SigLino: Efficient Multi-Teacher Distillation for Agglomerative Vision Foundation Models

Supplementary Material

6. Supplementary Results: Small Model Sizes

To explore the trade-off between model size and performance, we evaluate two intermediate-scale models distilled using the same recipe as the main SigLino-0.6B: SigLino-30m (30M parameters) and SigLino-70m (70M parameters). These models validate the scalability of our distillation approach across different parameter budgets and provide useful baselines for compute-constrained settings. Results show that even at significantly reduced model sizes, the multi-teacher distillation recipe maintains strong performance on classification, retrieval, and segmentation tasks (Tables 8, 9, and 10).

7. Analysis of PHI-S Transformation on Registers

We apply PHI-S [31] to evenly distribute the statistical influence of diverse channels and teacher representations. PHI-S operates by rotating the feature space via an invertible transform, composed of PCA whitening and a Hadamard rotation, such that the variance is distributed uniformly across all channels. This normalization assumes that the underlying feature distributions can be reasonably approximated by their first and second-order moments (*i.e.*, Gaussian-like). While this assumption holds for global summary tokens and patch embeddings, we observe that the DINOv3 first register token has a multi-mode distribution. As illustrated in Figure 4, the first register (Row 4) forms distinct, separated clusters. Thus, standard moment estimation captures the statistics between these modes rather than the variance within them. This discrepancy is highlighted by the synthetic data generated from these estimated moments (Column 2), which fails to reproduce the structure of the original data (Column 1) as compared to the zeroth register, global, and patch representations. When PHI-S is applied based on these ill-fitted statistics, it results in a transformed distribution (Column 3) that diverges significantly from the intended standardized target (Column 4). In practice, forcing this transformation on this multi-mode register leads to incorrect scaling and centering, resulting in training instability. Therefore, we exclude registers from the PHI-S normalization pipeline and supervise them in their original space.

8. Impact of Asymmetric Relational Knowledge Distillation (ARKD)

As introduced in the main text (Section 3.2), we propose Asymmetric Relational Knowledge Distillation (ARKD) to enforce pairwise geometric consistency in the student embedding space. Here, we provide an empirical analysis of its effect on training dynamics. Figure 5 visualizes the evolution of both global representation (cosine) losses and relational (ARKD) losses throughout training, comparing a model trained with the full AMoE objective (pink) against a baseline trained without the ARKD term (green).

For SigLIP2 (plots 1 and 3), the global loss and relational loss decrease together even without explicit relational supervision, suggesting that SigLIP2's contrastive objective naturally induces a consistent pairwise structure. However, for DINOv3 (plots 2 and 4), in the baseline experiment (green curve, rightmost plot), the relational error actually fluctuates in both directions as the global cosine loss is optimized. This indicates that DINOv3's pointwise supervision alone is insufficient to preserve the teacher's geometry.

By explicitly optimizing the ARKD objective (pink curve), we force the student to respect these pairwise constraints. The loss trajectory shows that ARKD acts as a regularizer, enforcing relational geometry between samples. This enforced structural alignment directly correlates with the significant improvements observed in zero-shot image-text classification for the DINOv3 head.

9. Positional Encoding Analysis

We investigate the impact of the Rotary Positional Embedding (RoPE) strategy on the student's ability to generalize to unseen high resolutions. Specifically, we compare the standard Axial RoPE against normalizing the input coordinates based on the image aspect ratio (mapping coordinates roughly to $[-1, 1]$) rather than using absolute integer indices. This ensures that the relative frequency distribution remains consistent regardless of the absolute image resolution. Figure 6 demonstrates the generalization capabilities of both methods. We visualize the feature maps of the distilled DINOv3 head across resolutions ranging from the training size (256×256) to an unseen high resolution (2048×2048). With standard Axial RoPE (bottom row), we observe a breakdown in feature coherence at high resolutions: the global structure degrades, and grid-like artifacts appear; the model struggles to extrapolate the axis-aligned frequencies beyond the training distribution. In con-

Model		Image–Text Classification @ 512×512 (Top-1)							kNN Classification @ 512×512 (Top-1)				
Model	Head	IN	C101	Food	Flowers	DTD	Air	Avg	IN	Food	DTD	Air	Avg
SigLino-30m	DINOv3	60.41	85.24	75.22	74.38	58.98	40.76	69.20	76.47	85.24	71.84	70.35	76.23
	SigLIP2	64.22	86.54	78.85	74.88	63.09	48.26	73.76	78.47	87.49	75.74	77.28	80.23
	Ensemble	65.14	87.16	80.34	77.43	62.85	48.26	70.20	78.97	87.49	75.74	77.28	83.73
SigLino-70m	DINOv3	67.91	87.22	82.42	81.34	63.81	56.44	75.50	80.12	89.66	77.34	82.01	82.28
	SigLIP2	71.67	88.66	84.98	84.15	65.66	60.10	78.38	81.73	90.18	77.34	83.30	83.14
	Ensemble	71.20	88.33	85.19	83.98	65.66	60.10	75.74	81.73	90.18	77.34	83.30	86.44

Table 8. Classification performance for small model sizes at 512×512 resolution. SigLino-30m and SigLino-70m demonstrate the effectiveness of our distillation recipe across different parameter budgets.

Model	Head	MSCOCO5k		Flickr30k	
		T2I@1	I2T@1	T2I@1	I2T@1
SigLino-30m	DINOv3	38.90	49.18	65.66	78.40
	SigLIP2	43.38	56.22	71.18	82.80
	Ensemble	46.58	59.72	72.86	82.20
SigLino-70m	DINOv3	43.66	61.10	70.88	85.80
	SigLIP2	47.01	58.30	77.32	90.20
	Ensemble	50.42	65.44	77.54	90.50

Table 9. Retrieval performance (Recall@1) for intermediate model sizes on MSCOCO5k and Flickr30k at 512×512 resolution.

Model	ADE20k ↑	PASCAL-VOC ↑	Parameters
SigLino-30m	43.68	82.43	30M
SigLino-70m	45.33	84.26	70M

Table 10. Segmentation performance (mIoU) for intermediate model sizes on linear probing at 512^2 resolution. These smaller models demonstrate that the distillation recipe scales effectively across different parameter budgets.

trast, the normalized version (top row) exhibits strong scale invariance and good generalization on unseen resolutions. The feature maps at 2048×2048 retain the semantics and smoothness of the low-resolution inputs.

10. Expert Specialization Analysis via Linear CKA

To investigate the semantic specialization of individual experts within the student model, we analyze the similarity between the representations routed to each expert and the hierarchical features of our teacher models (e.g., SigLIP2, DINOv3). We use **Linear Centered Kernel Alignment (CKA)** [20] as our similarity metric, chosen for its invariance to orthogonal transformations and isotropic scaling, making it suitable for comparing representation spaces of differing dimensions.

Experimental Protocol. For a given MoE layer in the student model, we iterate through 1k images. For each expert e, we aggregate the set of token embeddings $\mathbf{X}_e$ that the router assigns to that expert. Simultaneously, we extract the spatially corresponding token embeddings $\mathbf{Y}_{e,l}$ from layer l of the teacher model. This spatial alignment ensures that we compare the student's routed features directly against the teacher's representation of the exact same image patches.

Formulation. Linear CKA measures the similarity between these two sets of representations based on the Frobenius norm of their cross-covariance matrix. Formally, for the collection of N tokens routed to expert e across the entire dataset, we compute:

$$\text{CKA}(\mathbf{X}_e, \mathbf{Y}_{e,l}) = \frac{\|\text{cov}(\mathbf{X}_e, \mathbf{Y}_{e,l})\|_F^2}{\|\text{cov}(\mathbf{X}_e, \mathbf{X}_e)\|_F \|\text{cov}(\mathbf{Y}_{e,l}, \mathbf{Y}_{e,l})\|_F} \tag{9}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and the centered cross-covariance matrix is defined as $\text{cov}(\mathbf{A}, \mathbf{B}) = \mathbf{A}^\top \mathbf{B} - \frac{1}{N}(\sum \mathbf{a}_i)(\sum \mathbf{b}_i)^\top$.

10.1. Analysis of Expert Specialization

Figure 8 visualizes the Linear CKA alignment between the routed inputs of MoE experts at various depths (layers 1, 2, 10, 16) and the hierarchical representations of our teacher models, SigLIP2 and DINOv3. First, we observe a clear layer-wise progression: earlier student layers (e.g., Layers 1 and 2) align primarily with the shallow layers of the teachers, while deeper student layers shift their alignment towards the final teacher representations. This trend is particularly pronounced for SigLIP2, where student experts in early layers focus entirely on the first ≈ 10 teacher layers. This is likely due to the emergence of high-magnitude activations in SigLIP2's deeper layers (potentially from the absence of register tokens).

More importantly, our analysis reveals teacher-specific specialization among experts, validating the choice of the Mixture-of-Experts architecture for multi-teacher distilla-

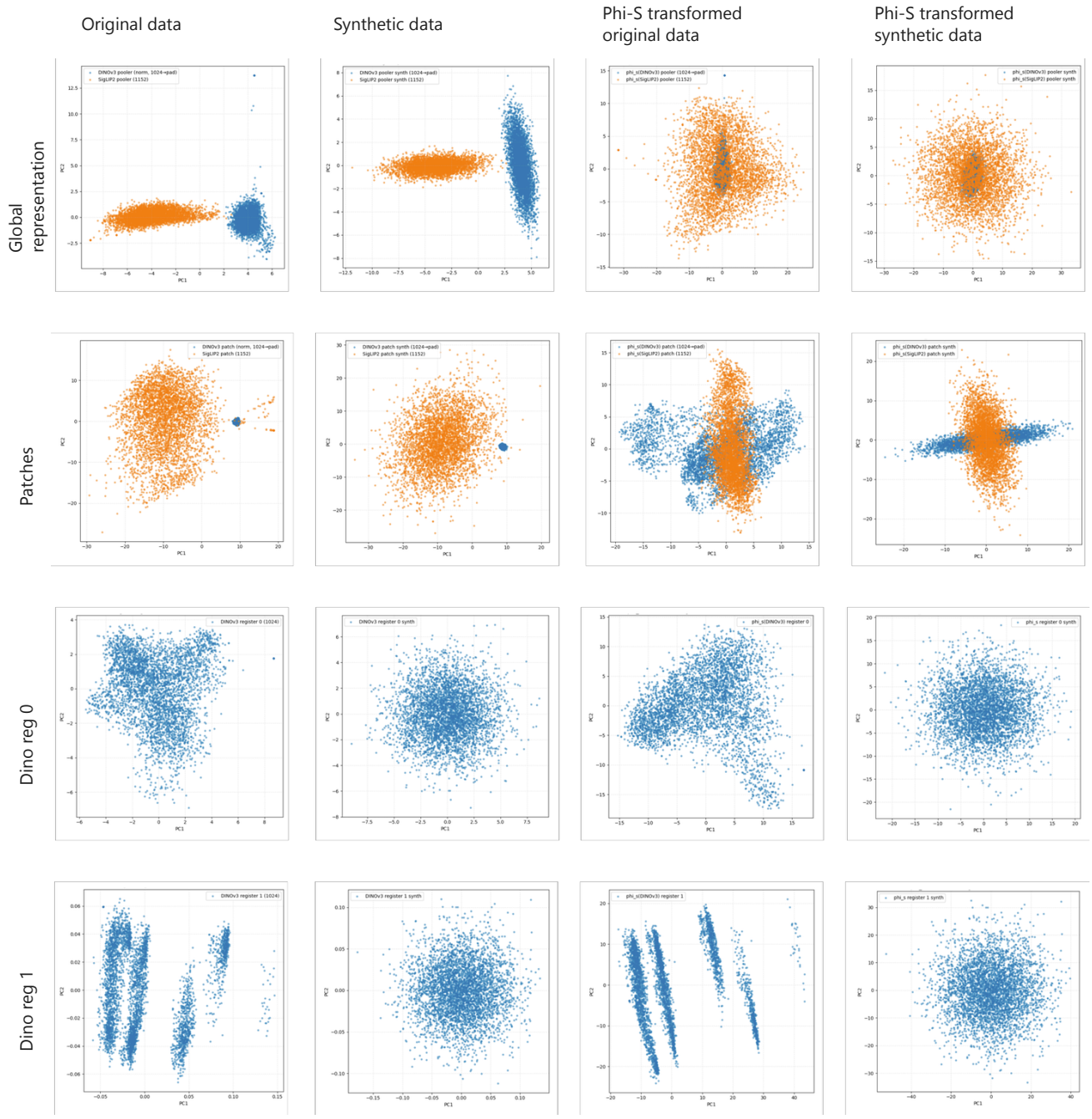

Figure 4. We visualize PCA projections of global features, patches, and DINOv3 registers (0 and 1): original data (Col 1), synthetic Gaussian data generated from estimated moments (Col 2), and their respective versions after Phi-S transformation (Cols 3 and 4). While global, patch embeddings, and the 0th register are well-approximated by Gaussian statistics and effectively whitened by Phi-S, the first register exhibits multi-mode distributions (Row 4) where simple moments capture inter-mode statistics. Hence, applying Phi-S to this register yields incorrect transformations.

tion. In early layers, certain experts specialize exclusively in one teacher's features. For instance, in Layer 1, experts E4 and E22 show strong alignment with DINOv3 but low correlation with SigLIP2, whereas E5 specializes in SigLIP2 features. Similarly, in Layer 2, E5 is highly aligned with SigLIP2 while showing low similarity to DINOv3. We also observe shared experts that maintain alignment with both feature spaces.

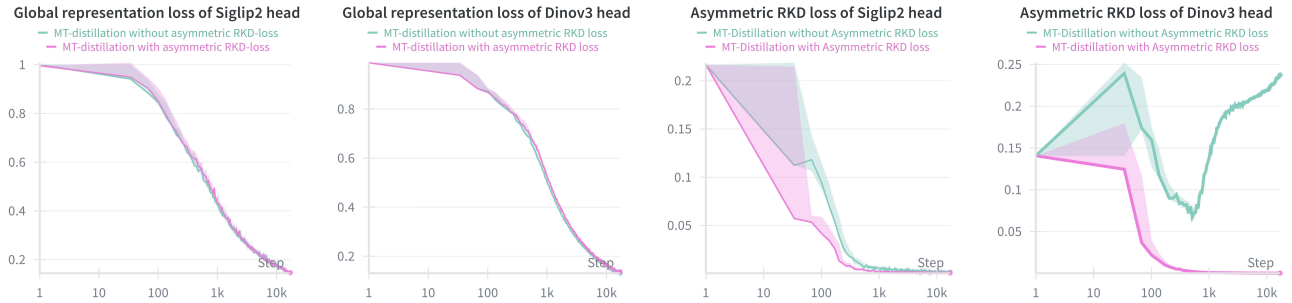

Figure 5. Impact of Asymmetric Relational Knowledge Distillation (ARKD) on training dynamics.

In deeper layers (Layers 10 and 16), the specialization mechanism adapts to handle the high-magnitude activations characteristic of the SigLIP2 teacher. We observe a subset of experts, such as E25 in Layer 10 and E17 in Layer 16, that are strongly aligned with the latest layers of SigLIP2. These experts seem to be responsible for injecting these high-norm features into the student's representation space. Interestingly, other experts in these deep layers initially appear unaligned with SigLIP2. However, when we clip the teacher representations to the range $[-10, 10]$ (third column), we observe some alignments (e.g., experts E25 and E26 in Layer 16). This indicates that while a few experts handle the extreme value distribution, others continue to process the underlying semantic content of the SigLIP2 features, confirming that teacher-specific specialization persists throughout the network depth.

11. Qualitative Analysis of Distilled Representations

We provide a qualitative comparison of the distilled student features against the teacher baselines in Figure 7. This qualitative analysis demonstrates that we successfully learn both teacher representations with high fidelity and that the SigLino patch representations constitute a synthesis of SigLIP2 and DINOv3. The shared SigLino backbone (Column 2) demonstrates nice synergies. While SigLIP2 features often suffer from artifacts harming performance on dense downstream tasks, and DINOv3 lacks inherent image-text alignment, the student's backbone converges on a representation that balances these characteristics. It retains the text-aware features in SigLIP2 with the geometric consistency provided by DINOv3. The resulting feature maps appear to have better object discriminability compared to each teacher individually.

12. Training Implementation Details

We train our 18-layer MoE student model (d=768, 28 experts, top-k=6) on 4 nodes with 8×A100 GPUs each. We use the AdamW optimizer with β_1=0.9, β_2=0.999, and

```python
# 1. Student Architecture (Agglomerative-MoE)
def StudentForward(packed_tokens, packing_mask):
    # Input: Packed sequence of multiple images (Token-Balanced Batching)
    # 1. Prepend CLS + 4 Registers (DINOv3 style) per image in sequence
    x = AddSpecialTokens(packed_tokens, num_regs=4)

    # 2. MoE Backbone with FlexAttention (prevents inter-image attn)
    h_latent = MoETransformer(x, mask=packing_mask)

    # 3. Project features to Teacher Spaces via Learnable Adapters
    # DINOv3: Project all tokens (CLS, Regs, Patches) to 1024-dim
    z_dino = Adapter_DINO(h_latent)

    # SigLIP2: Project to 1152-dim, then apply Frozen Attention Pooling
    # Pooler uses a learned probe query attending only to valid patches
    h_siglip = Adapter_SigLIP(h_latent)
    z_sig_summ = FrozenSigLIPPooler(h_siglip, query=Probe, mask=packing_mask)
    z_sig_patch = h_siglip[patches_only]

    return { "dino": z_dino, "siglip": (z_sig_summ, z_sig_patch) }
```

Listing 1. SigLino forward pseudo-code

ϵ=10^{-15}. The learning rate follows a linear decay schedule from 10^{-3} to 10^{-4} after a 500-step warmup, with weight decay set to 0.02. We summarize the pseudo-code of the distillation pipeline in Listings 11 and 15. The algorithm outlines the Agglomerative-MoE student forward pass, detailing how shared backbone features are projected into distinct DINOv3 and SigLIP2 embedding spaces via teacher-specific adapters and pooling mechanisms. It also formalizes the calculation of our multi-objective loss, explicitly showing how dense feature alignment is normalized by per-image token counts and combined with the global Asym-

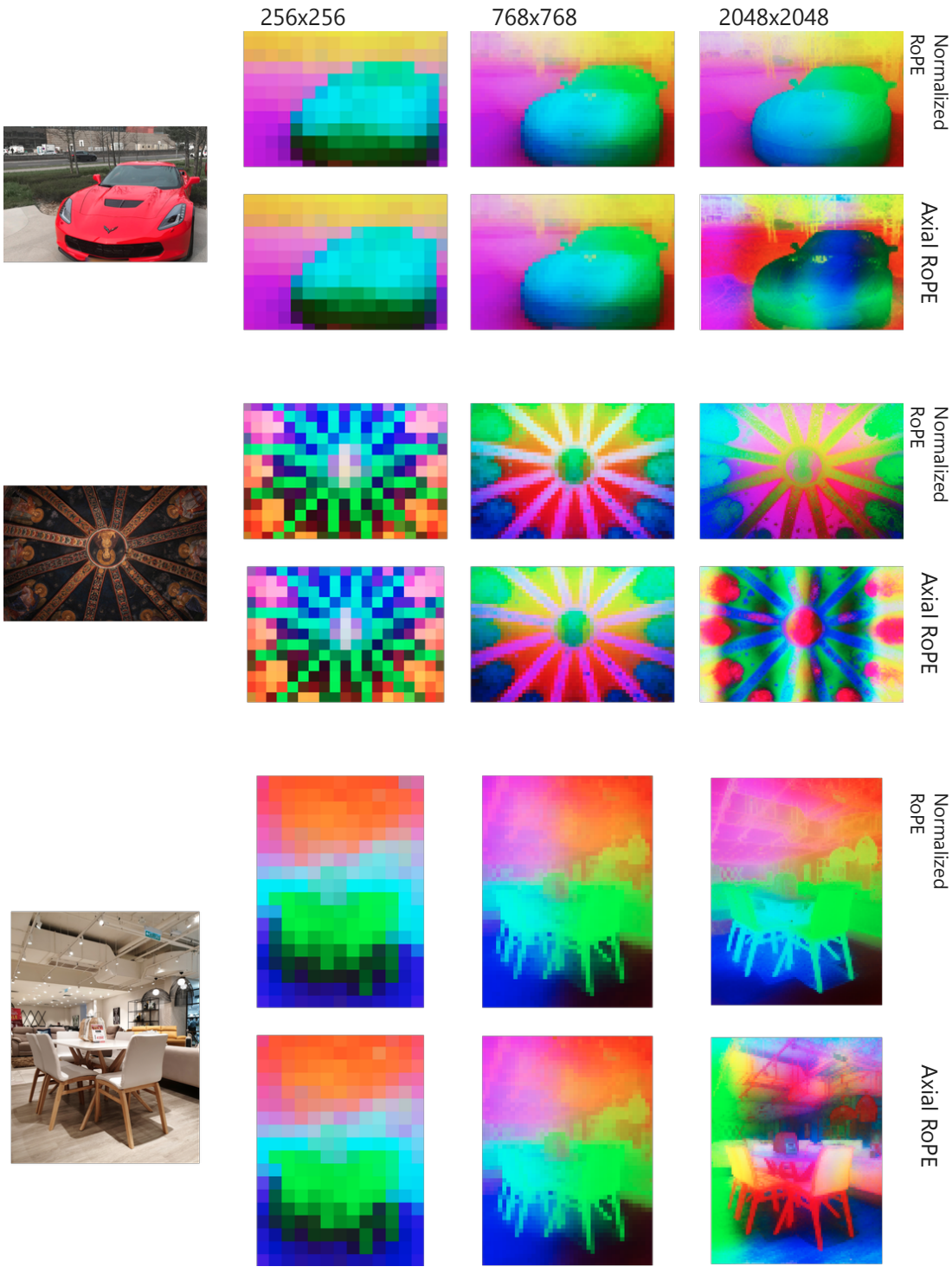

Figure 6. Impact of positional encoding on unseen resolutions. We compare feature map consistency across resolutions (256×256 to 2048×2048 pixels) for Normalized RoPE (top) versus standard Axial RoPE (bottom) using the distilled DINOv3 head. While both methods perform comparably at the training resolutions (up to 768×768 pixels), Axial RoPE degrades at high resolutions, losing object consistency and introducing artifacts. In contrast, Golden RoPE maintains strong scale invariance and feature coherence even at extreme, unseen resolutions (2048×2048 pixels, *i.e.*, 16k patches), demonstrating better extrapolation capabilities for MT-distillation.

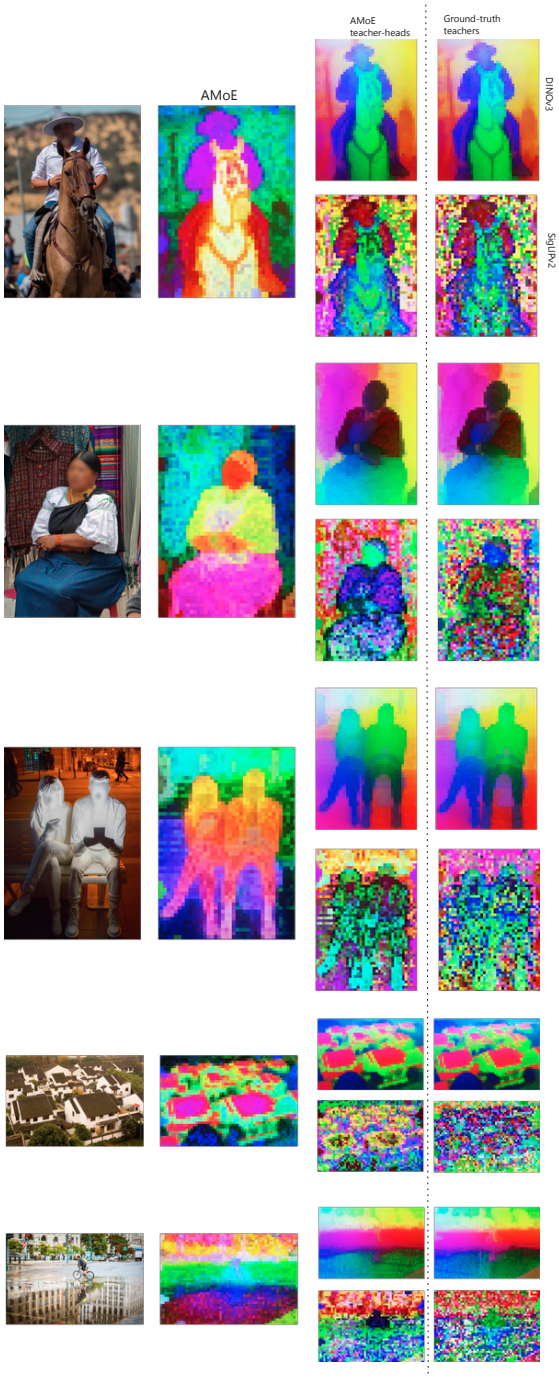

Figure 7. PCA-maps of learned representations: the original image, the shared SigLino backbone features, the student's teacher-specific projections (top: DINOv3 head, bottom: SigLIP2 head), and the corresponding ground-truth teacher features. The student closely reconstructs the teacher's distributions.

metric Relational Knowledge Distillation (ARKD) term to ensure structural consistency across the token-balanced batch.

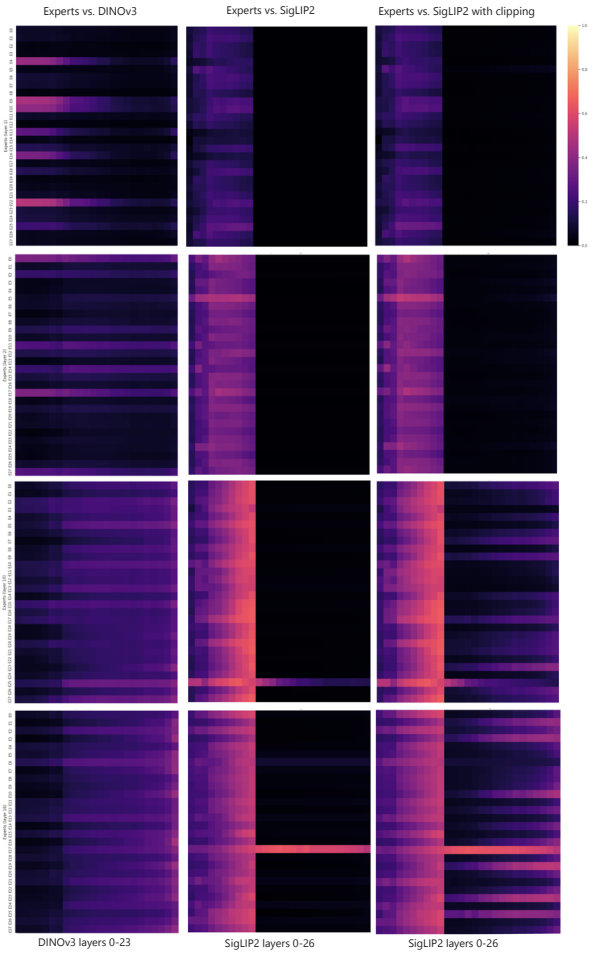

Figure 8. Linear CKA alignments between MoE experts and teacher layers at several SigLino layers.

13. Detailed Ablation Benchmarks

We provide the full per-dataset results for our ablations. Table 11 and Table 14 detail the comparison between our curated OpenLVD200M dataset and random subsampling, highlighting the consistent gains across fine-grained classification and retrieval tasks. Similarly, Table 12 and Table 13 present the full breakdown of the ARKD ablation.

14. Supplementary Results: Intermediate Model Sizes

To explore the trade-off between model size and performance, we evaluate two intermediate-scale models distilled using the same recipe as the main SigLino-0.6B: SigLino-30m (30M parameters) and SigLino-70m (70M parameters). These models validate the scalability of our distillation approach across different parameter budgets and provide useful baselines for compute-constrained settings. Re-

Method		Image–Text Classification @ 256×256 (Top-1)								kNN Classification @ 256×256 (Top-1)					
Model	Head	IN	C101	CUB	Food	Flow	DTD	Air	Avg	IN	CUB	Food	DTD	Air	Avg
Random	DINOv3	68.97	87.76	68.67	87.94	83.37	62.57	47.42	72.39	76.45	82.21	90.98	77.61	66.73	82.27
	SigLIP2	66.42	88.36	59.21	86.31	78.16	64.01	50.21	70.38	71.60	69.76	90.72	73.30	66.61	78.54
	Ensemble	70.51	**89.47**	70.40	**88.84**	85.32	67.02	53.18	74.96	76.18	81.93	**91.53**	76.70	69.96	82.66
OpenLVD	DINOv3	72.45	87.57	**74.38**	87.69	87.14	63.10	62.56	76.41	77.89	**84.12**	90.94	**78.51**	74.64	84.31
	SigLIP2	70.29	88.12	63.38	86.10	86.17	64.84	66.49	75.06	74.25	73.02	90.51	74.36	79.80	81.89
	Ensemble	**73.74**	89.44	73.95	88.53	**88.71**	**67.55**	**71.82**	**79.11**	**78.07**	83.33	91.32	77.23	**80.76**	**85.08**

Table 11. Ablation of data curation strategy (OpenLVD200M vs. Random Uniform Sampling) on Image-Text and kNN classification tasks at 256×256 resolution. OpenLVD yields consistent gains across all benchmarks, especially on fine-grained tasks like FGVC-Aircraft.

Method		Image–Text Classification @ 256×256 (Top-1)								kNN Classification @ 256×256 (Top-1)					
Loss	Head	IN	C101	CUB	Food	Flow	DTD	Air	Avg	IN	CUB	Food	DTD	Air	Avg
Vanilla	DINOv3	63.00	85.00	39.59	75.24	81.31	58.28	43.56	63.71	78.13	84.26	91.06	78.51	75.87	81.57
	SigLIP2	71.03	87.92	66.81	85.64	87.38	64.88	73.41	76.72	74.97	76.16	90.74	74.36	**85.79**	80.40
	Ensemble	72.03	88.58	69.07	85.67	87.99	66.51	73.53	77.62	**79.07**	84.41	**91.70**	77.18	85.34	83.54
RKD	DINOv3	72.57	87.86	**76.64**	87.58	87.14	63.67	66.94	77.48	77.71	84.33	90.87	77.77	76.11	81.36
	SigLIP2	70.61	88.32	67.45	85.11	86.89	64.34	69.63	76.05	74.69	75.90	90.57	74.15	82.76	79.61
	Ensemble	74.07	89.15	75.84	**88.03**	**88.96**	66.70	73.65	79.49	78.10	84.21	91.42	76.81	82.52	82.61
ARKD	DINOv3	72.75	88.29	75.93	87.66	86.89	63.81	68.44	77.68	78.05	**84.91**	91.04	**79.10**	76.83	81.99
	SigLIP2	70.77	87.82	67.29	84.70	86.89	64.70	74.19	76.62	74.70	76.03	90.56	75.16	85.76	80.44
	Ensemble	**74.28**	**89.24**	76.17	87.97	88.71	**67.45**	**77.67**	**80.21**	78.33	84.72	91.52	77.93	85.64	**83.63**

Table 12. Ablation of Asymmetric vs. Symmetric Relational Knowledge Distillation (RKD) on classification tasks at 256×256. ARKD preserves the gains in image-text alignment from Symmetric RKD while recovering the kNN performance lost by the symmetric constraint.

Loss	Head	MSCOCO5k		Flickr30k	
		T2I@1	I2T@1	T2I@1	I2T@1
Vanilla	DINOv3	38.78	53.76	66.22	82.30
	SigLIP2	45.69	61.12	71.00	84.80
	Ensemble	48.15	64.10	74.30	89.50
Sym. RKD	DINOv3	42.17	60.16	70.22	85.80
	SigLIP2	45.31	60.26	70.12	84.30
	Ensemble	48.32	**66.28**	74.70	**89.50**
Asym. RKD	DINOv3	42.68	60.52	69.86	86.70
	SigLIP2	45.11	59.82	71.36	83.60
	Ensemble	**48.51**	65.92	**74.90**	89.40

Table 13. Impact of ARKD on retrieval (Recall@1) for MSCOCO5k and Flickr30k at 256×256. Relational distillation provides a significant boost over the Vanilla baseline, especially for the DINOv3 head.

sults show that even at significantly reduced model sizes, the multi-teacher distillation recipe maintains strong performance on classification, retrieval, and segmentation tasks.

Method	Head	MSCOCO5k		Flickr30k	
		T2I@1	I2T@1	T2I@1	I2T@1
Random	DINOv3	42.87	60.22	69.94	87.00
	SigLIP2	46.02	58.98	71.72	84.00
	Ensemble	48.78	65.86	74.58	89.80
OpenLVD	DINOv3	43.62	60.94	72.32	88.70
	SigLIP2	47.03	60.34	72.64	84.20
	Ensemble	**49.51**	**66.02**	**76.36**	**91.10**

Table 14. Retrieval performance (Recall@1) on MSCOCO5k and Flickr30k at 256×256, comparing OpenLVD200M against Random Uniform Sampling.

Benchmark	Metric	Random	OpenLVD200M	Δ
FGVC-Aircraft	IT	53.18	**71.82**	**+18.64**
CUB-200	IT	70.40	**73.95**	**+3.55**
ImageNet	(I-T)	70.51	**73.74**	**+3.23**
ImageNet	(kNN)	76.18	**78.07**	**+1.89**

Table 15. OpenLVD200M: benchmark-specific improvements.

15. Details on OpenLVD200M Curation

As outlined in §3, we construct OpenLVD200M using the hierarchical clustering and sampling pipeline proposed by [41] to mitigate the long-tail biases inherent in web-scraped data. Figure 9 visually demonstrates the semantic structure captured by this process. The hierarchy organizes concepts from broad, high-level categories (Level 4, grey borders)—such as "text-heavy images", "flowers", or "musical instruments"—down to increasingly specific subtypes. By sampling uniformly across these nodes rather than the raw data distribution, we ensure that rare, fine-grained concepts (the leaves of the tree) are selected with the same probability as common head concepts.

Implementation and Efficiency. To scale this approach to our 2.3B image pool (DFN + LAION) using limited compute (12 nodes of 8×A100), we introduce specific efficiency modifications to the original algorithm [41]. Instead of clustering the full dataset globally, we adopt a two-step assignment strategy: (i) We embed all images using the DINOv3 ViT-B encoder. (ii) We uniformly subsample a representative set of 1B images to learn the hierarchy via 4-level k-means, resulting in a tree structure with 20k (Level 4), 50k (Level 3), 500k (Level 2), and 20M (Level 1) centroids. (iii) We assign the remaining 1.3B images to these pre-computed Level-1 centroids. (iv) We perform hierarchical sampling on the fully assigned population to produce the balanced 200M subset.

```
def ComputeLoss(student, teachers, global_batch):
    L_total = 0
    # Gather global batch stats for stable
    normalization
    N_global = Sum(global_batch.num_images)
    For T in ["dino", "siglip"]:
        # Unpack per-image student (s) and
    teacher (t) features
        # s_sum/t_sum: Global Summary Token (CLS
    or Pooler)
        # s_pat/t_pat: Dense Patch Tokens
        s_sum, s_pat, s_reg = student[T]
        t_sum, t_pat, t_reg = teachers[T]

        # --- A. Local & Representation Alignment
     ---
        # Note: Patch loss normalized by token
    count per image (N_q)
        L_patch = Sum([MSE(s_pat[q], t_pat[q]) /
    N_q for q in batch])
        L_sum   = Sum([1 - CosineSim(s_sum[q],
    t_sum[q]) for q in batch])

        # DINOv3 specific: Align Registers
        if T == "dino":
             L_total += MSE(s_reg, t_reg)

        # --- B. ARKD ---
        # 1. Compute Global Distance Matrices
        t_all = AllGather(t_sum)  # Gather from
    all ranks
        s_all = AllGather(s_sum)
        D_t = PairwiseDist(t_sum, t_all) #
    Teacher geometry
        D_s = PairwiseDist(s_sum, s_all) #
    Student geometry

        # 2. Normalize by Teacher Scale (Scale
    Invariance)
        scale = Mean(D_t)
        D_t, D_s = D_t / scale, D_s / scale

        # 3. Asymmetric Weighting (Intra-batch
    Median Split)
        median_dist = Median(D_t)
        # Penalize expansion only if samples are
    close (Intra-cluster)
        # Penalize shrinkage only if samples are
    far (Inter-cluster)
        W_expand = (D_t < median_dist)
        W_shrink = 1 - W_expand

        L_arkd = Mean(W_expand * SmoothL1(Max(D_s
     - D_t, 0)) +
                      W_shrink * SmoothL1(Max(D_t
     - D_s, 0)))

        # Accumulate (Normalized by Global Batch
    Size)
        L_total += (L_patch + L_sum + L_arkd) /
    N_global

    return L_total
```

Listing 2. SigLino loss pseudo-code

Figure 9. Concept hierarchy captured by the 4-level clustering. Each column represents a high-level semantic cluster (Level 4, grey borders), containing progressively finer granularities: Level 3 (brown borders), Level 2 (cyan borders), and Level 1 (black borders). From left to right, we show clusters for text-heavy images, flowers, and toys. The hierarchy naturally organizes concepts from broad categories to specific sub-types and fine-grained instances.