Supplementary Material for
Uncertainty-guided Compositional Alignment with Part-to-Whole Semantic Representativeness in Hyperbolic Vision-Language Models

Hayeon Kim[1,*] Ji Ha Jang[1,*] Junghun James Kim[2] Se Young Chun[1,2,†]
[1] Dept. of Electrical and Computer Engineering, [2] INMC & IPAI
Seoul National University, Republic of Korea
{khy5630, jeeit17, jonghean12, sychun}@snu.ac.kr
*Authors contributed equally. †Corresponding author.

S.1. Implementation details

S.1.1. Model architecture

Our text encoder follows the CLIP [23] design and uses a 12-layer, 512 dimensional Transformer [28]. The maximum input length is set to 77 tokens with a vocabulary size of $49,408$. For images, we adopt a Vision Transformer [8] and experiment with two capacity configurations, ViT-S and ViT-B [5, 26], both using a patch size of 16. These architectural choices are consistent with prior works [7, 21]. During training, we apply the same image augmentations as OpenCLIP [13], including random cropping, random grayscale conversion, and random color jittering, and resize all images to 224×224.

S.1.2. Model initialization

The curvature of Lorentz space is initialized to $\kappa = 1.0$ and treated as a learnable parameter, while being clamped in $[0.1, 10.0]$ for numerical stability. The final learned value converges to $\kappa = 0.1$, consistent with those used in prior hyperbolic methods [7, 21, 24]. Before projecting representations onto the Lorentz model, we apply learnable scaling factors to image and text vectors. These scalars are initialized as $c_{\text{img}} = c_{\text{txt}} = \frac{1}{\sqrt{512}}$, following prior work [7, 21]. The temperature parameters are also learnable. The global-local logit scale τ_{gl} is initialized to 0.06, while the local and global logit scales, τ_l and τ_g, are initialized to 0.07. All temperature values are clipped at a minimum of 0.01. Values of η parameter are set to $\eta_{intra} = 1.2$ for intra-modality entailments and $\eta_{inter} = 0.7$ for inter-modal entailments (Eq.14) and $K = 0.1$ (Eq.13), following [21]. In Eq.14, we set $\alpha = 0.1$. For Eq.17, the weighting coefficients are $\lambda_1 = 0.5$ and $\lambda_2 = 10.0$. In Eq.18, we use $\lambda_{ent} = 0.2$.

S.1.3. Optimizer and hardware

Our model is trained for 500K steps using four A100 GPUs with a batch size of 768. We employ the AdamW optimizer [19], setting $\beta_1 = 0.9, \beta_2 = 0.98$, and a weight decay of 0.2. The decay is excluded for the learnable parameters, including the temperature parameters, curvature, and the scaling factors c_{img} and c_{txt}. We adopt a cosine learning-rate scheduler [18] with a maximum learning rate of 5×10^{-4}, with a 4k-step linear warm-up period.

S.2. Additional details on experiments

S.2.1. Training details on other models

We employ CLIP [23], MERU [7], and HyCoCLIP [21] models trained on the Grounded Image-Text Pairs (GRIT) dataset [22], using the reproduced version released by [21]. For CLIP [23] and MERU [7], we adopt the variants trained without part images, as their original training pipelines do not incorporate part-level data and prior work [21] reports that including part images does not lead to performance improvements. The GRIT dataset contains 20.5 million grounded vision-language pairs and 35.9 million box-level annotations describing objects within each scene, derived from the larger COYO-700M corpus [3]. In addition, we train ATMG [24] on the same GRIT dataset using a batch size of 768 for 500K iterations, preserving the optimization settings specified in their original implementation.

S.2.2. Downstream tasks

S.2.2.1. Zero-shot image classification

For MERU [7], HyCoCLIP [21] and UNCHA (Ours), similarity between text and image embedding is computed with Lorentzian inner product. For CLIP [23], similarity is measured using the Euclidean inner product, while for ATMG [24], we adopt its exterior angle-based similarity. The same similarity formulation for each model is consistently applied across all remaining downstream tasks. In zero-shot image classification, we treat the label set as a collection of text queries [9] and apply prompt ensembling

for each label by encoding multiple prompt variants and averaging their embeddings before generating the final textual representations, following previous works [7, 21, 24]. Using these embedded text queries, we compute image-text similarities and report top-1 accuracy averaged over classes.

S.2.2.2. Zero-shot retrieval

In zero-shot text-to-image retrieval, we compare every text caption embedding against all image embeddings and sort the images in descending order of similarity. The same procedure is applied symmetrically for image-to-text retrieval. We compute recall@K for both directions using the ground-truth associations provided by COCO [17] and Flickr30K [15, 32], where a retrieval is counted as correct if at least one paired item appears within the top-K results. All recall metrics are averaged over the full set of queries to produce final results.

S.2.2.3. Hierarchical classification

For hierarchical classification task, we follow the prior work [21] and use the WordNet hierarchy [20] of the ImageNet class labels [6, 25]. The Tree-Induced Error (TIE) quantifies how far the predicted label is from the ground-truth label within the given tree. The Lowest Common Ancestor (LCA) error captures how far each label is from their deepest shared ancestor, defined as the sum of the edge-weighted distances from the predicted and true labels to the LCA. Set-based metrics compare the ancestor sets of the predicted and true labels: using all ancestor nodes for each label, we compute Jaccard similarity, hierarchical precision, and hierarchical recall based on their set intersection.

S.2.2.4. Zero-shot multi-label classification

Multi-label classification. We perform multi-label classification experiments on the MS-COCO [17] and VOC [10] datasets and report performance using mean Average Precision (mAP). This task evaluates whether the VLM can correctly predict the set of classes present in each image by comparing its predictions against the binary ground-truth labels. Because the baseline models include both hyperbolic and Euclidean variants, their similarity score ranges differ substantially: Euclidean VLMs typically output similarities within $[0, 1]$, whereas hyperbolic similarity scores generally fall at or below –10. To ensure a fair comparison across models, we apply an additional normalization step to the similarity scores before computing the evaluation metrics.

Multi-object representation. This benchmark is designed to evaluate more complex multi-object scenarios using the ComCo and SimCO datasets [1]. As described in [1], this setting allows us to assess how well a VLM's image encoder represents individual objects within multi-object scenes and to analyze whether its representations exhibit biases with respect to object size. ComCo consists of images containing realistic 3D asset objects, such as cars or airplanes, arranged in sets of N, while SimCo contains synthetic 3D assets such as blue spheres, cones, and other primitive shapes. In both datasets, each image contains between two and five objects, so the labels '2 obj.', '3 obj.' in Tab.5 of the main text refers to sets of images that contain exactly two or three objects, respectively. These images include various combinations of object sizes and spatial arrangement. For instance, '3 obj.' set contain one large object and two smaller objects in different location. A separate classifier is trained for each set on top of the features produced by the VLM's image encoder, grouped by the number of objects. The model is evaluated on its ability to distinguish all components across different sizes and positions, and at test time we assess whether the trained classifier can correctly identify each component in response to new text queries. Extended results evaluated with the ViT-S backbone are provided in Tab. S.3.

S.2.2.5. Part-level alignment with hard negatives

This benchmark, introduced in [27], evaluates whether a VLM can correctly associate captions with the appropriate image subregions when multiple submasks and captions exist for the same image, using the 7,805 images from the summarized Densely Captioned Images (sDCI) dataset. The original DCI dataset provides dense textual annotations, including multiple captions, subcaptions, and visual descriptions per image. To align these annotations with CLIP-style input constraints, all LLM-generated captions are truncated to 77 tokens to form sDCI. Each image contains several subcrops, each paired with one or more summarized captions as well as LLM-generated negatives. Retrieval-style evaluations are constructed by placing multiple subcrops and captions from the same image within a single batch, requiring the model to identify which caption corresponds to which region.

We report the result of 'All Pick5-SCM', 'All Pick5-Neg', and 'All-Hard Negs' in the main paper, and include all metrics below, tested with both ViT-B, ViT-S at Tab. S.2. In 'All-SCM', one summarized caption is paired with each subcrop, and the model must identify the caption that describes that specific region, distinguishing it from captions corresponding to other subcrops of the same image as well as from other in-batch captions. In 'All-Neg', each subcrop's caption is evaluated against an LLM-generated negative to test positive-negative discrimination. The 'All Pick5-SCM' setting follows the structure of 'All-SCM' but uses five captions per subcrop, with success only if the correct caption scores higher than all positives from other images. In 'All Pick5-Neg', five summarized captions are paired with each subcrop, and the model succeeds only if all positives score above the negative. In 'Base-Neg', only full images (no subcrops) are used, and each image is paired with

its LLM-generated negative caption to test the models' ability to distinguish between an LLM generated caption and its corresponding LLM-generated negative. Finally, 'All-Hard Negs' follows the same setup as 'All-Neg' but replaces the negative caption with the hardest (highest-scoring) LLM-generated negative across the entire negative pool.

S.2.3. Additional ablation study

S.2.3.1. Ablation study on hyperbolic radius

As discussed in the main paper, for a point $\mathbf{x} \in \mathbb{L}^n$, we define the uncertainty u using the Euclidean ℓ_2 norm of $\mathbf{x}$, since this norm is monotonically proportional to its hyperbolic radius. We represent a point $\mathbf{x} \in \mathbb{R}^{n+1}$ in the Lorentz model using its time–space decomposition:

$$\mathbf{x} = [x_{\text{time}}, \mathbf{x}_{\text{space}}], \qquad x_{\text{time}} \in \mathbb{R},\ \mathbf{x}_{\text{space}} \in \mathbb{R}^n \tag{S.1}$$

The origin of the hyperboloid corresponds to the point $\mathbf{o} = [\sqrt{1/\kappa}, \mathbf{0}]$. Therefore, the hyperbolic radius, defined as the geodesic distance between $\mathbf{x}$ and the origin, can be calculated as:

$$\begin{aligned} d_{\mathbb{L}}(\mathbf{x}, \mathbf{o}) &= \sqrt{\frac{1}{\kappa}}\, \cosh^{-1}(-\kappa \langle \mathbf{x}, \mathbf{o} \rangle_{\mathbb{L}}) \\ &= \sqrt{\frac{1}{\kappa}}\, \cosh^{-1}\big(x_{\text{time}} \sqrt{\kappa}\big) \end{aligned} \tag{S.2}$$

where we used the Lorentzian inner product

$$\langle \mathbf{x}, \mathbf{o} \rangle_{\mathbb{L}} = -x_{\text{time}} \sqrt{\frac{1}{\kappa}} \tag{S.3}$$

To obtain an explicit expression, we use the hyperboloid constraint:

$$\langle \mathbf{x}, \mathbf{x} \rangle_{\mathbb{L}} = -x_{\text{time}}^2 + \|\mathbf{x}_{\text{space}}\|_2^2 = -\frac{1}{\kappa}, \tag{S.4}$$

which implies

$$x_{\text{time}} = \sqrt{\|\mathbf{x}_{\text{space}}\|_2^2 + \frac{1}{\kappa}} \tag{S.5}$$

As we mentioned in preliminaries of main text, we only parameterize the space component of $\mathbf{x}$. Hence, the Euclidean norm satisfies $\|\mathbf{x}_{\text{space}}\|_2 \equiv \|\mathbf{x}\|_2$ for our parameterization. Therefore, the geodesic distance (hyperbolic radius) from the origin to a point $\mathbf{x} \in \mathbb{R}^D$ is given by:

$$d_{\mathbb{L}}(\mathbf{x}, \mathbf{o}) = \frac{1}{\sqrt{\kappa}} \cosh^{-1}\left(\sqrt{1 + \kappa \|\mathbf{x}\|_2^2}\right) \tag{S.6}$$

This expression reveals that the hyperbolic radius is closely related to the Euclidean norm of $\mathbf{x}$, $\|\mathbf{x}\|_2$.

For small $\mathbf{x}$, $\|\mathbf{x}\|_2$, we have the approximation

$$\sqrt{1 + \kappa \|\mathbf{x}\|_2^2} \approx 1 + \frac{\kappa}{2} \|\mathbf{x}\|_2^2 \tag{S.7}$$

Table S.1. **Ablation study on hyperbolic radius.** Replacing our Euclidean-norm surrogate with the explicit hyperbolic radius slightly degrades both classification and retrieval performance. Bold numbers indicate the best within each task group.

	Classification			Retrieval	
Model	Gen.	Fine.	MISC.	Text	Image
Ours (full)	**68.98**	**25.53**	**27.55**	**83.80**	**73.90**
with $d_{\mathbb{L}}(\mathbf{x}, \mathbf{o})$	67.41	24.81	25.55	79.43	72.00

and using $\cosh^{-1}(1+u) \approx \sqrt{2u}$, it follows that

$$d_{\mathbb{L}}(\mathbf{x}, \mathbf{o}) \approx \|\mathbf{x}\|_2 \tag{S.8}$$

showing that the hyperbolic radius grows approximately proportionally to the Euclidean norm for small $\|\mathbf{x}\|_2$.

For large norms, using $\cosh^{-1}(t) \approx \log(2t)$, the radius behaves as:

$$d_{\mathbb{L}}(\mathbf{x}, \mathbf{o}) \approx \frac{1}{\sqrt{\kappa}} \log\big(2\sqrt{\kappa}\|\mathbf{x}\|_2\big) \tag{S.9}$$

indicating a transition to logarithmic growth. Overall, the hyperbolic radius is approximately proportional to the Euclidean norm for small $\|\mathbf{x}\|_2$, but grows logarithmically for large $\|\mathbf{x}\|_2$. This monotonic relationship validates the use of the Euclidean norm of $\mathbf{x}$ as a proxy for its hyperbolic radius. This enables us to avoid the unnecessary hyperbolic computations while preserving the same ordering. The ablation result obtained when training directly with the hyperbolic radius in Eq. S.6 is reported in Tab. S.1, showing slightly reduced performance compared to our full model. This confirms that our Euclidean norm proxy provides an effective surrogate for the hyperbolic radius, enabling more reliable uncertainty estimation during training.

S.2.3.2. Analysis experiments

Analysis of uncertainty modeling. In Fig. 4(a), we investigate how uncertainty reflects the semantic representativeness of local regions within an image. To this end, we randomly crop multiple patches from a single image and compute the uncertainty for each patch. The patches are then arranged according to their uncertainty values, from low to high, progressing from the top-left to the bottom-right. We observe that patches with low uncertainty tend to correspond to semantically meaningful and well-aligned regions, such as prominent objects or structurally informative parts of the scene. In contrast, patches with high uncertainty are often blurred, textureless, or less informative, making them less representative of the overall scene. This qualitative observation suggests that our uncertainty measure effectively captures how well a local region aligns with the global semantics of the image. Additional results on

uncertainty-based ordering are provided in Fig. S.8. In Fig. 4(b), we further provide a quantitative analysis of this behavior using a subset of ImageNet [25]. For each image, we compute the semantic similarity between each cropped part and the corresponding whole image, and examine its relationship with the estimated uncertainty. The resulting scatter plot reveals a strong negative correlation (Corr = -0.739), indicating that parts with higher semantic similarity to the whole tend to have lower uncertainty, while less representative parts exhibit higher uncertainty. This consistent trend supports the interpretation that our uncertainty measure serves as a reliable proxy for semantic representativeness, which is crucial for accurate and robust part-level alignment.

S.2.4. Additional experimental results

S.2.4.1. Part-level alignment with hard negatives

Experimental setting. The experimental setting is described in Sec.4.2.5 of the main text and further detailed in Sec. S.2.2.5.

Experimental results. Tab. S.2 presents the results for the part-level alignment benchmark with hard negatives across evaluation settings described in Sec. S.2.2.5. Across both ViT-S/16 and ViT-B/16 backbones, UNCHA (Ours) consistently achieves the best or second-best performance in nearly every setting. The gains are especially noticeable in the more challenging 'All Pick5-SCM' and 'All Pick5-Neg' settings, where multiple positives per sub-crop make the matching task substantially harder. Even in the 'All-Hard Negs' setting, where each sub-crop must be distinguished from the hardest negative caption selected from the entire LLM-generated negative pool, UNCHA achieves the best performance, demonstrating its robustness against challenging negative distractors. This result indicates that UNCHA (Ours) effectively identifies and differentiates distinct subregions within an image, demonstrating its ability to understand images in a more fine-grained manner.

S.2.4.2. Multi-object representation

Experimental setting. The experimental setting is described in Sec.4.2.4 of the main text and further detailed in Sec. S.2.2.4.

Experimental results. We extend the multi-object representation experiments from the main paper by additionally evaluating ViT-S models. As presented in Tab. S.3, UNCHA (Ours) consistently achieves superior performance across diverse object counts and datasets. This reflects its ability to reliably represent and distinguish individual objects within complex multi-object scenes, demonstrating strong fine-grained and compositional understanding.

S.2.4.3. Zero-shot semantic segmentation

Experimental setting. Zero-shot semantic segmentation refers to benchmark settings where additional attention-modulation methods (such as SCLIP [29] and NACLIP [12]) are integrated into the model to extract not only class-level features but also the dense features produced by the backbone. Using these dense features, the model performs classification by comparing them against the class texts from existing datasets. In our experiments, we first use NACLIP to extract dense features and then compute their similarity to class texts, evaluating how accurately the model localizes fine-grained regions based on the mIoU metric. However, semantic segmentation is substantially more challenging than standard image classification, so instead of relying solely on text–image matching as in typical classification, we further reduce the modality mismatch by extrapolating the text embeddings from the root of the hyperbolic space for all hyperbolic-based models.

Experimental results. As shown in Tab. S.4 and Fig. S.6–S.7, our method consistently achieves strong performance across both the ViT-S and ViT-B backbones, indicating that it captures fine-grained details in images more effectively than existing approaches. Furthermore, the results demonstrate that our method produces more coherent region assignments and reliably handles scenes containing multiple objects, correctly separating and allocating each instance. Taken together, these observations highlight the robustness and strong fine-grained awareness capability of our model in zero-shot segmentation settings.

S.2.4.4. Bounding box classification

Experimental setting. Bounding box classification evaluates a model's ability to recognize objects within localized regions using only textual descriptions. Following prior work [14, 30], we crop bounding boxes from COCO-val2017 [17], LVIS [11], and Open Images [16] and classify them in a zero-shot manner.

Experimental results. We report Top-1 and Top-5 accuracy in Tab. S.5. These results demonstrate that UNCHA (Ours) achieves consistently superior performance across all datasets, COCO, LVIS, and OpenImages, showing large gains over existing approaches. The improvements are particularly prominent in the Top-1 accuracy, reaching margins as high as 32.89%, which highlights the model's ability to precisely associate localized visual regions with their corresponding textual concepts under zero-shot settings. This suggests that UNCHA (Ours) produces representations that remain stable and discriminative even when object regions are tightly cropped, where contextual cues are minimized.

Table S.2. **Full results of part-level alignment with hard negatives.** Comparison across all settings of part-level alignment with hard negatives for ViT-S and ViT-B. UNCHA (Ours) consistently outperforms prior models, including the challenging 'All Pick5' and 'All-Hard Negs' settings, demonstrating its strong capability in accurately identifying and distinguishing fine-grained visual regions within images.

		All		All Pick5		Base	All
	Model	**SCM**	**Neg**	**SCM**	**Neg**	**Neg**	**Hard Negs**
ViT-S/16	CLIP [23]	39.87	63.60	12.52	23.88	82.41	57.31
	ATMG [24]	40.45	61.51	12.30	22.29	73.15	55.79
	MERU [7]	40.81	**64.18**	12.23	23.81	79.63	56.30
	HyCoCLIP [21]	36.61	60.13	10.85	22.29	80.56	52.03
	UNCHA (Ours)	**41.10**	63.89	**12.88**	**25.04**	**83.33**	**57.45**
ViT-B/16	CLIP [23]	39.22	59.33	13.10	22.94	74.07	52.89
	ATMG [24]	**40.38**	62.08	12.23	23.08	**82.41**	53.91
	MERU [7]	40.09	**62.37**	12.59	20.69	81.48	54.56
	HyCoCLIP [21]	35.96	60.78	11.65	23.52	75.93	53.33
	UNCHA (Ours)	39.58	62.23	**13.53**	**23.81**	80.56	**56.51**

Table S.3. **Multi-object representation performance on ComCo and SimCo (mAP).** UNCHA (Ours) generally outperforms all baselines across object counts and datasets in the extended ViT-S and ViT-B evaluation (Tab. S.3), demonstrating strong fine-grained and compositional understanding in complex multi-object scenes.

		ComCo				SimCo			
		2 obj	**3 obj**	**4 obj**	**5 obj**	**2 obj**	**3 obj**	**4 obj**	**5 obj**
ViT-S/16	CLIP [23]	**69.59**	**71.97**	72.44	72.06	72.49	80.05	82.45	82.65
	MERU [7]	67.42	69.31	70.04	69.60	71.69	78.56	80.65	81.20
	ATMG [24]	44.01	43.94	44.12	43.97	62.17	63.02	61.83	62.00
	HyCoCLIP [21]	64.47	65.67	66.37	65.74	72.91	78.25	79.55	79.43
	UNCHA (Ours)	68.91	71.54	**72.90**	**72.58**	**74.41**	**81.79**	**83.55**	**83.13**
ViT-B/16	CLIP [23]	77.55	80.31	81.41	80.22	77.15	84.58	87.40	88.48
	MERU [7]	72.90	77.25	78.15	77.34	77.82	83.91	85.79	86.90
	ATMG [24]	45.91	45.97	45.80	45.82	65.52	65.32	65.28	65.12
	HyCoCLIP [21]	72.90	73.22	73.51	72.90	75.71	81.13	82.41	82.85
	UNCHA (Ours)	**77.92**	**80.96**	**81.83**	**81.18**	**79.72**	**86.93**	**89.75**	**90.65**

Table S.4. **Zero-shot segmentation performance on VOC21.** UNCHA (Ours) model achieves the highest mIoU on both the ViT-S/16 and ViT-B/16 backbones, showing clear improvements over prior methods. This result demonstrates that our hyperbolic alignment enables the model to effectively capture fine-grained region-level features.

VOC 21 dataset		
Model	**ViT-S/16**	**ViT-B/16**
CLIP	36.02	28.47
MERU	36.18	26.05
AtMG	7.63	6.51
HyCoCLIP	36.79	26.03
UNCHA (Ours)	**39.03**	**32.28**

S.2.5. Analysis

S.2.5.1. Hyperbolic embedding analysis

We conduct several visualization studies on the COCO val2017 dataset [17]. First, Fig. S.1 shows the relative distribution of the embeddings produced by HyCoCLIP and our method, visualized using HoroPCA [4] according to their distance from the origin. HyCoCLIP embeddings lie closer to the origin, whereas ours are positioned farther from the origin in the hyperbolic space. In addition, our embeddings are more widely dispersed, with reduced overlap between part and whole image/text representations. This indicates that our hyperbolic alignment utilizes the available hyperbolic volume more effectively.

In addition, Fig. S.2 presents qualitative examples in which we visualize a subset of COCO part texts and part images using HoroPCA. As shown, the global image con-

Table S.5. **Box-level zero-shot classification accuracy on COCO [17], LVIS [11], and OpenImages [16].** We report Top-1 and Top-5 accuracy. UNCHA (Ours) achieves consistently superior performance across all datasets, showing substantial improvements over CLIP [23], MERU [7], ATMG [24], and HyCoCLIP [21] with Top-1 gains reaching up to 32.89%. These results indicate that our hyperbolic alignment mechanism enables more reliable region-level grounding and captures part-whole semantic structure more faithfully than prior baselines.

		COCO		LVIS		OpenImages	
	Model	**Top-1**	**Top-5**	**Top-1**	**Top-5**	**Top-1**	**Top-5**
ViT-S/16	CLIP	34.98	60.74	5.81	13.97	13.81	35.76
	MERU	43.51	66.77	6.43	15.06	16.51	41.26
	ATMG	19.24	34.85	5.45	13.49	9.72	26.28
	HyCoCLIP	45.36	73.17	11.12	25.28	20.79	47.57
	Ours	**51.57**	**77.11**	**13.65**	**29.03**	**24.36**	**53.26**
ViT-B/16	CLIP	35.22	62.84	6.84	16.16	14.90	38.18
	MERU	44.55	68.10	7.41	16.37	18.14	42.23
	ATMG	21.19	37.61	6.19	14.84	10.52	29.09
	HyCoCLIP	47.88	74.79	12.92	27.31	22.16	48.78
	Ours	**54.14**	**79.03**	**17.17**	**33.21**	**23.81**	**52.53**

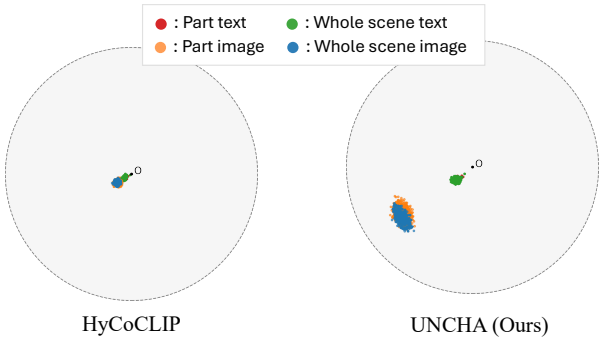

Figure S.1. **Hyperbolic embedding visualization using HoroPCA.** On the COCO dataset [17], we compare the hyperbolic embeddings of our model with those of HyCoCLIP [21]. While HyCoCLIP embeddings are largely concentrated near the origin, ours are distributed farther away, enabling a broader and more effective utilization of the hyperbolic space.

cept "bedroom" and its corresponding text representation reside farther from the origin in the hyperbolic space, while multiple part-level objects distribute across different regions according to their uncertainty. Note that the part text "chair" appears multiple times in the part-text dataset, so we depict its labels as stacked in the visualization. A similar pattern also emerges in the PCA visualization shown in the green box region of Fig. S.2, where several part-text embeddings overlap due to the dataset.

S.2.5.2. Hyperparameter sensitivity analysis

We conduct an analysis on λ_1 and λ_2. Following prior studies on Leaky-ReLU activations [31], we use a small α to preserve sufficient non-linearity while preventing unstable optimization. Results for λ_1 and λ_2 are summarized in Tab. S.6, where all models are trained for 100k iterations. For consistency, we follow the same training protocol and architectural setup as in our main experiments, using the ViT-S configuration. In Tab. S.6, we report both classification (Cls.) and retrieval (Ret.) performance, where each value corresponds to the average over all classification and retrieval tasks, respectively. As shown in the table, our method consistently maintains stable performance across different choices of λ_1 and λ_2, with only minor variations. Notably, our approach tends to achieve either stronger classification or retrieval performance depending on the hyperparameter setting, while avoiding significant degradation in either metric. This demonstrates that our method is robust to the choice of hyperparameters and does not require sensitive tuning to achieve competitive performance.

Table S.6. **Hyperparameter sensitivity analysis at 100k iterations.**Hyperparameter sensitivity analysis of λ_1 and λ_2 at 100k iterations. Each entry reports classification (Cls.) and retrieval (Ret.) performance averaged across all tasks. Our method demonstrates stable performance across a wide range of values, with $\lambda_1 = 0.5$ and $\lambda_2 = 10.0$ selected as the default setting.

λ_1	0.3	0.4	**0.5**	0.6	0.7
	31.9 / 63.6	31.5 / 64.2	31.6 / 63.8	31.5 / 64.2	31.1 / 63.4
λ_2	9.0	9.5	**10.0**	10.5	11.0
	31.3 / 64.2	31.5 / 64.9	31.6 / 63.8	31.5 / 62.9	31.4 / 63.2

S.2.5.3. Role and influence of individual loss terms

We analyze the role of each loss component at 100k iterations. Fig. S.3(a) shows the cosine similarity between gradients of different loss terms. The uncertainty calibration loss exhibits an opposing gradient direction to the entailment loss, acting as a regularizer that prevents representation collapse and stabilizes training. In contrast, the uncertainty-guided contrastive loss remains well aligned with the standard contrastive objective, reinforcing the primary learning signal. Fig. S.3(b) visualizes the embedding distributions on COCO [17] using HoroPCA [4]. In the full model ((b)-1), embeddings are well-structured with clear relationships between scene text (★) and part images (★). Removing the

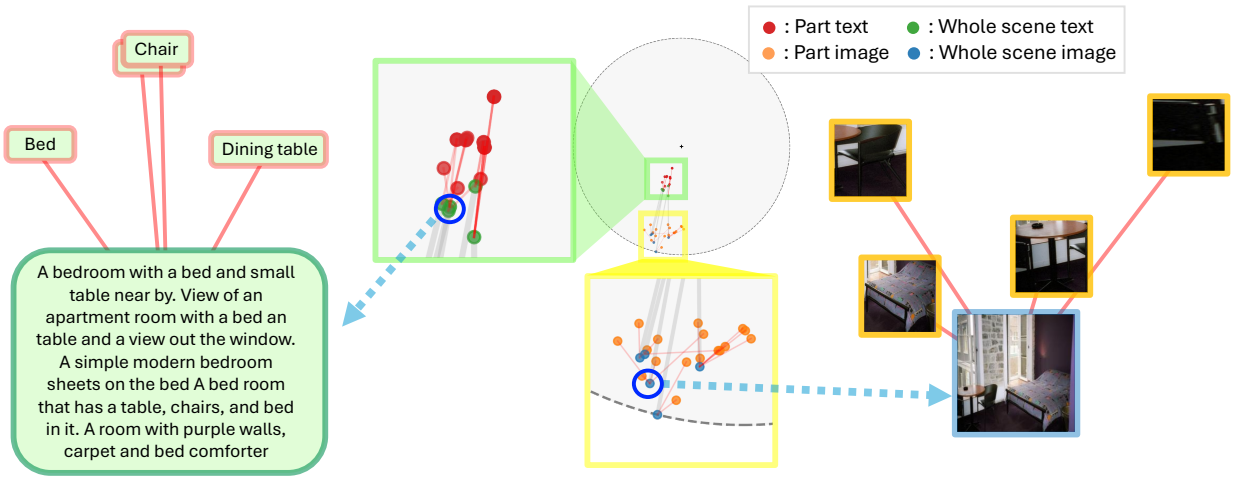

Figure S.2. **Hyperbolic embedding of whole vs. part representations.** Whole-scene images and texts lie deeper in the hyperbolic space, while part-level representations cluster closer to the origin. The zoom-in view and examples illustrate how parts such as chair, bed, and dining table are organized relative to the whole-scene embedding.

uncertainty-guided contrastive loss ((b)-2) weakens this relational alignment, while removing the uncertainty calibration loss ((b)-3) causes the embeddings to concentrate in a narrower region (approximately $0.57R$), reducing representational capacity. Overall, the uncertainty-guided contrastive loss improves relational alignment, whereas the uncertainty calibration loss maintains a well-distributed embedding space and prevents such contraction.

S.2.5.4. Embedding analysis on hyperbolic radius.

In Fig.4, following prior work [21], we first visualize embedding distances using the Euclidean norm. However, this does not fully reflect the geometry of hyperbolic space. To address this, we re-plot the results using the hyperbolic distance from the origin, $d_{\mathbb{L}}(\mathbf{o}, \mathbf{p})$, in Fig. S.4. Due to the exponential expansion of hyperbolic space with radius [2], points farther from the origin lie in regions with significantly larger effective volume. Therefore, analyzing embeddings with $d_{\mathbb{L}}(\mathbf{o}, \mathbf{p})$ provides a more faithful view of their distribution and better captures hierarchical and semantic structure.

S.2.5.5. Hyperbolic distribution during training

To investigate how our hyperbolic alignment organizes part-whole relationships within the hyperbolic space, we visualize the distribution of embedding distances from origin for whole images and their corresponding part-level crops, using both cropped and full images from the ImageNet [6, 25] dataset. As shown in Fig. S.5, as training progresses, part-image distance from the origin decreases (*i.e.*, the uncertainty associated with part images steadily increases), and the separation between the two distributions becomes more pronounced. This pattern indicates that the model gradually enhances its ability to distinguish part-level content from full-scene contexts.

The bottom row of Fig. S.5 reports three statistical distances, Maximum Mean Discrepancy (MMD), Wasserstein-1 distance (W1), and Wasserstein-2 distance (W2), computed at every iteration, quantitatively confirming the growing divergence between the part and whole image distributions. Consistent with the visual trends, all three metrics rise sharply during the early stages of training and gradually stabilize as the model converges. W1 measures the minimum amount of mass that must be transported to align one distribution with the other, reflecting differences in their overall locations. W2 extends this by incorporating squared deviations, making it more sensitive to changes in distributional spread. MMD evaluates the discrepancy between two distributions by comparing their kernel-based mean embeddings, capturing differences in both central tendency and higher-order statistical structure.

S.2.5.6. Dense feature localization visualization

We follow a setting analogous to S.2.4.3 and perform dense localization on the VOC dataset [10] by computing the similarity between text queries and dense features. The resulting visualizations are presented in Fig. S.6 and Fig. S.7. As shown, our method consistently provides the most fine-grained and accurate localization across a diverse set of object classes and input images. Notably, our model is able to correctly highlight objects that competing methods either fail to capture (*e.g.*, person, sofa) or detect with substantially less precision (*e.g.*, dining table, potted plant). These findings demonstrate that our approach achieves a more detailed and robust understanding of complex, multi-object scenes compared to existing baselines. Quantitative results supporting these observations are reported in S.2.4.3.

S.2.5.7. Uncertainty-based ordering of part images

We investigate how well part images are organized within the hyperbolic space by sorting them based on uncertainty and comparing them with HyCoCLIP. Because the Eu-

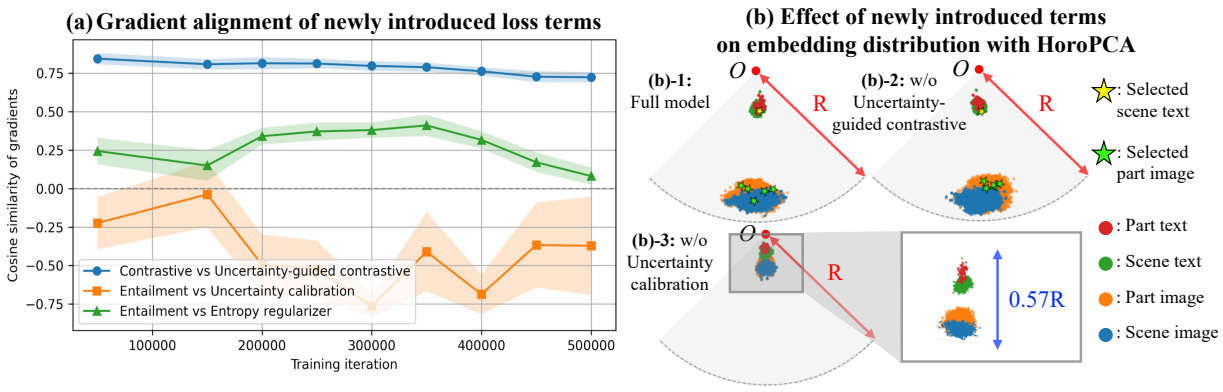

Figure S.3. **Analysis of our newly introduced loss terms.** (a) Cosine similarity between gradients of different loss components, showing that the uncertainty calibration loss acts as a regularizer by opposing the entailment loss, while the uncertainty-guided contrastive loss remains aligned with the main contrastive objective. (b) Visualization of embedding distributions using HoroPCA on COCO, where the full model exhibits well-structured representations, while removing each loss term leads to degraded alignment or concentrated embeddings.

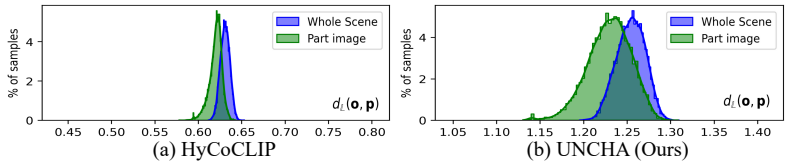

Figure S.4. **Hyperbolic embedding analysis using hyperbolic radius.** Distances are measured by $d_{\mathbb{L}}(\mathbf{o}, \mathbf{p})$ instead of the Euclidean norm to better reflect the intrinsic geometry of hyperbolic space. The results show that embeddings are distributed across different radial regions, corresponding to varying levels of semantic granularity and representational capacity.

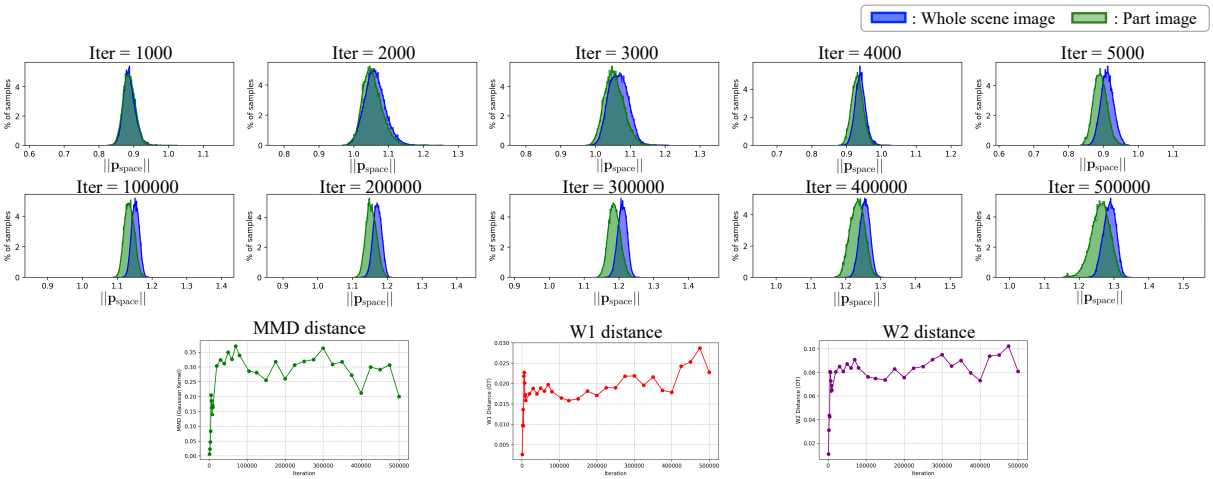

Figure S.5. **Hyperbolic embedding distributions of whole images vs. part images across training iterations.** As training progresses, the uncertainty distributions of whole images and small crops gradually diverge, indicating increasing part–whole separation in the learned hyperbolic space. The bottom row reports iteration-wise distributional distances (MMD, W1, W2), which quantitatively confirm the growing discrepancy between the two distributions.

clidean norm, hyperbolic radius, and uncertainty are monotonic measures (differing only in direction), we sort HyCoCLIP embeddings by their Euclidean norms for a fair comparison with our uncertainty-based ordering. The results are presented in Fig. S.8. As shown, HyCoCLIP produces several misordered cases where abstract or highly-representative samples appear in inconsistent positions, whereas our method yields a more coherent ordering in which part images align naturally according to their scene-level representativeness.

S.2.5.8. Hyperbolic embedding visualization with various dataset

We analyze how part images, part texts, whole scene images, and whole scene texts are distributed within the hyperbolic embedding space by conducting the visualization shown in Fig. S.9. All experiments are performed using our ViT-B model on both the COCO [17] and OpenImages [16]. As illustrated, part-level data consistently occupy regions closer to the origin compared to whole-scene representations, and this trend remains stable across different datasets.

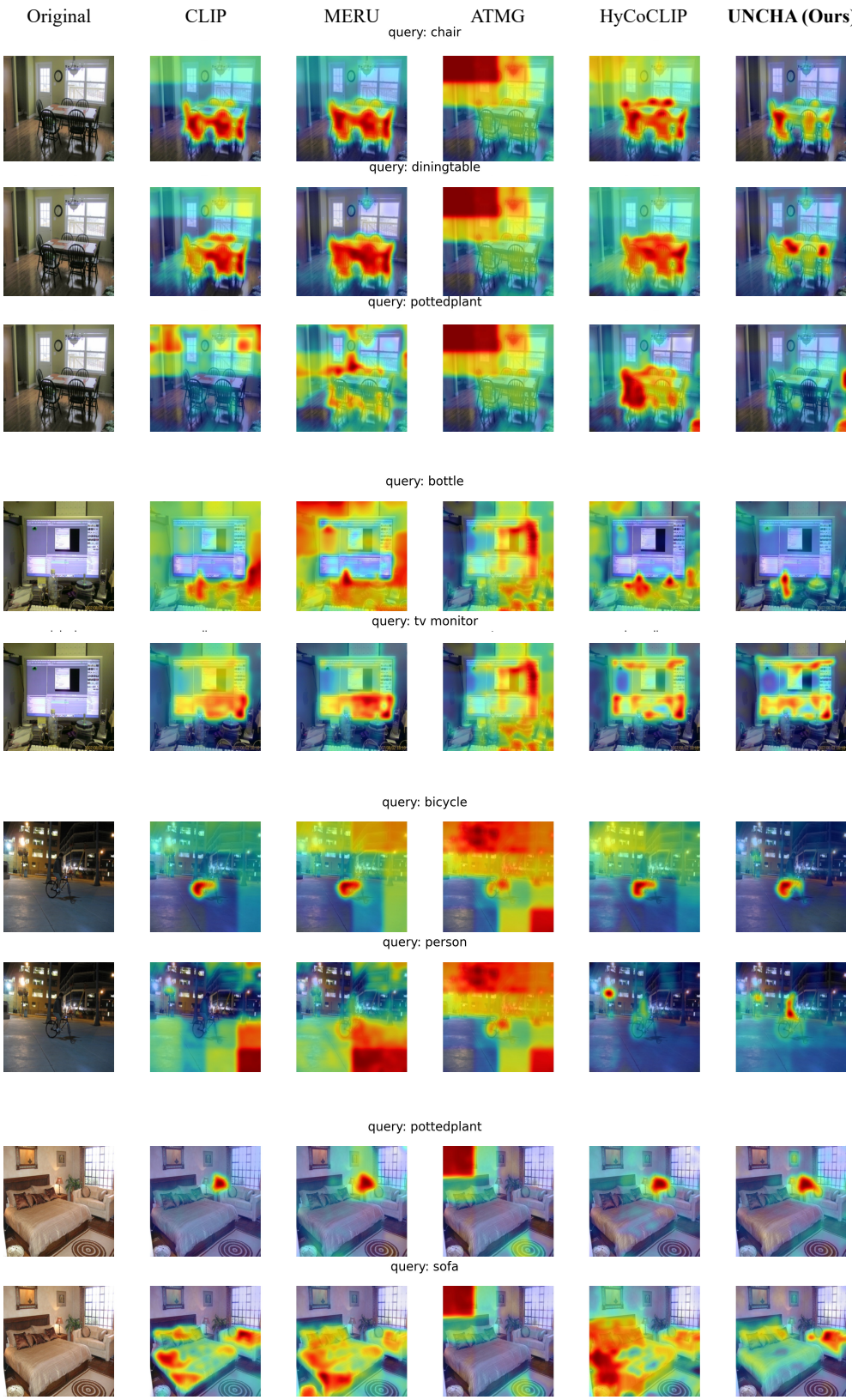

Figure S.6. **Dense feature localization visualizations for zero-shot semantic segmentation.** Following the procedure described in Sec. S.2.4.3, similarity maps on the VOC dataset are generated by extracting dense features and computing their correspondence to text queries. Our method produces sharper and more localized activations that align more accurately with the queried object categories.

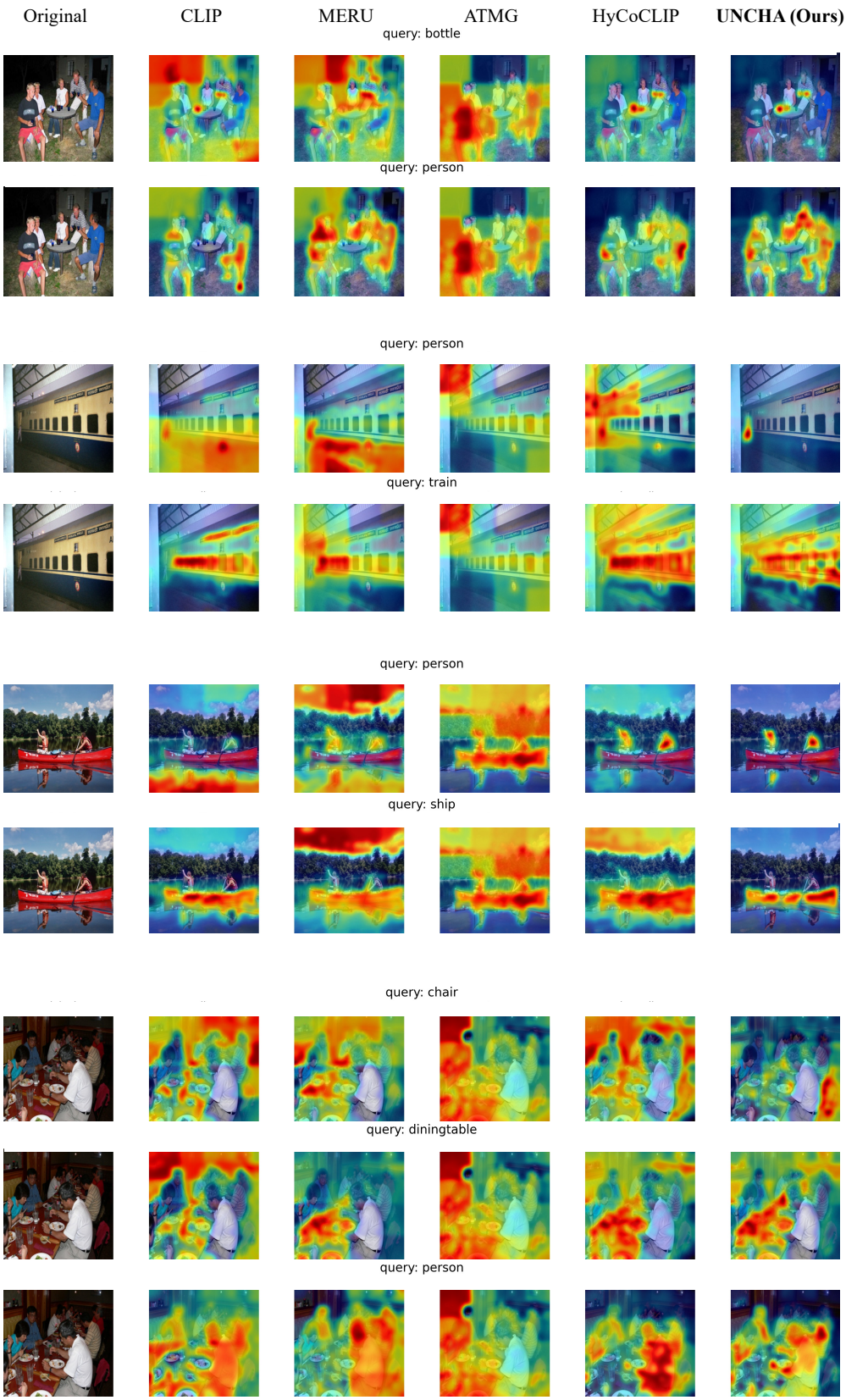

Figure S.7. **Dense feature visualizations for zero-shot semantic segmentation.** Similarity maps on the VOC dataset are generated by extracting dense features and computing their correspondence to text queries, following the procedure described in Sec. S.2.4.3. Our method produces sharper and more localized activations that align more accurately with the queried object categories.

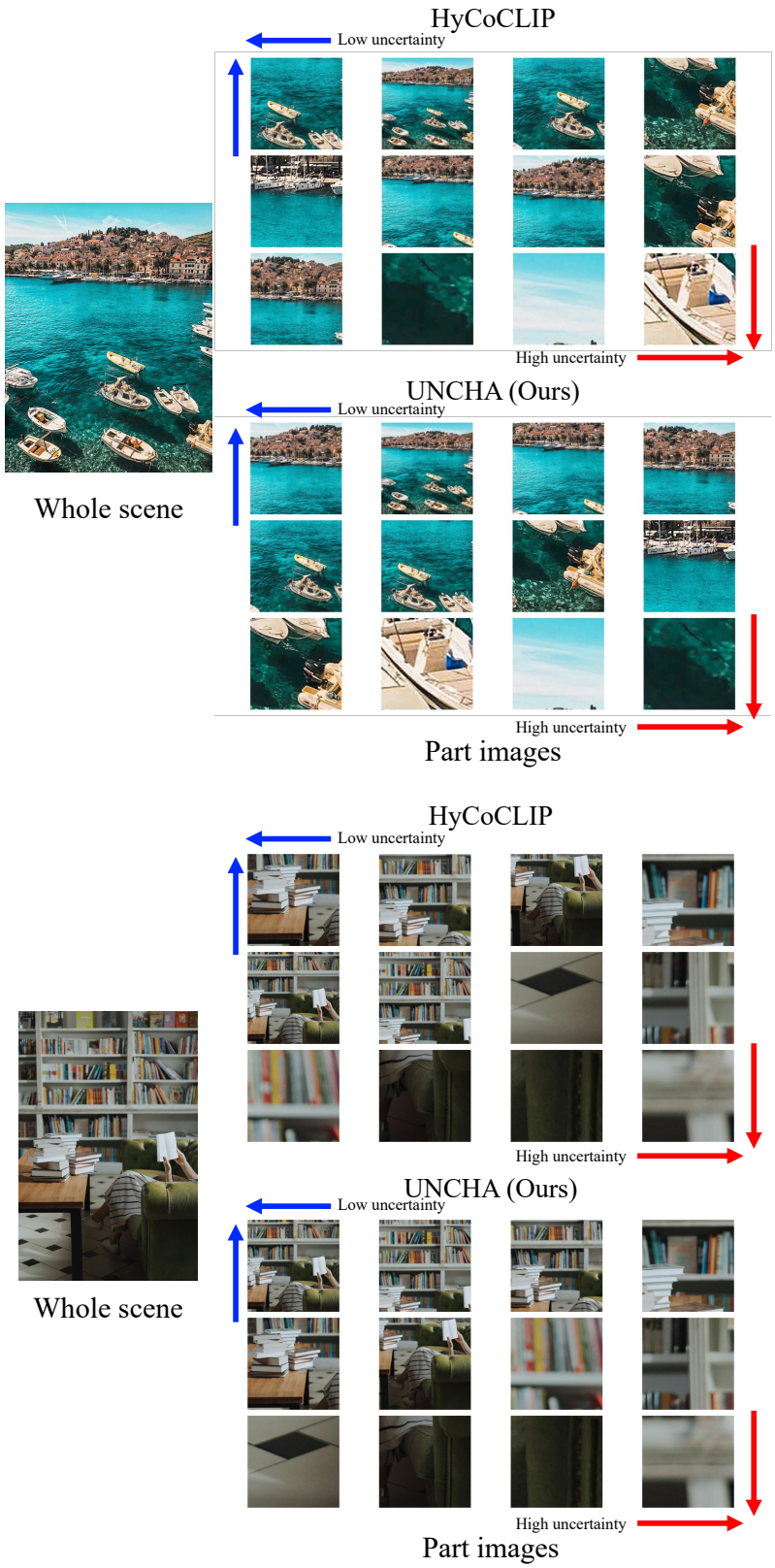

Figure S.8. **Comparison of uncertainty-based ordering of part images.** Comparison of uncertainty-based ordering of part images between HyCoCLIP [21] and UNCHA (Ours) shows that UNCHA produces a coherent ordering in which part images are arranged according to their scene-level representativeness.

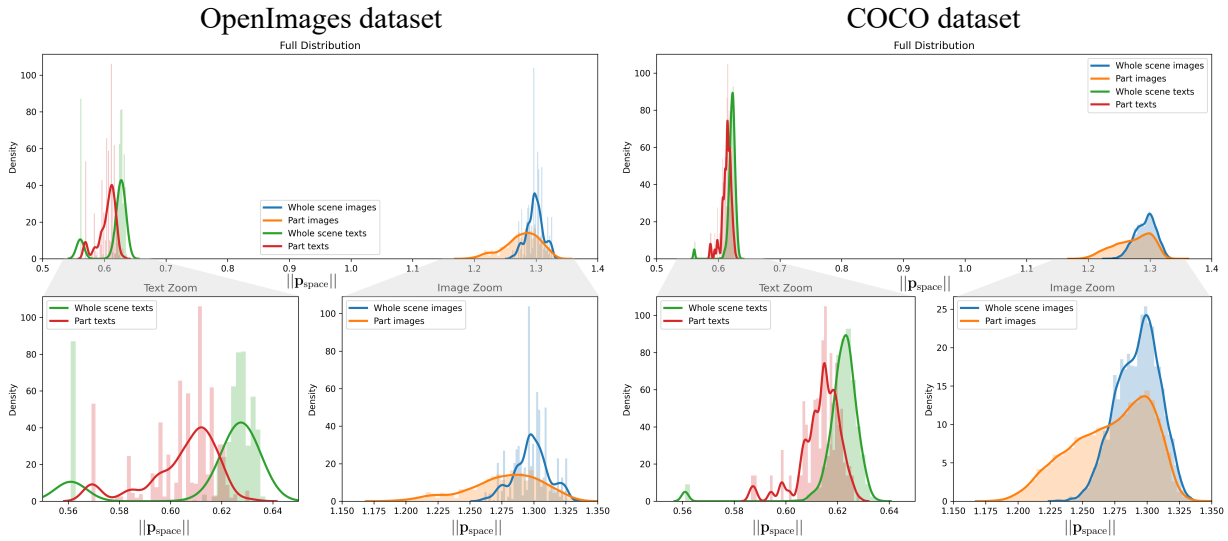

Figure S.9. **Distribution of hyperbolic embeddings across datasets.** Using UNCHA (ViT-B), we visualize part and whole representations from OpenImages [16] and COCO [17] Across both datasets, part-level embeddings appear closer to the origin, while whole-scene embeddings lie farther away, consistently reflecting their hierarchical structure.

References

[1] Reza Abbasi, Ali Nazari, Aminreza Sefid, Mohammadali Banayeeanzade, Mohammad Hossein Rohban, and Mahdieh Soleymani Baghshah. Clip under the microscope: A fine-grained analysis of multi-object representation. In *CVPR*, 2025. 2

[2] Martin R Bridson and André Haefliger. *Metric spaces of non-positive curvature*. Springer Science & Business Media, 2013. 7

[3] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. Dataset. 1

[4] Ines Chami, Albert Gu, Dat P Nguyen, and Christopher Ré. Horopca: Hyperbolic dimensionality reduction via horospherical projections. In *International Conference on Machine Learning*, pages 1419–1429. PMLR, 2021. 5, 6

[5] Minshuo Chen, Yu Bai, Jason D Lee, Tuo Zhao, Huan Wang, Caiming Xiong, and Richard Socher. Towards understanding hierarchical learning: Benefits of neural representations. *NeurIPS*, 2020. 1

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 2, 7

[7] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *ICML*, 2023. 1, 2, 5, 6

[8] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 1

[9] Mohamed Elhoseiny, Babak Saleh, and Ahmed Elgammal. Write a classifier: Zero-shot learning using purely textual descriptions. In *CVPR*, 2013. 1

[10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 2, 7

[11] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 4, 6

[12] Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. Pay attention to your neighbours: Training-free open-vocabulary semantic segmentation. In *WACV*, 2025. 4

[13] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. 1

[14] Dong Jing, Xiaolong He, Yutian Luo, Nanyi Fei, Wei Wei, Huiwen Zhao, Zhiwu Lu, et al. Fineclip: Self-distilled region-based clip for better fine-grained understanding. *NeurIPS*, 2024. 4

[15] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 2

[16] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 4, 6, 8, 12

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2, 4, 5, 6, 8, 12

[18] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *ICLR*, 2017. 1

[19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *ICLR*, 2019. 1

[20] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 1995. 2

[21] Avik Pal, Max van Spengler, Guido Maria D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. Compositional entailment learning for hyperbolic vision-language models. *ICLR*, 2024. 1, 2, 5, 6, 7, 11

[22] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 1

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 5, 6

[24] Sameera Ramasinghe, Violetta Shevchenko, Gil Avraham, and Ajanthan Thalaiyasingam. Accept the modality gap: An exploration in the hyperbolic space. In *CVPR*, 2024. 1, 2, 5, 6

[25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2, 4, 7

[26] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 1

[27] Jack Urbanek, Florian Bordes, Pietro Astolfi, Mary Williamson, Vasu Sharma, and Adriana Romero-Soriano. A picture is worth more than 77 text tokens: Evaluating clip-style models on dense captions. In *CVPR*, 2024. 2

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 1

[29] Feng Wang, Jieru Mei, and Alan Yuille. Sclip: Rethinking self-attention for dense vision-language inference. In *ECCV*, 2024. 4

[30] Chunyu Xie, Bin Wang, Fanjing Kong, Jincheng Li, Dawei Liang, Gengshen Zhang, Dawei Leng, and Yuhui Yin. Fg-clip: Fine-grained visual and textual alignment. *ICML*, 2025. 4

[31] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015. 6

[32] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New

similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2014. 2