CALICO: Part-Focused Semantic Co-Segmentation with Large Vision-Language Models

Kiet A. Nguyen Adheesh Juvekar Tianjiao Yu Muntasir Wahed Ismini Lourentzou
University of Illinois Urbana - Champaign
{kietan2, adheesh2, ty41, mwahed2, lourent2}@illinois.edu
https://plan-lab.github.io/calico

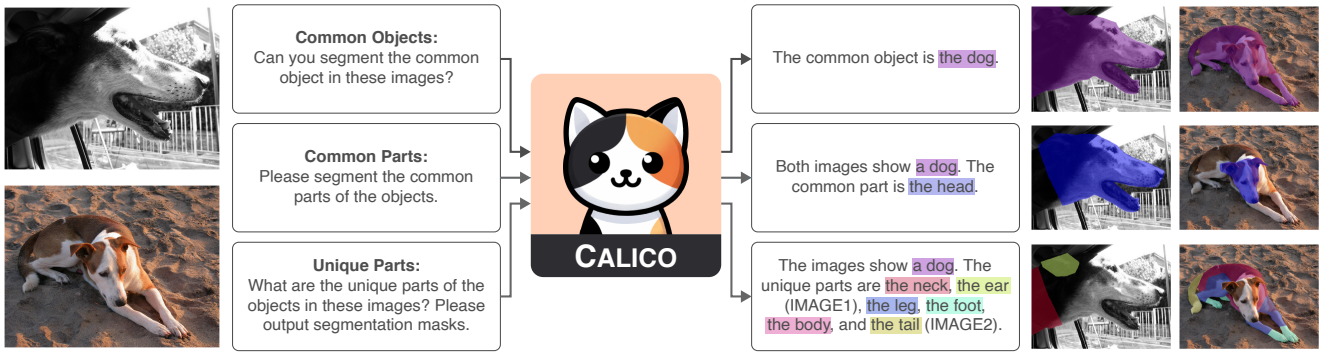

Figure 1. **Multi-Image Part-focused Object Comparison with CALICO.** Our pixel-grounded Large Vision-Language Model, CALICO, can perform part-focused semantic co-segmentation, a newly introduced task where the goal is to identify, segment, and label common objects, as well as common and unique object parts across multiple images.

Abstract

Recent advances in Large Vision-Language Models (LVLMs) have sparked significant progress in general-purpose vision tasks through visual instruction tuning. While some works have demonstrated the capability of LVLMs to generate segmentation masks that align phrases with natural language descriptions in a single image, they struggle with segmentation-grounded comparisons across multiple images, particularly at finer granularities such as object parts. In this paper, we introduce the new task of part-focused semantic co-segmentation, which seeks to identify and segment common and unique objects and parts across images. To address this task, we present CALICO*, the first LVLM that can segment and reason over multiple masks across images, enabling object comparison based on their constituent parts.* CALICO *features two proposed components, a novel Correspondence Extraction Module, which captures semantic-rich information to identify part-level correspondences between objects, and a Correspondence Adaptation Module, which embeds this information into the LVLM to facilitate multi-image understanding in a parameter-efficient manner. To support training and evaluation, we curate* MIXEDPARTS*, a comprehensive multi-image segmentation dataset containing ∼2.4M samples across ∼44K images with diverse object and part categories. Experimental results show* CALICO*, finetuned on only 0.3% of its architecture, achieves robust performance in part-focused semantic co-segmentation.*

1. Introduction

Analyzing objects by decomposing them into their constituent parts can enhance understanding of inter-object relationships both within and across categories. This part-level understanding is crucial for applications requiring detailed object comparisons, such as robotic manipulation, medical imaging, and educational tools. By recognizing similarities and differences at the part level across multiple images, these applications can enable more context-driven analysis and actions. Detailed comparisons that identify shared and unique parts offer insights into the critical features and functions of objects. For example, while both

spoons and forks have handles, this part is not central to their primary functions. Instead, distinguishing the fork's tines from the spoon's bowl allows tasks like robotic grasping or visual comparisons to differentiate and interact with these objects based on their unique functions.

Designing effective methods to analyze multiple images featuring diverse objects, and locate and identify their shared and distinct parts, presents an intriguing challenge. Given two images of similar objects, the goal is to generate segmentation masks for the shared or distinct parts (locate), establish one-to-one correspondences across images for comparative analysis (compare), and assign descriptive labels to these parts (identify). We term this task ***part-focused semantic co-segmentation***.

Research in part-focused semantic segmentation has explored various facets of this related granular task. Many studies have centered on single-image localized part learning [4, 5, 29, 38, 58, 63], focusing on segmenting parts within an object as a terminal task [4, 38] or as a means to support broader tasks like object or human parsing [53, 75]. However, these methods are not designed for multi-image comparisons, as the segmented parts from different images lack a consistent mapping of corresponding semantic components across images. Conversely, some research in this realm has tackled the related task of part co-segmentation, where models learn to segment objects across multiple images in a semantically consistent manner, despite variations in pose, shape, camera angle, *etc*. [1, 5, 13, 20, 33]. Yet, these approaches often require the number of part classes to be specified as input and struggle to identify *unique* parts between objects. Most part co-segmentation methods segment the entire object into parts, covering all areas of the object, which limits the model's flexibility in pinpointing and analyzing subtle, part-specific variations that are essential for comparing different objects. Furthermore, the unsupervised or semi-supervised nature of recent methods hampers their ability to accurately label object parts.

Recently, vision research has experienced a surge of interest in leveraging LVLMs for object segmentation tasks [24, 46, 50, 60, 67, 68]. Following LISA [24], many studies have adopted approaches that output additional segmentation tokens [46, 50, 68], which contain mask embeddings of objects within an image. These tokens are then processed by a decoder to generate the final segmentation masks. Such methods have proven highly effective for unifying tasks like semantic segmentation and referring expression segmentation within a single architecture, thanks to the adaptability of LVLMs in handling diverse input and output types. Although some studies have demonstrated varying levels of part understanding [24, 46, 67], they do not address reasoning about parts across multiple images or segmenting shared and distinctive parts between objects.

To this end, we introduce **C**omponent-Focused **A**daptive **L**earning for Multi-**I**mage **C**o-Localization of **O**bjects (**CALICO**), a Vision-Language Model (VLM) designed to perform localized object comparison across image pairs. CALICO's architecture is unique in its design, combining a Correspondence Extraction Module to capture part-level semantic relationships and a Correspondence Adaptation Module to embed these relationships efficiently into an LVLM. These components enable multi-image, part-level understanding without substantially increasing the parameter count. CALICO augments a pretrained segmentation-based LVLM with a parameter-efficient adapter module to learn multi-image co-localization at multiple granularities. To extract semantic correspondence information between images, we propose a novel Correspondence Extraction Module, which employs a frozen encoder with strong semantic correspondence capabilities learned through self-supervised training [3]. This extracted information is then incorporated within the model using a Correspondence Adaptation Module applied at various layers of the pretrained LVLM to facilitate inter-image understanding. Due to the lightweight nature of these adaptive modules, the trainable parameters in CALICO represent only **0.3%** ($\sim$29M) of the entire architecture. To the best of our knowledge, CALICO represents the pioneering effort in training an LVLM to perform multi-image part co-segmentation.

To enable effective training for this novel task, we introduce the **M**ulti-**I**mage **Cross**-S**e**gmentation of **D**istinctive and Common **Parts** (**MIXEDPARTS**) dataset that contains diverse and logically comparable object pairs, allowing CALICO to generalize across varied categories and visual details. Leveraging widely used, publicly available part segmentation datasets [15, 45, 61, 73, 74], we manually curate pairs of object labels that are logically comparable and share at least one common part label. For instance, pairing a "chair" with an "ottoman" is logical since both belong to the category of seating furniture, making them more comparable than, *e.g*., a "chair" and a "microwave." We then construct image pairs based on these labels to train the model to co-segment not only the objects themselves but also their shared and unique parts across multiple images.

Lastly, as our work is the first to address multi-image part co-segmentation, we create baselines using publicly available pretrained models to tackle this novel task. We conduct extensive experiments on MIXEDPARTS to evaluate the effectiveness of our approach, including ablation studies to assess the contributions of the proposed correspondence extraction and adaptation modules. CALICO achieves superior performance in part-focused semantic co-segmentation, showing a 20.8% relative mean IoU improvement on MIXEDPARTS compared to the next best baseline. In summary, our contributions are as follows:

- We introduce the novel task of ***part-focused semantic co-segmentation***, which aims to co-segment and label com-

mon and unique parts between objects across multiple images for granular object comparison. To the best of our knowledge, this is the first work to formalize this multi-image object/part co-segmentation task.
- We propose **CALICO**, an LVLM designed for part-focused semantic co-segmentation. CALICO incorporates a novel correspondence extraction module and an adaptation module to learn semantic correspondences and localized co-segmentation across multiple images in a parameter-efficient manner.
- We introduce the **MIXEDPARTS** dataset for part-focused semantic co-segmentation, compiled from diverse part segmentation datasets and featuring images of logically comparable objects and parts.
- We construct baselines using publicly available pretrained models and conduct experiments to evaluate CALICO's performance on MIXEDPARTS, including ablation studies to analyze the contributions of the correspondence extraction and adaptation modules.

2. Related Work

2.1. Image Segmentation

Image segmentation, a fundamental task in computer vision, has garnered extensive research over the years, with seminal works [16, 22, 35] paving the way for modern approaches. Recently, SAM [23] has emerged as a leading segmentation method due to its zero-shot capabilities and ability to generate accurate masks for novel objects. Despite its strong zero-shot performance, SAM cannot identify masked objects given arbitrary text input and relies on region inputs to operate effectively. To address this limitation, Semantic-SAM [26] employs text encoder outputs atop its decoder to label segmentation masks, while also enabling labeled segmentation at any granularity. SEEM [77] introduces a joint image-text representation space allowing labeling of segmentation masks. SAM demonstrates robust segmentation capabilities even when combined with LVLMs [24, 46, 60]. Thus, we incorporate a SAM decoder in CALICO. Building upon these works, we further fine-tune SAM's pixel decoder using the LLM's special token output embeddings to produce the desired segmentation masks. While the aforementioned works focus on segmenting individual images, CALICO tackles multi-image object and part co-segmentation.

2.2. Part Segmentation

Part segmentation enables a more granular understanding by parsing objects into meaningful components or parts. While numerous approaches have been proposed to address this task, many are tailored to specific domains [21, 65, 66, 76] or are constrained to closed-set vocabularies [9, 29, 38, 39]. A recent approach, VLPart [56], efficiently tackles this task with open vocabulary predictions by utilizing a pre-trained LVLM [44] and pretrained DINO features [3, 41] to perform semantic correspondence with base objects, enabling the model to label any part of the object. However, in practice, the labels are constrained by the input to the text encoder. Many recent works [24, 46, 50, 60, 67, 68] integrate the reasoning capabilities of LVLMs with segmentation models to augment visual understanding of the LVLMs. These models enable open-vocabulary object segmentation, without being constrained to the input prompt. Most of these models, motivated by LISA [24], utilize a pretrained LVLM such as LLaVA [32] in tandem with a decoder to predict segmentation masks.

While several of these works can also segment object parts [24, 46, 67] since their training data consists of large annotated datasets containing part information, their part segmentation capabilities often lag behind their object segmentation capabilities. Additionally, they also require explicit mention of object parts in the input prompt for successful segmentation. In other words, simply requesting to "segment multiple parts of the given object" does not yield the desired results. Instead, each part must be individually specified, *e.g.* by requesting to segment the leg of the chair to obtain the desired segmentation maps. In contrast, CALICO augments a pretrained LVLM with object and part co-segmentation capabilities without the need for explicit part-specific prompts, simplifying user input requirements and allowing flexibility in instructing the model to infer and delineate various objects and parts across different images.

2.3. Object/Part Co-Segmentation

Co-segmentation aims to discover common objects across multiple images. Early co-segmentation works [25, 69, 72] employ CNNs with fully supervised training or fine-tuning on co-segmentation datasets. LCCo [11] leverages the semantic understanding of CLIP [44] to perform object co-segmentation. However, these approaches do not extend to co-segmenting object parts. Conversely, part co-segmentation involves simultaneously segmenting corresponding parts across multiple images. Previous methods [1, 5, 13, 20, 33] tackle this task with unsupervised or self-supervised learning. DFF [6] employs matrix factorization with features from a pretrained CNN, while SCOPS [20] trains an encoder-decoder CNN with an equivariance loss for robust part segmentation and a semantic consistency loss for improved co-segmentation. Moreover, a recent work [1] identifies that DINO features encapsulate semantic and spatial correspondences that can be utilized to find similar parts of objects across different images.

However, although these methods can co-segment parts, they lack the ability to predict semantic part labels and generate segmentation masks for unique object parts. In this work, we introduce CALICO, a model that harnesses the zero-shot capabilities of SAM and the rich semantic infor-

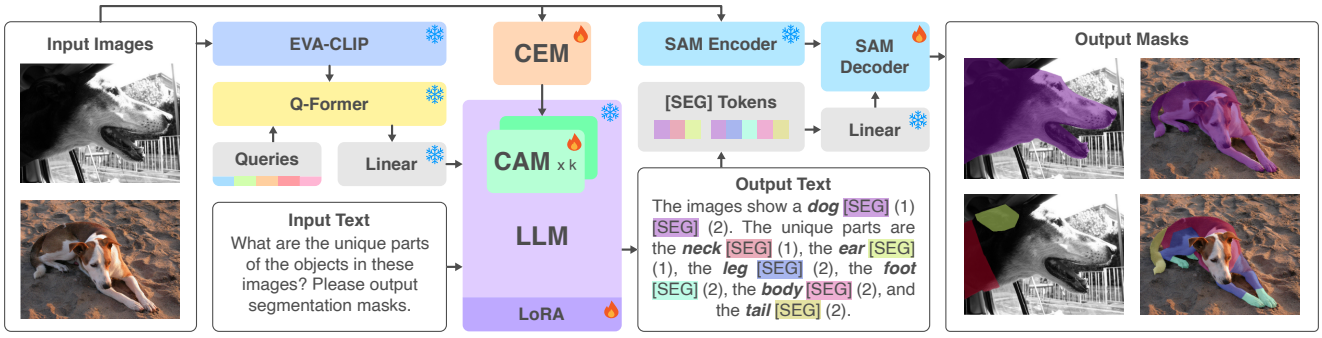

Figure 2. **Overview of the CALICO Architecture for Part-Focused Semantic Co-Segmentation.** CALICO uses a Q-Former cross-attention module to query efficient image embeddings from a pretrained image encoder, which are passed into a Vicuna-based LLM as image features. We extract `[SEG]` tokens from the output text, which are used to prompt a SAM decoder to output corresponding segmentation masks. We propose two modules, the Correspondence Extraction Module (CEM) and the Correspondence Adaptation Module (CAM), to learn semantic-rich features for multi-image correspondence information; details in Figure 3.

mation obtained from DINOv2 features [8, 41] for part co-segmentation in conjunction with part labeling. CALICO integrates LVLMs with co-segmentation to improve object/part reasoning and comparison across different images with diverse scenes, by predicting co-segmentation masks and semantic labels for both unique and common parts.

3. Method

3.1. Problem Definition

In part-focused semantic co-segmentation, the objective is to generate a set of segmentation masks for a given set of input images. Here, each mask corresponds to an input image and indicates the pixels containing either the common object across all images or the shared and distinct parts belonging to semantically similar objects (*i.e.*, intuitively comparable objects) within these images. Formally, given N_I input images $\mathbf{X}_{\text{image}} = \{\mathbf{X}_{\text{image }1}, \cdots, \mathbf{X}_{\text{image }N_I}\}$, where each $\mathbf{X}_{\text{image }i} \in \mathbb{R}^{3\times H_i \times W_i}$ has height H_i and width W_i, the goal is to train a co-segmentation model $\mathcal{F} : \mathbf{X}_{\text{image}} \mapsto \mathbf{M}$ to obtain a set of mask sets $\mathbf{M} = \{\mathbf{M}_1, \cdots, \mathbf{M}_{N_I}\}$, with each $\mathbf{M}_i = \{(\mathbf{m}_{i1}, c_{i1}), \cdots, (\mathbf{m}_{iM_j}, c_{iM_j})\}$ containing M_j masks and corresponding class labels associated with image i. Here, each binary mask $\mathbf{m}_{ik} \in \{0,1\}^{H_i \times W_i}$ assigns each pixel to a value of 1 if it covers the visual element with semantic class label c_{ik} and 0 otherwise. We denote $\mathbf{c}_i = \{c_{i1}, \cdots, c_{iM_j}\}$ as the set of class labels corresponding to image i. When generating common object or part masks across all images, we want $\bigcap_{i=1}^{M_j} \mathbf{c}_i \neq \varnothing$, whereas when obtaining unique masks, we want $\mathbf{c}_i \cap \mathbf{c}_{i'} = \varnothing \ \forall i \neq i', 1 \leq i, i' \leq M_j$. To learn a model $\mathcal{F}$ that can address this multifaceted task, we opt for an LVLM-based solution, leveraging LLMs' ability to tackle multiple tasks with a single architecture and their flexibility in input/output processing. The rest of this section details our model's architecture.

3.2. CALICO Architecture

CALICO is an LVLM designed to output multiple segmentation masks per image for a series of images that address commonalities and differences across images. In addition to its core functionality, our model incorporates modules aimed at integrating semantic correspondences between similar objects across images, alongside multi-image understanding and segmentation. As illustrated in Fig. 2, the architecture is composed of a Vicuna-based large language model $\mathcal{M}$ in tandem with a vision module $\mathcal{I}$ and a vision-to-language projection layer, which projects image embeddings from $\mathcal{I}$ into $\mathcal{M}$'s language space.

Interleaved Vision-Language Inputs. CALICO is trained to understand interleaved multi-image inputs. Given N_I input images $\mathbf{X}_{\text{image}} \in \mathbb{R}^{N_I \times 3 \times H \times W}$ and a vision module $\mathcal{I} : \mathbb{R}^{3\times H\times W} \mapsto \mathbb{R}^{S_I \times D_I}$, we obtain image embeddings $\mathbf{X}_{\text{embed}} \in \mathbb{R}^{N_I \times S_I \times D_I}$ by $\mathbf{X}_{\text{embed}} = \mathcal{I}(\mathbf{X}_{\text{image}})$, where S_I and D_I are the image embedding sequence length and hidden size, respectively. We then project these embeddings into the language model space with hidden size D via $f_{\text{image}} : \mathbb{R}^{D_I} \mapsto \mathbb{R}^D$ to get

$$\mathbf{I}^0 = f_{\text{image}}(\mathbf{X}_{\text{embed}}) = \{\mathbf{I}_1^0, \cdots, \mathbf{I}_{N_I}^0\} \in \mathbb{R}^{N_I \times S_I \times D}. \quad (1)$$

The final input $\mathbf{T}^0 \in \mathbb{R}^{S\times D}$ into $\mathcal{M}$ is composed of interleaved text and image tokens $\mathbf{T}^0 = \{\mathbf{t}_1^0, \cdots, \mathbf{v}_{11}^0, \cdots, \mathbf{v}_{1S_I}^0, \cdots, \mathbf{t}_i^0, \cdots, \mathbf{v}_{j1}^0, \cdots, \mathbf{v}_{jS_I}^0, \cdots, \mathbf{t}_{S_T}^0\}$, where $\mathbf{t}_i^0$ is the i^{th} embedded text token at layer 0 ($1 \leq i \leq S_T$), $\mathbf{I}_j^0 = \{\mathbf{v}_{j1}^0, \cdots, \mathbf{v}_{jS_I}^0\}$ are the S_I tokens pertaining to the j^{th} image, and the superscript k of $\mathbf{T}^k$ represents the LLM output at layer k or input at layer $k+1$. The sequence length S of $\mathbf{T}^0$ thus sums up the text and image lengths, *i.e.*, $S = S_T + N_I \times S_I$. Finally, we obtain predicted outputs from the N-layered LLM by $\hat{\mathbf{T}}^N = \mathcal{M}(\mathbf{T}^0)$. In practice, we encourage the LLM to learn comprehensive multimodal understanding through prompts such as "`The <image> (IMAGE1) and <image> (IMAGE2)`

provide an overview of the pictures. Can you segment the common object in these images?", where the <image> tokens are replaced with the projected embeddings of the corresponding image. Each image is also associated with a unique identifier (*e.g.*, IMAGE1, IMAGE2) for more convenient and clear reference, avoiding any potential ambiguity from using ordinal terms to refer to individual images in multi-image settings, *e.g.*, "the first" or "the second" image.

Vision Module. Observing the effectiveness and efficiency of BLIP-2's Q-Former cross-attention mechanism [27], especially in multi-image settings [28], we propose using Q-Former in tandem with a strong CLIP vision encoder [44, 57] to extract visual embeddings from the input images. Whereas projecting CLIP embeddings directly into the language model space preserves their long sequence lengths (*e.g.*, 256 or 576 tokens [24, 46]) and thus increasing compute, Q-Former uses a much shorter set of learnable query tokens to extract visual information (*e.g.*, 32 tokens [27, 28]). Formally, our vision module $\mathcal{I}$ consists of an EVA-CLIP-g model $\mathcal{C} : \mathbb{R}^{3\times H\times W} \mapsto \mathbb{R}^{S_C\times D_C}$ and a Q-Former cross-attention module $\mathcal{Q}$ alongside a set of learnable query tokens $\mathbf{q} \in \mathbb{R}^{S_I\times D_I}$. We first pass input images $\mathbf{X}_{\text{image}}$ through the EVA-CLIP global encoder to obtain $\mathbf{X}_{\text{global}} = \mathcal{C}(\mathbf{X}_{\text{image}}) \in \mathbb{R}^{N_I\times S_C\times D_C}$. We then obtain our final visual embeddings by querying $\mathcal{Q}$ using $\mathbf{q}$ as the query and $\mathbf{X}_{\text{global}}$ as the key and value:

$$\mathbf{X}_{\text{embed}} = \mathcal{Q}(\mathbf{q}, \mathbf{X}_{\text{global}}) \in \mathbb{R}^{N_I\times S_I\times D_I}. \quad (2)$$

Pixel-Grounded Outputs. To enable pixel-level grounding, we augment the model's vocabulary with the segmentation token [SEG] and teach it to output grounding tags and , following recent work [46]. Through supervision, the model learns to ground a noun phrase associated with the following segmentation token by enclosing it in the grounding tags, which immediately precede the corresponding [SEG] token (*e.g.*, The unique parts of the objects are the seat cushion [SEG] (IMAGE1) and the back pillow [SEG] (IMAGE2).). We append the image identifiers immediately following [SEG] tokens to distinguish between tokens belonging to different images.

To transform the [SEG] tokens $\mathbf{S} = \{\mathbf{S}_1, \cdots, \mathbf{S}_{N_I}\} \subset \mathbf{T}^K$, where each $\mathbf{S}_i \in \mathbb{R}^{S_j\times D}$ corresponds to the set of S_j predicted masks associated with image i, into segmentation mask sets $\hat{\mathbf{M}} = \left\{\hat{\mathbf{M}}_1, \cdots, \hat{\mathbf{M}}_{N_I}\right\}$, our architecture incorporates a Transformer-based grounding model [23], composed of a grounding encoder $\mathcal{G}$ and a pixel decoder $\mathcal{D}$. The input images $\mathbf{X}_{\text{image}}$ are passed through the frozen encoder to obtain the grounding embeddings $\mathbf{X}_{\text{ground}}$ as the vision signal for the decoder by $\mathbf{X}_{\text{ground}} = \mathcal{G}(\mathbf{X}_{\text{image}}) \in \mathbb{R}^{N_I\times S_D\times D_D}$. For an encoded image $\mathbf{X}_{\text{ground}\,i}$, the tokens in

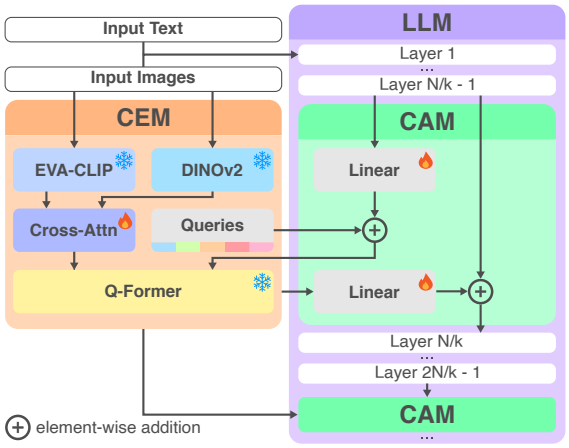

Figure 3. **Overview of our Correspondence Extraction Module and Adaptation Modules.** In CALICO, k CAMs are placed at every $\frac{N}{k}$ layers in the N-layered LLM.

$\mathbf{S}_i$ act as prompts for the finetuned pixel decoder after being passed through a projection layer $f_{\text{segmentation}} : \mathbb{R}^D \mapsto \mathbb{R}^{D_D}$. Finally, the decoder $\mathcal{D}$ produces binary segmentation masks accordingly by:

$$\hat{\mathbf{M}}_i = \mathcal{D}(\mathbf{X}_{\text{ground}\,i}, f_{\text{segmentation}}(\mathbf{S}_i)). \quad (3)$$

3.3. Correspondence Extraction Module (CEM)

Image features obtained from self-supervised Vision Transformers (ViTs) [3, 8, 41] have been shown to exhibit rich semantic information at part-level granularity across similar, yet distinct object categories [1, 3, 71]. Motivated by these findings, we design a fusion module to extract such semantic information and facilitate part correspondence learning within the model. We define $\mathcal{E}$ to be the semantic extraction process using a self-supervised ViT [3] similarly to Amir et al. [1], and obtain semantic embeddings $\mathbf{X}_{\text{semantic}} \in \mathbb{R}^{N_I\times S_S\times D_S}$, where S_S and D_S are the sequence length and hidden size of the semantic image embeddings, respectively. We then use $\mathbf{X}_{\text{semantic}}$ as the key and value for a cross-attention extraction mechanism $\mathcal{A}$ with the queried EVA-CLIP embedding $\mathbf{X}_{\text{global}}$ to get semantic-rich global embeddings $\mathbf{X}'_{\text{global}}$. This process is formalized by:

$$\mathbf{X}_{\text{global}} = \mathcal{C}(\mathbf{X}_{\text{image}}) \quad \in \mathbb{R}^{N_I\times S_C\times D_C} \quad (4)$$

$$\mathbf{X}_{\text{semantic}} = \mathcal{E}(\mathbf{X}_{\text{image}}) \quad \in \mathbb{R}^{N_I\times S_S\times D_S} \quad (5)$$

$$\mathbf{X}'_{\text{global}} = \mathcal{A}(\mathbf{X}_{\text{global}}, \mathbf{X}_{\text{semantic}}) \quad \in \mathbb{R}^{N_I\times S_C\times D_C} \quad (6)$$

This fusion process produces strong semantic embeddings $\mathbf{X}'_{\text{global}}$, which are subsequently utilized for the visual extraction process performed by the Correspondence Adaptation Modules, detailed in the next section.

3.4. Correspondence Adaptation Module (CAM)

Due to the high cost of training LLMs with billions of parameters, many works have taken advantage of adaptive

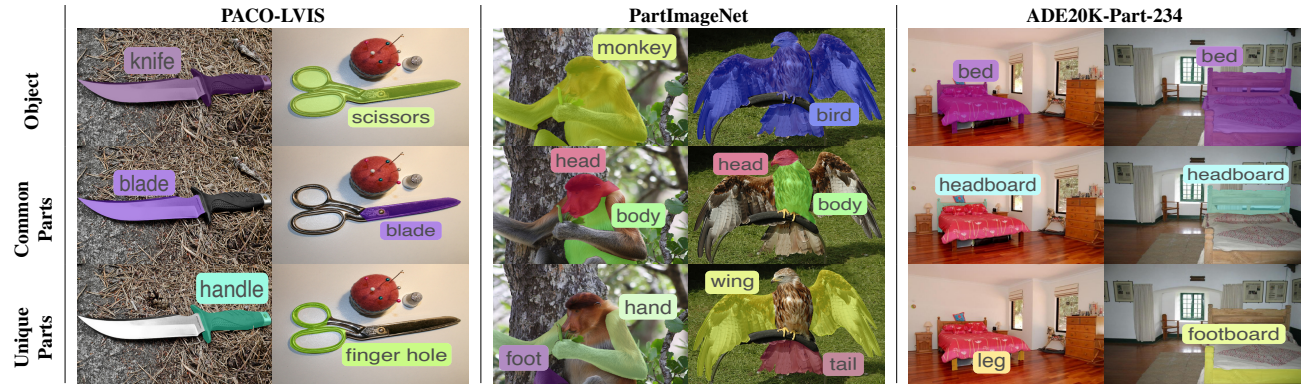

Figure 4. **Example image pairs in MIXEDPARTS with objects, common parts, and unique parts segmented and labeled.** Each column represents a different image pair, derived from a set of diverse datasets with various levels of detail, PACO, PartImageNet, and ADE20K-Part-234, covering both rigid and non-rigid objects and parts. Each image pair is displayed across 3 rows to illustrate (i) the (possibly common or different) object(s), (ii) the common object part(s), and (iii) the unique object part(s) in each pair.

modules, with sizes merely a fraction of the LLMs' original sizes, by freezing the LLMs and only training the modules for downstream tasks [17, 18, 28, 70]. These works have demonstrated strong performance and high efficiency while also avoiding the problem of catastrophic forgetting [49, 59]. In particular, the VPG-C module proposed by Li et al. [28] effectively exhibits the capability to adapt an LLM to multi-image reasoning while only accounting for 0.09% of the entire model's parameter count. Inspired by this finding, we leverage VPG-C for multi-image correspondence adaptation. Specifically, at select layers $l \in L$, we linearly project the last input token $\mathbf{t}^l_{S_T} \in \mathbb{R}^D$ via $f_{\text{adaptation}} : \mathbb{R}^D \to \mathbb{R}^{D_I}$ to get an instruction-specific guidance embedding. We then use this embedding to enrich the query tokens $\mathbf{q}$ and obtain new query embeddings by:

$$\mathbf{q}' = \mathbf{q} + f_{\text{adaptation}}\left(\mathbf{t}^l_{S_T}\right) \in \mathbb{R}^{S_I \times D_I}. \tag{7}$$

These context-guided tokens are finally used to query semantic- and context-rich visual information from $\mathbf{X}'_{\text{global}}$ from the Correspondence Extraction Module to get:

$$\mathbf{X}'_{\text{embed}} = \mathcal{Q}\left(\mathbf{q}', \mathbf{X}'_{\text{global}}\right) \in \mathbb{R}^{N_I \times S_I \times D_I}. \tag{8}$$

Finally, these embeddings are projected into the language model space via $f_{\text{integration}} : \mathbb{R}^{D_I} \mapsto \mathbb{R}^D$ and added into the visual portions of the input $\mathbf{T}^l$ to layer l of the LLM by:

$$\mathbf{I}^l_{\text{fused}} = \mathbf{I}^l + f_{\text{reintegration}}\left(\mathbf{X}'_{\text{embed}}\right) \in \mathbb{R}^{N_I \times S_I \times D}. \tag{9}$$

This process facilitates the integration of embeddings imbued with strong context and semantic information directly into CALICO. In practice, we inject our Correspondence Adaptation Modules to layers $l \in L = \{11, 22\}$, which is a third and two-thirds of the way, respectively, through a 32-layer LLM, to learn semantic correspondence at multiple granularities (*e.g.*, object and part). Ablations on different layers L in Section 5.2 validate our design choice.

3.5. Training Objective

Following past works [24, 46], the training loss $\mathcal{L}$ is composed of the next-token prediction loss $\mathcal{L}_{\text{text}}$ and the segmentation mask loss $\mathcal{L}_{\text{mask}}$. Here, $\mathcal{L}_{\text{text}}$ is a causal cross-entropy (CE) loss computed from the predicted and right-shifted ground truth tokens $\hat{\mathbf{T}}^K$ and $\mathbf{y}$, while $\mathcal{L}_{\text{mask}}$ combines a focal loss [31] and a DICE loss [40] derived from the predicted and ground truth masks $\hat{\mathbf{M}}$ and $\mathbf{M}$. The training objective can be summarized as follows:

$$\mathcal{L} = \lambda_{\text{text}}\mathcal{L}_{\text{text}} + \mathcal{L}_{\text{mask}} \tag{10}$$

$$\mathcal{L}_{\text{test}} = \text{CE}\left(\hat{\mathbf{T}}^K, \mathbf{y}\right) \tag{11}$$

$$\mathcal{L}_{\text{mask}} = \lambda_{\text{focal}}\mathcal{L}_{\text{focal}}\left(\hat{\mathbf{M}}, \mathbf{M}\right) + \lambda_{\text{Dice}}\mathcal{L}_{\text{Dice}}\left(\hat{\mathbf{M}}, \mathbf{M}\right), \tag{12}$$

where λ_{text}, λ_{focal}, and λ_{Dice} represent the respective weighting coefficients for the loss components. These objectives enable an end-to-end training process focusing on optimizing the quality of both text and mask generation outputs.

4. MIXEDPARTS Dataset

Although multi-image datasets of various scales are available, they exhibit combinations of limitations, making them unsuitable for the part-focused semantic co-segmentation task. Limitations include the absence of fine-grained masks for segmentation [19, 28, 37, 54, 55], datasets being too small or domain-specific to facilitate generalizable LVLM training despite containing localized labels [2, 12, 51, 52, 62], or the lack of part-level information altogether [43, 51]. To address these challenges and enable effective training and evaluation of our part-focused semantic co-segmentation model, we introduce MIXEDPARTS, a novel dataset curated from publicly available datasets. Figure 4 provides examples from our dataset, demonstrating its diversity in object categories and visual details. We provide an overview of the public datasets utilized as part of

MIXEDPARTS and detail the dataset construction process in Appendix A. Dataset statistics can be found in Appendix B.

5. Experiments

We evaluate CALICO's performance on the challenging MIXEDPARTS dataset, reporting the mean Intersection-over-Union (**mIoU**), Average Precision (**AP50**), and **Recall**, which assess the model's segmentation performance. To evaluate its semantic label generation capability, following existing works [7, 67], we employ Semantic Similarity (**SS**) and Semantic IoU (**S-IoU**). Implementation details and descriptions of the evaluation metrics can be found in Appendix C. We perform evaluation on ∼1K image pairs, ensuring an equal distribution of image pairs from each original dataset and maintaining equal representation across all tasks. We report performance on all three subtasks of part-focused semantic co-segmentation – common objects, common parts, unique parts – and their average, which reflects overall performance on the MIXEDPARTS test set.

Baselines. To the best of our knowledge, we represent the first effort to tackle multi-image part-focused co-segmentation with part label identification. There is thus a lack of baselines for this new task, which we rectify by designing our own baselines from strong pretrained models in the semantic segmentation literature. Our baselines include a traditional zero-shot Mask R-CNN-based method and two finetuned LVLM-based approaches: **(1) Multi-Image VLPart:** VLPart [56] is an open-vocabulary part segmentation model which can identify object parts at different granularities. It utilizes a conventional Mask R-CNN [16] with a modern Swin Transformer backbone [34] and a CLIP [44] text classifier for open-world image classification. Although VLPart cannot perform co-segmentation, due to its strong zero-shot object-part segmentation capabilities, we perform segmentation on individual images and simply examine the common and unique predictions. Our implementation of Multi-Image VLPart, which ensures fairness, is detailed in Appendix C. **(2) Multi-Image GLaMM:** GLaMM [46] is a strong single-image segmentation-based LVLM constructed for the Grounded Conversation Generation (GCG) task for multi-round pixel-grounded conversations. The architecture incorporates a Vicuna-based backbone with SAM and a novel RoIAlign-based region encoder. Since GLaMM was not trained for multi-image processing, we replicate CALICO's multi-image implementation on the GLaMM codebase and finetune the mask decoder on MIXEDPARTS for GLaMM to learn multi-image reasoning, alongside LoRA, using the exact same hyperparameters. We initialize both CALICO and Multi-Image GLaMM on GLaMM's full model weights.[1] **(3) Multi-Image LISA:** LISA [24] is an LVLM trained to perform referring segmentation, built on a Vicuna-7B backbone and SAM, similarly to GLaMM. Likewise, we implement multi-image processing for LISA by sequentially feeding the mask decoder with the encoded images and corresponding segmentation tokens to obtain segmentation masks. We initialize LISA on the most recent 7B checkpoint[2] and finetune the mask decoder alongside LoRA.

Method	mIoU	AP50	Recall	SS	S-IoU
Multi-Image VLPart [56]	12.4	42.0	34.0	58.6	45.4
Multi-Image GLaMM [46]	30.3	51.6	40.8	63.9	56.3
Multi-Image LISA [24]	30.7	52.1	42.3	66.3	59.9
CALICO (ours)	**37.1**	**57.2**	**50.5**	**76.5**	**70.2**

Table 1. **Experimental Results on MIXEDPARTS.** The first three metrics are segmentation-based, while the last two are text-based. CALICO outperforms baselines across all metrics.

5.1. Experimental Results

Table 1 presents results comparing CALICO against our baselines, Multi-Image VLPart, Multi-Image GLaMM, and Multi-Image LISA. CALICO demonstrates superior performance on all metrics compared to all zero-shot and finetuned approaches, achieving relative gains of 20.8%, 9.8%, and 19.4% on segmentation-based metrics mIoU, AP50, and Recall, respectively, and 15.4% and 17.2% on text-based metrics SS and S-IoU. Although VLPart demonstrates strong zero-shot single-image object-part segmentation performance, it struggles on our multi-image tasks, even on text-based metrics where the model has direct access to the ground truth class labels. When finetuned on MIXEDPARTS, Multi-Image GLaMM and LISA exhibit performance improvements compared to the zero-shot VLPart baseline. However, GLaMM still significantly lags behind CALICO's performance despite our model being initialized from the same weights, demonstrating the effectiveness of our proposed modules, which we further ablate in Section 5.2. We present qualitative examples in Figure 2 and additional experiments in Appendix E.

5.2. Ablations

CALICO Components. To validate the effectiveness of the individual components of CALICO, we conduct an ablation that isolates the impact of the Correspondence Extraction Module (CEM) and the Correspondence Adaptation Module (CAM). In Figure 5, we report results of CALICO without the CEM module (w/o CEM), without the CAM module (w/o CAM), or removing both (w/o CEM w/o CAM). Specifically, the CALICO variant without the CEM module utilizes only the image embeddings for fusion with the LVLM outputs' last hidden states, while the CALICO variant without the CAM module only injects DINOv2 features

[1] https://huggingface.co/MBZUAI/GLaMM-FullScope.

[2] https://huggingface.co/xinlai/LISA-7B-v1-explanatory.

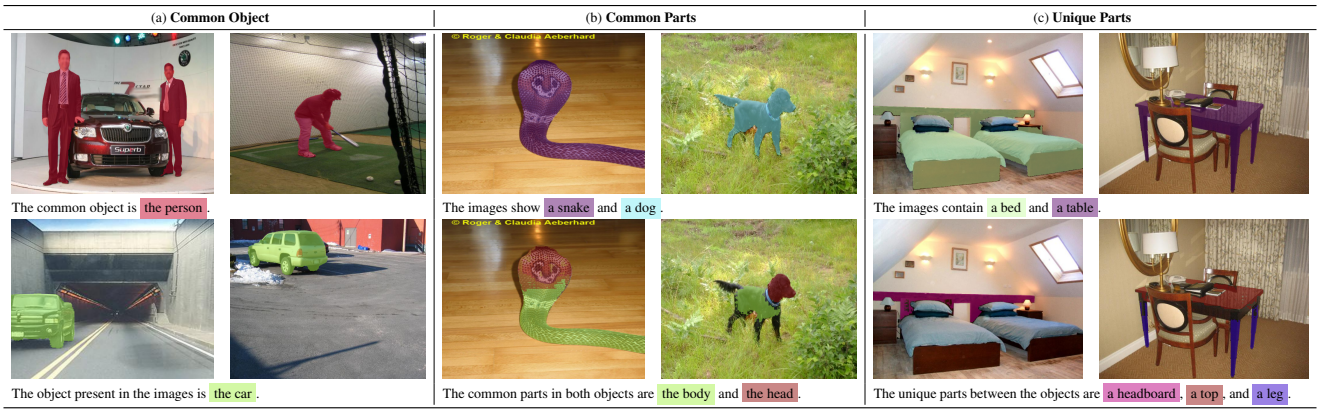

Table 2. **CALICO Qualitative Results.** Across various object types, CALICO can segment (a) common objects in the images, (b) shared parts between objects from different classes, and (c) distinct parts unique to each object.

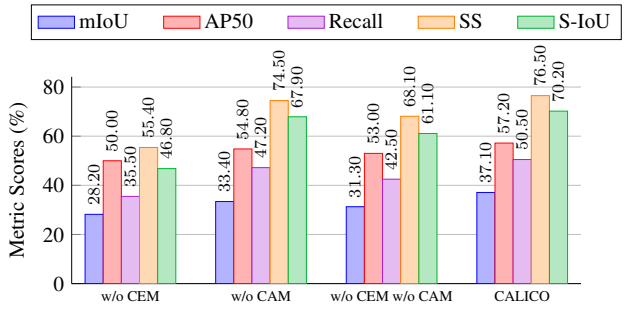

Figure 5. **Ablations on the Correspondence Extraction and Correspondence Adaptation Modules.**

without fusing them with the last hidden states.

Ablation results indicate that the proposed modules play a significant role in CALICO's ability to accurately co-segment and identify common objects, common parts, and unique parts. Specifically, removing CEM results in a notable decrease across all metrics. Similarly, excluding the CAM module leads to a degradation w.r.t. segmentation and labeling performance. CEM substantially contributes to the segmentation performance as this module enables the model to learn semantic relationships in multi-image contexts. Interestingly, keeping CAM while removing CEM results in degraded performance compared to the version without both modules. This may mean that CAM does not work well when there is no external signal from CEM, in which case it mostly duplicates the image features in the input and potentially confuses the model. However, when CEM is included, the variant without both modules underperforms the one without CAM, which in turn is outperformed by CALICO. This demonstrates that CAM is beneficial when there are external semantic signals for the part-focused semantic co-segmentation task.

CALICO Injecting CAMs. We perform further ablations to examine the efficacy of injecting 2 evenly spaced Correspondence Adaptation Modules (CAM) into our CALICO LVLM, and present results in Table 3. Past works [28, 64] have demonstrated the effectiveness of using or injecting information at the intermediate layer $\frac{K}{2}$ as guidance for learning. Since our task involves multimodal understanding at multiple granularities (object and part), we use 2 evenly spaced layers to incorporate semantic features at different levels of the LVLM's learning, encouraging the model to focus on various object-part correspondences. This results in the best segmentation performance, as shown in Table 3, compared to the 1-layer or 3-layer variants. Although injecting 3 adaptive modules interestingly improves results on our text metrics, this comes at the expense of segmentation performance; therefore, we opt for injecting 2 layers to preserve grounding performance.

Layers	mIoU	AP50	Recall	SS	S-IoU
16	33.7	54.8	47.3	73.6	66.7
11, 22 (CALICO)	**37.1**	**57.2**	**50.5**	76.5	70.2
8, 16, 24	36.4	56.7	50.2	**77.4**	**70.8**

Table 3. **Ablations on CAM layer injection.**

6. Conclusion

This paper introduces the novel task of part-focused semantic co-segmentation, which involves the segmentation of common and unique parts across multiple images, laying the groundwork for future research in enhancing the capability of Large Vision-Language Models (LVLMs) to analyze and interpret complex visual data in a granular manner. To solve this task, we propose CALICO, an LVLM that incorporates a novel correspondence extraction module and an adaptation module to handle multi-image part relationships. Experiments conducted on the newly curated MIXEDPARTS dataset, demonstrate that CALICO can effectively identify and segment common/unique parts with high accuracy, outperforming existing models.

Acknowledgments

This research is based upon work partially supported by U.S. DARPA ECOLE Program No. HR00112390062. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of DARPA or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

[1] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep ViT features as dense visual descriptors. In *Workshop Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCVW) What is Motion For?*, 2022. 2, 3, 5

[2] Dhruv Batra, Adarsh Kowdle, Devi Parikh, Jiebo Luo, and Tsuhan Chen. icoseg: Interactive co-segmentation with intelligent scribble guidance. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010. 6

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)*, 2021. 2, 3, 5

[4] Jang Hyun Cho, Philipp Krähenbühl, and Vignesh Ramanathan. Partdistillation: Learning parts from instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[5] Subhabrata Choudhury, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Unsupervised part discovery from contrastive reconstruction. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2021. 2, 3

[6] Edo Collins, Radhakrishna Achanta, and Sabine Susstrunk. Deep feature factorization for concept discovery. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 3

[7] Alessandro Conti, Enrico Fini, Massimiliano Mancini, Paolo Rota, Yiming Wang, and Elisa Ricci. Vocabulary-free image classification. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 7, 3

[8] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023. 4, 5

[9] Daan de Geus, Panagiotis Meletis, Chenyang Lu, Xiaoxiao Wen, and Gijs Dubbelman. Part-aware panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 1

[11] Xin Duan, Yan Yang, Liyuan Pan, and Xiabi Liu. Lcco: Lending clip to co-segmentation. *Pattern Recognition*, page 111252, 2024. 3

[12] Alon Faktor and Michal Irani. Co-segmentation by composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2013. 6

[13] Qingzhe Gao, Bin Wang, Libin Liu, and Baoquan Chen. Unsupervised co-part segmentation through assembly. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021. 2, 3

[14] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[15] Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jie-Neng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2022. 2, 1

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017. 3, 7

[17] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. LoRA: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 6

[18] Chengsong Huang, Qian Liu, Bill Yuchen Lin, Chao Du, Tianyu Pang, and Min Lin. Lorahub: Efficient cross-task generalization via dynamic lora composition. In *R0-FoMo: Robustness of Few-shot and Zero-shot Learning in Large Foundation Models*, 2023. 6

[19] Yupan Huang, Zaiqiao Meng, Fangyu Liu, Yixuan Su, Nigel Collier, and Yutong Lu. Sparkles: Unlocking chats across multiple images for multimodal instruction-following models. In *ICLR Workshop on Navigating and Addressing Data Problems for Foundation Models*, 2024. 6

[20] Wei-Chih Hung, Varun Jampani, Sifei Liu, Pavlo Molchanov, Ming-Hsuan Yang, and Jan Kautz. SCOPS: Self-supervised co-part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3

[21] Ruyi Ji, Dawei Du, Libo Zhang, Longyin Wen, Yanjun Wu, Chen Zhao, Feiyue Huang, and Siwei Lyu. Learning semantic neural tree for human parsing. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2020. 3

[22] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[23] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 3, 5

[24] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 3, 5, 6, 7, 4

[25] Bo Li, Zhengxing Sun, Qian Li, Yunjie Wu, and Anqi Hu. Group-wise deep object co-segmentation with co-attention recurrent neural network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

[26] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2024. 3

[27] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning (ICML)*, 2023. 5

[28] Juncheng Li, Kaihang Pan, Zhiqi Ge, Minghe Gao, Wei Ji, Wenqiao Zhang, Tat-Seng Chua, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Fine-tuning multimodal llms to follow zero-shot demonstrative instructions. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023. 5, 6, 8

[29] Xiangtai Li, Shilin Xu, Yibo Yang, Guangliang Cheng, Yunhai Tong, and Dacheng Tao. Panoptic-partformer: Learning a unified model for panoptic part segmentation. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2022. 2, 3

[30] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 1

[31] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017. 6

[32] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 36, 2024. 3

[33] Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Unsupervised part segmentation through disentangling appearance and shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[34] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 7

[35] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3

[36] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 2

[37] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. IconQA: A new benchmark for abstract diagram understanding and visual language reasoning. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021. 6

[38] Umberto Michieli and Pietro Zanuttigh. Edge-aware graph matching network for part-based semantic segmentation. *International Journal of Computer Vision*, 130(11):2797–2821, 2022. 2, 3

[39] Umberto Michieli, Edoardo Borsato, Luca Rossi, and Pietro Zanuttigh. Gmnet: Graph matching network for large scale part semantic segmentation in the wild. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2020. 3

[40] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016. 6

[41] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2023. 3, 4, 5

[42] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the Conference in Neural Information Processing Systems (NeurIPS)*, 2019. 2

[43] Shraman Pramanick, Guangxing Han, Rui Hou, Sayan Nag, Ser-Nam Lim, Nicolas Ballas, Qifan Wang, Rama Chellappa, and Amjad Almahairi. Jack of all tasks master of many: Designing general-purpose coarse-to-fine vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 6

[44] 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 *Proceedings of the International Conference on Machine Learning (ICML)*, 2021. 3, 5, 7

[45] Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. Paco: Parts and attributes of common objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 1

[46] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Erix Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *Pro-*

ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024. 2, 3, 5, 6, 7, 4

[47] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020. 2

[48] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019. 3

[49] Weijieying Ren, Xinlong Li, Lei Wang, Tianxiang Zhao, and Wei Qin. Analyzing and reducing catastrophic forgetting in parameter efficient tuning. *arXiv preprint arXiv:2402.18865*, 2024. 6

[50] Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. PixelLM: Pixel reasoning with large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 3

[51] Michael Rubinstein, Armand Joulin, Johannes Kopf, and Ce Liu. Unsupervised joint object discovery and segmentation in internet images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 6

[52] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2006. 6

[53] Rishubh Singh, Pranav Gupta, Pradeep Shenoy, and Ravikiran Sarvadevabhatla. Float: Factorized learning of object attributes for improved multi-object multi-part scene parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[54] Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. A corpus of natural language for visual reasoning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL Volume 2: Short Papers)*, 2017. 6

[55] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019. 6

[56] Peize Sun, Shoufa Chen, Chenchen Zhu, Fanyi Xiao, Ping Luo, Saining Xie, and Zhicheng Yan. Going denser with open-vocabulary part segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 3, 7, 2, 4

[57] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 5

[58] Robert van der Klis, Stephan Alaniz, Massimiliano Mancini, Cassio F Dantas, Dino Ienco, Zeynep Akata, and Diego Marcos. Pdisconet: Semantically consistent part discovery for fine-grained recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2

[59] Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuan-Jing Huang. Orthogonal subspace learning for language model continual learning. In *Findings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. 6

[60] Cong Wei, Haoxian Tan, Yujie Zhong, Yujiu Yang, and Lin Ma. LaSagnA: Language-based segmentation assistant for complex queries. *arXiv preprint arXiv:2404.08506*, 2024. 2, 3

[61] Meng Wei, Xiaoyu Yue, Wenwei Zhang, Shu Kong, Xihui Liu, and Jiangmiao Pang. Ov-parts: Towards open-vocabulary part segmentation. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2024. 2, 1

[62] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. 6

[63] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2

[64] Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. Deebert: Dynamic early exiting for accelerating bert inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2246–2251, 2020. 8

[65] Lu Yang, Qing Song, Zhihui Wang, and Ming Jiang. Parsing r-cnn for instance-level human analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[66] Lu Yang, Qing Song, Zhihui Wang, Mengjie Hu, Chun Liu, Xueshi Xin, Wenhe Jia, and Songcen Xu. Renovating parsing r-cnn for accurate multiple human parsing. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2020. 3

[67] Yuqian Yuan, Wentong Li, Jian Liu, Dongqi Tang, Xinjie Luo, Chi Qin, Lei Zhang, and Jianke Zhu. Osprey: Pixel understanding with visual instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)*, 2024. 2, 3, 7

[68] Ao Zhang, Liming Zhao, Chen-Wei Xie, Yun Zheng, Wei Ji, and Tat-Seng Chua. Next-chat: An lmm for chat, detection and segmentation. *arXiv preprint arXiv:2311.04498*, 2023. 2, 3

[69] Chi Zhang, Guankai Li, Guosheng Lin, Qingyao Wu, and Rui Yao. Cyclesegnet: Object co-segmentation with cycle refinement and region correspondence. *IEEE Transactions on Image Processing*, 30:5652–5664, 2021. 3

[70] Jinghan Zhang, Junteng Liu, Junxian He, et al. Composing parameter-efficient modules with arithmetic operation. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 6

[71] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan

Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2024. 5

[72] Kaihua Zhang, Jin Chen, Bo Liu, and Qingshan Liu. Deep object co-segmentation via spatial-semantic network modulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. 3

[73] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 1

[74] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019. 2, 1

[75] Tianfei Zhou, Wenguan Wang, Si Liu, Yi Yang, and Luc Van Gool. Differentiable multi-granularity human representation learning for instance-aware human semantic parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[76] Tianfei Zhou, Yi Yang, and Wenguan Wang. Differentiable multi-granularity human parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023. 3

[77] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2024. 3

CALICO: Part-Focused Semantic Co-Segmentation with Large Vision-Language Models

Supplementary Material

A. MIXEDPARTS

A.1. Single-Image Part Segmentation Datasets

To construct a robust dataset for part-focused semantic co-segmentation, we carefully curate generalizable and diverse data at various levels of detail. This entails selecting datasets that cover a wide range of objects and parts, both rigid (*e.g.*, utensils, vehicles, scissors) and non-rigid (*e.g.*, animals, humans), while not being too domain-specific (*e.g.*, birds or celebrities' faces), which could potentially encourage overfitting or introduce bias in training. Therefore, we select the following datasets to construct MIXEDPARTS:

- **PartImageNet** [15] is a high-quality part-focused extension of ImageNet [10] covering a variety of object classes with mostly animals (*e.g.* bird, fish, *etc.*) to facilitate non-rigid part understanding. Each image in PartImageNet contains only one foreground object, which can encourage the model to focus on important foreground objects when comparing two images. Prior to use, we make modifications to the original dataset to enhance generalizability, as detailed in Appendix D.
- **ADE20K-Part234** [61] is a revised version of the ADE20K scene parsing dataset [73, 74] with an emphasis on object parts. The original ADE20K dataset encompasses a wide variety of scenes, including indoor, outdoor, and urban environments, which naturally lends itself to more complex visual signals covering multiple objects in contrast to PartImageNet. However, less than 15% of the dataset contains part annotations. In addition, some part labels are too granular, which may encourage overfitting while not being too beneficial for general part understanding (*e.g.* "table stretcher" and "table h-stretcher"). To amend these disadvantages, Wei et al. [61] introduces ADE20K-Part234, a clean, part-focused version of ADE20K for improved part analysis.
- **PACO-LVIS** [45], likewise, is a component-centric version of LVIS [14] that is based on COCO [30] and focuses on diverse common everyday objects, further contributing to the diversity of object categories in MIXEDPARTS. PACO contains an extensive list of object-part categories as well as complex images with multiple objects and parts, providing finer granularity for part understanding.

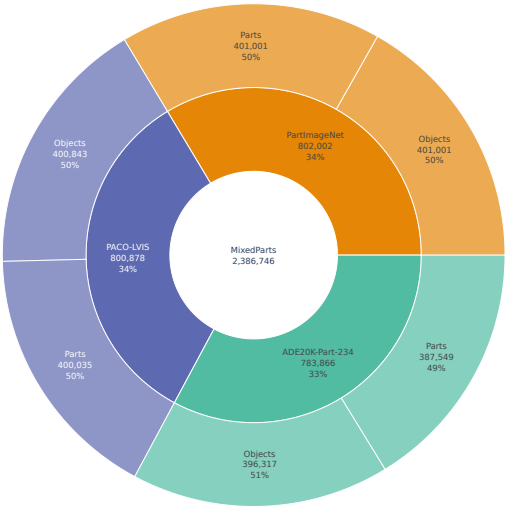

Figure 6. **MIXEDPARTS Dataset Overview.**

A.2. Constructing MIXEDPARTS

To build MIXEDPARTS, we first select 1,885 pairs of object categories across all 3 datasets that have at least one common part label (*e.g.* "armchair's seat" and "swivel chair's seat") to ensure annotation availability. However, due to the flexibility of natural language, the same words may be used to describe parts of objects that are not typically intuitively comparable. For example, even though both a bus and a microwave oven may have a door as a constituent part, they are not commonly compared. Therefore, we manually curate intuitively comparable object pairs from all possible pairs, resulting in 964 pairs of categories across all 3 datasets.

With the object pairings available, we pair up individual images corresponding to our common object, common part, and unique part localization subtasks. For common object parts, we select images that have at least a common visible object and/or object part. However, since different object classes can share the same parts, we also include images of logically comparable objects of different classes, for instance, a chair and an ottoman (both seating furniture) or an airplane and a bird (both having wings and usually compared as flying objects). We ensure stratification of object categories in MIXEDPARTS to reflect the data distribution in the original datasets.

Statistic \ Task	Common Objects	Common Parts	Unique Parts
Total # Pairs	1,257,247	1,724,937	5,568,099
Average # Pairs/Sample	1.049	2.019	5.272
Maximum # Pairs/Sample	5	28	72
Median # Pairs/Sample	1	2	3

Table 4. **MIXEDPARTS statistics** on # pairs of objects/parts. Each sample in MIXEDPARTS corresponds to an image pair. For unique parts, pair corresponds to one unique part present in a single image instead of both images, thus explaining the higher number of instances.

B. MIXEDPARTS Dataset Statistics

We provide details for the datasets used in constructing MIXEDPARTS. Figure 6 and Table 4 offer a comprehensive overview of the dataset, illustrating its structure and the distribution of object and part pairings. The inner circle of the donut chart represents the entire dataset, comprising a total of 2,386,746 samples (image pairs), segmented into three primary sources: PartImageNet, PACO-LVIS, and ADE20K-Part234, each contributing roughly a third to the total dataset. The outer ring of the chart further divides these sources into categories of objects and parts, with each segment roughly occupying half of its respective section, ensuring a balanced representation of various part-object relationships. In MIXEDPARTS, there are 1,257,247 pairs of common objects, 1,724,937 pairs of common parts, and 5,568,099 unique parts. On average, each sample comprises approximately 1 common object, 2 common parts, and 5.3 unique parts. The maximum number of pairs per sample is 5 for common objects, 28 for common parts, and 72 for unique parts, indicating a wide range in the complexity of the sample pairings. The median number of pairs per sample is 1 for common objects, 2 for common parts, and 3 for unique parts, reflecting that the majority of samples contain a single pair of common objects and more than 1 common/unique part, since one object pair may consist more than 1 common and unique parts. MIXEDPARTS consists of 141 object categories and 199 part categories covering a wide variety of objects and parts with 69,911 object instances and 159,140 part instances. Figures 8-9 present the top-30 object and part distributions in the dataset, while Figure 10 and Figure 11 present the distributions of top-30 common and unique parts, respectively.

C. Additional Experimental Setup Details

C.1. Implementation Details

CALICO. We use PyTorch [42] to implement and optimize CALICO and DeepSpeed [47] for efficient training. We initialize our model on GLaMM's [46] FullScope checkpoint.[3] All training is conducted on four NVIDIA A40 GPUs with 48GB memory. LoRA layers were set with a rank of 8 and a scaling factor of 16. We train for 10 epochs, each comprising 500 steps, using the AdamW optimizer [36] with 3e-4 initial learning rate and beta coefficients set to 0.9 and 0.95, respectively. We employ a dropout rate of 0.05, 1.0 gradient clipping, and set the batch size to 4. The loss coefficients are set to $\lambda_{\text{text}} = 1.0$, $\lambda_{\text{focal}} = 2.0$, and $\lambda_{\text{Dice}} = 0.5$.

Baselines. We provide implementation details for our zero-shot VLPart baseline [56], as well as the fine-tuned LISA [24] and GLaMM [46] baselines. For VLPart, we use the default parameters from the official repository[4], including the detection confidence threshold of 0.7 (*i.e.*, retaining predictions with confidence scores above 0.7) to initialize the model. For a given two-image data sample, we perform zero-shot inference on both images and analyze all predicted object and part masks. For common objects, we compare the sets of predicted objects across the images. For common and unique parts, we aggregate the sets of all predicted parts and derive the associated objects from these parts. For GLaMM and LISA, which are not natively designed to handle multiple images, we adapt the code to distinguish segmentation tokens belonging to different images. This is achieved by appending image identifiers to the `[SEG]` tokens, *e.g.*, `(IMAGE1)`, enabling the models to process the image-specific token sets separately.

C.2. Evaluation Metrics

Our evaluation consists of two parts: 1) evaluating the segmentation performance and 2) assessing the performance in generating semantic labels for objects and parts. To evaluate the performance of models on segmentation tasks, we employ mean Intersection over Union (**mIoU**), a widely used metric that effectively measures how much the predicted segmentation masks overlap with the ground truth masks across a dataset. We also employ Average Precision at a 50% IoU threshold (**AP50**), a metric for evaluating segmentation masks that measures the average precision across all recall levels at a fixed IoU threshold. AP50 considers a predicted mask correct if its IoU with the ground truth mask is at least 50%, summarizing model performance at

[3] https://huggingface.co/MBZUAI/GLaMM-FullScope.

[4] https://github.com/facebookresearch/VLPart.

this specific threshold. Additionally, following recent work [46], we leverage a mask **Recall** metric, which evaluates region-specific grounding by using a two-phase validation approach based on IoU and SentenceBERT [48] similarity thresholds. Specifically, given a ground truth match, we choose the prediction with the highest IoU surpassing a threshold of 50% that also has a text similarity score higher than 50% as a True Positive for recall computation.

To evaluate the performance of models in predicting semantic labels for the segmented objects and parts, we compute Semantic Similarity and Semantic IoU. Semantic Similarity (**SS**) is the semantic similarity between predicted and true labels in the SentenceBERT [48] embedding space, following past works [7, 67], while Semantic IoU (S-IoU) computes the word overlap between predicted and ground truth labels, *i.e.*,

$$\text{S-IoU} = \frac{1}{N}\sum_{i=1}^{N}\frac{|V(y_i) \cap V(\hat{y}_i)|}{|V(y_i) \cup V(\hat{y}_i)|},$$

where $V(y)$ represents the set of words comprising label y.

D. PartImageNet Modifications

While PartImageNet provides high-quality segmentation masks across a diverse range of object classes, its categories are often too abstract and not commonly used in natural language due to their broad scope. For instance, PartImageNet classifies all four-legged animals under the "quadruped" supercategory. Although technically accurate, such classifications are too generic for practical use in everyday language. To address this, we manually selected the most commonly used object class associated with each category provided by ImageNet. The dataset includes WordNet synset IDs (*e.g.*, "n02071294") that correspond to specific object classes. Using these synset IDs, we extracted the hierarchical path of each class within the WordNet graph. For example, the synset ID "n02071294" maps to the following WordNet path: living_thing → organism → animal → chordate → vertebrate → mammal → placental → aquatic_mammal → cetacean → whale → toothed_whale → dolphin → killer_whale. From this path, we selected *whale* as the representative object category for this class. We repeat this process for all object classes provided by PartImageNet. The object classes we use alongside the original supercategory in parentheses are shown in Table 6.

E. Additional Experiments

E.1. Computational Efficiency Evaluation

For multi-image tasks, it is essential that the model is efficient so that inference and training do not consume more time as the number of images increases. Therefore, we opt to use Q-Former to query image embeddings, resulting in only 32 tokens per image, which is 8 times fewer tokens than LISA (256 tokens) and 18 times fewer tokens than GLaMM (576 tokens), reducing TFLOPs over LISA by 42.84% and over GLaMM by 22.62% (LISA:19.63 TFLOPs *vs*. GLaMM:14.5 TFLOPs *vs*. CALICO:11.22 TFLOPs). This corresponds to an $\sim$1.75$\times$speed-up and 19.39% relative performance gain over LISA. These results demonstrate that CALICO achieves significant improvements in both computational efficiency and performance gains compared to baselines.

E.2. Per-Task Experimental Results

In Table 5, we report experimental results decomposed into the 3 MIXEDPARTS subtasks (common objects, common parts, and unique parts). The results delineate the incremental difficulty of the tasks by decreasing scores across all models, with common objects being the easiest task, followed by common parts, and finally unique parts as the hardest task. CALICO demonstrates superior performance across all 3 tasks, with large leaps in scores across all metrics compared to the next best baseline.

E.3. Low-Tail Performance

We show results for low-tail analysis of CALICO's performance in Figure 7. We compute the model's average recall across different object-part frequency levels on the three MIXEDPARTS subtasks. We observe no clear correlation between object-part frequencies and model performance. This highlights CALICO's robust performance across diverse object-part classes in the MIXEDPARTS image sets.

Task	Method	mIoU	AP50	Recall	SS	S-IoU
Common Objects	Multi-Image VLPart [56]	15.8	41.3	44.3	57.8	53.7
	Multi-Image GLaMM [46]	49.2	61.6	59.4	73.6	72.0
	Multi-Image LISA [24]	49.6	62.8	60.4	78.2	76.9
	CALICO (ours)	**58.3**	**68.8**	**68.9**	**88.5**	**87.6**
Common Parts	Multi-Image VLPart [56]	15.5	43.4	28.4	53.9	38.8
	Multi-Image GLaMM [46]	24.8	45.5	33.7	58.1	46.4
	Multi-Image LISA [24]	26.2	46.8	37.3	61.2	51.9
	CALICO (ours)	**30.3**	**50.1**	**42.9**	**68.4**	**57.9**
Unique Parts	Multi-Image VLPart [56]	6.0	41.4	29.4	64.1	43.7
	Multi-Image GLaMM [46]	16.8	47.6	29.4	60.0	50.5
	Multi-Image LISA [24]	16.3	46.8	29.1	59.6	50.9
	CALICO (ours)	**22.8**	**52.7**	**39.7**	**72.7**	**65.1**
MIXEDPARTS	Multi-Image VLPart [56]	12.4	42.0	34.0	58.6	45.4
	Multi-Image GLaMM [46]	30.3	51.6	40.8	63.9	56.3
	Multi-Image LISA [24]	30.7	52.1	42.3	66.3	59.9
	CALICO (ours)	**37.1**	**57.2**	**50.5**	**76.5**	**70.2**

Table 5. **Per-Task Experimental Results on MIXEDPARTS**. The first three metrics are segmentation-based while the last two are text-based. CALICO surpasses baselines across all three MIXEDPARTS tasks (common objects, common parts, and unique parts).

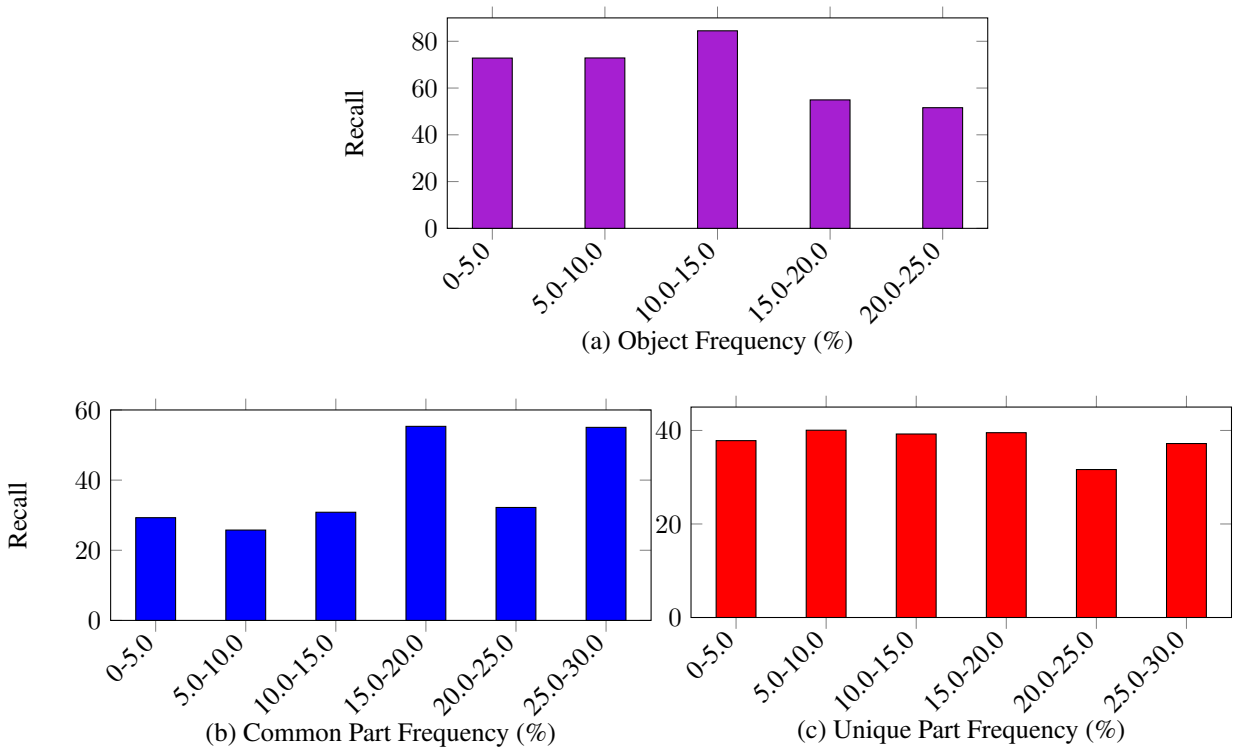

Figure 7. **Low-tail analysis of the three subtasks in MIXEDPARTS**. (a), (b), and (c) demonstrate CALICO's performance across varying object and part frequency levels, showing CALICO remains robust under different object-part frequencies.

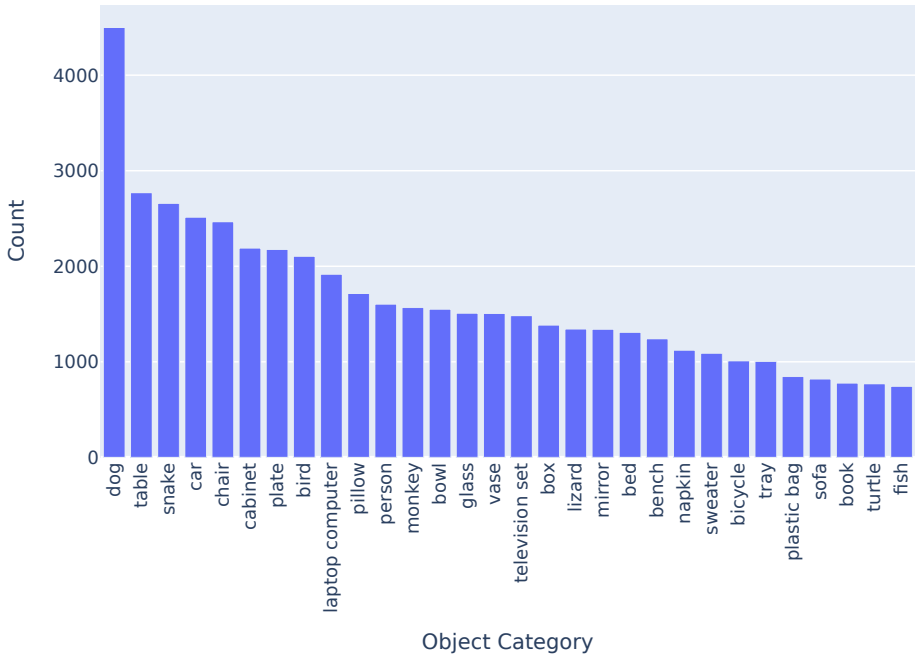

Figure 8. **MIXEDPARTS: Object Instance Distribution.**

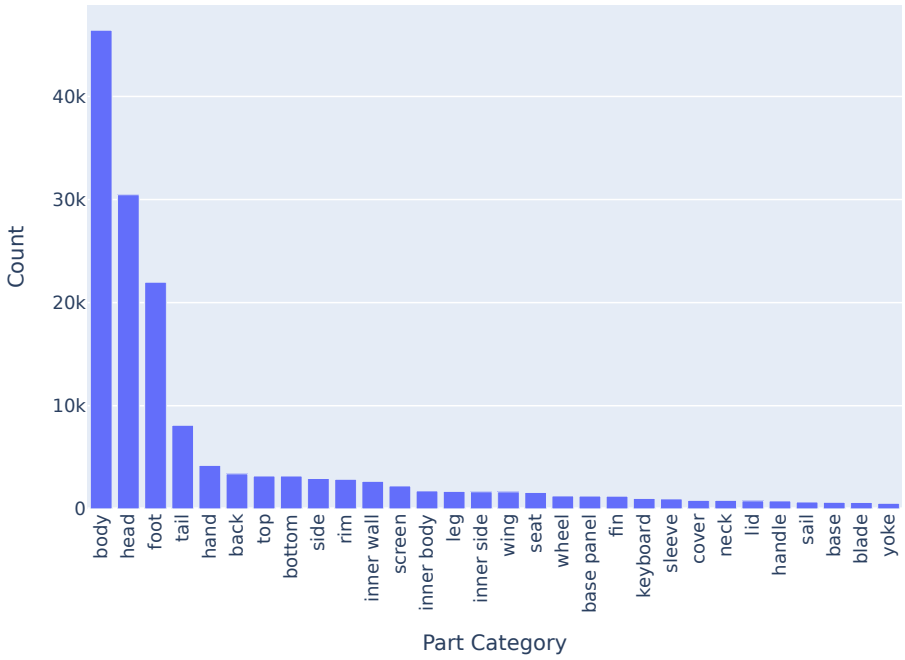

Figure 9. **MIXEDPARTS: Part Instance Distribution.**

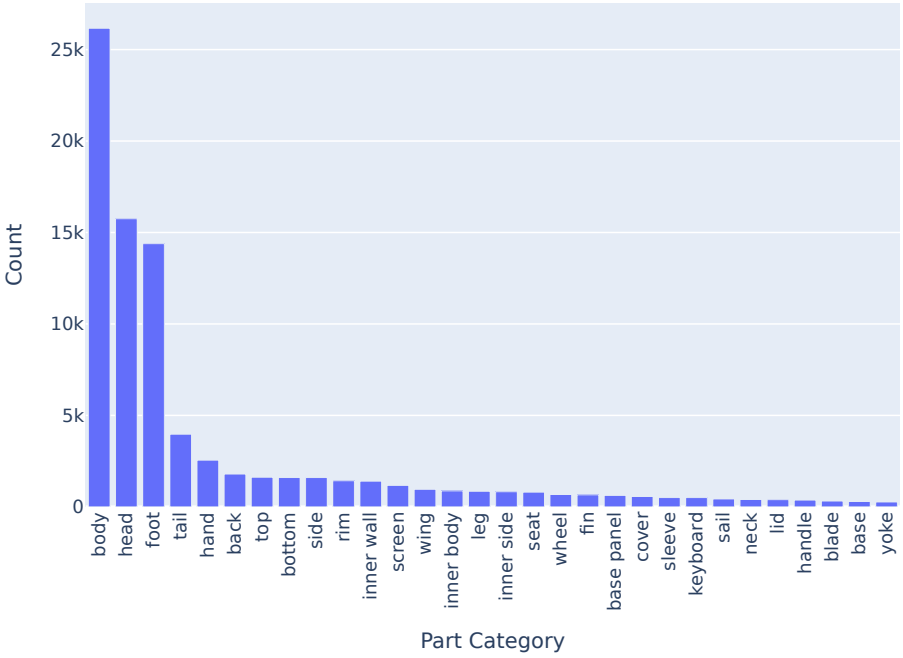

Figure 10. **MIXEDPARTS: Common Parts Instance Distribution.**

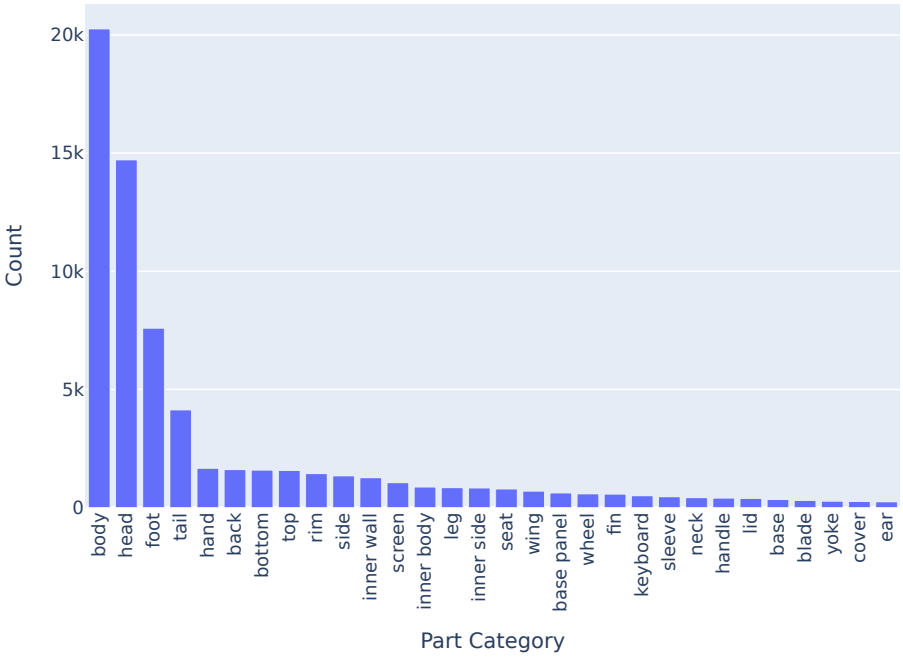

Figure 11. **MIXEDPARTS: Unique Parts Instance Distribution.**

Dataset	Object	Parts
ADE20K-Part234	airplane	door, fuselage, landing gear, propeller, stabilizer, turbine engine, wing
ADE20K-Part234	armchair	apron, arm, back, back pillow, leg, seat, seat base
ADE20K-Part234	bed	footboard, headboard, leg, side rail
ADE20K-Part234	bench	arm, back, leg, seat
ADE20K-Part234	bookcase	door, drawer, front, side
ADE20K-Part234	bus	bumper, door, headlight, license plate, logo, mirror, wheel, window, wiper
ADE20K-Part234	cabinet	door, drawer, front, shelf, side, skirt, top
ADE20K-Part234	car	bumper, door, headlight, hood, license plate, logo, mirror, wheel, window, wiper
ADE20K-Part234	chair	apron, arm, back, base, leg, seat, skirt, stretcher
ADE20K-Part234	chandelier	arm, bulb, canopy, chain, cord, highlight, light source, shade
ADE20K-Part234	chest of drawers	apron, door, drawer, front, leg
ADE20K-Part234	clock	face, frame
ADE20K-Part234	coffee table	leg, top
ADE20K-Part234	computer	computer case, keyboard, monitor, mouse
ADE20K-Part234	cooking stove	burner, button panel, door, drawer, oven, stove
ADE20K-Part234	desk	apron, door, drawer, leg, shelf, top
ADE20K-Part234	dishwasher	button panel, handle, skirt
ADE20K-Part234	door	door frame, handle, knob, panel
ADE20K-Part234	fan	blade, canopy, tube
ADE20K-Part234	glass	base, bowl, opening, stem
ADE20K-Part234	kitchen island	door, drawer, front, side, top
ADE20K-Part234	lamp	arm, base, canopy, cord, highlight, light source, pipe, shade, tube
ADE20K-Part234	light	aperture, canopy, diffusor, highlight, light source, shade
ADE20K-Part234	microwave	button panel, door, front, side, top, window
ADE20K-Part234	minibike	license plate, mirror, seat, wheel
ADE20K-Part234	ottoman	back, leg, seat
ADE20K-Part234	oven	button panel, door, drawer, top
ADE20K-Part234	person	arm, back, foot, gaze, hand, head, leg, neck, torso
ADE20K-Part234	pool table	bed, leg, pocket
ADE20K-Part234	refrigerator	button panel, door, drawer, side
ADE20K-Part234	sconce	arm, backplate, highlight, light source, shade
ADE20K-Part234	shelf	door, drawer, front, shelf
ADE20K-Part234	sink	bowl, faucet, pedestal, tap, top
ADE20K-Part234	sofa	arm, back, back pillow, leg, seat base, seat cushion, skirt
ADE20K-Part234	stool	leg, seat
ADE20K-Part234	swivel chair	back, base, seat, wheel
ADE20K-Part234	table	apron, drawer, leg, shelf, top, wheel
ADE20K-Part234	television receiver	base, buttons, frame, keys, screen, speaker
ADE20K-Part234	toilet	bowl, cistern, lid
ADE20K-Part234	traffic light	housing, pole
ADE20K-Part234	truck	bumper, door, headlight, license plate, logo, mirror, wheel, window
ADE20K-Part234	van	bumper, door, headlight, license plate, logo, mirror, taillight, wheel, window, wiper
ADE20K-Part234	wardrobe	door, drawer, front, leg, mirror, top
ADE20K-Part234	washer	button panel, door, front, side
PACO-LVIS	basket	base, bottom, cover, handle, inner side, rim, side
PACO-LVIS	belt	bar, buckle, end tip, frame, hole, loop, prong, strap
PACO-LVIS	bench	arm, back, leg, seat, stretcher, table top
PACO-LVIS	bicycle	basket, down tube, fork, gear, handlebar, head tube, pedal, saddle, seat stay, seat tube, stem, top tube, wheel
PACO-LVIS	blender	base, blade, cable, cover, cup, food cup, handle, inner body, seal ring, spout, switch, vapour cover
PACO-LVIS	book	cover, page
PACO-LVIS	bottle	base, body, bottom, cap, capsule, closure, handle, heel, inner body, label, neck, punt, ring, shoulder, sipper, spout, top
PACO-LVIS	bowl	base, body, bottom, inner body, rim
PACO-LVIS	box	bottom, inner side, lid, side
PACO-LVIS	broom	brush, brush cap, handle, lower bristles, ring, shaft
PACO-LVIS	bucket	base, body, bottom, cover, handle, inner body, loop, rim
PACO-LVIS	calculator	body, key
PACO-LVIS	can	base, body, bottom, inner body, lid, pull tab, rim, text
PACO-LVIS	car	antenna, bumper, fender, grille, handle, headlight, hood, logo, mirror, rim, roof, runningboard, seat, sign, splashboard, steeringwheel, taillight, tank, trunk, turnsignal, wheel, window, windowpane, windshield, wiper
PACO-LVIS	carton	bottom, cap, inner side, lid, side, tapering top, text, top
PACO-LVIS	cellular telephone	back cover, bezel, button, screen
PACO-LVIS	chair	apron, arm, back, base, leg, rail, seat, skirt, spindle, stile, stretcher, swivel, wheel
PACO-LVIS	clock	base, cable, case, decoration, finial, hand, pediment
PACO-LVIS	crate	bottom, handle, inner side, lid, side
PACO-LVIS	cup	base, handle, inner body, rim
PACO-LVIS	dog	body, ear, eye, foot, head, leg, neck, nose, tail, teeth
PACO-LVIS	drill	body, handle
PACO-LVIS	drum	base, body, cover, head, inner body, loop, lug, rim
PACO-LVIS	earphone	cable, ear pads, headband, housing, slider
PACO-LVIS	fan	base, blade, bracket, canopy, fan box, light, logo, motor, pedestal column, rod, string
PACO-LVIS	glass	base, body, bottom, inner body, rim
PACO-LVIS	guitar	back, body, bridge, fingerboard, headstock, hole, key, pickguard, side, string
PACO-LVIS	hammer	face, grip, handle, head
PACO-LVIS	handbag	base, body, bottom, handle, inner body, rim, zip
PACO-LVIS	hat	inner side, logo, pom pom, rim, strap, visor
PACO-LVIS	helmet	face shield, inner side, logo, rim, strap, visor
PACO-LVIS	jar	base, body, bottom, cover, handle, inner body, lid, rim, sticker, text
PACO-LVIS	kettle	base, body, cable, handle, inner body, lid, spout, switch
PACO-LVIS	knife	blade, handle
PACO-LVIS	ladder	foot, rail, step, top cap
PACO-LVIS	lamp	base, bulb, cable, finial, pipe, shade, shade cap, shade inner side, switch
PACO-LVIS	laptop computer	back, base panel, cable, camera, keyboard, logo, screen, touchpad
PACO-LVIS	microwave oven	control panel, dial, door handle, inner side, side, time display, top, turntable

Table 6. **Object and part taxonomies by dataset** (continued in next page).

Dataset	Object	Parts
PACO-LVIS	mirror	frame
PACO-LVIS	mouse	body, left button, logo, right button, scroll wheel, side button, wire
PACO-LVIS	mug	base, body, bottom, drawing, handle, inner body, rim, text
PACO-LVIS	newspaper	text
PACO-LVIS	pan	base, bottom, handle, inner side, lid, rim, side
PACO-LVIS	pen	barrel, cap, clip, grip, tip
PACO-LVIS	pencil	body, eraser, ferrule, lead
PACO-LVIS	pillow	embroidery
PACO-LVIS	pipe	colied tube, nozzle, nozzle stem
PACO-LVIS	plastic bag	body, handle, hem, inner body, text
PACO-LVIS	plate	base, body, bottom, inner wall, rim
PACO-LVIS	pliers	blade, handle, jaw, joint
PACO-LVIS	remote control	back, button, logo
PACO-LVIS	scarf	body, fringes
PACO-LVIS	scissors	blade, finger hole, handle, screw
PACO-LVIS	screwdriver	handle, shank, tip
PACO-LVIS	shoe	backstay, eyelet, heel, insole, lace, lining, outsole, quarter, throat, toe box, tongue, vamp, welt
PACO-LVIS	slipper	insole, lining, outsole, strap, toe box, vamp
PACO-LVIS	soap	base, body, bottom, cap, capsule, closure, handle, label, neck, punt, push pull cap, ring, shoulder, sipper, spout, top
PACO-LVIS	sponge	rough surface
PACO-LVIS	spoon	bowl, handle, neck, tip
PACO-LVIS	stool	footrest, leg, seat, step
PACO-LVIS	sweater	body, cuff, hem, neckband, shoulder, sleeve, yoke
PACO-LVIS	table	apron, drawer, inner body, leg, rim, shelf, stretcher, top, wheel
PACO-LVIS	tape	roll
PACO-LVIS	telephone	back cover, bezel, button, screen
PACO-LVIS	television set	base, bottom, button, side, top
PACO-LVIS	tissue paper	roll
PACO-LVIS	towel	body, border, hem, terry bar
PACO-LVIS	trash can	body, bottom, hole, inner body, label, lid, pedal, rim, wheel
PACO-LVIS	tray	base, bottom, inner side, inner wall, outer side, rim
PACO-LVIS	vase	body, foot, handle, mouth, neck
PACO-LVIS	wallet	flap, inner body
PACO-LVIS	watch	buckle, case, dial, hand, lug, strap, window
PACO-LVIS	wrench	handle, head
PartImageNet	airplane (aeroplane)	body, engine, head, tail, wing
PartImageNet	alligator (reptile)	body, foot, head, tail
PartImageNet	antelope (quadruped)	body, foot, head, tail
PartImageNet	ape (biped)	body, foot, hand, head, tail
PartImageNet	badger (quadruped)	body, foot, head, tail
PartImageNet	bear (quadruped)	body, foot, head, tail
PartImageNet	bird (bird)	body, foot, head, tail, wing
PartImageNet	boat (boat)	body, sail
PartImageNet	camel (quadruped)	body, foot, head, tail
PartImageNet	cat (quadruped)	body, foot, head, tail
PartImageNet	cheetah (quadruped)	body, foot, head, tail
PartImageNet	cougar (quadruped)	body, foot, head, tail
PartImageNet	crocodile (reptile)	body, foot, head, tail
PartImageNet	dog (quadruped)	body, foot, head, tail
PartImageNet	fish (fish)	body, fin, head, tail
PartImageNet	fox (quadruped)	body, foot, head, tail
PartImageNet	frog (reptile)	body, foot, head, tail
PartImageNet	goat (quadruped)	body, foot, head, tail
PartImageNet	leopard (quadruped)	body, foot, head, tail
PartImageNet	lizard (reptile)	body, foot, head, tail
PartImageNet	mink (quadruped)	body, foot, head, tail
PartImageNet	monkey (biped)	body, foot, hand, head, tail
PartImageNet	otter (quadruped)	body, foot, head, tail
PartImageNet	ox (quadruped)	body, foot, head, tail
PartImageNet	panda (quadruped)	body, foot, head, tail
PartImageNet	polecat (quadruped)	body, foot, head, tail
PartImageNet	shark (fish)	body, fin, head, tail
PartImageNet	sheep (quadruped)	body, foot, head, tail
PartImageNet	snake (snake)	body, head
PartImageNet	squirrel (quadruped)	body, foot, head, tail
PartImageNet	swine (quadruped)	body, foot, head, tail
PartImageNet	tiger (quadruped)	body, foot, head, tail
PartImageNet	turtle (reptile)	body, foot, head, tail
PartImageNet	water buffalo (quadruped)	body, foot, head, tail
PartImageNet	weasel (quadruped)	body, foot, head, tail
PartImageNet	whale (fish)	body, fin, head, tail
PartImageNet	wolf (quadruped)	body, foot, head, tail

Table 6. **Object and part taxonomies by dataset** (continued).