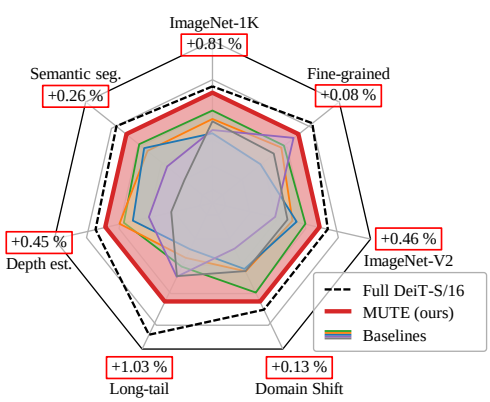
MARGINAL UTILITY TOKEN ELIMINATION FOR VISION TRANSFORMERS

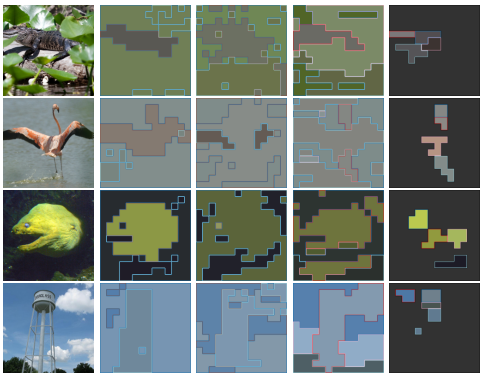
Anonymous authors
Paper under double-blind review

ABSTRACT

Existing token reduction methods leave their framework choice to ad-hoc heuristics over the backbone's inductive bias or to task-specific learning of an auxiliary reducer module. We instead organize token reduction into two axes, a partition that groups tokens into clusters and an aggregation that produces a representative for each cluster, and state a single explicit objective over both axes: a sensitivity-weighted clustering distortion in the key-projected space, which a first-order Taylor expansion of the next block motivates and three stated approximations make computable. The proposed MUTE minimizes this objective by alternating its two analytic coordinate updates, a weighted-mean aggregation and a nearest-representative partition, which descend the objective monotonically and recover the size bias of bipartite merging as a derived correction rather than as a heuristic. Prior token pruning, merging, and clustering reducers occupy restricted cells of the same design grid, each replacing at least one of the two coordinate updates with a mechanism chosen outside the objective. Under the unified protocol MUTE attains the first-rank training-free classification accuracy, and the same reducer transfers without modification to backbones spanning supervised and self-supervised pretraining and to tasks spanning dense prediction, fine-grained classification, and zero-shot domain shift. A systematic grid search shows that design choices inside the framework make up an equivalent family within multi-seed noise while design choices outside it incur substantial losses, locating the sensitivity of token reduction on the partition axis as the structure of the objective anticipates.



(a) Accuracy retained across tasks.



(b) Token partition visualization.

Figure 1: (a) Scores relative to MUTE, the regular heptagon, with radial gaps magnified for visibility; boxes are MUTE's margin in %p over the best baseline. (b) Partitions are coherent, not semantic.

1 INTRODUCTION

Transformer has become the standard architecture across modalities (Vaswani et al., 2017; Gong et al., 2021; Guo et al., 2021), and vision has followed the same trajectory through the spread of the Vision Transformer (Dosovitskiy et al., 2021) as well. This spread comes at the cost of self-attention's quadratic dependence on the number of tokens, which simultaneously presses the latency

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

budget of mobile inference (Mehta & Rastegari, 2022; Li et al., 2022) and the throughput cost of online services. Among the axes that compress this cost, distillation (Yu et al., 2026) and quantization (Yuan et al., 2022) act on the backbone's weights and on the numerical precision of its activations, whereas token reduction acts directly on the cause of the cost by reducing the number of inputs to self-attention itself.

Token reduction has branched into two streams. The training-free reducer uses the K-vector or attention distribution of a pretrained backbone as a heuristic signal and spans the sub-families of merging, hybrid prune-and-aggregate, and clustering (Bolya et al., 2023; Wu et al., 2023; Haurum et al., 2024); the learned reducer additionally trains the reducer component with task labels (Rao et al., 2021; Yin et al., 2022; Zong et al., 2022). The learned stream pays for its capacity with the training cost of the reducer component, additional parameters and memory, the train-test distribution mismatch, and the deployment rigidity of backbone and keep ratio decisions hard-coded at training time, whereas the training-free stream avoids every one of these costs, plugs into a freshly released backbone immediately, and adjusts the keep ratio at deployment time. Yet within the training-free stream the reducer is specified as a mechanism rather than as the solution of a stated objective, so its behavior remains an ad-hoc heuristic tied to the pretrained backbone's inductive bias and its framework choice stays opaque to architectural or task-level analysis.

We organize token reduction into the two axes of partition and aggregation and minimize a single explicit objective over both. A first-order Taylor expansion of the next block motivates penalizing each token's deviation from its representative in a metric induced by the block's local Jacobian; replacing that metric with the block's key projection and weighting each deviation by the token's attention mass at that block yields a sensitivity-weighted clustering distortion whose two coordinate updates are analytic. The M-step returns the sensitivity-weighted mean of each cluster under a fixed partition and the E-step assigns each token to its nearest representative under fixed representatives, so alternating them descends the objective monotonically. We call the per-token quantity this objective scores the token's *marginal utility*: the distortion avoided by keeping a token as its own representative instead of folding it into a cluster; the objective scores every token by this quantity, and the weighted mean absorbs the tokens it folds. The resulting reducer keeps every advantage of the training-free stream while replacing its ad-hoc mechanism with an explicit objective, attains the highest accuracy on the matched-protocol classification benchmark, and transfers without modification to backbones spanning supervised and self-supervised pretraining and to tasks spanning dense prediction, fine-grained classification, and zero-shot domain shift.

In summary, our contributions are:

- We pose token reduction as the minimization of a single sensitivity-weighted clustering distortion over the partition and aggregation axes, state the three approximations that make it computable, and give both coordinate updates analytically, replacing the mechanism choice of prior reducers.
- The resulting reducer attains the highest training-free ImageNet-1K accuracy while preserving the deployment advantages of the training-free stream, and transfers without modification across backbones spanning supervised and self-supervised pretraining and across diverse vision tasks.
- A systematic grid search shows that design choices make up an equivalent family inside the framework and degrade sharply outside it, locating the sensitivity of token reduction on the partition axis.

2 Related Works

Token reducers span four mechanism families, which Appendix A reviews method by method. **Token pruning** drops tokens by a per-token score, learned or read off attention (Rao et al., 2021; Yin et al., 2022; Liang et al., 2022; Fayyaz et al., 2022; Choi et al., 2025). **Token merging** fuses tokens into shared representatives instead: by one bipartite matching between alternating sets with a size attribute propagated on subsequent attention (Bolya et al., 2023), by an attention-driven many-to-many soft assignment (Zeng et al., 2025), by trainable merging vectors or a learned slimming projector (Renggli et al., 2022; Zong et al., 2022), or by multi-criteria bidirectional matching under a consistency loss (Lee et al., 2024). **Hybrid prune-and-aggregate** switches between the two on attention variance or an ATS-style score (Wu et al., 2023; Wei et al., 2023). **Explicit clustering** treats

reduction as a partition problem, by agglomerative linkage on K-vectors, density peaks, medoids, entropy-regularized transport, or weighted K-means (Haurum et al., 2024; Zeng et al., 2024; Park & Jun, 2009; Sinkhorn & Knopp, 1967; Marin et al., 2023).

Positioning. Despite this mechanism diversity, almost all of them specify the reduction logic as a mechanism rather than as the solution of a stated objective, which leaves the framework choice opaque to architectural or task-level analysis. The exception is Token Pooling (Marin et al., 2023), which does state an objective and solves it by clustering, but the objective is the reconstruction error of the token set itself in feature space, a quantity defined without reference to the block that consumes those tokens; our objective is instead the deviation the next block sees, under a metric that block induces, and it is this choice that makes the aggregation weight and the size attribute follow from the objective rather than being set by hand. Token Transforming (Zeng et al., 2025) is the closest prior attempt at unification and the appropriate point of contrast: it subsumes pruning and merging as instances of a single many-to-many transformation matrix, but that matrix is *read off* attention weights rather than obtained as the solution of an objective, so the unification is representational rather than variational. We instead state the objective explicitly in Section 3 and obtain both axes as its analytic coordinate updates, which additionally makes the size attribute of merging a consequence rather than an assumption.

3 METHOD

3.1 TOKEN REDUCTION AS PARTITION AND AGGREGATION

We propose MUTE, a token reducer defined as the coordinate descent of an explicit distortion objective. We organize token reduction as a two-step process: a partition $\pi : \{x_i\}_{i=1}^{N} \to \{C_k\}_{k=1}^{K}$ that groups input tokens at reduction layer l into K clusters, and an aggregation $\mathcal{A} : C_k \to a_k \in \mathbb{R}^D$ that produces a representative for each cluster. The CLS token is always preserved, and the downstream forward of the subsequent transformer block f^{l+1} runs on $A = \{a_k\}_{k=1}^{K}$ instead of $X = \{x_i\}_{i=1}^{N}$. The quality of a reduction is ultimately the discrepancy between the downstream outputs of the two token sets, a set-level functional between collections of unequal cardinality that admits no tractable expression. Section 3.2 therefore replaces it with a per-token surrogate under three explicitly stated approximations, and every subsequent claim is a claim about that surrogate.

3.2 THE WEIGHTED DISTORTION OBJECTIVE

The next block's output near a representative a_k admits a first-order Taylor expansion,

$$f^{l+1}(x_i) = f^{l+1}(a_k) + J(a_k)(x_i - a_k) + O(\|x_i - a_k\|^2), \tag{1}$$

where $J(a_k) = \partial f^{l+1}/\partial x|_{a_k} \in \mathbb{R}^{D\times D}$ is the local Jacobian, so the leading term of the per-token discrepancy is $\|J(a_k)(x_i - a_k)\|^2 = \|x_i - a_k\|^2_{M_k}$ with $M_k = J(a_k)^\top J(a_k) \succeq 0$ and $\|y\|^2_M := y^\top My$. Equation 1 *motivates* rather than determines the objective we minimize: the objective follows from three approximations, which we state here so that the propositions of Section 3.3 are read as statements about the surrogate and not about the true next-block output. **(A1) Set to per-token.** We replace the set-level discrepancy by a sum of per-token deviations, which treats the block as acting token-wise at the linearization point and ignores the change of attention normalization caused by the reduced token count; that residual is handled by the $\log s_k$ size bias of Section 3.4, and the Taylor cross term is absorbed, an orthogonality assumption rather than a consequence (Appendix F). **(A2) Metric surrogate.** M_k depends on the unknown a_k and costs a Jacobian evaluation per cluster to build, so we replace it with one fixed positive semi-definite metric $M = W_K^\top W_K$ built from the block's key projection, common to all clusters and independent of a_k. This is an *operational* surrogate, adopted for its cost and its measured agreement with the J-induced assignment, and not a subspace identity: Appendix D reports that the strict alignment between M and the leading singular subspace of $J^\top J$ is at chance level while the induced assignments and the downstream top-1 agree closely. **(A3) Sensitivity weighting.** The first-order sensitivity of the block output to a displacement of token i scales with the squared attention mass $\omega_i = \sum_j (P_{ji})^2$ that the block places on the token (Appendix C), so token i enters the sum with weight $w_i = \omega_i$ rather than with weight one; ω_i is measured once per reduction stage and held fixed during the descent.

Definition 1 (Weighted first-order distortion)**.** *Under (A1)–(A3), we define the distortion of a token reduction as*

$$\mathcal{L}(C, A) := \sum_{k=1}^{K} \sum_{i \in C_k} w_i \left\| W_K(x_i - a_k) \right\|_2^2 = \sum_{k=1}^{K} \sum_{i \in C_k} w_i \left\| x_i - a_k \right\|_M^2, \qquad w_i = \omega_i > 0, \quad (2)$$

that is, the weighted K-means distortion in the key-projected space. $\mathcal{L}$ is the quantity MUTE minimizes and measures in Section 4.5.1; it is not a bound on the next-block output error, and Appendix F quantifies the gap, the higher-order remainder growing roughly thirtyfold from the first to the third reduction stage.

3.3 Analytic Coordinate Updates

$\mathcal{L}$ is jointly non-convex in $(\{C_k\}, \{a_k\})$, but because M and $\{w_i\}$ are held fixed, each coordinate block admits an analytic minimizer. We call the alternation EM-style for its structure only: it is the hard-assignment coordinate descent of Lloyd's algorithm on a weighted objective, not expectation–maximization on a latent-variable likelihood.

Proposition 1 (M-step: weighted mean)**.** *For a fixed partition $C = \{C_k\}$, every minimizer of $\mathcal{L}(C, \cdot)$ over the k-th representative satisfies*

$$a_k^\star = \frac{\sum_{i \in C_k} w_i \, x_i}{\sum_{i \in C_k} w_i}, \quad (3)$$

and $a_k^\star$ is the unique minimizer whenever $M \succ 0$; when M is singular the minimizer is unique modulo $\ker M$ and Equation 3 is the canonical representative.

Since M does not depend on a_k, the partial objective $\mathcal{L}_k(a_k) = \sum_{i \in C_k} w_i (x_i - a_k)^\top M (x_i - a_k)$ is a quadratic whose stationarity condition $\sum_{i \in C_k} w_i M(x_i - a_k) = 0$ is solved by Equation 3 for *any* choice of M (Appendix G), so the update requires no matrix inverse and no evaluation of J, costing one weighted mean per cluster. The sensitivity-weighted mean is therefore the minimizer of Equation 2 under (A3); the size-weighted mean that bipartite merging introduces as a heuristic (Bolya et al., 2023) corresponds to it under the constant-attention approximation $\omega_i \approx s_i^2 c$, and in a matched swap the two aggregations land within 0.02%p of each other (Appendix C), so within our formulation the size attribute approximates a derived weighting rather than being an assumption.

Proposition 2 (E-step: nearest representative)**.** *For a fixed aggregation $A = \{a_k\}$, the minimizer of $\mathcal{L}(\cdot, A)$ assigns each token to its nearest representative in the M-induced metric,*

$$c^\star(i) = \arg\min_k \left\| x_i - a_k \right\|_M^2 = \arg\min_k \left\| W_K(x_i - a_k) \right\|_2^2. \quad (4)$$

The weights leave the assignment unchanged, since $w_i > 0$ is constant in k and therefore $\arg\min_k w_i \|x_i - a_k\|_M^2 = \arg\min_k \|x_i - a_k\|_M^2$.

This follows from the additivity of $\mathcal{L}$ over i; since both updates are the minimizers of their coordinate block, alternating them makes $\mathcal{L}$ non-increasing and terminates after finitely many steps by the standard monotonicity of weighted K-means under a fixed metric (Appendix U). The two axes enter $\mathcal{L}$ asymmetrically, and that asymmetry is the structural expectation tested in Section 4.5. Proposition 3 in Appendix H makes this precise: at a fixed partition the aggregation gap is an identity bounded by how tightly a centroid estimate concentrates *inside* each cluster, whereas at optimal aggregation the partition gap is an identity equal to the between-cluster energy that the partition fails to capture, so the partition is the load-bearing axis. The ratio of the two is bounded only after restricting to $\mathrm{range}(M)$ and carries a metric-conditioning constant we do not control, and it bounds a ratio of $\mathcal{L}$ gaps rather than of top-1 differences; we therefore use the asymmetry qualitatively and report its magnitude in Section 4.5 as an empirical quantity rather than a predicted factor.

3.4 Implementation of MUTE

Equation 2 and its two updates fix MUTE's structure but leave five choices open, which we resolve as follows and characterize empirically in Section 4.5. The **E-step metric** is $M = W_K^{l\top} W_K^l$, so that Equation 4 is the Euclidean nearest-representative rule in the key-projected space $W_K^l x_i$; this is the surrogate (A2) itself, adopted for its cost and its measured agreement with the J-induced assignment (Appendix D), and not an instantiation of M_k itself. The **M-step aggregation** is the ω-weighted mean of Equation 3, carried with the $\log s_k$ bias on subsequent attention that accounts for a representative's accumulated mass (Bolya et al., 2023). The **initialization** is ω-weighted K-means++ farthest-first seeding (Arthur & Vassilvitskii, 2007) with the same ω_i that weights the objective, so a

single sensitivity score plays both roles; the seeding choice affects which local minimum the descent reaches (Appendix C). The **iteration count** is $t \in \{1, 2\}$ per reduction stage, truncated for compute parity with single-pass merging baselines rather than run to convergence. The **K-vector layer** is the current block's W_K^l rather than the next block's W_K^{l+1} that (A2) nominally targets, which avoids an additional forward pass (Appendix E). We refer to the instantiation aggregating these choices as MUTE in the tables of Section 4.

3.5 Relation to Prior Reducers

Prior reducers occupy restricted cells of the same partition and aggregation grid. Pruning collapses the partition to singleton self-clusters retained by a top-K score, with the aggregation reduced to the identity. Bipartite merging (Bolya et al., 2023) applies pairwise greedy matching on the partition axis and the size-weighted mean on the aggregation axis; that mean is the constant-attention approximation of MUTE's M-step (Appendix C), while its matching step optimizes no stated objective. Many-to-many merging (Zeng et al., 2025) relaxes the partition to attention-weighted soft assignment. Multi-criteria merging (Lee et al., 2024) replaces the metric of Equation 4 with a heuristic combination of similarity, attention, and size scores. Hybrid switching (Wu et al., 2023; Wei et al., 2023) branches between pruning and merging by per-token variance or score thresholds. Explicit clustering (Haurum et al., 2024; Zeng et al., 2024; Park & Jun, 2009; Sinkhorn & Knopp, 1967) replaces the same metric with linkage, density peak, medoid, or entropy-regularized transport, and typically aggregates by an unweighted mean. MUTE is the cell whose two axes are the coordinate minimizers of Equation 2, and each reducer above departs from one or both of them. This is a statement about which objective a cell optimizes, not a claim that the cell optimizing $\mathcal{L}$ must be the most accurate; Section 4 tests that separately.

4 Experiments

4.1 Setup

All experiments use a DeiT-S/16 (Touvron et al., 2021) backbone pretrained on ImageNet-1k (Russakovsky et al., 2015) unless otherwise stated, with top-1 reported on the full ImageNet-1k validation split. Training-free baselines comprise ToMe (Bolya et al., 2023), ATC (Haurum et al., 2024), PPT (Wu et al., 2023), Token Transforming (Zeng et al., 2025), MCTF (Lee et al., 2024) in its training-free mode, Representation Shift (Choi et al., 2025), Top-K, and three clustering baselines DPC-kNN (Zeng et al., 2024), K-Medoids (Park & Jun, 2009), and Sinkhorn (Sinkhorn & Knopp, 1967). Table 1 reports the eight training-free baselines together with MUTE under the matched three-location protocol; Top-K and Sinkhorn appear in the fine-tuned comparison of Appendix K with the full eleven-reducer set. Learned baselines require fine-tuning and comprise DynamicViT (Rao et al., 2021), SiT (Zong et al., 2022), A-ViT (Yin et al., 2022), TPS in its learned variant dTPS (Wei et al., 2023), and MCTF in its full fine-tuned mode. All reducers share the same backbone, reduction protocol, and fine-tuning schedule when fine-tuning applies; the full protocol is in Appendix B. Throughout this section "MUTE" is the reference implementation of Section 3.4 with the Mahalanobis E-step unless a table states otherwise, as the head-only multi-task table of Section 4.4 and the cosine row of Table 6 do; within-framework alternatives are characterized in Section 4.5 and Appendix D.

4.2 Main Result

MUTE attains the highest training-free top-1 in Table 1 at 78.57%, exceeding ToMe by 1.76%p, ATC by 0.81%p, and PPT by 1.35%p at the same pretrained backbone, the same reduction location, and the same keep ratio, so the gap reflects the choice of reducer alone. MUTE also attains the highest fine-tuned top-1 at 79.24%, above the strongest learned reducer DynamicViT at 76.48% and above A-ViT at 70.68%, whose halting-head schedule transfers poorly to the matched three-location protocol used here; the learned rows are reported for completeness, lie outside the training-free axis on which our claim is made, and add 0.24M to 0.72M predictor or slimming parameters where training-free reducers add none. Two rows need a protocol note. The MCTF rows use the audited configuration that recovers the paper-effective 76.73% from the codebase defaults of 68.74%, with the same audited temperatures and the official consistency loss in the fine-tuned column, superseding an earlier 73.93% taken at default temperatures (Appendix I). The Representation Shift row at 38.13% reflects the unified protocol while the paper-faithful schedule recovers 70.90% (Appendix J); we report both because the unified protocol, though necessary for a matched comparison,

Method	TF (%)	FT (%)	FLOPs (G)	Params (M)	Throughput (img/s)
Full DeiT-S/16	79.82		4.61	22.05	3,965
Training-free					
ToMe	76.81	78.36	2.29	22.05	7,367
PPT	77.22	78.32	2.29	22.05	5,926
MCTF	76.73	77.50	2.29	22.05	7,201
ATC	<u>77.76</u>	<u>78.93</u>	2.13	22.05	2,712
Token Transforming	74.37	77.91	2.30	22.05	7,299
Representation Shift	38.13	74.69	2.49	22.05	7,077
DPC-kNN	68.95	76.94	2.15	22.05	4,973
K-Medoids	70.36	76.15	2.15	22.05	959
MUTE (ours)	**78.57**	**79.24**	2.34	22.05	6,027
Learned reducers (require fine-tuning)					
DynamicViT		76.48	2.23	22.77	7,759
SiT		75.02	2.19	22.31	7,942
A-ViT		70.68	2.63	22.05	3,623
dTPS		70.01	2.52	22.29	5,988

Table 1: Top-1 on ImageNet-1k, DeiT-S/16, $\alpha = 0.5$. Bold/underline: best/2nd-best TF top-1.

is not neutral across reducers, penalizing methods whose per-stage budget assumptions differ from ours, and the paper-faithful figure is the operative claim of the original work.

The last three columns of Table 1 rank reducers differently under FLOPs and under throughput; MUTE is Pareto-dominated by no baseline while also being the cheapest under neither, and Appendix O scopes its operating regime to moderate-to-aggressive compression on that basis. Under the conservative Schedule-B regime MUTE ranks first as well, at 79.55%, so the partition structure it selects is not an artifact of one optimization regime; Appendix K reports both schedules, the 5 to 8%p catch-up that the aggressive Schedule-A of Table 1 grants the weaker clustering reducers, and why the Schedule-A entry there reads 79.28% rather than 79.24%.

4.3 Multi-Backbone Generalization

Table 2 (a) reports zero-shot top-1 on five supervised backbones at the reduction protocol of §4.2, testing whether the framework's backbone-agnostic structure transfers across ViT families and model scales without modifying the reducer. Across the DeiT (Touvron et al., 2021) and ViT (Dosovitskiy et al., 2021) families at multiple capacities MUTE's reference implementation attains the highest top-1 on every supervised backbone, with margins of 0.49 to 3.30%p over the second-best method per backbone and a cross-backbone mean of 76.10% versus the next-best ATC at 74.36%, under one unchanged E-step metric and M-step aggregation.

Method	DeiT-Tiny	DeiT-Small	DeiT-Base	ViT-S/16	ViT-B/16	Mean
Full (no reduction)	72.20	79.82	81.80	81.40	84.50	79.94
ToMe	65.33	76.81	67.95	73.00	73.36	71.29
PPT	66.20	77.22	68.57	73.15	75.51	72.13
ATC	<u>68.61</u>	<u>77.76</u>	73.35	<u>74.00</u>	<u>78.07</u>	<u>74.36</u>
Token Transforming	55.47	74.37	<u>74.12</u>	67.51	75.33	69.36
MUTE (ours)	**69.10**	**78.57**	**75.43**	**77.30**	**80.12**	**76.10**

(a) Supervised backbones.

Method	MAE-B	MAE-L	MAE-H	MoCo-v3	DINO B/8	BEiT-B	BEiT-L	Mean
Full (no reduction)	83.60	85.96	86.88	76.55	80.06	85.27	87.44	83.68
ToMe	63.53	76.81	81.23	65.14	<u>78.55</u>	73.86	79.80	74.13
ATC	**73.20**	81.16	84.06	<u>70.10</u>	78.40	<u>78.20</u>	<u>81.72</u>	<u>78.12</u>
Token Transforming	72.18	**82.44**	<u>84.56</u>	66.94	74.70	77.76	81.12	77.10
MUTE (ours)	<u>72.39</u>	<u>82.15</u>	**85.11**	**70.99**	**78.87**	**79.66**	**82.89**	**78.87**

(b) Self-supervised backbones.

Table 2: Zero-shot top-1 (%) at $\alpha = 0.5$. Bold/underline: best/2nd-best per column.

On self-supervised backbones (Table 2 (b)) MUTE attains the top rank on five of seven backbones spanning four pretraining paradigms (MAE (He et al., 2022), MoCo-v3 (Chen et al., 2021), DINO (Caron et al., 2021), BEiT (Bao et al., 2022)), with the highest cross-backbone mean at 78.87% versus the next-best ATC at 78.12%. ATC surpasses MUTE on MAE-B by 0.81%p and Token Transforming on MAE-L by 0.29%p; on the other five backbones MUTE leads by 0.32 to

1.46%p, the narrowest being DINO B/8 at 0.32%p, which we do not read as a separated result. Both deficits are specific to the MAE feature geometry and are within-family variant dependent rather than partition-quality deficits: the cosine variant of the same family attains 73.23% on MAE-B, level with ATC to within seed noise (Appendix L), while the surrogate distortion of §4.5.1 is lower for MUTE than for the second-best method on both backbones at stage 3, so we report the MAE cells as variant-sensitive rather than as evidence either way. MUTE's advantage over ToMe also grows as the token budget tightens on every backbone, from parity at light reduction to +1.74%p at the moderate keep ratio on DeiT-S/16 and to +6.0%p and +24.0%p on DeiT-III-S/16 (Touvron et al., 2022b) and DINOv2-S/14 (Oquab et al., 2024). That advantage is a property of the matched-location protocol at tight budgets rather than a uniform dominance, since a six-location ToMe workaround rises above MUTE's three-location curve by up to 1.2%p in the mid-budget range; because our protocol fixes one keep ratio and one location set it is a specific rather than a neutral operating point, and the budget study of Appendix P is what probes the sensitivity to that choice.

4.4 Multi-Task Generalization

We evaluate MUTE on iNat 2019 (Van Horn et al., 2018) fine-grained classification, ADE-20K (Zhou et al., 2017) semantic segmentation, NYU-Depth-v2 (Ignatov et al., 2024) monocular depth estimation, four fine-grained datasets (FG) (Maji et al., 2013; Nilsback & Zisserman, 2006; Bossard et al., 2014; Parkhi et al., 2012), and zero-shot domain shift (DS) (Hendrycks et al., 2021; Recht et al., 2019), under a protocol that isolates the reducer's transferability: the ImageNet-1k reducer and backbone are frozen and only the task head is trained. Two asymmetries favor the learned baselines rather than us, and Appendix N states them with the remaining protocol detail: DynamicViT and SiT use their ImageNet fine-tuned backbone and router, which co-adapts the backbone to the reducer, and on the domain-shift column they use keep-ratio-matched checkpoints where the training-free reducers have none. The MUTE row of Table 4 is the cosine within-family variant, which §4.5.2 places in the same band as the Mahalanobis reference under the classification protocol; on dense prediction the two variants separate and Mahalanobis is the stronger one, so this row is not the better of the two on the segmentation and depth columns (Appendices D, M and Q).

	$\alpha = 0.5$					$\alpha = 0.7$				
Method	iNat (↑)	ADE (↑)	NYUd (↓)	FG (↑)	DS (↑)	iNat (↑)	ADE (↑)	NYUd (↓)	FG (↑)	DS (↑)
Full DeiT-S/16	44.71	31.95	0.6622	67.41	55.17	44.71	31.95	0.6622	67.41	55.17
ToMe	38.97	29.53	**0.6952**	64.11	52.85	43.17	31.10	0.6729	66.39	54.54
ATC	40.12	<u>29.98</u>	0.6987	64.39	<u>54.15</u>	42.96	<u>31.33</u>	<u>0.6716</u>	<u>66.43</u>	**54.78**
Token Transforming	37.57	29.74	0.7139	60.16	52.97	42.47	31.03	0.6815	65.15	53.30
DynamicViT	41.09	24.88	0.8382	63.03	52.50	<u>43.55</u>	29.04	0.7457	65.75	54.36
SiT	<u>41.14</u>	27.59	0.7519	<u>65.03</u>	49.88	42.35	27.90	0.7193	66.00	51.86
MUTE (ours)	**42.17**	**30.24**	<u>0.6958</u>	**65.11**	**54.45**	**43.85**	**31.57**	**0.6698**	**66.71**	<u>54.73</u>

Table 4: Multi-task head-only evaluation with frozen ImageNet-1k reducer and backbone.

MUTE attains the best reducer score in eight of the ten task and keep-ratio cells and is within noise of the leader in the remaining two, where ToMe leads NYU-Depth-v2 by 0.0006 RMSE and ATC leads domain shift by 0.05%p; on the fine-grained mean MUTE leads at both keep ratios, by +0.08%p over SiT at $\alpha = 0.5$ and +0.28%p over ATC at $\alpha = 0.7$, and those razor-thin cells are labeled as such rather than claimed as separated results. The frozen-router comparison is the direct test of whether learned reducers transfer: DynamicViT and SiT carry routers trained end to end on ImageNet-1k, yet both fall behind every training-free clustering reducer on dense prediction, with DynamicViT at 0.8382 RMSE versus the training-free band of 0.6952 to 0.7139, because routers tuned to classification logits do not preserve the spatial structure that dense decoders consume. Because MUTE's reduction rule contains no architecture-specific or task-specific term, the head-only evidence tests transfer under one uniform rule and finds it holds while every method is held to the same frozen ImageNet-1k artifact.

4.5 MUTE's Design Space

We characterize MUTE's design space along three complementary measurements that jointly probe the framework's structural claims: the surrogate distortion ($\mathcal{L}$ of Equation 2), which ranks reducers by the objective MUTE minimizes; the within-framework grid, which characterizes the family variants attaining the same performance band; and the across-framework axes, which quantify the partition-axis sensitivity discussed in Proposition 3.

4.5.1 PARTITION DISTORTION

We measure the surrogate distortion of Equation 2 directly on ImageNet validation data, to check whether the objective MUTE minimizes is in fact lower for MUTE than for prior training-free reducers. Table 5 reports the per-cluster distortion across DeiT-S/16 and three self-supervised backbones; MUTE attains the lowest distortion on 11 of 12 measurements (four backbones $\times$ three stages), the exception being MAE-L stage 1 where Token Transforming leads by 0.27%.

Backbone	Method	S1	S2	S3
DeiT-S/16	PPT	31	700	5,622
	ToMe	31	<u>430</u>	<u>1,933</u>
	Token Transforming	<u>26</u>	462	2,271
	MUTE (ours)	**24**	**350**	**1,660**
DINO ViT-B/8	ToMe	3,716	4,348	2,277
	PPT	3,731	4,105	2,570
	Token Transforming	<u>1,914</u>	<u>2,393</u>	<u>1,150</u>
	MUTE	**1,688**	**1,698**	**1,031**

Backbone	Method	S1	S2	S3
MAE-B	ToMe	10,986	140,723	36,026
	ATC	<u>8,875</u>	<u>119,617</u>	<u>28,977</u>
	Token Transforming	9,037	130,270	30,547
	MUTE	**6,314**	**118,313**	**21,880**
MAE-L	ToMe	363,413	381,976	285,322
	ATC	285,364	312,573	252,485
	Token Transforming	**274,950**	308,855	237,898
	MUTE	<u>275,680</u>	**262,242**	**158,327**

Table 5: Per-cluster partition distortion at $\alpha = 0.5$, lower is better. Bold/underline: best/2nd-best.

Two structural observations follow from Table 5: PPT's distortion grows approximately $3.4\times$ faster than MUTE's at the aggressive stage on DeiT-S/16, and Token Transforming's distortion exceeds ToMe's at later stages despite its many-to-many soft-assignment aggregation, which gives the partition-level mechanism behind its training-free deficit relative to ToMe in Table 1; the same partition-side ranking holds on the three self-supervised backbones. Within MUTE's family, the Mahalanobis E-step attains lower stage-2 distortion than the K-vector cosine surrogate on every measured backbone and lower stage-3 distortion on three of the four, the exception being DINO B/8 at $+4.9\%$, the largest reductions being -12.7% on MAE-B and -17.6% on MAE-L at stage 3 (Appendix D). One negative result bounds what this measurement can support: a lower $\mathcal{L}$ does not imply a higher top-1, since that same -12.7% on MAE-B comes with 0.84%p lower top-1. We therefore read $\mathcal{L}$ as the quantity MUTE optimizes and as a diagnostic of partition quality that agrees with accuracy on most but not all cells, not as a proxy that orders accuracy monotonically. The absolute scales of Table 5 and of Table 9 in Appendix D come from two separate measurement campaigns with different per-stage normalization and image subsets, so only the relative comparisons *within* each table are meaningful.

4.5.2 MUTE'S FAMILY

We search over the five choices of Section 3.4 within the clustering family, which yields 288 variants, each evaluated at multiple random seeds on DeiT-S/16 with ImageNet-1k validation, among which we identify an eight-variant family attaining the same performance band ($\Delta \leq 0.07\%$p, multi-seed standard deviation below 0.013%p). Two axes are load-bearing and present in every family variant: the ω-weighted mean aggregation at the M-step, which Proposition 1 prescribes, and the ω-weighted K-means++ initialization. The other three admit alternatives, the partition metric splitting as K-means cosine and Mahalanobis under $W_K^\top W_K$, the iteration count across the measured settings $t \in \{1, 2, 8, \text{converged}\}$, and the K-vector layer as W_K^l and W_K^{l+1}. Cosine and Mahalanobis attain the same band on DeiT-S/16 ($\Delta \leq 0.08\%$p across all measured keep ratios) and are matched at every supervised backbone within seed noise, but on the B-sized self-supervised backbones they separate by up to 0.84%p (Appendix D), so the family band is a statement about the supervised regime and about DeiT-S/16, not a universal claim. We adopt Mahalanobis as the reference for its direct correspondence to the metric M of Equation 2 and its -25.3% per-image cost at batch size 1; that a band of instantiations rather than a single point is equivalent is the expected signature of the aggregation-side bound of Proposition 3(a).

4.5.3 AXES OUTSIDE MUTE

We extend the grid with partition classes outside MUTE's clustering-aligned axis: matching-based partition with the bipartite matching of ToMe and Hungarian matching, spectral graph partition, agglomerative hierarchical clustering with average and single linkage, and a random-partition reference. Table 6 reports the partition-class marginal means under the optimal aggregation choice.

Within MUTE's clustering axis (Mahalanobis 78.22%, K-means cosine 78.14%), the grid attains the highest top-1 band. The three non-degenerate classes outside it incur deficits at $p < 0.001$: matching-based partition 2.08%p to 2.23%p below MUTE's family and spectral graph 1.56%p below. The two agglomerative cells fall much further, by 18.5%p and 35.9%p, and we read them as failure cases rather than as competitive rules; Appendix T states why they are not a measurement of ATC. The pooled ANOVA across the 280 completed cells attributes $\eta^2 = 0.988$ of the top-1 variance to the partition axis and $\eta^2 \leq 0.006$ to the aggregation axis within MUTE's family.

Partition class	Top-1 (%)	Δ (%p)
MUTE	**78.22**	0
MUTE (cosine surrogate)	**78.14**	−0.08
Spectral graph	<u>76.67</u>	−1.56
Bipartite matching (ToMe)	76.14	−2.08
Hungarian matching	75.99	−2.23
Agglomerative (average)	59.69	−18.53
Agglomerative (single)	42.35	−35.87
Random partition	66.26	−11.96

Table 6: Partition-class marginal top-1 (%) on DeiT-S/16, $\alpha = 0.5$. Bold/underline: best/best outside family.

Three readings of Table 6 require care and Appendix T states them in full: the agglomerative cells substitute plain linkage at a fixed K into MUTE's pipeline and are *not* a re-measurement of ATC (Haurum et al., 2024), whose own number is the 77.76% of Table 1; because those cells fall below the random-partition reference, random assignment is not the floor here, so what the random cell establishes is that a framework-optimal aggregation cannot recover from a partition outside the clustering-aligned axis; and η^2 is inflated by those two degenerate cells and should not be quoted as a general effect size, though the partition axis stays dominant when they are excluded. With those readings in place, the partition-axis variance exceeds the aggregation-axis variance by a factor of approximately 165 on the full grid, which we report as an empirical measurement of the asymmetry that Proposition 3 anticipates qualitatively and not as a predicted constant.

5 Conclusion

This work casts token reduction as a partition–aggregation problem and shows that a single weighted K-means surrogate in the key-projected space, obtained from a first-order Taylor expansion under three stated approximations, is enough to place prior pruning, merging, and clustering reducers as restricted cells of one grid. The surrogate is not a bound on the reduction error of the network and we do not claim that minimizing it is equivalent to maximizing accuracy; what it provides is a common coordinate system in which the design choices of prior reducers become comparable, together with an aggregation rule and a size attribute that follow from the surrogate rather than being posited. Within that coordinate system the cell the surrogate selects is the seat of the strongest training-free reducer we measure, and the same instantiation transfers without modification to backbones spanning supervised and self-supervised pretraining and to dense-prediction tasks, while design choices inside the identified family land in an approximately equivalent band and several choices outside it incur substantial losses.

Limitations and Future Works. Four limitations bound the present formulation and instantiation. First, Eq. 2 is a surrogate, not the quantity we ultimately care about: (A1)–(A3) discard the Taylor cross term, replace the per-cluster metric M_k by the fixed $M = W_K^\top W_K$, and weight deviations by a current-attention sensitivity score, and we verify the resulting metric only operationally (Appendix D) rather than proving it approximates M_k; consistently, lower surrogate distortion does not always mean higher top-1 (§4.5.1), so we use the distortion as a diagnostic and not as a model-selection criterion. Second, M is empirically ill-conditioned, with $\lambda_{\min}(M) \approx 0$ in practice, so the range-restricted bound of Proposition 3(c) carries a constant $\lambda^+_{\min}(M)/\lambda_{\max}(M)$ that we do not control, which makes the measured sensitivity asymmetry an empirical observation rather than a consequence of the bound. Third, the visualized partitions are spatially coherent but do not always follow semantic boundaries at the level of instances, classes, or parts, so they should not be read as a semantic segmentation. Fourth, the first-order Taylor expansion weakens at the late reduction stage as the keep ratio tightens, so the approximation quality is not uniform across stages. Future work will explore stage-aware refinement of the partition update, a tighter treatment of the metric surrogate that does not require (A2), objectives that relate more directly to the reduced network's output, and architecture-specific instantiations within the identified family, which the grid search suggests is a more promising direction than further mechanism exploration.

REFERENCES

David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pp. 1027–1035, USA, 2007. Society for Industrial and Applied Mathematics. ISBN 9780898716245.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=p-BhZSz59o4.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JroZRaRw7Eu.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9650–9660, October 2021.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9640–9649, October 2021.

Joonmyung Choi, Sanghyeok Lee, Byungoh Ko, Eunseo Kim, Jihyung Kil, and Hyunwoo J. Kim. Representation shift: Unifying token compression with flashattention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20456–20466, October 2025.

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. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *European conference on computer vision*, pp. 396–414. Springer, 2022.

Yuan Gong, Yu-An Chung, and James Glass. AST: Audio Spectrogram Transformer. In *Proc. Interspeech 2021*, pp. 571–575, 2021. doi: 10.21437/Interspeech.2021-698.

Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R. Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, 2021. doi: 10.1007/s41095-021-0229-5.

Joakim Bruslund Haurum, Sergio Escalera, Graham W Taylor, and Thomas B Moeslund. Agglomerative token clustering. In *European Conference on Computer Vision*, pp. 200–218. Springer, 2024.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, June 2022.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8340–8349, October 2021.

Dmitry Ignatov, Andrey Ignatov, and Radu Timofte. Virtually enriched nyu depth v2 dataset for monocular depth estimation: Do we need artificial augmentation? In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 6177–6186, 2024. doi: 10.1109/CVPRW63382.2024.00622.

Sanghyeok Lee, Joonmyung Choi, and Hyunwoo J. Kim. Multi-criteria token fusion with one-step-ahead attention for efficient vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15741–15750, June 2024.

Yanyu Li, Geng Yuan, Yang Wen, Ju Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, and Jian Ren. Efficientformer: Vision transformers at mobilenet speed. *Advances in neural information processing systems*, 35:12934–12949, 2022.

Youwei Liang, Chongjian GE, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. EVit: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=BjyvwnXXVn_.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Dmitrii Marin, Jen-Hao Rick Chang, Anurag Ranjan, Anish Prabhu, Mohammad Rastegari, and Oncel Tuzel. Token pooling in vision transformers for image classification. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 12–21, 2023. doi: 10.1109/WACV56688.2023.00010.

Sachin Mehta and Mohammad Rastegari. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=vh-0sUt8HlG.

M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 1447–1454, 2006. doi: 10.1109/CVPR.2006.42.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Hae-Sang Park and Chi-Hyuck Jun. A simple and fast algorithm for k-medoids clustering. *Expert systems with applications*, 36(2):3336–3341, 2009.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3498–3505, 2012. doi: 10.1109/CVPR.2012.6248092.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12179–12188, October 2021.

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*, 34:13937–13949, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5389–5400. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/recht19a.html.

Cedric Renggli, André Susano Pinto, Neil Houlsby, Basil Mustafa, Joan Puigcerver, and Carlos Riquelme. Learning to merge tokens in vision transformers. *arXiv preprint arXiv:2202.12015*, 2022.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers & distillation through attention. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10347–10357. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/touvron21a.html.

Hugo Touvron, Matthieu Cord, Alaaeldin El-Nouby, Jakob Verbeek, and Hervé Jégou. Three things everyone should know about vision transformers. In *European Conference on Computer Vision*, pp. 497–515. Springer, 2022a.

Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *European conference on computer vision*, pp. 516–533. Springer, 2022b.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

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.

Siyuan Wei, Tianzhu Ye, Shen Zhang, Yao Tang, and Jiajun Liang. Joint token pruning and squeezing towards more aggressive compression of vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2092–2101, June 2023.

Xinjian Wu, Fanhu Zeng, Xiudong Wang, and Xinghao Chen. Ppt: Token pruning and pooling for efficient vision transformers. *arXiv preprint arXiv:2310.01812*, 2023.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Hongxu Yin, Arash Vahdat, Jose M. Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10809–10818, June 2022.

Geunhyeok Yu, Sunjae Jeong, Yoonyoung Choi, Jaeseung Kim, and Hyoseok Hwang. SiNGER: A clearer voice distills vision transformers further. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=exjrxVc2yT.

Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, and Guangyu Sun. Ptq4vit: Post-training quantization for vision transformers with twin uniform quantization. In *European conference on computer vision*, pp. 191–207. Springer, 2022.

Fanhu Zeng, Deli Yu, Zhenglun Kong, and Hao Tang. Token transforming: A unified and training-free token compression framework for vision transformer acceleration. *arXiv preprint arXiv:2506.05709*, 2025.

Wang Zeng, Sheng Jin, Lumin Xu, Wentao Liu, Chen Qian, Wanli Ouyang, Ping Luo, and Xiaogang Wang. Tcformer: Visual recognition via token clustering transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9521–9535, 2024. doi: 10.1109/TPAMI.2024.3425768.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5122–5130, 2017. doi: 10.1109/CVPR.2017.544.

Zhuofan Zong, Kunchang Li, Guanglu Song, Yali Wang, Yu Qiao, Biao Leng, and Yu Liu. Self-slimmed vision transformer. In *European Conference on Computer Vision*, pp. 432–448. Springer, 2022.

MARGINAL UTILITY TOKEN ELIMINATION FOR VISION TRANSFORMERS

APPENDIX

A EXTENDED RELATED WORK

This appendix expands the four mechanism families of Section 2 method by method, and states how each occupies a restricted cell of the partition–aggregation grid of Section 3.5.

Token pruning. Pruning drops tokens deemed unimportant by a per-token score, which collapses the partition to singleton self-clusters retained by a top-K rule and reduces the aggregation to the identity. DynamicViT (Rao et al., 2021) learns a predictor head and selects with a Gumbel softmax; A-ViT (Yin et al., 2022) attaches a halting head and prunes tokens whose accumulated halting probability exceeds a threshold; EViT (Liang et al., 2022) ranks by attention received from the class token; ATS (Fayyaz et al., 2022) samples proportional to attention rank; and Representation Shift (Choi et al., 2025) scores by the magnitude of the token's representation change.

Token merging. Merging fuses tokens into shared representatives rather than dropping them, so it acts on both axes. ToMe (Bolya et al., 2023) performs a single bipartite matching between alternating token sets and propagates a size attribute on subsequent attention, which places it on MUTE's M-step already and departs only on the E-step, where a matching replaces the nearest-representative rule and optimizes no stated objective. Token Transforming (Zeng et al., 2025) relaxes the hard matching to an attention-driven many-to-many soft assignment, PatchMerger (Renggli et al., 2022) merges through a small set of trainable merging vectors and SiT (Zong et al., 2022) through a learned token slimming projector from N tokens to K, and MCTF (Lee et al., 2024) adds multi-criteria attraction, bidirectional matching, and a one-step-ahead attention signal under a token-reduction consistency loss, which replaces the metric of Equation 4 with a heuristic combination of similarity, attention, and size scores.

Hybrid prune-and-aggregate. Hybrids interleave the two, branching between the axes by per-token statistics. PPT (Wu et al., 2023) switches between pruning and pooling using a variance threshold over attention scores, and TPS (Wei et al., 2023) prunes by an ATS-style score and then redistributes the pruned token's content into the surviving tokens via a similarity-weighted squeeze, with a learned predictor variant that introduces the only fine-tuned component.

Explicit clustering. Clustering reducers treat token reduction as a partition problem and therefore sit closest to MUTE, differing in the partition rule and typically aggregating by an unweighted mean rather than the weighted mean of Equation 3. ATC (Haurum et al., 2024) applies hierarchical agglomerative clustering on K-vectors, DPC-kNN (Zeng et al., 2024) clusters by density peaks under a k-nearest-neighbor kernel, K-Medoids (Park & Jun, 2009) chooses K tokens as medoids and averages the assigned tokens, and Sinkhorn (Sinkhorn & Knopp, 1967) formulates the assignment as entropy-regularized optimal transport. None of the four states the objective its partition rule minimizes, which is the gap Section 3 closes.

B EXPERIMENT DETAILS

Backbone. The pretrained DeiT-S/16 (Touvron et al., 2021) checkpoint is loaded through timm 0.4.12 with the standard `deit_small_patch16_224` weights. Self-supervised backbones (MAE-B/L/H (He et al., 2022), MoCo-v3 (Chen et al., 2021), DINO ViT-B/8 (Caron et al., 2021), BEiT-B/L (Bao et al., 2022)) use the official checkpoints with linear-probe classifier heads.

Reduction protocol. All reducers run with the matched configuration: keep ratio $\alpha = 0.5$ at reduction location $\{3, 6, 9\}$ on DeiT-S/16, applied compoundingly so that the patch-token count after block 9 is $0.5^3 \cdot 196 = 24$ plus the CLS token. For deeper backbones (depth d), the reduction location

scales proportionally to $\{d/4, d/2, 3d/4\}$. The CLS token is always preserved. Representation Shift's paper-faithful schedule departs from this convention with `[0.2, 0, 0, 0.2, 0, 0, 0.2, 0, 0, 0, 0, 0]` and is reported separately in Appendix J. We stress that this matched configuration is a protocol choice and not a neutral one: reducers whose original papers tune the location set or the per-stage schedule are evaluated away from their published operating point, which is why we additionally report the paper-faithful Representation Shift schedule, the six-location ToMe variant (Appendix P), and the audited MCTF configuration (Appendix I).

Fine-tuning schedules. We evaluate two complementary fine-tuning regimes. Schedule-A (aggressive regime): AdamW with learning rate 10^{-3} and weight decay 0.05, warmup over 20 epochs, 30 total epochs, batch size 256 with gradient accumulation 2, and AMP throughout. Schedule-B (conservative regime of standard fine-tuning): AdamW with learning rate 10^{-4} and weight decay 0.05, warmup over 5 epochs, 30 total epochs; remaining settings matching Schedule-A. In both regimes we select the best checkpoint by validation top-1.

Throughput measurement. Throughput in Table 1 is measured at batch size 256 with AMP and 8 data-loading workers, after 50 warmup iterations and 200 timed iterations.

C The Sensitivity Weight and the Size Attribute

This appendix records two things that the main text uses but does not derive: where the sensitivity weight ω_i of (A3) comes from, and why the size attribute of ToMe (Bolya et al., 2023) follows from the same first-order expansion rather than being a heuristic. Under (A3) the objective weight is $w_i = \omega_i$, and the same score also selects the initial representatives before the first E-step, so one quantity plays both roles. The monotone-descent argument of Proposition 1 and Appendix U requires the objective weights to stay fixed during the descent, which holds because ω_i is computed once per reduction stage before the first E-step.

Squared attention mass as the sensitivity weight. Expanding the attention output of block $l+1$ to first order in a perturbation of token i, the sensitivity factorizes through the subsequent attention probability P^{l+1}_{ji}, and the aggregate sensitivity of the block output to token i scales with

$$\omega_i = \sum_j (P^{l+1}_{ji})^2, \tag{5}$$

the squared attention mass that the next block places on token i. Tokens with large ω_i are the ones whose displacement costs the most at the next block, which makes ω_i the natural per-token weight in Eq. 2 and, by the same reasoning, the natural score for selecting representatives. In the reference implementation we use the current block's attention P^l in place of P^{l+1}, which avoids an additional forward pass at the next block; this is a further approximation on top of (A1)–(A3) and applies wherever ω is used.

Log-additive size bias. A representative that aggregates s_k original tokens absorbs the attention mass of all s_k of them. Under the same first-order factorization, the pre-softmax logit of the representative at the next block must therefore be raised by $\log s_k$ for the reduced sequence to reproduce the unreduced attention distribution to first order. This recovers the size attribute that ToMe (Bolya et al., 2023) introduces as a heuristic, and it also accounts for the attention-normalization change that (A1) absorbs: within the surrogate, the size attribute is the correction term, not an add-on.

Sensitivity weight versus size count. The sensitivity weight and the size count behave differently as clusters grow. A crude first-order estimate gives $\omega_i \approx s_i^2 c$ with c set by the unreduced attention distribution at block $l+1$, so the two weightings differ by a power of s_i; we present this only as a rough scaling, since c is not constant across tokens in practice and we do not measure it. At the first reduction stage all $s_i = 1$, so the size count carries no signal while ω_i still varies across tokens through c, and at later stages on cumulative clusters the two weightings separate by a power of s_i. Replacing the ω-weighted aggregation of Equation 3 by the size-weighted one, with every other axis held fixed, moves DeiT-S/16 top-1 from 78.57 to 78.59%, a 0.02%p difference that sits inside multi-seed noise and slightly favours the size-weighted variant. We therefore use ω_i throughout and

read ToMe's size weighting as a size-monotone approximation of the sensitivity weighting, and we do not claim the two are interchangeable in general.

D COSINE SURROGATE: WITHIN-FAMILY ALTERNATIVE

The K-vector cosine similarity is the within-family alternative to MUTE's Mahalanobis E-step (Section 3.4). This appendix collects the cosine reanalysis across the main classification, multi-backbone, multi-task, partition distortion, and generalization measurements, together with the agreement of both rules against the J-induced metric. The summary is that the two rules land in the same band on most axes, with two exceptions that we identify explicitly: the B-sized self-supervised backbones, where cosine is ahead by more than half a point, and dense prediction, where Mahalanobis is clearly ahead.

Equivalence identity. For $M = W_K^\top W_K$, the Mahalanobis distance in the original space equals the Euclidean distance in the K-vector projected space:

$$\|x_i - a_k\|_M^2 = (x_i - a_k)^\top W_K^\top W_K (x_i - a_k) = \|W_K(x_i - a_k)\|^2 = \|K_{\text{vec}}(x_i) - K_{\text{vec}}(a_k)\|^2. \quad (6)$$

This is an identity, and it is the reason the E-step of Proposition 2 can be evaluated as a Euclidean nearest-centroid search on the key vectors. The cosine surrogate replaces this Euclidean distance with the angular distance on the same projection, removing the magnitude term; unlike Eq. 6 this is a substitution, not an identity.

Main result on DeiT-S/16. Cosine attains 78.59% top-1 versus Mahalanobis 78.57% under the protocol of §4.2, within multi-seed noise (three-seed standard deviation 0.005%p).

Multi-backbone (Table 7). The two rules agree within 0.5%p on 10 of the 12 backbones. The two exceptions are MAE-B (-0.84%p) and BEiT-B (-0.53%p), both in favor of cosine, and both in the B-sized self-supervised regime; the L-sized and H-sized self-supervised variants (MAE-L, MAE-H, BEiT-L) agree within $\pm 0.18\%$p. We report this separation rather than folding it into the family band, and we do not have an explanation for why it is confined to the B-sized self-supervised checkpoints. Because the main tables report the Mahalanobis instantiation, the MAE-B and BEiT-B entries there are the conservative choice within the family.

Backbone	Cosine	Mahalanobis	Δ (Mah$-$Cos)
DeiT-Tiny	69.27	69.10	-0.17
DeiT-Small	78.59	78.57	-0.02
DeiT-Base	75.04	75.43	$+0.39$
ViT-S/16	77.06	77.30	$+0.24$
ViT-B/16	80.36	80.12	-0.24
MAE-B	73.23	72.39	-0.84
MAE-L	82.10	82.15	$+0.05$
MAE-H	85.03	85.11	$+0.08$
MoCo-v3	71.17	70.99	-0.18
DINO B/8	78.86	78.87	$+0.01$
BEiT-B	80.19	79.66	-0.53
BEiT-L	83.04	82.89	-0.15

Table 7: Cosine vs. Mahalanobis top-1 (%) at $\alpha = 0.5$ across all backbones.

Multi-task (Table 8). Mahalanobis dominates on dense prediction (NYU-Depth-v2 -7.4% to -11.2% RMSE, ADE-20K $+1.6\%$ mIoU at $\alpha = 0.7$); the cosine and Mahalanobis variants are within multi-seed noise on classification (iNat 2019). The two axes therefore prefer different members of the family, which is why we describe the family as approximately equivalent under the classification protocol rather than equivalent in general.

Partition distortion (Table 9). Mahalanobis attains lower distortion than cosine at stage 2 on all four measured backbones (-3.3% on DeiT-S/16, -9.3% on DINO B/8, -3.0% on MAE-B, -10.8% on MAE-L) and at stage 3 on three of the four (-2.8%, -12.7%, -17.6%). DINO B/8 stage 3 is the exception, where Mahalanobis is 4.9% *worse* than cosine; stage 1 falls within measurement

	iNat 2019 (%, ↑)		ADE-20K (mIoU %, ↑)		NYU-Depth-v2 (RMSE, ↓)	
Method	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.5$	$\alpha = 0.7$
Cosine	73.85	73.79	33.35	33.38	0.6392	0.6451
Mahalanobis	73.92	74.50	33.19	33.92	0.5922	0.5731
Δ (Mah−Cos)	$+0.07$	$+0.71$	-0.16	$+0.54$	-0.0470	-0.0720

Table 8: Cosine vs. Mahalanobis multi-task reanalysis at $\alpha \in \{0.5, 0.7\}$, under the fine-tuned protocol of Appendix M.

noise on DeiT-S/16, MAE-B and MAE-L, and shows a $+6.4\%$ cosine advantage on DINO B/8. The direction of the distortion comparison therefore does not track the direction of the top-1 comparison: on MAE-B, Mahalanobis attains 12.7% lower stage-3 distortion yet 0.84%p lower top-1 (Table 7). We read this as a concrete instance of the limitation stated in §4.5.1: the surrogate of Eq. 2 is a diagnostic of what the objective optimizes, not a predictor of accuracy.

	Cosine			Δ Mahalanobis (%)		
Backbone	S1	S2	S3	S1	S2	S3
DeiT-S/16	964	7,006	10,742	$+0.3$	-3.3	-2.8
DINO B/8	1,587	1,872	983	$+6.4$	-9.3	$+4.9$
MAE-B	6,515	121,949	25,057	-3.1	-3.0	-12.7
MAE-L	274,183	293,872	192,211	$+0.5$	-10.8	-17.6

Table 9: Cosine vs. Mahalanobis partition distortion; stage 2/3 deltas reported as relative change.

Agreement with the J-induced metric. This measurement is the operational justification for approximation (A2). We verify both surrogate rules against the autograd-computed Jacobian on 100 ImageNet validation images at the deepest reduction stage (block 9, $K = 24$ centroids). By Equation 6 the K-vector Euclidean distance *is* the Mahalanobis distance under $M = W_K^\top W_K$, so the row labeled "Mahalanobis (K-vector Euclidean)" is the reference E-step measurement. The J-induced row instantiates $M_k = J_k^\top J_k$ with the per-token Jacobian block $J_k = \partial f^{l+1}(H)_k / \partial H_k$ evaluated at the representative position; cross-position blocks $\partial f^{l+1}(H)_j / \partial H_k$ for $j \neq k$ are excluded. Including the cross-position blocks changes approximately 5.3% of token assignments in a 25-image probe; its downstream top-1 is not separately measured, and the rows of Table 10 refer to the per-token block instantiation throughout.

Partition rule	ARI vs. J-induced	NMI vs. J-induced	Top-1 (%)	Per-image time (s)
Mahalanobis (K-vector Euclidean)	0.6423	0.9128	78.57	~ 0.001
Cosine surrogate	0.6636	0.9176	78.59	~ 0.001
J-induced (per-token Jacobian block)	1.00	1.00	78.51	~ 0.96

Table 10: Within-family partition rules vs. J-induced metric on DeiT-S/16 block 9, $K = 24$.

The Mahalanobis and cosine partitions attain the same agreement band with the J-induced metric ($\Delta_{\text{ARI}} = 0.021$, $\Delta_{\text{NMI}} = 0.005$ between the two rules), and produce indistinguishable downstream top-1 (range 0.08%p, within multi-seed noise) at approximately $960\times$ the speed of the J-induced metric. We are explicit about what this does and does not license. It licenses the *operational* reading of (A2): replacing M_k by the fixed M costs 0.06%p of top-1 relative to actually computing $J_k^\top J_k$, at three orders of magnitude less cost. It does not license a subspace claim. The strict alignment between $\mathrm{range}(W_K^\top W_K)$ and the top-r singular subspace of $J^\top J$ is empirically indistinguishable from chance (mean cosine 0.12 to 0.15 across $r \in \{1, 4, 8, 16, 32\}$ versus a random baseline of 0.16). M is therefore an operational surrogate that reproduces the argmin of the J-induced E-step on this measurement, not an approximation of M_k in any norm, and we state (A2) in the main text on those terms.

Per-image latency. Mahalanobis attains 32.18 ms per image versus cosine 43.10 ms (-25.3%) at batch size 1 under AMP on an RTX 3090, measured over 500 timed iterations after 50 warmup iterations. This number is a relative comparison of the two E-step rules under an identical eager-mode single-image loop, in which kernel launch overhead rather than arithmetic dominates; it is not

a deployment latency figure for MUTE, and the deployment-relevant measurement is the batched throughput of Table 1. At batched throughput the gap narrows to $+2.1\%$ under batch 256 with AMP, with Mahalanobis still slightly ahead.

Generalization benchmark. We complement the cosine reanalysis with the linear-evaluation fine-grained and zero-shot domain-shift benchmarks of Appendix Q under the cosine E-step.

α	Fine mean (cosine)	Fine mean (Mahalanobis)	Shift mean (cosine)	Shift mean (Mahalanobis)
0.5	68.52	68.64	53.95	53.83
0.7	70.50	70.48	54.98	54.87

Table 11: Cosine surrogate on the generalization benchmark of Appendix Q. Each column reports top-1 (%) under the cosine reference within MUTE's family.

On the generalization benchmark the two rules agree within $|\Delta| \leq 0.12$%p at both keep ratios, and the per-dataset breakdown (Table 12) agrees within multi-seed noise across all twelve combinations of dataset and keep ratio.

α	Dataset	Cosine	Mahalanobis	Δ
0.5	Pet	92.07	91.99	-0.08
	Aircraft	37.32	37.62	$+0.30$
	Flowers	75.20	75.48	$+0.28$
	Food	69.48	69.49	$+0.01$
	ImageNet-R	40.73	40.65	-0.08
	ImageNet-V2	67.16	67.01	-0.15
0.7	Pet	92.34	92.42	$+0.08$
	Aircraft	39.51	39.30	-0.21
	Flowers	78.45	78.42	-0.03
	Food	71.68	71.79	$+0.11$
	ImageNet-R	41.87	41.78	-0.09
	ImageNet-V2	68.10	67.95	-0.15

Table 12: Cosine vs. Mahalanobis per-dataset on the generalization benchmark.

Summary. Cosine is a valid within-family alternative and is preferable on the B-sized self-supervised backbones; Mahalanobis is preferable on dense prediction and in the small-batch inference regime. The main paper reports Mahalanobis as the reference instantiation because it is the direct realization of the E-step under the surrogate metric M of Eq. 2 (Equation 6), whereas cosine additionally discards the magnitude term; this is a statement about which rule matches the stated objective, not a claim that it matches the true M_k-induced metric.

E Layer Stability of the K-vector Partition

The first-order expansion of Section 3.2 targets the next block's output, whose key projection is W_K^{l+1}; the reference implementation instead uses the current-block W_K^l, which avoids an additional forward pass at the next block (§3.4). We quantify how much the partition depends on this layer choice with an isolated single-stage layer-swap study at the block-9 reduction step. Fixing the patch hidden states, the seeding weights ω, and the target count K, we replace only the projection matrix with each of the twelve layers' W_K, recompute the partition under the same K-means++ initialization and nearest-centroid assignment in the projected space, and measure the Adjusted Rand Index (ARI) between every pair of the resulting partitions, averaged over $1{,}000$ ImageNet validation images.

Identical projections agree at ARI 1 by construction. Partition agreement decays smoothly and monotonically with the layer distance $|i - j|$ (Table 13): adjacent layers agree at ARI 0.70, the maximum over all pairs, declining to 0.43 for the most distant pair (layers 0 and 11).

Two points follow. First, the clean monotonic gradient shows that the projection carries genuine layer-local structure; a projection whose partition were independent of the matrix would show no distance dependence. Second, adjacent layers are maximally aligned, so among all twelve layers the

current-block W_K^l yields the partition closest to the W_K^{l+1} target that the expansion points to. We note the scope of this argument: ARI 0.70 between adjacent layers is agreement, not equivalence, so the W_K^l choice is a measured approximation whose end-to-end cost is the within-family gap reported in §4.5.2, not a free substitution.

Layer distance $\lvert i-j\rvert$	ARI	# pairs
1 (adjacent)	0.697	11
2	0.663	10
3	0.641	9
4	0.624	8
5	0.609	7
6	0.594	6
7	0.576	5
8	0.551	4
9	0.534	3
10	0.498	2
11	0.434	1

Table 13: Single-stage (block 9) partition agreement under W_K^l layer swap, by layer distance. ARI averaged over $1{,}000$ ImageNet validation images.

F First-Order Taylor Diagnostics

This appendix measures how far the first-order expansion that motivates Eq. 2 is from the quantity it expands, and is the empirical content behind approximation (A1).

The cross term absorbed by (A1). The first-order expansion of f^{l+1} around a_k produces a remainder $R = O(\|x_i - a_k\|^2)$, so the full squared deviation contains a cross term $2\langle J(a_k)(x_i - a_k), R\rangle$ in addition to $\|J(a_k)(x_i - a_k)\|^2$ and $\|R\|^2$. Approximation (A1) drops the cross term. This is a modelling choice, not a derivation: it would follow from $J(a_k)\delta$ being orthogonal to R near the representative, and residual directions near a_k are not orthogonal to the Jacobian-mapped deviation a priori. We therefore do *not* claim that minimizing Eq. 2 bounds the next block's output error, either in expectation or worst case. What Eq. 2 is, is the leading term of that error under (A1)–(A3), which is what makes it a tractable objective with an analytic M-step; the diagnostics below say how large the discarded part is.

First-order regime. We measure the empirical magnitude of the higher-order remainder $\|R\|^2$ relative to the first-order cluster spread $\|J\delta\|^2$ across reduction stages. The ratio is computed per-cluster on 100 ImageNet validation images and averaged over clusters and images.

Stage	1	2	3
Remainder $\|R\|^2$ / spread $\|J\delta\|^2$	75	1,795	2,330

Table 14: Remainder-to-spread ratio across reduction stages on DeiT-S/16, $\alpha = 0.5$.

The ratio grows from 75 at the first reduction stage to $2{,}330$ at the third, a thirty-fold increase. We read these numbers as a negative result about the expansion and state it plainly: already at stage 1 the discarded remainder is roughly two orders of magnitude larger than the retained first-order term, so Eq. 2 should be understood as a clustering objective in the key-projected space that the expansion *motivates*, and not as a quantitatively accurate model of the reduction error at any stage we measure. MUTE's empirical advantage at aggressive keep ratios accordingly cannot be attributed to the tightness of the expansion. The consistent reading is the weaker one: minimizing weighted spread in the key-projected space is a good heuristic for token reduction, the surrogate makes that heuristic explicit and gives it a derived aggregation rule, and §4.5.1 shows the reducers that minimize it tend to be, without being guaranteed to be, the accurate ones.

G M-STEP DERIVATION

We derive Equation 3 from the stationarity condition of Eq. 2. Fix the partition $C = \{C_k\}$ and the metric $M = W_K^\top W_K \succeq 0$ of approximation (A2), and consider the partial objective in a_k,

$$\mathcal{L}_k(a_k) = \sum_{i \in C_k} w_i \left\| W_K(x_i - a_k) \right\|_2^2 = \sum_{i \in C_k} w_i \, (x_i - a_k)^\top M (x_i - a_k), \qquad w_i = \omega_i > 0. \quad (7)$$

This is a convex quadratic in a_k, since $M \succeq 0$ and $w_i > 0$. Differentiating and using the symmetry of M,

$$\frac{\partial \mathcal{L}_k}{\partial a_k} = -2\, M \sum_{i \in C_k} w_i \, (x_i - a_k) = 0, \quad (8)$$

where M factors out of the sum precisely because (A2) makes the metric independent of both i and k. Any a_k with $\sum_{i \in C_k} w_i (x_i - a_k) \in \ker M$ is therefore stationary, and in particular so is the solution of

$$\sum_{i \in C_k} w_i \, (x_i - a_k) = 0 \quad \Longleftrightarrow \quad \Big(\sum_{i \in C_k} w_i \Big) a_k = \sum_{i \in C_k} w_i \, x_i, \quad (9)$$

which, since $\sum_{i \in C_k} w_i > 0$, rearranges to the weighted mean of Equation 3:

$$a_k^\star = \frac{\sum_{i \in C_k} w_i \, x_i}{\sum_{i \in C_k} w_i}. \quad (10)$$

Three remarks fix the scope of this result. First, no matrix inverse appears and the derivation never requires M to be invertible: the metric drops out of the M-step entirely, so Equation 3 is the minimizer for *any* positive semidefinite M, including an ill-conditioned or singular M. When $M \succ 0$ the minimizer is unique; when M is singular it is unique modulo $\ker M$, and Equation 3 is the canonical representative, which is also the one the implementation computes. Second, the weights enter the M-step but not the E-step: because $w_i > 0$ does not depend on k, scaling the per-token term leaves $\arg\min_k \|x_i - a_k\|_M^2$ unchanged (Proposition 2). Third, under the constant-attention approximation $\omega_i \approx s_i^2 c$ with c common across tokens, Equation 3 reduces to a size-based average with weights s_i^2, encoding the same monotone dependence on cluster size as the s_i-weighted mean that ToMe (Bolya et al., 2023) adopts as a heuristic. Within the surrogate it is not a heuristic but the minimizer, which is what we mean in §3.5 by saying that the size attribute is derived rather than posited.

H PROOF OF PROPOSITION 3

We restate the proposition that Section 3.3 summarizes, then prove it.

Proposition 3 (Partition-aggregation sensitivity asymmetry)**.** *For the objective of Equation 2 with fixed M and fixed weights $\{w_i\}$:*

(a) Aggregation gap (identity, with upper bound). For any aggregation $A = \{a_k\}$ at a fixed partition C,

$$\mathcal{L}(C, A) - \mathcal{L}(C, A^\star(C)) = \sum_k W_k \, \|a_k - \mu_k\|_M^2 \;\le\; \lambda_{\max}(M) \cdot W_{\max} \cdot K \cdot \sigma_a^2, \quad (11)$$

where $W_{\max} = \max_k W_k$ and $\sigma_a^2 = K^{-1} \sum_k \|a_k - \mu_k\|_2^2$, which scales as $O(\sigma_{\text{within}}^2 / |C_k|)$ for centroid-concentration rules (weighted mean, median, Tukey biweight).

(b) Partition gap (identity). With the optimal aggregation on both sides, $\mathcal{L}(C, A^\star(C)) = V_M - \mathrm{BCSS}_M(C)$ and therefore

$$\mathcal{L}(C, A^\star(C)) - \mathcal{L}(C^\star, A^\star(C^\star)) = \mathrm{BCSS}_M(C^\star) - \mathrm{BCSS}_M(C), \quad (12)$$

the between-cluster energy that C fails to capture.

(c) Ratio (bound). Dividing the gap of (b) by the upper bound of (a), and restricting both to deviations lying in $\mathrm{range}(M)$,

$$\frac{\Delta\mathcal{L}(\text{partition})}{\Delta\mathcal{L}(\text{aggregation})} \;\ge\; \frac{\lambda_{\min}^+(M)}{\lambda_{\max}(M)} \cdot \frac{\mathrm{BCSS}_2(C^\star) - \mathrm{BCSS}_2(C)}{W_{\max} \cdot K \cdot \sigma_a^2}, \quad (13)$$

where BCSS_2 is the Euclidean between-cluster sum of squares and $\lambda_{\min}^+(M)$ is the smallest nonzero eigenvalue of M.

Throughout, $M = W_K^\top W_K \succeq 0$ is the fixed metric of (A2), $w_i = \omega_i > 0$ are the weights of (A3), $W_k = \sum_{i \in C_k} w_i$ is the mass of cluster k, $W_{\max} = \max_k W_k$, and $\mu_k = a_k^\star(C) = W_k^{-1} \sum_{i \in C_k} w_i x_i$ is the M-step optimum of Equation 3. Because (A2) makes the metric common to all clusters, the derivation below holds without approximation for the objective $\mathcal{L}$ of Eq. 2; it is not a proof about the pre-(A2) objective with cluster-dependent M_k, for which we make no claim.

Weighted variance decomposition. Writing $\bar{x} = W^{-1} \sum_i w_i x_i$ with $W = \sum_i w_i$, the weighted total-variance identity gives

$$V_M = \mathrm{WCSS}_M(C) + \mathrm{BCSS}_M(C), \qquad V_M = \sum_i w_i \|x_i - \bar{x}\|_M^2, \tag{14}$$

with $\mathrm{WCSS}_M(C) = \sum_k \sum_{i \in C_k} w_i \|x_i - \mu_k\|_M^2$ and $\mathrm{BCSS}_M(C) = \sum_k W_k \|\mu_k - \bar{x}\|_M^2$. Since $\mathcal{L}(C, A^\star(C)) = \mathrm{WCSS}_M(C)$, we have $\mathcal{L}(C, A^\star(C)) = V_M - \mathrm{BCSS}_M(C)$, which is the decomposition quoted in Proposition 3.

Aggregation gap (a). For a fixed partition C and any aggregation $A = \{a_k\}$, the weighted Bessel identity (the cross term vanishes because $\sum_{i \in C_k} w_i (x_i - \mu_k) = 0$) gives the *identity*

$$\mathcal{L}(C, A) - \mathcal{L}(C, A^\star(C)) = \sum_k \sum_{i \in C_k} w_i \big(\|x_i - a_k\|_M^2 - \|x_i - \mu_k\|_M^2 \big) = \sum_k W_k \, \|a_k - \mu_k\|_M^2. \tag{15}$$

Note the factor W_k: the per-cluster displacement is amplified by the cluster mass and does not cancel. Applying the Rayleigh bound $\|y\|_M^2 \le \lambda_{\max}(M) \|y\|_2^2$ termwise and substituting $\sigma_a^2 = K^{-1} \sum_k \|a_k - \mu_k\|_2^2$ gives

$$\mathcal{L}(C, A) - \mathcal{L}(C, A^\star(C)) \;\le\; \lambda_{\max}(M) \cdot W_{\max} \cdot K \cdot \sigma_a^2, \tag{16}$$

which is part (a). For centroid-concentration aggregation rules (mean, median, Tukey biweight) the per-cluster displacement is itself small, $\|a_k - \mu_k\|_2^2 = O(\sigma_{\mathrm{within}}^2 / |C_k|)$ by standard concentration of the cluster mean, so σ_a^2 is small for the aggregation rules that populate the grid.

Partition gap (b). For the optimal aggregation at each partition, the decomposition above gives the identity

$$\mathcal{L}(C, A^\star(C)) - \mathcal{L}(C^\star, A^\star(C^\star)) = \mathrm{BCSS}_M(C^\star) - \mathrm{BCSS}_M(C), \tag{17}$$

where $C^\star$ is the minimizer of $\mathcal{L}(\cdot, A^\star(\cdot))$ over partitions into K cells. We state (b) as this identity and stop there. The original version of this step attempted to lower-bound the right-hand side by assuming $\mathrm{BCSS}_M(C) = O(V_M / N)$ for a partition that fails to separate the cluster structure and $\mathrm{BCSS}_M(C^\star) = \Theta(N \sigma_{\mathrm{between}}^2)$ under cluster-separated data. Those two assumptions amount to assuming the conclusion, namely that a bad partition is much worse than the optimal one, so we do not use them.

Asymmetry ratio (c). Combining the identity (b) with the upper bound (a), and lower-bounding the M-norm by the Rayleigh bound restricted to $\mathrm{range}(M)$,

$$\frac{\Delta\mathcal{L}(\text{partition})}{\Delta\mathcal{L}(\text{aggregation})} \;\ge\; \frac{\lambda_{\min}^+(M)}{\lambda_{\max}(M)} \cdot \frac{\mathrm{BCSS}_2(C^\star) - \mathrm{BCSS}_2(C)}{W_{\max} \cdot K \cdot \sigma_a^2}, \tag{18}$$

where BCSS_2 is the Euclidean between-cluster sum of squares and $\lambda_{\min}^+(M)$ is the smallest *nonzero* eigenvalue of M; the inequality holds for the component of the centroid displacements lying in $\mathrm{range}(M)$. The restriction is necessary rather than cosmetic. $M = W_K^\top W_K$ is empirically ill-conditioned, with $\lambda_{\min}(M) \approx 0$ in practice, so the unrestricted version of the bound is uninformative. This is the reason Proposition 3(c) is stated as a conditional, range-restricted inequality whose constant $\lambda_{\min}^+(M)/\lambda_{\max}(M)$ we do not control, and the reason we describe the partition–aggregation asymmetry as a qualitative consequence of the surrogate rather than a quantitative prediction. In particular we make no claim of the type $\Omega(K)$: nothing in the argument produces a factor that grows with the cluster count, and the numerator above is a data-dependent BCSS difference, not a function of K. The factor of approximately 165 measured on the grid search (§4.5.3) is an empirical observation whose magnitude depends on which partition classes enter the grid; it is not a value the bound predicts, and we do not relate it to K.

I MCTF HYPERPARAMETER AUDIT

The official MCTF codebase exposes seven hyperparameters with default values that differ from the paper-effective values; we report the audit in Table 15.

Hyperparameter	Codebase default	Paper-effective	Top-1 contribution
τ_{info} (informativeness temperature)	1	0.05	+7.99%p (dominant)
τ_{size} (size temperature)	1	0.025	(with τ_{info})
One-step-ahead attention (OSA)	off	off	0
Bidirectional matching	off	off	0
Minimum-token threshold	0	0	0
r schedule	compounding	compounding	0
Fine-tuning consistency loss weight	N/A	N/A (training-free)	0
Top-1 at codebase default			68.74
Top-1 at paper-effective τ_{info}, τ_{size}			**76.73**

Table 15: MCTF audit on DeiT-S/16, $\alpha = 0.5$: codebase default vs. paper-effective top-1.

We additionally audited ATC, PPT, Token Transforming, and DPC-kNN against their official codebases and reference papers; the default configurations match the paper-effective values within rounding (no analogous +7.99%p disparity exists for the other reducers). The MCTF row in Table 1 reports the paper-effective 76.73%; the codebase default 68.74% serves as the calibration point. We report this audit because it illustrates how much of a reducer's measured standing can depend on configuration rather than mechanism, and the same caution applies to our own numbers for reducers whose published operating point differs from our unified protocol.

J REPRESENTATION SHIFT SCHEDULE VERIFICATION

Representation Shift (Choi et al., 2025) is designed for a schedule of 20% keep-ratio drops at blocks $1, 4, 7$ (`[0.2, 0, 0, 0.2, 0, 0, 0.2, 0, 0, 0, 0, 0]`), differing from the unified reduction protocol of Section 4.1. Under its paper-faithful schedule, we reproduce 70.902% top-1 on DeiT-S/16, matching the paper's reported 70.90% within rounding. Under the unified protocol of Table 1 ($\alpha = 0.5$ at $\{3, 6, 9\}$), the method attains 38.13%. The gap between 70.90% and 38.13% is the clearest single illustration that our unified protocol is not a neutral operating point: it is a 32.8%p swing produced entirely by the schedule, not by the reduction mechanism. We report the matched-protocol number in Table 1 so that all reducers are compared under one rule, but the paper-faithful number is the operative claim of the original paper and the 38.13% entry should not be read as a characterization of that method.

K SCHEDULE-B SENSITIVITY CHECK

We complement the training-free comparison of §4.2 with two fine-tuning regimes that favor different reducer classes. Schedule-A (AdamW, lr 10^{-3}, warmup over 20 epochs, 30 total) is the aggressive regime that lets weak reducers catch up by relearning their reduced representation, and it closes 5 to 8%p of the explicit clustering family's training-free gap, with DPC-kNN improving from 68.95 to 76.94 and K-Medoids from 70.36 to 76.15; it is the convention of the FT column of Table 1. Schedule-B (lr 10^{-4}, warmup 5 epochs, 30 total) is the conservative regime of standard fine-tuning, which preserves a strong reducer's pretrained alignment. Table 16 reports both regimes for the full eleven-reducer set. That MUTE leads under both indicates the partition structure it selects is not an artifact of one optimization regime; we do not read it as evidence that the surrogate of Equation 2 is what the fine-tuning recovers.

MUTE is first under both schedules, and ATC is second under both. Below the top two the ordering does change: under Schedule-A the third through fifth ranks are ToMe, PPT, and Token Transforming, whereas under Schedule-B they are PPT, Token Transforming, and DPC-kNN, with ToMe dropping to sixth and DPC-kNN gaining the most (+1.78%p). The reducers that benefit most from the conservative schedule are the explicit clustering methods, which is the opposite of what Schedule-A favors, so we describe the two schedules as complementary rather than as one preserving the other's ranking. Ten of the eleven reducers improve under Schedule-B, with a mean

Method	Schedule-A (%)	Schedule-B (%)	Δ (%p)
ToMe	78.36	78.63	+0.27
PPT	78.32	78.78	+0.46
MCTF (audited)	77.50	77.66	+0.16
ATC	78.93	79.36	+0.43
Token Transforming	77.91	78.74	+0.83
Representation Shift	74.69	75.86	+1.17
DPC-kNN	76.94	78.72	+1.78
K-Medoids	76.15	77.10	+0.95
Top-K	77.55	77.85	+0.30
Sinkhorn	64.42	60.50	−3.92
MUTE (ours)	**79.28**	**79.55**	+0.27
mean (excluding Sinkhorn)			+0.66

Table 16: Eleven training-free reducers under Schedule-B; Schedule-A is the main-table convention.

improvement of $+0.66$%p excluding the single negative entry. Sinkhorn is that entry, collapsing by 3.92%p under the lower learning rate. We do not have a verified explanation for this collapse and do not offer one; it is a setup-specific outlier that we report as measured, and it is why we quote the mean improvement excluding it rather than over all eleven reducers.

The Schedule-A column here reads 79.28% for MUTE while Table 1 reports 79.24%. The two entries are independent fine-tuning runs of the same configuration measured in two campaigns; we keep each campaign's own value rather than unifying them, and we note the 0.04%p offset as the scale of run-to-run variation that should be assumed for all fine-tuned entries in this paper.

L MAE-B Multi-Seed Resolution

Method	Run 1	Run 2	Run 3	Mean ± std
ToMe	63.50	63.55	63.54	63.53 ± 0.02
ATC	73.19	73.22	73.20	73.20 ± 0.03
Token Transforming	72.16	72.21	72.17	72.18 ± 0.03
MUTE (Mahalanobis reference)	72.40	72.37	72.39	72.39 ± 0.02
MUTE (cosine variant)	73.22	73.24	73.23	$\mathbf{73.23 \pm 0.02}$

Table 17: MAE-B top-1 across three runs at $\alpha = 0.5$, mean ± std. Runs differ only in the clustering seed; the two MUTE variants were measured in separate campaigns, so the columns are not paired across rows.

Table 17 reports both MUTE variants, the Mahalanobis reference of §4.1 and the *cosine* within-family alternative; MAE-B is one of the two backbones where the two within-family variants separate (Appendix D). Read together with Table 7, the picture on MAE-B is the following. Under the cosine variant, MUTE attains 73.23 ± 0.02 against ATC's 73.20 ± 0.03, leading in three of three seeds by a mean margin of 0.03%p. Under the Mahalanobis variant, MUTE attains 72.39 ± 0.02 over its own three runs and ATC leads by 0.81%p, a deficit an order of magnitude above that spread and therefore stable rather than a single unlucky run. We state both. The 0.03%p cosine margin is an order of magnitude smaller than the 0.81%p Mahalanobis deficit, so the honest summary of MAE-B is that MUTE and ATC are tied at the resolution of our measurements under the better within-family variant, and that ATC is ahead under the other; we do not claim MAE-B as a win.

M Backbone Fine-Tuned Multi-Task Protocol

This appendix reports the original mixed-protocol variant of the multi-task comparison of §4.4. Under this protocol, iNat 2019, ADE-20K, and NYU-Depth-v2 train the backbone together with the task head under the task-standard schedule; the fine-grained column uses the 50 epoch linear probe of Appendix Q; domain shift remains zero-shot. The head-only protocol of the main text isolates the reducer's transferability under one uniform rule; the protocol here instead measures the achievable task performance when task adaptation is allowed. The MUTE row here reports the Mahalanobis E-step reference, whose cosine counterpart under the same protocol appears in Table 8; the head-only

table of §4.4 reports the cosine variant instead, as stated there. Because the two protocols and the two within-family variants are both in play, we label each table with the variant it reports rather than presenting a single MUTE number across the paper.

	$\alpha = 0.5$					$\alpha = 0.7$				
Method	iNat (↑)	ADE (↑)	NYUd (↓)	FG (↑)	DS (↑)	iNat (↑)	ADE (↑)	NYUd (↓)	FG (↑)	DS (↑)
Full DeiT-S/16	74.88	34.25	0.6206	71.29	55.17	74.88	34.25	0.6206	71.29	55.17
ToMe	70.51	32.71	0.6587	67.87	50.76	71.75	33.13	0.6425	70.03	54.71
ATC	72.28	32.90	0.6612	67.62	52.46	73.01	33.04	0.6558	69.93	54.63
Token Transforming	71.83	32.58	0.6731	64.01	50.63	72.50	32.87	0.6645	69.07	54.45
MUTE (ours)	**73.92**	**33.19**	**0.5922**	**68.64**	**53.83**	**74.50**	**33.92**	**0.5731**	**70.48**	**54.87**

Table 18: Multi-task comparison under the backbone fine-tuned protocol.

Under this protocol MUTE attains the best reducer score in all ten cells, with 73.92% top-1 on iNat 2019 within 0.02%p of the Three Things (Touvron et al., 2022a) ViT-S unreduced baseline of 73.9%, 33.19% mIoU on ADE-20K with a UPerNet (Xiao et al., 2018) decoder, and 0.5922 RMSE on NYU-Depth-v2 with a DPT (Ranftl et al., 2021) decoder at $\alpha = 0.5$. The advantage over the baselines is largest on dense prediction, with NYU-Depth-v2 RMSE reduced by 7.4% to 11.2% and ADE-20K mIoU raised by 1.6%p at $\alpha = 0.7$, which is consistent with the reduced encoder's spatial layout mattering more when the decoder consumes a dense grid; we offer that as a reading of the pattern, not as a demonstrated mechanism. We also observe that MUTE's NYU-Depth-v2 RMSE falls below the Full DeiT-S/16 value of 0.6206 at both keep ratios. This crossing is specific to the DeiT-S/16 backbone under the fine-tuned protocol: on the stronger DeiT-III-S/16 and DINOv2-S/14 backbones the full model recovers the lead (0.5719 and 0.4707 versus MUTE's 0.5870 and 0.4986 at $\alpha = 0.7$, Table 22), while MUTE remains the best reducer on every backbone and keep ratio. We therefore read the crossing as a regularization effect on a weaker backbone rather than a general property, and we scope the claim accordingly.

N Multi-Task Implementation Detail

We train each reducer on iNaturalist 2019 (Van Horn et al., 2018), ADE-20K (Zhou et al., 2017), and NYU-Depth-v2 (Ignatov et al., 2024), with task heads from the literature: UPerNet (Xiao et al., 2018) for segmentation and DPT (Ranftl et al., 2021) for depth. The same heads and schedules serve both the head-only protocol of §4.4, where the backbone and reducer stay frozen, and the fine-tuned protocol of Appendix M, where the backbone trains jointly with the head. The fine-grained column of Table 4 trains a linear classifier for 30 epochs at learning rate 5×10^{-3} on the frozen backbone; the fine-tuned protocol's fine-grained column instead uses the 50 epoch linear probe of Appendix Q. In Table 4, bold and underline mark the best and next-best per column among reducers, and iNat 2019, ADE-20K, and NYU-Depth-v2 use the training splits above while domain shift is evaluated zero-shot with each method's ImageNet-1k fine-tuned model, so no head is trained for that column. Two protocol asymmetries favor the learned baselines rather than ours, and we state them because the head-only protocol is otherwise described as uniform. First, the training-free reducers operate on the ImageNet pretrained backbone whereas DynamicViT and SiT use their own ImageNet fine-tuned backbone and router, which co-adapts the backbone to the reducer. Second, the training-free reducers have no fine-tuned checkpoint at the lighter keep ratio, so those domain-shift entries evaluate the moderate keep-ratio backbone with the stateless reducer set to the lighter ratio, whereas the learned reducers use keep-ratio-matched checkpoints. Neither asymmetry is in our favor, which is why we retain the comparison rather than restricting it. Figure 1(a) shows all five baselines of Table 4; its ImageNet-1k axis takes the training-free entry of Table 1 for the training-free reducers and the fine-tuned entry for DynamicViT and SiT, which have none, and it plots depth as the $\delta < 1.25$ recall rather than as RMSE so that every axis shares a percentage scale; at $\alpha = 0.5$ that recall is 65.63 for MUTE, 65.18 for ToMe, 64.86 for ATC, 63.93 for Token Transforming, 61.38 for SiT, and 55.91 for DynamicViT, against 67.41 for the unreduced backbone. The two depth metrics rank the top two methods differently at this resolution, with MUTE ahead by 0.45%p on recall and behind by 0.0006 RMSE, which is the same near-tie read through two summaries of the same predictions.

Task	Dataset (train / val)	Head	Schedule	Recovery
iNat 2019	268,243 / 24,426	Linear classifier	30 ep, lr 10^{-3}	N/A
ADE-20K	20,210 / 2,000	UPerNet (PPM + FPN, c=256)	20 ep, lr 6×10^{-5}	N/A
NYU-Depth-v2	795 / 654 (Eigen)	DPT (c=96, 4 stages)	40 ep, lr 10^{-5}	MAE-style mask

Table 19: Per-task training setup shared by the head-only and fine-tuned multi-task protocols.

MAE-style mask recovery (NYU-Depth-v2). The DPT head expects a dense token grid; the reduced backbone produces only K tokens. We recover the dense input by scattering each cluster representative to its representative spatial position (the source token with maximum assignment weight in the cluster) and filling the remaining $196 - K$ positions with a learnable mask token (D=384, trunc-normal initialization, std 0.02). The mask token is shared across the four DPT stages and trained end-to-end with the DPT decoder. This recovery step is applied identically to every reducer, so it does not favor a particular partition family, but it does mean the dense-prediction numbers measure the reducer together with this specific recovery rule rather than the reducer alone.

O Efficiency Detail

Token reduction trades accuracy against FLOPs and wall-clock throughput, and the two rank reducers differently, as the last three columns of Table 1 show: ATC computes the fewest FLOPs (2.13G) yet is the slowest training-free reducer (2,712 img/s), while ToMe and Token Transforming are the fastest (7,367 and 7,299 img/s) at higher FLOPs. MUTE is dominated under neither cost metric, attaining the highest training-free top-1 at 6,027 img/s and 2.34 GFLOPs, 52% above the unreduced DeiT-S/16's 3,965 img/s, but it is the cheapest reducer under neither, and the clustering iterations are the reason. The advantage concentrates in the moderate-to-aggressive regime: under the aggressive label MUTE realizes the compound budget as targeted (1.84 GFLOPs) while bipartite matching holds ToMe at its realizable floor (2.43 GFLOPs), so MUTE reaches budgets the strongest throughput-oriented baseline cannot at three reduction locations (Appendix P). At light reduction the regime reverses: MUTE's realized compute (4.11 GFLOPs) approaches the unreduced 4.61 GFLOPs while the clustering overhead remains, so lighter reducers are preferable there, and we scope MUTE's operating regime to moderate-to-aggressive compression rather than claiming an efficiency advantage at every budget.

P Budget Profiles Across Backbones

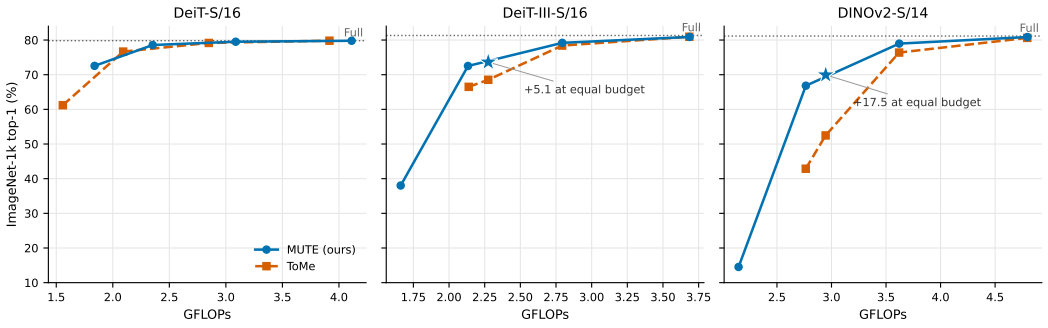


Figure 2: Top-1 versus compute budget (GFLOPs) on ImageNet-1k validation across three backbones. Stars mark MUTE evaluated at ToMe's realized token schedule, giving an equal-compute match. ToMe on DeiT-S/16 uses the six stage fair-budget schedule; DeiT-III-S/16 and DINOv2-S/14 use the standard three location protocol.

At matched three location budgets MUTE's margin over ToMe grows from -0.02%p at $\alpha = 0.9$ (within noise) through $+0.13$%p at $\alpha = 0.7$ to $+1.74$%p at $\alpha = 0.5$. Note that at $\alpha = 0.9$ MUTE is not the best method in the table: ATC leads at 79.83% and both ToMe rows are ahead of MUTE. The advantage appears only as the budget tightens, which is the claim we make, and it is not a uniform

Method	$\alpha = 0.9$	$\alpha = 0.7$	$\alpha = 0.5$	$\alpha = 0.3$
ToMe (three location)	79.81	79.38	76.83	$77.58^{\dagger}$
ToMe (six location fair budget)	<u>79.82</u>	79.16	76.67	61.20
ATC	**79.83**	<u>79.41</u>	<u>77.73</u>	<u>68.35</u>
Token Transforming	79.64	78.46	74.34	61.44
MUTE (ours)	79.79	**79.51**	**78.57**	**72.55**

Table 20: Top-1 (%) at four keep ratio labels on ImageNet-1k validation, DeiT-S/16, re-evaluated in a single campaign. The three location ToMe row realizes the bipartite floor rather than the compound target at $\alpha = 0.3$ (†); the six location row reaches the compound budgets. Bold/underline: best/2nd-best among budget-consistent cells.

advantage at every budget. The monotone growth runs in the same direction as MUTE's lower partition distortion at each reduction stage (Table 5), consistent with a lower spread compounding more favorably as more original tokens are eliminated; since distortion does not order top-1 in general (§4.5.1), we present this as a consistent reading rather than a causal account. All cells are re-evaluated in a single campaign, and the $\alpha = 0.5$ entries agree with Table 1 within 0.05%p.

The keep-ratio label is budget-inconsistent across partition families. Bipartite matching merges at most half of the surviving tokens at each stage, so ToMe's realized schedule cannot follow the compound target below $\alpha \approx 0.5$ at three locations. At $\alpha = 0.3$ the three location ToMe realizes 98, 57, 45 patch tokens per stage against the compound target of 58, 17, 5 that MUTE realizes as targeted, computing 2.43 GFLOPs against MUTE's 1.84: the same label assigns ToMe nine times more final tokens and 32% more compute, so the dagger cell of Table 20 is not a budget-matched comparison. The same floor explains ToMe's non-monotone label curve, since the $\alpha = 0.3$ label realizes more tokens than the $\alpha = 0.5$ label (45 versus 25 final patch tokens). The six location row is the fair-budget workaround: doubling the reduction locations lets bipartite matching reach the compound budgets, and at the aggressive end MUTE attains 72.55% at 1.84 GFLOPs against the six location ToMe's 61.20% at 1.56 GFLOPs; linear interpolation of the six location curve at MUTE's budget gives 69.3%, a $+3.2$%p margin at equal compute. In the mid budget range, however, the six location curve rises *above* MUTE's three location curve by up to 1.2%p (76.67% at 2.09 GFLOPs against MUTE's interpolated 75.5%), so on DeiT-S/16 the two frontiers interleave and ToMe with six locations is the better choice in that range. We report this because it is the sharpest limitation of the DeiT-S/16 comparison: granted twice the reduction locations, the bipartite family is competitive with MUTE at moderate budgets, and MUTE's DeiT-S/16 advantage is specific to the matched-location protocol and to the aggressive end. All aggressive-regime comparisons below are therefore made either at ToMe's realized schedule (equal per-stage budgets) or at $\alpha \in \{0.5, 0.7, 0.9\}$, where the realized schedules of the two families approximately coincide and the comparison is budget matched by construction.

Modern backbones. DeiT-III-S/16 (Touvron et al., 2022b) and DINOv2-S/14 (Oquab et al., 2024) both use LayerScale, which we fold losslessly into the attention output projection and the MLP output projection; the folded checkpoints reproduce the published top-1 within 0.09%p (81.31 versus 81.4) and 0.05%p (81.15 versus 81.1). DINOv2 classification uses the official linear head on the concatenation of the class token and the patch average; after reduction the patch average is taken over the surviving tokens under the same rule for every method.

Backbone	Method	$\alpha = 0.9$	$\alpha = 0.7$	$\alpha = 0.5$	$\alpha = 0.3$	BM
DeiT-III-S/16 (Full 81.31)	ToMe	80.95	78.41	66.50	68.53	68.53
	ATC	**81.16**	**79.48**	71.39	33.04	
	MUTE (ours)	80.89	79.19	**72.52**	38.03	**73.64**
DINOv2-S/14 (Full 81.15)	ToMe	80.63	76.39	42.87	52.48	52.48
	ATC	80.86	78.44	65.02	14.07	
	MUTE (ours)	**80.86**	**78.97**	**66.83**	14.53	**69.94**

Table 21: Top-1 (%) on ImageNet-1k validation for the modern backbones. BM denotes MUTE evaluated at ToMe's realized $\alpha = 0.3$ schedule (equal-compute match). The raw $\alpha = 0.3$ column is budget-inconsistent across families and is shown for completeness only.

At $\alpha \in \{0.5, 0.7, 0.9\}$ the realized schedules of MUTE and ToMe coincide, so those columns are budget matched by construction. MUTE's margin over ToMe at $\alpha = 0.5$ grows from $+1.74$%p on DeiT-S/16 to $+6.02$%p on DeiT-III-S/16 and $+23.95$%p on DINOv2-S/14. At ToMe's minimum realizable budget, MUTE leads by $+5.11$%p on DeiT-III-S/16 and $+17.46$%p on DINOv2-S/14. ATC is ahead of MUTE at $\alpha \in \{0.7, 0.9\}$ on DeiT-III-S/16 and at $\alpha = 0.7$ on DINOv2-S/14, so the modern-backbone result is a clear separation from the bipartite family at tight budgets rather than a dominance over all reducers at all budgets. ToMe's apparent non-monotonicity on DINOv2-S/14, where $\alpha = 0.5$ scores below $\alpha = 0.3$, is a label artifact: its realized final token count at $\alpha = 0.3$ (58) exceeds that at $\alpha = 0.5$ (32), and the curve is monotone in the budget plane of Figure 2. The clustering compound schedule at the raw $\alpha = 0.3$ label reduces to 5 to 6 final tokens, where all clustering variants degrade on the modern backbones, MUTE included (38.03% and 14.53%); the within-family Mahalanobis variant tracks the cosine variant within 2.3%p at every matched cell.

Dense prediction on the modern backbones. We repeat the NYU-Depth-v2 protocol of §4.4 on the two modern backbones, training the full model and the reduced models through the same DPT pipeline. DINOv2-S/14 uses a 196×196 input so that its patch grid matches the 14×14 layout of the dense head; the comparison is within backbone at a fixed resolution.

Backbone	Full	MUTE α=0.7	ToMe α=0.7	MUTE α=0.5	ToMe α=0.5
DeiT-III-S/16	**0.5719**	0.5870	0.5961	0.6116	0.6368
DINOv2-S/14	**0.4707**	0.4986	0.5025	0.5373	0.5486

Table 22: NYU-Depth-v2 RMSE (lower is better) on the modern backbones under the fine-tuned protocol.

The full model leads on both modern backbones, so the crossing observed on DeiT-S/16 in §4.4 does not generalize; MUTE remains the best of the two reducers in every cell, by margins of 0.0039 to 0.0252 RMSE.

Q Generalization Benchmark: Per-Dataset Results

This appendix reports the per-dataset top-1 accuracy underlying the FG and DS columns of Table 18. Four fine-grained classification datasets (FGVC-Aircraft, Flowers-102, Food-101, Oxford-IIIT Pet) are evaluated under linear probing on each dataset's train split; two domain shift benchmarks (ImageNet-R, ImageNet-V2) are evaluated zero-shot with the frozen ImageNet-1k classifier head applied to the reduced backbone.

Setup. DeiT-S/16 pretrained on ImageNet-1k, frozen, with MUTE Mahalanobis reduction at `reduction_loc=[3,6,9]` at two keep ratios $\alpha \in \{0.5, 0.7\}$. Linear classifier (single layer): AdamW lr 10^{-3}, weight decay 0, batch size 256, 50 epochs, cosine schedule, AMP fp16. ImageNet-R uses the standard 200 class wnid list (Hendrycks et al., 2021) and selects the corresponding logits from the 1000 class head before argmax; ImageNet-V2 uses the matched-frequency split (Recht et al., 2019) with 1000 class direct argmax. All tables in this appendix use bold/underline for the best/next-best per column among reducers; Full DeiT-S/16 is reported as the unreduced upper bound.

Method	Pet	Aircraft	Flowers	Food	Mean
Full DeiT-S/16	92.50	39.75	79.92	72.97	71.29
ToMe	90.65	36.09	**75.49**	<u>69.26</u>	<u>67.87</u>
ATC	<u>91.55</u>	37.35	73.30	68.29	67.62
Token Transforming	91.33	**38.94**	63.41	62.37	64.01
MUTE (ours)	**91.99**	<u>37.62</u>	<u>75.48</u>	**69.49**	**68.64**

Table 23: Fine-grained linear evaluation at $\alpha = 0.5$, top-1 (%).

Summary. On the fine-grained mean across four datasets, MUTE attains the highest among reducers at both $\alpha = 0.5$ (68.64% versus the next-best ToMe at 67.87%) and $\alpha = 0.7$ (70.48% versus ToMe at 70.03%); the gap to the Full DeiT-S/16 upper bound narrows from 2.65%p at $\alpha = 0.5$ to

Method	Pet	Aircraft	Flowers	Food	Mean
Full DeiT-S/16	92.50	39.75	79.92	72.97	71.29
ToMe	92.40	39.27	77.02	71.42	70.03
ATC	**92.45**	39.21	76.91	71.16	69.93
Token Transforming	92.29	**40.68**	73.67	69.62	69.07
MUTE (ours)	92.42	39.30	**78.42**	**71.79**	**70.48**

Table 24: Fine-grained linear evaluation at $\alpha = 0.7$, top-1 (%).

Method	ImageNet-R	ImageNet-V2	Mean
Full DeiT-S/16	41.90	68.43	55.17
ToMe	37.80	63.72	50.76
ATC	39.62	65.29	52.46
Token Transforming	37.72	63.53	50.63
MUTE (ours)	**40.65**	**67.01**	**53.83**

Table 25: Zero-shot domain shift at $\alpha = 0.5$, top-1 (%).

0.81%p at $\alpha = 0.7$, reflecting the expected capacity trade-off of token reduction. On the domain-shift mean, MUTE attains the highest at both keep ratios, exceeding the next-best baseline ATC by 1.37%p at $\alpha = 0.5$ and the next-best baseline ToMe by 0.16%p at $\alpha = 0.7$. The $\alpha = 0.7$ margins are small: on the domain-shift mean the four reducers span 0.42%p and on the fine-grained mean 1.41%p, so at the light keep ratio these benchmarks separate the reducers weakly, and only the $\alpha = 0.5$ column carries a margin we would describe as resolved. The within-family comparison between Mahalanobis and the cosine rule on this benchmark is reported in Appendix D.

R Qualitative Token Visualization

We visualize the partition assignments produced by MUTE and the matched baseline reducers on eight ImageNet validation images. Each subfigure shows the input image, its PCA feature, and the per-token partition at the three reduction stages of the matched protocol of §4.1; the four rows correspond to the reducers PPT, ToMe, Token Transforming, and MUTE, and tokens assigned to the same cluster share the same color within each panel. These panels are illustrative and are selected to span a range of scene types; they are not a quantitative measurement, and as noted in the Limitations the partitions are spatially coherent without being semantic segmentations.

S Partition Distortion on Self-Supervised Backbones

The ATC entry on DINO ViT-B/8 in Table 5 is omitted because the hierarchical agglomerative clustering of ATC scales as $O(N^2)$ in the input token count, and DINO ViT-B/8 with patch size 8 produces $N = 784$ tokens per image, which exceeds the per-image budget at which the other ATC measurements were collected. The omission affects one of the twelve backbone–method–stage groups in that table; we note it explicitly rather than leaving a blank, because ATC is the strongest baseline on several other axes and its absence on the backbone with the largest token count is a gap in the comparison rather than a neutral omission. Across the cells that are populated, MUTE attains the lowest distortion on 11 of the 12 backbone $\times$ stage measurements, the exception being MAE-L stage 1, where Token Transforming is lower by 0.27% (§4.5.1).

T Readings of the Partition-Class Grid

Three readings of Table 6 require care, and we state them here rather than leave them to inference.

The agglomerative rows are not a measurement of ATC. ATC (Haurum et al., 2024) also uses agglomerative linkage and attains 77.76% in Table 1, well inside the competitive band, whereas the agglomerative cells of Table 6 fall to 59.69% and 42.35%. The two are not measurements of the same object. ATC couples linkage with its own similarity graph, stopping criterion, and per-stage token budget; the grid cells instead hold every other axis at MUTE's setting and substitute only

Method	ImageNet-R	ImageNet-V2	Mean
Full DeiT-S/16	41.90	68.43	55.17
ToMe	41.52	<u>67.89</u>	<u>54.71</u>
ATC	<u>41.72</u>	67.54	54.63
Token Transforming	41.53	67.37	54.45
MUTE (ours)	**41.78**	**67.95**	**54.87**

Table 26: Zero-shot domain shift at $\alpha = 0.7$, top-1 (%).

the partition rule with plain average- or single-linkage agglomeration at a fixed K, a configuration in which single linkage chains into one dominant cluster. The deficit therefore measures the naive linkage substitution inside MUTE's pipeline and says nothing about the ATC method, whose own reported and reproduced numbers are the ones in Table 1.

Random partition is not the floor. Because the two agglomerative cells fall below the random-partition reference at 66.26%, random assignment is not the lower end of this table: a mis-specified partition rule can be worse than no partition structure at all. What the random cell establishes is therefore not a floor but a control, namely that a framework-optimal aggregation cannot recover from a partition that exits the clustering-aligned axis.

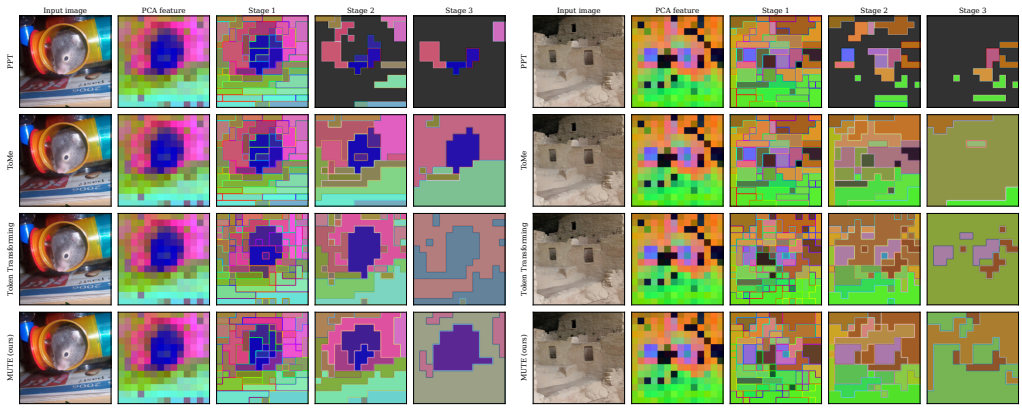
The effect size is inflated by the degenerate cells. The pooled $\eta^2 = 0.988$ quoted in §4.5.3 is computed over all 280 completed cells, including the two degenerate agglomerative cells whose 18.5 and 35.9 percentage-point deficits dominate the between-class sum of squares. Excluding them, the partition axis still accounts for the dominant share of the top-1 variance and the ordering of the remaining classes is unchanged, but the numerical value of η^2 drops and should not be quoted as a general effect size for the partition axis. The same caution applies to the factor of approximately 165 between the partition-axis and aggregation-axis variances: it is an empirical property of this grid, it depends on which partition classes enter it, and Appendix H states that no such constant is predicted by Proposition 3.

U Monotone Convergence of the Coordinate Descent

We verify that alternating the M-step of Proposition 1 and the E-step of Proposition 2 produces a non-increasing sequence of the objective $\mathcal{L}$ of Equation 2, with termination in finitely many iterations from any initialization. We note at the outset that this is the textbook argument for weighted Lloyd's algorithm under a fixed metric, which is what the surrogate of Eq. 2 is after approximations (A2) and (A3); we include it for completeness and claim no novelty for it. It is also the reason §3.3 describes the procedure as "EM-style" in the structural sense of hard-assignment coordinate descent only, and not as expectation–maximization on a latent-variable likelihood: there is no probabilistic model here, no soft posterior, and no evidence lower bound.

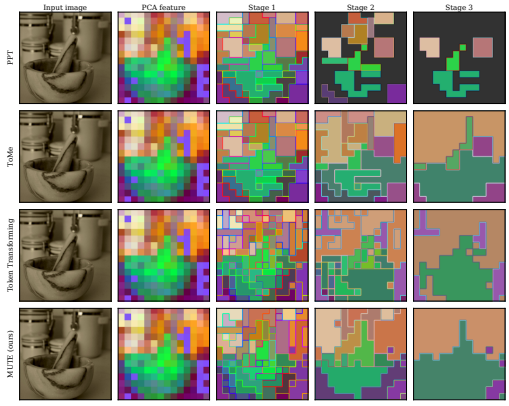
Monotone descent. Let $(C^{(t)}, A^{(t)})$ denote the partition and aggregation at iteration t. The M-step solves $A^{(t+1)} = \arg\min_A \mathcal{L}(C^{(t)}, A)$ under the weighted mean of Equation 3 (Appendix G), so $\mathcal{L}(C^{(t)}, A^{(t+1)}) \leq \mathcal{L}(C^{(t)}, A^{(t)})$. The E-step solves $C^{(t+1)} = \arg\min_C \mathcal{L}(C, A^{(t+1)})$ under the nearest-representative rule of Equation 4, so $\mathcal{L}(C^{(t+1)}, A^{(t+1)}) \leq \mathcal{L}(C^{(t)}, A^{(t+1)})$. Composing the two inequalities yields $\mathcal{L}(C^{(t+1)}, A^{(t+1)}) \leq \mathcal{L}(C^{(t)}, A^{(t)})$. Both steps minimize the same function $\mathcal{L}$, which is what makes the composition valid; this requires the fixed metric of (A2), since with a cluster-dependent M_k that changes with a_k the M-step would no longer minimize the objective it is evaluated under.

Finite termination. The partition C ranges over the finite set of assignments of N tokens to K cells, of cardinality at most K^N. The aggregation A is determined by C through Equation 3, so the state (C, A) takes only finitely many distinct values along the trajectory. Strict monotone descent forbids revisiting a state at a strictly lower objective value, so the objective sequence is non-increasing on a finite value set and the trajectory terminates. The terminal point is a local minimum in the coordinate-descent sense, fixed under both steps; as with Lloyd's algorithm it is not in general the

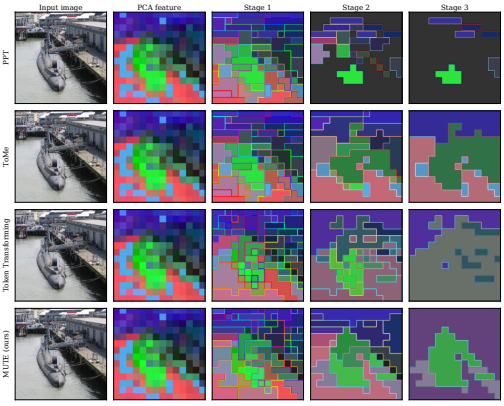


(a) Index 16,671

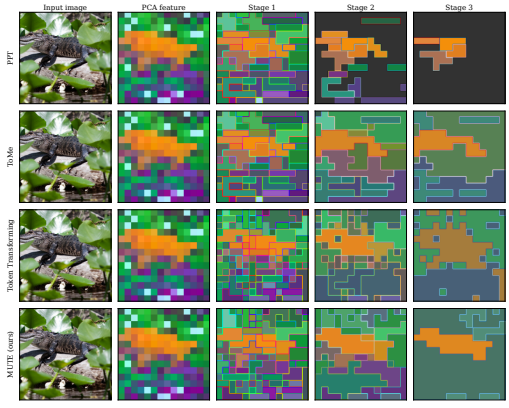
(b) Index 25,004



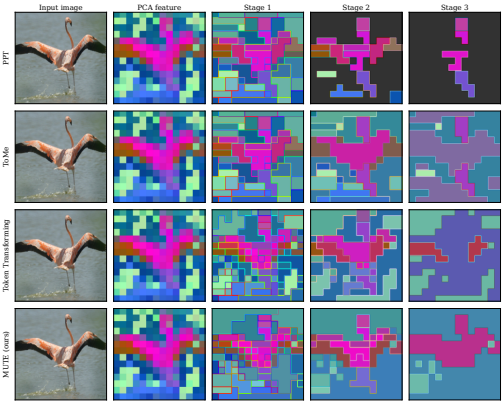
(c) Index 33,337



(d) Index 41,670



(e) Index 2,500



(f) Index 6,500

Figure 3: Qualitative partition visualization on eight ImageNet validation images. Each subfigure shows four reducers (PPT, ToMe, Token Transforming, MUTE) as rows and the input image, its PCA feature, and three reduction stages as columns; tokens belonging to the same cluster share the same color.

global minimizer of $\mathcal{L}$, and §3.3 does not claim otherwise. In practice we run a fixed small number of iterations rather than to termination, and §4.5.2 reports the accuracy across iteration counts.

V USE OF LARGE LANGUAGE MODELS

We used a large language model as a coding assistant and for language polishing. On the code side it helped implement and refactor the evaluation, measurement, and figure scripts within the shared benchmark codebase, and the authors read, ran, and verified that code themselves. On the writing side it was used to polish text the authors had written, for readability rather than for content, and to cross-check bibliographic entries. It did not generate research ideas, design experiments, or produce any of the numbers in this paper, all of which come from the authors' code and the public pretrained checkpoints. Every claim, every citation, and every number was checked by the authors, who take responsibility for the content of this work.

W ETHICS STATEMENT

This work studies token reduction for vision transformer inference, evaluated on the public ImageNet-1k, iNaturalist 2019, ADE-20K, NYU-Depth-v2, fine-grained classification, and domain shift benchmarks, and it involves no human subjects and no personal data beyond what these established datasets already contain. The contribution is a framework for deciding which tokens to group and how to aggregate them, and we see no application of it that creates a capability for harm beyond those already present in image classification and dense prediction. By reducing the compute of transformer inference at matched accuracy, the method lowers the energy cost of deployment, which we regard as a positive practical impact.

X REPRODUCIBILITY STATEMENT

The code and the complete configurations for every experiment are provided as anonymized supplementary material with this submission, and will be released publicly upon acceptance. The analytical claims are self-contained and stated with their assumptions: approximations (A1)–(A3) are named in §3.2 and their empirical cost is measured in Appendix F (A1), Appendix D (A2), and Appendix C (A3); the M-step is derived in Appendix G, the partition–aggregation asymmetry in Appendix H with its range restriction stated explicitly, and the convergence argument in Appendix U. The training-free evaluations are deterministic given the public checkpoints and the reduction protocol of Appendix B; the fine-tuning runs use the two schedules of Appendix B and select the best checkpoint by validation top-1, and Appendix K records the run-to-run spread we observe for repeated runs of the same configuration. The multi-task heads, schedules, and dense recovery mechanism are specified in Appendix N, and the modern backbone weight conversion in Appendix P. Where numbers in different tables come from different measurement campaigns or different within-family variants, we label the campaign or variant at the table rather than reconciling them silently (Appendix K, Appendix L, Appendix M).