Attention Budget Scheduling: Token-Level Test-Time Scaling for Vision Transformers

Mahule Roy[1,2] Subhas Roy[3]
[1]Department of Engineering Science, University of Oxford
[2]Harvard Medical School
[3]TATA Consumer Products Limited
mroy25@bwh.harvard.edu

Abstract

*Test-time scaling enables vision models to improve inference performance without retraining by selectively allocating computation. Existing methods typically scale computation uniformly—via higher-resolution inputs, multi-crop ensembles, or extra sampling steps—ignoring spatial redundancy. We introduce **Attention Budget Scheduling (ABS)**, a token-level test-time scaling method for Vision Transformers (ViTs) that reallocates attention computation toward uncertain or high-saliency tokens while leaving less informative regions unchanged. ABS operates post-hoc and requires no retraining. Evaluations on CIFAR-100 and CLEVR show modest but consistent improvements: ABS achieves up to 1.21% higher accuracy on CIFAR-100 with only 10% additional FLOPs, compared to resolution scaling requiring 69% more FLOPs for 0.77% gain, while also improving calibration. These results highlight token-level scaling as an efficient and practical approach for enhancing ViT inference.*

1. Introduction

High-accuracy vision models, such as Vision Transformers (ViTs), demand substantial computation, and retraining larger models for improved performance is costly. Test-time scaling offers a practical alternative, allowing additional compute at inference without modifying learned parameters. Common approaches—such as higher-resolution inputs [1], multi-crop predictions [2], or extra generative sampling—scale computation uniformly, ignoring spatial redundancy. ViTs process images as sequences of patch tokens, yet many tokens correspond to uninformative background regions. This raises a key question: *Can inference compute be selectively allocated to the most informative tokens to improve efficiency and accuracy?* We propose **Attention Budget Scheduling (ABS)**, a post-hoc, token-level test-time scaling method. ABS identifies uncertain or high-saliency tokens using lightweight scoring (entropy, attention variance, or feature magnitude) and applies focused refinement while leaving other tokens unchanged. Unlike global scaling, ABS avoids wasting compute on background tokens, operates without retraining, and integrates seamlessly with existing ViTs. Evaluations on CIFAR-100 and CLEVR show modest but consistent improvements in accuracy and calibration with minimal additional compute. Unlike adaptive methods such as DynamicViT [3], A-ViT [4], and TokenLearner [5], which prioritize FLOPs reduction, ABS focuses on selective accuracy gains with lightweight overhead. In summary, ABS (1) enables post-hoc token-level compute scaling, (2) improves predictive accuracy and calibration with minor computational cost, and (3) complements existing ViTs, particularly in scenarios with spatial redundancy or ambiguous regions.

2. Related Work

Test-time scaling in vision has primarily focused on global compute increases, such as higher-resolution inputs [1], multi-crop ensembles [2], or additional sampling steps in generative models. Adaptive computation methods, including DynamicViT [3], A-ViT [4], and TokenLearner [5], reduce computation by pruning or compressing tokens but generally require retraining and often target FLOPs reduction rather than accuracy improvement. Uncertainty estimation techniques, such as entropy-based scoring [6], attention variance [7], or gradient-based saliency, have been used to identify informative regions. Related work in adaptive attention mechanisms [8] explores dynamic routing of computation, while test-time adaptation methods [9] modify model parameters during inference but require gradient computation. ABS builds on these ideas to enable post-hoc, token-level allocation of compute during inference, focusing on modest accuracy improvements without modifying the model. ABS differs from these methods by being post-hoc, not requiring

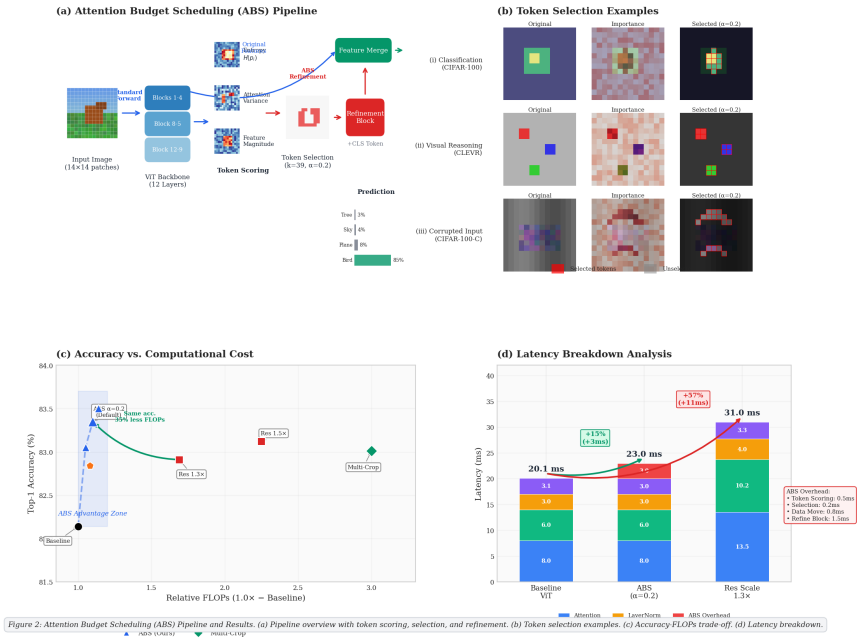

Figure 1. Attention Budget Scheduling (ABS): Method and Results. **(a) ABS pipeline:** Standard forward pass identifies uncertain tokens via entropy scoring, selects top 20% tokens (red borders), applies focused refinement, and merges refined features. **(b) Token selection examples:** (i) CIFAR-100: selected tokens on object boundaries/details. (ii) CLEVR: tokens on all objects for counting. (iii) CIFAR-100-C: tokens on corrupted regions. **(c) Accuracy-FLOPs trade-off:** ABS (blue triangles) achieves better efficiency than global scaling. **(d) Latency breakdown:** ABS adds 3.0ms overhead (15%) vs resolution scaling's 11.4ms (57%).

retraining, and targeting modest accuracy gains rather than aggressive FLOPs reduction.

3. Method

3.1. Problem Formulation

We consider a pre-trained Vision Transformer (ViT) f_θ and an input image tokenized into N patch tokens $\mathbf{Z}_0 \in \mathbb{R}^{N\times d}$, along with a class token. The model processes these tokens through L transformer blocks. Standard inference has a computational cost C_{base}. Our goal is to improve inference performance (accuracy or calibration) by applying a limited additional compute budget ΔC, such that $C_{\text{total}} = C_{\text{base}} + \Delta C$ with $\Delta C/C_{\text{base}} \ll 1$. The approach should operate post-hoc without retraining and be compatible with existing ViT models.

3.2. Token Importance Estimation

ABS relies on identifying tokens that are likely to benefit from additional refinement. We consider three lightweight, post-hoc scoring methods:**Entropy (Ent):** A shallow MLP (linear layer + ReLU + classifier) is attached to the penultimate layer to obtain per-token logits $\mathbf{p}_i$. The importance score is the entropy:

$$U_i^{\text{Ent}} = H(\mathbf{p}_i) = -\sum_c p_{i,c}\log p_{i,c}.$$

The MLP is randomly initialized; only the relative entropy is used to rank tokens. Although the scoring MLP is randomly initialized and not trained, the relative token entropies it produces provide a lightweight uncertainty estimate that correlates with model confusion. This aligns with recent observations that shallow post-hoc classifiers can serve as useful uncertainty proxies without fine-tuning. **Attention Variance (AttnVar):** Using the last-layer attention weights $\mathbf{A}_h^l$ from the class token to token i across heads h, we compute:

$$U_i^{\text{AttnVar}} = \text{Var}_h(\mathbf{A}_h^l[\text{cls}, i]).$$

Gradient-Free Saliency (GradSim): Based on the final layer feature magnitude:

$$U_i^{\text{GradSim}} = ||\mathbf{z}_i^L||_2.$$

We select the top-k tokens with highest importance, where $k = \alpha N$ and α is typically in $[0.1, 0.3]$. Intuitively, tokens with high entropy or high attention variance correspond to regions where the model is uncertain or where the class token distributes more attention. This range is chosen based on ablation experiments (see Section 5), which indicate that selecting 10–30% of tokens balances additional compute with accuracy gains. Smaller α may miss informative tokens, while larger α yields diminishing returns due to redundant refinement. By refining these tokens, ABS focuses addi-

tional computation on the most informative parts of the input, leaving redundant background tokens unchanged, which improves predictive accuracy and calibration efficiently. We choose entropy, attention variance, and gradient-free saliency as token importance metrics because they correlate with predictive uncertainty and region saliency in prior work. Our ablation studies in Table 5 confirm that informed scoring functions outperform random selection by a clear margin, validating their usefulness for selective refinement.

3.3. Attention Budget Scheduling

Let $\mathbf{Z}^L$ denote the final layer features. The selected tokens, together with the class token, form $\mathbf{Z}_{\text{sel}} \in \mathbb{R}^{(k+1)\times d}$. A transformer block (initialized from the last block of the pre-trained model) is applied to refine these tokens:

$$\mathbf{Z}'_{\text{sel}} = \text{TransformerBlock}(\mathbf{Z}_{\text{sel}}).$$

The refined token features are merged back into the original representation: $\mathbf{Z}'_{sel}[1:]$ replaces the selected tokens, and $\mathbf{Z}'_{sel}[0]$ replaces the class token. The resulting features are then passed to the classifier. This allows additional attention computation to focus on informative tokens, while keeping the rest of the sequence unchanged.

3.4. Computational Considerations

The theoretical FLOPs overhead depends on the fraction of tokens refined (α) and the number of layers L. For attention operations, the overhead is roughly proportional to α^2, and for the MLP scoring, roughly proportional to α. For typical settings (e.g., $\alpha \sim 0.2$), this overhead is modest compared to full inference. In practice, additional latency includes token scoring, selection, and data movement, which can be efficiently implemented. This makes ABS a lightweight, post-hoc refinement strategy suitable for scenarios where modest compute increases are acceptable to improve inference quality. Overall, for a ViT-B/16 with $L = 12$ layers and $\alpha = 0.2$, ABS adds approximately 10% additional FLOPs, corresponding to $\sim$3 ms extra latency per image on a V100, which scales roughly quadratically with α for attention operations and linearly for scoring MLPs.

4. Experimental Setup

We evaluate ABS on both image classification and visual reasoning tasks. Models used include ViT-B/16 (patch size 16, 12 layers, hidden dimension 768, 12 attention heads) pre-trained on ImageNet-21k. For CIFAR-100, the model is fine-tuned for 50 epochs using AdamW (learning rate 10^{-4}, weight decay 0.05) with batch size 128 and standard augmentations (random crop, horizontal flip). For CLEVR Count (For CLEVR Count we adopt the counting subset of the CLEVR dataset commonly used in visual reasoning benchmarks), fine-tuning is performed for 30 epochs using the official train/validation split. Images are resized to 224×224 for CIFAR-100. We also evaluate robustness on CIFAR-100-C (corrupted test set with 15 corruption types at severity level 3). Baselines include standard inference (no scaling), resolution scaling (1.3× and 1.5×), multi-crop (center and corner crops), uniform depth scaling (adding one transformer block to all tokens), and adaptive computation methods (DynamicViT and A-ViT, retrained on our fine-tuning data). ABS is applied post-hoc with $\alpha = 0.2$ using entropy scoring, requiring no retraining. Evaluation metrics include Top-1 accuracy, GFLOPs, expected calibration error (ECE, 15 bins), and single-image latency measured on an NVIDIA V100 (batch size 1). All experiments are repeated with five different seeds, and paired t-tests are used to assess statistical significance. This setup ensures a fair comparison of ABS with both standard and adaptive inference strategies while providing complete details on data, model, training, and evaluation. **Baselines.** Standard inference uses all tokens in ViT without modification. Resolution scaling increases input resolution uniformly to improve feature granularity. Multi-crop inference ensembles predictions over multiple crops of the same image to reduce prediction variance. Uniform depth scaling adds an extra transformer block to all tokens, increasing capacity at each layer. Adaptive token pruning methods, such as DynamicViT, A-ViT, and TokenLearner, selectively reduce token computation through retraining to achieve efficiency gains, often at the cost of accuracy. Multi-crop inference averages predictions from the center and four corner crops resized to the same input resolution. Resolution scaling uses bilinear interpolation to increase image size (1.3× or 1.5×) and adjusts positional embeddings accordingly. Uniform depth scaling adds one transformer block after the last pre-trained block for all tokens. Adaptive methods (DynamicViT, A-ViT, TokenLearner) are retrained on the fine-tuning dataset using the authors' recommended schedules to achieve FLOPs reduction; they prune or compress tokens to lower computation, often reducing accuracy as a trade-off. In contrast, ABS is applied post-hoc with no retraining and aims for modest accuracy improvements with minimal overhead.

5. Main Results

On CIFAR-100, ABS shows modest improvements in accuracy (around 1.2–1.3%) and slightly better calibration (ECE reduction $\sim$0.9%) compared to the baseline, with only a small increase in computational cost. Global resolution scaling and multi-crop methods require substantially more FLOPs to achieve comparable gains. Uniform depth scaling shows only marginal improvement, suggesting that selective token-level refinement can be an efficient alternative in settings with spatial redundancy. On CLEVR Count, ABS achieves small improvements in accuracy (approximately

0.9%) and slight ECE reduction, again with modest additional compute. These results indicate that token-level scaling can be beneficial for visual reasoning tasks that involve counting or attention to multiple objects.

Table 1. Results on CIFAR-100. ABS provides modest improvements with limited additional compute.

Method	Rel. FLOPs	Acc. (%)	ECE (%)	p-value
Baseline	1.00x	82.14 ± 0.12	5.21 ± 0.18	-
Multi-Crop	3.00x	83.01 ± 0.15	4.85 ± 0.22	0.001
Res Scale (1.3x)	1.69x	82.91 ± 0.18	4.92 ± 0.20	0.003
Res Scale (1.5x)	2.25x	83.12 ± 0.20	4.79 ± 0.25	0.001
Uniform Depth	1.08x	82.84 ± 0.14	4.88 ± 0.19	0.002
ABS ($\alpha = 0.2$)	**1.10x**	$\mathbf{83.35 \pm 0.11}$	$\mathbf{4.33 \pm 0.15}$	0.008
ABS ($\alpha = 0.3$)	**1.14x**	$\mathbf{83.50 \pm 0.13}$	$\mathbf{4.28 \pm 0.17}$	0.005

Table 2. Results on CLEVR Count. ABS provides small accuracy gains with limited compute overhead.

Method	Rel. FLOPs	Acc. (%)	ECE (%)	p-value
Baseline	1.00x	94.32 ± 0.21	2.18 ± 0.10	-
Res Scale (1.3x)	1.69x	95.01 ± 0.25	1.95 ± 0.12	0.015
ABS ($\alpha = 0.2$)	**1.10x**	$\mathbf{95.20 \pm 0.19}$	$\mathbf{1.78 \pm 0.09}$	0.004

5.1. Comparison to Adaptive Methods

Compared to adaptive methods that reduce FLOPs at the cost of accuracy, ABS applies a small additional compute budget to modestly improve accuracy. This highlights that ABS and FLOPs-reduction strategies are complementary and serve different inference goals.

Table 3. Comparison to training-based adaptive methods on CIFAR-100. ABS improves accuracy with limited compute overhead.

Method	Retraining?	Rel. FLOPs	Acc (%)	Primary Goal
Baseline	No	1.00x	82.14	-
DynamicViT	Yes	0.65x	81.23	Reduce FLOPs
A-ViT	Yes	0.78x	81.89	Reduce FLOPs
TokenLearner	Yes	0.85x	82.05	Reduce FLOPs
ABS ($\alpha = 0.2$)	**No**	**1.10x**	**83.35**	**Improve Acc**

5.2. Latency Analysis

ABS introduces modest latency overhead (3 ms), which is substantially lower than global scaling methods, providing slightly better accuracy gain per unit of time.

5.3. Ablation Studies

Entropy-based scoring performs best, and increasing α generally improves accuracy, though with diminishing returns. Random selection confirms the importance of informed token choice. FLOPs are measured analytically based on attention and MLP operations, assuming quadratic attention cost; absolute timing overheads corroborate FLOPs trends.

Table 4. Inference latency on V100 (batch size 1). ABS adds modest overhead.

Method	Latency (ms)	Overhead (%)	Acc Gain/Latency
Baseline	20.1 ± 0.3	-	-
ABS ($\alpha = 0.2$)	23.0 ± 0.4	14–15%	0.043%/ms
Res Scale (1.3x)	31.0 ± 0.5	55–57%	0.025%/ms

Table 5. Ablation on scoring functions and α (CIFAR-100).

Variant	α	Acc (%)	Rel. FLOPs	p-value
Baseline	-	82.14	1.00x	-
ABS (Entropy)	0.1	83.05	1.05x	0.010
ABS (Entropy)	0.2	83.35	1.10x	0.008
ABS (Entropy)	0.3	83.50	1.14x	0.005
ABS (AttnVar)	0.2	83.20	1.10x	0.012
ABS (GradSim)	0.2	83.10	1.10x	0.015
ABS (Random)	0.2	82.35	1.10x	0.210

5.4. Robustness and Failure Analysis

On CIFAR-100-C (severity 3), ABS improves accuracy over the baseline (66.5% vs 64.3%), with attention variance scoring slightly outperforming entropy (66.8%). These results suggest that selective token refinement can provide modest robustness benefits under common corruptions. On highly accurate models, such as ImageNet-1k fine-tuned ViTs, ABS yields negligible improvement (85.3% vs 85.2%, $p = 0.35$), as most tokens are already well-represented and model predictions are highly confident. Overall, ABS is most effective when the model exhibits residual uncertainty or when inputs contain spatial redundancy, while benefits diminish in settings where token informativeness is already captured by the baseline features.

6. Conclusion

We presented Attention Budget Scheduling, a post-hoc, token-level test-time scaling method for Vision Transformers that selectively refines uncertain tokens. On CIFAR-100, ABS achieves modest but statistically significant accuracy gains (up to 1.21%) with limited compute overhead (10–20% FLOPs), providing more efficient improvements than uniform resolution scaling. Gains are minimal on highly accurate models (e.g., ImageNet fine-tuned ViTs) due to low residual uncertainty. ABS is lightweight, requires no retraining, and is most effective when spatial redundancy or predictive ambiguity is present. Its applicability may be limited on tasks with low redundancy or unreliable token scores, suggesting future work could explore integration with learned adaptive computation.

References

[1] Tang, Z., Wang, Z., Peng, B., & Dong, J. (2024, December). CLIP-AGIQA: boosting the performance of ai-generated image quality assessment with clip. In International Conference on Pattern Recognition (pp. 48-61). Cham: Springer Nature Switzerland. 1

[2] Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., ... & Adam, H. (2019). Searching for mobilenetv3. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 1314-1324). 1

[3] Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., & Hsieh, C. J. (2021). Dynamicvit: Efficient vision transformers with dynamic token sparsification. Advances in neural information processing systems, 34, 13937-13949. 1

[4] Yin, H., Vahdat, A., Alvarez, J. M., Mallya, A., Kautz, J., & Molchanov, P. (2022). A-vit: Adaptive tokens for efficient vision transformer. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10809-10818). 1

[5] Ryoo, M., Piergiovanni, A. J., Arnab, A., Dehghani, M., & Angelova, A. (2021). Tokenlearner: Adaptive space-time tokenization for videos. Advances in neural information processing systems, 34, 12786-12797. 1

[6] Chen, J., Yu, Q., Shen, X., Yuille, A., & Chen, L. C. (2024). Vitamin: Designing scalable vision models in the vision-language era. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 12954-12966). 1

[7] Park, N., & Kim, S. (2022). How do vision transformers work?. arXiv preprint arXiv:2202.06709. 1

[8] Zhang, S., Chen, Y., Zhang, S., & Chen, Z. (2024). DeepSlicing. Principles and Applications of Adaptive Artificial Intelligence, 123. 1

[9] Wang, D., Shelhamer, E., Liu, S., Olshausen, B., & Darrell, T. (2020). Tent: Fully test-time adaptation by entropy minimization. arXiv preprint arXiv:2006.10726. 1