GOLP: GLOBAL OBJECT-CENTRIC LEARNING WITH SPATIAL AND CATEGORY PRIORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Grouping objects within an image does not automatically yield object knowledge that is reusable across scenes. This paper introduces GOLP, which uses spatial and category-related visual priors to learn global object representations for cross-scene recognition and compositional scene synthesis. A frozen SAM3 generates candidate masks with fixed dataset-level prompts, organizing object observations through their spatial extent and geometry. A frozen DINOv2 provides regional visual descriptors whose category-related structure guides shared prototype matching. The model retrieves content from a global object bank and separately encodes object states, background, and scene attributes, learning these factors through compositional feature reconstruction. In the second learning stage, the representation module and pretrained RAE are frozen, and only a Transformer adapter is trained for image decoding. Reconstruction, editing, and generation share the same latent variables and decoding pathway, without additional raw-image detail at decoding time. We evaluate cross-scene recognition, object discovery, and identity–state interventions, and provide additional analyses of mask supervision and prompt selection. The results examine how shared object content can be reused across scene conditions.

1 INTRODUCTION

Visual systems must not only describe currently observed objects but also generalize distinct observations into reusable object knowledge. When objects of the same category appear across varying backgrounds, locations, and viewing conditions, an ideal representation preserves their shared content while delegating the specific presentation to independent state variables. This enables the model to recognize objects across changing observations using consistent content and to interpret novel scenes by varying states or object combinations. Global object-centric learning aims for object representations to evolve from mere descriptions of single observations into content units that bridge cross-scene recognition and compositional modeling (Chen et al., 2025b; Kori et al., 2024).

The challenge in learning such representations lies in the fact that correctly decomposing a scene does not automatically determine which content should be shared across scenes. Feature-reconstruction-based object-centric methods leverage pre-trained visual representations to discover objects, elevating learning from pixel-level appearance to a feature space with richer semantic structure (Seitzer et al., 2022). Global modeling approaches further incorporate shared prototypes, object dictionaries, and attribute decomposition, enabling local observations to link to reusable object content (Kori et al., 2024; Chen et al., 2025b). While these advancements underscore the importance of object grouping and global sharing, they also raise a specific learning question: how can we leverage the object and category knowledge already acquired by pre-trained models to directly guide the formation of shared content, rather than treating this knowledge merely as a target to be reconstructed?

Foundation models provide complementary spatial and category-related visual priors. SAM3 extracts candidate object regions from prompts, whereas DINOv2 supplies visual features with transferable, category-related structure (Carion et al., 2026; Oquab et al., 2024). Our central idea is to connect these priors to different parts of global object learning: spatial priors define the evidence associated with an object observation, while regional semantic structure guides which observations share global content. These roles are complementary rather than exclusive, as SAM3 descriptors also provide auxiliary semantic guidance. Regions alone do not establish cross-scene correspon-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

dence, and visual similarity alone does not specify how shared content should be decoded with observation-dependent states. We therefore use the priors as evidence for learning reusable object content rather than as the endpoint of learning.

Building on this idea, we propose GOLP, a global object-centric framework guided by spatial and category-related visual priors. Frozen SAM3 provides candidate regions using fixed dataset-level prompts without image-specific category annotations, and frozen DINOv2 extracts regional visual descriptors. These descriptors directly guide identity assignments to a learnable global object bank, from which shared content is retrieved for compositional decoding. Related observations across scenes can consequently update and reuse the same representations rather than producing independent appearance encodings. Mask geometry, background, and scene attributes are represented by separate latent variables that provide observation-specific conditions for the shared content.

GOLP further links identity matching with compositional reconstruction, enabling global prototypes to serve both category discrimination and object content representation. Matched prototypes, combined with object states and scene attributes, generate object features and soft masks; a background branch independently generates background features and their weights, after which the branches synthesize the complete scene features via spatial weighting. Category-related visual priors guide which observations share representations, while reconstruction objectives constrain the content preserved within these shared representations. This joint learning process organizes visual similarities found in pre-trained features into global object representations suitable for both recognition and conditional decoding. Image decoding occurs in a separate stage; reconstruction, editing, and prior generation share the same set of object, state, background, and scene variables, without introducing raw image details as additional decoding conditions.

We evaluate the framework through cross-scene recognition, object discovery, global-template visualization, and identity–state interventions. These experiments examine whether the learned content can be recognized and reused under different scene conditions. Additional analyses investigate direct mask supervision and prompt selection. Our focus is whether the perceptual evidence supplied by foundation models can be organized into shared object content that supports both recognition and scene composition.

The main contributions of this paper are as follows:

1. We propose a global content learning approach guided by foundation-model priors. Spatial priors determine object observations, while category-relevant regional structures directly guide cross-scene prototype matching, explicitly linking local perception to global sharing.

2. We construct a representation mechanism that synergizes category matching with compositional reconstruction. Through shared prototype retrieval and attribute-branch encoding, category-correlated observations collaboratively learn decodable object content, rather than merely forming local features or category labels.

3. We evaluate shared object content through cross-scene recognition, object discovery, global templates, and state interventions.

2 Related Work

Related work spans compositional object discovery, foundation-model-guided representation learning, and globally shared object content for cross-scene recognition and scene modeling.

Object Discovery through Compositional Scene Modeling. Compositional methods discover objects by reconstructing scenes from object-level representations. IODINE uses iterative variational inference, while Slot Attention groups visual features into exchangeable slots through competitive attention (Greff et al., 2019; Locatello et al., 2020). DINOSAUR reconstructs pretrained features rather than pixels, extending discovery to complex real-world images (Seitzer et al., 2022). FT-DINOSAUR studies input-encoder adaptation and high-resolution transfer with a frozen target encoder (Didolkar et al., 2024). SPOT uses self-training with decoder attention and patch-order permutation (Kakogeorgiou et al., 2024), and MUFASA integrates multi-layer ViT features through mask matching and slot fusion (Bock et al., 2026). SlotAdapt aligns grouping with generative conditioning through slot-conditioned adapters and attention guidance in a frozen pretrained diffusion model (Akan & Yemez, 2025). Beyond within-image grouping, GOLP associates cross-scene observations

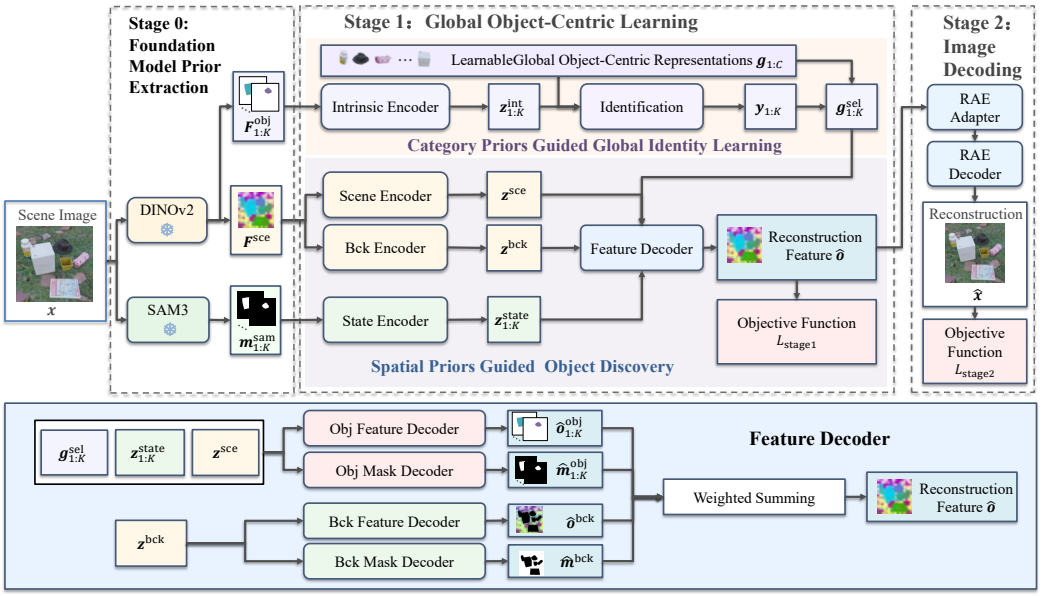


Figure 1: **Overview of GOLP.** Stage 0 extracts spatial and semantic priors using frozen SAM3 and DINOv2. Stage 1 associates object observations with shared global representations and combines the retrieved content with object states, scene attributes, and background for feature reconstruction. Stage 2 trains only a Transformer adapter for decoding through a frozen RAE. **Bottom:** Object and background features are composed using jointly normalized masks.

with global prototypes using regional visual descriptors. **Object Representation Learning with Foundation Model Priors.** DINO and DINOv2 provide transferable visual features with category-related structure, while SAM and SAM3 provide prompt-driven region priors (Caron et al., 2021; Oquab et al., 2024; Kirillov et al., 2023; Carion et al., 2026). COMUS generates semantic segmentation pseudo-labels from salient regions and clustered self-supervised object features (Zadaianchuk et al., 2023). OCCAM isolates and independently encodes externally segmented objects (Rubinstein et al., 2025). Object-centric LeJEPA learns a visual encoder from SAM region proposals using cross-augmentation object alignment, distribution regularization, and instance separation (Geusen & Konukoglu, 2026). GOLP uses spatial priors to define observations and pretrained regional semantics to assign them to shared prototypes, supporting cross-scene identity matching and compositional decoding.

Globally Shared Object Content for Compositional Modeling. Invariant Slot Attention handles spatial transformations through slot-centric reference frames; CoSA's Grounded Slot Dictionary links canonical object properties to parameterized slot priors for object-type specialization and compositional modeling (Biza et al., 2023; Kori et al., 2024). GOCL, GOLD, and CSGO connect global object representations with identity recognition and scene modeling (Chen et al., 2024; 2025b;a), with GOLD jointly inferring grouping, identity, and extrinsic attributes. GOLP instead retrieves reusable content from a shared object bank using foundation-model regions and pretrained regional semantics, combining it with separately encoded object states, background, and scene attributes for compositional reconstruction. A pretrained representation autoencoder (RAE) supplies the image decoder in a separate second training stage (Zheng et al., 2026).

3 METHOD

As shown in Figure 1, GOLP first extracts spatial and semantic priors using frozen foundation models (Stage 0). It then jointly learns global object identities and observation-dependent attributes through compositional feature reconstruction (Stage 1). Finally, a Transformer adapter connects the reconstructed features to a pretrained RAE decoder for image reconstruction (Stage 2). The resulting representations support cross-scene recognition and compositional scene manipulation.

3.1 Problem Setup and Foundation Model Prior Extraction

Given a collection of images $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^{N}$, we aim to learn C object-content representations shared across scenes, together with latent variables describing object states, scene attributes, and background. A global identity denotes a fixed object or a semantic category, depending on the dataset. Training does not use manual category or pixel-level annotations from the target dataset; validation annotations are used for model selection and evaluation.

A frozen SAM3 model produces candidate masks $\{\boldsymbol{m}_k^{\mathrm{sam}}\}_{k\in\mathcal{V}}$ using fixed dataset-level prompts without image-specific category annotations. On GSO, OCTB, and YCB, the generic `object` prompt is followed by generic ROI refinement. On VOC and COCO, we use the fixed prompt set `object`, `person`, `animal`, and `vehicle`; these additional prompts provide dataset-level semantic information. Prompt and post-processing details are given in Appendix A.8. Here, $\mathcal{V} \subseteq \{1, \ldots, K\}$ indexes valid candidates, and K is the maximum number of object slots. The masks define object observations and provide geometry for state encoding.

A frozen DINOv2 encoder E_D extracts the full-scene patch features, regional patch features, and regional CLS descriptors:

$$\boldsymbol{F}^{\mathrm{sce}} = E_D^{\mathrm{patch}}(\boldsymbol{x}), \quad \boldsymbol{F}_k^{\mathrm{obj}} = E_D^{\mathrm{patch}}\left(\mathcal{C}(\boldsymbol{x}, \boldsymbol{m}_k^{\mathrm{sam}})\right), \quad \boldsymbol{f}_k^{\mathrm{cat}} = E_D^{\mathrm{CLS}}\left(\mathcal{C}(\boldsymbol{x}, \boldsymbol{m}_k^{\mathrm{sam}})\right), \tag{1}$$

where $\mathcal{C}$ denotes mask-guided cropping and preprocessing. The intrinsic encoder uses the regional CLS descriptor rather than the patch feature map. The regional CLS descriptor $\boldsymbol{f}_k^{\mathrm{cat}}$ provides category-related visual evidence for identity learning, while $\boldsymbol{F}^{\mathrm{sce}}$ supplies contextual information and the scene reconstruction target.

The background support is defined as $\boldsymbol{m}^{\mathrm{bck}} = \mathbf{1} - \bigvee_{k\in\mathcal{V}} \boldsymbol{m}_k^{\mathrm{sam}}$, where $\bigvee$ denotes the pixel-wise union. It selects the regions of $\boldsymbol{F}^{\mathrm{sce}}$ used for background reconstruction. Both foundation models remain frozen, and the candidate masks are fixed throughout representation learning.

3.2 Category-Prior-Guided Global Object Identity Learning

An intrinsic encoder maps the category-related descriptor of each object to a local intrinsic representation:

$$\boldsymbol{z}_k^{\mathrm{int}} = E_{\mathrm{int}}\left(\boldsymbol{f}_k^{\mathrm{cat}}\right). \tag{2}$$

We associate these observations with a learnable global object-centric representation bank $\boldsymbol{G} \in \mathbb{R}^{C\times d}$, whose rows $\{\boldsymbol{g}_c\}_{c=1}^{C}$ represent shared object content. The identification module computes normalized similarities:

$$\ell_{kc} = 1/\tau_a \cdot \left\langle \mathrm{norm}(\boldsymbol{z}_k^{\mathrm{int}}), \mathrm{norm}(\boldsymbol{g}_c) \right\rangle, \tag{3}$$

where $\mathrm{norm}(\boldsymbol{v}) = \boldsymbol{v}/\max(\|\boldsymbol{v}\|_2, \epsilon)$ and τ_a is the matching temperature.

We obtain the assignment distribution $\boldsymbol{y}_k$ using Gumbel–Softmax with temperature τ_g during training and deterministic softmax during inference. The predicted identity and shared-content readout are

$$\hat{c}_k = \arg\max_c y_{kc}, \qquad \boldsymbol{g}_k^{\mathrm{sel}} = \sum\nolimits_{c=1}^{C} y_{kc}\boldsymbol{g}_c. \tag{4}$$

The identity $\hat{c}_k$ is used for recognition, while $\boldsymbol{g}_k^{\mathrm{sel}}$ conditions the subsequent feature decoder. Thus, the local intrinsic representation determines which global content to retrieve but does not directly enter object decoding. Observations associated with the same identity jointly update and reuse the corresponding global representation.

We additionally learn a categorical assignment prior $\boldsymbol{\pi} = \mathrm{Softmax}(\boldsymbol{a})$. This prior regularizes representation usage and provides the identity distribution for generation; it is not added to the matching logits.

3.3 Spatial-Prior-Guided Object Discovery

To explain observation-dependent variation, we encode object states separately from global content. The state encoder receives the object mask, whereas the scene and background encoders both receive the full-scene feature map:

$$\left(\boldsymbol{\mu}_k^{\mathrm{state}}, \boldsymbol{\ell}_k^{\mathrm{state}}\right) = E_{\mathrm{state}}\left(\boldsymbol{m}_k^{\mathrm{sam}}\right), \left(\boldsymbol{\mu}^{\mathrm{sce}}, \boldsymbol{\ell}^{\mathrm{sce}}\right) = E_{\mathrm{sce}}\left(\boldsymbol{F}^{\mathrm{sce}}\right), \left(\boldsymbol{\mu}^{\mathrm{bck}}, \boldsymbol{\ell}^{\mathrm{bck}}\right) = E_{\mathrm{bck}}\left(\boldsymbol{F}^{\mathrm{sce}}\right). \tag{5}$$

The scene and background encoders share a convolutional stem and use separate output branches. Each pair $(\boldsymbol{\mu}, \boldsymbol{\ell})$ parameterizes a diagonal Gaussian posterior with mean $\boldsymbol{\mu}$ and log-variance $\boldsymbol{\ell}$. Training uses reparameterized samples $\boldsymbol{z} = \boldsymbol{\mu} + \exp(\boldsymbol{\ell}/2) \odot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$; deterministic reconstruction uses posterior means (Kingma & Welling, 2014).

The selected global object-centric representation, object state, and scene attributes jointly condition the object feature and mask decoders:

$$\begin{aligned} \hat{\boldsymbol{o}}_k^{\mathrm{obj}} &= D_f^{\mathrm{obj}}\left(\boldsymbol{g}_k^{\mathrm{sel}}, \boldsymbol{z}_k^{\mathrm{state}}, \boldsymbol{z}^{\mathrm{sce}}\right), & \boldsymbol{s}_k^{\mathrm{obj}} &= D_m^{\mathrm{obj}}\left(\boldsymbol{g}_k^{\mathrm{sel}}, \boldsymbol{z}_k^{\mathrm{state}}, \boldsymbol{z}^{\mathrm{sce}}\right), \\ \hat{\boldsymbol{o}}^{\mathrm{bck}} &= D_f^{\mathrm{bck}}\left(\boldsymbol{z}^{\mathrm{bck}}\right), & \boldsymbol{s}^{\mathrm{bck}} &= D_m^{\mathrm{bck}}\left(\boldsymbol{z}^{\mathrm{bck}}\right). \end{aligned} \tag{6}$$

The feature decoders produce spatial feature maps, while the mask decoders produce logits. Following GOLD, feature decoding combines projected latent representations with positional embeddings and a position-wise MLP. Mask decoding progressively upsamples a coordinate-augmented latent canvas to the image resolution.

Valid objects and the background compete through a joint softmax. Writing b for the background index, we obtain

$$\hat{m}_j(p) = \frac{\exp(s_j(p)/\tau_\alpha)}{\sum_{i \in \mathcal{V} \cup \{b\}} \exp(s_i(p)/\tau_\alpha)}, \qquad j \in \mathcal{V} \cup \{b\}. \tag{7}$$

Invalid slots are excluded before normalization. The reconstructed masks are area-pooled to the feature resolution and used to compose the scene:

$$\tilde{\boldsymbol{m}}_j = \mathrm{AreaPool}\left(\hat{\boldsymbol{m}}_j\right), \quad \hat{\boldsymbol{o}} = \sum_{k \in \mathcal{V}} \tilde{\boldsymbol{m}}_k^{\mathrm{obj}} \odot \hat{\boldsymbol{o}}_k^{\mathrm{obj}} + \tilde{\boldsymbol{m}}^{\mathrm{bck}} \odot \hat{\boldsymbol{o}}^{\mathrm{bck}}. \tag{8}$$

Spatial priors define the candidate observations, whereas the reconstructed masks determine how each object and the background contribute to the scene. Compositional reconstruction therefore couples object discovery with global object-centric learning.

3.4 Image Decoding

We reconstruct RGB images from $\hat{\boldsymbol{o}}$ using a pretrained representation autoencoder (RAE). Its encoder uses DINOv2 with registers, whereas Stage 0 uses a register-free backbone (Darcet et al., 2024; Zheng et al., 2026). A residual Transformer adapter aligns the reconstructed features with the RAE representation space:

$$A_\eta(\boldsymbol{f}) = \boldsymbol{f} + \Delta_\eta(\boldsymbol{f}), \qquad \hat{\boldsymbol{x}} = \mathcal{R}\left(D_R\left(A_\eta(\hat{\boldsymbol{o}})\right)\right), \tag{9}$$

where D_R is the frozen RAE decoder and $\mathcal{R}$ resizes its output to the target image resolution. Δ_η is a three-layer pre-norm Transformer with a zero-initialized final projection.

The adapter is trained through full-scene reconstruction. The same mapping can also decode individual reconstructed object feature maps for visualization, without a separately trained object-image decoder.

3.5 Model Training

Stage 0 requires no optimization. We first train the global object-centric representation module, followed by the image adapter with the representation module frozen.

Stage 1: Global object-centric learning. We jointly optimize the intrinsic and attribute encoders, the global object-centric representation bank, the categorical assignment prior, and the compositional decoders. The objective combines global representation learning, object discovery, and feature reconstruction:

$$L_{\mathrm{stage1}} = L_{\mathrm{global}} + L_{\mathrm{discov}} + L_{\mathrm{recon}}. \tag{10}$$

L_{global} aligns local intrinsic representations with their assigned global representations and regularizes identity assignments. L_{discov} regularizes the continuous attribute posteriors and the spatial support of object features. L_{recon} supervises the composed scene features and background features using frozen DINOv2 representations. Cached SAM3 descriptors additionally provide semantic and assignment guidance. Full losses and their dataset-specific weights are given in Appendix A.2.

Table 1: The accuracy (ACC ↑) of cross-scene object recognition on GSO, OCTB and YCB datasets. † denotes a global K-means readout applied to object-level features. Other methods use their native identity predictions. The best result in each row is shown in bold.

Dataset	GOLP	CSGO	GOLD	CoSA†	MUFASA†
GSO	**0.871**	0.644	0.560	0.204	0.163
OCTB	**0.517**	0.106	0.105	0.157	0.104
YCB	**0.506**	0.149	0.179	0.214	0.138

Stage 2: Image decoding. We freeze all Stage 1 modules and the RAE encoder and decoder, optimizing only the Transformer adapter. The objective combines latent alignment, distribution matching, and image reconstruction:

$$L_{\text{stage2}} = \lambda_{\text{align}} L_{\text{align}} + \lambda_{\text{distr}} L_{\text{distr}} + \lambda_{\text{image}} L_{\text{image}}. \tag{11}$$

L_{align} aligns predicted and target latents in value and direction. L_{distr} matches channel-wise means, standard deviations, and channel covariance estimated over batch and spatial tokens. L_{image} measures pixel-wise reconstruction error and perceptual similarity (Zhang et al., 2018). Gradients propagate through the frozen RAE decoder to update the adapter. The latent-input curriculum is defined in Appendix A.2; validation and inference use reconstructed GOLP features only.

4 Experiments

We evaluate GOLP on one synthetic benchmark and four real-world datasets to comprehensively assess whether the learned global object representations support cross-scene recognition, object attribute disentanglement, compositional scene generation, and controllable scene editing.

Datasets. GSO, OCTB, and YCB are used for cross-scene recognition and object decomposition; COCO and VOC support grouping in natural images, without physical-instance re-identification where correspondence labels are unavailable. GSO uses scanned assets (Downs et al., 2022), and OCTB refers to the OCTScenes subset used here (Huang et al., 2023). YCB sources and splits follow the experiment files. COCO and VOC semantic and instance labels are treated separately (Lin et al., 2014; Everingham et al., 2010).

Comparison Methods. We compare GOLP with CSGO, GOLD, CoSA, and MUFASA, which adopt compositional modeling for object-centric representation learning. GOLD and CSGO explicitly learn global object representations, while CoSA conditions slot inference on a shared dictionary of canonical object properties and slot priors. GOLP differs in using foundation-model regions and regional semantic descriptors to guide shared-content retrieval together with separately encoded observation-dependent attributes.

Evaluations. Cross-scene recognition on GSO, YCB, and OCTB is measured by ACC after foreground-mask Hungarian matching and dataset-level identity-to-category alignment; only objects with valid category labels are counted. GOLP, GOLD, and CSGO use native identity predictions, while CoSA and MUFASA use global K-means with the number of clusters set to the category count. For decomposition, ARI-O measures foreground grouping and ARI-A includes background. Region overlap is reported as mBO^c/mIoU^c on GSO, OCTB, and YCB, mBO^i/mIoU^i on COCO, and connected-component proxy scores on VOC. Evaluation partitions and metric definitions are specified in Appendix A.5.

4.1 Cross-Scene Object Recognition

We evaluate whether the learned global object representations support recognition across changes in background, position, scale, and viewing conditions. GOLP is compared with GOLD, CSGO, CoSA, and MUFASA on GSO, OCTB, and YCB using the recognition protocol described above.

As shown in Table 1, GOLP substantially outperforms the evaluated baselines across all three datasets. Its advantage over GOLD and CSGO is consistent with the complementary use of spatial and category-related visual priors: SAM3 defines object-specific observations, while DINOv2

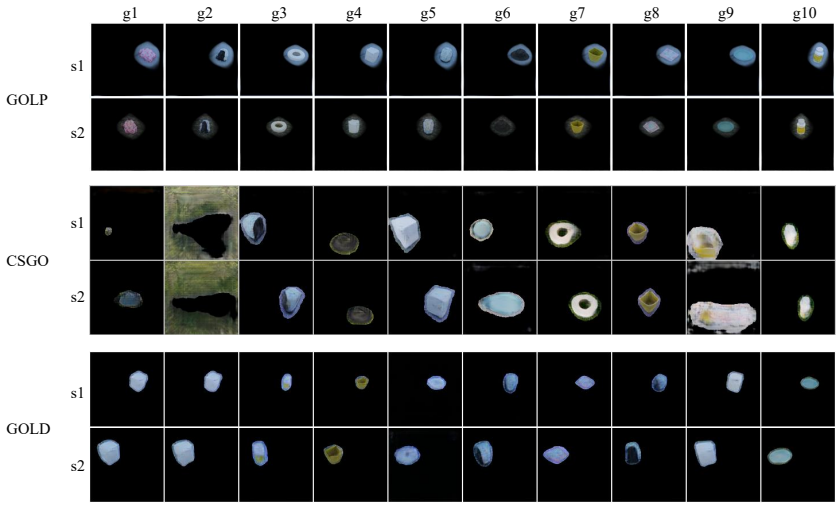


Figure 2: Visualization comparing the global object-centric representations learned by GOLP, CSGO, and GOLD on the GSO dataset. $g_1, \ldots, g_{10}$ denote the learned global object representations, while s_1 and s_2 denote two distinct object-state representations.

regional descriptors guide their association with shared global content. This supplies localized semantic evidence for identity learning and may reduce interference from scene-dependent context. The advantage remains clear on OCTB and YCB. CoSA and MUFASA, evaluated with clustering-based identity readouts, also obtain lower accuracy. On the two real-world datasets, GOLD and CSGO no longer retain the recognition advantage over these clustering readouts observed on GSO. Together with the decomposition results, this suggests that the quality of object observations and their global associations is important for recognition. These comparisons support the effectiveness of the complete GOLP pipeline, without isolating the causal contribution of each prior.

4.2 Global Object-Centric Representation Learning

We visualize the global object representations learned by GOLP, GOLD, and CSGO on GSO under two object-state conditions (Figure 2). Each column decodes a global representation, and each row uses a fixed state code. This comparison examines the content associated with different global representations and its response to changing state conditions.

Figure 2 shows localized object templates decoded from GOLP's shared representations under both state conditions. The templates have broadly consistent positions and scales within each fixed-state row, while different global representations produce different object appearances. CSGO exhibits visible background leakage in some columns, and the baseline templates show larger variations in apparent object extent. These qualitative observations are consistent with the intended separation between shared content and spatial state. The visualizations reflect both the learned representations and their image decoders, and therefore do not isolate representation quality from decoding quality. We further examine content–state separation through direct attribute swaps in Appendix A.3.

4.3 Compositional Scene Editing

This experiment aims to evaluate whether the learned object representations can support controllable scene editing by manipulating object-level latent variables. Since GOLP represents a scene as a combination of object content, object states, background, and scene attributes, editing can be performed by modifying specific latent variables without requiring access to raw image details. We present visualizations of scene editing using GOLP on the GSO and OCTB datasets, demonstrating the removal, addition, and movement of objects within the scenes.

As shown in Figure 3, GOLP supports object removal, insertion, and movement on both GSO and OCTB. Removing one or two objects largely preserves the remaining scene content, while the vacated regions are filled with visually plausible background structure. Similarly, new objects can be

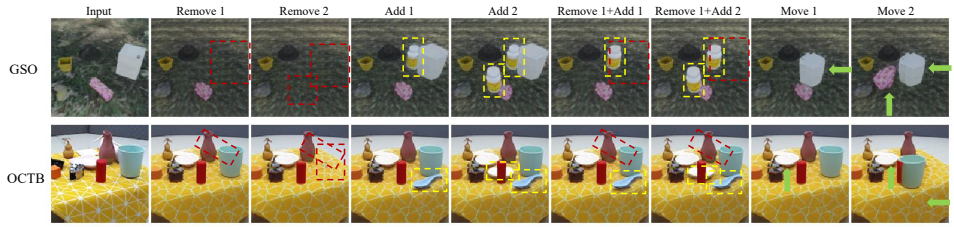


Figure 3: Visualization of scene editing by GOLP on the GSO and OCTB datasets. Red dashed boxes indicate objects to be removed, and yellow dashed boxes indicate objects to be added; the number of boxes corresponds to the number of objects removed or added. Green arrows indicate the direction of object movement.

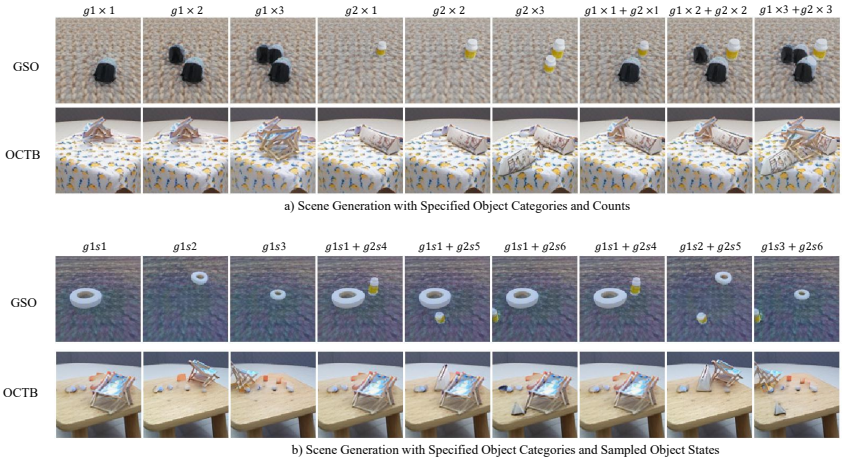


Figure 4: Visualization of scene image generation by GOLP on the GSO and OCTB datasets. $g_c \times n$ denotes a scene containing n objects generated from global representation g_c. The $+$ operator combines two specified object groups in the same scene, with their counts summed.

inserted without substantially altering surrounding objects. The combined removal and insertion results further demonstrate that multiple editing operations can be composed within a single scene. In the movement examples, the selected objects change position while retaining recognizable appearance, consistent with the separation of object content from spatial state. These editing capabilities also extend to the cluttered OCTB scene, where nearby objects and the overall tabletop structure remain largely intact after manipulation. Together, these qualitative results indicate that GOLP's global object representations support localized, compositional editing beyond reconstructing the original scene configuration.

4.4 Compositional Scene Generation

We evaluate compositional scene generation on GSO and OCTB by specifying object categories and counts and by varying object-state codes. The illustrated examples examine object-level changes within a consistent scene context. Full-prior and reference-conditioned generation use the common latent interface described in Appendix A.4; the figure does not require every scene factor to be resampled for each image.

As shown in Figure 4, GOLP supports compositional scene generation with specified object categories, counts, and states on GSO and OCTB. In Figure 4(a), reusing a global representation produces multiple instances of the corresponding category, while combining different representations generates scenes containing both categories. The objects retain recognizable category-specific appearance and shape across the illustrated count configurations, with the background remaining largely consistent. These results suggest that the learned global representations serve as reusable content units that support varying object multiplicities and category combinations. Figure 4(b) further examines state-dependent generation with the global object representations fixed. Different state codes produce variations in object size, position, and orientation while largely preserving category identity. In the middle group, changing only the second object's state largely preserves the

Table 2: **Compositional object discovery on GSO, OCTB, and YCB.** Scores are multiplied by 100; higher is better. The superscript c denotes overlap with category-merged ground-truth regions, without semantic label prediction. Best results within each dataset are shown in bold.

Dataset	Method	ARI-A ↑	ARI-O ↑	mBO^c ↑	mIoU^c ↑
GSO	CSGO	**71.55**	87.94	**55.70**	**55.37**
	GOLD	68.77	77.82	50.79	50.43
	CoSA	5.73	54.41	20.06	19.19
	MUFASA	22.80	33.46	22.99	20.50
	GOLP (ours)	52.45	**93.67**	45.10	44.68
OCTB	CSGO	56.41	46.08	37.89	37.30
	GOLD	54.96	43.18	31.65	30.81
	CoSA	6.95	62.55	28.35	27.58
	MUFASA	15.01	22.86	18.06	15.76
	GOLP (ours)	**64.31**	**80.79**	**46.92**	**45.95**
YCB	CSGO	0.00	0.00	4.92	1.98
	GOLD	0.00	0.00	4.92	1.98
	CoSA	4.11	38.56	20.07	18.05
	MUFASA	18.27	22.16	19.92	18.12
	GOLP (ours)	**58.28**	**62.76**	**37.73**	**36.04**

first object's configuration; in the final group, varying both objects' states changes their joint spatial arrangement. These observations support the separation of shared object content from observation-dependent spatial attributes. Together, the results provide qualitative evidence that GOLP can reuse the same global representations across different object combinations and spatial configurations.

4.5 Compositional Object Discovery

We evaluate whether GOLP's compositional representations support object discovery by decomposing scenes into object regions and background. We compare GOLP with CSGO, GOLD, CoSA, and MUFASA on GSO, OCTB, and YCB. ARI-A and ARI-O measure grouping over the full image and ground-truth foreground, respectively, while mBO^c and mIoU^c evaluate agreement with visible object regions. For GOLP, we evaluate the masks produced by its learned compositional decoder rather than the input SAM3 masks. All methods retain their native input preprocessing and trained slot capacities.

Table 2 shows that GOLP achieves the highest ARI-O across all three datasets and outperforms all evaluated baselines on every reported metric on OCTB and YCB. Its ARI-O exceeds the strongest baseline by 18.24 and 24.20 points on these two datasets, respectively. The accompanying gains in ARI-A and region overlap indicate that the improvements extend beyond foreground grouping to full-scene partitioning and spatial coverage of visible objects. On YCB, prediction diagnostics reveal that the evaluated GOLD and CSGO checkpoints predominantly produce all-background outputs, explaining their near-zero grouping scores. GOLP also outperforms CoSA and MUFASA, whose predictions do not exhibit this single-region collapse. Its advantage therefore is not limited to comparisons against collapsed outputs, but also includes better grouping and region overlap than the other evaluated decompositions. On GSO, however, the highest ARI-O does not translate into the best performance on all metrics: GOLP remains below GOLD and CSGO in ARI-A, mBO^c, and mIoU^c. This distinction highlights that consistent grouping of foreground pixels does not necessarily imply accurate object extent or full-image partitioning. Overall, the results support GOLP's effectiveness in compositional object discovery, particularly on OCTB and YCB, while identifying region coverage and full-scene mask agreement as remaining limitations on GSO.

5 Conclusion

GOLP organizes object observations with spatial priors and learns cross-scene shared content from category-relevant visual priors. Prototype retrieval, factorized attributes, and compositional reconstruction link recognition with scene interpretation. A unified latent-variable and decoding interface supports encoding-reconstruction, factor-specific editing, and composition from attribute priors without bypassing the representation with raw image detail. The contribution is not another segmentation or generation module, but a way to organize the perceptual evidence available in foundation models into reusable object content.

AI USE STATEMENT

We used generative AI tools for framework discussions, feedback on research methodology, parameter suggestions, brainstorming, and editing manuscript text and software code. The reported measurements were supplied by the authors. We reviewed the AI-assisted material and take responsibility for the accuracy and integrity of the final manuscript.

ETHICS STATEMENT

This work uses existing image datasets (GSO, OCTB, YCB, VOC2012, and COCO2017) and pretrained vision models; it does not recruit human participants or collect new personal data. The study concerns object-centric representation learning and image reconstruction, not identification of individuals or inference of sensitive personal attributes. The datasets and pretrained models are used under their applicable access terms and licenses.

REFERENCES

Ali Kemal Akan and Yucel Yemez. Slot-guided adaptation of pre-trained diffusion models for object-centric learning and compositional generation. In *International Conference on Learning Representations*, 2025.

Ondrej Biza, Sjoerd Van Steenkiste, Mehdi S. M. Sajjadi, Gamaleldin Fathy Elsayed, Aravindh Mahendran, and Thomas Kipf. Invariant slot attention: Object discovery with slot-centric reference frames. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 2507–2527. PMLR, 2023.

Sebastian Bock, Leonie Schüßler, Krishnakant Singh, Simone Schaub-Meyer, and Stefan Roth. Mufasa: A multi-layer framework for slot attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27750–27760, June 2026.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International conference on learning representations*, volume 2026, pp. 138846–138923, 2026.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Tonglin Chen, Zhimeng Shen, Bin Li, and Xiangyang Xue. Compositional scene modeling with global object-centric representations. *Machine Learning*, 113(6):3505–3524, 2024.

Tonglin Chen, Yinxuan Huang, Jinghao Huang, Bin Li, and Xiangyang Xue. Unsupervised learning of global object-centric representations for compositional scene understanding. *IEEE Transactions on Visualization and Computer Graphics*, 31(10):8385–8396, 2025a. doi: 10.1109/TVCG.2025.3570426.

Tonglin Chen, Yinxuan Huang, Zhimeng Shen, Jinghao Huang, Bin Li, and Xiangyang Xue. Learning global object-centric representations via disentangled slot attention. *Machine Learning*, 114 (2):40, 2025b.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=2dnO3LLiJ1`.

Aniket Didolkar, Andrii Zadaianchuk, Anirudh Goyal, Mike Mozer, Yoshua Bengio, Georg Martius, and Maximilian Seitzer. Zero-shot object-centric representation learning. *arXiv preprint arXiv:2408.09162*, 2024.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Benjamin Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *International Conference on Robotics and Automation*, 2022.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88 (2):303–338, 2010.

Jakob Geusen and Ender Konukoglu. Object-centric lejepa. *arXiv preprint arXiv:2607.02404*, 2026.

Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *International conference on machine learning*, pp. 2424–2433. PMLR, 2019.

Yinxuan Huang, Tonglin Chen, Zhimeng Shen, Jinghao Huang, Bin Li, and Xiangyang Xue. Octscenes: A versatile real-world dataset of tabletop scenes for object-centric learning. *arXiv preprint arXiv:2306.09682*, 2023.

Ioannis Kakogeorgiou, Spyros Gidaris, Konstantinos Karantzalos, and Nikos Komodakis. Spot: Self-training with patch-order permutation for object-centric learning with autoregressive transformers. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22776–22786. IEEE, 2024.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014. URL https://openreview.net/forum?id=33X9fd2-9FyZd.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003. IEEE, 2023.

Avinash Kori, Francesco Locatello, Fabio De Sousa Ribeiro, Francesca Toni, and Ben Glocker. Grounded object-centric learning. In *International Conference on Learning Representations*, volume 2024, pp. 42559–42603, 2024.

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 *European Conference on Computer Vision*, 2014.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Alexander Rubinstein, Ameya Prabhu, Matthias Bethge, and Seong Joon Oh. Are we done with object-centric learning? *arXiv preprint arXiv:2504.07092*, 2025.

Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. *arXiv preprint arXiv:2209.14860*, 2022.

Andrii Zadaianchuk, Matthaeus Kleindessner, Yi Zhu, Francesco Locatello, and Thomas Brox. Unsupervised semantic segmentation with self-supervised object-centric representations. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1_jFneF07YC.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018. doi: 10.1109/CVPR.2018.00068.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0u1LigJaab.

A Appendix

A.1 Limitations and Future Directions

GOLP has two main limitations. First, its object segmentation performance depends on the quality of the candidate masks provided by SAM3. Fixed prompts can yield incomplete or redundant regions in complex scenes; additional dataset-level prompts improve coverage on COCO and VOC but do not eliminate this limitation. Second, the learned global representations do not fully preserve fine-grained appearance details, which limits the fidelity of reconstructed and generated scenes. Future work will explore more effective integration of spatial and semantic priors for compositional object discovery, and the representation of object textures alongside shared content to improve image fidelity while preserving cross-scene recognition.

A.2 Training Objectives and Loss Terms

This appendix gives the component losses used in Section 3.5. Let B denote batch size, K the number of object slots per image, C the number of global prototypes, P_f the number of feature locations, P_m the number of mask pixels, and D the feature dimension. Unless stated otherwise, prototype, assignment-balancing, and categorical-prior losses are averaged over all BK slots, including padded slots. This reduction differs from the valid-slot selection used for scene composition; confidence-weighted and valid-slot terms are specified separately below.

Stage 1 combines three groups of losses: global object-centric representation learning, object discovery, and feature reconstruction. The following subsections specify the component losses.

A.2.1 Global Object-Centric Representation Learning

The global objective combines local prototype alignment, categorical assignment regularization, and auxiliary guidance:

$$\begin{aligned} L_{\text{global}} &= \lambda_{\text{proto}} L_{\text{proto}} + \lambda_{\text{kl},p} L_{\text{kl},p} + L^{\text{aux}}_{\text{global}}, \\ L_{\text{proto}} &= \frac{1}{BK} \sum_{b=1}^{B} \sum_{k=1}^{K} \sum_{c=1}^{C} y_{bkc} \left\| \boldsymbol{z}^{\text{int}}_{bk} - \boldsymbol{g}_c \right\|_2^2, \\ L_{\text{kl},p} &= \frac{1}{BK} \sum_{b=1}^{B} \sum_{k=1}^{K} D_{\text{KL}}(\boldsymbol{y}_{bk} \| \boldsymbol{\pi}) \,. \end{aligned} \tag{12}$$

For slot i, let $\boldsymbol{t}_i$ and s_i denote its cached SAM3 semantic descriptor and confidence. We use the confidence-weighted average and safe vector normalization

$$\mathcal{W}_s[f_i] = \frac{\sum_i s_i f_i}{\max(\sum_i s_i, 1)}, \qquad \text{norm}(\boldsymbol{v}) = \frac{\boldsymbol{v}}{\max(\|\boldsymbol{v}\|_2, \epsilon)}. \tag{13}$$

The global auxiliary objective is

$$L^{\text{aux}}_{\text{global}} = \lambda_{\text{bal}} L_{\text{bal}} + \lambda_{\text{sem}} L_{\text{sem}} + \lambda_{\text{qprior}} L_{\text{qprior}} + \lambda_{\text{swap}} L_{\text{swap}} + \lambda_{\text{proto}} L^{\text{tea}}_{\text{proto}}. \tag{14}$$

The final term is the teacher-guided prototype loss used by the implementation. For assignments y_{bkc}, the batch assignment mean and balancing loss are

$$\bar{y}_c = \frac{1}{BK} \sum_{b=1}^{B} \sum_{k=1}^{K} y_{bkc}, \qquad L_{\text{bal}} = \frac{1}{C} \sum_{c=1}^{C} \left(\bar{y}_c - \frac{1}{C} \right)^2. \tag{15}$$

Let $\bar{\boldsymbol{y}}_{bk}$ be the deterministic softmax assignment computed without Gumbel noise, and let $\bar{\boldsymbol{g}}^{\text{sel}}_{bk} = \sum_{c=1}^{C} \bar{y}_{bkc} \boldsymbol{g}_c$ be its retrieved content. Semantic guidance is the confidence-weighted squared distance between normalized retrieved content and the stop-gradient SAM3 descriptor:

$$L_{\text{sem}} = \mathcal{W}_s \left[\left\| \text{norm}(\bar{\boldsymbol{g}}^{\text{sel}}_{bk}) - \text{norm}(\text{sg}(\boldsymbol{t}_{bk})) \right\|_2^2 \right]. \tag{16}$$

Teacher prototype distances and assignments are

$$d^{\text{tea}}_{bkc} = \|\text{sg}(\boldsymbol{t}_{bk}) - \boldsymbol{g}_c\|_2^2, \qquad q^{\text{tea}}_{bkc} = \frac{\exp(-d^{\text{tea}}_{bkc}/\tau)}{\sum_{c'=1}^{C} \exp(-d^{\text{tea}}_{bkc'}/\tau)}. \tag{17}$$

The assignment-prior and teacher-prototype losses are

$$L_{\text{qprior}} = \mathcal{W}_s \left[\sum_{c=1}^{C} y_{bkc} \log \frac{y_{bkc}}{\text{sg}(q^{\text{tea}}_{bkc})} \right], \qquad L^{\text{tea}}_{\text{proto}} = \mathcal{W}_s \left[\sum_{c=1}^{C} q^{\text{tea}}_{bkc} d^{\text{tea}}_{bkc} \right]. \tag{18}$$

The teacher assignment is stop-gradient in L_{qprior}. In $L^{\text{tea}}_{\text{proto}}$, it remains dependent on the trainable prototype bank. For synchronized affine transformations T of the image and masks, let v_{bk} indicate slots valid both before and after transformation. The consistency loss is

$$L_{\text{swap}} = \frac{\sum_{b,k} v_{bk} \left\| \bar{\boldsymbol{g}}^{\text{sel}}_{bk}(\boldsymbol{x}, \boldsymbol{M}) - \bar{\boldsymbol{g}}^{\text{sel}}_{bk}(T\boldsymbol{x}, T\boldsymbol{M}) \right\|_2^2}{\max(\sum_{b,k} v_{bk}, 1)}. \tag{19}$$

Both views contribute gradients to this loss.

A.2.2 Object Discovery

For each continuous latent family $a \in \{\text{state}, \text{sce}, \text{bck}\}$, let μ^a_{ij} and $\ell^a_{ij} = \log((\sigma^a_{ij})^2)$ denote the posterior mean and log variance for sample i and dimension j. The KL penalty, averaged over samples and dimensions within each family and then summed across families, is

$$L_{\text{kl},z} = \sum_{a \in \{\text{state},\text{sce},\text{bck}\}} \frac{1}{2N_a d_a} \sum_{i=1}^{N_a} \sum_{j=1}^{d_a} \left[(\mu^a_{ij})^2 + \exp(\ell^a_{ij}) - 1 - \ell^a_{ij} \right]. \tag{20}$$

The discovery auxiliary loss used by the selected experiments has no additional outer coefficient:

$$L^{\text{aux}}_{\text{discov}} = \lambda_{\text{fg}} L_{\text{fg}} + \lambda_{\text{guard}} L_{\text{guard}}. \tag{21}$$

For the selected best Stage 1 configurations, the direct SAM3 mask-supervision coefficient is zero, so this term does not enter the optimized objective. Let $\alpha_{bj}(p) = \hat{m}_{bj}(p)$ denote the normalized assignment in Eq. equation 7 for object or background slot j. The foreground-area penalty discourages excessive foreground assignment:

$$L_{\text{fg}} = \frac{1}{BP_m} \sum_{b,p} \left(1 - \alpha^{\text{bck}}_b(p) \right). \tag{22}$$

To prevent all regions from being assigned to background, define the predicted foreground fraction $r_b = P_m^{-1} \sum_p \sum_{k=1}^{K} \alpha^{\text{obj}}_{bk}(p)$. With $[u]_+ = \max(u, 0)$ and $r_{\min} = 0.02$, the guard penalty is

$$L_{\text{guard}} = \frac{1}{B} \sum_b \left[\log r_{\min} - \log r_b \right]_+. \tag{23}$$

The out-of-support penalty averages feature energy over batch, slots, positions, and channels:

$$L_{\text{out}} = \frac{1}{BKP_f D} \sum_{b,k,p \in \mathcal{P}_f} \left(1 - \tilde{m}^{\text{obj}}_{bk}(p) \right) \left\| \hat{\boldsymbol{o}}^{\text{obj}}_{bk}(p) \right\|_2^2. \tag{24}$$

The discovery objective combines the out-of-support penalty, latent regularization, and the foreground constraints:

$$L_{\text{discov}} = \lambda_{\text{out}} L_{\text{out}} + \lambda_{\text{kl},z} L_{\text{kl},z} + L^{\text{aux}}_{\text{discov}}. \tag{25}$$

A.2.3 Feature Reconstruction

The feature-reconstruction objective combines composed-scene feature matching, background reconstruction, and the auxiliary term:

$$L_{\text{recon}} = \lambda_{\text{feat}} L_{\text{feat}} + \lambda_{\text{bck}} L_{\text{bck}}(t) + L^{\text{aux}}_{\text{recon}}, \qquad L_{\text{feat}} = \text{MSE}(\hat{\boldsymbol{o}}, \text{sg}(\boldsymbol{F}^{\text{sce}})). \tag{26}$$

Here, sg denotes stop-gradient, and the background support is aligned to the feature grid before computing its masked loss. Let $e^{\text{bck}}_b(p) = D^{-1} \| \hat{\boldsymbol{o}}^{\text{bck}}_b(p) - \text{sg}(\boldsymbol{F}^{\text{sce}}_b(p)) \|_2^2$. The hard-support background loss uses the SAM3 background support M^{bck}_b, aligned to the feature grid:

$$L^{\text{comp}}_{\text{bck}} = \frac{\sum_{b,p} M^{\text{bck}}_b(p) e^{\text{bck}}_b(p)}{\max(\sum_{b,p} M^{\text{bck}}_b(p), 1/D)}. \tag{27}$$

For soft background supervision, define $w_b(p) = \eta + (1-\eta)\,\mathrm{sg}(\tilde{m}_b^{\mathrm{bck}}(p))$ with $\eta = 0.1$. Then

$$L_{\mathrm{bck}}^{\mathrm{soft}} = \frac{\sum_{b,p} w_b(p) e_b^{\mathrm{bck}}(p)}{\max(\sum_{b,p} w_b(p), \epsilon)}, \qquad L_{\mathrm{bck}}(t) = (1-\beta_t) L_{\mathrm{bck}}^{\mathrm{comp}} + \beta_t L_{\mathrm{bck}}^{\mathrm{soft}}. \tag{28}$$

The reconstruction auxiliary loss is

$$L_{\mathrm{recon}}^{\mathrm{aux}} = \lambda_{\mathrm{objaux}} L_{\mathrm{objaux}}. \tag{29}$$

For the SAM3-supported auxiliary, let $\hat{\boldsymbol{o}}_{bk}^{\mathrm{aux}} = D_f^{\mathrm{obj}}(\mathrm{sg}(\boldsymbol{g}_{bk}^{\mathrm{sel}}), \mathrm{sg}(\boldsymbol{z}_{bk}^{\mathrm{state}}), \mathrm{sg}(\boldsymbol{z}_b^{\mathrm{sce}}))$. Its masked feature error is normalized within each valid slot and then averaged over valid slots:

$$L_{\mathrm{objaux}} = \frac{1}{\max(\sum_{b,k} v_{bk}, 1)} \sum_{b,k} v_{bk} \frac{\sum_p M_{bk}(p)\, D^{-1} \|\hat{\boldsymbol{o}}_{bk}^{\mathrm{aux}}(p) - \mathrm{sg}(\boldsymbol{F}_b^{\mathrm{sce}}(p))\|_2^2}{\max(\sum_p M_{bk}(p), \epsilon)}. \tag{30}$$

This term updates the object feature decoder but stops gradients to its latent inputs. The predicted-support auxiliary is disabled in every reported configuration ($\lambda_{\mathrm{objfeat}} = 0$) and is therefore omitted from the objective above.

A.2.4 Stage 2 Image-Decoding Losses

The frozen RAE encoder provides the target latent $\boldsymbol{z}_R = E_R(\boldsymbol{x})$. During training, the adapter input transitions from the RAE latent to the reconstructed scene feature:

$$\boldsymbol{f}_{\mathrm{in}} = \rho_t \boldsymbol{z}_R + (1-\rho_t)\hat{\boldsymbol{o}}, \qquad \hat{\boldsymbol{z}}_R = A_\eta(\boldsymbol{f}_{\mathrm{in}}). \tag{31}$$

The schedule decreases ρ_t from one to zero; the adapter output is decoded by the frozen RAE decoder D_R. Validation, editing, and generation use $\rho_t = 0$. Let the predicted and target RAE latents be $\hat{Z}, Z \in \mathbb{R}^{B\times D\times H\times W}$. We decompose the objective into alignment, distribution, and image terms:

$$\begin{aligned} L_{\mathrm{align}} &= \lambda_{\mathrm{huber}} L_{\mathrm{huber}} + \lambda_{\cos} L_{\cos}, \\ L_{\mathrm{distr}} &= \lambda_{\mathrm{stat}} L_{\mathrm{stat}} + \lambda_{\mathrm{cov}} L_{\mathrm{cov}}, \\ L_{\mathrm{image}} &= \lambda_{\ell_1} L_{\ell_1} + \lambda_{\mathrm{lpips}} L_{\mathrm{lpips}}, \\ L_{\mathrm{stage2}} &= \lambda_{\mathrm{align}} L_{\mathrm{align}} + \lambda_{\mathrm{distr}} L_{\mathrm{distr}} + \lambda_{\mathrm{image}} L_{\mathrm{image}}. \end{aligned} \tag{32}$$

The outer coefficients satisfy $\lambda_{\mathrm{align}} = \lambda_{\mathrm{distr}} = \lambda_{\mathrm{image}} = 1$.

The Smooth L1 penalty uses threshold one. For $u \in \mathbb{R}$, let $\phi(u) = \frac{1}{2}u^2$ if $|u| < 1$ and $\phi(u) = |u| - \frac{1}{2}$ otherwise. Then

$$L_{\mathrm{huber}} = \frac{1}{BDHW} \sum_{b,d,h,w} \phi(\hat{Z}_{bdhw} - Z_{bdhw}). \tag{33}$$

The cosine loss is computed after flattening each image's latent tensor:

$$L_{\cos} = 1 - \frac{1}{B} \sum_b \frac{\langle \mathrm{vec}(\hat{Z}_b), \mathrm{vec}(Z_b) \rangle}{\max(\|\mathrm{vec}(\hat{Z}_b)\|_2, \epsilon) \max(\|\mathrm{vec}(Z_b)\|_2, \epsilon)}. \tag{34}$$

This is an image-wise flattened cosine loss; token-wise cosine similarity is used only for logging. Treating the $N = BHW$ spatial tokens as samples, let Z_{nd} denote channel d of token n. Channel means and population standard deviations are $\mu_d(Z) = N^{-1}\sum_n Z_{nd}$ and $\sigma_d(Z) = \sqrt{N^{-1}\sum_n (Z_{nd} - \mu_d(Z))^2}$. The statistic-matching loss is

$$L_{\mathrm{stat}} = \frac{1}{D} \sum_d \Big(|\mu_d(\hat{Z}) - \mu_d(Z)| + |\sigma_d(\hat{Z}) - \sigma_d(Z)| \Big). \tag{35}$$

Define the channel covariance $\Sigma(Z) = N^{-1}(Z - \mathbf{1}\boldsymbol{\mu}(Z)^\top)^\top (Z - \mathbf{1}\boldsymbol{\mu}(Z)^\top)$. The covariance loss is

$$L_{\mathrm{cov}} = \frac{1}{D} \|\Sigma(\hat{Z}) - \Sigma(Z)\|_F. \tag{36}$$

Thus, this term measures channel covariance estimated over batch and spatial tokens. For decoded images $\hat{\boldsymbol{x}} = D_R(\hat{Z})$ resized to the target resolution, the image losses are

$$L_{\ell_1} = \frac{1}{3BH_xW_x} \sum_{b,c,p} |\hat{x}_{bcp} - x_{bcp}|, \qquad L_{\mathrm{lpips}} = \frac{1}{B} \sum_b \mathrm{LPIPS}_{\mathrm{AlexNet}}(2\hat{\boldsymbol{x}}_b - 1, 2\boldsymbol{x}_b - 1). \tag{37}$$

Table 3: Stage 1 loss coefficients for the selected best runs.

Group	Coefficient	GSO	OCTB	YCB	VOC2012	COCO2017
Global	λ_{proto}	0.05	0.05	0.05	0.05	0.05
	$\lambda_{\text{kl},p}$	0.05	0.05	0.01	0.01	0.01
	λ_{bal}	0.01	0.01	0.01	0.01	0.01
	λ_{sem}	0.05	0.05	0.05	0.05	0.05
	λ_{qprior}	0.05	0.05	0.05	0.05	0.05
	λ_{swap}	0.05	0.05	0.20	0.20	0.20
Discovery	λ_{out}	0.05	0.05	0	0	0
	$\lambda_{\text{kl},z}$	0.10	0.10	0.10	0.10	0.10
	λ_{fg}	0	0.001	0	0	0
	λ_{guard}	0	0	0.01	0.01	0.01
Reconstruction	λ_{feat}	1	1	1	1	1
	λ_{bck}	0.10	0.10	0.10	0.10	0.10
	λ_{objaux}	0	0	0.05	0.05	0.05
Mask ablation	λ_{mask}	0	0	0	0	0

Table 4: Stage 2 loss coefficients, shared across the selected runs.

Group	Coefficient	GSO	OCTB	YCB	VOC2012	COCO2017
Alignment	λ_{huber}	1	1	1	1	1
	$\lambda_{\cos}$	0.20	0.20	0.20	0.20	0.20
Distribution	λ_{stat}	0.05	0.05	0.05	0.05	0.05
	λ_{cov}	0.01	0.01	0.01	0.01	0.01
Image	λ_{ℓ_1}	1	1	1	1	1
	λ_{lpips}	0.50	0.50	0.50	0.50	0.50

The LPIPS inputs are rescaled from $[0, 1]$ to $[-1, 1]$.

All Stage 2 configurations use $\lambda_{\text{align}} = \lambda_{\text{distr}} = \lambda_{\text{image}} = 1$; Table 4 reports the component coefficients. Overall, Stage 1 combines global prototype learning and semantic guidance with latent regularization, object-discovery constraints, and compositional feature reconstruction. The prototype, balancing, semantic, and assignment-prior weights are shared across datasets (0.05, 0.01, 0.05, and 0.05, respectively), while the prototype KL weight is 0.05 on GSO and OCTB and 0.01 on YCB, VOC2012, and COCO2017. The swap-consistency weight is 0.05 on GSO and OCTB and 0.20 on the other datasets, and the continuous latent KL weight is 0.10 throughout. Discovery weights vary by dataset: the out-of-support term is 0.05 on GSO/OCTB, the foreground term is 0.001 on OCTB, and the guard term is 0.01 on YCB/VOC2012/COCO2017; the object-feature auxiliary has weight 0.05 on those latter three datasets. Feature reconstruction uses weight 1 and background reconstruction uses 0.10 throughout, while direct mask supervision is disabled in the selected configurations. Stage 2 uses the same coefficients across datasets: unit outer weights, Huber and image ℓ_1 weights of 1, cosine weight 0.20, statistic and covariance weights of 0.05 and 0.01, and LPIPS weight 0.50.

A.3 Object Attribute Disentanglement

We examine content–state separation in GOLP using six samples from GSO and OCTB, with three samples per dataset. For each sample, we select two objects and exchange their state representations while retaining their global content. Figure 5 shows the resulting reconstructions and the original object observations; this is a qualitative intervention study of GOLP rather than a cross-method comparison.

In Figure 5, the reconstructions in the second and fourth rows largely follow the spatial states of the objects in the third and first rows, respectively, while retaining recognizable content from the first and third rows. These interventions are consistent with a separation of shared appearance and shape from observation-dependent spatial attributes. In Sample 1 of OCTB, the decoded Object 2 appears complete despite substantial occlusion in the input. This suggests that shared content can support

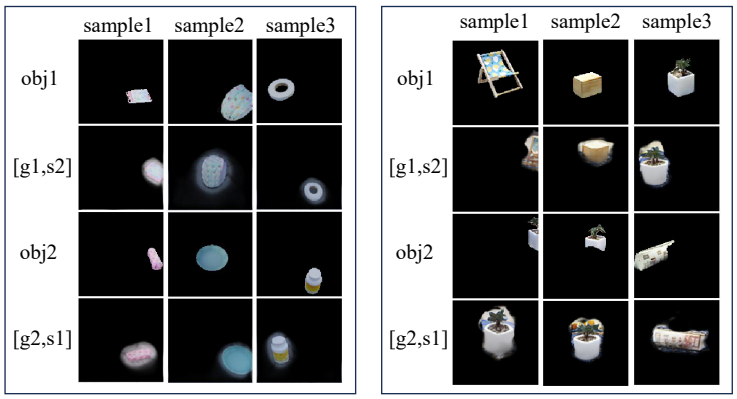


Figure 5: Visualization of object attribute disentanglement results by GOLP on the GSO and OCTB datasets. Each column in a subplot shows attribute-swapping results for one sample. $[g_i, s_j]$ denotes reconstruction using object i's global representation and object j's state.

completion of a partially observed object, although the example does not constitute a quantitative amodal-reconstruction evaluation.

A.4 Scene Reconstruction, Editing, and Generation

The latent scene representation consists of the selected global content, object states, scene attributes, background, and active object set:

$$\mathcal{Z} = \left\{ \boldsymbol{z}^{\mathrm{sce}}, \boldsymbol{z}^{\mathrm{bck}}, \left(\boldsymbol{g}_k^{\mathrm{sel}}, \boldsymbol{z}_k^{\mathrm{state}} \right)_{k \in \mathcal{V}}, \mathcal{V} \right\}. \tag{38}$$

Let $\mathcal{G}_\theta$ denote the compositional feature decoder. Reconstruction, editing, and generation share

$$\mathrm{decode}(\mathcal{Z}) = \mathcal{R}\left(D_R\left(A_\eta\left(\mathcal{G}_\theta(\mathcal{Z}) \right) \right) \right). \tag{39}$$

Reconstruction uses the inferred global identity and posterior means of the attribute variables. Editing replaces selected identities or states, or modifies the active object set to remove or insert objects; the features and masks are then recomputed.

For generation, object identities are specified or sampled as $c_k \sim \mathrm{Cat}(\boldsymbol{\pi})$, with $\boldsymbol{g}_k^{\mathrm{sel}} = \boldsymbol{g}_{c_k}$. Object-state, scene, and background variables are sampled from their standard normal priors. Reference-conditioned generation retains the unsampled factors from an encoded scene, while interpolation between object-state or scene-attribute codes varies the corresponding conditions. Once $\mathcal{Z}$ is constructed, decoding requires no additional access to the original image or its extracted features.

A.5 Evaluation Protocol and Scene-Decomposition Results

Evaluation protocol. ACC follows the recognition protocol in the main experiments. ARI-O is computed over ground-truth foreground pixels, whereas ARI-A includes background. For GSO, OCTB, and YCB, mBO^c and mIoU^c use visible ground-truth regions merged by category. Each evaluated image in these partitions has no repeated valid category, so category-merged and instance-overlap values coincide on these partitions. COCO uses visible instance regions, whereas VOC uses connected components of semantic masks as proxy instances. mBO selects the highest IoU for each target; mIoU uses one-to-one Hungarian matching and assigns zero to unmatched targets. Predicted background slots remain eligible for overlap matching, while ground-truth background is not an overlap target.

The evaluation uses the packaged partitions at 128×128 resolution, with one view per scene in the main datasets. The OCTB and COCO evaluation partitions coincide with their validation partitions. COCO and VOC follow the internal data-processing conventions rather than the official

Table 5: GOLP recognition and scene-decomposition results summarized from the main-paper tables. ACC is copied at its reported precision; ARI and overlap values are shown on the original score scale. The superscript c denotes category-merged reference regions.

Dataset	Categories / max. objects	ACC ↑	ARI-A ↑	ARI-O ↑	mBO^c ↑
GSO	10 / 6	0.871	0.5245	0.9367	0.4510
OCTB	34 / 14	0.517	0.6431	0.8079	0.4692
YCB	21 / 8	0.506	0.5828	0.6276	0.3773

benchmark protocols. VOC results concern connected-component proxies, including for ARI, and do not provide official instance-segmentation scores.

Table 5 summarizes GOLP's category counts and object capacities together with the main-paper results. ACC reproduces Table 1; decomposition scores reproduce Table 2 after division by 100.

GSO achieves the highest recognition accuracy and ARI-O among the three datasets. However, its full-image grouping and region overlap are lower than those of GOLD and CSGO in Table 2. Thus, consistent grouping of foreground pixels does not imply equally accurate background assignment or object extent.

OCTB yields GOLP's highest ARI-A and mBO^c among the three datasets despite having the largest category set and object capacity. On this dataset, GOLP leads the evaluated baselines in both grouping and overlap, indicating that its advantage extends beyond foreground assignments alone.

YCB gives the lowest ARI-O and mBO^c among the three datasets for GOLP, indicating remaining difficulties in both foreground grouping and object coverage. Nevertheless, GOLP outperforms all evaluated baselines on this dataset.

Together, these results show different strengths across datasets: GSO gives the strongest foreground grouping and recognition, OCTB the best full-image grouping and overlap, and YCB remains the most challenging of the three for GOLP's foreground grouping and coverage. They also illustrate why recognition, foreground grouping, and region overlap should be assessed separately.

A.6 Additional Object Discovery Results on COCO and VOC

Experimental objective. To complement the results on GSO, OCTB, and YCB, we evaluate GOLP's compositional object discovery on complex natural images from COCO and VOC. We compare GOLP with CSGO, GOLD, CoSA, and MUFASA. Evaluation covers 2,000 packaged COCO images and 1,203 VOC validation images, with all predicted masks evaluated at 128×128 resolution. GOLP is evaluated using its learned decoder masks rather than the input SAM3 candidates. Methods retain their native preprocessing and trained slot capacities; GOLP additionally receives its associated SAM3 inputs.

Quantitative results. Table 6 reports ARI-A, ARI-O, mBO^i, and mIoU^i, following the grouping and overlap definitions used in the main experiments. VOC overlap metrics are marked with an asterisk to distinguish their proxy-instance interpretation.

Analysis. On COCO, GOLP outperforms GOLD and CSGO on all four metrics. It also achieves higher ARI-A and ARI-O than CoSA, although its region overlap is lower. MUFASA obtains the strongest results across all four metrics under the evaluated configurations. On VOC, GOLP exceeds GOLD, CSGO, and CoSA on all reported metrics and achieves the highest ARI-O. Its ARI-A is close to MUFASA's, differing by only 0.61 points, but MUFASA retains a substantial advantage in region overlap. These rankings distinguish foreground grouping from accurate object extent. The VOC overlap results refer to connected-component proxies rather than annotated object instances. Overall, the results support GOLP's applicability to these natural-image subsets while identifying spatial coverage and reliable full-scene decomposition as remaining challenges.

Table 6: **Additional object-discovery results on COCO and VOC.** Scores are multiplied by 100; higher is better. Results use internal evaluation subsets and a single fixed inference seed. The highest numerical score for each dataset and metric is shown in bold. *VOC uses connected-component reference regions; its overlap scores are proxy-instance measures.

Dataset	Method	ARI-A ↑	ARI-O ↑	mBO^i ↑	mIoU^i ↑
COCO	CSGO	-4.22	6.63	11.87	10.27
	GOLD	-1.24	12.15	9.60	7.42
	CoSA	6.24	21.52	19.81	19.17
	MUFASA	**25.25**	**32.05**	**30.78**	**28.34**
	GOLP (ours)	20.58	22.65	17.47	16.25
VOC*	CSGO	-3.09	12.75	22.09	21.20
	GOLD	0.00	23.64	18.22	17.01
	CoSA	5.74	12.33	20.72	20.47
	MUFASA	**29.10**	19.05	**44.66**	**43.11**
	GOLP (ours)	28.49	**26.53**	30.22	29.73

A.7 SAM3 Mask-Supervision Analysis

A.7.1 Experimental Setup

We examine the reported configurations with and without direct SAM3 mask supervision. The additional term is

$$\mathcal{L}_{\text{mask}} = \mathcal{L}_{\text{CE}}(\hat{\alpha}, M^{\text{SAM3}}) + \mathcal{L}_{\text{Dice}}(\hat{\alpha}, M^{\text{SAM3}}), \tag{40}$$

where $\hat{\alpha}$ denotes the predicted normalized mask assignments and M^{SAM3} the SAM3 candidates. A positive coefficient applies CE/Dice guidance, whereas zero removes direct mask supervision. Candidate masks remain in use for regional observations and object-state encoding in both cases, so this comparison is distinct from removing the spatial prior. The main configurations summarized in Table 3 use $\lambda_{\text{mask}} = 0$.

The OCTB entry labeled 0.5 refers to the available run initialized at 0.5 and reduced toward zero during training; it is not evidence for a coefficient held fixed at 0.5. The retained label identifies that run, rather than a matched fixed-coefficient experiment.

A.7.2 Stage 1 Results

Table 7 uses the main-model results as the zero-supervision references: GSO, OCTB, and YCB values come from Tables 1 and 2; VOC and COCO overlap and ARI-O values come from Table 6. Recognition is not reported for those natural-image references and is marked as unavailable. Positive-supervision rows retain the previously recorded run measurements. Because consistent checkpoint-selection and evaluation protocols and matched training budgets have not been established across these sources, the table is a descriptive comparison of configurations rather than an isolated estimate of the mask coefficient's effect.

The reference comparisons exhibit different rankings across datasets and metrics. On GSO, the main zero-supervision configuration has higher recognition accuracy and ARI-O than the reported positive-supervision run, but lower overlap. On OCTB, the positive-supervision run has higher ARI-O and overlap, while both configurations report approximately 0.517 ACC at shared precision. On YCB, the main configuration has slightly higher ACC and ARI-O, whereas the positive-supervision run has higher overlap. On VOC, ARI-O favors the main configuration and overlap favors the positive-supervision run; on COCO, the latter has higher ARI-O and overlap. These observations describe the listed configurations and do not isolate the effect of mask supervision. In particular, differing result sources and the OCTB training schedule preclude treating all rows as matched fixed-coefficient ablations.

Table 7: Stage 1 results for selected mask-supervision configurations. Zero-supervision references reproduce the main results; positive-supervision rows retain the available run measurements. Scores use the original scale, and – denotes an unreported reference value.

Dataset	λ_{mask} setting	ARI-O ↑	mBO ↑	ACC ↑
GSO	0.5	0.8908	0.6221	0.8448
	0	0.9367	0.4510	0.871
OCTB	0	0.8079	0.4692	0.517
	0.5	0.8368	0.6300	0.5174
YCB	0.5	0.6207	0.5028	0.4981
	0	0.6276	0.3773	0.506
VOC2012	0.5	0.2065	0.4341	0.3047
	0	0.2653	0.3022	–
COCO2017	0.5	0.3561	0.3320	0.2007
	0	0.2265	0.1747	–

Table 8: Separately recorded Stage 2 reconstruction results for the original mask-supervision run pairs. Lower is better for L1 and LPIPS; higher is better for PSNR and SSIM.

Dataset	λ_{mask}	L1 ↓	PSNR ↑	SSIM ↑	LPIPS ↓
YCB	0.5	0.2586	9.67	0.282	0.4799
	0	0.2726	9.53	0.281	0.5050
VOC2012	0.5	0.2064	11.75	0.252	0.4135
	0	0.2115	11.74	0.223	0.4037
COCO2017	0.5	0.2124	11.75	0.220	0.4104
	0	0.2142	11.57	0.214	0.4210

A.7.3 Stage 2 Reconstruction Results

Table 8 retains the separately recorded Stage 2 comparisons on YCB, VOC, and COCO. These RGB measurements belong to the original decoder-run pairs and are not re-evaluations of the main-model reference rows introduced in Table 7.

Within these decoder-run pairs, reconstruction differences are modest and metric-dependent. Positive-supervision runs improve L1, PSNR, and SSIM on YCB and COCO, while VOC has small gains in these metrics and a slightly lower LPIPS without direct mask supervision. Final RGB fidelity depends on the composed features and the image-decoding pathway, so these results do not separately identify the contribution of mask supervision or decoder quality. A matched native-RAE reconstruction reference is not reported for these runs, preventing a direct comparison with the frozen decoder's reconstruction ceiling.

A.7.4 Qualitative Observations

The following observations concern the separately recorded mask-supervision runs, not all of the main-model references. On YCB, the positive-supervision run exhibits clearer object contours, whereas the paired zero-supervision run more often produces blurred or low-energy templates. Both VOC settings show repeated-texture artifacts after image decoding; the supervised examples more often retain object-aligned content. COCO remains challenging under both settings. These observations are qualitative and should not be interpreted as a controlled comparison against every main-paper visualization.

Overall, the available comparisons reveal a trade-off between region overlap and identity-related objectives rather than a uniform ranking of mask-supervision settings. In the tabulated references, positive-supervision runs have higher overlap, while the ARI-O and ACC rankings vary by dataset. SAM3 candidates remain spatial inputs even when direct mask supervision is disabled. Establishing

Table 9: SAM3 prompts and candidate post-processing by dataset. The final column gives the maximum number of candidates retained per image.

Dataset	Prompt(s)	Candidate generation and post-processing	Max. retained
GSO	object	Confidence threshold 0.01; resolution 1008; NMS, connected-component filtering, and ROI refinement	6
OCTB	object	Same settings, with at most 20 initial candidates	14
YCB	object	Confidence threshold 0.01; resolution 1008; NMS, connected-component filtering, and ROI refinement	8
VOC2012	object; person; animal; vehicle	Four prompt inferences; score-based NMS; minimum area 32 pixels; NMS IoU 0.7	15
COCO2017	object; person; animal; vehicle	Same as VOC2012	15

Table 10: Statistics of final SAM3 candidate masks written to `sam3_teacher.h5`.

Dataset	Mean candidates	Median	Empty-candidate rate	At retention cap
GSO	5.69	6	0.00%	81.96%
OCTB	13.02	14	0.00%	62.44%
YCB	7.92	8	0.18%	95.36%
VOC2012	14.26	15	0.37%	86.98%
COCO2017	13.71	15	1.20%	80.06%

the isolated effect of the coefficient requires matched configurations and evaluation protocols; the present comparisons are descriptive.

A.8 SAM3 Prompts and Candidate Masks

A.8.1 Prompts and Post-Processing

Table 9 summarizes the dataset-level prompts and candidate-processing settings. For GSO, OCTB, and YCB, SAM3 uses the generic prompt `object`, followed by the shared ROI-refinement instruction `segment every distinct object`. VOC2012 and COCO2017 use four fixed prompts (`object`, `person`, `animal`, and `vehicle`); candidates are generated for each prompt and merged by score-based NMS. The prompt set is fixed for each dataset and does not read per-image test annotations. The prompts `person`, `animal`, and `vehicle` are dataset-level semantic priors.

A.8.2 Candidate Counts

Table 10 reports statistics computed from the final candidates, rather than from later model predictions. Reaching the retention cap is common on YCB, VOC2012, and COCO2017, showing that `max_keep` acts as an active truncation limit rather than only as a padding size.

The high saturation rates on YCB, VOC2012, and COCO2017 indicate that these scenes often produce more candidates than the configured cap allows, consistent with overlapping or redundant regions and possible candidate truncation. GSO has no empty-candidate samples in this summary and a more regular scene structure, although its retention cap is also frequently reached.

A.8.3 Prompt Ablation on VOC2012 and COCO2017

A 100-image pilot compares the generic `object` prompt with the four-prompt set. As shown in Table 11, adding `person`, `animal`, and `vehicle` reduces the empty-candidate rate from 17% to 1% on VOC2012 and from 18% to 1% on COCO2017. It also increases mBO from 0.3072 to 0.6359 on VOC2012 and from 0.3466 to 0.6286 on COCO2017. The corresponding ARI-O changes are smaller on VOC2012 (0.3005 to 0.3204) and larger on COCO2017 (0.3499 to 0.5910).

These pilot results show that the additional prompts reduce empty candidate sets and improve candidate-mask overlap on both datasets, while increasing the number of overlapping or redundant

Table 11: 100-image pilot comparing the generic object prompt with four fixed prompts on VOC2012 and COCO2017.

Dataset	Prompt setting	Mean candidates	Empty-candidate rate	mBO ↑	ARI-O ↑
VOC2012	`object`	3.79	17%	0.3072	0.3005
	Four prompts	14.01	1%	0.6359	0.3204
COCO2017	`object`	4.67	18%	0.3466	0.3499
	Four prompts	13.99	1%	0.6286	0.5910

proposals. They are candidate-mask results from a 100-image pilot and do not establish improvements of the same magnitude in the final representation-learning metrics. The prompt set is fixed at the dataset level and does not read per-image test-image ground-truth labels.

A.8.4 Qualitative Observations

On GSO, the generic object prompt covers most isolated objects with reasonably complete boundaries. The main errors arise when floor regions, shadows, or large textured areas are proposed as objects, and when adjacent objects are merged. On YCB, object prompting with ROI refinement covers most tabletop objects, but contact and occlusion can produce merged or duplicate regions; checkerboard patterns and tabletops may also generate large masks. The near-saturated candidate count indicates high proposal volume but also substantial redundancy. On OCTB, the cap of 14 candidates is often approached in scenes with many objects. SAM3 generally captures salient objects, while tabletops, tablecloths, and background textures can become large candidates; severe occlusion may leave small objects merged or only partially segmented. On VOC2012 and COCO2017, the generic object prompt alone often misses people, animals, and vehicles. The four-prompt set improves their coverage, at the cost of more overlapping candidates and occasional background fragments.