Knowledge-Guided Part Segmentation

Xuejian Gou, Fang Liu*, Licheng Jiao, Shuo Li, Lingling Li, Hao Wang, Xu Liu, Puhua Chen, Wenping Ma
Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education,
International Research Center for Intelligent Perception and Computation,
Joint International Research Laboratory of Intelligent Perception and Computation,
School of Artificial Intelligence, Xidian University, Xi'an 710071, China
xjgou13@163.com, f63liu@163.com

Abstract

In real-world scenarios, objects and their parts inherently possess both coarse-grained differences and intricate fine-grained structural relationships. These characteristics can be formalized as knowledge, leveraged for fine-grained part comprehension. However, existing part segmentation models consistently fail to capture these complex inter-part relationships, treating parts as independent entities and disregarding object-level distinctions. To address these limitations, we propose a novel Knowledge-Guided Part Segmentation (KPS) framework. Our approach automatically extracts structural relationships between parts using a large language model (LLM) and integrates them into a knowledge graph. Subsequently, a structural knowledge guidance module employs a graph convolutional network (GCN) to model these relationships. Furthermore, a coarse-grained object guidance module captures object-specific distinctions and integrates them as visual guidance. The integrated insights from the part structure and object differentiation guide the fine-grained part segmentation. Our KPS achieves notable improvements in segmentation performance, with a 4.96% mIoU gain on PartImageNet and a 3.73% gain on Pascal-Part. Moreover, in the open-vocabulary setting on Pascal-Part-116, it improves hIoU by 3.25%, highlighting the effectiveness of knowledge guidance in enhancing fine-grained part segmentation.

1. Introduction

Fine-grained part semantic segmentation [20, 45, 52, 60] aims to achieve pixel-level segmentation of fine-grained object parts within a complete image by capturing distinguishing details of individual parts, such as the head of a bird, the wheels of a car, or the ears of a dog. In comparison, traditional semantic segmentation approaches

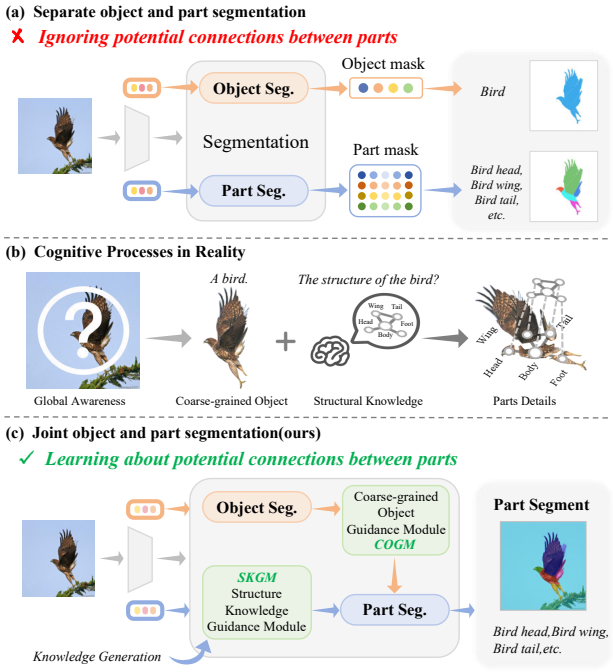

Figure 1. **Comparison of existing methods and our KPS framework inspired by human cognition.** **(a)** Existing methods treat parts as independent, ignoring relationships. **(b)** Human cognitive process: global recognition followed by part details. **(c)** Our KPS framework combines structured knowledge and coarse-grained object guidance inspired by cognitive processes to enhance part segmentation.

[1, 4, 21, 31, 40, 50, 63] primarily focus on coarse-grained information, performing well on large object regions but often neglecting the fine details required for part-level segmentation [8, 9, 58, 61]. To address these limitations, fine-grained part segmentation has become essential for capturing intricate details, especially in complex or detail-rich scenes where precise recognition is critical. By focusing on fine-grained features, these methods enhance feature representations and sensitivity to subtle variations, thereby im-

*Corresponding author

proving performance in applications such as image editing and robotics.

To tackle the need for fine-grained segmentation, existing methods have adopted various strategies. For example, He et al. [20] employ hierarchical feature representations to iteratively cluster pixels into parts, achieving refined part segmentation at the image level. In contrast, Pan et al. [45] adopt a class-agnostic strategy, using post-processing to remove independently predicted part masks that do not connect with other parts. In summary, most of these methods rely on object-level segmentation strategies (as shown in Figure 1(a)), treating each part as an independent category. Despite their progress in fine-grained part segmentation, they often lack mechanisms to effectively capture and utilize relationships between parts, resulting in the loss of critical information and limiting segmentation accuracy.

In contrast to these methods, human perception in real-world scenarios is characterized by a progression from coarse to fine granularity [25], as shown in Figure 1(b). Initially, humans perceive the overall category at the object level, gradually analyzing the structural relationships among parts, leveraging knowledge of object structures and part distributions to achieve fine-grained perception. For instance, when viewing an image of a bird, humans first perceive it at a coarse level and then, leveraging prior knowledge of bird anatomy, examine the spatial arrangement of its parts to identify specific components, such as the wings and tail. This aligns with cognitive psychologists' argument that human perception follows a layered and progressively detailed process [22, 37], transitioning from the whole to the details and from coarse-grained to fine-grained understanding, thereby enabling clear and meaningful insights into the structures of object parts, their roles within the whole, and their interrelationships.

Based on this cognitive process analysis, we identify that structural relationships among fine-grained parts and distinctions between coarse-grained objects can serve as guiding knowledge to enhance fine-grained part segmentation. To operationalize this insight, we present the Knowledge-Guided Part Segmentation (KPS) framework (Figure 1(c)), which encodes part relationships into a knowledge graph via a structured knowledge acquisition process while incorporating a Coarse-grained Perception Module (CPM) to capture object-level distinctions. This dual approach collaboratively addresses structural relationships among fine-grained parts and distinctions among coarse-grained objects. To effectively leverage this embedded knowledge, a Structural Knowledge Guidance Module (SKGM) applies graph convolution to refine part relationships, generating text-based guidance that captures structural connections. In parallel, a Coarse-Grained Object Guidance Module (COGM) captures distinctive object-level features, providing complementary visual guidance. Together with these text-based and visual modalities, a Fine-grained Perception Module (FPM) enables precise segmentation of fine-grained parts.

In summary, our KPS framework integrates fine-grained part relationships and coarse-grained object distinctions as guiding knowledge to enhance textual and visual embeddings for part segmentation. KPS achieves mIoU scores of 72.69% on PartImageNet [19] and 62.42% on Pascal-Part [10]. In the open-vocabulary setting on Pascal-Part-116 [56], it attains an hIoU of 33.92%, demonstrating notable improvements in segmentation accuracy and interpretability. The key contributions are summarized as follows:

- We propose a framework that encapsulates object-level distinctions and part-level structural relationships, embedding these elements as guiding knowledge to improve fine-grained segmentation.
- The Structural Knowledge Guidance Module embeds the knowledge graph, capturing structural relationships, into text features using graph convolution to provide guidance.
- The Coarse-Grained Object Guidance Module embeds object-level distinctions into visual features, offering coarse-grained guidance to support part segmentation.
- Comprehensive closed-set and open-vocabulary experiments on PartImageNet, Pascal-Part, and Pascal-Part-116 validate the effectiveness of our knowledge-guided KPS framework in part segmentation tasks.

2. Related Works

2.1. Part Segmentation

Fine-grained part segmentation, which decomposes images into detailed components for improved interpretability and analysis, has gained significant attention in recent years [20, 27, 44, 45, 55, 57, 60]. Advances in deep learning and large-scale annotated datasets have shifted research from coarse object-level tasks to more nuanced part-level segmentation [3].Early approaches were constrained by limited part-level annotations, relying on handcrafted features and traditional image processing, which hindered performance and scalability. The introduction of large-scale datasets like PartImageNet [19] and Pascal-Part [10] has revitalized research, driving significant progress. However, existing methods [20, 45, 60] often treat parts as independent categories, following a coarse-grained paradigm that overlooks structural and contextual relationships within objects, limiting their effectiveness in fine-grained part understanding.

2.2. Knowledge Guidance

In machine learning [26], traditional data-driven models often struggle to incorporate human knowledge, limiting their performance. Integrating structured knowledge, such as category relationships, has proven effective in enhancing deep neural networks for tasks like image classification [23, 36], improving accuracy [14] and robustness

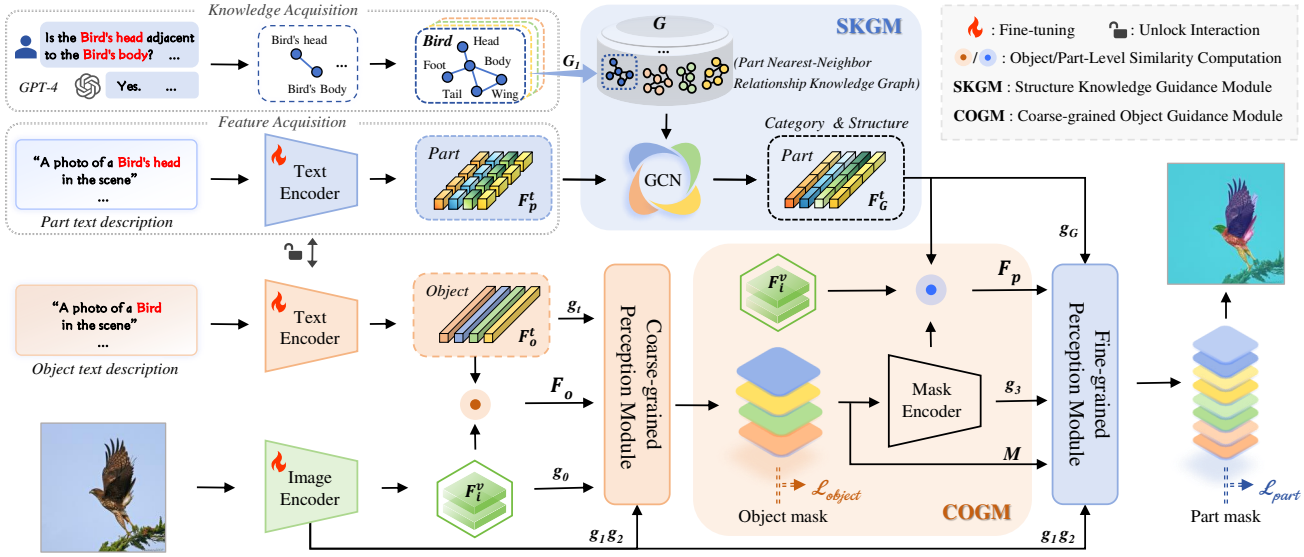

Figure 2. **Overview.** The proposed KPS framework enables fine-grained part segmentation through knowledge-guided modules. It comprises a Structural Knowledge Guidance Module for encoding part relationships, a Coarse-grained Object Guidance Module for capturing object-level distinctions, and two Perception Modules that respectively provide coarse-grained and fine-grained segmentation.

against adversarial attacks [18]. Knowledge graphs (KGs) provide a powerful framework for embedding structured knowledge, benefiting both visual and textual tasks [42]. They have been shown to enhance reasoning in language models [2, 5, 34] and improve computer vision models [24, 29, 30, 32, 33, 35, 46, 54]. For instance, [34] leverages inter-class relationships within KGs to enhance vision-language models, leading to more effective classification in downstream tasks. By integrating visual and textual information, knowledge-guided approaches offer substantial benefits, improving model accuracy, adaptability, and robustness in complex real-world scenarios.Overall, combining visual and textual information, knowledge-guided approaches offer substantial benefits, particularly in managing complex tasks and diverse scenarios. This guidance improves model accuracy and adaptability, enabling models to handle the nuanced demands of real-world applications better.

3. Methodology

This section introduces our approach, starting with an architecture overview in Section 3.1, followed by module details in Sections 3.2– 3.5 and learning objectives in Section 3.6.

3.1. Overall architecture

The proposed Knowledge-Guided Part Segmentation (KPS) model enhances fine-grained segmentation by embedding structural knowledge and object-level distinctions as auxiliary guidance throughout the segmentation pipeline. To achieve this, it employs a hybrid data-driven and knowledge-guided approach, as illustrated in Figure 2, building on a cost aggregation architecture [12] and incorporating innovative knowledge guidance modules that enhance segmentation accuracy and robustness. Specifically, the KPS architecture integrates part-level structural knowledge and coarse-grained object-level distinctions via two complementary modules, providing essential structural and contextual guidance. The Structural Knowledge Guidance Module (SKGM) embeds spatial relationships among object parts into text features, serving as core structural guidance. At the same time, the Coarse-grained Perception Module (CPM) captures object-level perceptual cues, supplying essential context for object differentiation. The Coarse-grained Object Guidance Module (COGM) then incorporates these distinctions into the visual modality, enhancing robustness and object-level differentiation. Finally, the Fine-grained Perception Module (FPM) combines structural and contextual cues from text and visual modalities to guide precise part segmentation. This hierarchical approach progressively enhances segmentation accuracy by embedding knowledge at different levels, producing precise and interpretable results.

3.2. Structure Knowledge Guidance Module

Knowledge Acquisition. Structural relationships among object parts are foundational to human cognition. Large language models like GPT-4, through extensive training, have similarly captured these spatial relationships. Building on this, our approach uses a knowledge-based Q&A process with GPT-4 to extract these relationships, embedding dataset categories into question templates to capture

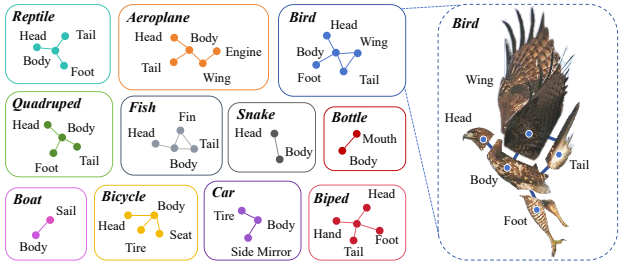

Figure 3. Knowledge graph based on PartImageNet dataset categories, constructed by aggregating adjacency relationships between parts extracted from a large language model through a knowledge acquisition process.

latent spatial connections. For example, querying "Is the bird's head adjacent to the bird's body?" yields a "Yes" response, which we binarize to convert unstructured insights into structured adjacency relationships, forming part-to-part connections within a nearest-neighbor knowledge graph. This Q&A process systematically translates responses into adjacency relationships among object parts, which are then aggregated to construct a knowledge graph $G = \{\text{GPT-4}(q_j) \mid j = 1, 2, \ldots, N_q\}$, where N_q is the total number of questions. Figure 3 illustrates a sample knowledge graph based on the PartImageNet dataset, with nodes as object parts and edges as adjacency connections structured by graph theory. This stable adjacency structure is auxiliary guidance, embedding structural context into part-level text features.

Feature Acquisition. Given a set of candidate part-level categories $D_p = \{d_p[n]\}_{n=1}^{N_p}$, where $d_p[n]$ denotes the textual description of part n and N_p represents the total number of parts, we use the Contrastive Language-Image Pre-training(CLIP[47]) text encoder, denoted as $\Phi^T(\cdot)$, to obtain the corresponding part-level text features $F_p^t = \Phi^T(D_p)$. The resulting features are represented as $F_p^t[n] \in \mathbb{R}^{N_p \times d}$, where n indexes each part and d denotes the dimensionality of the text feature vector.

Structural Knowledge Embedding. To effectively embed structural knowledge within the text semantic space, we design a graph convolutional network (GCN) [7] that utilizes part-level text features $F_p^t[n]$ and the knowledge graph G as inputs. In this GCN, text-based semantic features represent nodes, while adjacency relationships from the knowledge graph form the edges. By applying N_g layers of TAGConv [15] graph convolution, the network captures the structural relationships among nodes, embedding them into the text features to enhance part-level segmentation. Formally, we represent the output as:

$$F_G^t[n] = \text{GCN}(F_p^t[n], G), \tag{1}$$

where the enriched text features now contain embedded structural context for segmentation.

3.3. Coarse-grained Perception Module

Building upon established cost aggregation methods for object-level perception [12], this module is designed to capture object-level contextual cues, thereby enhancing the model's ability to differentiate between distinct coarse-grained objects within an image. Specifically, given an image I and a set of candidate object categories $D_o = \{d_o[m]\}_{m=1}^{N_o}$, where $d_o[m]$ denotes the text description of object m, and N_o represents the total number of objects, we employ CLIP's image and text encoders to extract image features $F_i^v = \Phi^V(I) \in \mathbb{R}^{(H \times W) \times d}$ and object-level text features $F_o^t = \Phi^T(D_o) \in \mathbb{R}^{N_o \times d}$, where $\Phi^V(\cdot)$ denotes the CLIP image encoder. Here, each text feature $F_o^t[m]$ is indexed by m, and d represents the dimensionality of the text feature vectors. Next, we compute the cosine similarity [49] between the image and text features to capture spatial associations between image content and object-level categories:

$$C_o(i, m) = \frac{F_i^v(i) \cdot F_o^t(m)}{\|F_i^v(i)\| \|F_o^t(m)\|}, \tag{2}$$

where i represents the 2D position in the image feature map, and m indexes an object-level category in the text features.

To manage the high-dimensional complexity of similarity maps, we apply a convolutional layer, processing each similarity slice $C_o(:, m) \in \mathbb{R}^{(H \times W) \times 1}$ independently. This operation produces an initial object-level similarity feature representation $F_o \in \mathbb{R}^{(H \times W \times N_o \times d_F)}$, where d_F denotes the dimensionality of each similarity feature.

Building on these similarity features, we employ three key components: a Swin Transformer module [39] for visual perception, a text transformer module [53] for text-based perception, and an upsampling decoder [12]. Leveraging the global comprehension abilities of the fine-tuned CLIP model, CPM iteratively alternates between the visual and text perception modules N_t times, followed by a final decoder for coarse-grained perception and segmentation. Specifically, spatial aggregation is performed using the Swin Transformer's visual perception module, operating on both the input visual guidance features g_0 and the object-level similarity features $F_o(:, n)$, as follows:

$$F_o'(:, n) = \Phi_{\mathcal{CA}}^{\mathrm{V}}([F_o(:, n); \mathrm{g}_0]), \tag{3}$$

where $F_o(:, n) \in \mathbb{R}^{(H \times W) \times d_F}$ and d_F represents the channel dimension for each token. Here, $\Phi_{\mathcal{CA}}^{\mathrm{V}}(\cdot)$ denotes a pair of two consecutive Swin Transformer blocks performing spatial aggregation. The first block applies self-attention within a local window, while the second operates with a shifted window to enhance context capture across the spatial domain. Notably, visual features $\mathrm{g}_0 = P^v(F_i^v)$, derived through a linear projection layer P^v, serve as guidance for

mapping, with $F_i^v \in \mathbb{R}^{(H\times W)\times d}$ and d representing the feature dimension. Then, the aggregated spatial features, with object-level text features as auxiliary information, are input into a Transformer layer to capture relationships among different object categories. This process can be expressed as:

$$F_o''(i,:) = \Phi_{\mathcal{CA}}^{\mathrm{T}}([F_o'(i,:); \mathrm{g}_t]), \tag{4}$$

where $F_o'(i,:) \in \mathbb{R}^{N_o\times d_F}$ and $\Phi_{\mathcal{CA}}^{\mathrm{T}}(\cdot)$ represents the Transformer block used for textual perception. Unlike the visual perception pathway, this layer employs a linear Transformer, followed by a linear projection layer after the multi-head attention mechanism. Such a architecture facilitates flexible adjustments within the feature space, thereby supporting the integration of auxiliary information. Additionally, the text features $\mathrm{g}_t = P^t(F_o^t)$, obtained through the linear projection layer P^t, guide the mapping of query and key features, where $F_o^t \in \mathbb{R}^{N_o\times d}$ and d denotes the dimensionality of the feature vector.

Finally, the text perception block output undergoes bilinear upsampling and combines corresponding feature maps extracted from CLIP. This merged feature set is then processed through a 3×3 convolutional layer. The sequence is repeated twice to generate a coarse-grained, object-level segmentation output, which is then passed to the prediction head for final inference:

$$\mathcal{M}(i) = \mathcal{D}_o(\mathrm{F}_o''(i,:)), \tag{5}$$

where $\mathcal{D}_o$ denotes the decoder and the segmentation head, and $\mathcal{M}(i)$ denotes the class probability maps obtained from coarse-grained perception, which corresponds to coarse-grained information. The detailed structure of this module is presented in Appendix A.1.

3.4. Coarse-grained Object Guidance Module

We use the differences between different coarse-grained objects to guide the visual semantic space. Specifically, we compute the similarity between visual features and text features enriched with structural knowledge as follows:

$$C_p^V(i,n) = \frac{F_i^v(i)\cdot F_G^t(n)}{\|F_i^v(i)\|\|F_G^t(n)\|}, \tag{6}$$

where n represents the index of an object-level class.

To effectively leverage coarse-grained object information, we design a mask encoder that extracts multi-scale features through convolutional layers. This architecture progressively captures hierarchical representations, enhancing model expressiveness through nonlinear activations and downsampling operations. The resulting features are then refined to generate the final coarse-grained representation:

$$\mathrm{g}_3(i) = \Phi_{\mathcal{E}}(\mathcal{M}(i)), \tag{7}$$

where $\Phi_{\mathcal{E}}(\cdot)$ denotes the mask encoder. The detailed structure of this module is presented in Appendix A.2.

Next, we compute the similarity between the mask features and the structural knowledge-embedded text features:

$$C_p^M(i,n) = \frac{\mathrm{g}_3(i)\cdot F_G^t(n)}{\|\mathrm{g}_3(i)\|\|F_G^t(n)\|}, \tag{8}$$

and integrate this into the original similarity information:

$$C_p(i,n) = C_p^V(i,n) + C_p^M(i,n). \tag{9}$$

This approach enhances coarse-grained guidance by capturing fine-grained local details and minimizing background interference, thus providing a more precise foundation for fine-grained segmentation. To address the complexity inherent in high-dimensional feature spaces, we input the combined similarity map $C_p(i,n)$ into a convolutional layer. This layer outputs the initial part-level similarity feature $F_p \in \mathbb{R}^{(H\times W\times N_p\times d_F)}$, where H and W represent the spatial dimensions of the feature map, N_p denotes the number of part categories, and d_F indicates the dimensionality of the similarity feature vector.

3.5. Fine-grained Perception Module

The fine-grained perception module extends the coarse-grained perception by embedding structural relationships among parts and integrating coarse-grained object distinctions as guiding knowledge. The detailed structure of this module is described in Appendix A.3. We enhance the visual perception path by stacking three consecutive Swin Transformer blocks to achieve fine-grained feature capture. Specifically, the first block applies self-attention within a local window, guided by coarse-grained information; the second introduces shifted window self-attention; and the third returns to local window self-attention with additional coarse-grained guidance. This architecture focuses on foreground elements, minimizes background noise, and refines understanding of parts relationships. In this enhanced module, we substitute the original CLIP guidance feature g_0 with coarse-grained information g_3 to focus more precisely on object-specific details:

$$F_p'(:,n) = \Phi_{\mathcal{FA}}^{\mathrm{V}}([F_p(:,n); \mathrm{g}_3; \mathcal{M}]), \tag{10}$$

where $F_p'(:,n) \in \mathbb{R}^{(H\times W)\times d_F}$ represents the refined part-level features in the visual perception pathway. Additionally, to replace the original textual guidance, we incorporate part structure knowledge graph information within the text features, g_G, in place of g_t:

$$F_p''(i,:) = \Phi_{\mathcal{FA}}^{\mathrm{T}}([F_p'(i,:); \mathrm{g}_G]). \tag{11}$$

3.6. Objective Function

We use cross-entropy loss functions to optimize segmentation at both the object and part levels. The final loss function combines the object-level and part-level losses with weighted terms to balance the optimization process:

$$\mathcal{L} = \lambda_o \mathcal{L}_{\text{object}} + \lambda_p \mathcal{L}_{\text{part}}, \tag{12}$$

where λ_o and λ_p are coefficients that control the relative contributions of the object and part losses, ensuring balanced optimization across segmentation levels.

4. Experiments

4.1. Datasets and Evaluation Metrics

To comprehensively evaluate the effectiveness of hierarchical perception and knowledge guidance in our method, we conduct experiments in both closed-set and open-vocabulary settings, utilizing three widely adopted datasets for a thorough assessment in each setting.

PartImageNet [19] is a large-scale dataset for part-level segmentation with COCO-comparable annotations. It covers 158 ImageNet categories (24,080 images), organized into 11 superclasses and 40 part categories, supporting fine-grained part segmentation.

Pascal-Part [10] augments Pascal VOC with part-level annotations for 10,103 images over 20 object classes. We use 16 classes with part annotations, merging similar labels (e.g., left-wing/right-wing $\rightarrow$ wing).

Pascal-Part-116 [56] is a benchmark for open-vocabulary part segmentation, with 8,431 training and 850 test images. It refines Pascal-Part by merging classes and removing directional indicators, resulting in 116 part categories across 20 object classes.

Evaluation Metrics. For closed-set evaluation, we use mean Intersection over Union (mIoU) and mean Pixel Accuracy (mACC). For open-vocabulary evaluation, we report mIoU on seen and unseen classes, and use harmonic IoU (hIoU) for balanced performance across both.

4.2. Implementation Details

For all experiments, we set the feature dimensionality d_F= 128. In the model architecture, N_m= 4 defines the number of convolutional layers in the mask encoder, N_t= 2 specifies the number of iterations for alternating visual and text perception modules within the perception module, and N_g= 2 denotes the number of graph convolutional layers in the GCN for structural knowledge embedding. Our framework is implemented in PyTorch and trained on a setup with 4 NVIDIA RTX 3090 GPUs. The CLIP ViT-B/16 text and vision encoders are fine-tuned and initialized with weights from the official pre-trained CLIP model, while the remaining model weights are initialized randomly. Input images are resized to 512×512 without any data augmentation applied. We use the AdamW optimizer with a cosine learning rate schedule. The learning rates are set to 0.003 for PartImageNet, 0.002 for Pascal-Part, and 0.001 for the Pascal-Part-116 dataset, with a scaling factor of 0.01 applied to the CLIP encoders to ensure stable fine-tuning. The batch size for training is set to 8, with a weight decay of 0.0001. Models are trained for a total of 50,000 iterations.

Method	Venue	Backbone	mIOU(%)	mACC(%)
SemanticFPN [28]	CVPR'19	ResNeXt	56.76	-
Deeplab v3+ [9]	ECCV'18	ResNet-50	60.57	71.07
MaskFormer [11]	NeurIPS'21	ResNet-50	60.34	72.75
MaskFormer-Dual	NeurIPS'21	ResNet-50	58.02	70.42
Compositor [20]	CVPR'23	ResNet-50	61.44	73.41
SegFormer [61]	NeurIPS'21	MiT-B2	61.97	73.77
MaskFormer [11]	NeurIPS'21	Swin-T	63.96	77.37
MaskFormer-Dual	NeurIPS'21	Swin-T	61.69	75.64
Compositor [20]	CVPR'23	Swin-T	64.64	78.31
MaskFormer [11]	NeurIPS'21	Swin-B	65.24	78.84
UniLSeg-20 [38]	CVPR'24	Swin-B	62.46	-
UniLSeg-100 [38]	CVPR'24	Swin-B	63.87	-
CAT-Seg [12]	CVPR'24	ViT-B/16	67.73	79.52
KPS (ours)	This Work	ViT-B/16	**72.69** (+4.96)	**82.46** (+2.94)

Table 1. Comparison of KPS and classical segmentation models on PartImageNet and existing methods for part segmentation.

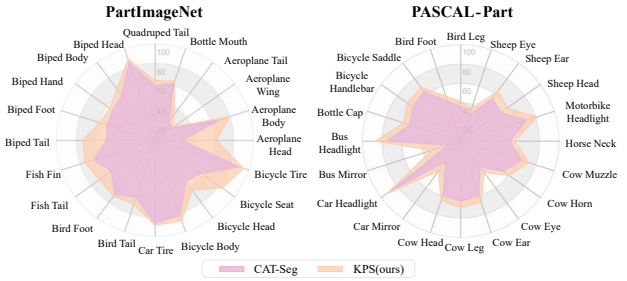

Figure 4. A radar chart comparing the performance of CAT-Seg [12] and KPS (ours) across various part categories, illustrating the effectiveness of KPS in fine-grained segmentation tasks.

4.3. Main Results

In a closed-set setting, we compare KPS with classical segmentation frameworks [9, 11, 61], the current state-of-the-art part segmentation methods [20, 38], and the baseline CAT-Seg [12]. As shown in Tables 1 and 2, KPS outperforms the baseline on multiple metrics.

Part Segmentation on PartImageNet and Pascal-Part. As shown in Table 1 and Table 2, our model outperforms all existing methods on both datasets. Compared to classical segmentation methods like MaskFormer[11] and part segmentation approaches such as Compositor[20], KPS consistently achieves superior performance. Furthermore, to account for differences in model backbone, we compare with the recent CAT-Seg [12] model, which serves as our base-

Method	Venue	Backbone	mIOU(%)	mACC(%)
MaskFormer [11]	NeurIPS'21	ResNet-50	47.61	58.59
MaskFormer-Dual	NeurIPS'21	ResNet-50	46.60	57.96
Compositor [20]	CVPR'23	ResNet-50	48.01	58.83
MaskFormer [11]	NeurIPS'21	Swin-T	55.42	67.21
MaskFormer-Dual	NeurIPS'21	Swin-T	54.21	66.42
Compositor [20]	CVPR'23	Swin-T	55.92	67.63
MaskFormer [11]	NeurIPS'21	Swin-B	56.83	68.46
CAT-Seg [12]	CVPR'24	ViT-B/16	58.69	69.85
KPS (ours)	This Work	ViT-B/16	**62.42** (+3.73)	**72.13** (+2.28)

Table 2. Comparison of KPS and classical segmentation models on Pascal-Part and existing state-of-the-art methods.

line and also incorporates CLIP, but lacks knowledge guidance. The results show that KPS achieves 72.69% mIoU and 82.46% mACC on PartImageNet, outperforming CAT-Seg by 4.96% and 2.94%, respectively. On the Pascal-Part dataset, KPS achieves 62.42% mIoU and 72.13% mACC, outperforming CAT-Seg [12] by 3.73% and 2.28%, respectively. These improvements highlight that the performance gain is not due to a stronger backbone but rather to the knowledge guidance, which leverages part-level structure and coarse-grained knowledge to guide fine-grained segmentation. This allows KPS to better capture the complex structural relationships between parts, thereby enhancing part segmentation performance. These comparisons ultimately validate the effectiveness of knowledge guidance in part segmentation tasks.

To thoroughly validate the effectiveness of knowledge-guided and hierarchical perception, we evaluate KPS in an open-vocabulary setting, comparing it with open-vocabulary semantic segmentation [12, 41, 62] and open-vocabulary part segmentation methods [13, 52, 56].

Part Segmentation on Pascal-Part-116. As shown in Table 3, KPS outperforms existing open-vocabulary segmentation methods, achieving a 3.25% improvement in hIoU. Compared to the best-performing approach [13], KPS improves hIoU by 3.25%, with mIoU gains of 8.32% and 1.55% on seen and unseen categories, respectively, validating the effectiveness of knowledge guidance and hierarchical perception. It also surpasses the baseline CAT-Seg [12] by 7.2% in hIoU. Notably, KPS achieves substantial gains on seen categories, highlighting its ability to leverage part-level structural knowledge while maintaining competitive performance on unseen ones. These results confirm that knowledge guidance enhances both segmentation accuracy and generalization in open-vocabulary part segmentation.

4.4. Visual Analysis and Ablation Study

We conduct visualization analysis on PartImageNet and perform ablation experiments on Pascal-Part to verify the effectiveness of each module.

Visual Analysis. Conventional part segmentation methods

Method	Venue	Backbone	Seen	Unseen	Harmonic
ZSSeg+ [62]	ECCV'22	ResNet-50	38.05	3.38	6.20
VLPart [52]	ICCV'23	ResNet-50	35.21	9.04	14.39
CLIPSeg [41, 56]	CVPR'22 NeurIPS'23	ViT-B/16	27.79	13.27	17.96
CAT-Seg [12, 56]	CVPR'24 NeurIPS'23	ViT-B/16	28.17	25.42	26.72
PartCLIPSeg [13]	NeurIPS'24	ViT-B/16	43.91	23.56	30.67
KPS (ours)	This Work	ViT-B/16	**52.23** (+8.32)	**25.11** (+1.55)	**33.92** (+3.25)

Table 3. Comparison of KPS with classical open-vocabulary and state-of-the-art part segmentation models on Pascal-Part-116.

EXP	Baseline	FPM	COGM	SKGM	mIOU(%)	mACC(%)
(I)	✓				58.69	69.85
(II)	✓	✓			59.10	69.96
(III)	✓	✓	✓		60.24	70.11
(IV)	✓	✓	✓	✓	**62.42**	**72.13**

Table 4. Impact of Different Modules in Our KPS on Pascal-Part, showing the effect of integrating various modules into the baseline.

treat each category as an independent entity, typically labeling parts in an "object-part" format (e.g., "Bird head"). This approach results in clustering identical parts across different objects while dispersing parts within the same object, as visualized on the PartImageNet dataset in Figure 6(a), thereby overlooking stable structural relationships. We introduce the SKGM to address this issue, which captures part connections using a graph convolutional structure. By embedding these complex structural relationships directly into the text features F_G^t, as shown in Figure 6(b), SKGM enhances feature representation and strengthens the model's capability for precise, fine-grained part segmentation.

Effectiveness of our modules. Table 4 presents the ablation study results on the Pascal-Part dataset, selected for its diverse part categories and fine-grained annotations, effectively highlighting the contribution of each module within the KPS framework. The baseline model (I) is built upon the cost aggregation model [12], performing part segmentation independently through the perceptual module without knowledge guidance. In setting (II), we incorporate the FPM to refine structural details within the fine-grained perceptual module. Setting (III) further introduces the COGM, providing object-level visual guidance as coarse-grained knowledge to enhance object distinction. Setting (IV) integrates the SKGM, embedding part-level structural relationships into textual features to improve the model's perception of fine-grained part structures. Together, these modules constitute the KPS framework, designed for accurate fine-grained part segmentation.

Effectiveness of Knowledge Graph Design. As shown in Table 5, results on the Pascal-Part dataset demonstrate that the LLM-based knowledge graph, which captures authentic structural relationships, consistently achieves the best performance across all configurations. Specifically, fully connecting all parts prevents the model from distinguishing hierarchical relationships, making it difficult to differentiate

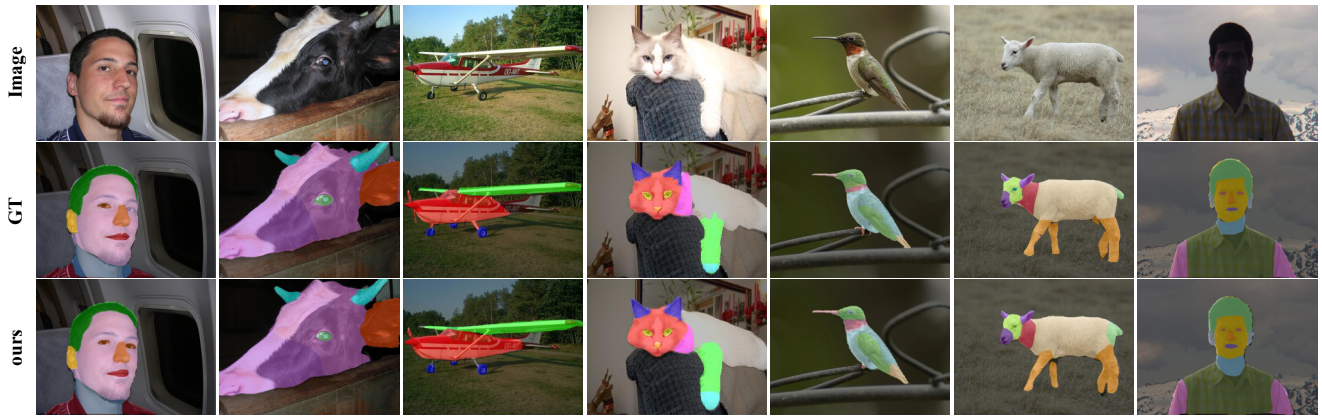

Figure 5. Visualizations of part segmentation on Pascal-Part. The first row shows the original images, the second row shows the ground truth, and the third row presents the predictions from our proposed KPS model. These high-quality part predictions demonstrate the effectiveness of our proposed approach.

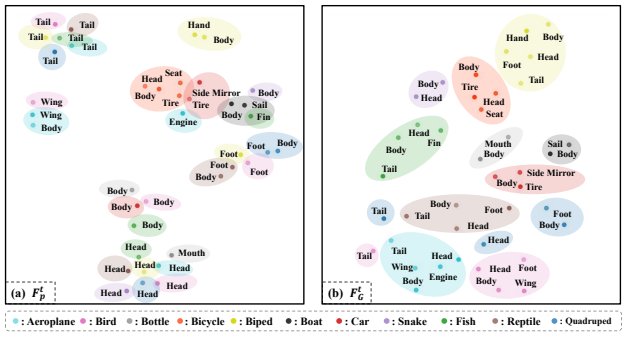

Figure 6. t-SNE visualization of text features on the PartImageNet dataset: **(a)** Original text features; **(b)** Text features embedded with part structure relationships from the knowledge graph. Similar colors represent different parts of the same object.

Connection Types	LLM	mIOU(%)	mACC(%)
Fully connected across all parts	✗	60.47	70.28
Fully connected within parts of individual objects	✗	61.21	70.43
True structural relationships between parts	GPT-4	**62.42**	**72.13**

Table 5. Ablation Study on Knowledge Graph Connection.

between semantically similar or distinct parts. Additionally, restricting connections to parts within the same object preserves local structure but overlooks potential commonalities across different objects, limiting the model's understanding of intra-object part relationships. In contrast, the knowledge graph based on true structural relationships effectively models both hierarchical and cross-object connections, enabling the model to maintain local structural integrity while accurately capturing relationships among different parts.

Effectiveness of the N_g in the GCN. As shown in Table 6, we analyze the impact of the number of convolutional layers (N_g) of the graph convolutional network (GCN) within the SKGM module, which embeds part structural knowledge into textual features. We evaluate different settings with $N_g \in \{1, 2, 3, 4\}$. The results show that increasing N_g enhances the GCN's ability to capture complex structural relationships, improving segmentation performance. However, when N_g= 3 or N_g= 4, the performance gain saturates, and slight degradation occurs due to overfitting in deeper GCNs, where feature representations become overly similar, weakening the model's ability to distinguish part boundaries.

	$N_g = 1$	$N_g = 2$	$N_g = 3$	$N_g = 4$
mIOU (%)	60.75	**62.42**	61.82	61.78
mACC (%)	70.69	**72.13**	71.41	71.04

Table 6. Ablation Study on the Number of Convolutional Layers N_g in the GCN using TAGConv.

Additional ablations are shown in Appendix.

5. Conclusion

In this work, we propose a knowledge-guided part segmentation framework that simulates real-world cognitive processes by capturing the differences between coarse-grained objects and the structural relationships among fine-grained parts. It transforms this information into structured knowledge to guide segmentation, enabling a deeper understanding of object composition and providing richer contextual awareness. Specifically, we employ the Knowledge Acquisition process to structure relationships among parts into a nearest-neighbor knowledge graph, embedding this structural information into text features through the Structural Knowledge Guidance Module. This module enhances the model's understanding of fine-grained structures by incorporating part-level relational knowledge. Meanwhile, we introduce object-level distinctions as auxiliary guidance through the Coarse-grained Perception and Object Guidance Module, which integrates coarse-grained knowledge to enhance fine-grained differentiation, facilitating hierarchical perception. We conduct experiments on PartImageNet, Pascal-Part, and Pascal-Part-116 datasets, and the results strongly validate the effectiveness of knowledge guidance.

Acknowledgement

This work was supported in part by the Key Project of National Natural Science Foundation of China (62431020, 62231027), the Joint Fund Project of National Natural Science Foundation of China (No.U22B2054), the Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) (No.B07048), the Postdoctoral Fellowship Program of China Postdoctoral Science Foundation (CPSF) (No.GZC20232033), the Program for Cheung Kong Scholars and Innovative Research Team in University (No.IRT_15R53), the Key Scientific Technological Innovation Research Project by Ministry of Education and the National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, Xi'an Jiaotong University (No. HMHAI-202404, No. HMHAI-202405).

References

[1] Researchers' work from xidian university focuses on networks (mfnet: a novel gnn-based multi-level feature network with superpixel priors). *Network Daily News*, (7):82–83, 2023. 1

[2] KM Annervaz, Somnath Basu Roy Chowdhury, and Ambedkar Dukkipati. Learning beyond datasets: Knowledge graph augmented neural networks for natural language processing. *arXiv preprint arXiv:1802.05930*, 2018. 3

[3] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4253–4262, 2020. 2

[4] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 1

[5] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. Comet: Commonsense transformers for automatic knowledge graph construction. *arXiv preprint arXiv:1906.05317*, 2019. 3

[6] Xavier Bresson and Thomas Laurent. Residual gated graph convnets. *arXiv preprint arXiv:1711.07553*, 2017. 2

[7] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013. 4

[8] Niccolò Cavagnero, Gabriele Rosi, Claudia Cuttano, Francesca Pistilli, Marco Ciccone, Giuseppe Averta, and Fabio Cermelli. Pem: Prototype-based efficient maskformer for image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15804–15813, 2024. 1

[9] Liang-Chieh Chen. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 1, 6

[10] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1971–1978, 2014. 2, 6

[11] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in neural information processing systems*, 34:17864–17875, 2021. 6, 7

[12] Seokju Cho, Heeseong Shin, Sunghwan Hong, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4113–4123, 2024. 3, 4, 6, 7

[13] Jiho Choi, Seonho Lee, Seungho Lee, Minhyun Lee, and Hyunjung Shim. Understanding multi-granularity for open-vocabulary part segmentation. *arXiv preprint arXiv:2406.11384*, 2024. 7

[14] Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 48–64. Springer, 2014. 2

[15] Jian Du, Shanghang Zhang, Guanhang Wu, José MF Moura, and Soummya Kar. Topology adaptive graph convolutional networks. *arXiv preprint arXiv:1710.10370*, 2017. 4

[16] Jian Du, Shanghang Zhang, Guanhang Wu, José MF Moura, and Soummya Kar. Topology adaptive graph convolutional networks. *arXiv preprint arXiv:1710.10370*, 2017. 2

[17] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. *Advances in neural information processing systems*, 28, 2015. 2

[18] Nezihe Merve Gürel, Xiangyu Qi, Luka Rimanic, Ce Zhang, and Bo Li. Knowledge enhanced machine learning pipeline against diverse adversarial attacks. In *International Conference on Machine Learning*, pages 3976–3987. PMLR, 2021. 3

[19] 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 *European Conference on Computer Vision*, pages 128–145. Springer, 2022. 2, 6

[20] Ju He, Jieneng Chen, Ming-Xian Lin, Qihang Yu, and Alan L Yuille. Compositor: Bottom-up clustering and compositing for robust part and object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11259–11268, 2023. 1, 2, 6, 7

[21] Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 11–19, 2017. 1

[22] Licheng Jiao, Ronghua Shang, Fang Liu, and Weitong Zhang. *Brain and nature-inspired learning, computation and recognition*. Elsevier, 2020. 2

[23] Licheng Jiao, Jie Gao, Xu Liu, Fang Liu, Shuyuan Yang, and Biao Hou. Multiscale representation learning for image classification: A survey. *IEEE Transactions on Artificial Intelligence*, 4(1):23–43, 2021. 2

[24] Licheng Jiao, Jie Chen, Fang Liu, Shuyuan Yang, Chao You, Xu Liu, Lingling Li, and Biao Hou. Graph representation learning meets computer vision: A survey. *IEEE Transactions on Artificial Intelligence*, 4(1):2–22, 2022. 3

[25] Licheng Jiao, Mengru Ma, Pei He, Xueli Geng, Xu Liu, Fang Liu, Wenping Ma, Shuyuan Yang, Biao Hou, and Xu Tang. Brain-inspired learning, perception, and cognition: A comprehensive review. *IEEE Transactions on Neural Networks and Learning Systems*, 2024. 2

[26] Licheng Jiao, Xue Song, Chao You, Xu Liu, Lingling Li, Puhua Chen, Xu Tang, Zhixi Feng, Fang Liu, Yuwei Guo, et al. Ai meets physics: a comprehensive survey. *Artificial Intelligence Review*, 57(9):256, 2024. 2

[27] Tsung-Wei Ke, Sangwoo Mo, and X Yu Stella. Learning hierarchical image segmentation for recognition and by recognition. In *The Twelfth International Conference on Learning Representations*, 2023. 2

[28] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6399–6408, 2019. 6

[29] Pengfang Li, Fang Liu, Licheng Jiao, Shuo Li, Lingling Li, Xu Liu, and Xinyan Huang. Knowledge transduction for cross-domain few-shot learning. *Pattern Recognition*, 141: 109652, 2023. 3

[30] Shuo Li, Fang Liu, Licheng Jiao, Lingling Li, Puhua Chen, Xu Liu, and Wenping Ma. Prompt-based concept learning for few-shot class-incremental learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 35, . 3

[31] Shuo Li, Fang Liu, Licheng Jiao, Xu Liu, Puhua Chen, and Lingling Li. Mask-guided correlation learning for few-shot segmentation in remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 62, . 1

[32] Shuo Li, Fang Liu, and Licheng Jiao. Self-training multi-sequence learning with transformer for weakly supervised video anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1395–1403, 2022. 3

[33] Shuo Li, Fang Liu, Zehua Hao, Xinyi Wang, Lingling Li, Xu Liu, Puhua Chen, and Wenping Ma. Logits deconfusion with clip for few-shot learning. 2025. 3

[34] Xin Li, Dongze Lian, Zhihe Lu, Jiawang Bai, Zhibo Chen, and Xinchao Wang. Graphadapter: Tuning vision-language models with dual knowledge graph. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[35] Fang Liu, Xiaoxue Qian, Licheng Jiao, Xiangrong Zhang, Lingling Li, and Yuanhao Cui. Contrastive learning-based dual dynamic gcn for sar image scene classification. *IEEE Transactions on Neural Networks and Learning Systems*, 35 (1):390–404, 2022. 3

[36] Yang Liu, Fang Liu, Licheng Jiao, Qianyue Bao, Lingling Li, Yuwei Guo, and Puhua Chen. A knowledge-based hierarchical causal inference network for video action recognition. *IEEE Transactions on Multimedia*, 2024. 2

[37] Yang Liu, Fang Liu, Licheng Jiao, Qianyue Bao, Long Sun, Shuo Li, Lingling Li, and Xu Liu. Multi-grained gradual inference model for multimedia event extraction. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 2

[38] Yong Liu, Cairong Zhang, Yitong Wang, Jiahao Wang, Yujiu Yang, and Yansong Tang. Universal segmentation at arbitrary granularity with language instruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3459–3469, 2024. 6

[39] 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*, pages 10012–10022, 2021. 4

[40] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[41] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7086–7096, 2022. 7

[42] Sebastian Monka, Lavdim Halilaj, and Achim Rettinger. A survey on visual transfer learning using knowledge graphs. *Semantic Web*, 13(3):477–510, 2022. 3

[43] Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, pages 4602–4609, 2019. 2

[44] Shujon Naha, Qingyang Xiao, Prianka Banik, Alimoor Reza, and David J. Crandall. Part segmentation of unseen objects using keypoint guidance. *IEEE*, 2021. 2

[45] Tai-Yu Pan, Qing Liu, Wei-Lun Chao, and Brian Price. Towards open-world segmentation of parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15392–15401, 2023. 1, 2

[46] Xiaoxue Qian, Fang Liu, Licheng Jiao, Xiangrong Zhang, Xinyan Huang, Shuo Li, Puhua Chen, and Xu Liu. Knowledge transfer evolutionary search for lightweight neural architecture with dynamic inference. *Pattern Recognition*, 143: 109790, 2023. 3

[47] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4

[48] Ekagra Ranjan, Soumya Sanyal, and Partha Talukdar. Asap: Adaptive structure aware pooling for learning hierarchical graph representations. In *Proceedings of the AAAI conference on artificial intelligence*, pages 5470–5477, 2020. 2

[49] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. Convolutional neural network architecture for geometric matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6148–6157, 2017. 4

[50] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 1

[51] Yunsheng Shi, Zhengjie Huang, Shikun Feng, Hui Zhong, Wenjin Wang, and Yu Sun. Masked label prediction: Unified message passing model for semi-supervised classification. *arXiv preprint arXiv:2009.03509*, 2020. 2

[52] 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*, pages 15453–15465, 2023. 1, 7

[53] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 4

[54] Hao Wang, Fang Liu, Licheng Jiao, Jiahao Wang, Zehua Hao, Shuo Li, Lingling Li, Puhua Chen, and Xu Liu. Vilt-clip: Video and language tuning clip with multimodal prompt learning and scenario-guided optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5390–5400, 2024. 3

[55] Wenxuan Wang, Tongtian Yue, Yisi Zhang, Longteng Guo, Xingjian He, Xinlong Wang, and Jing Liu. Unveiling parts beyond objects: Towards finer-granularity referring expression segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12998–13008, 2024. 2

[56] Meng Wei, Xiaoyu Yue, Wenwei Zhang, Shu Kong, Xihui Liu, and Jiangmiao Pang. Ov-parts: Towards open-vocabulary part segmentation. *Advances in Neural Information Processing Systems*, 36:70094–70114, 2023. 2, 6, 7

[57] Meng Wei, Xiaoyu Yue, Wenwei Zhang, Shu Kong, Xihui Liu, and Jiangmiao Pang. Ov-parts: Towards open-vocabulary part segmentation. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[58] Zhixiang Wei, Lin Chen, Yi Jin, Xiaoxiao Ma, Tianle Liu, Pengyang Ling, Ben Wang, Huaian Chen, and Jinjin Zheng. Stronger fewer & superior: Harnessing vision foundation models for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28619–28630, 2024. 1

[59] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. PMLR, 2019. 2

[60] Xin-Jian Wu, Ruisong Zhang, Jie Qin, Shijie Ma, and Cheng-Lin Liu. Wps-sam: Towards weakly-supervised part segmentation with foundation models. In *European Conference on Computer Vision*, pages 314–333. Springer, 2025. 1, 2

[61] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090, 2021. 1, 6

[62] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, pages 736–753. Springer, 2022. 7

[63] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1