Task-Induced Riemannian Metrics for Vision Transformer Feature Spaces

Andrew Bond *
EPFL
Lausanne, Switzerland

Ege Erdem Özlü
Koç University
Istanbul, Türkiye

Tuna Çimen
Koç University
Istanbul, Türkiye

Ilkin Umut Melanlioglu
Koç University
Istanbul, Türkiye

Tolga Birdal
Imperial College London
London, UK

Erkut Erdem
Hacettepe University
Ankara, Türkiye

Aykut Erdem
Koç University
Istanbul, Türkiye

Abstract

Methods operating on Vision Transformer (ViT) feature spaces typically rely on Euclidean distance or cosine similarity. This assumes that every direction is equally meaningful, but there is no reason to believe the true task geometry has this property. The task-sensitive geometry of the feature space is given by the pullback metric $g(F) = J(F)^\top J(F)$, where J is the Jacobian of the decoder's output fed to a task-specific distance, with respect to the features. Storing the full g is infeasible at modern scales, and for dense outputs such as depth maps even forming J is impractical. We show that whether a low-rank approximation of this metric can be learned depends on the model-decoder pair, and we characterize this with a matrix-free diagnostic $\kappa_{cap}(r)$ computable with a low number of Jacobian-vector products. For tractable pairs, we develop the *Spectral Pullback Network* (SPN), which learns a low-rank version of the metric from randomized power iteration, and we distill it into a 310K-parameter importance head that predicts token importance directly from the features. When the Jacobian spectrum is too spread out for a low-rank approximation, passing the decoder's input features through a VAE bottleneck can restore tractability. Across DPT, DINOv2, CLIP, and VGGT backbones, $\kappa_{cap}(r)$ predicts which learned-metric architectures are viable. The importance head reaches Spearman $\rho = 0.998$ on DINOv2 CLS, and our geometric token pruning reduces the additional depth error of ToMe-based token selection by 25% on DPT depth at prune ratio 0.5, without fine-tuning the ViT. Project page: `https://cyberiada.github.io/TaskInducedViTs/`.

1 Introduction

Vision Transformer [1, 2] feature representations are commonly compared using Euclidean distance or cosine similarity. This is convenient, but it assumes that all directions in feature space are equally relevant. Downstream decoders rarely behave this way. A change in a ViT feature may strongly affect a depth map, a class embedding, or a camera-pose estimate in some directions, while changes of similar Euclidean size in other directions may have almost no effect. Thus, the similarity that matters for a task is not simply "are these features close?", but rather "do these features induce similar task outputs?"

The decoder itself defines such a task-induced local geometry. Let $F \in \mathbb{R}^{N \times D}$ be a ViT feature map at some layer and let ϕ be a downstream decoder (with an optional task-specific distance) such as

*Corresponding author: andrew.bond@epfl.ch. Done while a student at Koç University.

Preprint.

DPT for depth, the CLS embedding used for classification, or VGGT for camera pose. If $J(F)$ is the Jacobian of the decoder's output with respect to F, then $v^\top J(F)^\top J(F)v = \|J(F)v\|^2$ measures how much the task output changes when F is perturbed infinitesimally in direction v. The pullback metric $J(F)^\top J(F)$ therefore tells us which feature directions the decoder is sensitive to and which it nearly ignores. It is a local, first-order object, so it says nothing about two distant feature maps. The corresponding global notion is the geodesic distance under this metric, which we only approximate (Appendix K).

Storing this metric is infeasible at ViT scale, since $J^\top J$ has $\mathcal{O}(N^2D^2) \approx 4 \times 10^{10}$ entries for a ViT-B/14 feature map. The Jacobian $J \in \mathbb{R}^{M \times ND}$ can be formed for small outputs, such as a 9-dimensional camera pose. For dense outputs such as a depth map ($M \approx 5 \times 10^4$), J has $\sim 10^{10}$ entries, and only its products with vectors are practical. This raises two questions. *Is the task sensitivity concentrated enough that a low-rank metric can capture it?* If so, *can we learn it in a form that is useful at inference time, without needing backpropagation or iterative algorithms?*

We make three contributions.

(1) A tractability diagnostic. We introduce a scalar $\kappa_{cap}(r) \in [0, 1]$, measuring the fraction of decoder sensitivity captured by the top-r singular directions of J. A second measurement, the coefficient of variation (CV) of the singular values, tells whether these directions are concentrated on a few tokens or spread across many. Both are computed offline with matrix-free Jacobian-vector products. We provide three theoretical results supporting our claims: a ceiling on the sensitivity that any rank-r restriction of the metric retains (Theorem 1), a bound showing when per-token scoring is uninformative (Theorem 2), and a convergence rate for power iteration (Theorem 3). The first and third adapt classical results to our setting.

(2) A learned low-rank metric and its distillation. The *Spectral Pullback Network* (SPN) predicts a low-rank factorization of the task-induced metric from ViT features and is trained with supervision from randomized power iteration. For applications that only need token importance, we distill [3] it into a small ($\sim$310K-parameter) *importance head* that predicts per-token scores from features alone. When no low-rank approximation is adequate in feature space, a VAE bottleneck [4] on the features the decoder uses makes one possible again.

(3) Geometric token pruning. For CLS-based decoders, we replace cosine-based token merging with a task-sensitive merge rule. For dense decoders, we combine hard pruning with a standard *Last-Layer Fusion* [5] step that re-attaches pruned tokens before decoding. A first-order error bound guides both (Theorem 4).

Empirically, the diagnostic separates the backbone–decoder pairs we evaluate into four regimes (depending on the κ_{cap} and CV) that predict which architecture is appropriate. On DINOv2 CLS, the importance head reaches Spearman $\rho = 0.998$ with its Jacobian-derived target. On DPT depth, geometric pruning reduces the additional error of ToMe-based token selection by 25% at prune ratio 0.50, while ToMe remains competitive at the lowest ratios (Table 2). These results suggest that learning *which* directions matter can be as valuable as learning better features.

2 Related Work

Metric learning and pullback geometry. Classical metric learning fits a Mahalanobis metric from pairwise constraints [6, 7] or contrastive embeddings [8]. Neural methods either output an SPD matrix directly [9, 10], sometimes only at keypoints [11, 12], which needs $O(d^2)$ memory and is prohibitive at $d = ND$, or recover the metric implicitly, for example as a pullback through a learned map [13]. Pullback metrics are also used to study the latent spaces of generative models [13, 14, 15]. Our $g = J^\top J$ is the Gauss–Newton form of the Fisher metric of the decoder output [16, 17], estimated on a frozen ViT feature space rather than over model parameters or a generative latent space. The SPN represents it implicitly through its leading eigenspace.

Token importance in ViTs. Prior analyses use input-gradient saliency, attention aggregated across layers [18], or attention weights and entropies [19] as proxies for token importance. We instead differentiate a probe map with respect to intermediate features and study the full pullback $J^\top J$. Large attention or gradient magnitude does not by itself separate task-sensitive from task-insensitive directions.

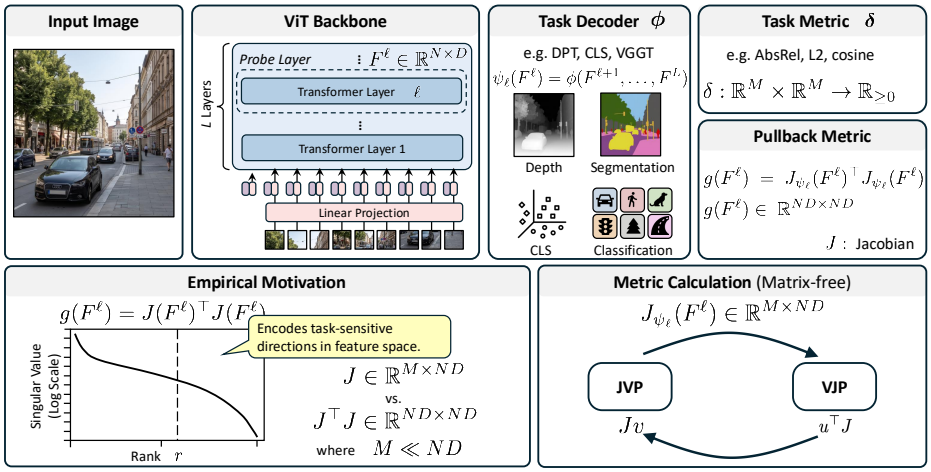

Figure 1: **The task-induced pullback metric and how we use it.** A ViT feature map $F^\ell \in \mathbb{R}^{N\times D}$ at probe layer ℓ passes through the remaining ViT layers and a task decoder ϕ (optionally followed by a task metric δ), composing into the probe map $\psi_\ell : \mathbb{R}^{N\times D} \to \mathbb{R}^M$. Its Jacobian $J_{\psi_\ell} \in \mathbb{R}^{M\times ND}$ defines the pullback metric $g(F^\ell) = J^\top J \in \mathbb{R}^{ND\times ND}$, accessed only via JVP/VJP at $\mathcal{O}(m + rq)$ cost. Rapid singular-value decay indicates that a rank-r truncation can capture most of the task sensitivity.

Token pruning and merging. ToMe [20] merges redundant tokens by cosine similarity with bipartite matching. DynamicViT [21], A-ViT [22], and EViT [23] discard tokens using learned or attention-derived scores, and Token Cropr [5] trains a scoring head end-to-end against task losses. DynamicViT, A-ViT, and Token Cropr fine-tune the backbone, whereas we keep the backbone–decoder pair frozen, so they are not direct comparisons in our setting. None of these methods measures token similarity with the decoder-induced geometry, which we do (Eq. 19), and we add theorems on when per-token scoring can succeed (Theorems 2 and 4).

3 Background and Problem Formulation

Problem formulation. At layer ℓ, a ViT produces features $F^\ell \in \mathbb{R}^{N\times D}$ ($N = 257$, $D = 768$ for ViT-B/14 at 224×224). A **task decoder** ϕ takes one or more feature layers and produces a vectorized output in $\mathbb{R}^M$, for example a DPT depth map [24] ($M \sim 50{,}000$), the final-layer CLS embedding itself ($M = D = 768$, with no classification head), or a VGGT camera pose ($M = 9$: translation, quaternion, and field of view). For a probe layer ℓ, the **probe map** ψ_ℓ composes the remaining ViT layers with the decoder,

$$\psi_\ell(F^\ell) \;=\; \phi\big(F^{\ell+1}, \ldots, F^L\big) \;\in\; \mathbb{R}^M. \tag{1}$$

Its Jacobian $J_{\psi_\ell}(F^\ell) \in \mathbb{R}^{M\times ND}$ defines the **pullback metric** on the feature space,

$$g(F^\ell) \;=\; J_{\psi_\ell}(F^\ell)^\top J_{\psi_\ell}(F^\ell) \;\in\; \mathbb{R}^{ND\times ND}. \tag{2}$$

For a perturbation $v \in \mathbb{R}^{ND}$, $v^\top g(F^\ell)v = \|J_{\psi_\ell}\, v\|^2$ is the squared change in the decoder output, so large values mark task-sensitive directions. A task distance δ on the output (scale-invariant log error, cosine, LPIPS [25], etc.) can be composed with the probe map, which we suppress in the notation. Appendix A reviews the Riemannian background [26].

What kind of metric is g? g is the pullback of the Euclidean output metric along ψ_ℓ. Since $M \ll ND$ for every decoder we consider, g is positive semidefinite, i.e. a degenerate (singular) metric [27], or equivalently a feature-to-output sensitivity form on the ambient feature space. Its kernel is the set of feature perturbations the decoder ignores. We make no manifold/sub-manifold assumptions on the features and use no tangent spaces. We claim only that the top-r eigendirections are the ambient directions of greatest decoder sensitivity. Our results (Theorems 1–4) are stated for this PSD form, and the metric deployed at inference (Eq. 5) adds an isotropic floor $\varepsilon > 0$ that makes it positive definite.

Choice of output metric. The Euclidean output metric $J^\top J$ is a modeling choice. For any positive-definite output metric $H = LL^\top$, the pullback $J^\top H J = (LJ)^\top (LJ)$ is the Euclidean pullback of the decoder followed by L, so all our methods apply with J replaced by LJ. To validate this, on the VGGT camera head, an SO(3)-aware H leaves tractability unchanged ($\kappa_{cap} = 0.99$ under both) and the token importance nearly unchanged (Spearman $\rho = 0.995$ between the two rankings, Appendix A.4). We thus use the Euclidean metric in the rest of the paper.

4 When is Metric Learning Tractable?

Before approximating or training anything, we want to know whether a rank-r metric (for some fixed choice of r) can represent the task sensitivity of a given backbone–decoder pair, and whether that sensitivity is concentrated on a few tokens, or diffuse. We now introduce two cheap measurements that answer these questions, and provide the relevant theory.

4.1 Captured Energy, Effective Rank, and Spatial Concentration

A rank-r pullback metric is most useful when its top-r directions account for most of the task sensitivity. We measure this with the *captured energy fraction*

$$\kappa_{cap}(r) = \frac{\sum_{j=1}^{r} \sigma_j^2}{\|J\|_F^2} \in [0, 1], \tag{3}$$

where σ_j is the j-th largest singular value of J. By Theorem 1, $\kappa_{cap}(r)$ is the quality of the best rank-r metric and an upper bound on the sensitivity that any restriction of g to an r-dimensional subspace can retain. We also use the *entropic effective rank* $r_{\text{eff}}(J) = \exp(H)$, computed from the spectral entropy H of the normalized squared singular values. Both $\|J\|_F^2$ and H are estimated without forming J, using Hutchinson's estimator [28] and Stochastic Lanczos Quadrature [29]. The top-r singular vectors are recovered by q rounds of randomized power iteration [30, 31]. Together these give the diagnostic in Algorithm 1 (Appendix B) at a cost of $\mathcal{O}(m + rq)$ JVP/VJP pairs, about 140 at our defaults $m = 100$, $r = 20$, $q = 2$.

The recovered top-r right singular vectors $v_1, \ldots, v_r$ also reveal which of the tokens of the feature map carry the task-sensitive directions. Splitting each $v_j \in \mathbb{R}^{ND}$ into N token blocks $v_j^{(t)} \in \mathbb{R}^D$ for $t = 1, \ldots, N$, we measure how unevenly the energy of v_j is distributed across tokens via the coefficient of variation

$$\text{CV} = \frac{1}{r} \sum_{j=1}^{r} \frac{\text{std}_t\left(\|v_j^{(t)}\|_2\right)}{\text{mean}_t\left(\|v_j^{(t)}\|_2\right)}. \tag{4}$$

Low CV (≈ 0) means that the recovered singular vectors put roughly equal mass on every token. This is the *delocalized* regime, where Theorem 2 shows that per-token scoring becomes uninformative. Higher CV means that the energy is concentrated on fewer tokens, and per-token scorers are then viable. CV therefore provides a way to predict whether a per-token architecture will work before one is trained.

4.2 Structural Characterization of Tractability

Informal Theorem 1 (Captured Sensitivity Ceiling)**.** *Consider the rank-r approximation formed from the top-r singular directions of $J(F)$, together with an isotropic floor ϵI. Among approximations of this form, the top-r eigenspace gives the minimum Frobenius error. Moreover, as $\epsilon \to 0$, the expected sensitivity captured by this approximation is exactly $\kappa_{cap}(r)$. Thus, $\kappa_{cap}(r)$ is the largest fraction of the expected task sensitivity that can be retained by restricting g to an r-dimensional subspace.*

Informal Theorem 2 (Impossibility of Factored Metrics)**.** *A* factored *(per-token) metric is one that scores each token independently and combines the per-token scores additively, with no cross-token interaction. If a unit task-sensitive direction v^* is ζ-delocalized (each token block has squared norm in $[(1-\zeta)/N, (1+\zeta)/N]$), then in expectation over the feature distribution, every factored metric assigns nearly the same sensitivity to v^* as to a uniformly random direction, up to relative error ζ. The bound scales linearly with ζ, so near-delocalized directions leave a factored scorer with proportionally little signal. (Proof: Appendix C, Theorem 2.)*

Remark. The amount of delocalization matters in practice. In §5.4 we train a small network, the importance head, to predict one score per token: how strongly that token's features contribute to the top-r singular directions of J, weighted by the singular values (Eq. 6). On DINOv2 CLS L10 ($\mathrm{CV} \approx 4.06$), its predictions match this target almost perfectly (Spearman $\rho = 0.998$). On delocalized DPT depth ($\mathrm{CV} \approx 0.026$ at the L02 and L10 probes of Table 1), the same network, reaches only $\rho \approx 0.20$. Trained instead to predict each token's full Jacobian norm $\|J_{:,t}\|_F$, it reaches only $\rho \approx 0.09$ (Appendix F). This is the weakness of per-token scoring that Theorem 2 predicts.

Informal Theorem 3 (Convergence–Tractability)**.** *Run q rounds of block power iteration on $J^\top J$ [30, 31], starting from a Gaussian random sketch with $r_0 \geq r$ columns. With high probability over the sketch, the resulting subspace approximates the true rank-r task-sensitive subspace with subspace-alignment defect $\leq C \cdot (\sigma_{r+1}/\sigma_r)^{2q}$, where C depends on the sketch but not on q. This is a geometric convergence rate in q, not a tight finite-q guarantee. The constant C grows as $\mathcal{O}(ND\,r_0)$ and can be very large, so the bound alone does not tell us how many iterations suffice. In practice we increase q until the recovered subspace stabilizes. Flat spectra need more iterations ($q \approx 15$ in our experiments). (Proof: Appendix C, Theorem 3.)*

4.3 Regime Taxonomy

We apply this analysis (written fully as Algorithm 1 in the Appendix) to six backbone–decoder–probe-layer combinations spanning DPT depth [32], DINOv2 [33], and VGGT [34]. An analogous CLIP CLS measurement appears in Appendix G. Table 1 reports $\kappa_{cap}(20)$, the per-token CV, and the full-spectrum effective rank $r_{\text{eff}}^{\text{SLQ}}$. The results fall into four distinct regimes.

Table 1: Tractability diagnostic (r=20, m=100, q=2, 50 images). Layer indices give the probe layer, *(1-layer)* marks a single-layer Depth Anything V2 (DA V2) probe, and *(full)* marks the full-resolution VGGT depth output. For the two VGGT dense rows, $\kappa_{cap}(20)$ is measured at the listed probe layer and $r_{\text{eff}}^{\text{SLQ}}$ at the input to VGGT's heads (App. E.1). CV is N/A for VGGT camera (rank-9 output). The last column gives the regime of §4.3.

Config	$\kappa_{cap}(20)$	CV	$r_{\text{eff}}^{\text{SLQ}}$	Regime
DA V2 depth, L02	95.6%	0.026	34	(ii)
DA V2 depth, L10 (1-layer)	96.6%	0.026	20	(ii)
DINOv2 CLS, L10	33.4%	4.06	270	(iii)
VGGT camera, L16	$\approx 100\%$	N/A	9	(iv)
VGGT pointcloud, L08	96.6%	0.59	212	(i)
VGGT depth, L16 (full)	76.2%	0.63	282	(i)

Four regimes. The six rows fall into four cases, listed in the last column of Table 1, and each calls for a different architecture.

(i) Rich spectrum. The two dense VGGT heads, depth and pointcloud, have effective ranks that exceed any practical rank budget by an order of magnitude, so no rank-r metric captures most of the task sensitivity. The VAE pathway of §6 addresses this regime and brings the effective rank down to $[30, 53]$ for VGGT depth.

(ii) Low rank, delocalized. The two Depth Anything V2 cases, which use the same DPT decoder at layers 2 and 10, fit within the rank budget, but their task-sensitive directions are spread evenly over tokens. By Theorem 2 a per-token metric cannot represent them, so learning the full metric needs cross-token attention (the SPN, §5.2). Per-token scores remain usable for pruning but are weak ($\rho \approx 0.20$), which is why dense pruning also relies on Last-Layer Fusion and multi-stage schedules (§7).

(iii) Marginal κ, high CV. The DINOv2 CLS Jacobian has a rich spectrum, so a rank-20 metric captures only a third of the task sensitivity. The regime is still workable because the concentration is spatial rather than spectral: a few tokens carry the task-sensitive directions, so per-token scoring is viable even though a faithful low-rank metric is out of reach. The importance head reaches $\rho = 0.998$ here, and CLIP CLS behaves similarly (Appendix G).

(iv) Structurally rank-bounded. The VGGT camera head outputs $M = 9$ scalars per view, so $\mathrm{rank}(J) \leq 9 < r$ and $\kappa_{cap}(20) \approx 100\%$ follows from the rank bound rather than from spectral concentration. This makes it a controlled setting for evaluating the SPN.

Two further observations hold across the rows. The convergence rate of Theorem 3 holds, and two power-iteration steps suffice where the spectral gap is large, while flatter spectra need more (Appendix C). A decoder that uses several feature layers is not intractable in itself, and stays tractable when those layers are close together (Appendix E.1).

5 Learning the Task-Induced Metric

When the diagnostic indicates that a rank-r metric is a meaningful target, we want a network that predicts it from features alone, so that the decoder's Jacobian is not needed at inference (as this would first require producing the output and applying backprop, losing all possible gains). We first see why simple supervision does not work, then introduce the SPN and its training procedure, and describe the small importance head used for token pruning.

5.1 Why Naive Supervision Fails

When trying to train a model to predict the metrics from features alone, there are several natural ideas. *Distance matching*, which fits $\|f_\theta(F_1) - f_\theta(F_2)\|$ to $\delta(\phi(F_1), \phi(F_2))$, constrains only the values and not the derivatives, and thus leaves the local geometry underdetermined between sampled pairs. Even a perfect match would approximate a straight-line distance, not geodesic distance. *Random-direction regression*, which fits $\|Jv\|^2$ on unit vectors v, fails because a uniform $v \in \mathbb{R}^{ND}$ overlaps the rank-r subspace with probability $r/ND \approx 10^{-5}$, so the signal is dominated by directions the metric does not model, and requiring an infeasible amount of training steps to learn anything. Theorem 3 motivates power iteration instead. Each step on $J^\top J$ amplifies the alignment with the task-sensitive subspace, so q steps will make the JVP/VJP results into a usable supervision signal.

5.2 The Spectral Pullback Network (SPN)

We parametrize the rank-r pullback metric as a neural network g_θ that takes probe-layer features F and returns a symmetric positive-definite matrix in spectral (eigen-decomposition) form:

$$v^\top g_\theta(F) v \;=\; \sum_{j=1}^{r} \underbrace{\lambda_j(F;\theta)}_{\text{learned}} \big(\underbrace{u_j(F;\theta)}_{\text{learned}}{}^\top v\big)^2 \;+\; \underbrace{\varepsilon}_{\text{fixed}} \|v\|^2, \qquad U(F;\theta)^\top U(F;\theta) = I_r. \tag{5}$$

The r mode vectors $u_j(F;\theta) \in \mathbb{R}^{ND}$ are the columns of $U \in \mathbb{R}^{ND\times r}$, which is constrained to be orthonormal. They are the network's estimates of the top-r right singular vectors v_j of $J(F)$, and the energies $\lambda_j(F;\theta) > 0$ predict the squared singular values σ_j^2. Both depend on the input features F and the trainable parameters θ. The constant $\varepsilon > 0$ is a fixed regularizer that keeps g_θ positive definite at inference, and is required for the Woodbury product below. Because this floor is isotropic, it does not change the informative rank-r eigenspace or the importance scores. The captured-energy and Eckart–Young statements of Theorem 1 therefore concern the structured part (the $\varepsilon \to 0$ limit), not the full-rank version. When $u_j \to v_j$ and $\lambda_j \to \sigma_j^2$, the SPN recovers the rank-r-plus-isotropic metric of Theorem 1.

Architecture. The network maps F through a linear projection ($D \to H$), two pre-norm multi-head attention blocks, and a per-token projection of Dr coordinates, which are reshaped to $ND \times r$ and QR-orthonormalized into $U(F;\theta)$. The energy network is a global pool, an MLP, and a Softplus, and it produces the r positive energies $\lambda(F;\theta)$. At inference, the product $g_\theta(F)\, v = \varepsilon v + U(\lambda \odot (U^\top v))$ costs $O(NDr)$ rather than $O((ND)^2)$. Per-token importance is available in closed form, $\mathrm{imp}_\theta(t) = \sqrt{\sum_j \lambda_j \, \|u_j^{(t)}\|^2}$, where $u_j^{(t)} \in \mathbb{R}^D$ denotes the t-th token block of u_j.

Cross-token attention is necessary in the delocalized regime. When the singular vectors are spread evenly across tokens (low CV), computing $u_j^\top v = \sum_t u_j^{(t)\top} v^{(t)}$ requires summing contributions from all N tokens, which per-token architectures cannot do. Theorem 2 formalizes this failure mode.

5.3 RandNLA Training

The SPN obtains supervision from power iteration using Jacobian-vector and vector-Jacobian products, without explicitly forming J. At each training step we sample a Gaussian matrix $\Omega \in \mathbb{R}^{ND\times r_0}$ with $r_0 \geq r$. We apply q rounds of power iteration $\Omega \leftarrow J^\top(J\Omega)$, using alternating JVP and VJP calls and re-orthonormalizing between rounds [35], and take the first r columns as the target mode matrix $\widehat{V}_r$. By Theorem 3, $\widehat{V}_r$ converges exponentially to the true top-r task-sensitive subspace as q grows. The training loss has two parts. The first is a Grassmannian *subspace loss* $\mathcal{L}_{\text{sub}} = 1 - \frac{1}{r}\|U(F;\theta)^\top \widehat{V}_r\|_F^2$, which measures the angle between the predicted and target subspaces and does not depend on the

order of the modes. The second is an *energy regression* $\mathcal{L}_{\text{en}} = \text{MSE}(\log \lambda(F;\theta), \log \widehat{S}^2)$ on the empirical squared norms $\widehat{S}_j^2 = \|J\hat{v}_j\|^2$ (Algorithm 2, Appendix B). When two adjacent singular values are nearly degenerate ($\sigma_j/\sigma_{j+1} < 1.05$), the modes can swap between images, so we exclude such modes from $\mathcal{L}_{\text{en}}$ for that image. The subspace loss is unaffected by this, though. If the SPN fits these targets exactly, its sensitivity ratio converges to the ceiling $\kappa_{cap}(r)$ as q grows (Corollary 1). In practice a single network shared across images does not fit every image exactly (Appendix J).

5.4 Importance Head and Inference-Time Distillation

Applications that only need a per-token importance score, such as token pruning (§7), do not need the SPN at inference time. The **importance head** $h_\theta : \mathbb{R}^{N\times D} \to \mathbb{R}^N$ is a small network (∼310K parameters) that predicts one scalar per token directly from F, with no Jacobian access and no SPN call at inference. Its target is the **exact (Jacobian-derived) importance**

$$\text{imp}^*(F)_t \;=\; \sqrt{\sum_{j=1}^{r} \sigma_j^2 \left\|v_j^{(t)}\right\|_2^2}, \tag{6}$$

to which the SPN's $\text{imp}_\theta(t) = \sqrt{\sum_j \lambda_j \|u_j^{(t)}\|^2}$ converges as $u_j \to v_j$ and $\lambda_j \to \sigma_j^2$. Since $\text{imp}^*(F)$ depends only on the top-r pairs $\{(\sigma_j, v_j)\}$, which a per-image randomized SVD recovers (Algorithm 2), we train h_θ from cached targets without instantiating the SPN (although the SPN can also be used). The σ^2-weighted top-r projection keeps spatial signal even when the plain block norm $\|J_{:,t}\|_F$ becomes uniform across tokens (Theorem 2, with the ablation in Appendix F).

Architecture and training. The head is one cross-token MHA block (4 heads), a per-token MLP, and a Softplus output, with loss

$$\mathcal{L}(\theta) = \mathbb{E}_F\Big[\big\|\log h_\theta(F) - \log \text{imp}^*(F)\big\|^2\Big] + w_{\text{rank}}\,\mathcal{L}_{\text{rank}}(h_\theta(F),\, \text{imp}^*(F))\,, \quad w_{\text{rank}} = 0.5. \tag{7}$$

The ranking term reflects that pruning depends on token ordering, not absolute scale. Theorem 4 motivates the target: with uniform residuals, the best pruning set removes the tokens with the smallest Jacobian block norms (Corollary 3). imp^* keeps the top-r part of these block norms for the probe-map Jacobian at the prune layer, which we use as a computable proxy for the decoder Jacobian at the final layer that appears in the bound.

5.5 Empirical Validation

On DINOv2 CLS L10, the importance head trained on 2,000 ImageNet-val [36] images reaches Spearman $\rho = 0.998 \pm 0.001$ against the target imp^*. Here ρ is the rank correlation over the tokens of each image, averaged over a 200-image held-out set. The head costs a small fraction of a backbone forward pass. The σ^2-weighted target is essential, as replacing it with plain block norms lowers ρ to ≈ 0.09 on DPT, predicted by Theorem 2 (ablation in Appendix F). The SPN itself reaches the deterministic supervision ceiling at the cheapest training setting on the VAE-compressed VGGT depth pipeline (§6, Appendix J).

Why not learn token sensitivity directly? A simpler alternative is to directly train on per-token sensitivity, i.e. the plain Jacobian block norms $\|J_{:,t}\|_F$, without the spectral construction. This is the ablation above. On DPT depth it reaches $\rho \approx 0.09$, against $\rho \approx 0.20$ for the σ^2-weighted top-r target, which can only be computed through the spectral construction. Where sensitivity is concentrated on a few tokens (high CV), a directly trained head can be enough. Where it is delocalized, the spectral target gives the head a signal it can learn. The SPN also provides more than a score per token, because its rank-r factors define a task-induced quadratic form on whole feature maps, and has many other potential applications.

6 Extending to Intractable Settings via VAE Compression

Now we consider the case where the diagnostic rules out a low-rank metric because the task sensitivity is spread over far more directions than any practical rank. Dense decoders for depth, point clouds, or

image reconstruction produce outputs in $\mathbb{R}^M$ with M in the tens of thousands. When the resulting Jacobian's effective rank exceeds the rank budget by an order of magnitude, no rank-r SPN can be a faithful approximation in the original feature space. This is the case for VGGT depth and pointcloud, whose effective ranks at the input to VGGT's heads are 282 and 212 (regime (i) of §4.3).

Note that the rank of a given Jacobian must be no larger than the minimum of the dimension of the input and output spaces, so $rk(J) \leq \min(M, ND)$. Since M is a property of the output space and cannot be changed, the logical way to reduce the Jacobian rank is to reduce ND. We do this with a pretrained variational autoencoder (VAE) on the ViT features which the task decoders take as input. For VGGT, let x_{vis} be the concatenated patch tokens of the four layers read by the heads (layers 4, 11, 17, 23). The VAE encoder E maps x_{vis} to a latent $z \in \mathbb{R}^{M'}$ made of 32 register tokens of dimension 128 ($M' = 4096$). Its decoder G reconstructs the features, and the frozen task head takes as input the reconstruction,

$$\tilde{\psi}(x_{\text{vis}}) \;=\; \phi\big(G(E(x_{\text{vis}}))\big) \;\in\; \mathbb{R}^M. \tag{8}$$

Every path from the features to the output passes through z, so $\operatorname{rank}(J_{\tilde{\psi}}) \leq M'$ however large M is. Empirically, the compression moves the effective spectrum into the tractable regime. VGGT depth probed at x_{vis} has $r_{\text{eff}}^{\text{SLQ}} = 282$ (regime i). With the VAE bottleneck in place, $r_{\text{eff}}^{\text{SLQ}} \in [30, 53]$ and $r_{0.90} \approx 40$ across our test configurations, a $\approx 6\times$ reduction that places the compressed metric in regime (ii). The same compression applies to VGGT pointcloud ($r_{\text{eff}}^{\text{SLQ}} = 212$ at x_{vis}). To train an SPN we probe directly at the latent. The map $z \mapsto \phi(G(z))$ has a Jacobian with only M' columns, so its right singular vectors live in $\mathbb{R}^{M'}$ and exact SVD targets are cheap to compute. In all our experiments we use the S^2-VAE [4] with power-spherical bottlenecks [37], a VAE tailored specifically to ViT features, which reaches a depth-reconstruction quality of $\delta_1 \approx 0.99$ on VGGT depth.

Approximation induced by VAE compression. The compressed pullback $g_{\tilde{\psi}}$ is the pullback of the task metric through the VAE-reconstructed features, not through the original ones, so the metric we learn here is not literally g but an approximation of it. When the VAE reconstructs the task-relevant content of the features well, the two pullbacks are expected to agree approximately in the task-sensitive directions.

Experimental validation. On the VAE-compressed VGGT depth pipeline, probing at the latent z, RandNLA with a single power-iteration step ($q = 1$) reaches a subspace alignment of $V_1 = 0.713$ with the true rank-r task-sensitive subspace. This essentially matches the deterministic *supervision ceiling* of $V_1 = 0.710$, which is the best this network reaches when given exact SVD targets at every step. A randomly initialized network has $V_1 = 0.005$, so the gap between the random and trained models reflects learned task geometry. The saturation at the cheapest setting, $q = 1$, is consistent with Theorem 3, which predicts fast convergence when the spectral gap is large. It supports both the VAE pathway and the RandNLA algorithm. Appendix J gives the full setup and ablations.

7 Token Merging and Pruning via Geometric Importance

We now use the importance head to remove or merge tokens in a frozen ViT while keeping the decoder's output close to the unpruned one. The prune ratio η is the fraction of patch tokens removed or merged. Let F^L and $\hat{F}$ be the final-layer features without and with pruning, let $\mathcal{T}$ be the set of removed tokens, and let ρ_t be the part of token t's feature update that is lost after correction (Appendix D). With $J_{:,t}(\hat{F})$ the block of the decoder Jacobian for token t and β a bound on the decoder's Hessian, a Taylor expansion (Theorem 4) gives

$$\big\|\phi(F^L) - \phi(\hat{F})\big\|_2 \;\leq\; \sum_{t\in\mathcal{T}} \underbrace{\big\|J_{:,t}(\hat{F})\big\|_F}_{\text{token importance}} \underbrace{\|\rho_t\|_2}_{\text{residual}} \;+\; \tfrac{\beta}{2} \sum_{t\in\mathcal{T}} \|\rho_t\|_2^2. \tag{9}$$

The first term is small when the removed tokens have low task sensitivity, which is what the importance head targets. The second term depends on how much of the removed tokens' content is recovered.

7.1 CLS Decoders: Soft Merge

A CLS decoder uses only the CLS token and is invariant to permutations of the other tokens, so tokens can be merged instead of removed. We follow ToMe [20]. The ηN tokens with the lowest importance scores are each merged into their nearest remaining token, with importance-weighted

Table 2: Token pruning vs. baselines, ImageNet-val @ 224 (mean ± std, 3 seeds; lower is better). Bold marks the best method per column. Importance outperforms ToMe at every ratio for CLS and at $\eta \geq 0.2$ for DPT.

Method	η=0.05	0.10	0.15	0.20	0.30	0.40	0.50
CLS cosine-distance degradation (×100), DINOv2-B/14							
Random	0.17 ±0.03	0.36 ±0.05	0.60 ±0.08	0.86 ±0.09	1.49 ±0.14	2.31 ±0.22	3.44 ±0.31
ToMe score	0.67 ±0.11	0.78 ±0.07	0.95 ±0.07	1.12 ±0.11	1.75 ±0.13	2.56 ±0.25	3.82 ±0.42
Importance	**0.001** ±0.000	**0.007** ±0.001	**0.024** ±0.002	**0.060** ±0.005	**0.25** ±0.03	**0.80** ±0.09	**2.09** ±0.13
DPT additional SILog (×100), Depth-Anything V2 single-stage [4]							
Random	4.59 ±0.09	3.70 ±0.10	**3.31** ±0.14	**3.19** ±0.19	3.16 ±0.17	3.20 ±0.15	3.29 ±0.18
ToMe score	**4.29** ±0.07	**3.66** ±0.09	3.51 ±0.17	3.50 ±0.10	3.52 ±0.13	3.67 ±0.13	4.09 ±0.13
Importance	4.88 ±0.06	4.86 ±0.07	3.85 ±0.11	3.28 ±0.15	**3.02** ±0.16	**3.01** ±0.16	**3.08** ±0.15

averaging. We replace ToMe's cosine similarity with the rank-r pullback metric restricted to a single token (Eq. 19, Appendix D). In our experiments this distance uses the SPN prediction for J. All CLS strategies share this matching step and differ only in which tokens they merge.

7.2 Dense Decoders: Hard Prune + Last-Layer Fusion

Dense decoders such as DPT [24] reassemble the tokens into a 2D map, which merging corrupts. We instead hard-prune the lowest-importance tokens at layer ℓ, freeze their features, and re-insert them at their original positions wherever the decoder needs. The frozen features miss the updates of the later layers. We use **Last-Layer Fusion (LLF)** (ex. [5]) to reduce this drift. It re-inserts the pruned tokens before the final ViT block and runs that block on the full sequence, so they attend to the updated kept tokens. LLF adds no parameters. All dense strategies (Random, ToMe score, and Importance) share this pipeline and differ only in which tokens they remove. We report the additional scale-invariant log error (SILog, $\times 100$) relative to the unpruned prediction.

7.3 Results

Table 2 shows the two settings. On DINOv2 CLS, importance-guided merging has the lowest degradation at every ratio, and at low ratios it is more than an order of magnitude below ToMe scoring. On DPT depth, importance reduces the additional error of ToMe scoring by 25% at $\eta = 0.50$ but loses at the lowest ratios. The CLS result carries over to CLIP and to ImageNet-Sketch, and importance also beats frozen-backbone baselines such as EViT (Appendices G and H). ImageNet top-1 accuracy stays between 0.66 and 0.68 for every method and ratio (0.674 unpruned). For dense depth, the 2-stage [4, 8] schedule beats ToMe scoring at every ratio and resolution on NYU-Depth-V2, by 25–35%, while single-stage results depend on the dataset. Pruning removes 12.1% of the backbone's theoretical FLOPs at $\eta = 0.20$, but a wall-clock speedup appears only at 448×448 ($1.12\times$). Appendix D discusses when importance-guided pruning wins, a non-monotone trend in Table 2, and cost.

8 Conclusion

We studied the geometry a task decoder induces on a frozen ViT feature space, given by the pullback metric $g = J^\top J$. Since g cannot be stored, and for dense outputs J cannot even be formed, we introduced a matrix-free diagnostic, $\kappa_{cap}(r)$ that measures how much of the decoder's sensitivity a rank-r metric can capture, together with a measure of how that sensitivity spreads across tokens. Across DPT, DINOv2, CLIP and VGGT probes, these two numbers sort backbone–decoder pairs into four regimes that predict which architecture is viable, and a VAE bottleneck can be used to move the intractable pairs into a tractable regime. Where the diagnostic allows it, the Spectral Pullback Network learns the metric from randomized power iteration, and a 310K-parameter importance head distills it into per-token scores that reach $\rho = 0.998$ on DINOv2 CLS. Using those scores for token pruning cuts the added depth error of ToMe-based selection by 25% at prune ratio 0.5, with the backbone and decoder left frozen. Appendices M and O give limitations and future directions.

Acknowledgements

This research was supported in part by the KUIS AI Center Research Awards to Andrew Bond. Aykut Erdem acknowledges funding from the TÜBİTAK 2247-A Program under Award 123C550. TB acknowledges support from the UKRI Engineering and Physical Sciences Research Council (EPSRC)

through the Future Leaders Fellowship [grant number MR/Y018818/1]. E. Erdem is funded by the project TUBITAK 2247-A National Outstanding Researchers Program Award No. 123C542.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[2] 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.

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

[4] Andrew Bond, Ilkin Umut Melanlioglu, Erkut Erdem, and Aykut Erdem. Beyond gaussian bottlenecks: Topologically aligned encoding of vision-transformer feature spaces, 2026.

[5] Benjamin Bergner, Christoph Lippert, and Aravindh Mahendran. Token cropr: Faster ViTs for quite a few tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[6] Eric P Xing, Andrew Y Ng, Michael I Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems*, 2002.

[7] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244, 2009.

[8] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[9] Christopher Scarvelis and Justin Solomon. Riemannian metric learning via optimal transport. In *The Eleventh International Conference on Learning Representations*, 2023.

[10] Dishank Bansal, Ricky TQ Chen, Mustafa Mukadam, and Brandon Amos. Taskmet: Task-driven metric learning for model learning. *Advances in Neural Information Processing Systems*, 36:46505–46519, 2023.

[11] Clément Chadebec, Clément Mantoux, and Stéphanie Allassonnière. Geometry-aware hamiltonian variational auto-encoder. *arXiv preprint arXiv:2010.11518*, 2020.

[12] Clément Chadebec, Elina Thibeau-Sutre, Ninon Burgos, and Stéphanie Allassonnière. Data augmentation in high dimensional low sample size setting using a geometry-based variational autoencoder. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):2879–2896, 2022.

[13] Georgios Arvanitidis, Lars Kai Hansen, and Søren Hauberg. Latent space oddity: On the curvature of deep generative models. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

[14] Binxu Wang and Carlos R Ponce. A geometric analysis of deep generative image models and its applications. In *International Conference on Learning Representations*, 2021.

[15] Yong-Hyun Park, Mingi Kwon, Jaewoong Choi, Junghyo Jo, and Youngjung Uh. Understanding the latent space of diffusion models through the lens of riemannian geometry. *Advances in Neural Information Processing Systems*, 36:24129–24142, 2023.

[16] Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

[17] Shun-ichi Amari and Hiroshi Nagaoka. *Methods of Information Geometry*. American Mathematical Society, 2000.

[18] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, 2020.

[19] Jesse Vig and Yonatan Belinkov. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP*, pages 63–76, 2019.

[20] 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.

[21] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. In *Advances in Neural Information Processing Systems*, 2021.

[22] 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*, 2022.

[23] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022.

[24] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12159–12168. IEEE, 2021.

[25] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pages 586–595. IEEE, 2018.

[26] Samuel Gruffaz and Josua Sassen. A review on riemannian metric learning: Closer to you than you imagine. *arXiv preprint arXiv:2503.05321*, 2025.

[27] Alessandro Benfenati and Alessio Marta. A singular riemannian geometry approach to deep neural networks i. theoretical foundations. *Neural Networks*, 158:331–343, 2023.

[28] Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

[29] Shashanka Ubaru, Jie Chen, and Yousef Saad. Fast estimation of tr(f(a)) via stochastic lanczos quadrature. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1075–1099, 2017.

[30] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

[31] Per-Gunnar Martinsson and Joel A Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.

[32] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything V2. In *Advances in Neural Information Processing Systems*, 2024.

[33] Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

[34] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[35] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2 edition, 2003.

[36] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[37] Nicola De Cao and Wilker Aziz. The power spherical distribution. In *Proceedings of the 37th International Conference on Machine Learning, INNF+ Workshop*, 2020.

[38] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[39] Leon Mirsky. Symmetric gauge functions and unitarily invariant norms. *The Quarterly Journal of Mathematics*, 11(1):50–59, 1960.

[40] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.

[41] Chandler Davis and William M Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[42] G. W. Stewart and Ji-Guang Sun. *Matrix Perturbation Theory*. Academic Press, 1990.

[43] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, 2019.

[44] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision*, pages 746–760, 2012.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021.

[46] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

[47] Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *International Conference on Learning Representations*, 2024.

[48] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328, 2017.

[49] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. SmoothGrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[50] Weitang Liu, Xiaoyun Wang, John D Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2020.

[51] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, 2018.

[52] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, 2013.

A Riemannian Geometry

We collect the Riemannian-geometry background used in the main text, in enough depth that a reader with basic linear algebra and calculus can follow the proofs in Appendix C. A more thorough treatment can be found in [26] or any standard text on differential geometry. We state the definitions formally, but each is followed by an *Intuition* note that reduces it to the linear algebra it amounts to in our setting, for those who only want the general idea and do not need the full formalism. We assume multivariable calculus and linear algebra, but no differential geometry or point-set topology.

A.1 Smooth Manifolds and Tangent Spaces

A **smooth manifold** $\mathcal{M}$ of dimension d is a Hausdorff topological space equipped with an atlas of charts $\{(U_\alpha, \varphi_\alpha)\}$, where each $U_\alpha \subseteq \mathcal{M}$ is open and each $\varphi_\alpha : U_\alpha \to \mathbb{R}^d$ is a homeomorphism, such that the transition maps $\varphi_\beta \circ \varphi_\alpha^{-1}$ are smooth wherever defined. Familiar examples are $\mathbb{R}^n$ itself, the n-sphere S^n, and matrix groups such as $\mathrm{SO}(n)$. In our setting, the manifold of interest is the ViT feature space at probe layer ℓ, taken as the open set $\mathbb{R}^{N\times D}$ with the global chart $\varphi(F^\ell) = \mathrm{vec}(F^\ell)$.

Intuition. The definition says that $\mathcal{M}$ can be covered by coordinate systems, each identifying a neighbourhood with an open subset of $\mathbb{R}^d$, and that the change-of-coordinate maps are smooth, so differentiation does not depend on which coordinates are used. This means that a region around a given point looks like $\mathbb{R}^d$, and we can smoothly switch between two regions while preserving smoothness. Our $\mathcal{M}$ is the vector space $\mathbb{R}^{N\times D}$ with the single global chart $\mathrm{vec}(\cdot)$, which is the degenerate case: one coordinate system, no transition maps, and every construction below reduces to linear algebra on the vectorized feature tensor. The manifold language buys us one thing, namely a metric that is allowed to vary from point to point.

At each point $p \in \mathcal{M}$, the **tangent space** $T_p\mathcal{M}$ is a d-dimensional real vector space whose elements are derivations of smooth real-valued functions at p. Concretely, an element $v \in T_p\mathcal{M}$ is the velocity $\dot\gamma(0)$ of some smooth curve $\gamma : (-\varepsilon, \varepsilon) \to \mathcal{M}$ with $\gamma(0) = p$. Two curves give the same tangent vector iff they have the same first-order Taylor expansion in any chart. The disjoint union $T\mathcal{M} = \bigsqcup_p T_p\mathcal{M}$ is itself a smooth manifold, the **tangent bundle**.

Intuition. $T_p\mathcal{M}$ is the space of first-order displacements at p, i.e. the directions along which a function can be differentiated there. For a vector space it is canonically identified with the space itself, so a tangent vector at F^ℓ is a perturbation $\Delta F \in \mathbb{R}^{N\times D}$ of the same shape, and the curve $\gamma(t) = F^\ell + t\,\Delta F$ gives it as a velocity. The definition through derivations exists because a general manifold has no ambient space in which to take differences of nearby points.

A smooth map $\phi : \mathcal{M} \to \mathcal{N}$ between manifolds induces a **differential** (or pushforward) at each p, $d\phi_p : T_p\mathcal{M} \to T_{\phi(p)}\mathcal{N}$, defined by $d\phi_p(v) = (\phi \circ \gamma)\dot{}(0)$ for any curve γ with $\dot\gamma(0) = v$. In coordinates this is the Jacobian matrix.

Intuition. The differential is the linear map that best approximates ϕ near p, in the sense that $\phi(p + \epsilon v) = \phi(p) + \epsilon\, d\phi_p(v) + O(\epsilon^2)$. In coordinates it is the Jacobian, and applying it to a single direction is a Jacobian-vector product, the primitive our algorithms are built on.

A.2 Riemannian Metrics, Lengths, and Geodesics

A **Riemannian metric** g on $\mathcal{M}$ assigns to each p an inner product $g_p : T_p\mathcal{M} \times T_p\mathcal{M} \to \mathbb{R}$ that varies smoothly with p. In a chart (U, φ) with coordinates $x^1, \dots, x^d$, g is represented by a smoothly varying symmetric positive-definite (SPD) matrix $g_{ij}(p)$, with $g_p(u, v) = \sum_{ij} g_{ij}(p)\, u^i v^j$. We use the same letter g for the metric and its coordinate matrix, $g(F) \in \mathbb{R}^{ND\times ND}$ in our case.

Intuition. A Riemannian metric is a field of quadratic forms. At each point it is a Mahalanobis-style inner product with an SPD matrix $g_{ij}(p)$, so the squared length of a perturbation v is $v^\top g(p)\, v$. It departs from the Euclidean case in two ways: the matrix need not be the identity, so directions have unequal cost and can be correlated, and it depends on p, so the same perturbation costs different amounts at different points. For $g = J^\top J$ the eigenvectors are the right singular vectors of J and the eigenvalues are σ_j^2, so length is measured in units of output change.

The metric defines an infinitesimal notion of length on $\mathcal{M}$: a smooth curve $\gamma : [0,1] \to \mathcal{M}$ has **length**

$$L(\gamma) \;=\; \int_0^1 \sqrt{g_{\gamma(t)}\big(\dot\gamma(t), \dot\gamma(t)\big)}\, dt. \tag{10}$$

The **Riemannian distance** $d_g(p,q)$ between two points is then the infimum of $L(\gamma)$ over piecewise-smooth curves with $\gamma(0) = p$ and $\gamma(1) = q$. This d_g satisfies all metric-space axioms whenever $\mathcal{M}$ is connected and g is positive definite. If g is only positive semidefinite, as for the pullback metrics below, d_g is a pseudometric, meaning that distinct points can be at distance zero.

Intuition. The length integrates the local norm $\|\dot\gamma(t)\|_{g(\gamma(t))}$ along the curve, and the distance is the infimum over connecting curves. If g were constant, say $g = L^\top L$, the integral would collapse to $d_g(p,q) = \|L(p-q)\|_2$ and the minimizers would be straight lines, which is the familiar Mahalanobis distance. The challenge of the Riemannian case is that g varies with position, so the minimizer generally bends away from the straight line towards regions where the metric is cheap.

A **geodesic** is a critical point of L in the calculus-of-variations sense. Equivalently, it satisfies the Euler–Lagrange (geodesic) equation

$$\ddot\gamma^{\,k} + \Gamma^k_{ij}(\gamma)\,\dot\gamma^{\,i}\dot\gamma^{\,j} \;=\; 0, \qquad \Gamma^k_{ij}(p) \;=\; \tfrac{1}{2}\, g^{k\ell}(p)(\partial_i g_{j\ell} + \partial_j g_{i\ell} - \partial_\ell g_{ij})(p), \tag{11}$$

where $g^{k\ell}$ is the inverse metric tensor and Γ^k_{ij} are the **Christoffel symbols**. Geodesics generalize straight lines. In flat $\mathbb{R}^n$ with the Euclidean metric, $\Gamma^k_{ij} \equiv 0$ and the geodesic equation reduces to $\ddot\gamma = 0$.

Intuition. Geodesics are the stationary points of the length functional, so the geodesic equation is the Euler–Lagrange condition of that variational problem. It is a second-order ODE in the coordinates of the curve, and the Christoffel symbols are built from first derivatives of g, which is why they vanish when g is constant and the solutions are then straight lines. Integrating this ODE is what makes exact geodesic distance impractical at $ND \approx 2 \times 10^5$, and it is why Appendix K uses a first-order surrogate.

The **exponential map** at p, $\exp_p : T_p\mathcal{M} \to \mathcal{M}$, sends a tangent vector v to $\gamma_v(1)$, where γ_v is the unique geodesic with $\gamma_v(0) = p$ and $\dot\gamma_v(0) = v$. On a sufficiently small neighbourhood, $\exp_p$ is a diffeomorphism, and its inverse $\log_p$ provides **normal coordinates** centred at p. In these coordinates the metric looks Euclidean to first order, with $g_{ij}(p) = \delta_{ij}$ and $\partial_k g_{ij}(p) = 0$.

Intuition. The exponential map integrates the geodesic ODE for one unit of time from initial position p and initial velocity v, and $\log_p$ inverts it near p. In the resulting coordinates $g(p) = I$ and all first derivatives of g vanish at p, so the metric agrees with the Euclidean one to first order and any deviation is second order. This is the geometric counterpart of a first-order Taylor expansion.

A.3 Pullback Metrics

The construction underlying our entire framework is the pullback metric. Given a smooth map $\phi : \mathcal{M} \to (\mathcal{N}, h)$ from a manifold without a specified metric to a Riemannian manifold, the **pullback metric** $\phi^* h$ on $\mathcal{M}$ is defined pointwise by

$$(\phi^* h)_p(u,v) \;=\; h_{\phi(p)}\big(d\phi_p(u),\, d\phi_p(v)\big), \qquad u, v \in T_p\mathcal{M}. \tag{12}$$

When ϕ is an immersion (i.e. $d\phi_p$ has full column rank everywhere), $\phi^* h$ is positive definite and therefore defines a Riemannian metric on $\mathcal{M}$. When $d\phi_p$ has a nontrivial kernel, $\phi^* h$ is only a positive-semidefinite tensor, and its degenerate directions are exactly those that the map collapses.

Singular metrics. Most properties we wish to use (local sensitivity, geodesic distance, etc.) still hold for singular Riemannian metrics, with the caveat that moving in the direction of the kernel at a given point doesn't change anything about the output. For more information about singular Riemannian geometry in the context of deep learning, see [27].

Intuition. The pullback is the quadratic form obtained by pushing a perturbation through the map and measuring it in the output, $v \mapsto h(d\phi_p(v), d\phi_p(v))$. It is a first-order object: it reports the squared output change caused by an infinitesimal input perturbation. Directions in $\ker d\phi_p$ produce no output change and receive zero length, which is why the pullback is only positive semidefinite unless $d\phi_p$ is injective.

In the Euclidean case, $\mathcal{N} = \mathbb{R}^M$ with the standard Euclidean metric. Writing ϕ in coordinates and using $J = J_\phi(p) \in \mathbb{R}^{M \times d}$ for its Jacobian, the pullback metric is

$$(\phi^* h)_{ij}(p) \;=\; \sum_a \partial_i \phi^a(p)\, \partial_j \phi^a(p) \;=\; \big[J^\top J\big]_{ij}. \tag{13}$$

A perturbation direction v gets length $\sqrt{v^\top J^\top J v} = \|Jv\|_2$ under the pullback. Directions $v \in \ker J$ are collapsed to zero by ϕ and have zero pullback length, and directions aligned with the top right singular vectors of J have the largest pullback length.

Intuition. With a Euclidean output metric the pullback is $J^\top J$, the Gram matrix of the Jacobian, and the length of v is $\|Jv\|_2$. Writing $J = U\Sigma V^\top$, the eigenvectors of $J^\top J$ are the columns of V and its eigenvalues are σ_j^2, so $v^\top J^\top J v = \sum_j \sigma_j^2 \langle v_j, v\rangle^2$. The rest of the paper follows from this decomposition: the top-r terms carry most of the output response, the tail contributes little, and $\kappa_{cap}(r)$ is the fraction they account for.

A.4 Specialization to ViT Probe Maps

In our setting, $\mathcal{M} = \mathbb{R}^{N \times D}$ is the feature space at probe layer ℓ, $\mathcal{N} = \mathbb{R}^M$ is the task-output space, and the smooth map is the probe map $\psi_\ell : \mathcal{M} \to \mathcal{N}$, which composes the remaining ViT layers, the task decoder, and optionally the task metric. The pullback of the Euclidean metric on $\mathbb{R}^M$ through ψ_ℓ is the ViT pullback metric of the main paper, $g(F) = J_{\psi_\ell}(F)^\top J_{\psi_\ell}(F)$. Because M may be much smaller than ND (e.g. $M = 9$ for VGGT camera pose, against $ND \approx 2 \times 10^5$), J_{ψ_ℓ} typically has a nontrivial kernel, and $g(F)$ is positive semidefinite rather than strictly positive. The directions in $\ker J_{\psi_\ell}$ are the feature-space perturbations that leave the task output unchanged to first order, i.e. the directions the decoder ignores. We work on the ambient space $\mathbb{R}^{N \times D}$ with its global chart and make no assumption that the features lie on a lower-dimensional submanifold.

Why a metric and not just a Jacobian. In principle, all of our diagnostics could be phrased directly in terms of J. We work with the pullback $g = J^\top J$ instead because (i) g lives intrinsically on the feature space and is invariant to post-composition of ϕ with isometries of $\mathbb{R}^M$, (ii) its eigendecomposition gives canonical directions and energies, and (iii) it makes the connection to classical Riemannian metric learning explicit, allowing direct comparison with explicit-SPD methods such as those of [9, 10]. The role of the rank-r approximation g_r is then to truncate g to its most informative directions while keeping it a (positive-semidefinite) metric tensor.

Choice of output metric. We have currently formulated everything in terms of the Euclidean output metric $J^\top J$, rather than a generic pullback $J^\top H J$. However, this is not required for our method. For any positive-definite output metric $H = LL^\top$, the H-weighted pullback is $J^\top H J = (LJ)^\top (LJ)$, which is the Euclidean pullback of the decoder post-composed with the fixed linear map L. The diagnostic, SPN, and importance head therefore apply unchanged with J replaced by LJ, at the cost of one extra matrix-vector product per JVP or VJP. For heterogeneous outputs such as camera pose, the principled choice is a block H that respects the geometry of rotations and puts translation and rotation on comparable scales. On the VGGT camera head we compared the Euclidean metric with such an SO(3)-aware, block-scaled H. This H uses the linearized geodesic metric on rotations, removes the non-physical quaternion-norm direction, and rescales translation and rotation to comparable units. Under this new version, tractability remains unchanged ($\kappa_{cap} = 0.99$ under both), per-token importance rankings correlate at Spearman $\rho = 0.995$, and the 50%-prune token sets agree on 97% of tokens. We thus use the Euclidean metric in the rest of the paper, and leave exploration of H to future work.

B Tractability Algorithms

We expand on the two estimators used in Algorithm 1: the Hutchinson trace estimator and Stochastic Lanczos Quadrature (SLQ).

B.1 Hutchinson Trace Estimator

For symmetric $A \in \mathbb{R}^{n\times n}$ accessed only via matrix-vector products, the Hutchinson estimator [28] is

$$\hat{T}_m = \tfrac{1}{m}\sum_{i=1}^{m} z_i^\top A z_i, \qquad z_i \overset{\text{iid}}{\sim} \mathcal{D},\ \mathbb{E}[z_i] = 0,\ \mathbb{E}[z_i z_i^\top] = I.$$

It has been shown that $\mathbb{E}[\hat{T}_m] = \mathrm{Tr}(A)$ and $\mathrm{Var}(\hat{T}_m) = \frac{1}{m}(\mathbb{E}[z^\top A z]^2 - \mathrm{Tr}(A)^2)$. For Rademacher probes ($z_i \in \{\pm 1\}^n$ iid), the variance is $\frac{2}{m}(\|A\|_F^2 - \sum_i A_{ii}^2)$. For SPD matrices (which is our focus), Chernoff-type bounds give exponential concentration: $\Pr[|\hat{T}_m - \mathrm{Tr}(A)| > \varepsilon\,\mathrm{Tr}(A)] \le 2\exp(-cm\varepsilon^2)$ for a constant c depending on the conditioning. We use $m = 100$ Rademacher probes throughout, giving relative error $\lesssim 5\%$ in our experiments.

B.2 Stochastic Lanczos Quadrature (SLQ)

SLQ [29] computes $\mathrm{Tr}\, f(A)$ for analytic f via the identity $\mathrm{Tr}\, f(A) = \sum_i f(\lambda_i) = \sum_i e_i^\top f(A) e_i$, replacing the standard basis with random probes and approximating $f(A)v$ by Lanczos-tridiagonalisation. The convergence rate is exponential in the number of Lanczos steps k and independent of the matrix dimension. We use $k = 30$ Lanczos steps with $m = 100$ probes for the full-spectrum effective rank $r_{\text{eff}}^{\text{SLQ}}$.

B.3 Numerical Verification

At $m = 100$, the per-image standard deviation of $\widehat{\kappa_{cap}(20)}$ is ≤ 0.02 across 50 ImageNet-val images for all configurations in Table 1 (Appendix E gives the full sweep). The truncated effective rank $r_{\text{eff}}^{\text{trunc}}$ stabilizes within 1% at $r = 20$ and $q = 2$ for all proceed-decision configurations.

B.4 Algorithm Pseudocode

Algorithm 1 Hutchinson Tractability Diagnostic (referenced in §4)

Require: Backbone, decoder ϕ, probe layer ℓ, rank r, Hutchinson samples m, power-iteration steps q

Ensure: $\kappa_{cap}(r)$, $r_{\text{eff}}^{\text{trunc}}$, CV, decision

Build probe map ψ_ℓ by composing blocks $\ell{+}1, \ldots, L$ with ϕ (and optionally δ)
Wrap in JACOBIANOPERATOR $\mathcal{J}$ (JVP/VJP interface only)
$S, V \leftarrow \text{RANDOMIZEDSVD}(\mathcal{J}, r, q)$ ▷ $O(rq)$ JVP+VJP pairs
$\widehat{\|J\|_F^2} \leftarrow \frac{1}{m}\sum_{i=1}^{m} z_i^\top \mathcal{J}^\top \mathcal{J} z_i,\ z_i \sim \text{Rademacher}^{ND}$ ▷ $O(m)$ JVP+VJP pairs
$\kappa_{cap}(r) \leftarrow \sum_j S_j^2 / \widehat{\|J\|_F^2}$
$r_{\text{eff}}^{\text{trunc}} \leftarrow \exp\Big(-\sum_j \tilde{p}_j \log \tilde{p}_j\Big),\ \tilde{p}_j = S_j^2 / \sum_k S_k^2$
$\text{CV} \leftarrow \frac{1}{r}\sum_j \text{std}_t(\|V_j[t]\|_2)/\text{mean}_t(\|V_j[t]\|_2)$ **return** $\kappa_{cap}(r)$

C Proofs of Theorems

This appendix collects formal statements and proofs of the four results referenced in the main text: the Captured Sensitivity Ceiling, Convergence of Power Iteration, the Token Pruning Error Bound, and Impossibility of Factored Metrics.

C.1 Theorem 1: Captured Sensitivity Ceiling

Theorem 1 (Captured Sensitivity and Rank-r Approximations)**.** *Let $J = U_J \Sigma V^\top$ with $\sigma_1 \ge \cdots \ge \sigma_p \ge 0$, and define $g_r(F) = \sum_{j=1}^{r} \sigma_j^2 v_j v_j^\top + \varepsilon I$ for $\varepsilon \ge 0$. Among all approximations of the form $\tilde{g} = \tilde{U}\tilde{\Lambda}\tilde{U}^\top + \varepsilon I$ with $\tilde{U} \in \mathbb{R}^{ND\times r}$ orthonormal and $\tilde{\Lambda} \succ 0$ diagonal, g_r minimises $\|g(F) - \tilde{g}\|_F$*

Algorithm 2 RandNLA Training Step (referenced in §5.3)

Require: Probe map ψ_ℓ, SPN g_θ, rank r, oversample $r_0 \geq r$, power-iteration steps q, features F, energy weight α

Draw $\Omega \in \mathbb{R}^{ND \times r_0}$ i.i.d. $\mathcal{N}(0,1)$
for $i = 1, \ldots, q$ **do**
  $\Omega \leftarrow J^\top(J\,\Omega)$ ▷ one VJP then one JVP per column; never form J
  $\Omega \leftarrow \mathrm{QR}(\Omega)$
end for
$\widehat{V}_r \leftarrow$ leading r columns of Ω ▷ aligned with $\mathrm{span}(V_r^*)$
$\widehat{S}_j^2 \leftarrow \|J\,\hat{v}_j\|^2$ for $j = 1, \ldots, r$ ▷ natural energy estimates
$\mathcal{L}_{\text{sub}} \leftarrow 1 - \frac{1}{r}\|U(F;\theta)^\top \widehat{V}_r\|_F^2$ ▷ Grassmannian subspace loss
$\mathcal{L}_{\text{en}} \leftarrow \mathrm{MSE}\big(\log \lambda(F;\theta),\, \log \widehat{S}^2\big)$
Update θ on $\mathcal{L}_{\text{sub}} + \alpha\,\mathcal{L}_{\text{en}}$

in the limit $\varepsilon \to 0$ (for fixed $\varepsilon > 0$ the minimizing energies are $\sigma_j^2 - \varepsilon$). When $\sigma_r > \sigma_{r+1}$, the minimizing subspace $\mathrm{span}(V_r^*)$ is unique. Furthermore, for $v \sim \mathrm{Uniform}(\mathbb{S}^{ND-1})$ and $\varepsilon \to 0$,

$$\frac{\mathbb{E}_v[v^\top g_r(F) v]}{\mathbb{E}_v[v^\top g(F) v]} \;=\; \kappa_{cap}(r) \;:=\; \frac{\sum_{j=1}^r \sigma_j^2}{\|J\|_F^2}. \tag{14}$$

No compression $Pg(F)P$ of g onto an r-dimensional subspace (P a rank-r orthogonal projector) achieves a larger ratio, so $\kappa_{cap}(r)$ is both the quality of g_r and the ceiling for any such rank-r restriction. Both quantities defining $\kappa_{cap}(r)$ can be computed to any accuracy using $\mathcal{O}(m + rq)$ JVP/VJP pairs.

Proof. Frobenius optimality. $g(F) = J^\top J$ is PSD with eigenvalues σ_j^2 and eigenvectors v_j. The Eckart–Young–Mirsky theorem [38, 39] gives that the minimizing PSD rank-r approximation (ignoring εI) is $\sum_{j=1}^r \sigma_j^2 v_j v_j^\top = g_r$. Uniqueness of the subspace follows from $\sigma_r^2 > \sigma_{r+1}^2$.

Sensitivity ratio. For any symmetric A and $v \sim \mathrm{Uniform}(\mathbb{S}^{ND-1})$, isotropy gives $\mathbb{E}_v[v^\top A v] = \mathrm{Tr}(A)/ND$. Therefore $\mathbb{E}_v[v^\top g(F) v] = \|J\|_F^2/ND$ and $\mathbb{E}_v[v^\top g_r(F) v] = (\sum_{j=1}^r \sigma_j^2)/ND + \varepsilon$. Dividing and taking $\varepsilon \to 0$ gives Equation 14. By Ky Fan's maximum principle, $\mathrm{Tr}(Pg(F)P) \leq \sum_{j=1}^r \sigma_j^2$ for every rank-r orthogonal projector P, with equality for the projector onto $\mathrm{span}(V_r^*)$, so no such restriction achieves a higher ratio. (Without this restriction the ratio is unbounded, since scaling any $\tilde{g}$ scales its trace.)

Computability. The denominator $\|J\|_F^2 = \mathrm{Tr}(J^\top J)$ is estimated by Hutchinson's estimator with m Rademacher probes, $\hat{T} = \frac{1}{m}\sum_i z_i^\top (J^\top J) z_i$, with sub-Gaussian concentration error $\mathcal{O}(1/\sqrt{m})$ [28]. The numerator $\sum_{j=1}^r \sigma_j^2$ is obtained from a rank-r randomized SVD with q power-iteration steps [30]. Each round costs r JVPs and r VJPs, for a total of $\mathcal{O}(rq)$ JVP/VJP pairs. The tail beyond rank r contributes the remaining fraction $1 - \kappa_{cap}(r)$ of the trace. □

C.2 Theorem 2: Impossibility of Factored Metrics

Theorem 2 (Impossibility of Factored Metrics)**.** *A metric $g_{\mathcal{F}}$ is* factored *if $v^\top g_{\mathcal{F}}(F) v = \sum_{t=1}^N \varphi_t(F)\|v_t\|^2$ for some measurable $\varphi_t : \mathbb{R}^{N\times D} \to \mathbb{R}_{\geq 0}$. A unit vector $v^* \in \mathbb{R}^{ND}$ is ζ-*delocalized *if $\|v_t^*\|^2 \in [(1-\zeta)/N, (1+\zeta)/N]$ for all t. For any distribution P on $\mathbb{R}^{N\times D}$ with $0 < \mathbb{E}_P[\varphi_t(F)] < \infty$, any ζ-delocalized v^*, and $v_{rand} \sim \mathrm{Uniform}(\mathbb{S}^{ND-1})$ independent of $F \sim P$,*

$$\left| \frac{\mathbb{E}_F[v^{*\top} g_{\mathcal{F}}(F) v^*]}{\mathbb{E}_F[v_{rand}^\top g_{\mathcal{F}}(F) v_{rand}]} - 1 \right| \;\leq\; \zeta. \tag{15}$$

The bound is tight. It also holds when $v^* = v^*(F)$ depends on F (as the singular vectors of $J(F)$ do), provided $v^*(F)$ is ζ-delocalized for every F.

Proof sketch. We compute numerator and denominator separately. Tonelli's theorem (applied to non-negative measurable functions) lets us interchange expectations: $\mathbb{E}[v_{\text{rand}}^\top g_{\mathcal{F}}(F) v_{\text{rand}}] =$

$\mathbb{E}_F\mathbb{E}_{v_{\text{rand}}}[v_{\text{rand}}^\top g_{\mathcal{F}}(F)v_{\text{rand}}]$. By isotropy of the uniform sphere, $\mathbb{E}_{v_{\text{rand}}}[\|v_{\text{rand},t}\|^2] = D/(ND) = 1/N$. Therefore

$$\mathbb{E}_F[v_{\text{rand}}^\top g_{\mathcal{F}}(F)v_{\text{rand}}] = \sum_t \tfrac{1}{N}\mathbb{E}_F[\varphi_t] =: \mu \in (0,\infty).$$

For the numerator, since v^* is fixed (non-random), $\mathbb{E}_F[v^{*\top} g_{\mathcal{F}}(F)v^*] = \sum_t \|v_t^*\|^2\mathbb{E}_F[\varphi_t]$. Writing $\delta_t := N\|v_t^*\|^2 - 1 \in [-\zeta,\zeta]$ with $\sum_t \delta_t = 0$ (since $\|v^*\| = 1$), we get $\mathbb{E}_F[v^{*\top} g_{\mathcal{F}}(F)v^*] = \mu + \frac{1}{N}\sum_t \delta_t\mathbb{E}_F[\varphi_t]$. Bounding $|\frac{1}{N}\sum_t \delta_t\mathbb{E}_F[\varphi_t]| \le \frac{\zeta}{N}\sum_t \mathbb{E}_F[\varphi_t] = \zeta\mu$ and dividing by μ gives Equation 15. Tightness: $N = 2$, $\|v_1^*\|^2 = (1+\zeta)/2$, $\|v_2^*\|^2 = (1-\zeta)/2$, $\varphi_1 \equiv 1$, $\varphi_2 \equiv 0$ saturates the bound. If v^* depends on F, the same computation holds pointwise: $\delta_t(F) \in [-\zeta,\zeta]$ and $\varphi_t \ge 0$ give $|\frac{1}{N}\sum_t \mathbb{E}_F[\delta_t(F)\varphi_t(F)]| \le \zeta\mu$. □

Remark on attention entropy. Heuristically, deeper ViT layers can have increasingly diffuse attention, which pushes the right singular vectors of the probe-map Jacobian toward the perfectly delocalized limit, where each token contributes $\approx 1/N$. A formal argument would chain a first-order Jacobian approximation through the attention weight matrices, a Perron–Frobenius characterization of the dominant attention eigenvector, and a Davis–Kahan bound on the resulting singular-vector delocalization. Each step requires regularity conditions that we do not state, so we use this connection only as motivation, not as a rigorous claim.

C.3 Theorem 3: Convergence of Power Iteration

Theorem 3 (Convergence of Power Iteration)**.** *Assume $\sigma_r > \sigma_{r+1}$. Let $\Omega \in \mathbb{R}^{ND\times r_0}$ be a Gaussian sketch and decompose $\Omega = V_r^*A + V_\perp B$ where $A = (V_r^*)^\top\Omega \in \mathbb{R}^{r\times r_0}$ and $B = V_\perp^\top\Omega \in \mathbb{R}^{(ND-r)\times r_0}$. Let $\widehat{V}_r^{(q)}$ denote the QR-orthonormalised image after q rounds of $M_q = (J^\top J)^q$ applied to Ω. Then with probability $\ge 1-\delta$ over the sketch,*

$$\varepsilon^{(q)} := 1 - \tfrac{1}{r}\big\|(\widehat{V}_r^{(q)})^\top V_r^*\big\|_F^2 \le \frac{\|B\|_F^2}{\sigma_{\min}(A)^2\, r}\left(\frac{\sigma_{r+1}}{\sigma_r}\right)^{2q}, \tag{16}$$

where $\sigma_{\min}(A) \ge c_\delta > 0$ depends only on r, r_0, δ and $\|B\|_F \le C_B = O(\sqrt{ND\cdot r_0})$ each hold with probability $\ge 1-\delta/2$.

Proof sketch. The argument has three steps. *(1) Random sketch structure.* Since V_r^* has orthonormal columns, A and B are independent iid Gaussian matrices. Gaussian concentration bounds $\sigma_{\min}(A)$ from below [40], and $\|B\|_F^2 \sim \chi^2((ND-r)r_0)$. *(2) Power iteration.* Applying $M_q = V_r^*\Sigma_r^{2q}(V_r^*)^\top + V_\perp\Sigma_\perp^{2q}V_\perp^\top$ to Ω gives a signal term in $\operatorname{span}(V_r^*)$ with Frobenius norm $\ge \sigma_r^{2q}\sigma_{\min}(A)\sqrt{r}$ and a tail term in $\operatorname{span}(V_\perp)$ with norm $\le \sigma_{r+1}^{2q}\|B\|_F$. *(3) Davis–Kahan.* The Davis–Kahan theorem for invariant subspaces [41, 42] converts the ratio of the tail and signal Frobenius norms into a bound on the principal angles between $\operatorname{span}(\widehat{V}_r^{(q)})$ and $\operatorname{span}(V_r^*)$, and squaring gives Equation 16. The full proof, including the propagation through QR orthogonalization, follows the structure of Halko, Martinsson, and Tropp [30], adapted to the subspace-alignment metric used here. □

Remark on the constant. The prefactor $C(\Omega) = \|B\|_F^2/(\sigma_{\min}(A)^2\, r)$ grows as $\mathcal{O}(ND\, r_0)$ through $\|B\|_F^2$, and $\sigma_{\min}(A)$ is small unless the sketch is oversampled ($r_0 > r$). At the small q used in practice, C can therefore dominate $(\sigma_{r+1}/\sigma_r)^{2q}$, and the bound may be loose or vacuous. Theorem 3 establishes the geometric rate in q for a fixed sketch, but it does not by itself say that a particular q suffices. We choose q with an offline check, increasing it until the recovered subspace stabilizes (§4.3).

Measured convergence. At $q = 0, 2, 5$, the subspace-alignment error is $0.198, 0.011, 0.001$ for DPT depth (L02), $0.617, 0.254, 0.083$ for DINOv2 CLS, and $0.802, 0.438, 0.214$ for CLIP CLS (both at L10). Two steps suffice where the spectral gap is large, and flatter spectra converge more slowly. This is an empirical observation, not a consequence of a small constant C.

Corollary 1 (Quality of the Trained SPN)**.** *Under the assumptions of Theorem 3, suppose the SPN converges to $U(F;\theta^*) = \widehat{V}_r^{(q)}$ (zero subspace loss) and uses natural energies $\lambda_j = \|J\hat{u}_j\|^2$. Then*

as $\varepsilon \to 0$,

$$\left| \frac{\mathbb{E}_v[v^\top g_\theta(F)v]}{\mathbb{E}_v[v^\top g(F)v]} - \kappa_{cap}(r) \right| \;\le\; \frac{(\sigma_1^2 + \sigma_{r+1}^2)\, r\, \varepsilon^{(q)}}{\|J\|_F^2}. \tag{17}$$

The right-hand side goes to zero as $q \to \infty$*, so the trained SPN's sensitivity ratio converges to the ceiling* $\kappa_{cap}(r)$ *of Theorem 1.*

Corollary 2 (Gradient Signal Ratio)**.** *At convergence (*$\hat{v} \to v_1$*), the gradient signal ratio* $\mathrm{GSR} = \sigma_1^2\, ND/\|J\|_F^2$ *satisfies* $\mathrm{GSR} \ge \kappa_{cap}(r) \cdot ND/r$ *(using* $\sigma_1^2 \ge \frac{1}{r}\sum_{j=1}^{r} \sigma_j^2$*). At* $q = 0$ *(random sketch),* $\mathrm{GSR} \approx 1$.

C.4 Theorem 4: Token Pruning Error Bound

Theorem 4 (Token Pruning Error)**.** *Let* ϕ *denote the decoder as a function of the final-layer tensor. Assume* ϕ *is twice continuously differentiable on* $\mathbb{R}^{N\times D}$ *with* $\|H_\phi(x)\|_{op} \le \beta < \infty$ *uniformly on the segment* $\{(1-t)\hat{F} + tF^L : t \in [0,1]\}$*. Suppose token set* $\mathcal{T}$ *is pruned. For each* $t \in \mathcal{T}$*, let* $\xi_t = F_t^L - F_t^\ell \in \mathbb{R}^D$ *be the feature drift,* $\Delta_t \in \mathbb{R}^D$ *the LLF correction, and* $\rho_t = \xi_t - \Delta_t$ *the residual. Then*

$$\left\|\phi(F^L) - \phi(\hat{F})\right\|_2 \;\le\; \sum_{t\in\mathcal{T}} \left\|J_{:,t}(\hat{F})\right\|_F \|\rho_t\|_2 \;+\; \frac{\beta}{2}\sum_{t\in\mathcal{T}} \|\rho_t\|_2^2, \tag{18}$$

where $J_{:,t}(\hat{F}) \in \mathbb{R}^{M\times D}$ *is the block of the Jacobian of* ϕ *for token* t*, evaluated at the LLF-corrected feature tensor.*

Proof. We expand $\phi(F^L) - \phi(\hat{F})$ around $\hat{F}$ using Taylor's theorem with Lagrange remainder:

$$\phi(F^L) - \phi(\hat{F}) = J_\phi(\hat{F})\mathrm{vec}(F^L - \hat{F}) + R_2, \qquad \|R_2\|_2 \le \tfrac{\beta}{2}\|F^L - \hat{F}\|_F^2.$$

Kept tokens satisfy $F_t^L = \hat{F}_t$ exactly, so only pruned tokens contribute. For $t \in \mathcal{T}$, $F_t^L - \hat{F}_t = \xi_t - \Delta_t = \rho_t$. Flattening row-major gives $\mathrm{vec}(F^L - \hat{F}) = \sum_{t\in\mathcal{T}}(I_D \otimes e_t)\rho_t$, and the Jacobian acts as $J_\phi(\hat{F})(I_D \otimes e_t)\rho_t = J_{:,t}(\hat{F})\rho_t$. The triangle inequality and the matrix–vector bound $\|J_{:,t}\rho_t\|_2 \le \|J_{:,t}\|_F\|\rho_t\|_2$ bound the first-order term, and $\|F^L - \hat{F}\|_F^2 = \sum_t \|\rho_t\|_2^2$ bounds R_2. □

Corollary 3 (Optimal Pruning Criterion)**.** *Assume uniform residuals* $\|\rho_t\|_2 = \bar{\rho}$*. Then the optimal pruning set of size* R *minimises* $\sum_{t\in\mathcal{T}} \|J_{:,t}(\hat{F})\|_F$*, the sum of per-token block norms. These block norms use the decoder Jacobian at* $\hat{F}$*, which is only available after the remaining layers have run. The target* imp^* *of Equation 6 instead keeps the top-*r *part of the block norms of* J_{ψ_ℓ} *at the prune layer, and serves as a computable proxy for them.*

D Token Pruning: Details and Extended Discussion

This appendix gives the full pruning setup summarized in §7, the CLS merge rule, the dense pruning pipeline, and a longer discussion of the results.

Setup and pruning bound. Let $\mathcal{T}$ be the set of $|\mathcal{T}| = R$ tokens pruned at layer ℓ. Each is either frozen (hard prune) or merged into another token (soft merge), and the resulting tensor propagates through layers $\ell{+}1, \ldots, L$ to a modified final tensor $\hat{F}$ in place of F^L. For each $t \in \mathcal{T}$, write $\xi_t = F_t^L - F_t^\ell$ for the *drift* that the frozen feature misses, Δ_t for the Last-Layer Fusion correction (§7.2), and $\rho_t = \xi_t - \Delta_t$ for the uncompensated *residual*. View the decoder ϕ as a function of the final-layer tensor, and let $J_{:,t}(\hat{F}) \in \mathbb{R}^{M\times D}$ be the column block of its Jacobian for token t. The norm $\|J_{:,t}(\hat{F})\|_F$ measures the task sensitivity of token t, and the importance score serves as a proxy for it (§5.4). A Taylor expansion of ϕ around $\hat{F}$ (Theorem 4) gives the bound (9). If residuals are roughly uniform across $\mathcal{T}$, minimizing its first-order term is exactly the optimal pruning criterion, which is to drop the tokens with the smallest importance. The importance head approximates this criterion through the proxy imp^*, and LLF reduces the second term.

CLS merge rule. A CLS decoder uses a single embedding (the CLS token) and is invariant to permutations of the remaining tokens, so tokens can be merged while preserving the information the decoder sees, up to the error introduced by the merge itself. We follow the structure of ToMe [20] but replace its cosine similarity with a task-induced distance. We score every token with the importance head ($s_t = h_\theta(F)_t$, with $s_0 = +\infty$ so that the CLS token is never merged). The ηN lowest-scoring tokens form the source set A, and the rest form the destination set B. For each $a \in A$ we find its nearest neighbour $b \in B$ under

$$d_{\text{task}}(a, b) \;=\; \sqrt{(F_a - F_b)^\top Q_a (F_a - F_b)}, \qquad Q_a \;:=\; \sum_{j=1}^{r} \sigma_j^2 \, v_j^{(a)} \, (v_j^{(a)})^\top \;\in\; \mathbb{R}^{D \times D}, \tag{19}$$

where $v_j^{(a)} \in \mathbb{R}^D$ is the a-th token block of the j-th right singular vector of J. The matrix Q_a is the per-token marginal pullback metric at token a, i.e. the rank-r pullback metric restricted to perturbations of token a alone. We then merge with importance weighting, $F_b \leftarrow (s_a F_a + s_b F_b)/(s_a + s_b)$. This distance measures how far apart two tokens are in the directions the task is most sensitive to, whereas cosine similarity weights all feature directions equally.

In our experiments, the σ_j and v_j in (19) come from the SPN prediction of J, the same computation that produces imp^*. All CLS strategies in our tables differ only in which tokens they merge. *Random* selects them uniformly, *ToMe score* uses ToMe's bipartite redundancy criterion, and *Importance* uses h_θ.

Dense hard pruning and Last-Layer Fusion. Dense decoders such as DPT [24] read features at multiple layers and reassemble them into a 2D map, which soft merging corrupts. We instead *hard-prune* the R lowest-importance tokens at layer ℓ, freeze their features, and re-insert them at their original positions at each DPT layer. The frozen features reflect layer ℓ rather than the layers the decoder uses, and (9) shows that this drift ξ_t is multiplied by the token's importance. **Last-Layer Fusion (LLF)** is a common method to reduce the drift. Before the final ViT block, we re-insert the pruned tokens and run that block on the full N-token sequence, so the pruned tokens attend to the updated kept-token features. LLF adds no parameters and needs no training. It shrinks the residual ρ_t in (9), heuristically from $O(\bar{\xi})$ to $O(\bar{\xi}^2)$. We train the importance head on the single-layer probe map at the prune layer (layer 4, and layer 8 for the second stage of 2-stage schedules), which lies between the L02 and L10 probes of Table 1, both in regime (ii). We then apply it to the full multi-layer decoder, whose DPT layers are at layers 2, 5, 8, and 11, and the signal transfers (Table 2).

Baselines and metric for dense decoders. All dense-decoder strategies use the same hard-prune and LLF pipeline and differ only in which tokens they remove. *Random* removes tokens uniformly at random. *ToMe score* uses ToMe's bipartite matching criterion [20]. Tokens are alternately assigned to two sets, and tokens in the first set are removed in order of their highest cosine similarity to any token in the second set, which is always kept. *Importance* removes the tokens with the lowest $h_\theta(F)_t$. We report the additional scale-invariant log error (SILog, $\times 100$) of the pruned depth prediction relative to the unpruned prediction.

Generalization. These two settings sit at opposite CV extremes. The CLS result extends to another backbone (on CLIP-ViT-B/16, ToMe scoring degrades sharply at $\eta = 0.50$ to 33.21 against 0.76 for importance, Wilcoxon $p < 10^{-50}$) and to a shifted distribution (ImageNet-Sketch [43], where importance keeps a more than $100\times$ advantage at low ratios). The dense result extends to other stage configurations: on NYU-Depth-V2 [44], the 2-stage [4, 8] schedule beats ToMe scoring at every ratio by 25–35% at 224, 336, and 448. Appendices E and G give the full numbers. Appendix H adds frozen-backbone baselines on DINOv2 CLS, namely EViT-style class attention, token norm, and similarity to the CLS token. The importance head is best at every ratio, and EViT is the strongest baseline. The same appendix reports ImageNet top-1 accuracy with a k-NN classifier on the pruned CLS embedding. Top-1 stays between 0.66 and 0.68 for every method and ratio (0.674 unpruned). This shows that pruning is safe for classification, but top-1 cannot separate the methods, which is why we report embedding fidelity.

When does importance-guided pruning win? The two terms of (9) explain part of the pattern. The importance score targets the first term, which is about which tokens to remove. It does nothing about the second term, which depends on how stale the frozen features are. ToMe's criterion picks

tokens by feature redundancy rather than task sensitivity. For CLS decoders it merges them, which keeps their content in the sequence. For dense decoders it only chooses which tokens to hard-prune, but a token that closely resembles a neighbour is plausibly one whose missing update LLF can recover well, so its residual ρ_t tends to be small. This gives the following cases.

- *CLS decoders, all ratios.* The decoder is invariant to permutations of the non-CLS tokens, so soft merging leaves little drift and only selection matters. Importance is best at every ratio on both backbones (Tables 2 and 10).
- *Dense decoders, single stage, ImageNet.* At aggressive ratios many tokens are removed and selection dominates the error. Importance beats ToMe scoring from $\eta = 0.20$ upward and is the best of the three strategies from $\eta = 0.30$ upward (Table 2). At low ratios few tokens are removed, so the selection term is small and the second term dominates. All three strategies pay the staleness cost of hard pruning, and ToMe scoring, which favours tokens that are easy to recover, is competitive or better.
- *Dense decoders, single stage, NYU-Depth-V2.* With heads trained on NYU, the pattern reverses. Importance beats ToMe scoring at $\eta = 0.05$–0.30 (Table 5) and is slightly worse at $\eta = 0.50$ at all three resolutions (Table 4). The account above does not explain this dependence on the dataset.
- *Multi-stage schedules.* Spreading removal across layers shrinks the per-stage residuals. The 2-stage [4, 8] schedule beats ToMe scoring at every ratio and every resolution we tested (Tables 5 and 6). The 4-stage schedule fails at $\eta \geq 0.30$. At these ratios repeated pruning compounds the residuals, and the quadratic term grows faster than the first-order savings.
- *Resolution.* Larger inputs increase N. If this spreads the task-sensitive directions over more tokens, per-token scoring becomes less informative (Theorem 2), which matches the single-stage loss at 448 on ImageNet (Table 4).

One pattern in Table 2 is not explained by this account. On ImageNet, the single-stage DPT error first decreases as the prune ratio grows, for all three strategies, reaching its minimum between $\eta = 0.20$ and $\eta = 0.40$ before rising again (e.g. random: 4.59 at $\eta = 0.05$, 3.16 at $\eta = 0.30$, 3.29 at $\eta = 0.50$). We have verified that this effect is real and not an evaluation artifact, but we do not yet know its cause. It does not appear on NYU-Depth-V2 (Table 5), where the error grows monotonically with the prune ratio.

In practice, we recommend importance-weighted soft merging for CLS-style decoders. For dense decoders, we recommend a 2-stage schedule with LLF, which is the only configuration that beat ToMe scoring at every ratio and resolution we tested (on NYU). Single-stage results depend on the dataset, as described above. The diagnostic gives the regime label needed to make this choice before training. We do not gate LLF dynamically. LLF is unnecessary for CLS soft merging and most useful for dense hard pruning, where drift after pruning is largest.

Cost. We profiled single-stage dense depth pruning on a single A40 at batch size 1. At 224×224 and $\eta = 0.20$, pruning removes 12.1% of the backbone's theoretical FLOPs, but the pruned model runs slightly slower than the unpruned one in wall-clock time ($0.96\times$, and $0.98\times$ with ToMe scoring). With 257 tokens, the attention savings are smaller than the fixed cost of scoring, removing, and re-inserting tokens. At 448×448 the FLOP saving at $\eta = 0.20$ rises to 16.7% and the wall-clock speedup becomes positive ($1.12\times$, and $1.15\times$ with ToMe scoring). The importance head and LLF together cost about 2–3$\%$ more than ToMe scoring. The head ($\sim$310K parameters) runs once per pruning stage, and LLF runs the final ViT block on all N tokens instead of the kept ones. The 2-stage [4, 8] schedule saves 8.6% of FLOPs at $\eta = 0.20$ and runs at $0.88\times$ at 224×224. The ToMe score is cheaper to compute than the head, but on ImageNet at $\eta \geq 0.20$ it gives larger degradation (Table 2).

E Full Sweep Results

Configurations of Table 1. The output dimensions of those probes are $M \approx 5 \times 10^4$ for both depth decoders, 768 for the CLS embedding, 9 for the VGGT camera head, and 192 for VGGT pointcloud, which subsamples 64 points. VGGT's depth head is also DPT-style, so we name each row of that table by its backbone rather than by its decoder type. Table 3 extends the Hutchinson tractability sweep with $r = 20$, $m = 100$ probes, and $q = 2$ power-iteration steps, averaged over

50 ImageNet-val images. We report $\kappa_{cap}(20)$, the truncated effective rank $r_{\text{eff}}^{\text{trunc}}$, the full-spectrum effective rank $r_{\text{eff}}^{\text{SLQ}}$, the tail-richness ratio, and the per-token CV, for Depth-Anything ViT-B/14 single-layer and for DINOv2 CLS at four probe layers. The regime structure of §4.3 holds across the rows. Depth-Anything single-layer sits in regime (ii) ($\kappa_{cap} \geq 96\%$, $r_{\text{eff}}^{\text{SLQ}} \approx 4$, ratio $\approx 1.2\times$). DINOv2 CLS sits in regime (iii), with κ_{cap} falling from 58% at L02 to 33% at L10, possibly because the CLS-token Jacobian spreads across more attention paths at later layers.

VGGT depth: an example of regime (i). VGGT depth at L16 (full) sits in regime (i) of §4.3. Its $\kappa_{cap}(20)$ is 76.2%, and its spectrum at the input to VGGT's heads is rich ($r_{\text{eff}}^{\text{SLQ}} = 282$ at x_{vis}). Two features distinguish it from the other configurations in Table 1. First, unlike DPT, its CV of 0.63 is high rather than near zero. Per-token architectures would therefore be viable if the rank budget were large enough, so the obstacle is rank and not delocalization. Second, the VAE compression of §6 brings the rank down to $r_{\text{eff}}^{\text{SLQ}} \in [30, 53]$, which moves the pipeline into regime (ii) and makes the SPN trainable.

E.1 Multi-layer probes and layer spacing

One might expect multi-layer dense decoders to appear intractable simply because they read from many layers. Our results indicate that the number of layers alone does not explain the observed intractability. We diagnose Depth Anything V2 at its natural multi-layer probe $\mathbf{x}_{\text{dpt}} = [F^2 \mid F^5 \mid F^8 \mid F^{11}] \in \mathbb{R}^{257\times 3072}$, the concatenated DPT hook features at all four layers of the 12-layer ViT-B backbone. With the same number of layers ($K = 4$) as VGGT, the spectral-entropy effective rank is $r_{\text{eff}}^{\text{SLQ}} = 36.0$, close to the single-layer result at L02 ($r_{\text{eff}}^{\text{SLQ}} \in [20, 34]$). The coverage rank satisfies $\kappa_{cap}(50) = 0.900$, so the probe is tractable at rank budget $r = 50$. By contrast, VGGT's span layers $\{4, 11, 17, 23\}$ of a 24-layer model and produce $r_{\text{eff}}^{\text{SLQ}} = 282.6$, $\kappa_{cap}(50) = 0.578$, and $r_{0.90} \gg 100$. We attribute the $282.6/36.0 \approx 7.8\times$ gap to how orthogonal the layers are. The DA V2 layers are only 3 layers apart in a 12-layer model, so their Jacobians are nearly aligned (effective layer multiplicity $K_{\text{eff}} \approx 1.1\times$) and reading several layers costs little extra rank. VGGT's layers span very different feature levels, so their Jacobians are nearly orthogonal and the penalty is the full $K_{\text{eff}} \approx K = 4\times$.

The two probes also differ in spatial structure. At the single-layer L02 probe, DA V2 has $\text{CV} \approx 0.026$, which is fully delocalized, and Theorem 2 rules out per-token scoring. At the multi-layer probe $\mathbf{x}_{\text{dpt}}$, $\text{CV} = 2.381$, which is highly localized. This reversal is not a contradiction. The low CV at L02 arises downstream of the 10 subsequent ViT blocks, whose global self-attention spreads any input perturbation across all tokens before it reaches the decoder layers. The $\mathbf{x}_{\text{dpt}}$ probe bypasses those blocks and so keeps the spatial locality of the depth task, in which object boundaries and foreground–background transitions contribute more to depth sensitivity than the interiors of regions. In the two architectures we measured, intractability at multi-layer probes therefore comes from layer layers that are nearly orthogonal, as in VGGT's deep, widely spaced layers, and not from dense decoding or multi-layer reading as such.

Probe-point effective ranks. The layer we choose to measure the spectrum at can have a big impact. *Tractable (structural).* VGGT camera at L16 single-layer has $r_{\text{eff}}^{\text{SLQ}} \leq 9$ and a 90%-cumulative rank $r_{0.90} \leq 9$, because the camera head has only nine outputs. *Tractable (concentrated).* DA V2 DPT at L02 single-layer has $r_{\text{eff}}^{\text{SLQ}} \in [20, 34]$ and $r_{0.90} \approx 15$ across seeds, consistent with the regime (ii) classification used for the importance head. *Intractable in the original space.* Probing VGGT depth and VGGT pointcloud at the input to VGGT's task heads, x_{vis} (the concatenated tokens of layers 4, 11, 17, 23), gives $r_{\text{eff}}^{\text{SLQ}} = 282$ and 212, with $r_{0.90} \gg 100$ in both cases. These spectra are far too rich for any rank-r SPN we can train. *Tractable after VAE compression.* Routing the same probe through the VAE bottleneck of §6 lowers the rank to $r_{\text{eff}}^{\text{SLQ}} \in [30, 53]$ and $r_{0.90} \approx 40$. Here $r_{0.90}$ is the smallest k at which the cumulative spectrum reaches 90% of $\|J\|_F^2$, i.e. $r_{0.90} = \min\{k : \kappa_{cap}(k) \geq 0.9\}$.

DPT pruning robustness across resolutions and datasets. Table 4 reports single-stage [4] DPT pruning quality at the aggressive ratio $\eta = 0.50$ on ImageNet (IN) and NYU-Depth-V2, at three input resolutions. Importance wins on ImageNet at 224 and 336 and loses to ToMe scoring at 448. On NYU at $\eta = 0.5$, the three strategies are within ~ 1 point of each other, and random selection (not shown) is nominally best at all three resolutions. Single-stage pruning is past its useful range once

Table 3: Full Hutchinson tractability sweep covering Depth-Anything ViT-B/14 (DPT depth, single-layer) at three probe layers, with the depth output downsampled to 8×8, 16×16, or 32×32, and DINOv2 ViT-B/14 CLS at four probe layers. "Ratio" is $r_{\text{eff}}^{\text{SLQ}}/r_{\text{eff}}^{\text{trunc}}$.

Config	$\kappa_{cap}(20)$	$r_{\text{eff}}^{\text{trunc}}$	$r_{\text{eff}}^{\text{SLQ}}$	Ratio	CV
Depth-Anything ViT-B/14 (DPT depth, single-layer, downsampled output)					
DA, L05, 16×16	96.30%	3.54	4.3	1.2×	1.06
DA, L05, 8× 8	96.36%	3.18	4.2	1.3×	1.17
DA, L05, 32×32	96.77%	3.67	4.4	1.2×	0.82
DA, L08, 16×16	97.53%	2.95	3.6	1.2×	1.16
DA, L10, 16×16	97.13%	3.45	3.9	1.1×	1.26
DA, L10, 8× 8	97.59%	3.19	4.2	1.3×	1.44
DA, L10, 32×32	96.21%	3.53	4.2	1.2×	0.98
DINOv2 ViT-B/14 CLS					
DINOv2 CLS, L02	58.29%	46.46	90.8	2.0×	0.83
DINOv2 CLS, L05	51.41%	52.91	122.1	2.3×	1.12
DINOv2 CLS, L08	41.88%	59.58	189.9	3.2×	2.03
DINOv2 CLS, L10	33.36%	63.68	270.2	4.2×	4.06

half the tokens are dropped on dense depth. The 2-stage [4, 8] configuration of Table 5 restores the importance advantage on NYU.

Table 4: Single-stage [4] DPT pruning at $\eta = 0.50$ on ImageNet and NYU-Depth-V2 at three input resolutions. Additional SILog (×100), lower is better. ToMe denotes ToMe-score token selection (§7.2). Heads are retrained for each (dataset, resolution) pair except IN@224, which reuses the head of Table 2. Mean over 3 seeds.

Method	IN@224	IN@336	IN@448	NYU@224	NYU@336	NYU@448
ToMe score	4.09	3.65	**4.70**	6.40	6.84	6.98
Importance	**3.08**	**3.56**	5.11	6.43	6.91	7.38

Dataset-matched multi-stage pruning on NYU. Table 5 reports DPT pruning quality on NYU with importance heads trained directly on NYU rather than transferred from ImageNet, under three progressive-pruning stage configurations. The 2-stage [4, 8] configuration beats ToMe scoring at every ratio, with $\approx$ 25–35% relative reduction. The other two configurations depend on the ratio. Single-stage wins everywhere except $\eta = 0.5$, where it ties ToMe scoring. The 4-stage [1, 4, 7, 10] schedule gives the best absolute numbers at low ratios ($\eta \leq 0.20$) but fails at high ratios ($\eta \geq 0.30$).

Table 5: DPT pruning on NYU-Depth-V2 @ 224 with NYU-trained importance heads. Additional SILog (×100), lower is better. ToMe denotes ToMe-score token selection. Variance ≤ 0.12 across 3 seeds. Bold marks the best per (stages, ratio) cell. The 2-stage [4, 8] configuration also wins at NYU@336 and @448 (App. E.1).

Stages	Method	η=0.05	0.10	0.20	0.30	0.50
[4]	ToMe score	4.12	4.89	5.17	5.32	**6.40**
	Importance	**3.07**	**3.82**	**4.58**	**5.00**	6.43
[4, 8]	ToMe score	3.52	4.44	5.19	5.37	5.58
	Importance	**2.42**	**3.07**	**3.86**	**4.34**	**5.12**
[1, 4, 7, 10]	ToMe score	2.16	2.51	3.08	3.59	**4.90**
	Importance	**1.03**	**1.56**	**2.72**	4.16	6.43

2-stage [4, 8] across NYU resolutions. Table 6 reports the 2-stage [4, 8] configuration at NYU@336 and NYU@448 with heads retrained at the target resolution. Importance beats ToMe at every ratio at

both resolutions, with relative reductions of 25–35% that match the @224 pattern reported in Table 5. Variance ≤ 0.14 across 3 seeds.

Table 6: 2-stage [4, 8] DPT pruning on NYU-Depth-V2 @ 336 and @ 448 with NYU-trained importance heads. Additional SILog ($\times 100$), lower is better. ToMe denotes ToMe-score token selection. Bold marks the best per (resolution, ratio) cell.

Resolution	Method	η=0.05	0.10	0.20	0.30	0.50
NYU@336	ToMe score	3.50	4.33	5.00	5.29	5.99
	Importance	**2.50**	**3.22**	**4.09**	**4.59**	**5.39**
NYU@448	ToMe score	3.72	4.70	5.46	5.58	6.13
	Importance	**2.29**	**2.99**	**3.93**	**4.53**	**5.51**

F Importance Head Ablations

We ablate four design choices of the importance head. Unless noted otherwise, numbers are mean $\pm$ std across 3 seeds and the metric is cosine distance degradation ($\times 100$) vs. the full-pass DINOv2 CLS embedding on ImageNet-val.

E.1 Block norms vs. σ^2-weighted target. On DPT depth at probe layer 4, training the importance head against the σ^2-weighted target of Equation 6 gives Spearman $\rho \approx 0.20$. Training against plain Jacobian block norms $\|J_{:,t}\|_F$ gives only $\rho \approx 0.09$. We run this ablation on DPT because it is the delocalized case (CV ≈ 0.026 at the L02 and L10 probes of Table 1), where Theorem 2 predicts that the block-norm signal is approximately constant across tokens.

E.2 Ranking-loss weight w_{rank}. Setting $w_{\text{rank}} = 0$ removes the ranking term and reduces the loss to log-space regression. Setting $w_{\text{rank}} = 1.0$ doubles the weight relative to the default $w_{\text{rank}} = 0.5$. Pruning quality is nearly the same across the three settings (Table 7). We expect the ranking term to matter more in harder regimes, where the importance head is far from the ceiling.

Table 7: Ablation E.2: ranking-loss weight w_{rank}. DINOv2 CLS L10 @ 224, $n_{\text{train}} = 2000$.

w_{rank}	η=0.05	0.10	0.15	0.20	0.30	0.40	0.50
0.0	0.001	0.007	0.024	0.058	0.255	0.811	2.142
0.5	0.001	0.007	0.024	0.060	0.250	0.803	2.089
1.0	0.001	0.007	0.024	0.058	0.249	0.793	2.171

E.3 Architecture size. We sweep the hidden width H over $\{128, 256, 512\}$ (default 256) and the number of attention heads over $\{2, 4, 8\}$ (default 4). Pruning quality plateaus at the default ($H = 256$, 4 heads, 310K parameters), and larger heads give no measurable improvement. Two heads are slightly better than four at every ratio except $\eta = 0.05$, where they tie. We keep four heads in all other experiments.

Table 8: Ablation E.3: hidden width and attention-head sweep. DINOv2 CLS L10 @ 224, $n_{\text{train}} = 2000$, $w_{\text{rank}} = 0.5$.

Config	η=0.05	0.10	0.15	0.20	0.30	0.40	0.50
$H = 128$	0.001	0.007	0.024	0.060	0.260	0.806	2.124
$H = 256$ (default)	0.001	0.007	0.024	0.060	0.250	0.803	2.089
$H = 512$	0.001	0.007	0.023	0.057	0.244	0.788	2.105
2 heads	0.001	0.006	0.020	0.052	0.226	0.741	2.011
4 heads (default)	0.001	0.007	0.024	0.060	0.250	0.803	2.089
8 heads	0.001	0.007	0.022	0.055	0.245	0.802	2.124

E.4 Training-set size. We vary the number of training images $n_{\text{train}} \in \{500, 1000, 2000, 4000\}$. Pruning quality saturates by $n_{\text{train}} = 2000$ on DINOv2 (the default). On DPT there is no consistent trend: $n_{\text{train}} = 4000$ is best at the two lowest ratios and $n_{\text{train}} = 500$ at the others.

Table 9: Ablation E.4: training-set size. Top: DINOv2 CLS L10 @ 224 (cosine $\times 100$). Bottom: DPT depth L4 @ 224 (SILog $\times 100$). Both blocks use the soft-merge pruning pipeline, not the hard-prune-plus-LLF pipeline of §7.2, so the DPT values are not comparable with those of Table 2. Only the trend across n_{train} is meaningful here.

n_{train}	η=0.05	0.10	0.15	0.20	0.30	0.40	0.50
DINOv2 CLS L10							
500	0.001	0.007	0.025	0.066	0.270	0.838	2.024
1000	0.001	0.008	0.024	0.058	0.246	0.781	2.131
2000	0.001	0.007	0.024	0.060	0.250	0.803	2.089
4000	0.001	0.007	0.025	0.064	0.273	0.872	2.196
DPT depth L4							
500	0.997	1.452	1.787	2.130	2.797	3.421	3.927
1000	1.018	1.472	1.835	2.193	2.860	3.492	4.020
2000	0.992	1.448	1.870	2.244	2.969	3.660	4.243
4000	0.953	1.401	1.815	2.181	2.890	3.566	4.117

G Additional Backbone: CLIP CLS Pruning

We replicate the CLS pruning experiment on CLIP-ViT-B/16 [45] with the same probe layer ($\ell = 10$), the same head architecture and training protocol, and the same evaluation set. Table 10 reports the result. Table 11 reports the per-ratio Wilcoxon paired test of $\Delta = h_\theta - h_{\text{ToMe}}$ across the evaluation set, with bootstrap 95% confidence intervals.

Table 10: CLIP CLS L10 pruning quality, ImageNet-val @ 224, mean $\pm$ std across 3 seeds. Cosine distance degradation $\times 100$. ToMe scoring degrades sharply at $\eta = 0.50$, approaching the random baseline, while the importance head's degradation is $44\times$ smaller.

Method	η=0.05	0.10	0.15	0.20	0.30	0.40	0.50
Random	3.10 ±0.44	7.03 ±1.48	13.31 ±1.30	13.55 ±2.00	23.73 ±2.41	30.10 ±1.09	36.31 ±0.69
ToMe score	0.12 ±0.01	0.17 ±0.04	0.21 ±0.03	0.21 ±0.05	0.33 ±0.04	2.40 ±0.38	33.21 ±1.09
Importance	**0.001** ±0.000	**0.004** ±0.000	**0.012** ±0.001	**0.026** ±0.002	**0.094** ±0.009	**0.29** ±0.01	**0.76** ±0.03

Table 11: Wilcoxon paired test, CLIP CLS @ 224. Δ = Importance $-$ ToMe per evaluation pair, in raw cosine-distance units (not $\times 100$). Negative Δ means importance is better. All ratios are significant at $p < 10^{-4}$, and all except $\eta = 0.30$ and $\eta = 0.40$ at $p < 10^{-50}$.

η	Mean Δ	95% bootstrap CI	Wilcoxon p
0.05	-0.00119	$[-0.00145, -0.00095]$	2.9×10^{-92}
0.10	-0.00170	$[-0.00227, -0.00124]$	9.2×10^{-90}
0.15	-0.00199	$[-0.00268, -0.00143]$	5.8×10^{-76}
0.20	-0.00180	$[-0.00240, -0.00129]$	5.5×10^{-57}
0.30	-0.00232	$[-0.00399, -0.00118]$	5.8×10^{-5}
0.40	-0.02110	$[-0.02970, -0.01339]$	3.2×10^{-9}
0.50	-0.32452	$[-0.33672, -0.31222]$	1.5×10^{-95}

Distribution shift: ImageNet-Sketch. We evaluate the same DINOv2 CLS head on ImageNet-Sketch (3 seeds, cosine $\times 100$). At $\eta \in \{0.05, 0.10, 0.15, 0.20, 0.30, 0.40, 0.50\}$, importance gives $0.001/0.005/0.018/0.048/0.216/0.777/2.169$ and ToMe scoring gives

$0.519/0.662/0.665/0.942/1.072/1.661/2.171$. The importance head keeps a more than $100\times$ advantage at low ratios. The gap closes at $\eta = 0.50$, but neither method is much worse than on in-distribution data.

H Additional Baselines and Classification Accuracy

Frozen-backbone baselines. Methods such as DynamicViT, A-ViT, and Token Cropr fine-tune the backbone together with their scoring mechanism, so they do not apply to a frozen backbone–decoder pair. We therefore compare against scoring rules that work on a frozen model, all inside the same CLS merge pipeline (§7.1). *EViT* scores tokens by the strength of the attention given by the CLS token [23]. *Token norm* uses the ℓ_2 norm of each token's features. *CLS similarity* uses the cosine similarity of each token to the CLS token. Table 12 reports the results on DINOv2 CLS at probe layer 10. The importance head is best at every ratio, or tied with EViT at the three ratios where both round to the same two decimals. EViT is the strongest baseline and comes close at low ratios, but it needs the attention weights, whereas the head uses a single forward pass on the features. Both training-free baselines beat ToMe scoring. This table comes from a separate single-seed run, so its values differ from Table 2, which averages three seeds. The differences are visible for the baselines, for example ToMe scoring reads 0.53 here against 0.67 there at $\eta = 0.05$.

Table 12: Frozen-backbone baselines, DINOv2 CLS L10, ImageNet-val @ 224. CLS cosine-distance degradation ($\times 100$), lower is better. Single seed, rounded to two decimals, so entries that tie at low ratios are not separated here.

Method	η=0.05	0.10	0.15	0.20	0.30	0.40	0.50
Importance	**0.00**	**0.01**	**0.02**	**0.06**	**0.24**	**0.82**	**2.17**
EViT	**0.00**	**0.01**	0.03	**0.06**	0.28	0.88	2.30
Token norm	0.04	0.14	0.25	0.38	0.78	1.45	2.59
CLS similarity	0.02	0.08	0.17	0.42	0.90	1.70	2.91
ToMe score	0.53	0.71	0.97	0.99	1.77	2.28	3.69
Random	0.14	0.36	0.58	0.83	1.29	2.07	3.48

Classification accuracy. To check that embedding fidelity translates into classification accuracy, we classify the pruned DINOv2 CLS embeddings on ImageNet-val with a weighted k-NN classifier ($k = 20$, temperature 0.07, a bank of 20,000 training embeddings). The unpruned model reaches top-1 0.674. For every method and every prune ratio, top-1 stays between 0.66 and 0.68. At $\eta = 0.50$ it is 0.670 for the importance head, 0.668 for ToMe scoring, and 0.664 for random selection. Pruning is therefore safe for classification in this setting. The differences between methods are too small to separate them, because the classifier only needs the CLS token, which is never pruned. Cosine distance to the unpruned embedding is more sensitive, which is why we use it as the main metric.

I Token Importance Visualizations

Figure 2 shows per-token importance maps for the DPT depth task on four ImageNet-val images, comparing the importance head with ToMe cosine scoring. Each heatmap is the per-token score reshaped to the 16×16 patch grid and alpha-blended over the input. White outlines mark the top-25% tokens that pruning would keep. The bottom row shows the depth map predicted by the frozen DPT decoder, so that the reader can see what task the importance signal serves. Each heatmap is annotated with its per-image Spearman ρ against the target imp^*.

Two qualitative observations stand out. First, the importance head's map concentrates on depth discontinuities, such as the curtain on the dark window, the dog's body against the rug, the hairdryer body, and the poodle's outline, and it agrees well with the target ($\rho \in [0.87, 0.95]$). These four images are illustrative. Their per-image ρ is well above the held-out average of ≈ 0.20 (§4.2), so they show where the head does well rather than typical agreement. The pattern is what the σ^2-weighted top-r projection of Equation 6 predicts. DPT is delocalized on average (CV ≈ 0.026 at the L02 and L10 probes of Table 1), but the locally task-salient directions cluster on object boundaries, and the importance head recovers them. Second, ToMe's cosine-similarity heatmaps are nearly uncorrelated

with the target ($\rho \in [-0.14, 0.27]$). They highlight tokens that differ from their neighbours rather than tokens the depth decoder is sensitive to, and so they pick scattered patches in the background and in textured regions instead of the depth-relevant regions.

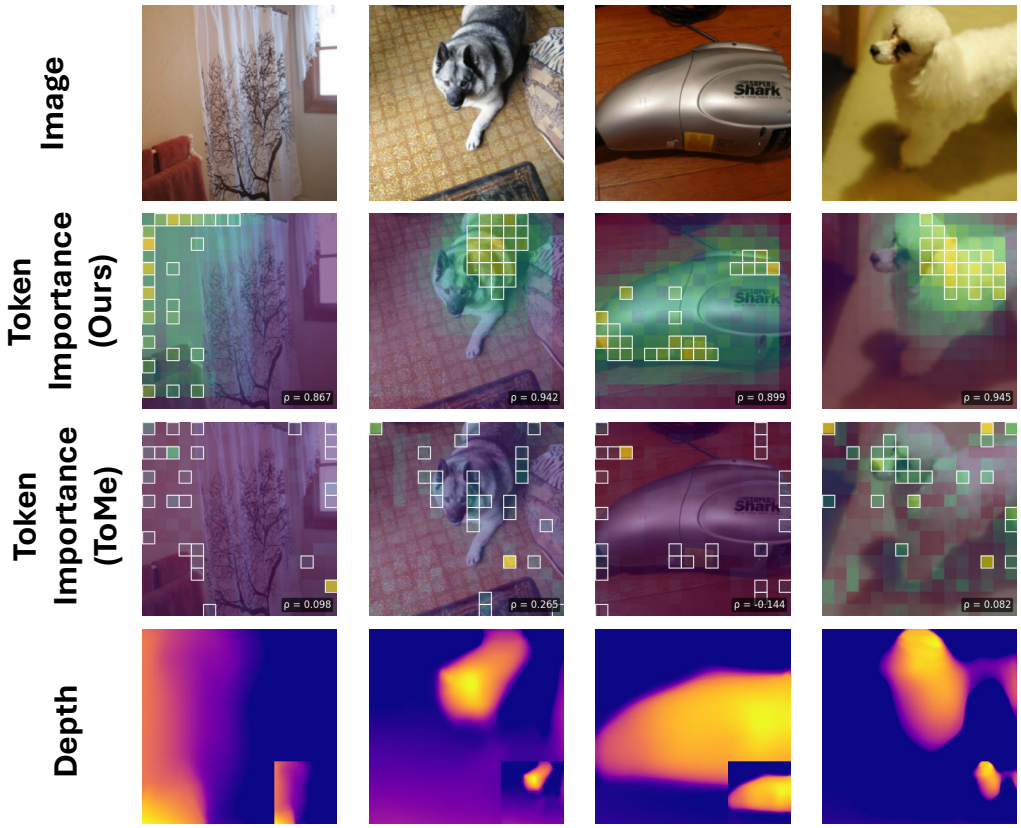

Figure 2: DPT depth token importance on four ImageNet-val images. *Row 1:* input. *Row 2:* importance-head scores, with white outlines on the top-25% tokens and per-image Spearman ρ against the target imp^* in the bottom-right corner. *Row 3:* ToMe cosine-similarity scores, with the same overlay. *Row 4:* depth predicted by the frozen DPT head. The importance head concentrates on depth discontinuities and matches the target at $\rho \in [0.87, 0.95]$ on these examples (the held-out average is ≈ 0.20). ToMe is nearly uncorrelated with the target ($\rho \in [-0.14, 0.27]$).

J SPN VAE Validation: Full Setup and Results

We instantiate the VAE-compression pipeline on VGGT depth, which is in regime (i) without compression (§6). The probe point is the VAE latent $z \in \mathbb{R}^{4096}$ and the probe map is $z \mapsto \phi(G(z))$: the VAE decoder reconstructs the head-input features, which the frozen VGGT depth head uses. The SPN takes VGGT layer-23 features as input and predicts $U \in \mathbb{R}^{4096 \times r}$ in the latent space. Because this Jacobian has only 4096 columns, an exact SVD gives the ground-truth subspace against which we measure V_1. We train the SPN via Algorithm 2 and ablate the role of the RandNLA scheme by comparing three SPN variants with otherwise identical architecture and training data:

- **Random init**, an SPN initialized at random with no supervision (worst-case baseline).
- **RandNLA** $q = 1$, our Algorithm 2 with a single power-iteration step (cheapest meaningful setting).
- **Spectral**, supervision computed at every step from a high-rank deterministic SVD of J. This is the *supervision ceiling* that RandNLA approximates, i.e. the best this SPN reaches when given exact targets.

Table 13 reports the subspace-alignment metric $V_1 = \frac{1}{r}\|U(F;\theta)^\top V_r^*\|_F^2 \in [0,1]$ between the SPN's predicted modes and the true rank-r task-sensitive subspace ($V_1 = 1$ is perfect alignment). Two conclusions follow.

Table 13: SPN training on VAE-compressed VGGT depth. Subspace alignment V_1 between predicted modes and the true rank-r subspace, higher is better. Random initialization has no signal, and RandNLA at the cheapest setting matches the supervision ceiling within noise. The difference between RandNLA and Spectral can be attributed to some noise and smoothing properties of RandNLA.

SPN variant	V_1
Random init	0.005
RandNLA, $q = 1$ (ours)	$\mathbf{0.713}$
Spectral (exact SVD targets)	0.710

First, the gap between random and structured supervision (0.005 vs. 0.713, a $\approx 140\times$ improvement) shows that the SPN learns task geometry and not training-set artifacts. A random initialization does not come close to the true subspace, so the improvement comes from fitting the Jacobian's top-r directions. Second, RandNLA at $q = 1$ matches the supervision ceiling within noise (0.713 vs. 0.710). The ceiling sits at ≈ 0.71 rather than 1 because V_1 is an *amortized* alignment. A single feature-conditioned network must predict the top-r subspace for all images from features alone, and it cannot reproduce every image's SVD exactly. The ≈ 0.29 shortfall is therefore the SPN's approximation error, not a limitation of the SVD or of RandNLA, which is why exact deterministic supervision (Spectral) does no better than cheap randomized supervision (RandNLA, $q = 1$). The saturation at $q = 1$ shows that, for the VAE-compressed Jacobian, a single power-iteration step already resolves the top-r subspace. This is consistent with the geometric rate of Theorem 3, but it is an empirical finding, since the theorem's constant is too loose to predict it.

This experiment supports two claims. (a) The VAE pathway of §6 works in practice. An intractable dense decoder becomes a compressed pipeline in which the SPN learns the rank-r task-sensitive subspace, up to the amortization gap discussed above. (b) The RandNLA training algorithm of §5.3 reaches the supervision ceiling at the cheapest setting, which supports both Algorithm 2 and Theorem 3.

K Task-Induced Retrieval

A second application of the learned geometry is task-sensitive image retrieval. Instead of scoring image similarity by Euclidean or cosine distance, we score it by a task-induced distance derived from the rank-r pullback metric.

Setup. Given a query feature F_q and a gallery $\{F_i\}$ extracted from DINOv2 ViT-B/14 at $\ell = 10$, we rank the gallery by predicted distance to F_q. The reference distance is the depth-decoder distance $\|\phi(F_q) - \phi(F_i)\|$ between the dense outputs (AbsRel-normalised). Galleries are constructed from ImageNet-val (in-distribution) and from NYU-Depth-V2 (cross-dataset, training on ImageNet only). We report Spearman ρ and Kendall τ between predicted and reference ranking, and MAE/RMSE between predicted and reference distance values, averaged over a held-out query set.

Both rank statistics compare two orderings of the same gallery, the one induced by the predicted distances and the one induced by the reference distances. *Spearman* ρ replaces every distance by its rank in its own ordering and takes the Pearson correlation of those ranks, so it is 1 whenever the predicted distances are any increasing function of the reference distances. *Kendall* τ instead counts pairs. For each pair of gallery items, the two orderings either agree (concordant) or disagree (discordant), and τ is the number of concordant pairs minus the number of discordant pairs, divided by the number of pairs. Both equal 1 for identical orderings, 0 for unrelated orderings, and -1 for exactly reversed orderings. Kendall τ is the stricter of the two, since a single swapped pair costs the same wherever it occurs, which is why its values are lower than ρ throughout. Neither sees the scale of the distances, so we also report MAE and RMSE on the distance values themselves.

Distance definition. The learned geometry is a field of seminorms on feature space. For a feature map F and a perturbation $\Delta \in \mathbb{R}^{N\times D}$, write

$$\|\Delta\|_{g_\theta(F)} \;=\; s\,\big\|A(F)\,\Delta\big\|_2, \qquad A(F)\,\Delta \;:=\; W^\top \sum_{t=1}^{N} h_\theta(F)_t\,\Delta_t \;\in\; \mathbb{R}^k, \tag{20}$$

where $s \in \mathbb{R}_{>0}$ is a learned global scale, $W \in \mathbb{R}^{D\times k}$ is a learned low-rank factor, and $h_\theta(F)_t$ is the importance-head score at token t. This is the quadratic form of $g_\theta(F) = s^2 A(F)^\top A(F)$, which is positive semidefinite of rank at most k and varies smoothly with F. The reading is that $\sum_t h_\theta(F)_t \Delta_t$ aggregates the per-token differences with task-dependent weights, and $W^\top$ projects that aggregate onto the rank-k task-sensitive subspace.

A field of seminorms induces a length structure, and the associated distance is

$$d_{g_\theta}(F_1, F_2) \;=\; \inf_{\gamma} \int_0^1 \big\|\dot\gamma(t)\big\|_{g_\theta(\gamma(t))}\,dt, \tag{21}$$

the infimum over piecewise-smooth paths from F_1 to F_2. This d_{g_θ} is a pseudometric. It is non-negative and vanishes on the diagonal, it is symmetric because reversing a path preserves its length, and it satisfies the triangle inequality because concatenating a path from F_1 to F_2 with one from F_2 to F_3 gives a path from F_1 to F_3. It is a pseudometric rather than a metric because g_θ is only positive semidefinite, so two distinct feature maps whose difference lies in $\ker A$ along the connecting path are at distance zero. This is the same degeneracy as in §3, and it is intended: those directions are the ones the decoder ignores.

Computing (21) requires minimizing over paths at dimensionality $ND = 257 \times 768 \approx 2 \times 10^5$, which is prohibitive. We therefore evaluate the straight segment $\gamma(t) = F_1 + t\,\delta$ with $\delta = F_2 - F_1$, and approximate its length by the trapezoidal rule using the two endpoints,

$$d_{\text{sym}}(F_1, F_2) \;=\; \tfrac{1}{2}\Big[\,\|\delta\|_{g_\theta(F_1)} \;+\; \|\delta\|_{g_\theta(F_2)}\,\Big] \;=\; \tfrac{s}{2}\Big[\,\big\|A(F_1)\,\delta\big\|_2 \;+\; \big\|A(F_2)\,\delta\big\|_2\,\Big]. \tag{22}$$

We call this the *Distilled Riemannian Metric* (DRM), and it is what the reported numbers use. Each evaluation costs $\mathcal{O}(NDk)$ and needs only features, with no decoder pass and no matrix inversion.

Two caveats about (22) are worth stating plainly. First, the one-sided quantity $\|\delta\|_{g_\theta(F_1)}$ used inside it is not symmetric, since the seminorm is evaluated at one endpoint, which is why we symmetrize. Second, d_{sym} is a quadrature of the length of one particular path rather than an infimum over paths, so it inherits non-negativity, symmetry and vanishing on the diagonal from (20), but it does not satisfy the triangle inequality in general. It is therefore a symmetric dissimilarity that approximates the pseudometric (21) to first order, and we use it as such. Where the triangle inequality is required, the corresponding object is d_{g_θ}, or the constant-metric special case $\bar g = \mathbb{E}_F[g_\theta(F)]$, for which $\sqrt{\operatorname{vec}(\delta)^\top \bar g \operatorname{vec}(\delta)}$ is a genuine pseudometric because it comes from a fixed seminorm.

Training. Triplets $(F_1, F_2, d_{\text{task}})$ are sampled by drawing image pairs from the training set, extracting layer-ℓ features, and computing $d_{\text{task}} = \|\phi(F_1) - \phi(F_2)\|$ from the frozen depth decoder. The loss combines a regression term on the task distance and a distillation term tying the per-token weights $h_\theta(F)_t$ to the Jacobian-derived target importance imp^* of §5.4:

$$\mathcal{L} \;=\; \mathcal{L}_{\text{reg}}\big(d_{\text{sym}}(F_1, F_2),\, d_{\text{task}}\big) \;+\; \lambda_d\,\mathcal{L}_{\text{distill}}.$$

Without $\mathcal{L}_{\text{distill}}$, the weights h_θ have no anchor to the underlying geometry, and without $\mathcal{L}_{\text{reg}}$, the scale s and projection W are unsupervised. The two losses play different roles. $\mathcal{L}_{\text{reg}}$ calibrates the overall magnitude, and $\mathcal{L}_{\text{distill}}$ shapes the per-token contributions.

Methods compared. Three feature-only distances on the same DINOv2 features. L_2 is $\|F_1 - F_2\|_F$ between layer-ℓ feature tensors. *Cosine* uses CLS-pooled cosine similarity, the standard ViT retrieval baseline. *DRM* is the symmetrised distance of Equation 22. All three operate at query time without re-running the depth decoder.

Results. Table 14 reports the in-distribution evaluation. DRM's predicted ranking matches the reference at Spearman $\rho = 0.783$, against 0.223 for L_2 and 0.206 for cosine, a $3.5\times$ improvement

in correlation. MAE drops from 11.43 (L_2) and 0.677 (cosine) to 0.233. Table 15 reports the cross-dataset case (trained on ImageNet, evaluated on NYU-Depth-V2). DRM keeps its lead on every metric, with $\rho = 0.457$ against 0.338 for cosine and MAE 0.246 against 0.627. The learned task-induced distance has the highest rank correlation in both settings. The gap narrows under distribution shift but does not collapse, which matches the pruning result that geometry trained on ImageNet transfers to NYU at lower but useful quality.

Table 14: In-distribution depth retrieval (train ImageNet, test ImageNet, same-distribution split). Lower is better for MAE/RMSE; higher is better for Spearman/Kendall. Bold: best per column among learned/baseline rows.

Method	Spearman ρ	Kendall τ	MAE	RMSE
L_2 baseline	0.223	0.153	11.43	11.55
Cosine baseline	0.206	0.139	0.677	0.805
DRM	**0.783**	**0.581**	**0.233**	**0.289**
Reference (decoder distance)	1.000	1.000	0.000	0.000

Table 15: Cross-dataset generalization: methods trained on ImageNet, evaluated on NYU-Depth-V2. The learned task-induced distance retains a clear lead over the Euclidean and cosine baselines on every metric.

Method	Spearman ρ	Kendall τ	MAE	RMSE
L_2 baseline	0.266	0.178	7.77	7.90
Cosine baseline	0.338	0.229	0.627	0.707
DRM	**0.457**	**0.312**	**0.246**	**0.310**

L Glossary of Notation

Symbol / term	Meaning
$F^\ell \in \mathbb{R}^{N\times D}$	ViT features at probe layer ℓ; N tokens of dimension D.
ϕ	Frozen task decoder (DPT depth head, CLS embedding, VGGT heads).
ψ_ℓ, M	Probe map from layer-ℓ features to the M-dimensional task output.
J	Jacobian of the probe map's output with respect to F^ℓ, of size $M \times ND$.
$g = J^\top J$	Pullback metric: $v^\top g v = \|Jv\|^2$ is the squared first-order output change under perturbation v. Positive semidefinite.
σ_j, v_j	Singular values and right singular vectors of J, largest first.
r	Number of directions kept in the low-rank metric ($r = 20$ by default).
$\kappa_{cap}(r)$	Fraction of $\|J\|_F^2$ carried by the top-r directions; the tractability diagnostic.
r_{eff}	Entropic effective rank of the squared singular values (*trunc*: top-r only; *SLQ*: full spectrum).
CV	Coefficient of variation of the per-token energy of the v_j (Eq. 4); low CV means delocalized.
m, q, r_0	Hutchinson probes, power-iteration rounds, and sketch width.
imp^*	Jacobian-derived per-token importance target (Eq. 6).
SPN, g_θ	Spectral Pullback Network; predicts the rank-r metric from features (Eq. 5).
Importance head, h_θ	Small feature-only network predicting imp^* per token.
LLF	Last-Layer Fusion; runs the final ViT block on the full token sequence to refresh hard-pruned tokens.
η	Prune ratio: the fraction of tokens removed.
V_1	Subspace alignment $\frac{1}{r}\|U^\top V_r^*\|_F^2$ between predicted and true top-r subspaces.

M Limitations

Scope of the measured geometry. We measure κ_{cap} at probe layers that feed task decoders, and not at internal attention or MLP outputs. Whether those internal layers fall into the same regimes is

open, and answering it would need a different definition of the probe map, but should be doable with the same techniques.

Additionally, the compressed pullback taken through the VAE-reconstructed features rather than the original ones approximates g well only in the directions the VAE reconstructs well. We have no analytic bound on the discrepancy, and the empirical evidence for the pathway is the drop in effective rank plus the SPN result of Appendix J.

Transfer of the learned models. The importance head is trained for one (decoder, probe layer) pair. We have not tested whether a head transfers to a different decoder, so each new task currently needs its own head. The diagnostic itself has no such restriction, since it is computed directly from the Jacobian of whatever pair is of interest.

When geometric pruning helps. The advantage over ToMe scoring depends on the configuration and on the dataset, discussed in §7.3. Three cases are worth repeating here. Single-stage pruning loses at low prune ratios on ImageNet, and at $\eta = 0.50$ on NYU-Depth-V2. Four-stage pruning fails at high ratios, where repeated pruning compounds the residuals. Single-stage pruning is also sensitive to input resolution and loses to ToMe scoring at 448×448 on ImageNet. The two-stage schedule is the one configuration that wins at every ratio and resolution we tested.

Cost at inference. The method targets output quality under pruning, not throughput. At 224×224 the pruned model is slightly slower in wall-clock time than the unpruned one, despite removing 12.1% of the backbone's FLOPs, because the fixed cost of scoring and re-inserting tokens exceeds the attention savings at 257 tokens. A speedup appears only at higher resolution, and even there it is modest: $1.12\times$ at 448×448, against $1.15\times$ for ToMe scoring.

Scope of the evaluation. The evaluation is diagnostic rather than a systems study. Experiments use a single A40 and between 50 and $2{,}000$ images, with ToMe scoring as the primary dense-pruning baseline. The additional frozen-backbone baselines of Appendix H cover DINOv2 CLS only, so we have no comparable baseline sweep on dense decoders.

N Discussion

A common assumption in deep representation learning is that better features lead to better downstream performance. Our results suggest a complementary view. When the features are already as expressive as those of a ViT-B/14 backbone, what matters most is which directions in feature space the downstream task actually uses, and these directions can be recovered from the decoder's Jacobian without retraining the backbone. In this sense, learning which directions matter can be as valuable as learning better features. We believe that this idea can be used to target a whole range of improved methods, not just restricted to tasks such as token pruning and merging, and not just restricted to ViTs.

O Future Work

Several follow-ups build on the importance head and the metric it distills. Because the head supplies a feature-only metric without backpropagating through the decoder, it enables *Riemannian flow matching* under task-induced metrics [46, 47], which raises the question of whether a task-sensitive velocity field produces straighter trajectories than Euclidean flow matching. It also enables *geodesic image interpolation and editing* as a metric-aware analogue of linear feature interpolation [13, 14].

Computing κ_{cap} per token would give a matrix-free local-reliability estimate for dense prediction without retraining. Two further directions use the diagnostic itself. Applying κ_{cap} to LLM feature spaces, with the probe map given by the unembedding composed with the residual stream, could indicate whether attribution methods such as integrated gradients [48] or SmoothGrad [49] are reliable for a given decoder, at a cost of $\mathcal{O}(m + rq)$ Jacobian-vector products per layer. Using $\mathrm{tr}(g(F))$ together with the κ_{cap} defect as out-of-distribution scores would give matrix-free OOD detection that can be benchmarked against energy-based [50] and Mahalanobis [51] baselines. Other directions are cross-task metric transfer, optimal-transport retrieval with d_g as the ground cost [52], and estimating the sectional curvature of (F, g) [11, 15].

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction list four contributions (the $\kappa_{cap}(r)$ diagnostic with three theorems, the Spectral Pullback Network and its RandNLA training, the importance-head distillation, and geometric token pruning), each of which is stated and validated in §4–§7 and the appendices. Quantitative claims in the abstract ($\rho = 0.998$ on DINOv2 CLS, 25% relative AbsRel reduction over ToMe, $V_1 = 0.713$ post-VAE) match the specific tables they summarise.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: §6 flags that the VAE-compressed pullback metric is a controlled approximation of the true metric and that we currently have no analytic bound on the discrepancy. The Conclusion explicitly lists open questions (analytic VAE-compression gap, Riemannian flow-matching, LLM feature spaces), and the regime taxonomy of §4.3 delimits exactly which (backbone, decoder) pairs the method applies to.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Each of the four theorems (Captured Sensitivity Ceiling, Convergence, Pruning Error, Impossibility of Factored Metrics) is stated informally in the body for intuition and given a formal version with its full assumptions and a complete proof in Appendix C (numbered A–D to disambiguate from in-body references). Lemmas, definitions, and externally-cited results (Davis–Kahan, Eckart–Young–Mirsky, Hutchinson trace, SLQ) are referenced where used.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Algorithms 1 and 2 give complete pseudocode for the diagnostic and SPN training. Pretrained backbones and decoders (Depth-Anything ViT-B/14 + DPT, DINOv2 ViT-B/14, CLIP ViT-B/16, VGGT) are publicly available and cited with versions, and Appendix E and the importance-head ablation appendix specify hyperparameters, seed counts, image budgets, ranking-loss weights, prune-layer schedules, and resolution grids for every reported number.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Code is not released at submission time to preserve anonymity; we plan to release the experiment scripts upon acceptance. All datasets used (ImageNet-val, NYU Depth V2) are publicly available and cited.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Each experimental block reports its training-image budget (n_{train}), evaluation budget (n_{eval}), seed count, ratio grid, ranking-loss weight β, and prune-layer schedule; these are summarised in Appendix E and the importance-head ablation appendix. Train/test splits come from the standard ImageNet-val and NYU-Depth-V2 partitions used by prior work.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Pruning experiments report the mean over 3 seeds with the per-cell standard deviation bounded in each table caption (≤ 0.20 in nearly every cell, ≤ 0.14 for the 2-stage NYU runs). For the CLS-pruning headline we additionally report paired Wilcoxon signed-rank p-values against ToMe in Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experiments run on a single A40 GPU. The diagnostic of §4 costs $\mathcal{O}(m+rq)$ Jacobian–vector products per image at $m = 100$, $r = 20$, $q = 2$; SPN training and importance-head distillation use the same per-image cost.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

Justification: The work uses only publicly available pretrained models and standard computer-vision benchmarks; no human subjects, no new data collection, no scraped content, and no deployment context that would raise fairness or privacy concerns. We have read the NeurIPS Code of Ethics and confirm conformance.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The token-pruning application (§7) reduces ViT inference cost, which has a positive impact on the energy footprint of deployed vision systems. The work is foundational and we see no direct path to negative applications: the learned metric is a similarity measure for ViT features, not a generative or decision-making system; it does not produce content, classify people, or interact with sensitive data.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The released artefacts (importance-head and SPN checkpoints) are small auxiliary networks that compute task-importance scores over ViT tokens; they have no generative capability, do not memorise training data, and pose no high-risk-misuse concerns. No new datasets are released.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all the datasets we use, and any papers we got inspiration, ideas, or code from.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not produce any new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: We did not do any experiments with human subjects or crowdsourcing.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: We didn't do anything with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are used solely for polishing the writing of this paper, and did not contribute to the core method development in nonstandard ways.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.