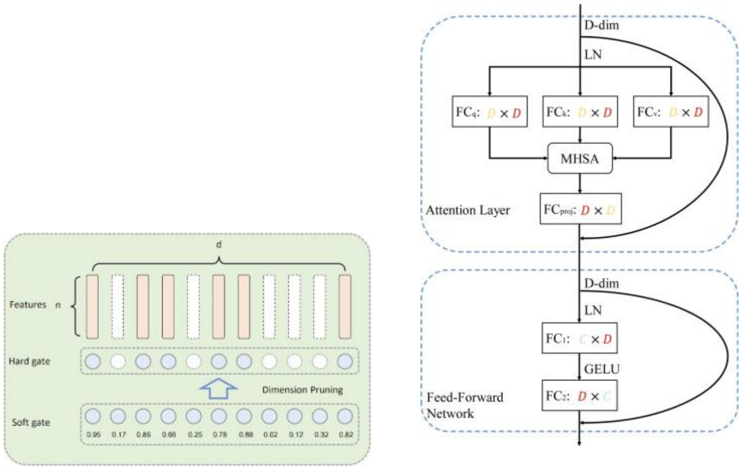
3.4 Low-rank Approximation

In Vision Transformers (ViTs), each self-attention block initially projects an input sequence X using weights W_Q, W_K, and W_V to obtain feature representations $Q = W_Q X$, $K = W_K X$, and $V = W_V X$. The self-attention mechanism, computed as $softmax(QK^T / \sqrt{d_q})V$, introduces computational and spatial complexities of $O(n^2)$, where n is the sequence length.

Given the formal proof of self-attention's low-rank nature, leveraging this property for low-rank approximation emerges as a strategic choice to enhance computational efficiency. Such approximations aim to preserve accuracy while significantly reducing both time and space complexity to approximately $O(n)$, even when integrating with pre-existing or newly trained models.

Notably, low-rank approximation does not inherently reduce the model size, as the original weights W_Q, W_K, W_V are retained. However, it does offer substantial reductions in computational time and memory usage, particularly during fine-tuning or validation phases for pre-trained models. This is because the approximation calculations are performed subsequent to input reception.

Various methodologies for low-rank approximation exist, including Nyström-based approaches like Nyströmformer and SOFT, which linearize self-attention through the Nyström method. Alternative linearization techniques, such as Linformer and Performer, along with strategies integrating both low-rank and sparse attention mechanisms, further enhance approximation accuracy.



(a) Soft-gate based model proposed by Zhu et. al. [28]

(b) Skip-dimension model proposed by Yu et. al. [30]

Figure 2: Different pruning models

Our experimental focus is on the Nyströmformer-based ViT, adapting the softmax attention matrix calculation to utilize the Nyström method across all self-attention blocks. This allows for the use of pre-trained Vanilla ViT weights, facilitating direct comparisons in performance. The Nyströmformer technique employs landmark points for an efficient approximation, circumventing the need for full QK^T calculations. We evaluate the efficacy of this approach with landmark counts (m) set at 24, 32, and 64, assessing its impact on model performance.

4 Experiments

This section delineates a thorough comparative analysis of various model compression techniques applied to Vision Transformers, including quantization, knowledge distillation, pruning, and low-rank approximation. Additionally, we investigate the synergistic potential of combining these methods to ascertain enhancements in performance metrics.

4.1 Experimental Settings

The experimental framework is established on a Tesla V100-SXM2 16GB GPU, with PyTorch serving as the primary platform for code implementation[1]. The scope of our dataset utilization is confined to CIFAR-10 and CIFAR-100, attributed to computational resource constraints. Our primary metrics of interest include model size and inference speed, acknowledging the inherent trade-off between accuracy and these efficiency parameters. An optimal compression technique is thus characterized by minimal impact on accuracy coupled with substantial reductions in model size and enhancements in inference speed. Results of the comparative analysis across CIFAR-10 and CIFAR-100 datasets are systematically presented in Table 1 and Table 2.

4.2 Comparison of Different Model Compression Methods

In assessing the impact of model compression on **Model Size**, we find that quantization and pruning strategies offer substantial size reductions with minimal accuracy compromise. Notably, quantization techniques, particularly Dynamic Quantization[2], have demonstrated superior efficacy, reducing model size to 25

Contrarily, weight pruning, particularly with simplistic importance scoring, does not facilitate an optimal balance between model size and accuracy. A pruning rate of 0.1 (indicating 10% of parameters

[1] It is pertinent to note that PyTorch's current support for quantization is limited to CPU-based operations, necessitating CPU-based inference speed tests for certain methodologies.

[2] Dynamic Quantization was implemented using the PyTorch Quantization API: https://pytorch.org/tutorials/advanced/dynamic_quantization_tutorial.html

Model	Method	Accuracy	GPU Speed	CPU Speed	Size(MB)
Vanilla ViT [14]	-	98.94	4.48	0.050	327
Dynamic Quantization	Quantization (PTQ)	98.73	-	0.062	84
FQ-ViT [32]	Quantization (PTQ)	97.31	-	-	-
DIFFQ with LSQ [33]	Quantization (QAT)	93.37	2.10	-	41
DIFFQ with diffq [34]	Quantization (QAT)	60.29	12.20	-	2
DeiT base [27]	Knowledge Distillation	98.47	7.04	0.096	327
DeiT tiny [27]	Knowledge Distillation	95.43	16.78	-	21
ViT-Pruning(r=0.1) [28]	Pruning	88.36	4.86	-	301
ViT-Pruning(r=0.2) [28]	Pruning	80.56	5.54	-	254
ViT-Nyströmformer(m=24) [21]	Low-rank Approximation	65.91	4.67	-	327
ViT-Nyströmformer(m=32) [21]	Low-rank Approximation	75.94	4.57	-	327
ViT-Nyströmformer(m=64) [21]	Low-rank Approximation	91.70	4.38	-	327
DeiT base + Dynamic Quantization	Knowledge Distillation + PTQ	96.75	-	0.117	84

Table 1: Evaluation results on CIFAR-10. The Speed values are iterations per second.

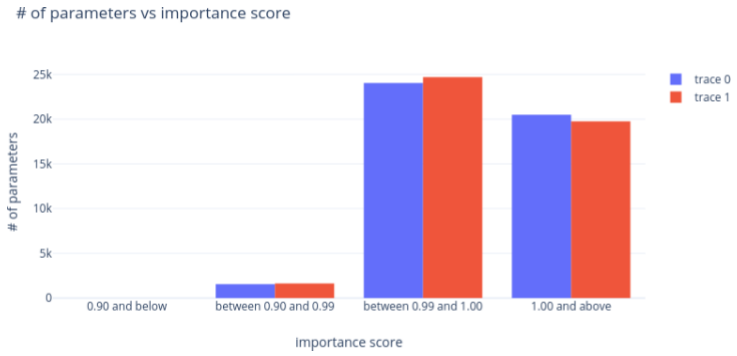


Figure 3: Number of parameters vs importance score. **Blue**: CIFAR-10. **Red**: CIFAR-100.

pruned) led to a significant accuracy reduction in both CIFAR-10 and CIFAR-100 datasets compared to the unpruned ViT. Further investigation, as depicted in figure 3, reveals that a majority of parameters are deemed critically important (scores above 0.99), suggesting inherent limitations in simple form importance scoring for weight pruning. Enhancements could stem from the integration of more sophisticated importance scores [30] or adopting strategies like input patch reduction or slimming, as opposed to direct weight pruning [36, 35].

For **Inference Speed**, a spectrum of enhancements is observed across different model compression strategies, with methods centered around knowledge distillation particularly standing out for their efficiency gains. Notably, the DeiT base model, despite not undergoing significant size reduction, achieves an inference speed nearly double that of the standard Vision Transformer (ViT), all while preserving accuracy to a remarkable degree. An intriguing case is observed with the DeiT tiny configuration on the CIFAR-10 dataset, where it attains 95.43% accuracy—a figure closely aligned with the Vanilla ViT—yet delivers a quadruple increase in speed and is compressed to merely 6% of the original model's size.

Furthermore, the application of Nyströmformer-based techniques to ViT illustrates a nuanced balance between accuracy and speed, particularly influenced by the selection of the number of landmarks (m). Opting for a larger m value enhances the precision of approximations at the expense of processing velocity. Additionally, Dynamic Quantization contributes to inference speed improvements in the range of 10-20% on CPU platforms, underscoring the practical benefits of model compression beyond just reductions in size.

Model	Method	Accuracy	GPU Speed	CPU Speed	Size(MB)
Vanilla ViT [14]	-	92.87	4.34	0.093	327
Dynamic Quantization	Quantization (PTQ)	90.87	-	0.122	84
FQ-ViT [32]	Quantization (PTQ)	84.87	-	-	-
DIFFQ with LSQ [33]	Quantization (QAT)	76.08	2.10	-	41
DIFFQ with diffq [34]	Quantization (QAT)	41.02	12.00	-	2
DeiT base [27]	Knowledge Distillation	87.35	6.97	0.149	327
DeiT tiny [27]	Knowledge Distillation	75.90	16.16	-	21
ViT-Pruning(r=0.1) [28]	Pruning	74.46	4.69	-	302
ViT-Pruning(r=0.2) [28]	Pruning	64.27	5.19	-	272
ViT-Nyströmformer(m=24) [21]	Low-rank Approximation	38.51	4.77	-	327
ViT-Nyströmformer(m=32) [21]	Low-rank Approximation	50.31	4.65	-	327
ViT-Nyströmformer(m=64) [21]	Low-rank Approximation	74.01	4.46	-	327
DeiT base + Dynamic Quantization	Knowledge Distillation + PTQ	82.61	-	0.196	84

Table 2: Evaluation results on CIFAR-100. The Speed values are iterations per second.

4.3 Exploration of Mixed Methods

The examination of individual model compression techniques suggests that a hybrid approach, leveraging the strengths of both quantization and knowledge distillation, warrants further investigation. Particularly, when a slight decrease in accuracy is acceptable, such a combined strategy appears promising for optimizing both model compactness and processing efficiency. As demonstrated in Tables 1 and 2, employing a composite method—integrating the DeiT base model with Dynamic Quantization—significantly enhances inference speed, achieving more than a twofold increase, while concurrently reducing the model's size to one-fourth of its original dimensions. This is achieved with a manageable trade-off in accuracy, highlighting the potential of mixed approaches in striking a balanced compromise between speed, size, and performance.

5 Conclusion

This study has been dedicated to an empirical investigation of model compression techniques aimed at enhancing the efficiency and deployment viability of Vision Transformers (ViTs). We meticulously examined four predominant compression methods—quantization, low-rank approximation, knowledge distillation, and pruning—complemented by a review of cutting-edge research in the field. Through comparative analyses conducted on the CIFAR-10 and CIFAR-100 datasets, our findings underscore the efficacy of post-training quantization and knowledge distillation as standout strategies. These methods not only significantly reduce model size but also expedite inference times, all while maintaining acceptable levels of performance degradation. Further exploration into the synergistic potential of combining quantization and knowledge distillation has revealed a compelling avenue for optimization. Particularly evident within the CIFAR-10 dataset, this hybrid approach markedly accelerated inference speeds—surpassing baseline speeds by more than a factor of two—while concurrently diminishing model size to merely a quarter of its initial footprint. The insights garnered from this comprehensive examination advocate for a holistic, multi-faceted approach to model compression. Integrating diverse compression methodologies holds substantial promise for refining the operational efficiency of Vision Transformers, heralding a robust direction for future research in this domain. Moreover, this paper can be further integrated in the field of machine learning, transportation engineering, biological engineering, etc.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[2] Yehui Tang, Kai Han, Yunhe Wang, Chang Xu, Jianyuan Guo, Chao Xu, and Dacheng Tao. Patch slimming for efficient vision transformers. ArXiv, abs/2106.02852, 2021.

[3] Xueting Pan, Ziqian Luo, and Lisang Zhou. Navigating the landscape of distributed file systems: Architectures, implementations, and considerations. arXiv preprint arXiv:2403.15701, 2024.

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

[5] Raghuraman Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper. arXiv preprint arXiv:1806.08342, 2018.

[6] Ziqian Luo and Xueting Pan. Visual question generation on vqa dataset. 2024.

[7] Steven K Esser, Jeffrey L McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dhar- mendra S Modha. Learned step size quantization. arXiv preprint arXiv:1902.08153, 2019.

[8] Lisang Zhou, Ziqian Luo, and Xueting Pan. Machine learning-based system reliability analysis with gaussian process regression. arXiv preprint arXiv:2403.11125, 2024.

[9] Longhui Wei, An Xiao, Lingxi Xie, Xiaopeng Zhang, Xin Chen, and Qi Tian. Circumventing outliers of autoaugment with knowledge distillation. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16, pages 608–625. Springer, 2020.

[10] Yifan Wu, Min Gao, Min Zeng, Jie Zhang, and Min Li. Bridgedpi: a novel graph neural network for predicting drug–protein interactions. *Bioinformatics*, 38(9):2571–2578, 2022.

[11] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. arXiv preprint arXiv:2006.04768, 2020.

[12] Feiyang Chen and Ziqian Luo. Sentiment analysis using deep robust complementary fusion of multi-features and multi-modalities. CoRR, 2019.

[13] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[14] 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, 2021.

[15] Ziqian Luo. Knowledge-guided aspect-based summarization. In 2023 International Conference on Communications, Computing and Artificial Intelligence (CCCAI), pages 17–22. IEEE, 2023.

[16] Sheng Shen, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Q-bert: Hessian based ultra low precision quantization of bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8815–8821, 2020.

[17] Beidi Chen, Tri Dao, Eric Winsor, Zhao Song, Atri Rudra, and Christopher Ré. Scatterbrain: Unifying sparse and low-rank attention approximation, 2021.

[18] Feiyang Chen and Ziqian Luo. Learning robust heterogeneous signal features from parallel neural network for audio sentiment analysis. arXiv preprint arXiv:1811.08065, 2018.

[19] Yunyang Xiong, Zhanpeng Zeng, Rudrasis Chakraborty, Mingxing Tan, Glenn Fung, Yin Li, and Vikas Singh. Nyströmformer: A nystöm-based algorithm for approximating self-attention.

In Proceedings of the... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence, volume 35, page 14138. NIH Public Access, 2021.

[20] Jiachen Lu, Jinghan Yao, Junge Zhang, Xiatian Zhu, Hang Xu, Weiguo Gao, Chunjing Xu, Tao Xiang, and Li Zhang. Soft: Softmax-free transformer with linear complexity, 2021.

[21] Ziqian Luo, Hua Xu, and Feiyang Chen. Audio sentiment analysis by heterogeneous signal features learned from utterance-based parallel neural network. In AffCon@ AAAI, pages 80–87, 2019.

[22] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[23] Xueting Pan, Ziqian Luo, and Lisang Zhou. Comprehensive survey of state-of-the-art convolutional neural network architectures and their applications in image classification. Innovations in Applied Engineering and Technology, pages 1–16, 2022.

[24] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[25] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021.

[26] Feiyang Chen, Ziqian Luo, Lisang Zhou, Xueting Pan, and Ying Jiang. Comprehensive survey of model compression and speed up for vision transformers. arXiv preprint arXiv:2404.10407, 2024.

[27] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[28] Mingjian Zhu, Kai Han, Yehui Tang, and Yunhe Wang. Visual transformer pruning. *arXiv preprint arXiv:2104.08500*, 2021.

[29] Huanrui Yang, Hongxu Yin, Pavlo Molchanov, Hai Li, and Jan Kautz. Nvit: Vision transformer compression and parameter redistribution. *arXiv preprint arXiv:2110.04869*, 2021.

[30] Ziqian Luo, Hua Xu, and Feiyang Chen. Utterance-based audio sentiment analysis learned by a parallel combination of cnn and lstm. arXiv preprint arXiv:1811.08065, 2018.

[31] Zhenhua Liu, Yunhe Wang, Kai Han, Wei Zhang, Siwei Ma, and Wen Gao. Post-training quantization for vision transformer. *Advances in Neural Information Processing Systems*, 34, 2021.

[32] Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. Fq-vit: Fully quantized vision transformer without retraining. *arXiv preprint arXiv:2111.13824*, 2021.

[33] Feiyang Chen, Ziqian Luo, Yanyan Xu, and Dengfeng Ke. Complementary fusion of multi-features and multi-modalities in sentiment analysis. arXiv preprint arXiv:1904.08138, 2019.

[34] Alexandre Défossez, Yossi Adi, and Gabriel Synnaeve. Differentiable model compression via pseudo quantization noise. *arXiv preprint arXiv:2104.09987*, 2021.

[35] Ziqian Luo, Xiangrui Zeng, Zhipeng Bao, and Min Xu. Deep learning-based strategy for macromolecules classification with imbalanced data from cellular electron cryotomography. In 2019 International Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE, 2019.

[36] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34, 2021.

[37] Shixing Yu, Tianlong Chen, Jiayi Shen, Huan Yuan, Jianchao Tan, Sen Yang, Ji Liu, and Zhangyang Wang. Unified visual transformer compression. In *International Conference on Learning Representations*, 2022.