Research Parhway

THE UNIVERSITY OF SYDNEY

Class-Attention-Guided Adaptive Token Clustering for Efficient Vision Transformers

Vision Transformers, Token Compress, lightweight

Presented by

Kai Ding

Supervised by

Dr. Karlos Ishac

FACULTY OF ENGINEERING

Question:

How many patches do you really need to **recognize** a spoon ?

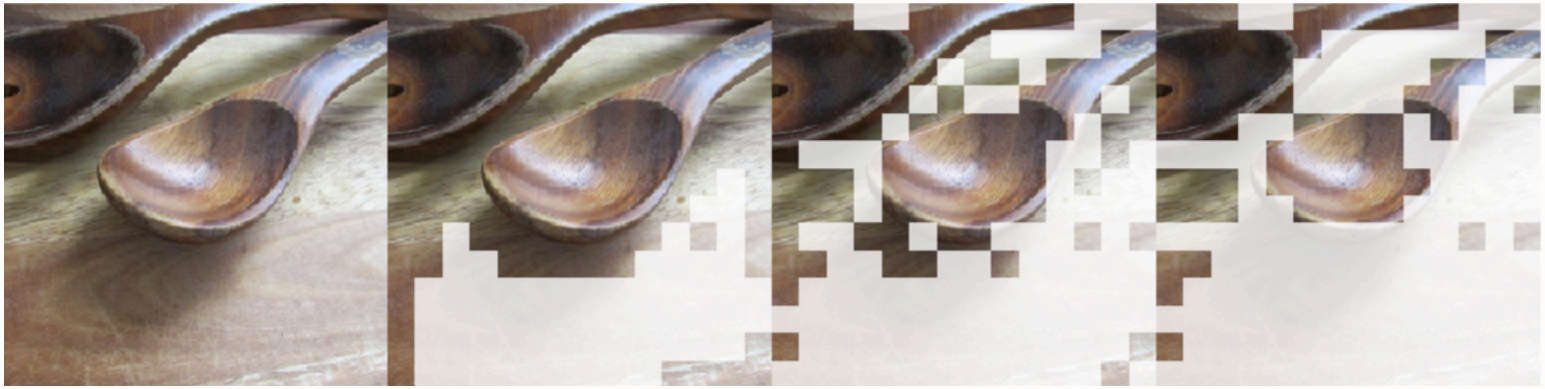


Fig. 4. Spoon [1]

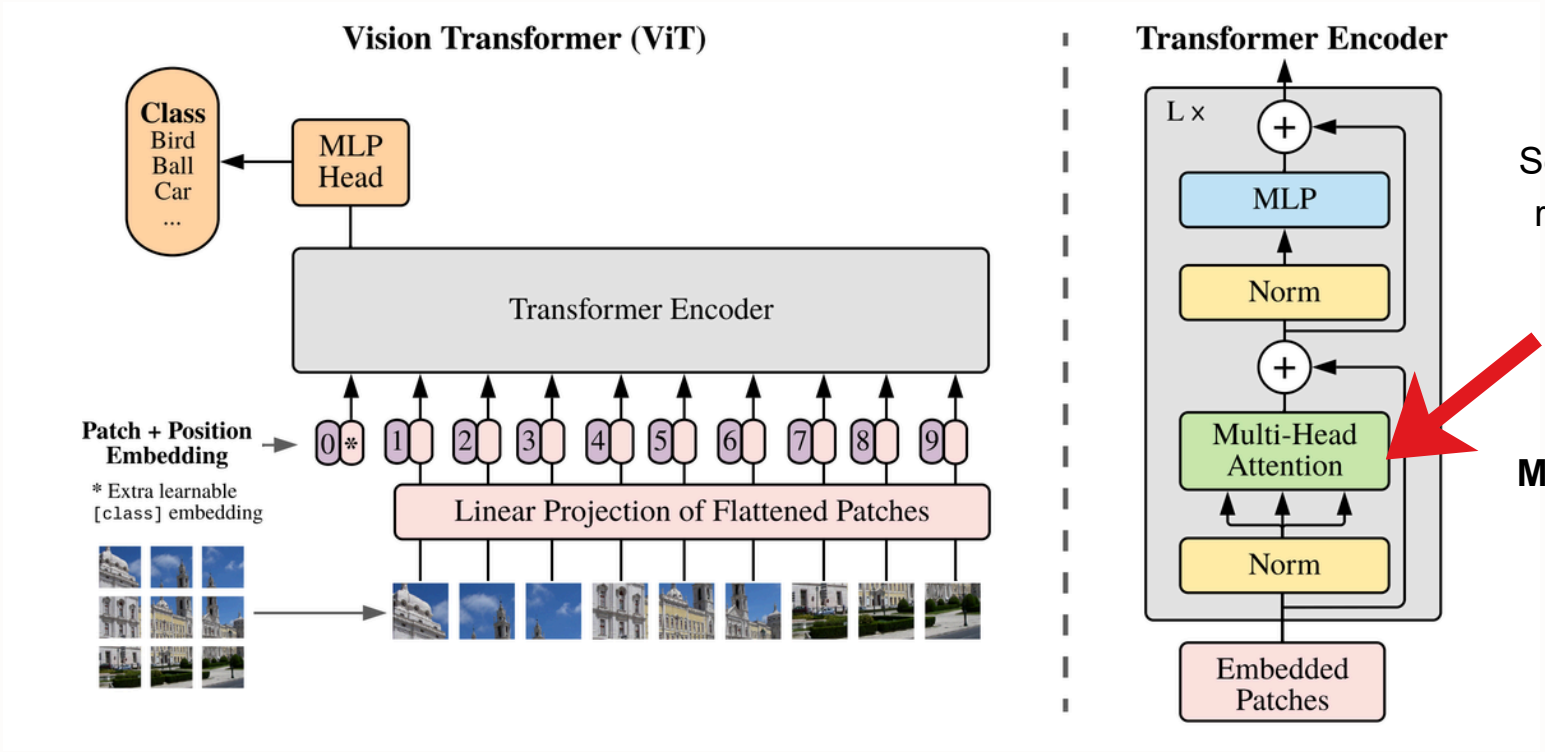


Fig. 2. VIT architecture [2]

Self-attention computes relationships between **All token pairs**.

complexity $O(n^2)$

More tokens → Higher computation.

ViT divides an image into fixed-size patches.
Each patch becomes a **visual token**

For these images, ViT will process **All patches**

Including:
Car window, Seats, Dashboard
and
Stage, dark background and empty regions.



Fig. 3. Human [3]

But

the key question is:

Are all patches really equally important for recognition?

Can we reduce redundant visual tokens
while preserving accuracy?

THE UNIVERSITY OF SYDNEY

A-ViT: Adaptive halting for token computation

It allows different tokens to stop computation at different depths

Limitation:

Do Not merge their information into retained tokens.

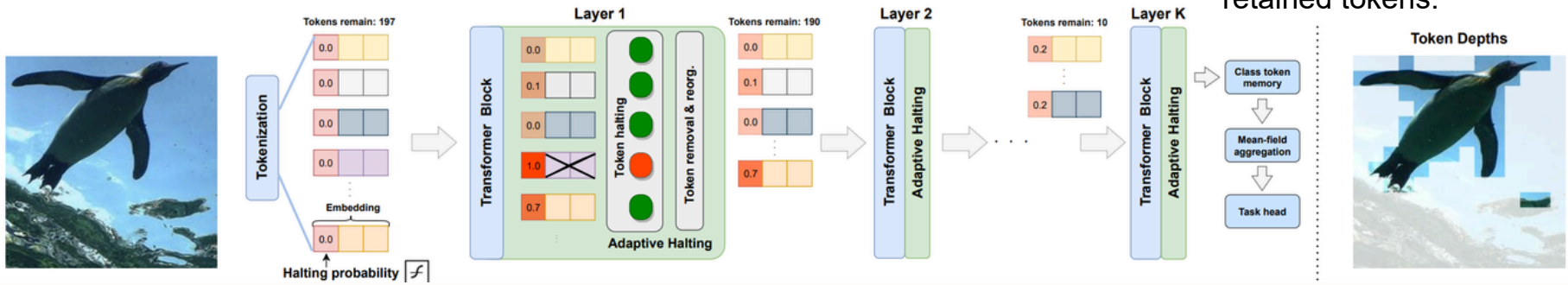


Fig. 4 A-ViT [4]

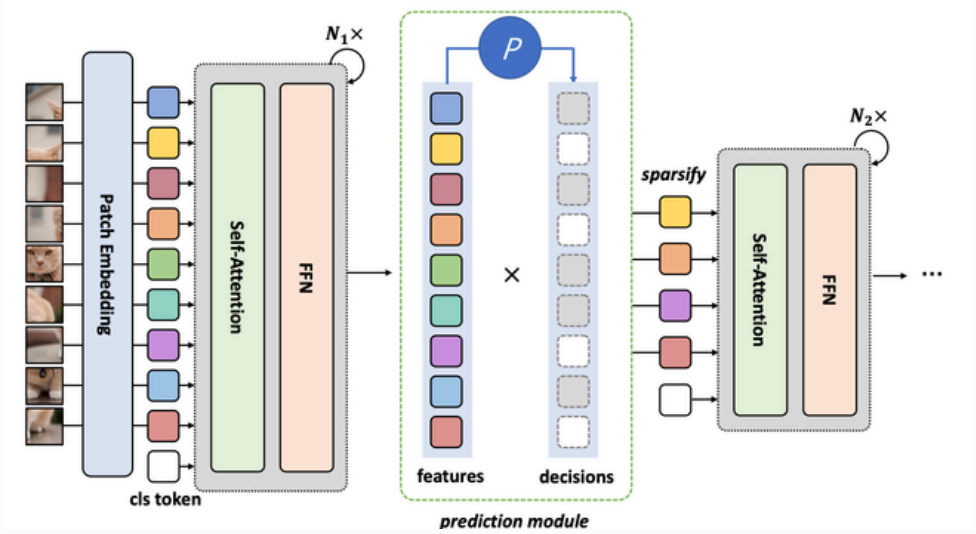


Fig. 5 DynamicViT [1]

DynamicViT: Learned token importance for pruning

Using a lightweight prediction module to estimate the importance

Limitation:

It needs an extra predictor

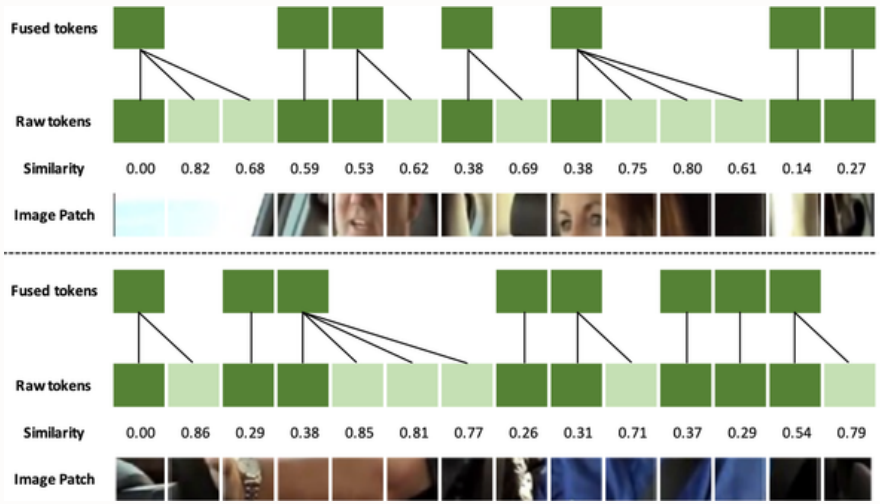


Fig. 6 DynTok [3]

DynTok：Similarity between **adjacent patches**

Limitation:

Local similarity does **not equal** classification importance.

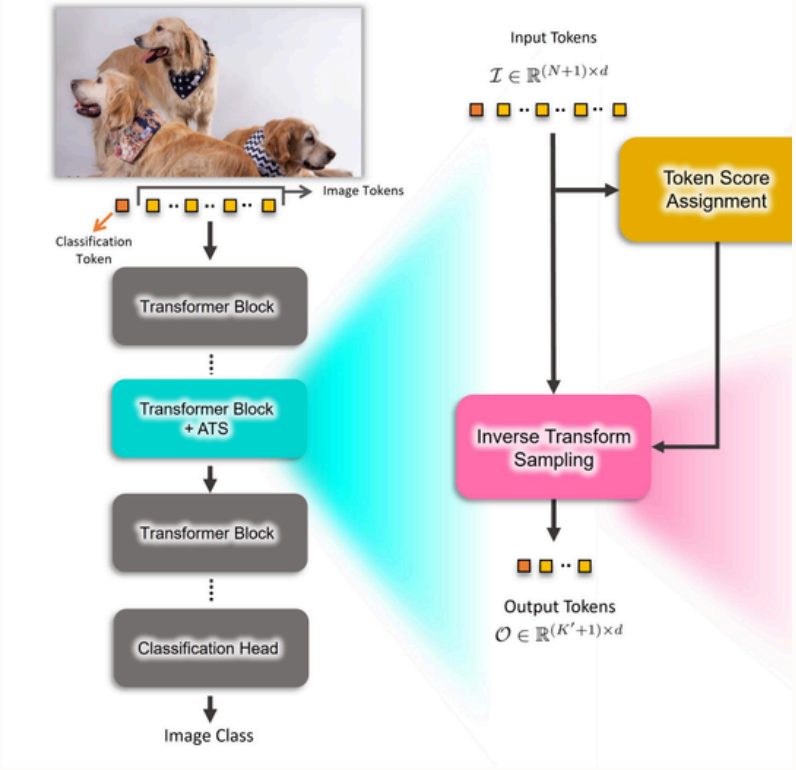


Fig. 7 ATS [5]

ATS：**Score-based** adaptive token sampling

Limitation:

Sampled-out tokens may lose useful information

Research Gap

1.Hard pruning may lose useful information

Low-attention patches are not always useless.
They may still contain context, object boundaries, or secondary cues.

2.Similarity merging is not task-aware

Similar patches may come from background or repeated textures.

Similarity alone does not indicate classification importance.

3.Adaptive sampling still selects, not preserves

Token numbers can change for each image.
But unselected tokens are not explicitly merged back.

This thesis bridges the gap

1.Class attention selects task-relevant centers.
2.Coverage ratio decides how many centers to keep
3.Feature-position clustering merges redundant tokens into centers.

Core Idea

1 Class attention

Use class-to-patch attention as a lightweight importance score.

Task-aware signal

2 Adaptive budget

Choose how many center tokens to keep based on each image's attention distribution.

Image-specific count

3 Soft merging

Assign tokens by feature and spatial similarity, then merge by importance weight.

Information-preserving

Shorter sequence for later blocks

Keep the class token; compress patch tokens progressively after intermediate blocks.

Overall Framework

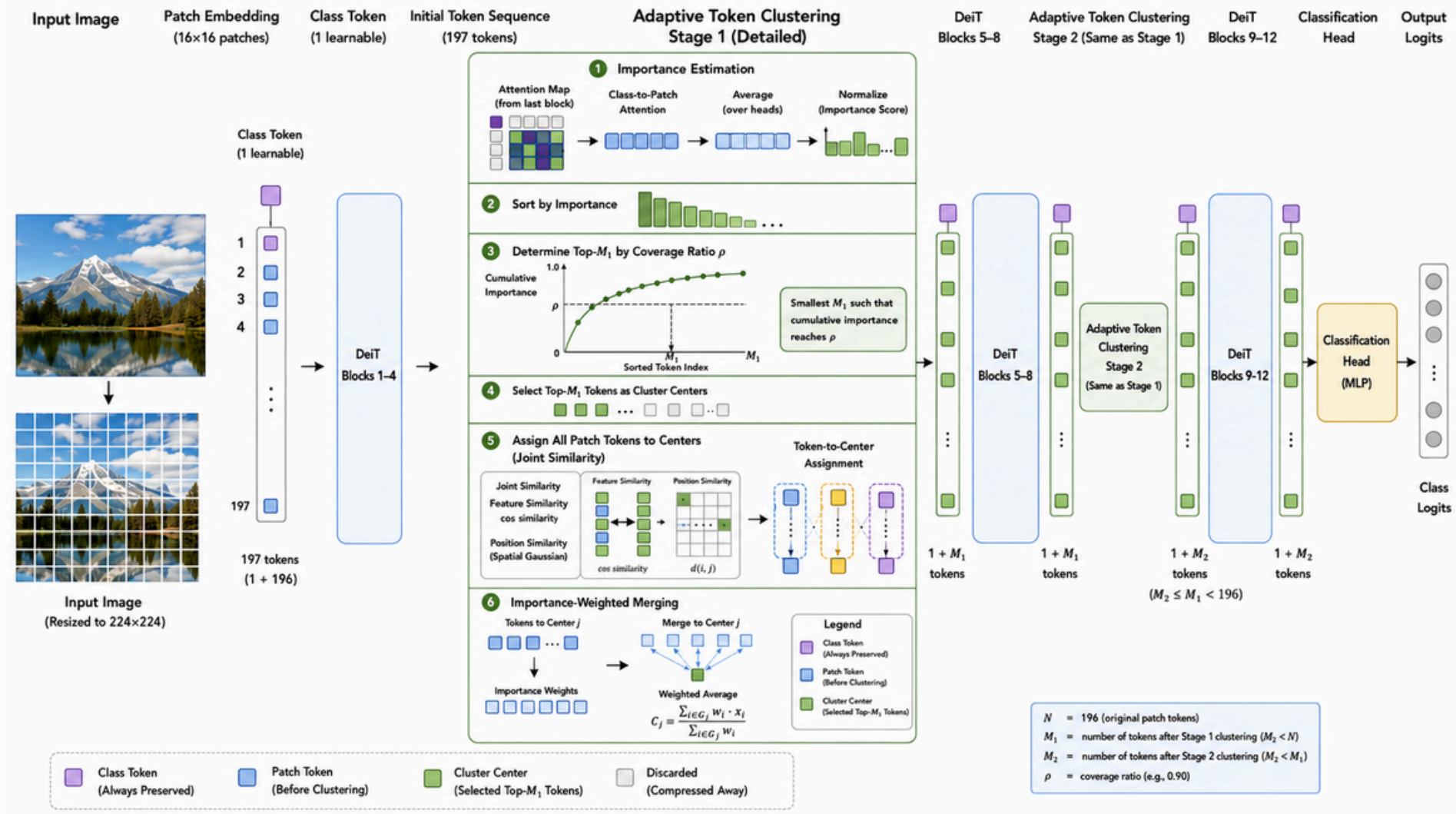
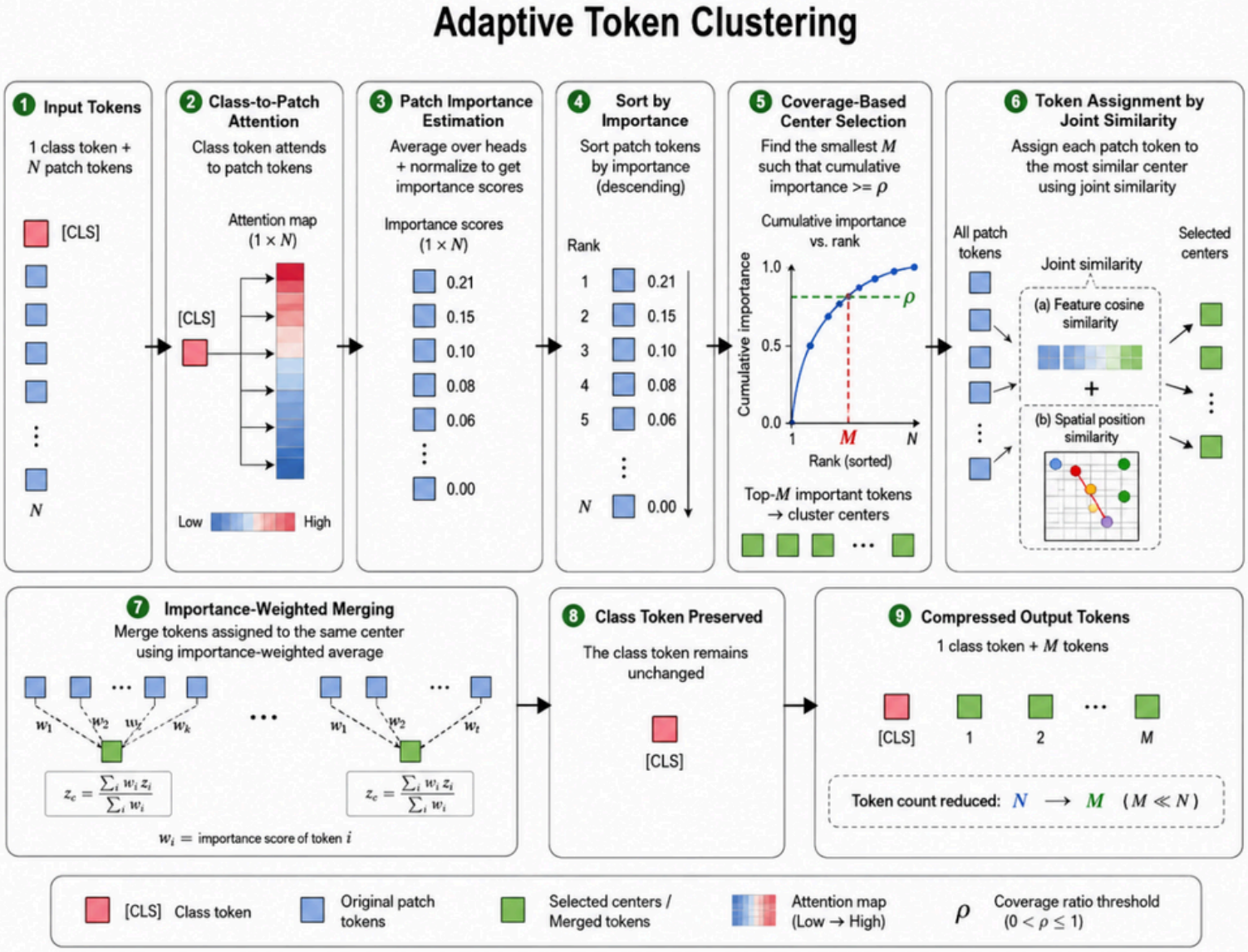


Fig. 8 Overall Framework

THE UNIVERSITY OF SYDNEY



Method Flow

1. Use class attention to score patch token.
2. Select important tokens as centers
3. Assign other tokens by feature-position similarity
4. Merge each cluster with importance weights.

Fig. 9 Core Module

Backbones

DeiT-Tiny

D=192, 12 blocks, 3 heads

DeiT-Small

D=384, 12 blocks, 6 heads

Datasets

ImageNet-100

Natural image subset, 100 classes

CIFAR-100

Compression

Coverage ratios

0.8 | 0.6 | 0.4

Conservative, balanced, aggressive

Evaluation metrics

Top-1 / Top-5
classification quality

GFLOPs
estimated compute

Latency
batch size 1

Throughput
batch size 32

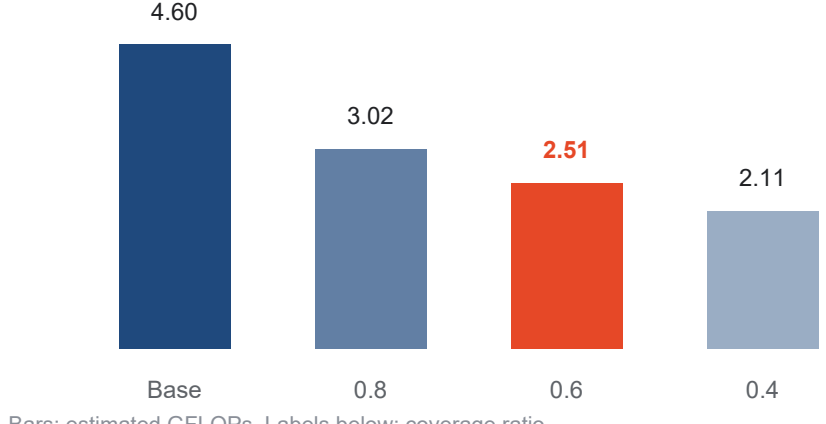
Main comparison: baseline DeiT vs proposed clustering variants

THE UNIVERSITY OF SYDNEY

Result on ImageNet 100

Model	Method	Ratio	GFLOPs	Top-1 (%)	Top-5 (%)	Latency (ms)	Throughput (img/s)
DeiT-S	Baseline	–	4.5985	88.72	98.34	6.82	1305.12
DeiT-S	Proposed	0.8	3.0161	88.12	97.84	11.21	1553.99
DeiT-S	Proposed	0.6	2.5097	87.30	96.44	10.13	1647.10
DeiT-S	Proposed	0.4	2.1063	85.88	96.00	9.95	1883.57
DeiT-T	Baseline	–	1.2535	86.18	96.66	5.44	3289.12
DeiT-T	Proposed	0.8	0.8199	86.08	96.74 ✓	15.96	1953.66
DeiT-T	Proposed	0.6	0.6835	85.56	96.68 ✓	11.53	2073.22
DeiT-T	Proposed	0.4	0.5773	83.98	95.82	9.71	2713.35

DeiT-Small: GFLOPs and Top-1 trade-off



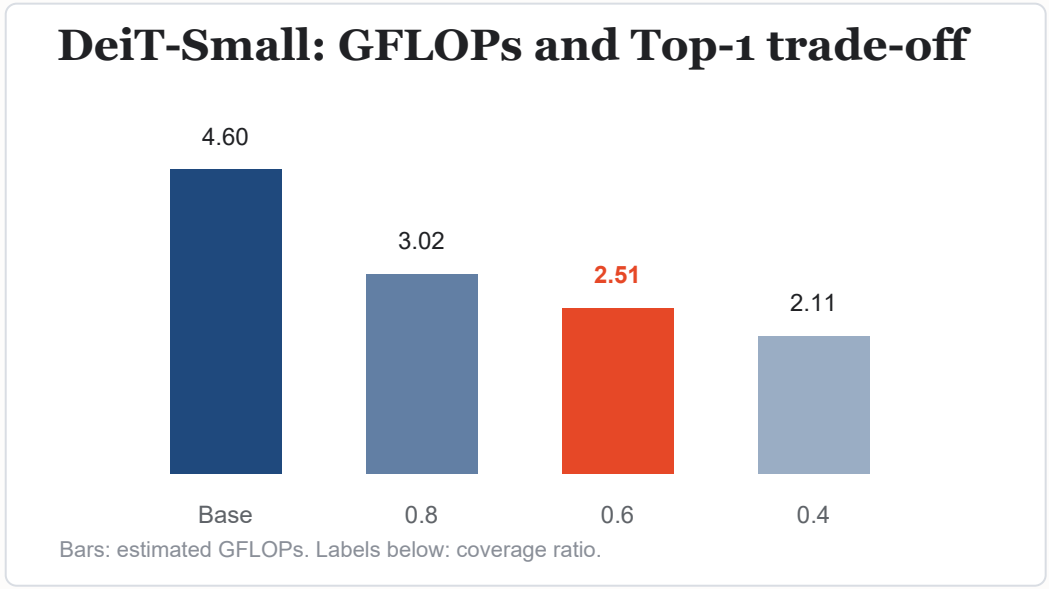
Bars: estimated GFLOPs. Labels below: coverage ratio.

Interpretation

- **Coverage ratios=0.8 is safest**
  Top-1 88.12%, but compute reduction is smaller.
- **Coverage ratios=0.6 is balanced**
  Large GFLOPs reduction with still-competitive accuracy.
- **Coverage ratios=0.4 is aggressive**
  Best compute saving but larger information loss.

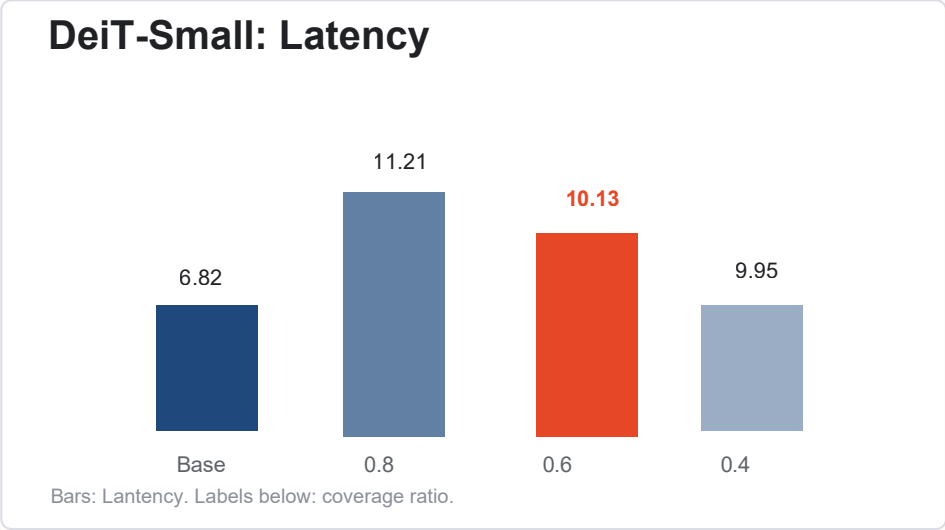
Result on CIFAR-100

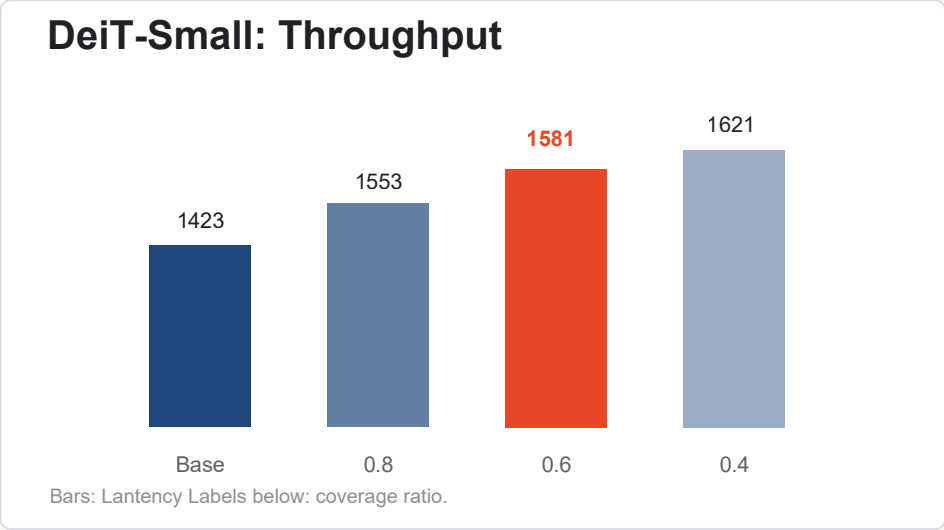
Model	Method	Ratio	GFLOPs	Top-1 (%)	Top-5 (%)	Latency (ms)	Throughput (img/s)
DeiT-S	Baseline	–	4.5985	84.63	95.71	6.00	1423.63
DeiT-S	Proposed	0.8	3.0161	84.43	95.76 ✓	15.10	1553.55
DeiT-S	Proposed	0.6	2.5097	84.42	95.75 ✓	14.13	1581.83
DeiT-S	Proposed	0.4	2.1063	83.65	95.38	12.11	1621.17
DeiT-T	Baseline	–	1.2535	83.22	95.03	5.79	3201.88
DeiT-T	Proposed	0.8	0.8199	83.16	95.28 ✓	15.81	1828.48
DeiT-T	Proposed	0.6	0.6835	82.49	94.87	10.99	2760.39
DeiT-T	Proposed	0.4	0.5773	81.88	94.46	9.81	2801.06



The trend is consistent: moderate compression preserves accuracy best.

DeiT-Small: Better Throughput, Higher Latency



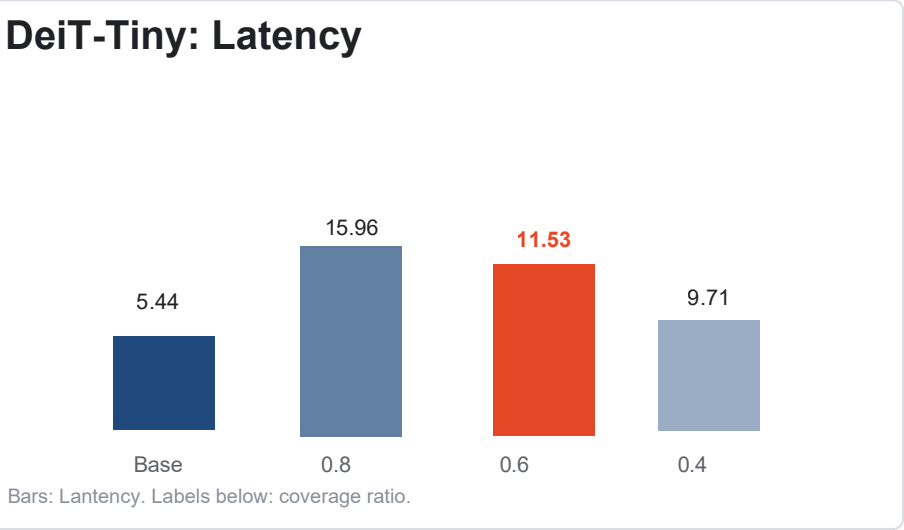


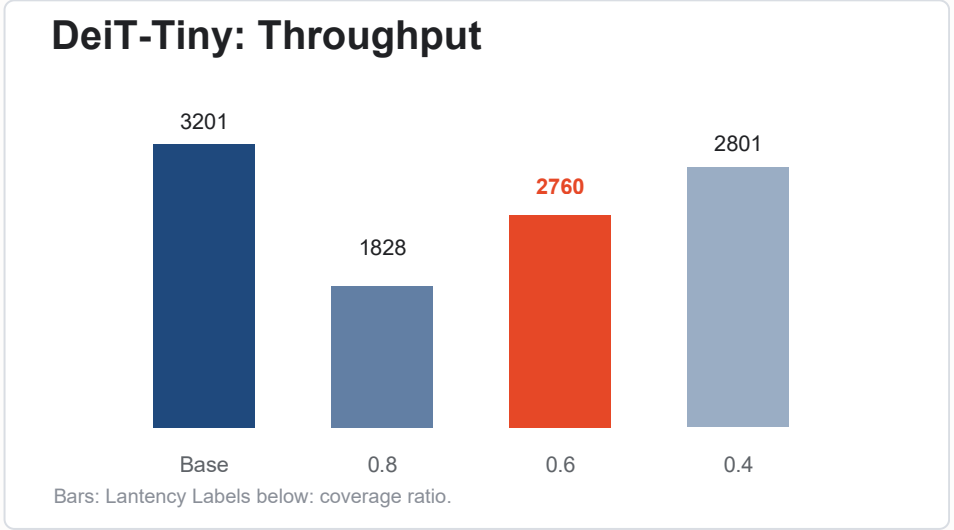
Latency is still **higher** than the baseline because
adaptive clustering introduces extra operations, such as token sorting, assignment, merging

As the coverage ratio **decreases**, throughput **improves** because
stronger token compression

For batch inference, the saved computation from shorter token sequences can compensate for the clustering overhead, so more images can be processed per second.

DeiT-Tiny: Clustering Overhead Dominates





For DeiT-Tiny, the backbone is already lightweight, so the saved computation cannot compensate for the clustering overhead.

Even with fewer tokens, the extra sorting, assignment, merging, and masking costs are too large for such a small model.

Unlike DeiT-Small, DeiT-Tiny does not benefit in real runtime. The model is already small, so reducing tokens saves only a small amount of computation. But the clustering module still adds sorting, assignment and merging costs. So the overhead is larger than the saving

THE UNIVERSITY OF SYDNEY

Visualization

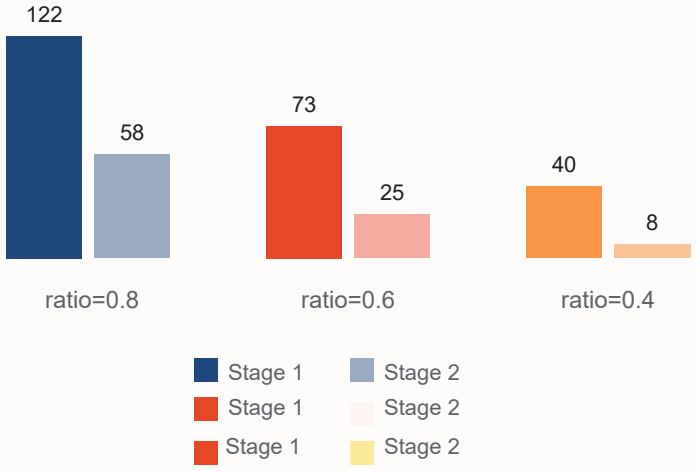


Fig. 10 Visulazation

The high-score regions are mainly around the **eyes, face, and hair.**
Background and less discriminative regions receive lower importance scores.

When the coverage ratio increases, **more surrounding context** and body regions are retained.

Average retained tokens, DeiT-S on ImageNet-100



The importance distribution is highly uneven.
Only around **40 tokens** are needed to reach 0.4,
while more than **120 tokens** are required to reach 0.8.

This suggests that a small number of tokens **contain much higher importance** scores than the others.

Limitation and Future work

1. Runtime overhead

Sorting, assignment, scatter merge, padding, and masking are less hardware-friendly than dense operations.

Next: fused kernels and regularized assignment.

2. Manual hyperparameters

Coverage ratio, stage budget affect compression behavior and transferability.

Next: adapt to image complexity or runtime target.

3. Attention-only importance

Class attention may focus on discriminative regions and underrepresent boundaries or context.

Next: combine diversity, local structure, uncertainty.

4. Limited to image classification

Dense prediction needs stronger spatial-detail preservation than image classification.

Next: object detection and segmentation constraints.

A practical efficient ViT must be evaluated with GFLOPs, latency, throughput, memory, and hardware behavior together.

Main takeaway

Class-attention-guided center aggregation can reduce redundant token computation while preserving much of the classification signal.

1 Redundancy exists

Many patch tokens can be compressed
with limited accuracy loss.

2 Coverage ratio is a useful knob

A moderate setting gives the best accuracy-efficiency balance.

3 Runtime efficiency is still challenging

Future work should make clustering cheaper and more GPU-friendly.

References

[1] Y. Rao, W. Zhao, B. Liu, J. Lu, J. Zhou, and C.-J. Hsieh, “Dynamicvit: Efficient vision transformers with dynamic token sparsification,” Advances in neural information processing systems, vol. 34, pp. 13 937–13 949, 2021.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Un-terthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly et al., “An imageis worth 16x16 words: Transformers for image recognition at scale,” arXiv preprint arXiv:2010.11929, 2020

[3] H. Zhang, J. Zhang, X. Ji, Q. Wang, and F. Zhang, “Dyntok: Dynamiccompression of visual tokens for efficient and effective video understanding,” arXiv preprint arXiv:2506.03990, 2025.

[4] Yin, Hongxu, et al. "A-vit: Adaptive tokens for efficient vision transformer." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022.

[5] YFayyaz, Mohsen, et al. "Adaptive token sampling for efficient vision transformers." European conference on computer vision. Cham: Springer Nature Switzerland, 2022.

[6] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. J´egou, “Training data-efficient image transformers & distillation through atten-tion,” in International conference on machine learning. PMLR, 2021, pp. 10 347–10 357.

END



THE UNIVERSITY OF SYDNEY

Thank you!

Feel free to ask any questions !