A More Word-like Image Tokenization for MLLMs

Supplementary Material

A. Implementation Details

Hyperparameters. Our implementation closely follows the standard LLaVA-1.5 training pipeline, with only minimal adjustments required to integrate DiVT into the multimodal architecture. Pretraining is performed for one epoch following the LLaVA-1.5 recipe; we use batch size of 256, initial learning rate of 10^{-3} with cosine decay and 3% warm-up, AdamW without weight decay, and DeepSpeed Stage 2. We update only the parameters of our projector. Finetuning adopts the same training configuration, differing only in the decreased batch size of 128, a reduced learning rate of 2×10^{-5}, and DeepSpeed Stage 3, through which the projector and LLM are jointly optimized. Unless stated otherwise, we adopt $\theta = 0.65$ as the default threshold for our DiVT.

More on Model Architecture. We use two ViT-based vision encoders, `facebook/dinov2-large` and `google/siglip-large-patch16-384` for experiment in Tab. 3. Since DINOv2 is originally trained with 224×224 inputs and therefore outputs merely 256 patch embeddings, we modify its preprocessing to accept 336×336 resolution so that it produces 576 patch features, making its output shape consistent with CLIP or SigLIP encoders. All experiments are conducted on a compute cluster equipped with eight NVIDIA RTX A6000 GPUs (48GB).

B. Performance with Various Thresholds θ

A key advantage of DiVT is that the similarity threshold θ controls semantic granularity of the clusters. We experiment with $\theta \in \{0.3, 0.4, 0.5, 0.62, 0.65, 0.75, 0.8\}$, spanning coarse to highly fine-grained clustering regimes.

Tab. I provides the full numerical results together with the corresponding average token counts. The table clearly shows a larger θ expands the token budget and this increase tends to align with the observed performance gains across most benchmarks. Once the threshold θ reaches beyond 0.75, however, the clusters become overly fragmented and semantically redundant, leading to a mild degradation in accuracy despite further increases in token count. This behavior confirms that a moderate granularity provides the best trade-off between accuracy and token efficiency.

C. Training and Inference Time Analysis

Tab. II summarizes the computational advantages of DiVT. By treating substantially fewer visual tokens than the 576-token MLP baseline, our DiVT significantly shortens the multimodal forward pass and leads to notable speedups at both training and inference.

The efficiency comparison in Tab. II highlights how DiVT substantially reduces computational cost across pretraining, finetuning, inference, and KV-cache memory usage. The largest improvement appears in the pretraining stage, where the LLM is frozen and the computational cost is largely determined by the sequence length processed through each transformer layer. Lower thresholds significantly shorten this sequence, leading to proportionally large reductions in attention computation and, in turn, overall pretraining time. Finetuning likewise benefits from the reduced token count, though the gains are somewhat moderated by the need to update the full LLM. Still, the lighter visual sequence consistently improves optimization efficiency, providing meaningful savings in both training phases.

Inference reveals additional practical advantages. Because the sequence length linearly scales KV-cache memory, reducing the number of tokens shrinks the KV-cache footprint by over 90% at coarse thresholds such as θ=0.4. Such reductions substantially ease the memory burden and suggest potential scalability benefits for scenarios involving multi-images or video, where limited KV-cache capacity and context length frequently become bottlenecks.

Prefill latency is influenced both by the number of visual tokens and by the cost of our clustering algorithm. At low thresholds, the substantial reduction in token count dominates the overall computation, making the clustering overhead comparatively minor and enabling nearly a 2× speedup over the MLP projector. Even at θ=0.75, where the clustering step becomes slightly heavier, the resulting prefill latency remains close to that of the MLP baseline, indicating that the additional cost introduced by clustering is not a major bottleneck in practice. Crucially, this overhead appears only once during the prefill stage. After tokenization, the subsequent decoding process depends solely on the final number of visual tokens, not on how they were formed. As a result, DiVT benefits from reduced inference-time computation throughout the entire generation process, whereas the MLP projector must continue to handle a much longer visual sequence at every decoding step. This separation demonstrates that the overhead associated with clustering is limited while the gains in end-to-end efficiency are substantial.

Overall, the threshold parameter θ allows practitioners to modulate computational cost with a single control knob, ranging from highly compact and efficient configurations to more detailed representations when resources permit. This simple controllability, together with consistently lower KV-cache usage and reduced decoding cost, makes DiVT an ap-

θ	# Tokens	MMB	VQA^{v2}	GQA	MME	MM-Vet	VQA^{Text}	SQA^{IMG}	POPE
0.3	13.5	64.2	75.3	59.2	1462.8	28.0	55.4	69.4	84.3
0.4	22.4	64.7	76.4	60.1	1450.9	31.7	56.1	69.1	84.8
0.5	35.7	65.0	77.0	60.6	1458.2	31.7	57.1	68.2	85.8
0.62	63.7	64.3	77.7	61.6	1463.0	30.6	57.0	**70.6**	**86.2**
0.65	74.1	65.5	77.7	61.4	**1465.7**	**32.1**	57.2	68.1	85.8
0.75	136.5	**66.7**	**78.2**	**62.0**	1457.6	30.2	**57.7**	70.0	**86.2**
0.8	175.3	65.3	**78.2**	61.9	1456.5	31.3	57.4	68.8	85.8

Table I. **Performance of our DiVT under varying similarity thresholds**

Method	Pretraining (h)	Finetuning (h)	Inference (h)	KV-Cache (MB)	Prefill Latency (ms)
MLP Projector	5.7 (100%)	20.0 (100%)	5.5 (100%)	288.0 (100%)	138.2 (100%)
DiVT$_{\theta=0.4}$	1.1 (19.3%)	12.9 (64.5%)	3.4 (61.8%)	11.0 (3.8%)	71.3 (51.6%)
DiVT$_{\theta=0.5}$	1.4 (24.6%)	13.1 (65.5%)	3.6 (65.5%)	17.6 (6.1%)	76.6 (55.4%)
DiVT$_{\theta=0.65}$	1.9 (33.3%)	13.7 (68.5%)	3.9 (70.9%)	36.8 (12.8%)	104.4 (75.6%)
DiVT$_{\theta=0.75}$	2.7 (47.4%)	14.5 (72.5%)	4.7 (85.5%)	68.1 (23.6%)	138.3 (100.1%)

Table II. **Training and inference cost of DiVT across different similarity thresholds**. Training time is measured using eight RTX A6000 GPUs, and inference time is measured on the VQAv2 evaluation set using a single RTX A6000 GPU. KV-cache memory is computed analytically from the LLaMA-7B architecture, where each visual token contributes approximately 0.5 MB of KV-cache. Prefill latency is measured by averaging multiple stable forward passes after warm-up.

pealing alternative to the MLP projector from an efficiency standpoint.

D. Additional Attention Map Visualizations

Fig. I illustrates additional examples of attention patterns comparing DiVT with the standard MLP projector. To visualize the attention received by each textual token, we aggregate the attention weights assigned to a given DiVT token and project them onto all patches belonging to that token's cluster. This cluster-level visualization highlights which semantic region the model relies on when it processes each text token.

Since DiVT (bottom) aggregates patches into coherent semantic clusters, the resulting attention maps reveal clear and localized patterns. Each textual token tends to focus on a distinct visual concept, making the grounding behavior easy to interpret. In contrast, MLP projectors (top) operate at the patch-level and thus often assign disproportionately high attention to a small subset of tokens, regardless of the query. This obscures which visual evidence the model is using, thereby leading to diffused or noisy activation patterns and hurting interpretability.

E. Additional Cluster Visualization

We provide additional qualitative examples in Fig. II that illustrates how DiVT forms semantically coherent clusters across diverse scenes. Each example assigns a distinct color to patches belonging to the same cluster, allowing us to inspect how the feature-space grouping translates into spatial regions in the original image. As in the main manuscript, the number of clusters is determined dynamically based on the image content, and the resulting visual patterns clearly reflect this adaptivity; that is, relatively simpler scenes yield compact clusters with large spatial support, while more complex or cluttered scenes produce a larger number of fine-grained clusters. Varying the similarity threshold θ also produces the expected behavior, where a higher value leads to a more fragmented grouping.

These examples highlight that DiVT consistently discovers semantic units such as objects, parts, and salient regions without any pixel-level annotation, segmentation masks, or bounding box supervision. Clusters formed purely from feature similarity often align with intuitive semantic boundaries, illustrating the effectiveness of our disentanglement mechanism.

F. Analysis on Resulting Token Counts

Tab. 5 of the main manuscript reports the average number of tokens produced at $\theta = 0.65$. In this section, we extend this by providing mean and standard deviation statistics across multiple thresholds. These measurements are computed over all images within each benchmark, illustrating how DiVT adapts its token budget depending on both the contents of input images and the similarity threshold.

In Tab. III, we observe clear and consistent trends of the resulting token counts across thresholds and datasets. A lower threshold such as $\theta = 0.4$ yields compact token

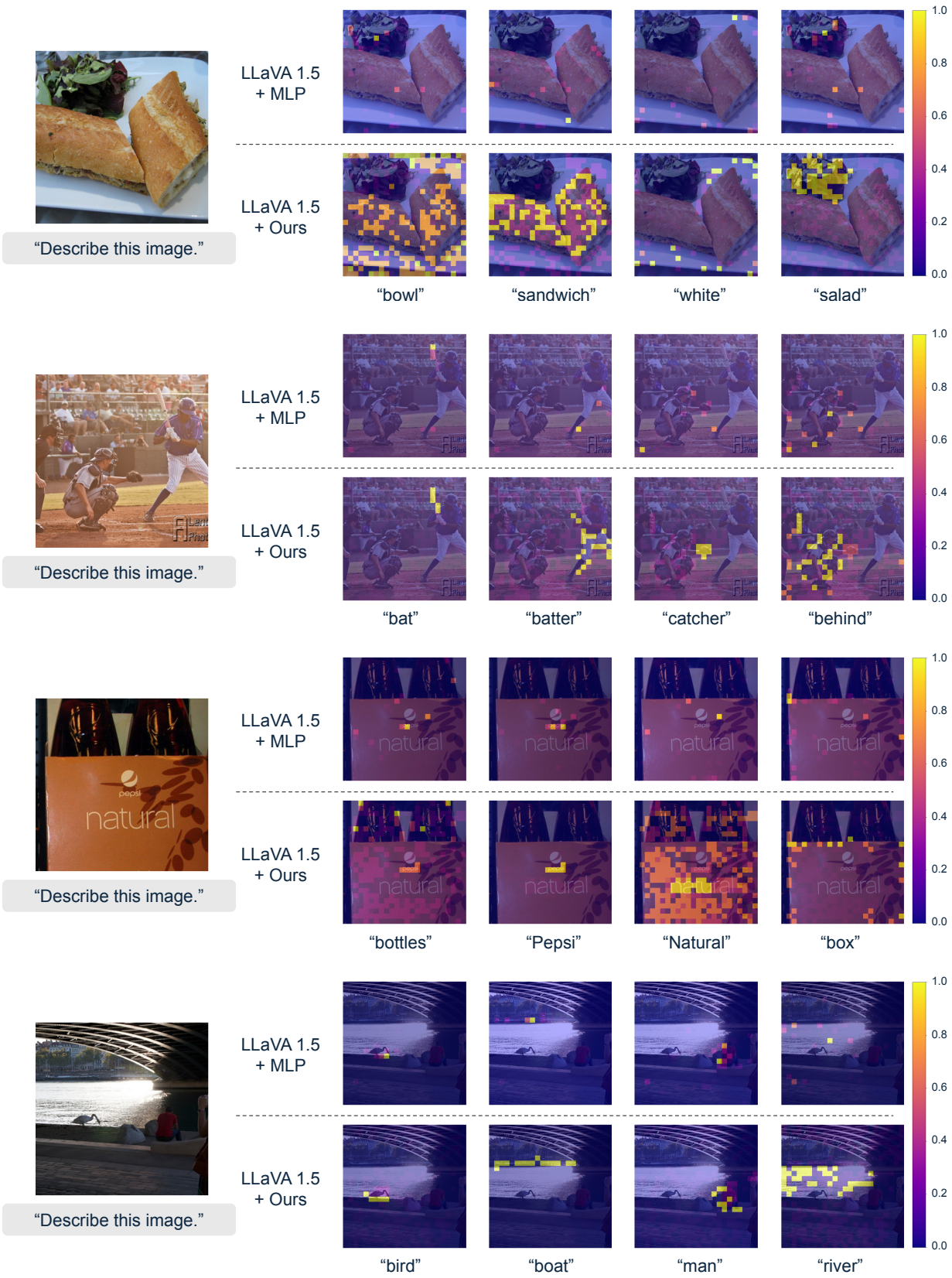

Figure I. **Additional attention map comparisons between DiVT and the MLP projector**. The cluster-based representation in DiVT leads to more consistent and interpretable attention behavior across textual tokens.

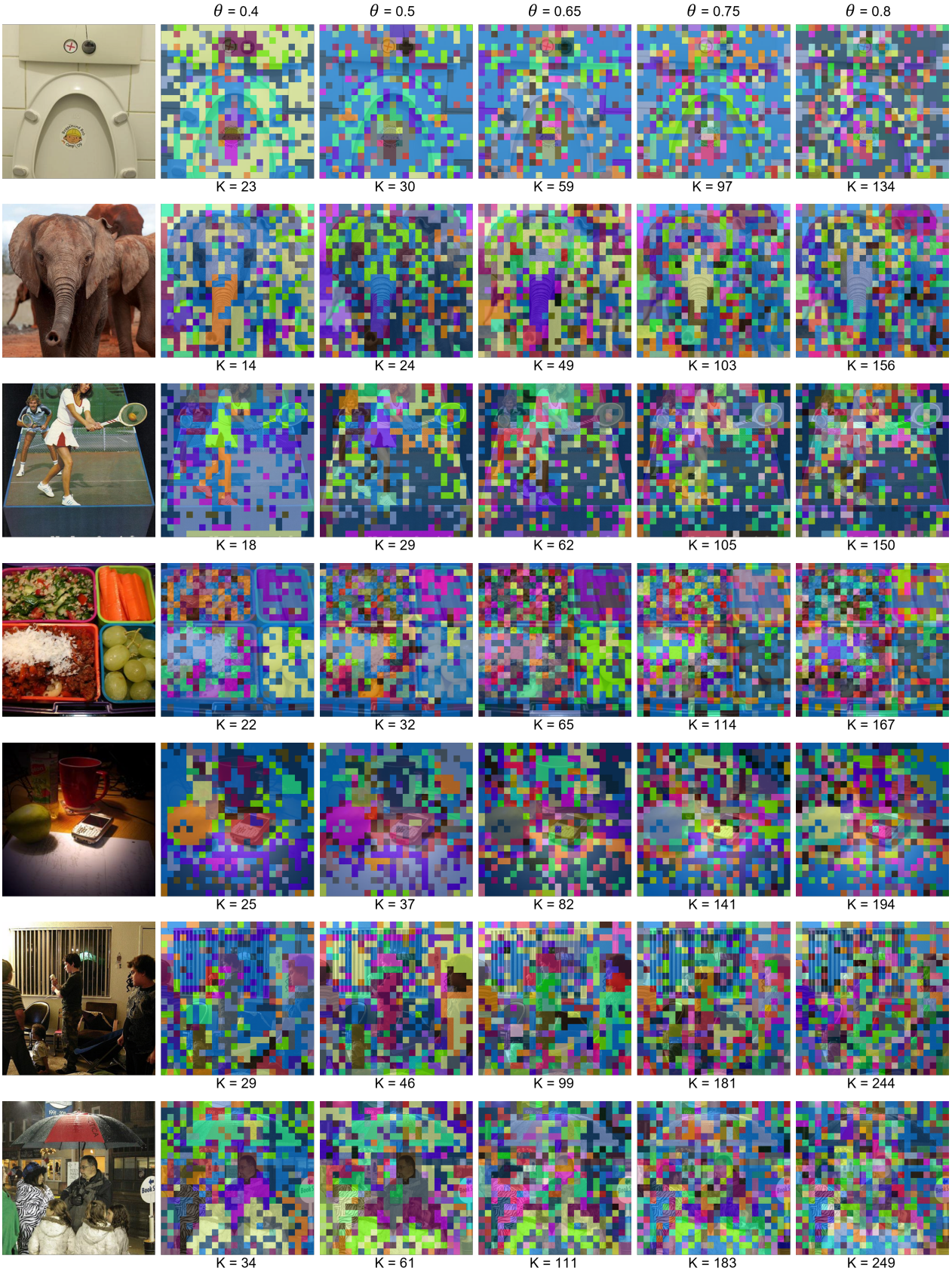

Figure II. **Additional cluster visualizations produced by DiVT.**

θ	**Pretrain**	**Finetune**	**MMB**	**VQA^{v2}**	**GQA**	**MME**	**MM-Vet**	**VQA^{Text}**	**SQA^{IMG}**	**POPE**	**Avg.**
0.3	14.8 ± 8.4	14.9 ± 5.3	11.2 ± 4.3	13.3 ± 3.9	14.6 ± 3.7	12.6 ± 5.1	15.4 ± 7.3	18.3 ± 8.1	10.6 ± 5.0	14.3 ± 3.9	13.5 ± 4.3
0.4	22.1 ± 11.7	23.7 ± 7.9	18.0 ± 6.5	22.0 ± 6.4	24.7 ± 6.2	20.5 ± 7.2	23.0 ± 9.9	28.7 ± 11.8	16.2 ± 6.5	24.0 ± 6.2	22.4 ± 6.9
0.5	32.4 ± 15.7	37.2 ± 12.2	27.1 ± 10.9	35.1 ± 10.3	40.0 ± 10.5	32.5 ± 11.4	33.0 ± 13.2	45.8 ± 19.4	22.4 ± 8.1	39.3 ± 11.8	35.7 ± 11.4
0.62	54.9 ± 23.4	66.0 ± 18.8	50.0 ± 17.6	63.1 ± 18.3	70.1 ± 17.1	59.1 ± 18.3	57.1 ± 18.1	72.8 ± 24.7	41.6 ± 13.4	68.7 ± 18.2	63.7 ± 18.9
0.65	62.3 ± 26.0	76.5 ± 24.7	58.3 ± 20.5	73.5 ± 21.1	81.3 ± 19.3	69.3 ± 21.3	65.2 ± 19.8	83.4 ± 27.7	48.3 ± 15.7	80.1 ± 21.0	74.1 ± 21.8
0.75	110.3 ± 42.0	138.8 ± 41.2	108.4 ± 42.0	136.2 ± 38.5	146.2 ± 33.8	133.2 ± 44.8	114.2 ± 36.1	145.8 ± 47.3	88.5 ± 31.1	147.1 ± 39.2	136.5 ± 39.6

Table III. **Resulting token counts of our proposed method across datasets for multiple thresholds**. Reported as mean $\pm$ standard deviation. The Avg. column averages over evaluation benchmarks only.

sets with small variance, as visually dominant regions are merged into broader clusters. Increasing θ makes the clustering more selective, producing more finer-grained tokens and higher token-count variance, particularly on benchmarks containing text, cluttered objects, or complex compositions (*e.g.*, TextVQA or POPE). Conversely, datasets with simpler scenes, such as SQA-IMG, maintain a narrow token-count range across all thresholds.

Collectively, DiVT adjusts its token budget according to the inherent visual complexity of each image rather than relying on a fixed grid-based reduction. The resulting distribution of token counts demonstrates that DiVT responds naturally to semantic density, enabling compute-efficient representations without sacrificing expressiveness.