Kernel Mean Embeddings of [CLS] Tokens in ViTs

Mason Faldet[*]
Department of Mathematics
Colorado State University
Fort Collins, CO 80521
mfaldet@colostate.edu

Abstract

We study the geometry of Vision Transformer (ViT) [CLS] representations across layers through the lens of reproducing kernel Hilbert spaces (RKHS). For each layer and class, we estimate class-conditional kernel mean embeddings (KMEs) and measure separability with maximum mean discrepancy (MMD), tuning the kernel and a per-layer PCA projection via a bootstrap-based signal-to-noise (SNR) objective. We further propose a layer-wise confidence signal by evaluating class mean embeddings along a query's [CLS] trajectory. On ImageNet-1k subsets, this exploratory, proof-of-concept analysis indicates that the RKHS framework can capture meaningful geometric and semantic signals in [CLS] representations across ViT layers. We make no SOTA claims; our contribution is a unified framework and practical recipe for probing [CLS] geometry.

1 Introduction

Vision Transformers (ViTs) [6] power a wide range of vision tasks, from image classification to object detection and video processing ([11], [12], [1], [4]). The [CLS] token, a learned summary token prepended to the sequence of patch embeddings, plays a pivotal role in both training and inference of ViTs. Through self-attention at each layer, [CLS] aggregates information from all spatial patch tokens and its final-layer embedding serves as a compact global representation that a lightweight classification head maps to label scores. DINO-pretrained ViTs [3] produce final-layer [CLS] attention maps whose support aligns with the main object, effectively providing unsupervised segmentations. Beyond classification and segmentation, recent work (e.g., Zou et al. [19]) shows that [CLS] also encodes domain cues beneficial for cross-domain few-shot transfer. Taken together, these observations position [CLS] as a semantically structured, task-relevant bottleneck, motivating a deeper analysis of its representation and dynamics across layers and domains.

In this work, we introduce a method to probe how the [CLS] token evolves across layers of a ViT pretrained for image classification. We study: (i) whether the trajectory $\{c_t(z)\}_{t=1}^{L}$, defined in Section 3, yields a *layer-wise confidence signal* via raw similarity scores produced by kernel mean embedding (KME) evaluation, and (ii) whether *class-conditional [CLS] distributions* are separable and semantically structured across layers. Our approach estimates, for each layer and class, the class-conditional distribution of [CLS] states via kernel mean embeddings and uses maximum mean discrepancy (MMD) to quantify separability across layers. By swapping the kernel—cosine for angular alignment and RBF for local geometry—we obtain a holistic picture of similarity. We present the theoretical background, describe our methodology, and report proof-of-concept results on ImageNet-1k. Potential applications in use cases where explainability and confidence are critical are motivating but out of scope.

[*] masonfaldet.com

1st Conference on Topology, Algebra, and Geometry in Data Science (TAG-DS 2025).

Contributions and scope. (1) A unified RKHS perspective on [CLS] trajectories via KME and MMD; (2) a per-layer, per-pair kernel tuning strategy using an SNR objective; (3) a simple confidence signal from evaluating class mean embeddings along query trajectories; (4) a proof-of-concept analysis on ImageNet-1k subsets. To reproduce the results presented in this paper, visit our GitHub repository masonfaldet/CLS-Hidden-Geometry, which contains the exact scripts and configuration files used for all figures. Our experiments intentionally focus on a small number of ViT-style backbones (ViT-Base and DeiT-Base) and a handful of ImageNet-1k class subsets to keep the analysis manageable; extending the framework to diverse architectures, pretraining regimes, and downstream tasks is an important direction for future work.

Related work. We highlight two related efforts that probe the hidden states of the [CLS] token in Vision Transformers. First, Joseph *et al.* [10] apply tools from information geometry to estimate the intrinsic dimension of spatial and [CLS] tokens across layers, and empirically report that the [CLS] token's dimensionality increases as it aggregates information. Second, Vilas *et al.* [17] project intermediate [CLS] states onto the classification head (the class-weight matrix) to score class identifiability at each layer. Our approach differs in that we compare hidden [CLS] states to one another *within the same layer*, and we quantify similarity more broadly, via kernel mean embeddings and MMD, rather than relying solely on cosine similarity.

2 Background

RKHS and Moore–Aronszajn theorem. Let X be a set. A Hilbert space $(\mathcal{H}, \langle\cdot,\, \cdot\rangle_{\mathcal{H}})$ of real-valued functions on X is called a *reproducing kernel Hilbert space* (RKHS) if, for every $x \in X$, the evaluation functional $\mathcal{F}_x : \mathcal{H} \to \mathbb{R}$ given by $\mathcal{F}_x[f] = f(x)$ is bounded. By the Riesz representation theorem, for each $x \in X$ there exists a unique element $K_x \in \mathcal{H}$ such that $\mathcal{F}_x[f] = \langle f,\, K_x\rangle_{\mathcal{H}} = f(x)$ for every $f \in \mathcal{H}$. The function $K : X \times X \to \mathbb{R}$ defined by

$$K(x, y) := \langle K_y,\, K_x\rangle_{\mathcal{H}} = K_y(x) \tag{1}$$

is called the *reproducing kernel* of $\mathcal{H}$. It is symmetric and positive definite. In particular, $K(x, x) = \|K_x\|^2_{\mathcal{H}}$ thus for every $f \in \mathcal{H}$

$$|f(x)| = |\,\langle f,\, K_x\rangle_{\mathcal{H}}\,| \leq \|f\|_{\mathcal{H}}\,\sqrt{K(x, x)}. \tag{2}$$

Conversely, given any symmetric positive-definite $K : X \times X \to \mathbb{R}$, there exists a unique RKHS $\mathcal{H}_K$ whose reproducing kernel is K. The space $\mathcal{H}_K$ is obtained as the completion of the linear span of $\{K(\cdot, x) : x \in X\}$ with inner product

$$\left\langle \sum_i \alpha_i K(\cdot, x_i),\, \sum_j \beta_j K(\cdot, y_j) \right\rangle_{\mathcal{H}_K} = \sum_{i,j} \alpha_i \beta_j \, K(x_i, y_j). \tag{3}$$

Define the feature map $\Phi : X \to \mathcal{H}_K$ by $\Phi(x) = K(\cdot, x)$ so that $K(x, y) = \langle \Phi(x),\, \Phi(y)\rangle_{\mathcal{H}_K}$. This is the "kernel trick," where algorithms expressible via inner products (e.g., SVMs) are carried out implicitly in $\mathcal{H}_K$. This identification is the Moore–Aronszajn theorem [2].

Kernel mean embeddings. Let $(X, \mathcal{A})$ be a measurable space, and let $K : X \times X \to \mathbb{R}$ be a symmetric positive-definite kernel with RKHS $\mathcal{H}_K$. Denote by $\mathcal{M}^+_1(X)$ the set of probability measures on X. For $p \in \mathcal{M}^+_1(X)$ satisfying $\mathbb{E}_{x\sim p}[\sqrt{K(x, x)}] < \infty$, the kernel mean embedding (KME) of p is defined as the Bochner integral

$$\mu_p := \mathbb{E}_{x\sim p}[K(x, \cdot)] \in \mathcal{H}_K. \tag{4}$$

By the reproducing property,

$$\mu_p(x) = \langle \mu_p,\, K(x, \cdot)\rangle_{\mathcal{H}_K} = \mathbb{E}_{x'\sim p}[K(x, x')]\,, \tag{5}$$

$$\langle \mu_p,\, f\rangle_{\mathcal{H}_K} = \mathbb{E}_{x\sim p}[f(x)], \qquad \forall f \in \mathcal{H}_K. \tag{6}$$

Given i.i.d. samples $x_1, \ldots, x_n \sim p$, the empirical estimator of μ_p is given by

$$\widehat{\mu}_p := \frac{1}{n}\sum_{i=1}^{n} K(x_i, \cdot) \in \mathcal{H}_K, \qquad \widehat{\mu}_p(x) = \frac{1}{n}\sum_{i=1}^{n} K(x_i, x). \tag{7}$$

Maximum mean discrepancy (MMD). The maximum mean discrepancy (MMD) metric on $\mathcal{M}_1^+(X)$ is

$$\mathrm{MMD}_K[p,q] := \sup_{\|f\|_{\mathcal{H}_K}\le 1} \big\{\mathbb{E}_{x\sim p}[f(x)] - \mathbb{E}_{x\sim q}[f(x)]\big\} = \|\mu_p - \mu_q\|_{\mathcal{H}_K}. \tag{8}$$

This metric can be expressed in terms of kernel evaluation:

$$\mathrm{MMD}_K^2[p,q] = \mathbb{E}_{x,x'\sim p}[K(x,x')] + \mathbb{E}_{y,y'\sim q}[K(y,y')] - 2\,\mathbb{E}_{x\sim p,\,y\sim q}[K(x,y)]. \tag{9}$$

Given i.i.d. samples $x_1,\dots,x_n \sim p$ and $y_1,\dots,y_m \sim q$, the unbiased U-statistic estimator is

$$\widehat{\mathrm{MMD}}_K^2[p,q] = \frac{1}{n(n-1)}\sum_{i\neq j} K(x_i,x_j) + \frac{1}{m(m-1)}\sum_{i\neq j} K(y_i,y_j) - \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} K(x_i,y_j), \tag{10}$$

which converges in distribution at rate $1/\sqrt{\min(m,n)}$ under $p \neq q$ [8]. MMD belongs to the broader family of integral probability metrics (IPMs) [14].

Vision Transformers (ViTs). We focus on ViTs pretrained for image classification. An input image is partitioned into n fixed-size patches. Each patch is linearly embedded into $\mathbb{R}^d$ and augmented with a positional encoding. A learned `[CLS]` token is prepended, yielding a sequence of length $n{+}1$ processed by L transformer blocks (LayerNorm → multi-head self-attention → MLP, with residual connections). The `[CLS]` token, having repeatedly attended to all patches, serves as a global representation that a classification head maps to logits.

3 Methodology

Fix a ViT M with hidden dimension d_M and L blocks, pretrained on labeled data $\mathcal{D} = \{(x_i, y_i)\}_{i\in\mathcal{I}}$ with $y_i \in [n_{\text{labels}}]$. For an image x, let $c_t(x) \in \mathbb{R}^{d_M}$ denote the hidden state of `[CLS]` after block $t \in [L]$. For each layer t and class y, define

$$P_{t,y} := \text{law of } c_t(X) \ \text{ with } X \sim P(X \mid Y = y), \tag{11}$$

where P denotes the underlying data-generating distribution. We study: (i) separability of $\{P_{t,y}\}$ via $\widehat{\mathrm{MMD}}_K^2$ and (ii) a confidence signal for a novel image z with predicted label $\hat{y} = M(z)$ via a similarity score obtained by evaluating $\widehat{\mu}_{t,\hat{y}}\big(c_t(z)\big)$.

Kernels. We consider cosine and Gaussian RBF mixture kernels:

$$K_{\cos}(x,x') = \Big\langle \tfrac{x}{\|x\|}, \tfrac{x'}{\|x'\|} \Big\rangle, \tag{12}$$

$$K_{\text{mk}}(x,x') = \sum_{i=1}^{n_{\text{scales}}} w_i \exp\Big(-\tfrac{\|x-x'\|^2}{2\sigma_i^2}\Big), \quad w_i \ge 0,\ \sum_i w_i = 1. \tag{13}$$

Because of observed scale drift across layers in ViT-base, bandwidths $\{\sigma_i\}$ are chosen per layer via the median heuristic [7] on the relevant pooled data; mixture weights $\{w_i\}$ are tuned by maximizing the SNR objective below.

PCA. Before computing separability we perform *per-layer* PCA on pooled features. For a retained-variance threshold $\theta \in (0,1]$, we project onto the first r_θ principal components, where

$$r_\theta \;=\; \min\Big\{r \in \{1,\dots,d_M\} :\ \frac{\sum_{i=1}^{r}\lambda_i}{\sum_{i=1}^{d_M}\lambda_i} \ge \theta\Big\} \tag{14}$$

and $\lambda_1 \ge \dots \ge \lambda_{d_M}$ are the PCA eigenvalues. Let $\pi_{y,y'}^{(\theta)} : \mathbb{R}^{d_M} \to \mathbb{R}^{r_\theta}$ denote the PCA projection fitted on $X_{t,y} \cup X_{t,y'}$ (per layer). We construct $\pi_{y,y'}^{(\theta)}$ independently for each layer t, but omit t in the subscript when clear from context. For confidence evaluation, we fit $\pi_y^{(\theta)}$ on $X_{t,y}$. This step is motivated by two considerations: (i) low-variance PCs are often dominated by noise, and (ii) for radial kernels such as RBF, reducing dimensionality mitigates distance concentration and can sharpen separation.

Algorithm 1 Per-layer separability and similarity scoring

```
Require: ViT $M$, labels $Y$, samples $\{X_y\}$, kernels $\mathcal{K}$, PCA thresholds $\Theta$, bootstrap $B$
for layer $t = 1$ to $L$ do
    for each class pair $(y, y')$, $y \neq y'$ do
        Build pooled set $S \leftarrow X_{t,y} \cup X_{t,y'}$
        for $\theta \in \Theta$ do
            Fit PCA on $S$ to get $\pi^{(\theta)}_{y,y'}$
            for $K \in \mathcal{K}$ and its hyperparameters do
                Compute $\widehat{\mathrm{MMD}}^2_K$ on $\pi^{(\theta)}_{y,y'}(S)$            ▷ Eq. 10
                Estimate $\widehat{\mathrm{VAR}}$ via bootstrap ($B$ resamples)
                Record $\widehat{\mathrm{SNR}}(\theta, K)$            ▷ Eq. 15
            end for
        end for
        Select $(\theta^*, K^*) = \arg\max \widehat{\mathrm{SNR}}$; report $\widehat{\mathrm{MMD}}^2$ at $(\theta^*, K^*)$
    end for
    for each class $y$ and query $z$ do
        Fit $\pi^{(\theta)}_y$ on $X_{t,y}$ (for chosen $\theta$)
        Compute similarity score $\widehat{\mu}_{t,y}\big(\pi^{(\theta)}_y(c_t(z))\big)$            ▷ Eq. 7
    end for
end for
```

SNR objective and selection. We tune θ and mixture kernel weights by maximizing a standardized MMD objective [9]:

$$\widehat{\mathrm{SNR}}(\theta, K) \;=\; \frac{\widehat{\mathrm{MMD}}^2_K}{\sqrt{\widehat{\mathrm{VAR}}\left(\widehat{\mathrm{MMD}}^2_K\right)}}, \tag{15}$$

where $\widehat{\mathrm{VAR}}$ denotes the bootstrap variance (within-class resampling; B=100) of $\widehat{\mathrm{MMD}}^2_K$. For each layer and class pair we select $(\theta^*, K^*) = \arg\max \widehat{\mathrm{SNR}}$ and report $\widehat{\mathrm{MMD}}^2$ under (θ^*, K^*). Intuitively, this criterion selects kernel mixtures whose between-class discrepancy (MMD) is large but also stable across bootstrap resamples. It favors scales where separability is reproducible rather than driven by noise.

4 Experiment

Implementation. To view or run the implementation of the experiments in this section, see our GitHub repository masonfaldet/CLS-Hidden-Geometry. The README contains a dedicated section with step-by-step instructions for reproducing all figures reported below.

Model and data. In the main text we use `google/vit-base-patch16-224` [18] pre-trained on ImageNet-1k [5]. We repeat separability and similarity experiments with `facebook/deit-base-patch16-224` [16] and report the corresponding results in the supplementary materials. For a proof-of-concept, we restrict attention to selected class subsets Y of the original label space. For each $y \in Y$, we sample $X_y = \{x^{(i)}_y\}^{n_y}_{i=1}$ from the training split ($n_y \approx 700$). It follows that for $t \in [L]$ the set $X_{t,y} := \{c_t(x^{(i)}_y)\}^{n_y}_{i=1}$ is sampled from the distribution of interest $P_{t,y}$.

Hidden-state extraction. We record c_t as the `[CLS]` vector after block t (post-block LayerNorm), with $L = 12$ for all models considered (ViT-Base and DeiT-Base).

Kernels and PCA grids. We choose PCA thresholds $\Theta = \{0.90, 0.95, 0.97, 0.99\}$. The cosine kernel (Equation 12) has no hyperparameters and is used as is. For the Gaussian mixture kernel K_{mk}, we construct a per-layer bandwidth grid $\{\sigma_i\}$ via the median heuristic on the pooled data, with

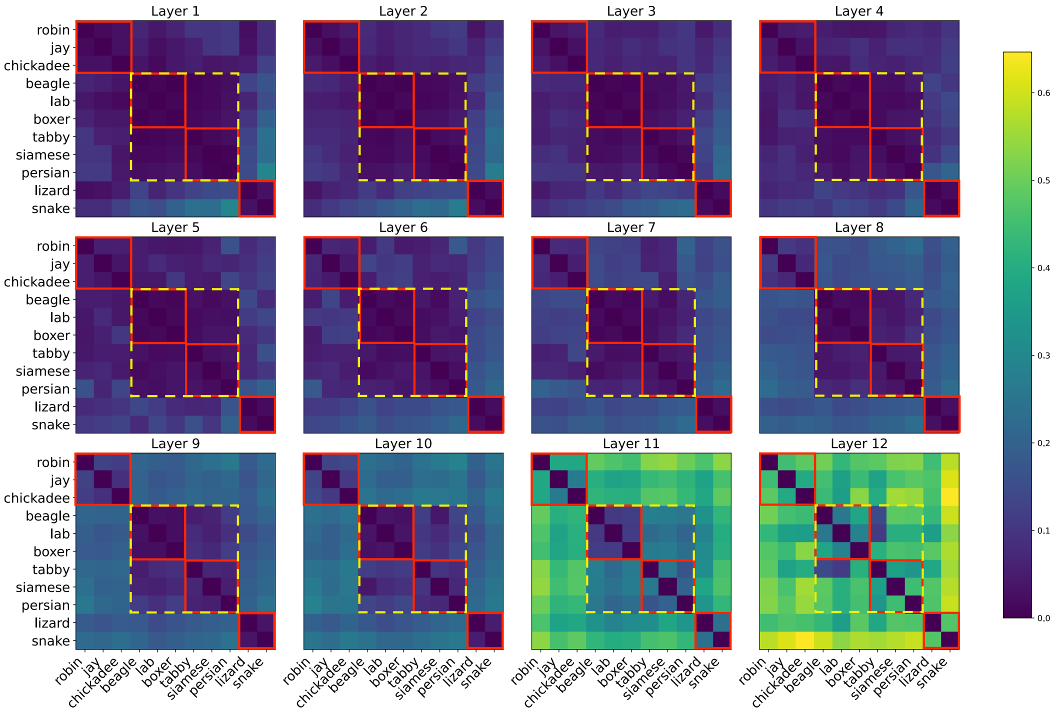


Figure 1: Layer-wise distance matrices $D_t^{K_{\text{mk}}}$ (Eq. 18) displayed as heat maps. Red boxes delineate semantic groups (birds, dogs, cats, reptiles). The yellow dashed box highlights a coarser group of small, furry quadrupeds (cats and dogs).

$n_{\text{scales}} \in \{1, 3, 5\}$. The mixture weights $\mathbf{w} = \{w_i\}_{i=1}^{n_{\text{scales}}}$ are then optimized by maximizing the empirical SNR, which takes a Rayleigh-quotient form:

$$\widehat{\mathrm{SNR}}(\mathbf{w}) \;=\; \frac{\mathbf{w}^\top \boldsymbol{\eta}}{\sqrt{\mathbf{w}^\top \mathbf{Q}\,\mathbf{w}}}, \qquad \nabla_{\mathbf{w}}\widehat{\mathrm{SNR}}(\mathbf{w}) \;=\; \frac{\boldsymbol{\eta}\,(\mathbf{w}^\top \mathbf{Q}\,\mathbf{w}) - (\mathbf{w}^\top \boldsymbol{\eta})\,\mathbf{Q}\mathbf{w}}{(\mathbf{w}^\top \mathbf{Q}\,\mathbf{w})^{3/2}}, \tag{16}$$

where $\eta_i = \widehat{\mathrm{MMD}}^2_{K_{\sigma_i}}$ is the unbiased MMD^2 estimator for scale σ_i, and $\mathbf{Q}$ is the empirical covariance (estimated via bootstrap) of the per-kernel statistics [9]. We perform projected gradient ascent on $\widehat{\mathrm{SNR}}(\mathbf{w})$ with projection onto the probability simplex $\Delta := \left\{\mathbf{w} \in \mathbb{R}^{n_{\text{scales}}}_{\geq 0} : \sum_i w_i = 1\right\}$ at each step:

$$\mathbf{w}^{(t+1)} \;=\; \Pi_\Delta\Big(\mathbf{w}^{(t)} + \alpha\,\nabla_{\mathbf{w}}\widehat{\mathrm{SNR}}(\mathbf{w}^{(t)})\Big)\,, \tag{17}$$

with fixed step size $\alpha > 0$.

Selection and reporting. Per layer, class pair, and kernel type, we select (θ^*, K^*) by $\widehat{\mathrm{SNR}}$ (bootstrap $B = 100$) and report the resulting $\widehat{\mathrm{MMD}}^2_{K^*}$. For queries, we report layer-wise trajectories of $\widehat{\mu}_{t,\hat{y}}\big(c_t(z)\big)$ with class-specific PCA (Algorithm 1).

MMD-based separability For several small subsets Y of ImageNet-1k, we computed pairwise $\widehat{\mathrm{MMD}}^2_K[P_{t,y_i}, P_{t,y_j}]$ across layers t for $y_i, y_j \in Y$. In all cases, we observed clear, detectable separability between the class-conditional `[CLS]` distributions. As expected, the degree of separability correlates with semantic similarity; that is, more closely related classes tend to have smaller MMD than unrelated classes. Due to space constraints, we report one representative experiment.

We set

$$Y = \{\text{robin},\ \text{jay},\ \text{chickadee},\ \text{beagle},\ \text{lab},\ \text{boxer},\ \text{tabby},\ \text{siamese},\ \text{persian},\ \text{lizard},\ \text{snake}\},$$

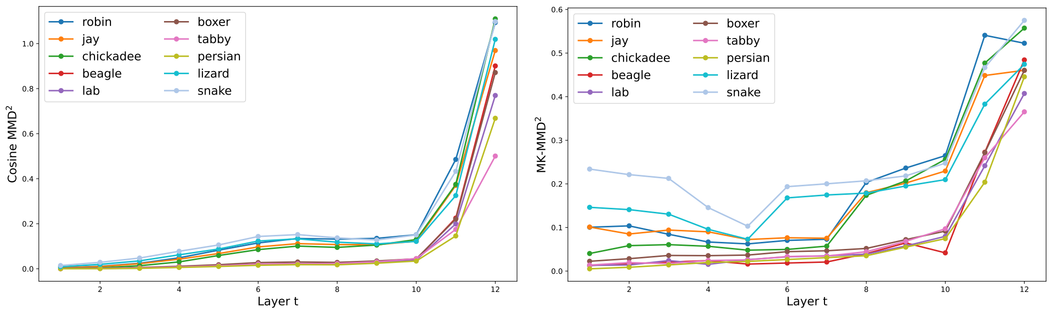


Figure 2: Layer t vs. $\widehat{\text{MMD}}^2_K[P_{t,\text{siamese}}, P_{t,y}]$ for $y \in Y \setminus \{\text{siamese}\}$ with model `google/vit-base-patch16-224`. Left: $K = K_{\cos}$ (trajectories largely entangled until the final layer). Right: $K = K_{\text{mk}}$ (class-specific separations emerge earlier, notably around layers 5 and 7).

using abbreviated ImageNet-1k labels. For each layer t and $K \in \{K_{\cos}, K_{\text{mk}}\}$ we form the distance matrix

$$D^K_t(i,j) \;=\; \widehat{\text{MMD}}^2_K\big[P_{t,y_i},\, P_{t,y_j}\big], \qquad y_i, y_j \in Y, \tag{18}$$

and visualize $\{D^K_t\}$ as heatmaps (Figure 1). Two consistent observations emerge: (i) the overall scale of distances typically increases with depth t, aligning with the intuition that the `[CLS]` token encodes more class-specific structure in later layers [17]; and (ii) when labels are grouped by semantic similarity (e.g., bird breeds, dog breeds, cat breeds, reptiles), dark blocks appear along the diagonal, indicating smaller intra-group distances relative to inter-group distances. We display results for the Gaussian-mixture kernel K_{mk} but note that the same trends are observed with the cosine kernel $K_{\cos}$.

In Figure 2 we plot $\widehat{\text{MMD}}^2_K\big[P_{t,\text{siamese}},\, P_{t,y}\big]$ for $K \in \{K_{\cos},\, K_{\text{mk}}\}$, $y \in Y \setminus \{\text{siamese}\}$, and $t \in [L]$. With the cosine kernel, the trajectories are largely similar until the final layer, where distances become ordered in a manner that reflects semantic groupings. In contrast, with the Gaussian-mixture kernel we observe more distinct, layer-dependent trajectories. Notably, the distances to the `[CLS]` distributions of *snake* and *lizard* increase sharply at layer 5, and those to *robin*, *jay*, and *chickadee* increase at layer 7. This pattern is consistent with a speculative interpretation that the model begins to separate *siamese* from reptiles near layer 5 and from birds near layer 7. We emphasize that these layer attributions are illustrative and should be viewed as preliminary, given the proof-of-concept nature of our study.

With $|Y| \approx 10$, $n_y \approx 700$, and $L = 12$, computing the sequence $\{D^K_t\}^L_{t=1}$ requires roughly 1 hour on a single GPU with 32 GB of memory.

KME-based confidence signal. We demonstrate how KME evaluation can provide a confidence signal for a model's prediction on a novel image. We simulate a case where the model's top-1 prediction is correct but weakly supported by selecting validation images (z, y) for which the predicted label $\hat{y}_1$ matches y and the gap between the top-2 softmax probabilities is small. For a kernel K and class $u \in Y$, define the layer-wise similarity score

$$S^K_{t,u}(z) \;:=\; \widehat{\mu}_{t,u}\big(\pi^{(\theta)}_u(c_t(z))\big), \qquad t \in [L]. \tag{19}$$

For each layer we compute $S^K_{t,\hat{y}_i}(z)$ for $i \in \{1,2\}$ and $K \in \{K_{\cos},\, K_{\text{mk}}\}$. If

$$S^K_{t,\hat{y}_1}(z) \;>\; S^K_{t,\hat{y}_2}(z) \quad \text{for a majority of } t \in [L], \tag{20}$$

then the `[CLS]` trajectory of z is, on average, more similar to the class-$\hat{y}_1$ distribution than to class-$\hat{y}_2$. We interpret this as a layer-wise, representation-level confidence signal supporting the correctness of the top-1 prediction $\hat{y}_1$ despite an ambiguous softmax. For a single summary score per class, we define the aggregate

$$\text{Agg}^K_u(z) \;:=\; \frac{1}{L}\sum_{t=1}^{L} S^K_{t,u}(z). \tag{21}$$

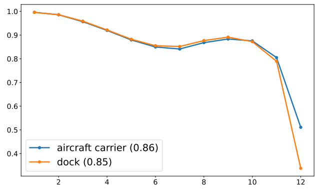
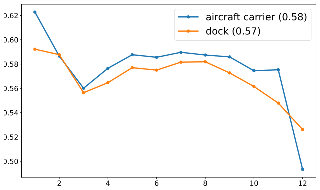
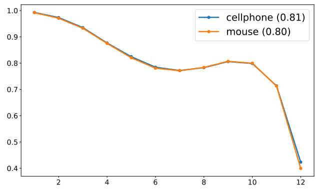
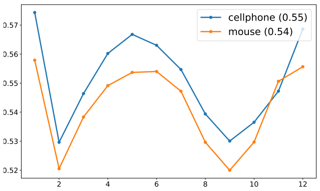
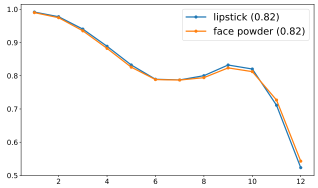
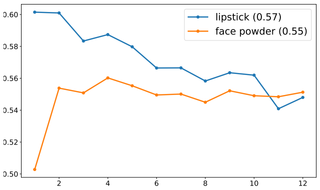
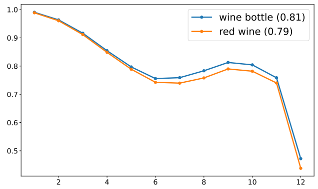
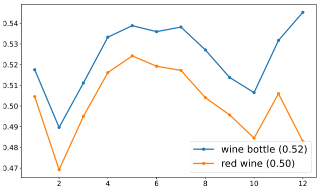
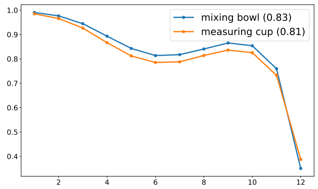
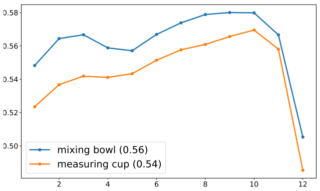
Input Image	Top-3 (prob.)	Cosine trajectories layer t vs $S_{t,y}^{K_{\cos}}(z)$	Mixture-kernel trajectories layer t vs $S_{t,y}^{K_{\mathrm{mk}}}(z)$
	1. aircraft carrier (41%) 2. dock (38%) 3. container ship (5.9%)		
	1. cellphone (44%) 2. mouse (41%) 3. modem (3.7%)		
	1. lipstick (50%) 2. face powder (48%) 3. paintbrush (0.1%)		
	1. wine bottle (43%) 2. red wine (40%) 3. tobacco shop (4.8%)		
	1. mixing bowl (33%) 2. measuring cup (30%) 3. strainer (9.3%)		

Figure 3: Confidence signal examples from the validation split. Each model's top-1 prediction was correct by a small margin. To demonstrate confidence in these predictions, for each image z we compare $c_t(z)$ with the class-conditional [CLS] distributions $P_{t,\hat{y}_i}$ for the top-2 predicted labels $\hat{y}_1$ and $\hat{y}_2$ via the layer-wise KME similarities $S_{t,\hat{y}_i}^{K}(z)$ (Eq. 19). The aggregate similarity $\mathrm{Agg}_{\hat{y}_i}^{K}(z)$ (Eq. 21) is reported in parentheses next to $\hat{y}_i$ in the legends.

We gain confidence in the top-1 selection by testing whether $\mathrm{Agg}_{\hat{y}_1}^{K}(z) > \mathrm{Agg}_{\hat{y}_2}^{K}(z)$.

In Figure 3 we show five weakly supported validation examples (small top-1 vs. top-2 softmax gaps). With the cosine kernel $K_{\cos}$, the trajectories $S_{t,\hat{y}_1}^{K_{\cos}}(z)$ and $S_{t,\hat{y}_2}^{K_{\cos}}(z)$ often exhibit very similar layer-wise patterns and yield a slightly smaller aggregate margin, $\mathrm{Agg}_{\hat{y}_1}^{K_{\cos}}(z) - \mathrm{Agg}_{\hat{y}_2}^{K_{\cos}}(z)$. At present, we do not have a clear explanation for why these patterns are so consistent across images z and labels $\hat{y}$.

With the Gaussian mixture kernel K_{mk}, the two trajectories are more visually distinct. In the first three examples, $S_{t,\hat{y}_1}^{K_{\mathrm{mk}}}(z) > S_{t,\hat{y}_2}^{K_{\mathrm{mk}}}(z)$ holds for all but at most two layers; in the fourth and fifth examples, the inequality holds for every layer t. For both kernels, the aggregate similarity is larger for the true label than for the runner-up, i.e., $\mathrm{Agg}_{\hat{y}_1}^{K}(z) > \mathrm{Agg}_{\hat{y}_2}^{K}(z)$.

In our experiments, KME evaluation was not consistently able to break ties when the softmax gap between the top two classes was $< 2\%$. Note that our estimates $\widehat{\mu}_{t,y}$ were computed from relatively small class samples ($n_y \approx 700$) in a d_M=768 representation, which can yield high-variance kernel averages. Further study is needed to determine when KME evaluation can reliably break ties. Promising directions include increasing n_y, stronger per-layer PCA, and reporting uncertainty (e.g., bootstrap CIs over layers) alongside the layer-wise KME scores.

Additionally, Figure 6 in the supplementary materials reports layer–similarity trajectories for a Siamese-cat image z: we pass z through both Google's ViT-Base and Facebook's DeiT-Base and plot $S^K_{t,y}(z)$ as a function of layer t for labels y spanning a range of semantic similarity to the Siamese class.

With $n_y \approx 700$ and $L = 12$ computing the sequences $\{S^K_{t,y}(z)\}_{t=1}^{L}$ requires roughly 30–60 seconds on a single GPU with 32 GB of memory.

5 Discussion

Our study has several limitations. First, we evaluate only a small family of ViT-style backbones (ViT-Base in the main text and DeiT-Base in the supplementary material) on subsets of ImageNet-1k, so conclusions about other architectures or domains remain speculative. Second, kernel parameters and PCA retained-variance thresholds are selected *per pair and per layer*, which improves descriptive power but may overfit. Third, our confidence signal is a similarity score obtained by evaluating class KMEs along a query trajectory; it is *not* a calibrated probability and should be interpreted as a layer-wise diagnostic rather than a decision-theoretic quantity. Lastly, the computational cost of both the MMD-based separability and KME-based confidence pipelines is dominated by $O\big(L\, C_K(n)\big)$ per class, where L is the number of layers and $C_K(n)$ is the per-kernel cost (naively $O(n^2)$ evaluations for n samples per class, scaled by n_{scales} for K_{mk} and by B for bootstrap variance). Thus, increasing n is the primary bottleneck, but this could be mitigated with approximation methods like mini-batching.

Despite these caveats, our proof-of-concept experiments on both ViT-Base and DeiT-Base indicate that the RKHS perspective has the potential to capture meaningful structure in `[CLS]` representations: class-conditional distributions typically become more separable with depth, and the resulting geometry aligns with semantic groupings in our experiment. Moreover, KME evaluation can leverage upstream `[CLS]` information to provide additional confidence in predictions that are otherwise weakly supported by the softmax. The pipeline itself is architecture-agnostic: any model that exposes per-layer `[CLS]` states can be plugged into the same KME/MMD analysis with no changes. We are optimistic that further development and experimentation within this framework could advance the explainability of transformer models for classification, offering interpretable diagnostics that complement traditional performance metrics.

Acknowledgments This material is based upon work supported by the National Science Foundation under Cooperative Agreement No. 2428052. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation. The author thanks Dr. Michael Kirby and Dr. Chris Peterson for their ongoing support and feedback.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer, 2021. URL `https://arxiv.org/abs/2103.15691`.

[2] Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950. doi: 10.1090/S0002-9947-1950-0051437-7.

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021. URL `https://arxiv.org/abs/2104.14294`.

[4] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition, 2022. URL `https://arxiv.org/abs/2205.13535`.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[6] Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

[7] Arthur Gretton, Olivier Bousquet, Alexander Smola, and Bernhard Schölkopf. Kernel methods for measuring independence. *Journal of Machine Learning Research*, 6:2075–2129, 2005. URL `https://jmlr.org/papers/v6/gretton05a.html`.

[8] Arthur Gretton, Karsten Borgwardt, et al. A kernel two-sample test. *JMLR*, 13:723–773, 2012.

[9] Arthur Gretton, Dino Sejdinovic, Heiko Strathmann, Sivaraman Balakrishnan, Massimiliano Pontil, Kenji Fukumizu, and Bharath K. Sriperumbudur. Optimal kernel choice for large-scale two-sample tests. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL `https://proceedings.neurips.cc/paper_files/paper/2012/file/dbe272bab69f8e13f14b405e038deb64-Paper.pdf`.

[10] Sonia Joseph, Kumar Krishna Agrawal, Arna Ghosh, and Blake A. Richards. On the information geometry of vision transformers. In *NeurIPS 2023 Workshop on Symmetry and Geometry in Neural Representations (NeurReps)*, 2023. URL `https://openreview.net/forum?id=ApeIFsnRvk`.

[11] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection, 2022. URL `https://arxiv.org/abs/2203.16527`.

[12] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, Xiao Wang, Xiaohua Zhai, Thomas Kipf, and Neil Houlsby. Simple open-vocabulary object detection with vision transformers, 2022. URL `https://arxiv.org/abs/2205.06230`.

[13] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends in Machine Learning*, 10(1–2):1–141, 2017. URL `https://arxiv.org/abs/1605.09522`.

[14] Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997. doi: 10.1239/aap/1034625258.

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[16] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers distillation through attention, 2021.

[17] Martina G. Vilas, Timothy Schaumlöffel, and Gemma Roig. Analyzing vision transformers for image classification in class embedding space. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.48550/arXiv.2310.18969. URL `https://proceedings.neurips.cc/paper_files/paper/2023/file/7dd309df03d37643b96f5048b44da798-Paper-Conference.pdf`.

[18] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Zhicheng Yan, Masayoshi Tomizuka, Joseph Gonzalez, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision, 2020.

[19] Yixiong Zou, Shuai Yi, Yuhua Li, and Ruixuan Li. A closer look at the cls token for cross-domain few-shot learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 85523–85545. Curran Associates, Inc., 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/file/9b77f07301b1ef1fe810aae96c12cb7b-Paper-Conference.pdf`.

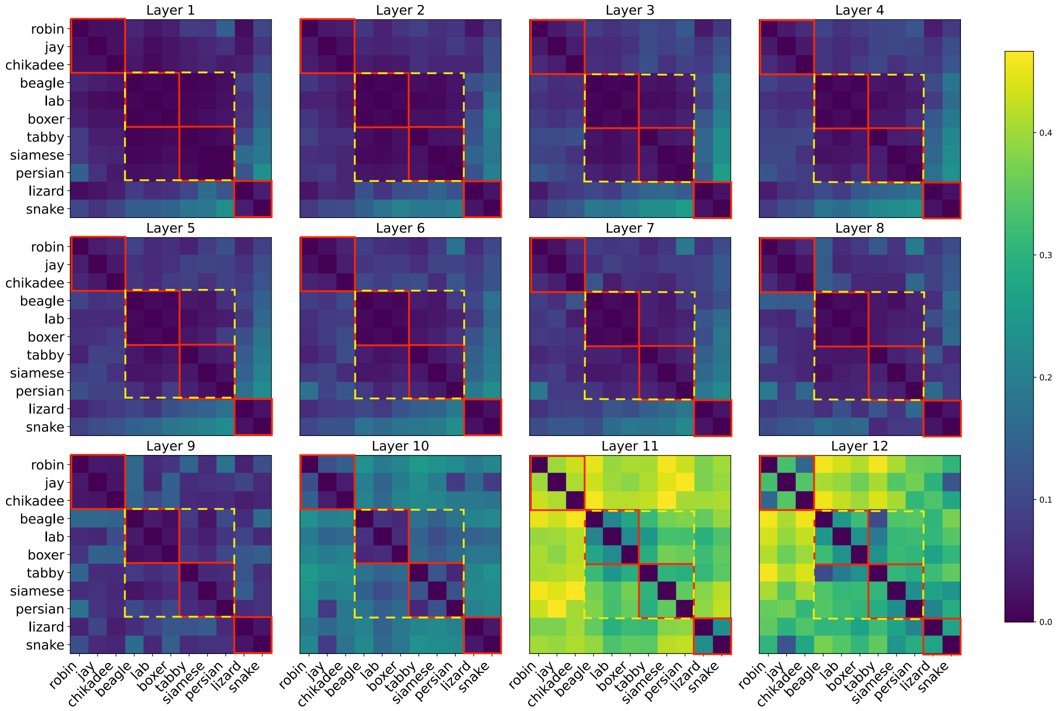


Figure 4: Layer-wise distance matrices $D_t^{K_{\text{mk}}}$ (Eq. 18) displayed as heat maps, with model `facebook/deit-base-patch16-224`. Red boxes delineate semantic groups (birds, dogs, cats, reptiles). The yellow dashed box highlights a coarser group of small, furry quadrupeds (cats and dogs).

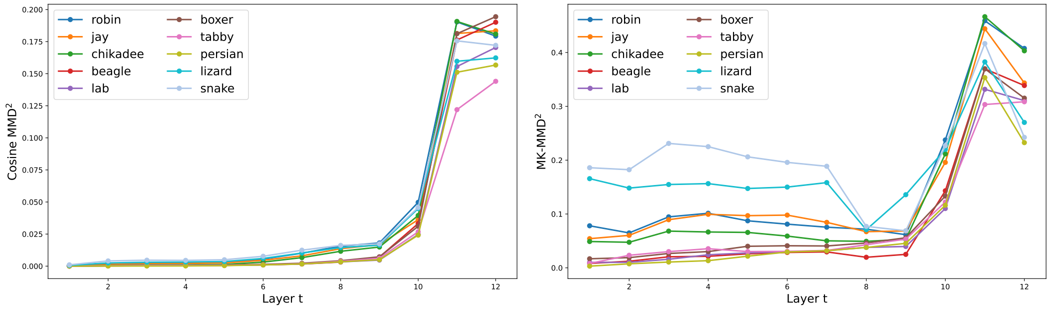


Figure 5: Layer t vs. $\widehat{\mathrm{MMD}}_K^2[P_{t,\text{siamese}}, P_{t,y}]$ for $y \in Y \setminus \{\text{siamese}\}$ with model `facebook/deit-base-patch16-224`. Left: $K = K_{\cos}$ (trajectories largely entangled until the final layer). Right: $K = K_{\text{mk}}$ (class-specific separations emerge earlier, notably around layers 5 and 7).

6 Supplementary materials

Facebook DeiT-Base results. In Figures 4 and 5 we reproduce the heatmaps and layer-wise trajectories from Figures 1 and 2 using Facebook's DeiT-Base model in place of Google's ViT-Base model. DeiT-Base is trained from scratch on ImageNet-1k with a different, data-efficient training recipe, so it provides an independent ViT-style backbone. We observe the same qualitative phenomena as before: (i) the overall scale of MMD^2 distances increases with model depth, and (ii) distances between semantically related classes remain systematically smaller than those between unrelated classes.

Similarity trajectories. Figure 6 shows trajectories of $S^K_{t,y}(z)$ as a function of layer $t \in [12]$ for a Siamese-cat image z taken from the ImageNet-1k validation split and labels

$$Y = \{\text{robin, jay, chickadee, beagle, lab, boxer, tabby, siamese, persian, lizard, snake}\}.$$

Across layers, the Siamese score $S^K_{t,\text{siamese}}(z)$ is already higher than $S^K_{t,y}(z)$ for all $y \in Y \setminus \{\text{siamese}\}$ by layer 2 and remains dominant thereafter. This further supports our claim that upstream `[CLS]` geometry carries a class-specific signal that can be exploited to assess confidence in the model's softmax predictions.

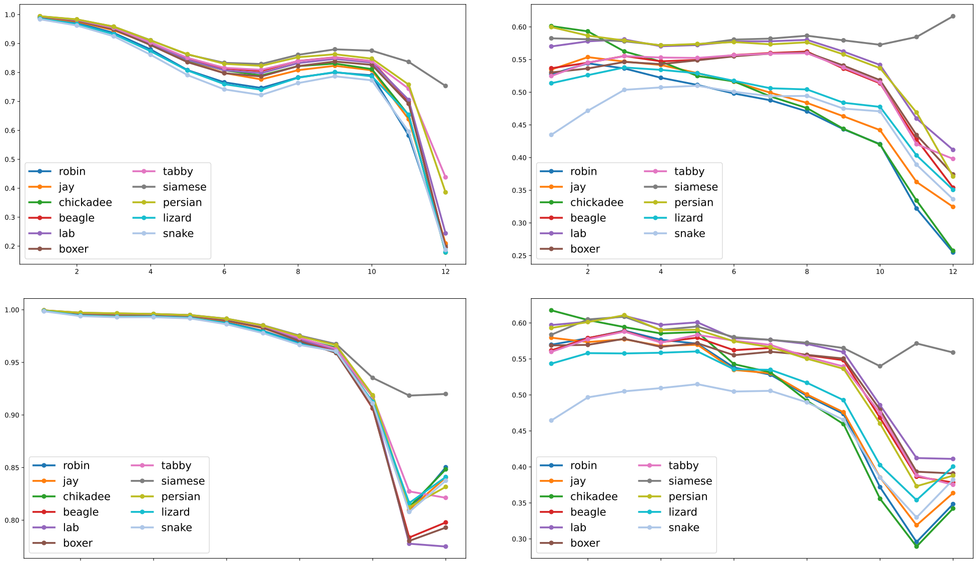


Figure 6: Layer t vs. $S^K_{t,y}(z)$ trajectories for a Siamese-cat image z, labels $y \in Y$, and layers $t \in [12]$. Top row: model = `google/vit-base-patch16-224`. Bottom row: model = `facebook/deit-base-patch16-224`. Left column: $K = K_{\cos}$. Right column: $K = K_{\text{mk}}$.

Google ViT-Base heatmaps. In Figures 7, 8, 9, and 10 we show MMD^2 distance matrices for two additional ImageNet-1k class subsets, computed using the model `google/vit-base-patch16-224`.

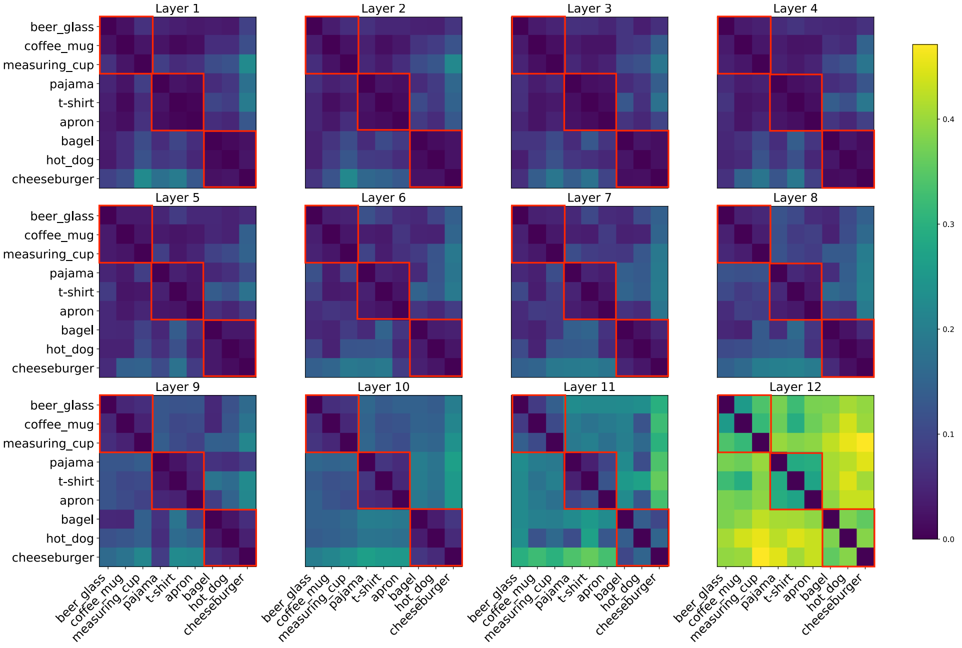


Figure 7: Layer-wise distance matrices $D_t^{K_{\text{mk}}}$ (Eq. 18) using `google/vit-base-patch16-224` model.

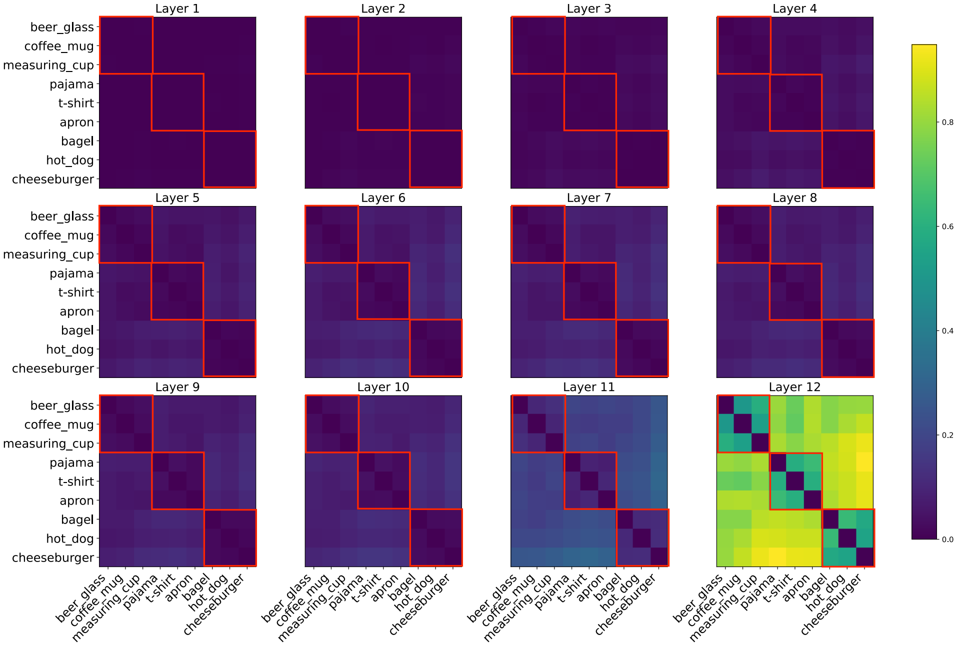


Figure 8: Layer-wise distance matrices $D_t^{K_{\cos}}$ (Eq. 18) using `google/vit-base-patch16-224` model.

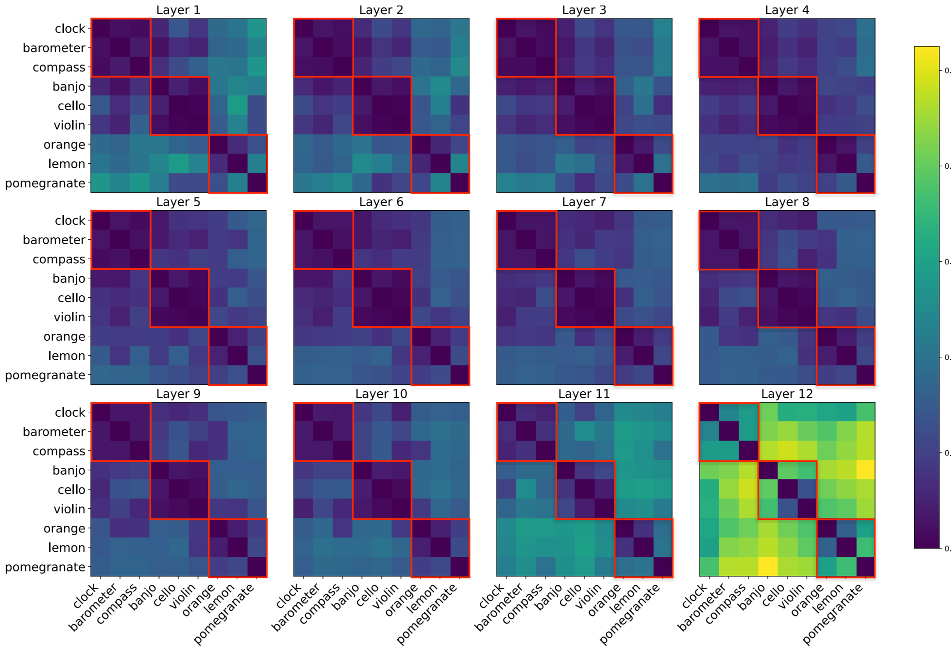


Figure 9: Layer-wise distance matrices $D_t^{K_{\mathrm{mk}}}$ (Eq. 18) using `google/vit-base-patch16-224` model.

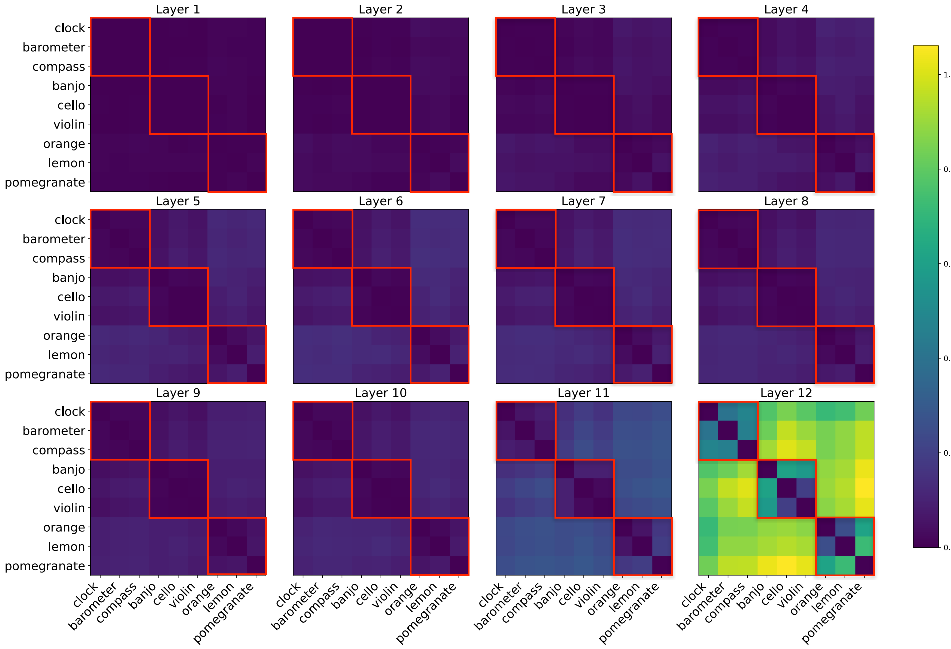


Figure 10: Layer-wise distance matrices $D_t^{K_{\cos}}$ (Eq. 18) using `google/vit-base-patch16-224` model.