UNIVERSITY OF TECHNOLOGY SYDNEY
Faculty of Engineering and Information Technology

Towards Structured Visual Percpetion

by

Liulei Li

A Thesis Submitted
in Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Sydney, Australia

2025

Certificate of Original Authorship

I, Liulei Li, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Computer Science, FEIT at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research was supported by an Australian Government Research Training Program (RTP) Scholarship doi.org/10.82133/C42F-K220.

Signature: Production Note:
Signature removed prior to publication.

Date: 4 June 2025

ABSTRACT

Towards Structured Visual Percpetion

by

Liulei Li

Human visual perception, the foundation for our understanding of the world, is characterized by its ability to interpret scenes as structured, coherent wholes rather than mere collections of isolated objects. Despite deep learning has driven significant progress in computer vision, current visual perception models still fall short in achieving this holistic comprehension. This thesis argues that attaining human-like visual intelligence requires a fundamental shift towards structured visual perception, and presents a body of research effort to develop computational methods that can explicitly model, learn, and reason with visual structures.

This dissertation advances this vision through three interconnected and progressively deepening research thrusts. First, I model dynamic visual structures by leveraging temporal correspondences to capture the evolution of scenes and objects over time. Then, the focus is extended to spatial relational structures, developing approaches to uncover the rich connections between objects and their components to build structured representations of scenes. Finally, I investigate general principles for structured perception through the integration of symbolic knowledge, using commonsense or domain-specific constraints to guide both the learning and inference processes of deep models. Collectively, this thesis outlines a comprehensive roadmap towards equipping machines with visual intelligence that more closely emulates the structured, holistic nature of human visual perception.

Dissertation directed by Prof. Yi Yang, Dr. Wenguan Wang, Prof. Xiaojun Chang
Australian Artificial Intelligence Institute, University of Technology Sydney

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Professor Yi Yang and Dr. Wenguan Wang, for their unwavering support, insightful guidance, and enduring belief in me throughout this journey. Their patience and wisdom have profoundly shaped not only this thesis, but also the way I think, question, and pursue research. Thank you for granting me the intellectual freedom to grow, and the encouragement to reach beyond what I once thought possible.

I am also immensely grateful to my friends who journeyed alongside me. The shared coffee breaks and mutual support made this long and difficult journey feel far less lonely. Their presence turned moments of struggle into memories of warmth, and their steady support was a quiet strength I could always count on.

Furthermore, I owe my heartfelt thanks to my parents. Their love has been the quiet, unfailing constant in a life full of challenges. They never imposed, never questioned; they simply offered their steadfast trust and stood firmly behind me. Their quiet belief becomes the foundation upon which I could build with confidence.

Ultimately, this thesis is not a triumph. It is a continuation — a quiet promise kept between who I was and who I am becoming.

Liulei Li

Sydney, Australia

June, 2025

List of Publications

Conference Paper:

C-1 **Liulei Li**, Tianfei Zhou, Wenguan Wang, Jianwu Li, Yi Yang. "Deep Hierarchical Semantic Segmentation". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*.

C-2 **Liulei Li**, Tianfei Zhou, Wenguan Wang, Lu Yang, Jianwu Li, Yi Yang. "Locality-Aware Inter-and Intra-Video Reconstruction for Self-Supervised Correspondence Learning". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*.

C-3 **Liulei Li**, Wenguan Wang, Tianfei Zhou, Jianwu Li, Yi Yang. "Unified Mask Embedding and Correspondence Learning for Self-Supervised Video Segmentation". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023)*.

C-4 **Liulei Li**, Wenguan Wang, Yi Yang. "LOGICSEG: Parsing Visual Semantics with Neural Logic Learning and Reasoning". In *IEEE/CVF International Conference on Computer Vision (ICCV 2023 Oral)*.

C-5 **Liulei Li**, Jianan Wei, Wenguan Wang, Yi Yang. "Neural-Logic Human-Object Interaction Detection". In *International Conference on Neural Information Processing Systems (NeurIPS 2023)*.

C-6 **Liulei Li**, Wenguan Wang, Yi Yang. "Human-Object Interaction Detection Collaborated with Large Relation-driven Diffusion Models". In *International Conference on Neural Information Processing Systems (NeurIPS 2024)*.

C-7 Yurong Zhang*, **Liulei Li***, Wenguan Wang, Rong Xie, Li Song, Wenjun Zhang. "Boosting Video Object Segmentation via Space-time Correspondence

Learning". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023).*

C-8 Lu Yang*, **Liulei Li***, Xueshi Xin, Yifan Sun, Qing Song, Wenguan Wang. "Large-scale person detection and localization using overhead fisheye cameras". In *IEEE/CVF International Conference on Computer Vision (ICCV 2023 Oral).*

C-9 Bo Zhou*, **Liulei Li***, Yujia Wang, Huafeng Liu, Yazhou yao, Wenguan Wang. "UniAlign: Scaling Multimodal Alignment within One Unified Model". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2025).*

C-10 Yuhang Ding, **Liulei Li**, Wenguan Wang, Yi Yang. "Clustering Propagation for Universal Medical Image Segmentation". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2024).*

C-11 Mu Chen, **Liulei Li**, Ruijie Quan, Wenguan Wang, Yi Yang. "General and Task-Oriented Video Segmentation". In *European Conference on Computer Vision (ECCV 2024).*

C-12 Mu Chen, **Liulei Li**, Wenguan Wang, Yi Yang. "DiffVsgg: Diffusion-Driven Online Video Scene Graph Generation". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2025).*

C-13 Tianfei Zhou*, **Liulei Li***, Gustav Bredell, Jianwu Li, Ender Konukoglu. "Quality-Aware Memory Network for Interactive Volumetric Image Segmentation". In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2021 Oral).*

Journal Paper:

J-1 **Liulei Li**, Wenguan Wang, Tianfei Zhou, Ruijie Quan, Yi Yang. "Semantic hierarchy-aware segmentation". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, pp. 2123-2138, 2024.

J-2 Tianfei Zhou, **Liulei Li**, Gustav Bredell, Jianwu Li, Jan Unkelbach, Ender Konukoglu. "Volumetric memory network for interactive medical image segmentation". *Medical Image Analysis*, vol. 83, pp. 102599, 2023.

J-3 Tianfei Zhou, **Liulei Li**, Xueyi Li, Jianwu Li. "Group-Wise Learning for Weakly Supervised Semantic Segmentation". *IEEE Transactions on Image Processing*, vol. 31, pp. 799-811, 2021.

J-4 Shan Li, Lu Yang, Pu Cao, **Liulei Li**, Huadong Ma. "Frequency-based Matcher for Long-tailed Semantic Segmentation". *IEEE Transactions on Multimedia*, vol. 26, pp. 10395-10405, 2024.

Contents

List of Figures

Chapter 1

Introduction

Our knowledge of the world is deeply rooted in the visual perception of the real environment [325]. Pioneering research in cognitive and biological psychology [224] has revealed that nearly half of the cerebral cortex is devoted to processing visual information, and approximately 90 percent of the data transmitted to the brain is visual in nature. This profound visual faculty underpins our intuitive understanding of complex scenes, our ability to navigate dynamic environments, and our capacity for advanced interaction and learning. Consequently, endowing artificial intelligence with analogous visual perception capabilities is not merely an ambitious technical challenge but a fundamental pursuit towards creating truly intelligent systems [21].

In recent years, the field of computer vision has witnessed remarkable advancements. Driven by the deep learning revolution [156], models based on convolutional neural networks [149] and, more recently, vision Transformers [61], have significantly enhanced the ability of machines to recognize visual content. These advances have led to systems that demonstrate impressive performance in tasks such as object detection [89, 355], image segmentation [329, 398], and instance recognition [318], providing machines with a foundational, albeit still limited, sense of sight.

However, the core challenge in equipping machines with human-like visual perception lies not merely in comprehending individual components within a scene, but also their inter-relations, organizations, and dynamic evolutions. This transcends category-level interpretations and strives towards a **structured perception** of the visual environment. Such a holistic perspective is vital for facilitating complex

reasoning, fostering robust generalization to novel configurations, and ultimately emulating the visual intelligence [14] embedded in human cognition.

This thesis presents our research on developing methods and mechanisms for structured visual perception, approached through three closely related and progressively advancing perspectives. First, I explore the modeling of structures inherent in dynamic visual information, in leverage of the visual correspondence across temporal observations to uncover how objects and scenes interact and evolve over time. Subsequently, the focus is extended from the temporal to the spatial dimension, studying the structural organization within visual scenes. This involves delving into relational modeling to capture the rich connections between objects and their components, which often leads to a structured representation (*e.g.*, graph-based or hierarchical) of scenes. Finally, moving beyond the tailored consideration to specific structural types, I investigate a general principle to realize structured visual perception through knowledge integration. This seeks to inform visual perception models with explicit symbolic knowledge — such as commonsense or domain-specific constraints — during both learning and inference processes. The following sections provide a detailed explanation for each perspective.

1.1 Learning Temporal Structures from Visual Correspondence

Visual correspondence matching across observations at different time steps is a fundamental problem in computer vision, empowering critical applications such as scene understanding [169], object dynamics modeling [88], and 3D reconstruction [147]. A primary obstacle in this area is the difficulty of supervising representation learning for visual correspondence, as dense manual annotations are costly and synthetic data often lacks real-world generalizability. Our research [168, 164, 377, 34] investigates how self-supervised learning from natural videos can provide rich super-

visory signals, circumventing the need for explicit labels and enabling models to learn temporal structures directly from real-world data.

Chapter 3 addresses key limitations in existing self-supervised temporal correspondence learning approaches, which often rely on reconstruction or cycle-consistency but overlook essential capabilities for robust learning.We identify three such missing pieces: the capacity for instance discrimination, explicit location awareness in representations, and consideration for spatial compactness in learned affinities. To bridge these, we introduce LIIR, a locality-aware inter- and intra-video reconstruction framework featuring three innovations: i) a joint inter- and intra-video reconstruction objective that enhances instance-discriminative features by leveraging cross-video context; ii) a position-shifting strategy that encodes spatial information into representations while mitigating absolute position bias; iii) a spatial compactness prior in intra-video pixel affinities, encouraging coherent and localized matches to regularize training and reduce outliers.

Building on the correspondence features from LIIR, Chapter 4 explores self-supervised one-shot Video Object Segmentation (VOS), a challenging task in temporal structure learning. Existing self-supervised VOS methods often separate correspondence learning from mask propagation, creating a mismatch between training objectives and inference needs—leading to error accumulation from flow-based warping. To address this, we propose a new framework that integrates mask embedding learning—proven effective in supervised settings—into the self-supervised paradigm. This is achieved via a self-taught cycle alternating between spacetime pixel clustering for pseudo-mask generation and mask-embedded segmentation learning, supported by dense correspondence learning. By aligning training with the VOS task and explicitly modeling the target object, this approach significantly improves robustness and reduces errors, especially under deformation or occlusion. Our method not only outperforms prior self-supervised techniques but also closes the gap with

fully-supervised approaches.

1.2 Relation Modeling for Structured Scene Understanding

A fundamental ambition in creating intelligent visual systems is to move beyond recognizing isolated objects towards a deeper comprehension of how these objects relate to each other and form coherent, structured scenes. This means perceiving not just what objects are present, but how they are structured — from part-whole compositions [259] and spatial-semantic organizations (*e.g.*, a monitor on a desk) to dynamic interactions [169] (*e.g.*, a person riding a bicycle). This thesis details my efforts in developing network architectures and learning paradigms that explicitly model such relational structures. My work [167, 165, 163, 35, 396] focuses on enabling machines with the ability to understand both the content and the organization of visual scenes, which is crucial for complex reasoning and robust generalization.

In Chapter 5, I address the structured organization of semantic categories. Traditional semantic segmentation treats classes as a flat, disconnected set. However, humans perceive the world hierarchically (*e.g.*, a car is a type of vehicle, which has wheels). We propose HSSN, to perform hierarchical semantic segmentation. Instead of complex architectural changes, HSSN reformulates the task as pixel-wise multi-label classification and introduces new learning strategies. The key point here is to enforce that pixel predictions are consistent with a predefined class hierarchy (*e.g.*, a pixel belonging to car must also belong to vehicle) and to reshape the pixel embedding space to reflect these hierarchical relationships. This allows existing segmentation networks to readily incorporate and benefit from explicit structural knowledge of how semantic concepts relate to one another.

Building on the idea of explicit relational understanding, Chapter 6 tackles the complex task of Human-Object Interaction (HOI) detection. Understanding HOIs requires comprehending not just objects and humans, but the specific actions and

relationships between them. This is challenging due to long-tailed distributions and the need for zero-shot generalization. We propose DiffusionHOI, a model that leverages the powerful compositional and semantic understanding capabilities of large-scale text-to-image diffusion models. The core innovation is to steer these generative models, typically focused on instance generation, towards modeling relations. We achieve this through a novel human-object relation inversion strategy, learning specific embeddings for relations. This allows us to then use the diffusion model to generate diverse training data for HOIs and to extract relation-conditioned features, significantly improving the ability of models to recognize and generalize interactions, especially for rare or unseen cases.

1.3 Structured Knowledge Integration via Neural-Logic Computing

While the ability of deep neural networks to learn complex patterns from vast amounts of visual data has revolutionized computer vision, current deep learning models operate primarily function in a sub-symbolic manner. As a result, they often struggle with systematic generalization beyond the statistical patterns observed in the training data. This limits their ability to address tasks that demand not just perception, but high-level understanding and advanced reasoning capabilities [324]. This thesis, therefore, investigates an alternative path towards more robust and trustworthy visual intelligence through the burgeoning field of neuro-symbolic computing [84, 85]. My specific focus has been on designing and implementing hybrid systems that combine the inductive learning power of neural networks with the deductive reasoning and explicit knowledge representation capabilities of symbolic systems [162, 166]. This aims to create visual understanding models that can not only perceive and interpret visual semantics but also reason about their interrelations, thereby fostering systems that are capable of handling complex situations

through structured reasoning rather than purely empirical pattern matching.

In Chapter 7, we address the problem of structured visual semantic parsing. Current semantic segmentation models often treat semantic concepts as a flat, unrelated set and lack mechanisms for explicit reasoning. We propose LogicSeg, a system that interprets visual scenes by explicitly considering the hierarchical relationships between semantic classes (*e.g.*, a chair is a type of furniture). The key idea is to represent this hierarchical knowledge using first-order logic. These logical rules are then translated into differentiable loss functions using fuzzy logic principles, allowing them to directly guide the training of networks. During inference, these logical constraints are used to refine predictions, ensuring the output is consistent with the predefined semantic structure. This allows LogicSeg to produce more robust and interpretable segmentation results that align with symbolic knowledge.

Building on the principles of neuro-symbolic computing, Chapter 8 tackles the complex task of Human-Object Interaction (HOI) detection. Understanding HOIs requires reasoning about the relationships between humans, objects, and actions. We introduce LogicHOI, a neural-logic framework designed to improve both the accuracy and generalization of HOI detection. The core innovation here is to explicitly incorporate commonsense knowledge about HOIs, such as object affordances (*e.g.*, a cup can be held) and proxemics (*e.g.*, riding implies human positioning over objects), into the learning process. This knowledge is formulated as first-order logic rules, which are then relaxed into continuous constraints via fuzzy logic to serve as optimization objectives for the neural network. By doing so, LogicHOI learns to predict interactions that are not only supported by visual evidence but also consistent with these structured commonsense rules, leading to better performance, especially in zero-shot scenarios where novel interactions must be inferred.

Chapter 2

Literature Review

This chapter surveys the key research areas that provide the foundation for this thesis. The review is organized according to the three primary themes of our work: (1) learning temporal structures from visual data, (2) modeling relational structures for scene understanding, and (3) integrating structured knowledge via neuro-symbolic computing. This structure provides a cohesive overview of the state-of-the-art and contextualizes the contributions made in the subsequent chapters.

2.1 Learning Temporal Structures from Visual Correspondence

2.1.1 Self-Supervised Temporal Correspondence Learning

In the video domain, correspondence matching plays a central role in many tasks (*e.g.*, video segmentation [114], flow estimation [62, 120] and object tracking [16, 45]). An emerging line of work tackles this problem in a self-supervised learning paradigm, by exploiting the temporal coherence in videos. One may group these work into two major classes. The first class of methods [307, 152, 151] poses a *colorization* proxy task, *i.e.*, reconstruct a query frame from an adjacent frame, according to their correspondence. The latter type of methods [313, 331, 172, 204, 122] performs forward and backward tracking and penalizes the inconsistency between the start and end positions of the tracked pixels or regions. The basic idea – *cycle-consistency* – is also adopted in un- supervised tracking [313, 384], optical flow [218, 403] and depth estimation [125, 363].

2.1.2 Self-Supervised Video Representation Learning

Correspondence learning approaches that use unlabeled video data fall in a broad field of self-supervised video representation learning. Towards learning transferable video representation, diverse pretext tasks are proposed to explore different intrinsic properties of videos as free supervisory signals, including temporal sequence ordering [225, 71, 334], predicting motion patterns [2, 295, 243, 79, 311], solving space-time cubic puzzles [143], anticipating future representations [199, 306], and temporally aligning videos [65, 59, 394, 395]. The learned representations are compact video descriptors that can be generalized to arious downstream tasks (*e.g.*, action recognition [103, 104, 337, 253], video captioning [283, 401], video retrieval [222]).

2.2 Relation Modeling for Structured Scene Understanding

2.2.1 Label Structure-aware Semantic Segmentation

Till now, only a rather small number of deep learning based segmentation models [343, 326, 330, 138, 167] are built with structured label taxonomies. The origin of this line of research can be traced back to the task of *image parsing* [326, 330, 294, 281, 282, 102, 361] raised in the pre-deep learning era. Basically, image parsing seeks for a holistic explanation of visual observation: scenes can be understood as a sum of novel objects, and the objects can be further broken down into fine-grained parts. In the deep learning era, the majority of structured segmentation models are dedicated to *human parsing* [400, 326, 330, 130], which is customized to human-part relation understanding. As for the case of general-purpose segmentation, there are far rare literature [343, 179, 176, 138, 167], and many of them incorporate label taxonomies into the network topology, losing generality [343, 179, 176]. As a notable exception, [167] converts the task as *pixel-wise multi-label classification* and exploits the class hierarchy for training regularization, with only trivial architectural change.

2.2.2 Human-Object Interaction Detection

According to the architecture design of networks, existing solutions for HOI detection can be broadly categorized into two groups: one-stage and two-stage. The one-stage methods [140, 181, 317, 69] typically employ a multi-task learning pipeline that jointly undertake the tasks of human-object detection and interaction classification in an end-to-end manner, therefore distinguished by fast inference. In contrast, two-stage methods [252, 308, 310, 174, 173, 368, 93, 399, 397, 33, 335] first detect entities with off-the-shelf detectors such as Faster R-CNN [258], and the predict the dense relationships among possible human-object pairs. This paradigm effectively disentangles the HOI detection process and results in improved performance. Inspired by DETR [24], recent advancements shift to adopt Transformer-based architectures [36, 141, 284, 402, 366, 391, 182, 388]. Several studies [119, 315, 254, 60, 182, 28] also supplement the Transformer-based HOI detectors with large-scale visual-linguistic models like CLIP [255] or visual knowledge [166, 325] to conduct logic-induced reasoning [162].

2.2.3 Knowledge Transfer from Diffusion Models

In light of the notable success achieved by diffusion models in applications, there is a growing interest in transferring knowledge acquired from large-scale pre-training to various tasks [248, 309, 184, 348, 206, 285, 372, 356]. For example, given the limited availability of data for constructing NeRFs and the unprecedented generalizability of diffusion models, researchers are motivated to explore generating 3D NeRFs via a 2D text-to-image diffusion model using diverse input text [248, 309, 184]. More recently, a notable trend has emerged where efforts are dedicated towards learning semantic representations from diffusion models by extracting intermediate feature maps. It finds diverse application in image segmentation [348], semantic correspondence learning [206, 285, 372], and general representation learning [356].

2.3 Structured Knowledge Integration via Neural-Logic Computing

2.3.1 Neuro-Symbolic Computing

There has been a line of research, called neural-symbolic computing (NSC), that pursues the integration of the symbolic and statistical paradigms of cognition [84, 136, 324]. NSC has a long history, dating back to McCulloch and Pitts 1943 paper [216], even before AI was recognized as a new scientific field. During 2000s, NSC received systematic study [247, 271, 291, 246]. Early NSC systems were meticulously designed for hard logic reasoning, but they are far less trainable, and fall short when solving real-world problems. NSC has recently ushered in its renaissance, since it shows promise of reconciling statistical learning of neural networks and logic reasoning of abstract knowledge – which is viewed as a key enable to the next generation of AI [154, 213]. Specifically, recent NSC systems [91, 92] show the possibility for modern neural networks to manipulate abstract knowledge with diverse forms of symbolic representation, including knowledge graph [50, 287, 183], propositional logic [115, 278, 350], and first-order logic [261, 58, 72]. They also demonstrate successful application in several domains and disciplines, *e.g.*, scientific discovery [268, 52], program generation [299, 233, 240], (visual) question-answering [303, 362], robot planning [227, 208, 354], and mathematical reasoning [4, 170, 155].

2.3.2 Compositional Generalization

Compositional generalization which pertains to the ability to understand and generate a potentially boundless range of novel conceptual structures comprised of similar constituents [139], has long been thought to be the cornerstone of human intelligence [74]. For example, human can grasp the meaning of *dax twice* or *dax and*

sing by learning the term *dax* [153], which allows for strong generalizations from limited data. In natural language processing, several efforts [145, 160, 77, 108, 3, 200, 272] have been made to endow neural networks with this kind of zero-shot generalization ability. Notably, the task proposed in [153], referred to as `SCAN`, involves translating commands presented in simplified natural language (*e.g.*, *dax twice*) into a sequence of navigation actions (*e.g.*, `I_DAX, I_DAX`). Active investigations into visual compositional learning also undergo in the fields of image caption [6, 201, 239] and visual question answering [1, 336, 12, 132]. For instance, to effectively and explicitly ground entities, [201] first creates a template with slots for images and then fills them with objects proposed by open-set detectors.

Chapter 3

Learning Temporal Structures: A Framework for Self-Supervised Correspondence Learning

In this chapter, I address the challenge of self-supervised temporal correspondence learning. By leveraging inter-video context enriched with position encoding, and promoting spatial compactness in affinity matrix between two observations, we aim to learn more robust and discriminative representations for the modeling of temporal structures without manual supervision.

3.1 Introduction

As a fundamental problem in computer vision, correspondence matching facilitates many applications, such as scene understanding [266], object dynamics modeling [114], and 3D reconstruction [87]. However, supervising representation for visual correspondence is not trivial, as obtaining pixel-level manual annotations is costly, and sometimes even prohibitive (due to occlusions and free-form object deformations). Although synthetic data would serve an alternative in some low-level visual correspondence tasks (*e.g.*, optical flow estimation [13]), they limit the generalization to real scenes. Using natural videos as a source of free supervision, *i.e.*, *self-supervised temporal correspondence learning*, is considered as appealing [152]. This is because videos contain rich realistic appearance and shape variations with almost infinite supply, and deliver valuable supervisory signals from the intrinsic coherence, *i.e.*, correlations among frames. Along this direction, existing solutions are typically built upon a *reconstruction* scheme (*i.e.*, each pixel from a 'query' frame is reconstructed by finding and assembling relevant pixels from adjacent frame(s))

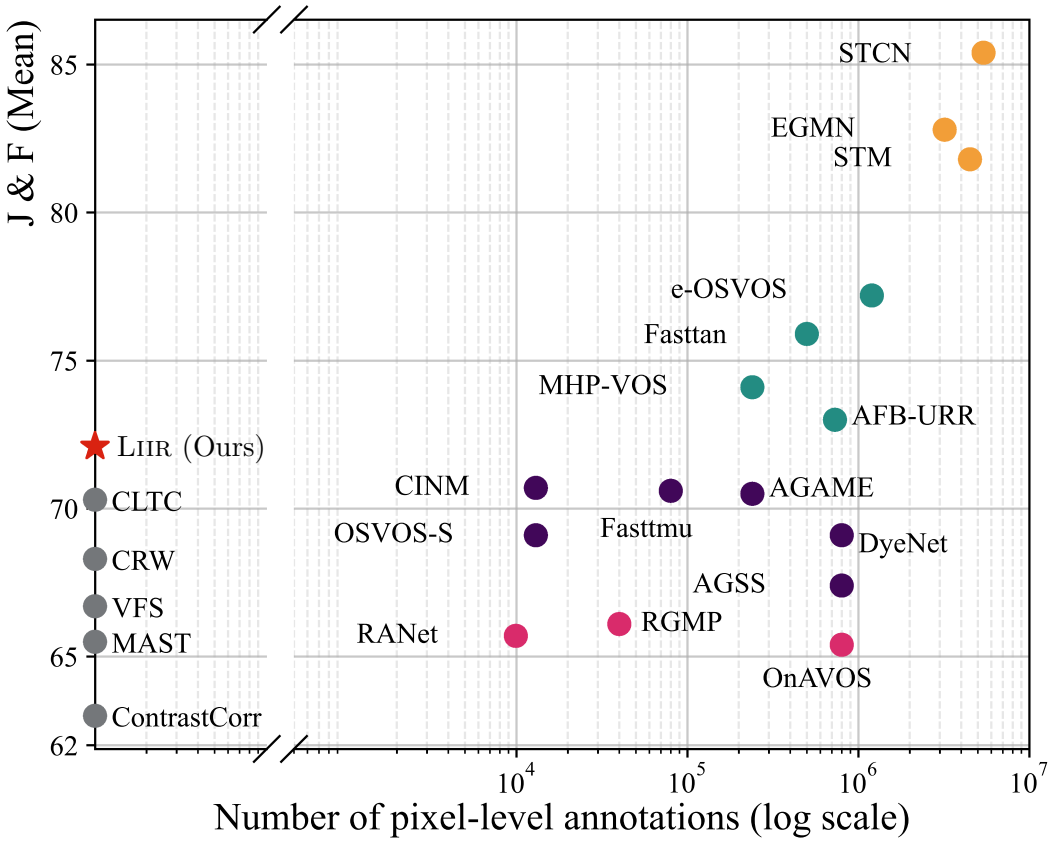


Figure 3.1 : **Performance comparison** over DAVIS_{17} `val`. Our LIIR surpasses all existing self-supervised methods, and is on par with many fully-supervised ones trained with massive annotations.

[307, 152, 151], and/or adopt a *cycle-consistent tracking* paradigm (*i.e.*, pixels/-patches are encouraged to fall into the same location after one cycle of forward and backward tracking) [313, 326, 172, 204, 122].

Unfortunately, these successful approaches largely neglect three crucial abilities for robust temporal correspondence learning, namely **instance discrimination**, **location awareness**, and **spatial compactness**. First, many of them share a narrow view that only considers intra-video context for correspondence learning. As it is hard to derive a free signal from a single video for identifying different object instances, the learned features are inevitably less instance-discriminative. Second, existing methods are typically built without explicit position representation. Such design seems counter-intuitive, given the extensive evidence that spatial position

is encoded in human visual system [109] and plays a vital role when human track objects [237]. Third, as the visual world is continuous and smoothly-varying, both spatial and temporal coherence naturally exist in videos. While numerous strategies are raised to address smoothness on the time axis, far less attention has been paid to the spatial case.

To fill in these three missing pieces to the puzzle of self-supervised correspondence learning, we present a locality-aware inter-and intra-video reconstruction framework – LIIR. **First**, we augment existing intra-video analysis based correspondence learning strategy with *inter-video context*, which is informative for instance-level separation. This leads to an inter-and intra-video reconstruction based training objective, that *inspires* intra-video positive correspondence matching, but *penalizes* unreliable pixel associations within and cross videos. We empirically verify that our inter-and intra-video reconstruction strategy can yield more discriminative features, that encode higher-level semantics beyond low-level intra-instance invariance modeled by previous algorithms. **Second**, to make our LIIR more location-sensitive, we learn to encode position information into the representation. Al- though position bias is favored for intra-video correspondence matching, it is undesired in the inter-video case. We thus devise a *position shifting* strategy to foster the strength and circumvent the weaknesses of position encoding. We experimentally show that, explicit position embedding benefits correspondence matching. **Third**, we involve a spatial compactness prior in intra-video pixel-wise affinity estimation, resulting in sparse yet compact associations. For each query pixel, the distribution of related pixels is fit by a mixture of Gaussians. This enforces each query pixel to match only a handful of spatially close pixels in adjacent frames. Our experiments show that such compactness prior not only regularizes training, but also removes outliers during inference.

These three contributions together make LIIR a powerful framework for self-

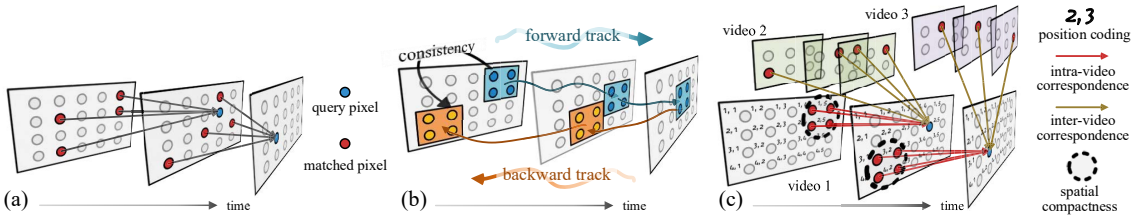


Figure 3.2 : **Illustration of different self-supervised architectures for temporal correspondence learning**: (a) reconstruction based, (b) cycle-consistency based, and (c) our LIIR that addresses instance discrimination, location awareness, and spatial compactness.

supervised correspondence learning. Without any adaptation, the learned representation is effective for various correspondence-related tasks, *i.e.*, video object segmentation, semantic part propagation, pose tracking. On these tasks, LIIR consistently outperforms unsupervised state-of-the-arts and is comparable to, or even better than, some task-specific fully-supervised methods (*e.g.*, Fig. 3.1).

3.2 Our Approach

We present LIIR, a self-supervised framework that learns dense correspondence from raw videos. Before elaborating on our model design (*cf.* §3.2.2), we first review the classic reconstruction based temporal correspondence learning stra- tegy (*cf.* §3.2.1), which serves as the basis of our LIIR.

3.2.1 Preliminary: Learning Temporal Correspondence through Frame Reconstruction

Due to the appearance continuity in video, one can con- sider pixels in a 'query' frame as being *copied* from some locations of other 'reference' frames. In light of this, a few studies [307, 152] raise a reconstruction-based correspondence learning scheme: each query pixel struggles to find pixels in a reference frame that can best reconstruct itself. Formally, let $I_q, I_r \in \mathbb{R}^{H\times W\times 3}$ respectively denote a query frame and

a reference frame from the same video. They are projected into a pixel embedding space by a ConvNet encoder (*e.g.*, ResNet [106]) $\phi : \mathbb{R}^{H\times W\times 3} \rightarrow \mathbb{R}^{h\times w\times c}$, such that $\boldsymbol{I}_q, \boldsymbol{I}_r = \phi(I_q), \phi(I_r)$. The *copy* operator can be approximated as an inter-frame affinity matrix $A \in [0,1]^{hw\times hw}$:

$$A(i,j) = \frac{\exp(\boldsymbol{I}_q(i)\cdot \boldsymbol{I}_r(j))}{\sum_{j'} \exp(\boldsymbol{I}_q(i)\cdot \boldsymbol{I}_r(j'))},\ i,j \in \{1,\cdots,hw\} \tag{3.1}$$

where $A(i,j) \in [0,1]$ refers to (i,j)-th element in A, signifying the similarity between pixel i in I_q and pixel j in I_r, and '$\cdot$' stands for the dot product. In this way, A gives the strength of all the pixel pair-wise correspondence between $\boldsymbol{I}_q$ and $\boldsymbol{I}_r$, according to which pixel i in I_q can be reconstructed by a weighted sum of pixels in I_r:

$$\hat{I}_q(i) = \sum\nolimits_j A(i,j)\ I_r(j). \tag{3.2}$$

The training objective of ϕ is hence defined as a reconstruction loss:

$$\mathcal{L}_{\text{res}} = ||I_q - \hat{I}_q||_2. \tag{3.3}$$

In practice, to avoid trivial solutions caused by information leakage, an *information bottleneck* is adopted over training samples, *e.g.*, RGB2gray operation [307], channel-wise drop- out in RGB [152] or Lab [151] colorspace. After training, the representation encoder ϕ is used for correspondence matching: similar to Eq. 3.2, the affinity A is estimated and used to *propagate* desired pixel-level entities (*e.g.*, instance masks, key-point maps), from a reference frame to a query frame.

3.2.2 LIIR: Locality-Aware Inter-and Intra-Video Reconstruction Framework

Building upon the reconstruction-by-copy scheme, LIIR is empowered with three crucial yet long overlooked abilities for robust correspondence learning: instance discrimination, location awareness, and spatial compactness.

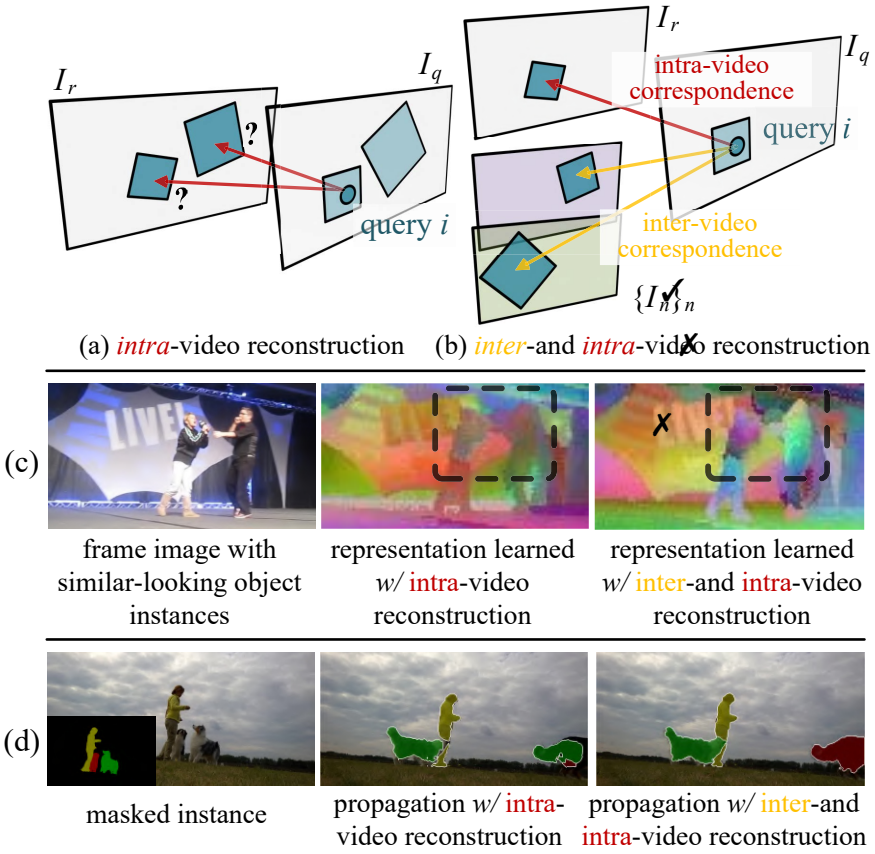


Figure 3.3 : **Inter-and intra-video reconstruction** (§3.2.2). (a) Pre- vious intra-video reconstruction based approaches struggle to offer supervisory signal for distinguishing between different instances. (b) In our inter-and intra-video reconstruction, each query pixel is forced to distinguish intra-video correspondence (→) from negative inter-video association (→), enabling cross-instance discrimi- nation. (c)-(d) Representation learned with inter-and intra-video reconstruction is more robust for multiple object instances.

Inter-and Intra-Video Reconstruction. With the computa- tion of the intra-video affinity A (Eq. 3.1), each query pixel is forced to distinguish its counterpart (positive) reference pixels from unrelated (negative) ones *within a same video*, with the indictor of the reconstruction quality $\mathcal{L}_{\text{res}}$ (Eqs. 3.2-3.3).

As both the positive and negative samples are sourced from the same video, there is less evidence for distinction among similar object instances with intra-video appearance only (Fig. 3.3(a)). As one single video only contains limited content, conducting correspondence matching within videos is less challenging, and inevitably hinders

the discrimination potential of the learned representation [314]. These insights motivate us to improve the intra-video affinity based reconstruction scheme by further accounting for negative *across-video* correspondence. Concretely, given the query (I_q) and reference (I_r) frames from the same video, an *intra-inter video affinity* $A' \in [0,1]^{hw \times hw}$ is computed:

$$A'(i,j) = \frac{\exp(\boldsymbol{I}_q(i) \cdot \boldsymbol{I}_r(j))}{\underbrace{\sum_{j'} \exp(\boldsymbol{I}_q(i) \cdot \boldsymbol{I}_r(j'))}_{\substack{\textit{intra}\text{-video} \\ \text{correspondence}}} + \underbrace{\sum_n \sum_k \exp(\boldsymbol{I}_q(i) \cdot \boldsymbol{I}_n(k))}_{\substack{\textit{inter}\text{-video} \\ \text{correspondence}}}}, \tag{3.4}$$

where $\{I_n\}_n$ refer to a collection of frames, which are sampled from the whole training dataset, except the source video of I_q (I_r). By additionally considering other irrelevant videos during affinity computation, both the quantity and diversity of negative samples are greatly improved, allowing us to derive a more challenging *inter-and intra-video reconstruction* scheme (Fig. 3.3(b)):

$$\hat{I}_q(i) = \sum\nolimits_j A'(i,j)\ I_r(j). \tag{3.5}$$

With Eqs. 3.4-3.5, each pixel i in the query frame I_q is required to distinguish its counterpart pixels from massive unrelated ones, which are from not only the reference frame I_r in current video, but a huge amount of irrelevant frames $\{I_n\}_n$ in other videos. This powerful idea, yet, is elegantly achieved by the same training objective as in Eq. 3.3. Note that Eq. 3.4 normalizes intra-video correspondence over both inter-and intra-video pixel-to-pixel relevance, while Eq. 3.5 only uses the pixels in the reference frame I_r for reconstruction. The rationale here is that, even if the encoder ϕ wrongly matches a query pixel i with a negative but similar-looking pixel k in I_n, *i.e.*, $\exp(\boldsymbol{I}_q(i) \cdot \boldsymbol{I}_n(k))$ will be large and $\sum_j A'(i,j) \ll 1$, the synthesized color $\hat{I}_q(i)$ will be *still* very different to $I_q(i)$ and ϕ will receive a large gradient from Eq. 3.3. Thus ϕ is driven to mine more high-level semantics and context-related clues, hence reinforcing the instance-level discrimination ability (Fig. 3.3(c)). Fig. 3.3(d) shows that the representation learned with our inter-and intra-video reconstruction strategy can distinguish similar-looking dogs nearby.

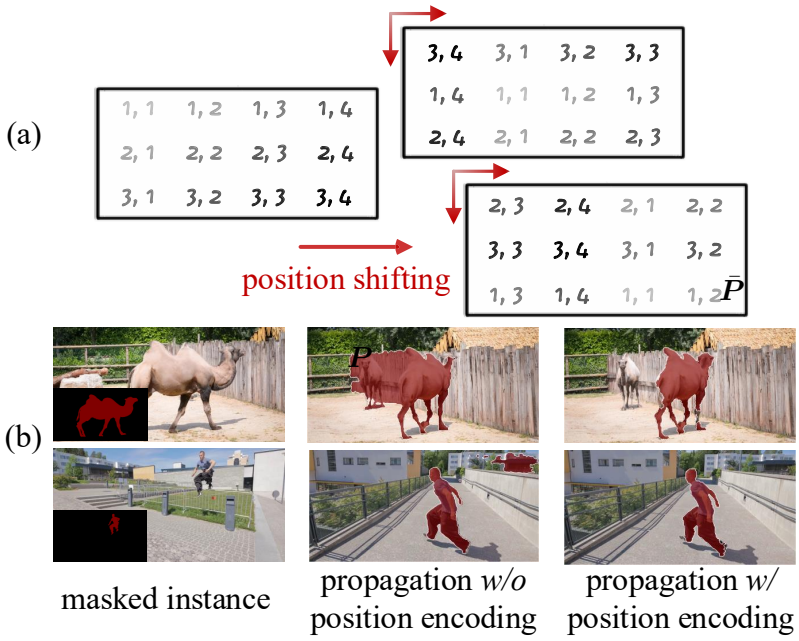


Figure 3.4 : (a) Illustration of **position shifting**. (b) Effect of **position encoding**. See §3.2.2 for details.

Although [314] also addresses inter-video analysis based reconstruction, it conducts embedding association on *patch level*, relying on a pre-trained tracker for patch alignment. Besides, the method consumes three loss terms for supervision, which is much more complicated than ours. Further, it separately conducts inter-and intra-video affinity based reconstruction. This is problematic; when both the reference frame in current video and an irrelevant frame from other videos contain query-like pixels, there is no explicit supervision signal to determine which one should be matched.

Position Encoding and Position Shifting. Plenty of literature in neuroscience has revealed that human visual system encodes both appearance and position information when we perceive and track objects [195, 109, 237]. Yet existing unsupervised correspondence methods put all focus on improving appearance based representation by ConvNets, ignoring the value of position information. Although [121, 137] suggest that ConvNets can implicitly capture position information by uti- lizing image boundary effects, explicit position encoding has already been a core of

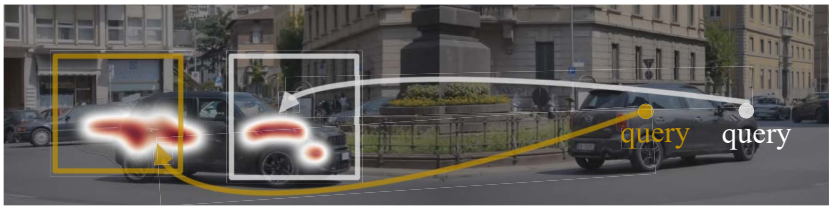
full attention networks (*e.g.*, Trans- former [301]), and facilitated a variety of tasks (*e.g.*, instance segmentation [332], tracking [193], video segmentation [48]). All these indicate that position encoding deserves more attention in the field of temporal correspondence learning. Along this direction, LIIR explicitly injects a position encoding map $\boldsymbol{P} \in \mathbb{R}^{h' \times w' \times c'}$ into the feature encoder ϕ:

$$\boldsymbol{I} = \phi(I, \boldsymbol{P}), \tag{3.6}$$

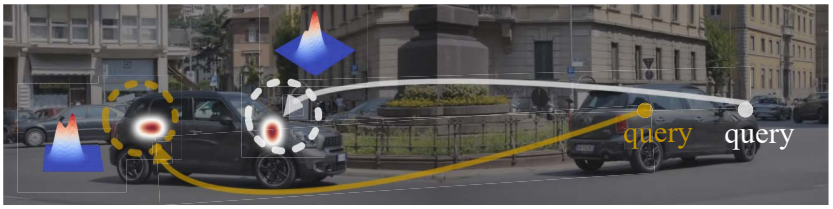
where $\boldsymbol{P}$ is added with the output feature of the first conv layer of ϕ and has the same size and dimension as the conv feature. We explore three **position encoding strategies**:

- *2D Sinusoidal Position Embedding* (2DSPE): $\boldsymbol{P}$ is given with a family of predefined sinusoidal functions, without introducing new trainable parameters:
$$\boldsymbol{P}(x, y, 2u) = \sin(x \cdot \varepsilon^{\frac{4u}{c'}}), \boldsymbol{P}(x, y, 2u+1) = \cos(x \cdot \varepsilon^{\frac{4u}{c'}}),$$
$$\boldsymbol{P}(x, y, 2v+\frac{c'}{2}) = \sin(y \cdot \varepsilon^{\frac{4v}{c'}}), \boldsymbol{P}(x, y, 2v+1+\frac{c'}{2}) = \cos(y \cdot \varepsilon^{\frac{4v}{c'}}),$$
where $x \in [0, w')$, $y \in [0, h')$ specify the horizontal and vertical positions, $u, v \in [0, c'/4)$ specify the dimension, and $\varepsilon = 10^{-4}$. The horizontal (vertical) positions are encoded in the first (second) half of the dimensions. 2DSPE naturally handles resolutions that are unseen during training.
- *1D Absolute Position Embedding* (1DAPE): 1DAPE is the most heavy-weight strategy: the whole $\boldsymbol{P}$ is a learnable parameter matrix without any constraint.
- *2D Absolute Position Embedding* (2DAPE): As in [61], two separate parameter sets: $\boldsymbol{X} \in \mathbb{R}^{w' \times c'/2}$ and $\boldsymbol{Y} \in \mathbb{R}^{h' \times c'/2}$, are learned for encoding the horizontal and vertical positions, respectively, and combined to generate $\boldsymbol{P}$.

With our intra-inter video affinity (Eq. 3.4), exploiting position information in intra-video correspondence matching, *i.e.*, $\{\exp(\boldsymbol{I}_q(i) \cdot \boldsymbol{I}_r(j))\}_j$, addresses local continuity resides in videos. However, for inter-video pixel relevance computation, *i.e.*, $\{\exp(\boldsymbol{I}_q(i) \cdot \boldsymbol{I}_n(k))\}_{n,k}$, such position prior is undesirable, as it inspires the query



(a) *w/o* spatial compactness prior



(b) *w/* spatial compactness prior

Figure 3.5 : Illustration of **spatial compactness prior** (§3.2.2).

pixel i in I_q to prefer matching these pixels with similar positions in other irrelevant videos $\{I_n\}_n$. To eliminate such position encoding induced bias from inter-video correspondence matching, we design a **position shifting** strategy (Fig. 3.4(a)). During training, for I_n from other videos, we circularly shift the position encoding vectors in $\boldsymbol{P}$ by a random step in horizontal and vertical axes, respectively. The reason why we adopt random shifting with circular boundary conditions, instead of random shuffling, is to preserve the spatial layout in the modulated position encoding map $\bar{\boldsymbol{P}}$. Then $\bar{\boldsymbol{P}}$ and I_n are fed into ϕ for inter-video correspondence matching, and $\bar{\boldsymbol{P}}$ related gradients are abandoned if learnable 1DAPE or 2DAPE is used. Note that the standard position encoding $\boldsymbol{P}$ is applied for the query (I_q) and reference (I_r) frames and updated normally. Fig. 3.4(b) intuitively shows that merging position information into visual representation can enable robust correspondence matching even with confusing background and fast motion. In §3.3.4, we will quantitatively verify that 1DAPE is more favored and indeed boosts the performance.

Spatial Compactness Prior. As the visual world is conti- nuous and smoothly-varying, it is reasonable to assume appearances in video data change smoothly both in spatial and temporal dimensions. For correspondence learning, the tem- poral

coherence has been extensively studied, while the spatial continuity received far less attention. To reduce search region, some existing methods [307, 152, 151] restrict corres- pondence matching within a local window, considering spatial regularities in a simple way. To make a better use of the spatial continuity, we augment the original reconstruction objective with an additional prior, termed as *spatial compa- ctness*. Such a prior poses constraints on the spatial distribution of associated pixels, leading to *sparse* and *coherent* solutions. Specifically, given the query (I_q) and reference (I_r) frames, we expect **i)** each query pixel i to be only matched with a small number of reference pixels, and **ii)** the matched reference pixels to be clustered. For a query pixel i and its matching 'heatmap': $A_i = [A(i,j)]_j \in [0,1]^{h \times w}$, w.r.t.$I_r$, we assume A_i follows a mixture of M 2D Gaussian distributions:

$$\mathcal{P}(x,y) = \sum\nolimits_{m=1}^{M} \omega_m \, \mathcal{N}(x,y|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m), \tag{3.7}$$

where (x,y) specifies the coordinate of a pixel location. We set $\{\boldsymbol{\mu}_m = [\mu_{x,m}, \mu_{y,m}]\}_m$ as the coordinates of top-M scores in A_i, and set $M = 2$ to address sparse robust matching. Other parameters, *i.e.*, $\{\boldsymbol{\Sigma}_m = [\sigma^2_{x,m}, 0; 0, \sigma^2_{y,m}]\}_m$, $\{\omega_m\}_m$, can be estimated efficiently from A_i without incurring high computational cost. In this way, we can derive a 'compact' matching heatmap $\tilde{A}_i \in [0,1]^{h \times w}$ for each query pixel i, and eventually have $\tilde{A} = [\tilde{A}_i]_i \in [0,1]^{hw \times hw}$. Such spatial compactness prior $\tilde{A}$ is fully aware of **i)** and **ii)**, and used to regularize our representation learning:

$$\mathcal{L}_{\text{com}} = ||\tilde{A} - A||_2. \tag{3.8}$$

Note that $\mathcal{L}_{\text{com}}$ is only applied for intra-video correspondence matching. Moreover, we replace A with $\tilde{A}$ during inference, which eliminates outliers effectively. Two recent fully supervised video segmentation methods also explore the local continuity via single-Gaussian locality prior [269] or top-k matching filtering [43], while both of them can be viewed as specific instances of our mixture-Gaussian based compactness prior, despite our different task settings. Fig. 3.5 shows that our spatial compactness

prior helps build reliable correspondence by inspiring sparse and compact solutions. Related experiments can be found in §3.3.4.

3.2.3 Implementation Details

Network Configuration. For fair comparison, our feature encoder ϕ is implemented as ResNet-18 [106] as in [307, 122, 349]. Following [152, 151, 128], $\times 2$ downsampling is only made in the third residual block. Thus ϕ finally outputs 256 feature maps of 1/4 size of the input, *i.e.*, $h{=}\frac{H}{4}, w{=}\frac{W}{4}, c{=}256$. The position embedding is added to feature after the first $7{\times}7$ `Conv-BN-ReLU` layer, *i.e.*, $h'{=}\frac{H}{2}, w'{=}\frac{H}{2}, c'{=}64$.

Training: LIIR is *trained from scratch* on two NVIDIA RTX-3090 GPUs and only uses the raw videos from Youtube-VOS [351]. Each training image is resized into $256{\times}256$ and channel-wise dropout in Lab colorspace [151] is adopted as the information bottleneck. Adam optimizer is used. At the initial 30 epochs, only intra-video reconstruction learning is adopted for warm-up, with a learning rate of 10^{-3} and batch size of 32. Then we conduct inter-video reconstruction learning with spatial compactness based regularization at the next 5 epochs, with a learning rate of 10^{-4} and batch size of 12. We online maintain a memory bank of 1,440 frames from different videos. For each query pixel, we sample 4 feature points from each stored frame, *i.e.*, a total of $1{,}440{\times}4$ negative samples are used for the inter-video correspondence computation, and we employ the moving average strategy [286, 341, 105] for parameter updating.

Testing: Once LIIR finishes training, there is no fine-tuning when applied to downstream tasks. Note that we utilize the compactness prior enhanced inter-frame affinity $\tilde{A}$ for mask propagation. As in [236, 151], we take multiple frames as reference for the full use of temporal context: at time step t, previous frames I_0, I_5, I_{t-5}, I_{t-3}, and I_{t-1} (if applicable) are referred for current-frame mask propagation.

Method	Backbone	Supervised	Dataset (size)	$\mathcal{J}$ & $\mathcal{F}$(Mean) ↑	$\mathcal{J}$(Mean) ↑	$\mathcal{J}$(Recall) ↑	$\mathcal{F}$(Mean) ↑	$\mathcal{F}$(Recall) ↑
Colorization [307]	ResNet-18		Kinetics (- , 800 hours)	34.0	34.6	34.1	32.7	26.8
CorrFlow [152]	ResNet-18		OxUvA (- , 14 hours)	50.3	48.4	53.2	52.2	56.0
TimeCycle [331]	ResNet-50		VLOG (- , 344 hours)	48.7	46.4	50.0	50.0	48.0
UVC [172]	ResNet-50		C + Kinetics (30k, 800 hours)	60.9	59.3	68.8	62.7	70.9
MuG [204]	ResNet-18		OxUvA (- , 14 hours)	54.3	52.6	57.4	56.1	58.1
MAST [151]	ResNet-18		Youtube-VOS (- , 5.58 hours)	65.5	63.3	73.2	67.6	77.7
CRW [122]	ResNet-18		Kinetics (- , 800 hours)	68.3	65.5	78.6	71.0	82.9
ContrastCorr [314]	ResNet-18		C + TrackingNet (30k, 300 hours)	63.0	60.5	-	65.5	-
VFS [349]	ResNet-18		Kinetics (- , 800 hours)	66.7	64.0	-	69.4	-
JSTG [383]	ResNet-18		Kinetics (- , 800 hours)	68.7	65.8	77.7	71.6	84.3
CLTC [128]†	ResNet-18		Youtube-VOS (- , 5.58 hours)	70.3	67.9	78.2	72.6	83.7
DINO [26]	ViT-B/8		I (1.28M, -)	71.4	67.9	-	**74.9**	-
LIIR	ResNet-18		Youtube-VOS (- , 5.58 hours)	**72.1**	**69.7**	**81.4**	74.5	**85.9**
ResNet [106]	ResNet-18	✓	I (1.28M, -)	62.9	60.6	69.9	65.2	73.8
OSVOS [22]	VGG-16	✓	I+D (1.28M, 10k)	60.3	56.6	63.8	63.9	73.8
FEELVOS [305]	Xception-65	✓	I + C + D + Y (1.28M, 663k)	71.5	69.1	79.1	74.0	83.8
STM [236]	ResNet-50	✓	I + D + Y (1.28M, 164k)	81.8	79.2	88.7	84.3	91.8

†: using task-specific architectures. I: ImageNet [57]. C: COCO [187]. D: DAVIS$_{17}$ [245].

Table 3.1 : **Quantitative results for video object segmentation** (§3.3.1) on DAVIS$_{17}$ [245] `val`. For size of datasets, we report (#*raw* images, length of *raw* videos) for self-supervised methods and (#image-level annotations, #pixel-level annotations) for supervised methods.

3.3 Experiment

We evaluate the learned representation on diverse video label propagation tasks, *i.e.*, video object segmentation (§3.3.1), body part propagation (§3.3.2), and pose keypoint tracking (§3.3.3). As in conventions [307, 122, 349], all these tasks are to propagate the first frame annotation to the whole video sequence, and we use our model to compute inter-frame dense correspondences. In §3.3.4, we conduct a set of ablative studies to examine the efficacy of our essential model designs.

Methods	Sup.	Overall	Seen		Unseen	
			$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$
Colorization [307]		38.9	43.1	38.6	36.6	37.4
CorrFlow [152]		46.6	50.6	46.6	43.8	45.6
MAST [151]		64.2	63.9	64.9	60.3	67.7
CLTC [128]†		67.3	66.2	67.9	63.2	71.7
LIIR		**69.3**	**67.9**	**69.7**	**65.7**	**73.8**
OSVOS [22]	✓	58.8	59.8	60.5	54.2	60.7
PreMVOS [205]	✓	66.9	71.4	75.9	56.5	63.7
STM [236]	✓	79.4	79.7	84.2	72.8	80.9

†: using task-specific architectures.

Table 3.2 : **Quantitative results for video object segmentation** (§3.3.1) on Youtube-VOS [351] `val`.

Methods	Sup.	VIP		JHMDB	
		mIoU ↑	AP ↑	PCK@0.1 ↑	PCK@0.2 ↑
ContrastCorr [314]		37.4	21.6	61.1	80.8
VFS [349]		39.9	-	60.5	79.5
CLTC [128]†		37.8	19.1	60.5	82.3
JSTG [383]		40.2	-	**61.4**	**85.3**
LIIR		**41.2**	**22.1**	60.7	81.5
ResNet [106]	✓	31.9	12.6	53.8	74.6
ATEN [276]	✓	37.9	24.1	-	-
TSN [393]	✓	-	-	68.7	92.1

†: using task-specific and architectures.

Table 3.3 : **Results for part propagation** (§3.3.2) and **pose tracking** (§3.3.3) on VIP [393] and JHMDB [129] `val`.

3.3.1 Results for Video Object Segmentation

Dataset. We first test our method on `val` sets of two popular video object segmentation datasets, *i.e.*, DAVIS$_{17}$ [245] and YouTube-VOS [351]. There are 30 and 474 videos in DAVIS$_{17}$ and YouTube-VOS `val` sets, respectively.

Evaluation Metric. Following the official protocol [245], we use the region similarity ($\mathcal{J}$) and contour accuracy ($\mathcal{F}$) as the evaluation metrics. Note that the scores on YouTube-VOS are respectively reported for *seen* and *unseen* categories, obtained from the official evaluation server.

Performance on DAVIS$_{17}$: As illustrated in Table 3.1, our LIIR consistently outperforms all existing self-supervised methods across all the evaluation metrics. For example, it surpasses current best-performing self-supervised method, *i.e.*, CLTC [128], in terms of mean $\mathcal{J}\&\mathcal{F}$ (**72.1** *vs.* 70.3). In addition, even without using *any* manual annotations for training, LIIR achieves very competitive segmentation performance in comparison with some famous supervised models [22, 305] trained with massive pixel-wise annotations.

Performance on YouTube-VOS `val`. Table 3.2 reports performance comparison of LIIR against four self-supervised competitors on YouTube-VOS `val`. It can be observed that LIIR sets new state-of-the-art. In particular, LIIR yields an overall score of **69.3%**, surpassing the second-best (*i.e.*, CLTC [128]) and third-best (*i.e.*, MAST [151]) approaches by **2.0%** and **5.1%**, respectively. Further, LIIR even outperforms some famous supervised methods (*i.e.*, OSVOS [22] and PreMVOS [205]), especially for the *unseen* categories, clearly demonstrating its remarkable generalization ability.

Qualitative Results. Fig. 3.6 depicts visual results on representative videos in the datasets. As seen, LIIR is able to establish accurate correspondences under various challenging scenarios, *e.g.*, scale changes, small objects and occlusions.

3.3.2 Results for Body Part Propagation

Dataset. We next evaluate our model performance for body part propagation. Experiments are conducted on VIP `val` [393], which contains 50 videos with annotations of 19 human semantic part categories (*e.g.*, hair, face, dress).

Evaluation Metric. As suggested by VIP [393], we adopt mean intersection-over-union (mIoU) and mean Average Precision (mAP) metrics for evaluation of semantic-level and instance-level parsing, respectively.

Performance. As shown in Table 3.3, LIIR achieves state-of-the-art performance on both semantic-level and instance-level parsing. This indicates that LIIR can generate strong representation which models both cross-instance discrimination and intra-instance invariance well. Fig. 3.7 depicts some visualization results on two representative videos. LIIR achieves temporally stable results and shows robustness to typical challenges (*e.g.*, pose variations, occlusions).

Figure 3.6 : **Qualitative results for video object segmentation** (§3.3.1), on DAVIS_{17} [245] `val` (left) and Youtube-VOS [351] `val` (right).

Figure 3.7 : **Qualitative results for part propagation** (§3.3.2) and **pose tracking** (§3.3.3), on VIP [393] `val` (left) and JHMDB [129] `val` (right).

3.3.3 Results for Pose Keypoint Tracking

Dataset. We then examine the model performance on human keypoint tracking, using JHMDB [129] `val`. JHMDB `val` has 268 videos. For each person, a total of 15 body joints, *e.g.*, torso, head, shoulder, elbow, are annotated.

Evaluation Metric. We use *probability of correct keypoint* (PCK) [357] to measure the accuracy between each tracking result and corresponding ground-truth with a threshold τ.

Performance. Table 3.3 shows that LIIR exhibits compelling overall performance. Note that CLTC [128] uses different checkpoints and model architectures for different tasks and datasets, while we only use a single model for evaluation. The visual results in Fig. 3.7 also demonstrate the strong capability of LIIR in establishing precise correspondence.

#	Inter-and Intra-Video Recons.	Position Encoding	Spatial Compactness	DAVIS $\mathcal{J}\&\mathcal{F}_m \uparrow$	VIP mIoU ↑
1				65.3	35.2
2	✓			68.7 (+3.4)	38.4 (+3.2)
3		✓		66.9 (+1.6)	37.0 (+1.8)
4			✓	68.4 (+3.1)	37.2 (+2.0)
5	✓	✓	✓	**72.1** (+6.8)	**41.2** (+6.0)

Table 3.4 : **Detailed analysis** of essential components of LIIR on DAVIS_{17} [245] `val` and VIP [393] `val`. See §3.3.4 for details.

3.3.4 Diagnostic Experiment

For further detailed analysis, we conduct a series of ablative studies on DAVIS_{17} [245] `val` and VIP [393] `val` sets.

Key Component Analysis. We first examine the efficacy of essential components of LIIR, *i.e.*, inter-and intra-video reconstruction, position encoding, and spatial compactness. The results are summarized in Table 3.4, where position encoding is implemented as 1DAPE, and compactness prior is used during both training and inference stages. When separately comparing row #2 - #4 with the baseline (MAST [151]) in row #1, we can observe that each individual module in- deed boosts the performance. For example, on DAVIS_{17} `val`, inter-and intra-video reconstruction, position encoding, and spatial compactness prior respectively bring **3.4%**, **1.6%**, and **3.1%** $\mathcal{J}\&\mathcal{F}$ gains. This verifies our core insight that these three elements are crucial for correspondence learning. Finally, in row #5, we combine all the three components together – LIIR, and obtain the best performance. This suggests that these modules are complementary to each other, and confirms the effectiveness of our whole design.

#	#Negative Samples	DAVIS $\mathcal{J}\&\mathcal{F}_m \uparrow$	VIP mIoU ↑
1	0	69.2	38.4
2	480	70.9 (+1.7)	39.8 (+1.4)
3	960	71.7 (+2.5)	41.0 (+2.6)
4	1,440	**72.1** (+2.9)	**41.2** (+2.8)

(a) Inter-and Intra-Video Recons.

Position Encoding	DAVIS $\mathcal{J}\&\mathcal{F}_m \uparrow$	VIP mIoU ↑
w/o PE	70.9	40.3
2DSPE	70.6 (-0.3)	40.2 (-0.1)
1DAPE	**72.1** (+1.2)	**41.2** (+0.9)
2DAPE	71.9 (+1.0)	41.1 (+0.8)

(b) Position Encoding

Position Modulation	DAVIS $\mathcal{J}\&\mathcal{F}_m \uparrow$	VIP mIoU ↑
N/A	71.3	40.6
shuffling	71.8 (+0.5)	41.0 (+0.4)
shifting	**72.1** (+0.8)	**41.2** (+0.6)

(c) Position Shifting

Spatial Compactness *training*	Spatial Compactness *inference*	DAVIS $\mathcal{J}\&\mathcal{F}_m \uparrow$	VIP mIoU ↑
		69.8	39.6
✓		71.5 (+1.7)	40.8 (+1.2)
	✓	70.8(+1.0)	40.3 (+0.7)
✓	✓	**72.1** (+2.3)	**41.2** (+1.6)

(d) Spatial Compactness Prior

Table 3.5 : **A set of ablation studies** on DAVIS$_{17}$ [245] `val` and VIP [393] `val`. See §3.3.4 for details.

Inter-and Intra-Video Reconstruction. We next study the impact of increasing the number of negative samples, *i.e.*, frames from other irrelevant videos used for inter-video correspondence computation (Eq. 3.4). In Table 3.5a, row #1 gives scores of learning without considering inter-video correspondence. In this case, the results are unsatisfactory. When more negative samples are involved (*i.e.*, 0→1,440), better performance can be achieved (*i.e.*, 69.2→72.1 on DAVIS$_{17}$ `val`, 38.4→41.2 on VIP `val`). Finally we use 1,440 negative samples for inter-video reconstruction based learning, which is the maximum number allowed by our GPU.

Position Encoding. To determine the effect of our position encoding module, we then report the performance with different encoding strategies in Table 3.5b. As

seen, the non-learnable strategy, 2DSPE, hinders the performance, while the learnable alternatives, *i.e.*, 1DAPE and 2DAPE, lead to better results. Compared with 2DAPE, 1DAPE is more favored, probably due to its high flexibility and capacity.

Position Shifting. We further study the influence of our position shifting strategy on performance in Table 3.5c. We consider two alternatives, *i.e.*, 'NAN' and 'position shuffling'. 'NAN' refers to using the normal position encoding map $\boldsymbol{P}$ without any modulation during inter-video correspondence matching. Compared with 'position shifting', 'NAN' suffers from performance degradation (*i.e.*, 72.1→71.3 on DAVIS_{17} `val`, 41.2→40.6 on VIP `val`), showing the negative effect of the position-induced bias. The other baseline, 'position shuffling', *i.e.*, randomly shuffling the position encoding map for inter-video affinity computation, though better than 'NAN', is still worse than 'position shifting'. This is because it destroys the spatial layouts.

Spatial Compactness Prior. The spatial compactness prior (Eq. 3.7) is used to regularize intra-video correspondence matching during both training and inference stages. In Table 3.5d, we quantitatively identify the performance contribution of our spatial compactness prior in different stages.

3.4 Conclusion

In this chapter, we presented a self-supervised temporal correspondence learning approach, LIIR, that makes contributions in three aspects. First, going beyond the popular intra-video analysis based learning scheme, we further enforce separation between intra- and inter-video pixel associations, enhancing instance-level feature discrimination. Second, with a clever position shifting strategy, we bring the advantages of position encoding into full play, while avoiding its undesirable impact at the same time. Third, a spatial compactness prior is introduced to regularize representation learning and improve correspondence inference. The effectiveness was thoroughly validated over various label propagation tasks.

Chapter 4

Learning Temporal Structures: Unified Mask Embedding for Self-Supervised Video Segmentation

Continuing our exploration within learning temporal structures from visual correspondence, in this chapter, we specializes these principles to a address the self-supervised video object segmentation task.

4.1 Introduction

We focus on a classic computer vision task: accurately segmenting desired objects in a video sequence, where the target objects are defined by pixel-wise masks in the first frame. This task is referred as (*one-shot*) *video object segmentation* (VOS) or *mask propagation* [327], playing a vital role in video editing and self-driving. Prevalent solutions [124, 244, 114, 39, 202, 220, 217, 44, 236, 359, 22, 211, 17, 260, 235, 305, 339, 63, 346, 180, 116, 322, 319] are built upon *fully supervised* learning techniques, costing intensive labeling efforts. In contrast, we aim to learn VOS from *unlabeled* videos — *self-supervised* VOS.

Due to the absence of mask annotation during training, existing studies typically degrade such self-supervised yet *mask-guided segmentation* task as a combo of *unsupervised correspondence learning* and correspondence based, *non-learnable mask warping* (*cf.* Fig. 4.1(a)). They first learn pixel- /patch-wise matching (*i.e.*, cross-frame correspondence) by exploring the inherent continuity in raw videos as free supervisory signals, in the form of **i)** a *photometric reconstruction* problem where each pixel in a target frame is desired to be recovered by *copying* relevant pixels in refer-

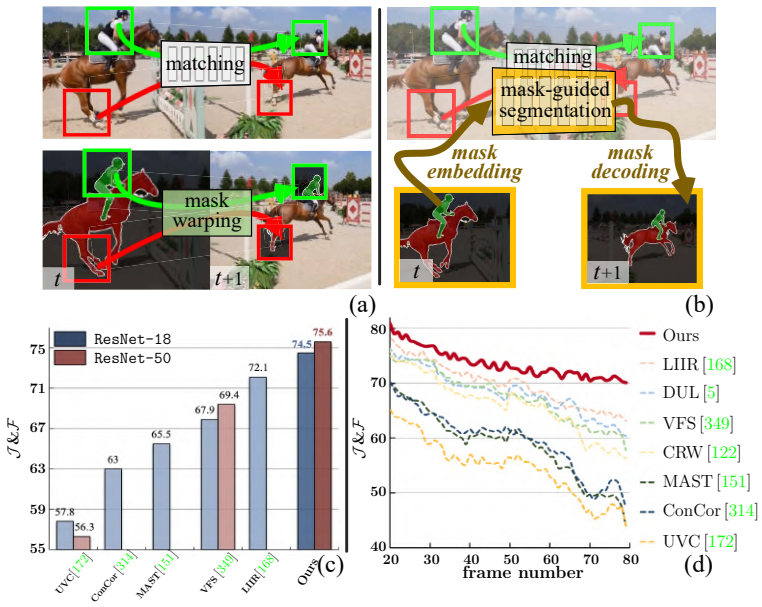


Figure 4.1 : (a) Correspondence learning based self-supervised VOS, where mask tracking is simply degraded as correspondence matching mask warping. (b) We achieve self-supervised VOS by jointly learning mask embedding and correspondence matching. Our algorithm explicitly embeds masks for target object modeling, hence enabling mask-guided segmentation. (c) Performance comparison and (d) Performance over time, reported on DAVIS_{17} [245] `val`.

ence frame(s) [307, 152, 151, 146, 353, 168]; **ii)** a *cycle-consistency* task that enforces matching of pixels/patches after forward-backward tracking [326, 313, 122, 172, 19]; and **iii)** a *contrastive matching* scheme that contrasts confident correspondences against unreliable ones [128, 5, 349, 275]. Once trained, the dense matching model is used to approach VOS in a cheap way (Fig.4.1(a)): the label of a query pixel/patch is simply borrowed from previously segmented ones, according to their appearance similarity (correspondence score).

Though straightforward, these correspondence based "expedient" solutions come with two severe limitations: **First**, they learn to match pixels instead of customizing VOS target – mask-guided segmentation, leaving a significant gap between the training goal and task/inference setup. During training, the model is optimized

purely to discovery reliable, target-agnostic visual correlations, with no sense of object-mask information. Spontaneously, during testing/inference, the model struggles in employing first-/prior-frame masks to guide the prediction of succeeding frames. **Second**, from the view of mask-tracking, existing self-supervised solutions, in essence, adopt an obsolete, matching-/flow-based mask pro- pagation strategy [10, 123, 7, 321, 323]. As discussed even before the deep learning era [66, 68, 320], such a strategy is sub-optimal. Specifically, without modeling the target objects, flow-based mask warping is sensitive to outliers, resulting in error accumula- tion over time [327]. Subject to the primitive matching-and-copy mechanism, even trivial errors are hard to be corrected, and often lead to much worse results caused by drifts or occlusions. This is also why current top-leading *fully supervised* VOS solutions [22, 211, 17, 260, 235, 114, 39, 236, 305, 339, 359, 63, 346, 180] largely follow a *mask embedding learning* philosophy — embedding *frame-mask pairs*, instead of only frame images, into the segmentation network. With such explicit modeling of the target object, more robust and accurate mask-tracking can be achieved [327, 387].

Motivated by the aforementioned discussions, we integrate mask embedding learning and dense correspondence modeling into a compact, end-to-end framework for self-supervised VOS (*cf.* Fig. 4.1(b)). This allows us to inject the mask-tracking nature of the task into the very heart of our algorithm and model training. However, bringing the idea of mask embedding into self-supervised VOS is not trivial, due to the lack of mask annotation. We therefore achieve mask embedding learning in a *self-taught* manner. Concretely, our model is trained by alternating between **i)** space-time pixel clustering, and **ii)** mask-embedded segmentation learning. Pixel clustering is to automatically discover spatiotempo- rally coherent object(-like) regions from raw videos. By utilizing such pixel-level video partitions as pseudo ground-truths of target objects, our model can learn how to extract target-specific context from frame-mask pairs, and how to leverage such high-level context to predict the

next-frame mask. At the same time, such self-taught mask embedding scheme is consolidated by self-supervised dense correspondence learning. This allows our model to learn transferable, locally discriminative representations by making full use of the spatiotemporal coherence in natural videos, and prevent the degenerate solution of the deterministic clustering.

Our approach owns a few distinctive features: **First**, it has the ability of directly learning to conduct mask-guided sequential segmentation; its training objective is completely aligned with the core nature of VOS. **Second**, by learning to embed object-masks into mask tracking, target-oriented context can be efficiently mined and explicitly leveraged for object modeling, rather than existing methods merely relying on local appearance correlations for label "copying". Hence our approach can reduce error accumulation (*cf.* Fig. 4.1(d)) and perform more robust when the latent correspondences are ambiguous, *e.g.*, deformation, occlusion or one-to-many matches. **Third**, our mask embedding strategy endows our self-supervised framework with the potential of being empowered by more advanced VOS model designs developed in the fully-supervised learning setting.

Through embracing the powerful idea of mask embedding learning as well as inheriting the merits of correspondence learning, our approach favorably outperforms state-of-the-art competitors, *i.e.*, **3.2**%, **2.5**%, and **2.2**% mIoU gains on DAVIS_{17} [245] `val`, DAVIS_{17} `test-dev` and YouTube-VOS [351] `val`, respectively. In addition to narrowing the performance gap between self- and fully-supervised VOS, our approach establishes a tight coupling between them in the aspect of model design. We expect this work can foster the mutual collaboration between these two relevant fields.

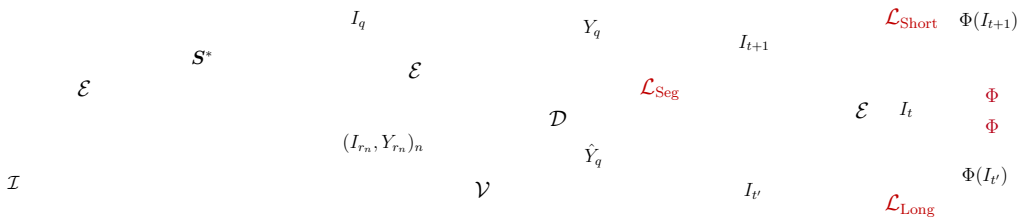


Figure 4.2 : Our self-supervised VOS framework: **(a-b)** space-time pixel clustering based mask embedding learning (§4.2.2) for the whole network (including $\mathcal{E}$, $\mathcal{V}$, and $\mathcal{D}$), and **(c)** short- and long-term correspondence learning (§4.2.3) for the visual encoder $\mathcal{E}$ only.

4.2 Methodology

4.2.1 Algorithm Overview

Learning VOS in a self-supervised manner is appealing, as it eliminates the heavy annotation budget required by the fully supervised algorithms. Due to the absence of mask annotation, existing self-supervised methods take an *expedient* solution: they learn to find correspondence between two frames, instead of learning mask-guided segmentation. During inference, the first-frame mask is directly copied to the rest frames based on cross-frame correspondence. Specifically, given two frames I_r and I_q, their dense representations $\boldsymbol{I}_r, \boldsymbol{I}_q \in \mathbb{R}^{HW \times D}$ are first extracted by a shallow neural encoder $\mathcal{E}$ (typically `ResNet-18` [106]), and their pairwise affinity matrix can be computed as:

$$A_r^q = \texttt{softmax}(\boldsymbol{I}_r \boldsymbol{I}_q^\top) \quad \in [0,1]^{HW \times HW}, \tag{4.1}$$

where `softmax` is row-wise. The resultant affinity A_r^q gives the strength of all the pixel pairwise correspondence between $\boldsymbol{I}_r$ and $\boldsymbol{I}_q$. One main benefit is that, once $\mathcal{E}$ is trained, it can be used to estimate cross-frame correspondence; then VOS is approached by warping the mask Y_r of a reference frame I_r to the query frame I_q based on: $Y_q = A_r^{q\top} Y_r$. Thus the central problem is to design a surrogate task to supervise $\mathcal{E}$ to estimate reliable intra-frame affinity A_r^q.

Our self-supervised VOS solution, at a high level, jointly learns mask embedding and visual correspondence from raw videos. It absorbs the powerful idea of mask-embedded segmentation in fully supervised VOS; meanwhile, it inherits the merits of existing unsupervised correspondence based regime (*cf.* Eq. 4.1) in learning generic, dense features. As a result, our solution can be formulated as (*cf.* Fig. 4.2):

$$Y_q = \mathcal{D}\big(\mathcal{E}(I_q), \{\mathcal{V}([I_{r_n}, Y_{r_n}])\}_n\big) \tag{4.2}$$

self-supervised dense correspondence learning §4.2.3 — self-supervised mask embedding learning §4.2.2

where $[\,\cdot\,]$ stands for concatenation. Basically, our model utilizes a set of reference frame-mask pairs, *i.e.*, $(I_{r_n}, Y_{r_n})_n$, to predict/decode the mask of each query frame I_q, learnt in a self-supervised manner. Our model has three core parts:

- ***Visual Encoder*** $\mathcal{E}$, which maps each query frame I_q into a dense representation tensor: $\boldsymbol{I}_q = \mathcal{E}(I_q) \in \mathbb{R}^{HW \times D}$. We instantiate $\mathcal{E}$ as `ResNet-18` or `ResNet-50`.
- ***Frame-Mask Encoder*** $\mathcal{V}$ for mask embedding. It takes a pair of a reference frame I_r and corresponding mask Y_r as inputs, and extracts target-specific context, *i.e.*, $\boldsymbol{V}_r = \mathcal{V}([I_r, Y_r]) \in \mathbb{R}^{HW \times D'}$, to guide the segmentation/mask decoding of I_q. $\mathcal{V}$ has a similar network architecture with $\mathcal{E}$, but the input and output dimensionality are different and the network weights are unshared.
- ***Mask Decoder*** $\mathcal{D}$, which is a small CNN for mask decoding. With the help of target-rich context $\{\boldsymbol{V}_{r_n}\}_n$ collected from a set of reference frame-mask pairs $\{(I_{r_n}, Y_{r_n})\}_n$, $\mathcal{D}$ makes robust prediction, *i.e.*, Y_q, for the query frame I_q.

As for training, to mitigate the dilemma caused by the absence of true labels of $\{Y_{r_n}\}_n$ and Y_q, we conduct unsupervised space-time pixel clustering for automatic mask creation and train the whole network, including $\mathcal{E}$, $\mathcal{V}$, and $\mathcal{D}$, for mask embedding and decoding (§4.2.2). Moreover, unsupervised contrastive correspondence learning (§4.2.3) is introduced to boost dense visual representation learning of $\mathcal{E}$.

4.2.2 Self-supervised Mask Embedding Learning

For self-supervised mask embedding learning, we alternatively perform two steps: **Step 1**: clustering of video pixels on the visual feature space $\mathcal{E}$ so as to generate spatiotemporally compact segments; and **Step 2**: the space-time cluster assignments serve as pseudo masks to supervise our whole network (including $\mathcal{E}$, $\mathcal{V}$, and $\mathcal{D}$), which learns VOS as mask-embedded sequential segmentation. After that, the improved visual representation $\mathcal{E}$ will in turn facilitate clustering.

Step 1: Space-time Clustering. The goal of this step is to partition each training video $\mathcal{I}$ into M space-time consistent segments (see Fig. 4.2(a)). For each pixel $i \in \mathcal{I}$, let $\boldsymbol{i} \in \mathbb{R}^D$ denote its visual embedding (extracted from the visual encoder $\mathcal{E}$), and $\boldsymbol{s}_i \in \{0,1\}^M$ its one-hot cluster assignment vector. Clustering of all the pixels in $\mathcal{I}$ into M clusters can be achieved by solving the following optimization problem:

$$\min_{\boldsymbol{C},\boldsymbol{S}} \sum\nolimits_{i\in\mathcal{I}} \|\boldsymbol{i} - \boldsymbol{C}\boldsymbol{s}_i\|, \quad s.t. \;\; \boldsymbol{s}_i \in \{0,1\}^M, \quad \mathbf{1}^\top \boldsymbol{s}_i = 1. \tag{4.3}$$

Here $\boldsymbol{C} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M] \in \mathbb{R}^{D\times M}$ is the cluster centroid matrix, where $\boldsymbol{c}_m \in \mathbb{R}^D$ refers to the centroid of m-th cluster, and $\boldsymbol{S} = [\boldsymbol{s}_i]_i$ stores the cluster assignments of all the pixels in $\mathcal{I}$. $\mathbf{1}$ is a M-dimensional all-one vector. While many clustering methods have been designed to solve Eq. 4.3, for simplicity, we use the most classic one – k-means, which finds the optimal $\boldsymbol{C}^*$ and $\boldsymbol{S}^*$ in an EM fashion. Moreover, to pursue spatiotemporally compact clusters, for each pixel $i \in \mathcal{I}$, we supply its visual embedding $\boldsymbol{i}$ with a 3D sinusoidal position encoding vector [302, 61]. In practice, only a small number of EM steps (*i.e.*, 100) can deliver satisfactory clustering results, taking about 2 seconds per video, averaged on our training dataset – YouTube-VOS [351].

Step 2: Mask-embedded Segmentation Learning. In this step, our model utilizes clustering results as pseudo ground-truths (see Fig. 4.2(b)), to directly learn VOS as mask embedding and decoding. For each training video $\mathcal{I}$, we sample $N+1$

frames $\{I_{r_1}, I_{r_2}, \cdots, I_{r_N}, I_q\}$ and their masks $\{Y_{r_1}, Y_{r_2}, \cdots, Y_{r_N}, Y_q\}$, as training examples. The pseudo masks are naturally derived from the assignment matrix $\boldsymbol{S}^*$, corresponding to the pixel-level assignment of a certain cluster. The training examples are used to teach our model to refer to the first N frame-mask pairs $\{(I_{r_n}, Y_{r_n})\}_n$ to segment the last query frame I_q — predicting Y_q. As such, our model can learn i) *mask embedding*: how to extract target-specific context from $\{(I_{r_n}, Y_{r_n})\}_n$; and ii) *mask decoding*: how to make use of target-specific context to segment the target in I_q.

More specifically, we first respectively apply our visual encoder $\mathcal{E}$ and frame-mask encoder $\mathcal{V}$ over each reference frame I_{r_n} and each reference frame-mask pair (I_{r_n}, Y_{r_n}), to obtain visual and target-specific embeddings:

$$\boldsymbol{I}_{r_n} = \mathcal{E}(I_{r_n}) \in \mathbb{R}^{HW \times D}, \quad \boldsymbol{V}_{r_n} = \mathcal{V}([I_{r_n}, Y_{r_n}]) \in \mathbb{R}^{HW \times D'}. \tag{4.4}$$

We respectively stack all the reference visual and target-specific embeddings: $\boldsymbol{I}_r = [\boldsymbol{I}_{r_1}, \cdots, \boldsymbol{I}_{r_N}] \in \mathbb{R}^{NHW \times D}$, and $\boldsymbol{V}_r = [\boldsymbol{V}_{r_1}, \cdots, \boldsymbol{V}_{r_N}] \in \mathbb{R}^{NHW \times D'}$. To leverage $\boldsymbol{V}_r$ to boost the prediction of I_q, we need to mine useful context, related to I_q, from $\boldsymbol{V}_r$. Given the visual embedding $\boldsymbol{I}_q \in \mathbb{R}^{HW \times D}$ of I_q (extracted from $\mathcal{E}$), we estimate the affinity between the query I_q and reference frames $\{I_{r_n}\}_n$ (analogous to Eq. 4.1):

$$A = \texttt{softmax}(\boldsymbol{I}_r \boldsymbol{I}_q^\top) \ \in \mathbb{R}^{NHW \times HW}. \tag{4.5}$$

Hence target-specific, supportive features are accordingly assembled to yield:

$$\boldsymbol{V}_q = A^\top \boldsymbol{V}_r \ \in \mathbb{R}^{HW \times D'}. \tag{4.6}$$

Here $\boldsymbol{V}_q$ absorbs existent object observations in the reference set $\{(I_{r_n}, Y_{r_n})\}_n$, revealing for I_q whether each pixel thereof belongs to the target object or not. Given precise segmentation groundtruths, it is relatively easy for fully supervised methods [236, 44, 269] to learn to directly decode $\boldsymbol{V}_q$ into segmentation mask. However, this strategy does not work well in our case since the pseudo labels are inevitably noisy and less accurate, compared with the real groundtruths. To tackle this, we achieve mask decoding through a *mask refinement* scheme, which makes more explicit use of reference masks. Specifically, we first construct a coarse mask $\bar{Y}_q$ for I_q

by warping the reference masks $\{Y_{r_n}\}_n$ *w.r.t.* the affinity A:

$$\bar{Y}_q = A^\top [Y_{r_1}, Y_{r_2}, \cdots, Y_{r_n}] \ \in \mathbb{R}^{HW}. \tag{4.7}$$

The segmentation prediction $\hat{Y}_q$ for the query I_q is made as:

$$\hat{Y}_q = \mathcal{D}([\boldsymbol{V}_q, \bar{\boldsymbol{V}}_q]), \qquad \bar{\boldsymbol{V}}_q = \mathcal{V}([I_q, \bar{Y}_q]) \ \in \mathbb{R}^{HW \times D'}. \tag{4.8}$$

Here the frame-mask encoder $\mathcal{V}$ (*cf.* Eq. 4.4) is smartly revoked to get another target-specific embedding $\bar{\boldsymbol{V}}_q$, from the pair of the query frame I_q and warped coarse mask $\bar{Y}_q$. This also elegantly resembles the mask copying strategy adopted in existing correspondence-based self-supervised VOS models. Conditioned on the concatenation of $\boldsymbol{V}_q$ and $\bar{\boldsymbol{V}}_q$, the mask decoder $\mathcal{D}$ outputs a finer mask $\hat{Y}_q$. In practice we find our mask refinement strategy can ease training and bring better

performance (related experiments can be found in Table 4.4e).

Given the pseudo segmentation label Y_q and prediction $\hat{Y}_q$ of I_q, our whole model is supervised by minimizing the standard cross-entropy loss $\mathcal{L}_{\text{CE}}$:

$$\mathcal{L}_{\text{Seg}} = \sum\nolimits_{\mathcal{I}} \mathcal{L}_{\text{CE}}(\hat{Y}_q, Y_q). \tag{4.9}$$

4.2.3 Self-supervised Dense Correspondence Learning

An appealing aspect of our mask embedding framework is that it is general enough to naturally incorporate unsupervised correspondence learning to specifically reinforce visual representation $\mathcal{E}$. This comes with a few advantages: First, this allows our model to exploit the inherent coherence in natural videos as free supervisory signals to promote the transferability and sharpen the discriminativeness of $\mathcal{E}$. Second, correspondence learning provides initial meaningful features for clustering (*cf.* Eq. 4.3), which is prone to degeneracy (*i.e.*, allocating most samples to the same cluster) caused by poor initialization [25]. Third, our segmentation model involves the computation of intra-frame affinity A (*cf.* Eqs. 4.4-4.6), raising a strong demand for efficiently modeling dense correspondence within our framework. Along

with recent work of contrastive matching based correspondence learning [128, 5, 349], we comprehensively explore intrinsic continuity within raw videos in both *short-term* and *long-term* time scales, to boost the learning of $\mathcal{E}$ (see Fig. 4.2(c)).

Short-term Appearance Consistency. Temporally adjacent frames typically exhibit continuous and trivial appearance changes [118, 204]. To accommodate this property, we enforce *transformation-equivariance* [232, 289, 288, 234] between our adjacent frame representations. Given two **successive** frames $I_t, I_{t+1} \in \mathcal{I}$, their representations, delivered by $\mathcal{E}$, are constrained to be **equivariant** against geometric transformations (*i.e.*, scaling, flipping, and cropping). Specifically, denote Φ as a random transformation, our *equivariance based short-term appearance consistency constraint* can be expressed as:

$$\left.\begin{array}{l} ❶\ \mathcal{E}(I_t) \approx \mathcal{E}(I_{t+1}) \\ \text{short-term consistency} \\ ❷\ \mathcal{E}(\Phi(I_t)) = \Phi(\mathcal{E}(I_t)) \\ \text{transformation-equivariance} \end{array}\right\} \Rightarrow \mathcal{E}(\Phi(I_t)) \approx \Phi(\mathcal{E}(I_{t+1}))\ \ ❸. \tag{4.10}$$

Here ❶ states the short-term consistency property; ❷ refers to the equivariance constraint on a single image [234], *i.e.*, an imagery transformation Φ of I_t should lead to a correspondingly transformed feature [5]. By bringing ❷ into ❶, we prevent trivial solution, *i.e.*, $\mathcal{E}(I_t) \equiv \mathcal{E}(I_{t+1})$, when directly optimizing $\mathcal{E}$ via ❶, and eventually get ❸. Following ❸, we first get the feature of transformed I_t: $\boldsymbol{X}'_t = \mathcal{E}(\Phi(I_t)) \in \mathbb{R}^{HW \times D}$, and transformed feature of I_{t+1}: $\boldsymbol{X}_{t+1} = \Phi(\mathcal{E}(I_{t+1})) \in \mathbb{R}^{HW \times D}$. Denote k-*th* pixel feature of $\boldsymbol{X}_{t+1}$ (resp. $\boldsymbol{X}'_t$) as $\boldsymbol{x}^k_{t+1} \in \mathbb{R}^D$ (resp. $\boldsymbol{x}'^k_t \in \mathbb{R}^D$)*, our short-term consistency loss is computed as:

$$\mathcal{L}_{\text{Short}} = -\sum\nolimits_{\mathcal{I}} \sum\nolimits_{k} \log \frac{\exp(\langle \boldsymbol{x}^{k\top}_{t+1} \boldsymbol{x}'^k_t \rangle)}{\sum_l \exp(\langle \boldsymbol{x}^{k\top}_{t+1} \boldsymbol{x}'^l_t \rangle)}, \tag{4.11}$$

*For clarity, the symbols for frame and pixel features in §4.2.3 are slightly redefined as $\boldsymbol{X}$ and $\boldsymbol{x}$, instead of using $\boldsymbol{I}$ and $\boldsymbol{i}$ as in §4.2.2.

where $\langle \boldsymbol{x}_{t+1}^{k\top} \boldsymbol{x}_t^{\prime l} \rangle$ gives cosine similarity based affinity between k-*th* pixel feature of $\boldsymbol{X}_{t+1}$ and l-*th* pixel feature of $\boldsymbol{X}_t^{\prime}$. Eq. 4.11 captures local appearance continuity by contrasting affinity between aligned pixel feature pairs, *i.e.*, $\boldsymbol{x}_{t+1}^{k}$ and $\boldsymbol{x}_t^{\prime k}$ against non-corresponding ones, *i.e.*, $\boldsymbol{x}_{t+1}^{k}$ and $\{\boldsymbol{x}_t^{\prime l}\}_{l \neq k}$, with an extra transformation equivariance based constraint.

Long-term Semantic Dependency. In addition to considering the local consistency among adjacent frames, we exploit long-term coherence of visual content among distant frames [226, 352]. To address this property, we enforce transformation equivariance between representations of *arbitrary* frame pairs (sampled from the same video) after *alignment*. Given two **distant** frames $I_t, I_{t'} \in \mathcal{I}$ (*s.t.* $|t - t'| \geq 5$), their representations, after being **aligned** *w.r.t.* their affinity $A_{t'}^{t}$, are constrained to be **equivariant** against geometric transformations. In particular, denote $A_{t'}^{t} \in [0,1]^{HW \times HW}$ (resp. $A_{\Phi(t')}^{t} \in [0,1]^{HW \times HW}$) as the affinity between I_t and $I_{t'}$ (resp. I_t and $\Phi(I_{t'})$), our *equivariance based long-term semantic dependency constraint* can be expressed as:

$$\left.\begin{array}{l} \text{❹ } \mathcal{E}(I_t) \approx A_{t'}^{t\top} \mathcal{E}(I_{t'}) \\ \text{long-term dependency} \\ \\ \text{❷ } \mathcal{E}(\Phi(I_t)) = \Phi(\mathcal{E}(I_t)) \\ \text{transformation-equivariance} \end{array}\right\} \Rightarrow \mathcal{E}(I_t) \approx A_{\Phi(t')}^{t\top} \Phi(\mathcal{E}(I_{t'}))\ \text{❺}. \tag{4.12}$$

Here ❹ states the long-term dependency property; ❷ poses the equivariance constraint, as in Eq. 4.10. By bringing ❷ into ❹, we prevent trivial solution, *i.e.*, $\mathcal{E}(I_t) \equiv \mathcal{E}(I_{t'})$, when directly optimizing $\mathcal{E}$ via ❹, and eventually get ❺. Specifically, similar to ❹, we have $\mathcal{E}(I_t) \approx A_{\Phi(t')}^{t\top} \mathcal{E}(\Phi(I_{t'}))$; then with ❷, we obtain $\mathcal{E}(I_t) \approx A_{\Phi(t')}^{t\top} \mathcal{E}(\Phi(I_{t'})) = A_{\Phi(t')}^{t\top} \Phi(\mathcal{E}(I_{t'}))$.

Following ❺, we get the feature of transformed $I_{t'}$: $\boldsymbol{X}_{t'}^{\prime} = \mathcal{E}(\Phi(I_{t'})) \in \mathbb{R}^{HW \times D}$, transformed feature of $I_{t'}$: $\boldsymbol{X}_{t'} = \Phi(\mathcal{E}(I_{t'})) \in \mathbb{R}^{HW \times D}$, and the original feature of I_t: $\boldsymbol{I}_t = \mathcal{E}(I_t) \in \mathbb{R}^{HW \times D}$. For k-*th* pixel (feature) of $\boldsymbol{X}_{t'}^{\prime}$, we first find the matching (*i.e.*,

the most similar) pixel o_k in $\boldsymbol{I}_t$ as:

$$o_k = \underset{o\in\{1,\cdots,HW\}}{\arg\max}\ a_{k,o}, \qquad a_{k,o} = \frac{\exp(\langle \boldsymbol{x}_{t'}'^{k\top} \boldsymbol{i}_t^o \rangle)}{\sum_l \exp(\langle \boldsymbol{x}_{t'}'^{k\top} \boldsymbol{i}_t^l \rangle)}, \tag{4.13}$$

where $\boldsymbol{i}_t^o \in \mathbb{R}^D$ refers to o-th pixel feature of $\boldsymbol{I}_t$, and $a_{k,o}$ corresponds to (k,o)-th element of the affinity $A^t_{\Phi(t')}$ between $\Phi(I_{t'})$ and I_t. Then, the dominant index ok serves as pseudo labels for our temporally-distant matching and our long-term dependency loss is computed as:

$$\mathcal{L}_{\text{Long}} = -\sum\nolimits_{\mathcal{I}} \sum\nolimits_k \log \frac{\exp(\langle \boldsymbol{x}_{t'}^{k\top} \boldsymbol{i}_t^{o_k} \rangle)}{\sum_l \exp(\langle \boldsymbol{x}_{t'}^{k\top} \boldsymbol{i}_t^l \rangle)}. \tag{4.14}$$

Eq. 4.14 addresses global semantic dependencies by contrasting affinity between aligned pixel feature pairs, *i.e.*, $\boldsymbol{x}_{t'}^k$ and $\boldsymbol{i}_t^{o_k}$, against non-corresponding ones, *i.e.*, $\boldsymbol{x}_{t'}^k$ and $\{\boldsymbol{i}_t^l\}_{l\neq o_k}$, under an equivariant representation learning scheme.

4.2.4 Implementation Details

Full Loss. The overall training loss is:

$$\begin{aligned} \mathcal{L} =& \mathcal{L}_{\text{Seg}} + \mathcal{L}_{\text{Corr}} \\ =& \mathcal{L}_{\text{Seg}} + \lambda_1 \mathcal{L}_{\text{Short}} + \lambda_2 \mathcal{L}_{\text{Long}}, \end{aligned} \tag{4.15}$$

where the coefficients are empirically set as: $\lambda_1 = 0.1$ and $\lambda_2 = 0.5$.

Network Configuration. For the *visual encoder* $\mathcal{E}$, we instantiate it as `ResNet-18` or `ResNet-50` in our experiments. For `ResNet-18`, the spatial strides of the second and last residual blocks are removed to yield an output stride of 8, as in [122, 314, 5]. For `ResNet-50`, we follow [326] to take features from `res4`, and drop its stride to preserve more spatial details. For the *frame-mask encoder* $\mathcal{V}$, it has a similar structure as $\mathcal{E}$, expect for the input and output dimensionality. On the top of $\mathcal{E}$ and $\mathcal{V}$, two 1×1 convolution layers are separately added to reduce the output dimensions of $\mathcal{E}$ and $\mathcal{V}$ to $D = 128$ and $D' = 512$, respectively. For the *mask decoder* $\mathcal{D}$, it consists of two Residual blocks that are connected with $\mathcal{E}$ through skip layers, and a 1×1 convolution layer to produce the final segmentation prediction.

Method	Backbone	Dataset(size)	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{J}_r\uparrow$	$\mathcal{F}_m\uparrow$	$\mathcal{F}_r\uparrow$
Colorization [307]	ResNet-18	Kinetics(- , 800 hours)	34.0	34.6	34.1	32.7	26.8
CorrFlow [152]	ResNet-18	OxUvA(- , 14 hours)	50.3	48.4	53.2	52.2	56.0
TimeCycle [326]	ResNet-50	VLOG(- , 344 hours)	48.7	46.4	50.0	50.0	48.0
UVC [172]	ResNet-18	C+Kinetics(30K, 800 hours)	57.8	56.3	65.0	59.2	64.1
MuG [204]	ResNet-18	OxUvA(- , 14 hours)	54.3	52.6	57.4	56.1	58.1
MAST [151]	ResNet-18	Youtube-VOS(- , 5.58 hours)	65.5	63.3	73.2	67.6	77.7
CRW [122]	ResNet-18	Kinetics(- , 800 hours)	68.3	65.5	78.6	71.0	82.9
ConCorr [314]	ResNet-18	C+TrackingNet(30K, 300 hours)	63.0	60.5	70.6	65.5	73.0
CLTC [128]	ResNet-18	Youtube-VOS(- , 5.58 hours)	70.3	67.9	78.2	72.6	83.7
JSTG [383]	ResNet-18	Kinetics(- , 800 hours)	68.7	65.8	77.7	71.6	84.3
VFS [349]	ResNet-18	Kinetics(- , 800 hours)	67.9	65.0	77.2	70.8	82.3
	ResNet-50		69.4	66.7	78.6	72.0	85.2
DINO [26]	ResNet-50	I(1.28M, -)	56.2	54.5	58.1	57.9	60.3
	ViT-B/8		71.4	67.9	81.6	74.9	85.4
DUL [5]	ResNet-18	Youtube-VOS(- , 5.58 hours)	69.3	67.1	81.2	71.6	84.9
SCR [275]	ResNet-18	Kinetics(- , 800 hours)	70.5	67.4	78.8	73.6	84.6
LIIR [168]	ResNet-18	Youtube-VOS(- , 5.58 hours)	72.1	69.7	81.4	74.5	85.9
Ours	ResNet-18	Youtube-VOS(- , 5.58 hours)	**74.5**	**71.6**	**82.9**	**77.4**	**86.9**
	ResNet-50		**75.6**	**73.3**	**83.6**	**77.8**	**87.3**
OSVOS [22]	VGG-16	I+D(1.28M, 10k)	60.3	56.6	63.8	63.9	73.8
STM [236]	ResNet-50	I+D+Youtube-VOS(1.28M, 164k)	81.8	79.2	88.7	84.3	91.8

- I: ImageNet [57]; C: COCO [187]; D: DAVIS_{17} [245].

Table 4.1 : **Quantitative segmentation results** (§4.3.1) on DAVIS_{17} [245] `val`. For dataset size, we report (#*raw* images, length of *raw* videos) for self-supervised methods and (#image-level annotations, #pixel-level annotations) for supervised methods.

Training. We follow [26] to pre-train the backbone network $\mathcal{E}$ on YouTube-VOS for 300 epochs, enabling reliable initial clustering. Then, we conduct the main training for a total of 400 epochs using Adam optimizer with batch size 16 and base learning rate 1e-4, on one Tesla A100 GPU. In the first 300 epochs, the whole network is trained with only the correspondence loss $\mathcal{L}_{\text{Corr}}$. The learning rate is scheduled following a "step" policy, decayed by multi-plying 0.5 every 100 epochs. In the last 100 epochs, the whole network is trained using the full loss $\mathcal{L}$, with fixed learning rate 1e-5. The first time-space clustering is made at epoch 300 for creating initial pseudo segmentation labels. Afterwards, the pseudo labels are updated by conducting re-clustering at every 10 epochs. During clustering, we abandon over-size clusters, *i.e.*, accounting for more than 40% of video pixels. These big clusters are typically scene background, like sky and grass; only the remaining pixel clusters/segments are used as pseudo labels. Random scaling, cropping, and flipping are used for data augmentation, and the training image size is set to 256×256. In each mini-batch, we sample 3 frames per video, and adopt the strategy in [44, 236] to learn mask decoding with two reference frames (*i.e.*, $N = 2$).

Testing. Once trained, our model is applied to test videos without any fine-tuning. Following [122, 5], for each query frame, we take the first frame (providing reliable object mask information), and, if applicable, its prior 20 frames (capturing diverse object patterns), as well as their masks, as reference for segmentation prediction. In addition, we repeatedly feed the prediction $\hat{Y}_q$ back to the mask decoder $\mathcal{D}$ for iterative refinement. We find this strategy brings better results while requiring no extra parameters, with only marginal sacrifice of inference speed (see Table 4.4e).

4.3 Experiments

Dataset. We evaluate our approach on two VOS datasets, *i.e.*,

DAVIS$_{17}$ [245] and YouTube-VOS [351]. They have 30 and 474 videos in `val` sets, re-

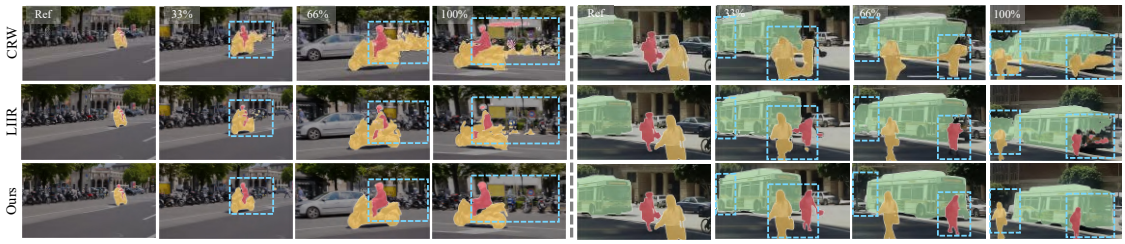


Figure 4.3 : **Visual comparison results** (§4.3.1) on two videos from DAVIS_{17} [245] `val` (left) and Youtube-VOS [351] `val` (right), respectively. CRW [122] and LIIR [168] suffer from error accumulation during mask tracking, due to the simple matching-based mask copy-paste strategy. However, our approach performs robust over time and yields more accurate segmentation results, by learning to embed target masks.

spectively. The videos are companied with pixel-wise annotations and cover various challenges like occlusion, complex background, and motion blur.

Evaluation Metric. Following the official evaluation protocols [245, 351], we adopt region similarity ($\mathcal{J}_m$), contour accuracy ($\mathcal{F}_m$) and their average ($\mathcal{J}\&\mathcal{F}_m$). For DAVIS_{17}, we additionally report the recall values ($\mathcal{J}_r$ and $\mathcal{F}_r$), at IoU threshold 0.5. For YouTube-VOS, scores are obtained by submitting the results to the official evaluation server and separately computed for *seen* and *unseen* categories.

4.3.1 Comparison with State-of-the-Art

Performance on DAVIS_{17}. Table 4.1 gives comparison results against 15 recent self-supervised VOS methods on DAVIS_{17} `val`. We also include two famous supervised alternatives [22, 236] for reference. As seen, using a relatively small amount of training data (*i.e.*, 5.58 hours of raw videos in YouTube-VOS `train`) and weak backbone architecture – `ResNet-18`, our approach outperforms all competitors across multiple evaluation metrics. When adopting `ResNet-50`, our approach yields far better performance, up to **75.6**% $\mathcal{J}\&\mathcal{F}_m$.

Method	Backbone	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{J}_r\uparrow$	$\mathcal{F}_m\uparrow$	$\mathcal{F}_r\uparrow$
MAST [151]	ResNet-18	54.3	50.7	58.9	57.8	64.5
CRW [122]	ResNet-18	55.9	52.3	-	59.6	-
DUL [5]	ResNet-18	57.0	53.5	60.4	60.5	67.6
SCR [275]	ResNet-18	59.9	55.9	-	64.0	-
LIIR [168]	ResNet-18	57.5	55.2	63.1	59.8	68.6
Ours	ResNet-18	**61.3**	**59.4**	**66.5**	**63.1**	**73.7**
	ResNet-50	**62.4**	**60.6**	**66.9**	**64.2**	**74.3**
RGMP [235]	ResNet-50	52.9	51.3	-	54.4	-
STM [236]	ResNet-50	72.2	69.3	-	75.2	-

Table 4.2 : **Quantitative results** on DAVIS_{17} [245] `test-dev` (§4.3.1).

Method	Backbone	$\mathcal{J}\&\mathcal{F}_m\uparrow$	Seen		Unseen	
			$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$
CorrFlow [152]	ResNet-18	46.6	50.6	46.6	43.8	45.6
MAST [151]	ResNet-18	64.2	63.9	64.9	60.3	67.7
CRW [122]	ResNet-18	68.7	67.4	69.1	65.1	73.2
CLTC [128]	ResNet-18	67.3	66.2	67.9	63.2	71.7
DUL [5]	ResNet-18	69.9	69.6	71.3	65.0	73.5
LIIR [168]	ResNet-18	69.3	67.9	69.7	65.7	73.8
Ours	ResNet-18	**71.6**	**71.0**	**74.2**	**66.0**	**75.3**
	ResNet-50	**72.4**	**71.7**	**74.6**	**67.0**	**76.2**
OSVOS [22]	VGG-16	58.8	59.8	60.5	54.2	60.7
STM [236]	ResNet-50	79.4	79.7	84.2	73.5	80.9

Table 4.3 : **Quantitative results** on YouTube-VOS [351] `val` (§4.3.1).

In particular, compared with `ResNet-18` based top-leading models, *i.e.*, LIIR [168], SCR [275], DUL [5], and CLTC [128], our approach earns **2.4**%, **4.0**%, **5.2**%, and **4.2**% $\mathcal{J}\&\mathcal{F}_m$ gains, respectively. Note that, CLTC adopts different net- work architectures and model weights for different datasets. Apart from this, VFS and JSTG make use of much more training data than ours (800 *vs*5.58 hours of videos). As for DINO, a recent state-of-the-art, contrastive image representation learning based method, our approach still outperforms it by **3.1**% and **4.2**% $\mathcal{J}\&\mathcal{F}_m$ based on `ResNet-18` and `ResNet-50`, respectively. This is particularly impressive, considering our backbone is *desperately inferior* to DINO (*i.e.*, `ResNet-18/-50` *vs* `ViT-B`) and the training data used by these two methods are completely not comparable in both quality and quantity (*i.e.*, 3.5K videos *vs*1.28M images). When using the same `ResNet-50` backbone, the performance gap is huge, *e.g.*, **19.4%** in $\mathcal{J}\&\mathcal{F}_m$. Table 4.2 reports our performance on DAVIS_{17} `test-dev`. We can clearly observe that, our approach, again, suppresses all the recent alternatives by a solid margin.

Performance on YouTube-VOS. We further conduct experiments on YouTube-

VOS `val`. As shown in Table 4.3, our approach, again, achieves remarkable performance, evidencing its efficacy and generalization ability across different VOS datasets. Specifically, when opting for `ResNet-18` backbone network architecture, our approach obtains **1.7**% absolute $\mathcal{J}\&\mathcal{F}_m$ improvement, over the current top leading method — DUL. Moreover, with a stronger backbone — `ResNet-50`, our approach further improves the $\mathcal{J}\&\mathcal{F}_m$ score to **72.4**%, setting a new state-of-the-art.

Visual Comparison Results. Fig. 4.3 depicts the visual comparison results of our approach and two competitors, MAST and DUL, on two challenging videos from DAVIS$_{17}$ `val` and YouTube-VOS `val`, respectively. We can find CRW and LIIR, as classic, correspondence-based methods, suffer from drifting errors during mask propagation; small prediction errors on past frames are hard to be corrected in later frames and further lead to worse results after processing more frames. This is due to their matching-based propagation strategy. In contrast, our approach generates more reasonable segments that better align object boundaries, and performs robust to small outlier predictions, hence reducing error accumulation over time. These results verify the efficacy of our model and support our insight that encoding mask information is crucial for self-supervised VOS. Further detailed quantitative analyses can be found in §4.3.2.

4.3.2 Diagnostic Experiments

To thoroughly examine our core hypotheses and model designs, we conduct a series of ablative studies on DAVIS$_{17}$ `val`. The reported baselines are built upon `ResNet-18` and trained by the default setting, unless otherwise specified.

Training Objective. Our model is jointly trained for mask-embedded segmentation $\mathcal{L}_{\text{Seg}}$ (*cf.* Eq. 4.15) and correspondence matching $\mathcal{L}_{\text{Corr}}$ $(= \mathcal{L}_{\text{Short}} + \mathcal{L}_{\text{Long}})$. Table 4.4a analyzes the influence of different training objectives. We can find that, using $\mathcal{L}_{\text{Short}}$ or $\mathcal{L}_{\text{Long}}$ individually only yields $\mathcal{J}\&\mathcal{F}_m$ scores of 57.4% and 67.2%,

Loss	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$
$\mathcal{L}_{\text{Short}}$	57.4	55.8	58.9
$\mathcal{L}_{\text{Long}}$	67.2	64.9	69.5
$\mathcal{L}_{\text{Short}}$ + $\mathcal{L}_{\text{Long}}$	68.8	66.7	70.9
$\mathcal{L}_{\text{Seg}}$	62.3	60.5	64.0
$\mathcal{L}_{\text{Seg}}$+$\mathcal{L}_{\text{Short}}$+$\mathcal{L}_{\text{Long}}$	**74.5**	**71.6**	**77.4**

(a) loss design

#Ref. Frame	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$
First	68.8	65.7	71.9
First + Last 1:15	73.2	70.4	76.0
First + Last 1:20	**74.5**	**71.6**	**77.4**
First + Last 1:25	73.5	70.9	76.1
First + Last 1:30	72.8	70.2	75.3

(b) number of reference frames

#Centroid	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$
$M = 2$	67.5	65.2	69.8
$M = 3$	71.6	69.0	74.2
$M = 5$	**74.5**	**71.6**	**77.4**
$M = 8$	72.5	69.6	75.4
$M = 10$	70.1	67.3	72.9

(c) number of cluster centers

Mask update	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$
No update	71.1	68.3	73.9
Per 20 epoch	72.8	69.9	75.7
Per 15 epoch	73.9	70.8	77.0
Per 10 epoch	**74.5**	**71.6**	**77.4**
Per 5 epoch	72.5	69.5	75.5
Every epoch	69.7	66.7	72.6

(d) pseudo mask update

Round	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$	FPS
0	69.7	67.3	72.1	1.86
1	72.6	69.8	75.4	1.84 (-1.1%)
2	73.9	71.1	76.7	1.80 (-3.2%)
3	**74.5**	**71.6**	**77.4**	1.77 (-4.8%)
4	74.3	71.2	**77.3**	1.73 (-7.0%)
5	74.0	71.0	77.0	1.69 (-9.2%)

(e) recurrent refinement

Strategy	Loss	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}_m\uparrow$	$\mathcal{F}_m\uparrow$	FPS
photometric	MAST [151]	65.5	63.3	67.6	1.13
reconstruction	MAST [151]+$\mathcal{L}_{\text{Seg}}$	**69.0** (+3.5)	**66.4**	**71.6**	1.01
cycle-consistency	CRW [122]	67.6	64.6	70.6	1.86
tracking	CRW [122]+$\mathcal{L}_{\text{Seg}}$	**71.8** (+4.2)	**68.3**	**75.3**	1.77
contrastive	$\mathcal{L}_{\text{Corr}}$ (ours)	68.8	66.7	70.9	1.86
matching	$\mathcal{L}_{\text{Corr}}$+$\mathcal{L}_{\text{Seg}}$	**74.5** (+5.7)	**71.6**	**77.4**	1.77

(f) correspondence learning schema

Table 4.4 : A set of ablative studies on DAVIS$_{17}$ [245] `val` (§4.3.2). The adopted settings are marked in red.

respectively. Their combination uplifts the performance to 68.8%, confirming their complementarity. However, the baseline is still weaker in comparison with current top-leading correspondence-based methods, *e.g.*, LIIR [168] with 72.1%. Moreover, when using $\mathcal{L}_{\text{Seg}}$ solely, the model only achieves 62.3%. This is because, without the regularization of the correspondence learning term, k-means suffers from random initialization of the representation and easily return trivial solutions, *e.g.*, fragile or massive clusters. When considering all the training goals together, performance boosts can be clearly observed, *e.g.*, **74.5%** in $\mathcal{J}\&\mathcal{F}_m$. Under such a scheme, unsupervised correspondence learning makes the features informative for meaningful clustering; then the produced high-quality pseudo masks allow the model to learn to make a better use of the object mask to guide segmentation.

Reference Frame. As usual [236, 151, 44], we leverage the first frame and several previous segmented frames as well as their corresponding masks, to support the

segmentation of the current frame. Table 4.4b reports the related experiments.

k-means Clustering. Next we probe the impact of the number of cluster centers, *i.e.*, M, in Table 4.4c. The best performance is obtained at $M=5$, roughly equal to the obvious objects number, *i.e.*, $3\sim4$ on average in each training video.

Pseudo Mask Update. During training, our approach alternates between clustering based pseudo mask generation and mask guided segmentation learning. In Table 4.4d, we study such training strategy. 'No update' means that, after the initial correspondence learning stage (first 300 training epochs; see §4.2.4), we create pseudo masks and use them throughout the whole joint correspondence and segmentation learning stage (last 100 epochs). This baseline achieves 71.1% $\mathcal{J}\&\mathcal{F}_m$. If we improve the frequency of pseudo mask update from once to twice every 20 epochs, the score is improved to **74.5**%. But further more frequently re-estimating the pseudo masks leads to inferior performance. We speculate that it is because, when learning with the noisy pseudo masks, it needs more epochs to optimize the network parameters, while updating the pseudo masks too frequently will easily suffer from the impact of sub-optimal features.

Recurrent Refinement. We feed our predicted masks to the segmentation decoder $\mathcal{D}$ for iterative refinement. Table 4.4e reports the related results. *Round 0* means we follow Eq. 4.6 to leverage $\boldsymbol{V}_q$ for mask decoding. In *Round 1*, the model follows Eq. 4.8 to warp and refine the coarse prediction $\bar{Y}_q$ and from *Round 2* onwards, we replace $\bar{Y}_q$ with the output $\hat{Y}_q$ from the prior round. As seen, after two rounds of refinement, $\mathcal{J}\&\mathcal{F}_m$ score is improved from 69.7% to **74.5**%, with only negligible delay in inference speed (*i.e.*, -4.8%).

Versatility. As our self-supervised mask embedding learning (*cf.* §4.2.2) is a general framework, it is interesting to test its efficacy with different correspondence learning regimes. In Table 4.4f, we apply our mask embedding learning method

to MAST [151] (reconstruction based), CRW [122] (cycle-consistency based), and our correspondence learning strategy $\mathcal{L}_{\text{Corr}}$ (*cf.* §4.2.3; contrastive matching based). Impressively, notable performance gains are achieved over different baselines, *e.g.*, **3.5%** on MAST, **4.2%** on CRW, and **5.7%** on our $\mathcal{L}_{\text{Corr}}$, in terms of $\mathcal{J}\&\mathcal{F}_m$. The last column of Table 4.4f gives inference speed, showing the additional computational budget brought by mask embedding is negligible.

4.4 Limitation

Currently we directly leverage the k-means algorithm to cluster pixels. The k-means clustering, though simple, is less efficient compared with some more advanced ones, such as [51, 64] which consider clustering from the perspective of optimal transport. We leave this as a part of our future work.

4.5 Conclusion

In this chapter, we devised a new framework that investigates both mask embedding and correspondence learning for mask propagation, in an annotation-free manner. Through space-time clustering, coherent video partitions are automatically generated for teaching the model to directly learn mask embedding and tracking. Meanwhile, self-supervised correspondence learning is naturally incorporated as extra regularization. In this way, our approach successfully bridges the gap between fully- and self-supervised VOS models in both performance and network architecture design.

Chapter 5

Relational Scene Understanding: Modeling Hierarchical Structures in Semantic Segmentation

Expanding on the previous exploration of temporal structures learned through visual correspondence in dynamic scenes, this chapter shifts the focus to the spatial dimension and the relational organization inherent in individual visual scenes.

5.1 Introduction

Semantic segmentation, which aims to identify semantic categories for pixel observations, is viewed as a vital step towards intelligent scene understanding [329]. The vast majority of modern segmentation models simply assume that all the target classes are disjoint and should be distinguished ex- clusively during pixel-wise prediction. This fails to capture the structured nature of the visual world [203]: complex scenes arise from the composition of simpler entities. Walking city, vehicles and pedestrian fill our view (Fig. 5.1). After focusing on the vehicles, we identify cars, buses, and trucks, which consist of more fine-grained parts like wheel and window. On the other hand, structured understanding of our world in terms of relations and hierarchies is a central ability in human cognition [277, 358]. We group chair and bed as furniture, while cat and dog as pet. We understand this world over multiple levels of abstraction, in order to maintain stable, coherent percepts in the face of complex visual inputs [134]. The ubiquity of hierarchical decomposition serves as a core motivation behind many structured machine learning models [55, 333], which have shown wide success in document classification [148, 215] and protein function prediction [297, 18].

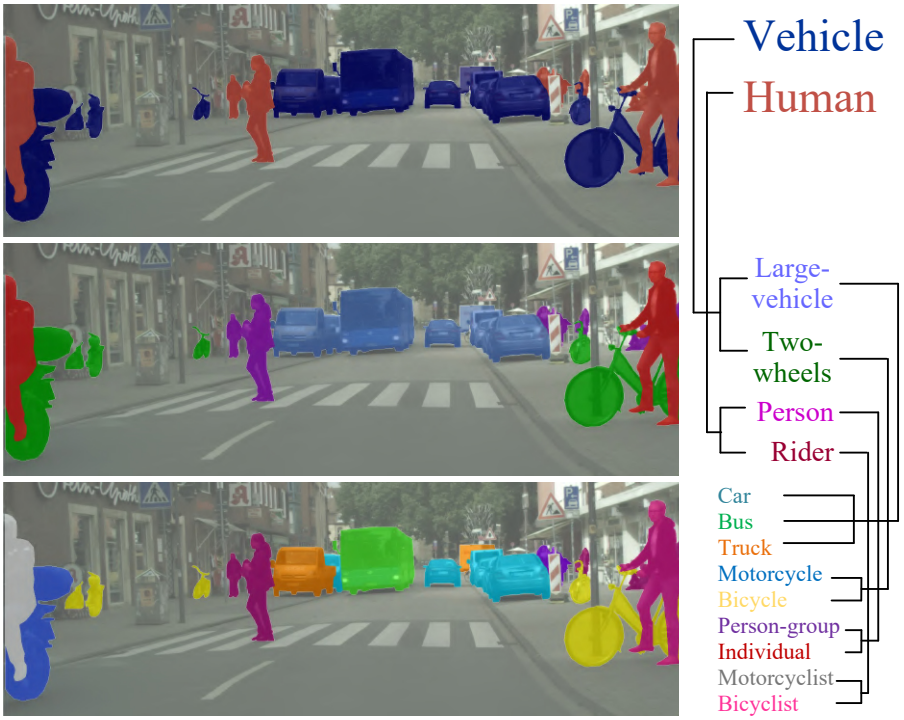


Figure 5.1 :**Hierarchical semantic segmentation** explains visual scenes with multi-level abstraction (*left*), by considering structured class relations (*right*). The class taxonomy is borrowed from [228].

In semantic segmentation literature, surprisingly little is understood about how to accommodate pixel recognition into semantic hierarchies. [326, 330, 343, 179, 219, 176, 328] are rare exceptions that exploit class hierarchies in segmentation networks. Nevertheless, they either focus specifically on the structured organization of human body parts [326, 330, 328], or introduce hierarchy-induced architectural changes to the segmentation network [343, 179, 219, 176], both hindering generality. More essentially, these methods are more aware of making efficient information propagation over the hierarchies (*e.g.*, graph message passing [330, 176, 397], multi-task learning [343]), without imposing tree-structured label dependencies/constraints into prediction and learning.

To mimic human hierarchical visual perception, we propose a novel approach for *hierarchical semantic segmentation* (HSS). In HSS, classes are not arranged in a

"flat" structure, but organized as a tree-shaped hierarchy. Thus each pixel observation is associated to a root-to-leaf path of the class hierarchy (*e.g.*, `human`→`rider`→ `bicyclist`), capturing general-to-specific relations between classes. Our algorithm, called HSSN, addresses two core issues in HSS, yet untouched before. **First**, instead of previous structured segmentation models focusing on sophisticated network design, HSSN directly formulates HSS as a pixel-wise multi-label classification task. This allows to easily adapt existing segmentation models to the HSS setting, densely linking the fields of classic hierarchy-agnostic segmentation and HSS together. **Second**, HSSN makes full use of the class hierarchy in HSS network learning. To make pixel predictions coherent with the class hierarchy, HSSN explores two *hierarchy constraints*, *i.e.*, **i)** a pixel sample belonging to a given class must also belong to all its ancestors in the hierarchy, **ii)** a pixel sample not belonging to a given class must also not belong to all its descendants, as optimization criterion. This leads to a *pixel-wise hierarchical segmentation learning* strategy, which enforces segmentation predictions to obey the hierarchy structure during training. HSSN further encodes the structured knowledge introduced by the class hierarchy into the pixel embedding space. This leads to a *pixel-wise hierarchical representation learning* strategy, which inspires tree-induced margin separation for embedding space reshaping. As the hierarchy characterizes the underlying relationships between classes, HSSN is able to enrich pixel embeddings by pulling semantically similar pixels (*e.g.*, `bicycle` and `motorcycle`) closer, while pushing semantically dissimilar pixels (*e.g.*, `pedestrian` and `lamppost`) farther away. This leads to more efficient learning by discovering and reusing common patterns [86], facilitating hierarchical segmentation eventually. This also allows our model to take different levels of mistakes into consideration. This is essential for some critical systems [15]. Take autonomous driving as an example: mistaking a `bicycle` for a `motorcycle` is less of a problem than confusing a `pedestrian` with a `lamppost`.

This work represents a solid step towards HSS. Our approach is elegant and principle; it is readily incorporated to arbitrary previous hierarchy-agnostic segmentation networks, with only marginal modification on the segmentation head. We train and test HSSN over four public benchmarks (*i.e.*, Mapillary Vistas 2.0 [228], Cityscapes [49], LIP [178], PASCAL-Person-Part [342]), with different class hierarchies for urban street scene parsing and human semantic parsing. Extensive experimental results with different segmentation network architectures (*i.e.*, DeepLabV3+ [32], OCRNet [365], MaskFormer [42]) and backbones (*i.e.*, ResNet-101 [106], HRNetV2-W48 [312], Swin-Small [194]) verify the generalization and effectiveness of HSSN.

5.2 Our Approach

Our goal is to accommodate standard semantic segmentation networks to the HSS problem and then exploit structured class relations in order to generate hierarchy-coherent representations and predictions, and improve performance. Given this goal, we develop HIERARCHICAL SEMANTIC SEGMENTATION NETWORKS (HSSN), a general framework for HSS network design (§5.2.1) and training (§5.2.2).

5.2.1 Hierarchical Semantic Segmentation Networks

Rather than typical segmentation methods treating semantic classes as disjoint labels, in the HSS setting, the underlying dependencies between classes are considered and formalized in a form of a tree-structured hierarchy, $\mathcal{T}=(\mathcal{V},\mathcal{E})$. Each node $v\in\mathcal{V}$ denotes a semantic class/concept, while each edge $(u,v)\in\mathcal{E}$ encodes the decomposition relationship between two classes, $u,v\in\mathcal{V}$, *i.e.*, parent node v is a more general, superclass of child node u, such as $(u,v)=(\texttt{bicycle},\texttt{vehicle})$. We assume $(v,v)\in\mathcal{E}$, thus every class is both a subclass and superclass of itself. The root node of $\mathcal{T}$, *i.e.*, v^r, denotes the most general class. The leaf nodes, *i.e.*, $\mathcal{V}_\chi$, refer to the most fine-grained classes, such as $\mathcal{V}_\chi=\{\texttt{tree},\texttt{bicyclist},\cdots\}$ in urban street scene

parsing, and $\mathcal{V}_\chi = \{\texttt{head}, \texttt{leg}, \cdots\}$ in human parsing.

For a typical hierarchy-agnostic segmentation network, an encoder f_{ENC} is first adopted to map an image I into a dense feature tensor $\boldsymbol{I} = f_{\text{ENC}}(I) \in \mathbb{R}^{H \times W \times C}$, where $\boldsymbol{i} \in \boldsymbol{I}$ is the embedding of pixel $i \in I$. Then a segmentation head f_{SEG} is used to get a score map $\boldsymbol{Y} = \texttt{softmax}(f_{\text{SEG}}(\boldsymbol{I})) \in [0,1]^{H \times W \times |\mathcal{V}_\chi|}$ w.r.t.**the leaf node set** $\mathcal{V}_\chi$. Given the *score vector* $\boldsymbol{y} = [y_{v_\chi}]_{v_\chi \in \mathcal{V}_\chi} \in [0,1]^{|\mathcal{V}_\chi|}$ and *groundtruth leaf label* $\hat{v}_\chi \in \mathcal{V}_\chi$ for pixel i, the categorical cross-entropy loss is optimized:

$$\mathcal{L}^{\text{CCE}}(\boldsymbol{y}) = -\log(y_{\hat{v}_\chi}). \tag{5.1}$$

During inference, pixel i is associated to *a single leaf node*: $v^*_\chi = \arg\max_{v_\chi}(y_{v_\chi})$.

To accommodate classic segmentation networks to the HSS setting with minimum change, our HSSN first formulates HSS as a pixel-wise multi-label classification task, *i.e.*, map pixels with their corresponding classes in the hierarchy as a whole. Specifically, *only* the segmentation head f_{SEG} is modified to predict an *augmented* score map $\boldsymbol{S} = \texttt{sigmoid}(f_{\text{SEG}}(\boldsymbol{I})) \in [0,1]^{H \times W \times |\mathcal{V}|}$ w.r.t.**the entire class hierarchy** $\mathcal{V}$. Given the score vector $\boldsymbol{s} = [s_v]_{v \in \mathcal{V}} \in [0,1]^{|\mathcal{V}|}$ and *groundtruth binary label set* $\hat{\boldsymbol{l}} = [\hat{l}_v]_{v \in \mathcal{V}} \in \{0,1\}^{|\mathcal{V}|}$ for pixel i, the binary cross-entropy loss is optimized:

$$\mathcal{L}^{\text{BCE}}(\boldsymbol{s}) = \textstyle\sum_{v \in \mathcal{V}} -\hat{l}_v \log(s_v) - (1-\hat{l}_v)\log(1-s_v). \tag{5.2}$$

During inference, each pixel i is associated with the top-scoring root-to-leaf path in the class hierarchy $\mathcal{T}$:

$$\{v^*_1, \cdots, v^*_{|\mathcal{P}|}\} = \arg\max_{\mathcal{P} \subseteq \mathcal{T}} \textstyle\sum_{v_p \in \mathcal{P}} s_{v_p}, \tag{5.3}$$

where $\mathcal{P} = \{v_1, \cdots, v_{|\mathcal{P}|}\} \subseteq \mathcal{T}$ denotes a feasible root-to-leaf path of $\mathcal{T}$, *i.e.*, $v_1 \in \mathcal{V}_\chi$, $v_{|\mathcal{P}|} = v^r$, and $\forall v_p, v_{p+1} \in \mathcal{P} \Rightarrow (v_p, v_{p+1}) \in \mathcal{E}$. Although Eq. 5.3 ensures the coherence between pixel-wise prediction and the class hierarchy during the inference stage, there is no any class relation information used for segmentation network training, as the binary cross-entropy loss in Eq. 5.2 is computed over each class independently.

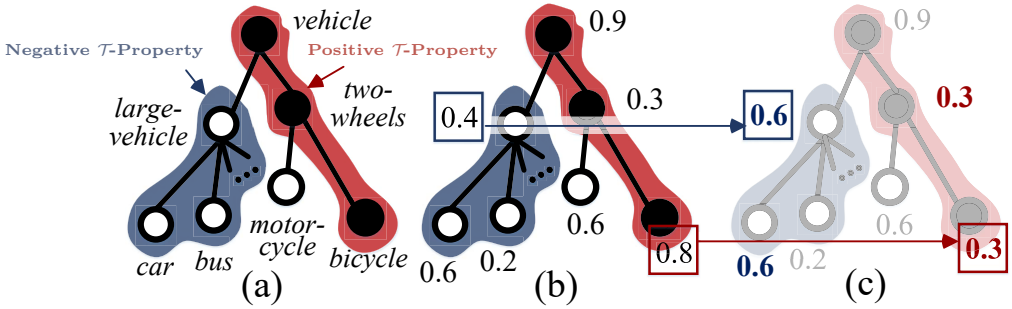


Figure 5.2 : **Hierarchy constraints** used in our pixel-wise hierarchical segmentation learning (§5.2.2). (a) In the class hierarchy, the filled circles represent the positive classes, while empty circles indicate the negative classes. The positive and negative $\mathcal{T}$-properties are highlighted in the red and blue regions, respectively. (b) The original score vector $\boldsymbol{s}$ predicted for the class hierarchy. The predictions which violate the positive and negative $\mathcal{T}$-constraints are highlighted in the red and blue rectangles, respectively. (c) The updated score vector $\boldsymbol{p}$, which satisfies the $\mathcal{T}$-constraints. With $\mathcal{L}^{\text{TM}}$, the penalties for the wrong predictions, *i.e.*, '**0.6**' and '**0.3**', are increased twice, compared with applying $\mathcal{L}^{\text{BCE}}$.

To alleviate this issue, we propose a hierarchy-aware segmentation learning scheme (§5.2.2), which incorporates the semantic structures into the training of HSSN.

5.2.2 Hierarchy-Aware Segmentation Learning

Our hierarchy-aware segmentation learning scheme includes two major components: i) a *pixel-wise hierarchical segmentation learning* strategy (§5.2.2) which supervises the segmentation prediction $\boldsymbol{S}$ in a hierarchy-coherent manner, and ii) a *pixel-wise hierarchical representation learning* strategy (§5.2.2) that makes hierarchy-induced margin separation for reshaping the pixel embedding space f_{ENC}.

Pixel-Wise Hierarchical Segmentation Learning

For each pixel, the assigned labels are hierarchically consistent if they satisfy the following two properties (Fig. 5.2):

Definition 5.2.2.1 (Positive $\mathcal{T}$-Property)**.** *For each pixel, if a class is labeled positive, all its ancestor nodes* (i.e., *superclasses) in* $\mathcal{T}$ *should be labeled positive.*

Definition 5.2.2.2 (Negative $\mathcal{T}$-Property)**.** *For each pixel, if a class is labeled negative, all its child nodes (*i.e.*, subclasses) in $\mathcal{T}$ should be labeled negative.*

The first property, also known as $\mathcal{T}$-property [18], was explored in some hierarchical classification work [304, 333, 90], while the second property is ignored. Actually, these two properties are complementary and crucial for consistent hierarchical prediction. Specifically, to incorporate these two label consistency properties into the supervision of HSSN, we further derive the following two hierarchy constraints w.r.t.per-pixel prediction, *i.e.*, $\boldsymbol{s}=[s_v]_{v\in\mathcal{V}}\in[0,1]^{|\mathcal{V}|}$:

Definition 5.2.2.3 (Positive $\mathcal{T}$-Constraint)**.** *For each pixel, if v class is labeled positive, and u is an ancestor node (*i.e.*, superclass) of v, it should hold that $s_v \leq s_u$.*

Definition 5.2.2.4 (Negative $\mathcal{T}$-Constraint)**.** *For each pixel, if v class is labeled negative, and u is a child node (*i.e.*, subclass) of v, it should hold that $1-s_v \leq 1-s_u$.*

With the positive $\mathcal{T}$-constraint, the positive $\mathcal{T}$-property can be always guaranteed. Similar conclusion is also hold for the negative $\mathcal{T}$-constraint (*cf.* Def. 5.2.2.4) and negative $\mathcal{T}$-property (*cf.* Def. 5.2.2.2).

Tree-Min Loss. To ensure the satisfaction of the two hierarchy constraints, we estimate a hierarchy-coherent score map $\boldsymbol{P}\in[0,1]^{H\times W\times|\mathcal{V}|}$ from $\boldsymbol{S}$. For pixel i, the updated score vector $\boldsymbol{p}=[p_v]_{v\in\mathcal{V}}\in[0,1]^{|\mathcal{V}|}$ in $\boldsymbol{P}$ is given as:

$$\begin{cases} p_v = \min\limits_{u\in\mathcal{A}_v}(s_u) & \text{if } \hat{l}_v=1, \\ 1-p_v = \min\limits_{u\in\mathcal{C}_v}(1-s_u) = 1-\max\limits_{u\in\mathcal{C}_v}(s_u) & \text{if } \hat{l}_v=0, \end{cases} \tag{5.4}$$

where $\mathcal{A}_v$ and $\mathcal{C}_v$ denote the superclass and subclass sets of v in $\mathcal{T}$ respectively, and $\boldsymbol{s}=[s_v]_{v\in\mathcal{V}}\in\boldsymbol{S}$ refers to the original score vector of pixel i. Note that, according to our definition $(v,v)\in\mathcal{E}$ (*cf.* §5.2.1), we have $v\in\mathcal{A}_v$ and $v\in\mathcal{C}_v$. With Eq. 5.4,

Figure 5.3 : Effect of $\mathcal{L}^{\text{BCE}}$ in Eq. 5.2 (top) vs $\mathcal{L}^{\text{FTM}}$ in Eq. 5.6 (bottom).

the pixel-wise prediction $\boldsymbol{p}$ is guaranteed to always satisfy the hierarchy constraints (*cf.* Defs. 5.2.2.3 and 5.2.2.4).

We thus build a hierarchical segmentation training objective, *i.e.*, tree-min loss, to replace $\mathcal{L}^{\text{BCE}}(\boldsymbol{s})$ in Eq. 5.2:

$$\begin{aligned}\mathcal{L}^{\text{TM}}(\boldsymbol{p}) &= \sum_{v\in\mathcal{V}} -\hat{l}_v \log(p_v) - (1-\hat{l}_v)\log(1-p_v), \\ &= \sum_{v\in\mathcal{V}} -\hat{l}_v \log(\min_{u\in\mathcal{A}_v}(s_u)) - \\ &\qquad (1-\hat{l}_v)\log(1-\max_{u\in\mathcal{C}_v}(s_u)).\end{aligned} \tag{5.5}$$

Compared with $\mathcal{L}^{\text{BCE}}(\boldsymbol{s})$, $\mathcal{L}^{\text{TM}}(\boldsymbol{p})$ is more favored as the structured score distribution $\boldsymbol{p}$ is constructed by strictly following the hierarchy constraints (*cf.* Eq. 5.4), and hence the violation of the hierarchy properties (*i.e.*, any undesired prediction of $\boldsymbol{p}$) can be explicitly penalized (see Fig. 5.2(c)).

Focal Tree-Min Loss. Inspired by the focal loss [186], we add a modulating factor to the tree-min loss (*cf.* Eq. 5.5), so as to reduce the relative loss for well-classified pixel samples and focus on those difficult ones:

$$\begin{aligned}\mathcal{L}^{\text{FTM}}(\boldsymbol{p}) &= \sum_{v\in\mathcal{V}} -\hat{l}_v(1-p_v)^{\gamma}\log(p_v) - (1-\hat{l}_v)(p_v)^{\gamma}\log(1-p_v), \\ &= \sum_{v\in\mathcal{V}} -\hat{l}_v(1-\min_{u\in\mathcal{A}_v}(s_u))^{\gamma}\log(\min_{u\in\mathcal{A}_v}(s_u)) - \\ &\qquad (1-\hat{l}_v)(\max_{u\in\mathcal{C}_v}(s_u))^{\gamma}\log(1-\max_{u\in\mathcal{C}_v}(s_u)),\end{aligned} \tag{5.6}$$

where $\gamma \geq 0$ is a tunable focusing parameter controling the rate at which easy classes

are down-weighted. When $\gamma = 0$, $\mathcal{L}^{\text{FTM}}(\boldsymbol{p})$ is equivalent to $\mathcal{L}^{\text{TM}}(\boldsymbol{p})$. Fig. 5.3 shows representative visual effects of $\mathcal{L}^{\text{FTM}}$ against $\mathcal{L}^{\text{BCE}}$. We see that $\mathcal{L}^{\text{FTM}}$ yields more precise and coherent results. In §5.3.4, we provide quantitative comparison results for $\mathcal{L}^{\text{BCE}}(\boldsymbol{s})$ (*cf.* Eq. 5.2), $\mathcal{L}^{\text{TM}}(\boldsymbol{p})$ (*cf.* Eq. 5.5), and $\mathcal{L}^{\text{FTM}}(\boldsymbol{p})$ (*cf.* Eq. 5.6).

Pixel-Wise Hierarchical Representation Learning

Through mapping pixels with their corresponding semantic classes in the hierarchy $\mathcal{T}$ as a whole (*cf.* §5.2.1), we exploit intrinsic properties of $\mathcal{T}$ (*cf.* Defs. 5.2.2.1-5.2.2.2) as constraints (*cf.* Defs. 5.2.2.3-5.2.2.4) to encourage hierarchy-coherent segmentation prediction $\boldsymbol{S}$ (*cf.* Eqs. 5.5-5.6). As the class hierarchy provides rich semantic relations among categories over different levels of concept abstraction, next we will exploit such structured knowledge to reshape the pixel embedding space f_{ENC}, so as to generate more efficient pixel representations and improve final segmentation performance.

With this purpose, we put forward a margin based pixel-wise hierarchical representation learning strategy, where the learned pixel embeddings are well separated with structured margins imposed by the class hierarchy $\mathcal{T}$. Specifically, for any pair of labels $u, v \in \mathcal{V}$, let $\psi(u, v)$ denote their *distance* in the tree $\mathcal{T}$. That is, $\psi(u, v)$ is defined as the length (in edges) of the shortest path between u and v in $\mathcal{T}$. The distance function $\psi(\cdot, \cdot)$ is in fact a semantic similarity metric defined over $\mathcal{T}$ [55]; it is a non-negative and symmetric function, $\psi(v, v) = 0$, $\psi(u, v) = \psi(v, u)$, and the triangle inequality always holds with equality.

In HSSN, the structured margin constraints are defined by the tree distance $\psi(\cdot, \cdot)$, leading to a **tree-triplet loss**. This loss is optimized on a set of pixel triplets $\{i, i^+, i^-\}$, where i, i^+, i^- are anchor, positive and negative pixel samples, respectively. $\{i, i^+, i^-\}$ are sampled from the whole training batch, such that $\psi(\hat{v}_\chi, \hat{v}_\chi^+) < \psi(\hat{v}_\chi, \hat{v}_\chi^-)$, where $\hat{v}_\chi$, $\hat{v}_\chi^+, \hat{v}_\chi^- \in \mathcal{V}_\chi$ are the groundtruth leaf labels of i, i^+, and i^-,

respectively. As such, in our tree-triplet loss, the positive samples are more semantically similar to the anchor pixels (*i.e.*, closer in $\mathcal{T}$), compared with the negative pixels. Note that this is different from the classic, hierarchy-agnostic triplet loss [267], where the anchor and positive samples are from the same class, while the anchor and negative samples are from different classes, *i.e.*, $\hat{v}_\chi = \hat{v}_\chi^+$, and $\hat{v}_\chi \neq \hat{v}_\chi^-$. With a valid training triplet $\{i, i^+, i^-\}$, our loss is given as:

$$\mathcal{L}^{\text{TT}}(\boldsymbol{i}, \boldsymbol{i}^+, \boldsymbol{i}^-) = \max\{\langle \boldsymbol{i}, \boldsymbol{i}^+ \rangle - \langle \boldsymbol{i}, \boldsymbol{i}^- \rangle + m, 0\}, \tag{5.7}$$

where $\boldsymbol{i}, \boldsymbol{i}^+, \boldsymbol{i}^- \in \mathbb{R}^C$ are the embeddings of i, i^+, and i^-, respectively, obtained from the encoder f_{ENC}, $\langle \cdot, \cdot \rangle$ is a distance function to measure the similarity of two inputs; we use the cosine distance, *i.e.*, $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \frac{1}{2}(1 - \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\| \|\boldsymbol{y}\|}) \in [0, 1]$. The margin m forces the gap of $\langle \boldsymbol{i}, \boldsymbol{i}^- \rangle$ and $\langle \boldsymbol{i}, \boldsymbol{i}^+ \rangle$ larger than m. When the gap is larger than m, the loss value would be zero. The separation margin m is determined as:

$$\begin{aligned} m &= m_\varepsilon + 0.5 m_\tau \\ m_\tau &= (\psi(\hat{v}_\chi, \hat{v}_\chi^-) - \psi(\hat{v}_\chi, \hat{v}_\chi^+))/2D, \end{aligned} \tag{5.8}$$

where m_ε=0.1 is set as a *constant* for the tolerance of the intra-class variance, *i.e.*, maximum intra-class distance, $m_\tau \in [0, 1]$ is a *dynamic* violate margin, which is computed according to the semantic relationships among i, i^+, and i^- over the class hierarchy $\mathcal{T}$, and D refers to the height of $\mathcal{T}$.

Eq. 5.7 encourages f_{ENC} as a hierarchically-structured embedding space (Fig. 5.4): pixels with similar semantics (*i.e.*, nearby in $\mathcal{T}$) are pushed closer than those with dissimilar semantics (*i.e.*, faraway in $\mathcal{T}$), guided by the hierarchy-induced margin m. Related experiments are given in §5.3.4.

5.2.3 Implementation Detail

Network Architecture. HSSN is a general HSS framework; it is readily applied to any hierarchy-agnostic segmentation models. **i)** The *segmentation encoder* f_{ENC}

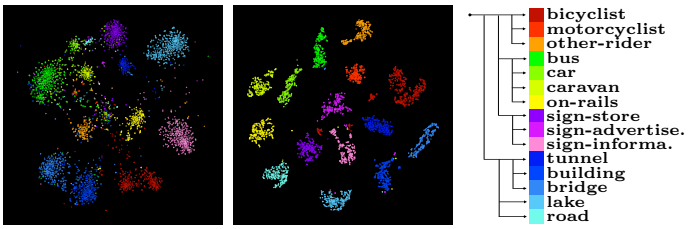


Figure 5.4 : **Visualization of the hierarchical embedding space** f_{ENC} learned on Mapillary Vistas 2.0 [228] (§5.2.2). The different colors correspond to different categories. It can be seen that, with $\mathcal{L}^{\text{TT}}$, f_{ENC} (middle) nicely embraces the hierarchical semantic structures (right), in comparison with the one without $\mathcal{L}^{\text{TT}}$ (left).

(§5.2.1) maps each input image I into a dense feature $\boldsymbol{I} \in \mathbb{R}^{H \times W \times C}$, and can be implemented as any backbone networks. In §5.3, we experiment with two CNN-based (*i.e.*, ResNet-101 [106] and HRNetV2-W48 [312]) and a Transformer-based (*i.e.*, Swin-Transformer [194]) backbones. **ii)** The *segmentation head* f_{SEG} (§5.2.1) projects $\boldsymbol{I}$ into a structured score map $\boldsymbol{S} \in \mathbb{R}^{H \times W \times |\mathcal{V}|}$ for all the classes in $\mathcal{V}$. Segmentation heads used in recent segmentation models (*i.e.*, DeepLabV3+ [32], OCRNet [365], MaskFormer [42]) are used and modified.

Training Objective. HSSN is end-to-end trained by minimizing the combinatorial loss of our *focal tree-min* loss ($\mathcal{L}^{\text{FTM}}$ in Eq. 5.6) and *tree-triplet* loss ($\mathcal{L}^{\text{TT}}$ in Eq. 5.7): $\mathcal{L}^{\text{FTM}} + \beta \mathcal{L}^{\text{TT}}$, where the coefficient $\beta \in [0, 0.5]$ is scheduled following a cosine annealing policy [197]. The focusing parameter γ in $\mathcal{L}^{\text{FTM}}$ is set as 2. Furthermore, following the common practice in metric learning, a *projection function* f_{PROJ} is used in $\mathcal{L}^{\text{TT}}$. It maps each pixel embedding $\boldsymbol{i}$ into a 256-d vector. f_{PROJ} consists of two 1×1 convolutional layers and one ReLU between them, and is discarded after training, causing no extra computational cost in deployment.

Inference. For each pixel, the label assignment follows Eq. 5.3.

5.3 Experiment

5.3.1 Experimental Setup

Datasets. We conduct experiments on two popular urban street scene parsing datasets [228, 49] and two human body parsing datasets [342, 178]. The corresponding class hierarchies are either the officially provided ones [228, 49] or generated by following the conventions [342, 178].

- **Mapillary Vistas 2.0** [228] is an urban egocentric street-view dataset with high-resolution images. It contains 18,000, 2,000 and 5,000 images for `train`, `val` and `test`, respectively. It provides annotations for 144 semantic concepts, which are organized in a three-level hierarchy, covering 4/16/124 concepts, respectively.
- **Cityscapes** [49] contains 5,000 elaborately annotated urban scene images, which are split into 2,975/500/1,524 for `train`/`val`/`test`. It is associated with 19 fine-grained concepts, which are grouped into 6 super-classes.
- **PASCAL-Person-Part** [342] has 1,716 and 1,817 images for `train` and `test`, with precise annotations for 6 human parts. Following [326, 330], we group 20 fine-grained parts (*e.g.*, `head`, `left-arm`) into two superclasses `upper-body` and `lower-body`, which are further combined ito `full-body`.
- **LIP** [178] includes 50,462 single-person images gathered from real-world scenarios, with 30,462/10,000/10,000 for `train`/`val`/`test` splits. The hierarchy is similar to the one in PASCAL-Person-Part, but the leaf layer has 19 fine-grained semantic parts.

Training. For fair comparison, we follow [376, 326, 379, 32, 171] to set the training hyper-parameters. Specifically, for CNN-based models, we use SGD as the optimizer with base learning rate 1e-2, momentum 0.9 and weight decay 1e-4. For Transformer-based models, we use AdamW [198] with base learning rate 6e-5 and weight decay 0.01. The learning rate is scheduled by the polynomial annealing policy [30]. All

Method		Backbone	mIoU^3↑	mIoU^2↑	mIoU^1↑
DeepLabV3+ [32]	[ECCV18]	ResNet-101	81.86	68.17	37.43
Seamless [249]	[CVPR19]	ResNet-101	-	-	38.17
OCRNet [365]	[ECCV20]	HRNet-W48	83.19	69.32	38.26
HMSANet [330]	[ArXiv19]	HRNet-W48	84.63	70.71	39.53
MaskFormer [42]	[NeurIPS21]	ResNet-101	84.56	70.82	39.60
MaskFormer [42]	[NeurIPS21]	Swin-Small	87.93	73.88	42.16
DeepLabV3+		ResNet-101	85.27	71.40	40.16
Hssn OCRNet		HRNet-W48	86.46	72.34	41.13
MaskFormer		Swin-Small	**90.02**	**75.81**	**43.97**

Table 5.1 : **Hierarchical semantic segmentation results** (§5.3.2) on the `val` set of Mapillary Vistas 2.0 [228].

backbones are initialized using the weights pre-trained on ImageNet-1K [57], while the remaining layers are randomly initialized. During training, we use standard data augmentation techniques, *i.e.*, horizontal flipping and random scaling with a ratio between 0.5 and 2.0. We train 240K and 80K iterations for Mapillary Vistas 2.0 and Cityscapes, with batch size 8 and crop size 512×1024. For PASCAL-Person-Part and LIP, we use batch size 16 and crop size 480×480, and train models for 80K and 160K iterations, respectively.

Testing. The inference follows Eq. 5.3. As in [117, 42, 365, 130, 330, 326], we report the segmentation scores at multiple scales ($\{0.5, 0.75, 1.0, 1.25, 1.5, 1.75\}$) with horizontal flipping.

Evaluation Metric. The mean intersection-over-union (mIoU) is adopted for evaluation. Particularly, we report the average score, *i.e.*, mIoU^l, for classes in each hierarchy level l independently. For reference, we also report the scores of each level for hierarchy-agnostic methods. The results of each non-leaf layer are obtained by merging the segmentation predictions of its subclasses together.

5.3.2 Quantitative Results

Mapillary Vistas 2.0 [228]. Table 5.1 presents comparisons of our HSSN against several top-leading semantic segmentation models on Mapillary Vistas 2.0 `val`. With the standard ResNet-101 as the backbone, HSSN outperforms the famous DeepLabV3+ [32] by solid margins across all three levels (**2.69%/3.21%/3.40%**). Consistent gains are also observed for a more recent segmentation model (*i.e.*, MaskFormer [42]), which relies on a heavy Transformer-based decoder. In addition, our HSSN further improves the performance when using more advanced CNN-based (*i.e.*, HRNetV2-W48) or Transformer-based (*i.e.*, Swin-Small) backbones. Concretely, it outperforms OCRNet [365] by **2.87%/3.02%/3.27%** and MaskFormer [42] by **1.81%/ 1.93%/2.09%** across the three levels. HSSN, with Swin-Small as the backbone, establishes a new state-of-the-art. These results clearly demonstrate the efficacy of our hierarchical semantic segmentation framework.

Cityscapes [49]. Table 5.2 compares our HSSN with several competitive models on Cityscapes `val`. Despite that the dataset has relatively simple semantic hierarchy and has been comprehensively benchmarked, our model still leads to appealing improvements. In particular, HSSN outperforms the top-leading MaskFormer [42] by **1.17%/1.43%** in terms of mIoU^1 and mIoU^2 when using Swin-Small as the backbone. Similar gains are obtained when applying CNN-based backbones (*i.e.*, ResNet-101 and HRNet-W48).

PASCAL-Person-Part [342]. Table 5.3 lists the detailed results on PASCAL-Person-Part `test`. Note that all the models use ResNet-101 as the backbone. As seen, our HSSN achieves the best performance for all human parts and hierarchical levels. Remarkably, HSSN outperforms all existing hierarchical human parsers (*i.e.*, HHP [330], SemaTree [130] and CNIF [326]) by significant margins. Results on this dataset are particularly impressive since it includes a very small number (*i.e.*, 1,713)

Method		Backbone	mIoU^2↑	mIoU^1↑
DeepLabV2 [29]	[CVPR17]	ResNet-101	-	70.22
PSANet [380]	[ECCV18]	ResNet-101	-	80.96
PAN [161]	[ArXiv18]	ResNet-101	-	81.12
DeepLabV3+ [32]	[ECCV18]	ResNet-101	92.16	82.08
DANet [76]	[CVPR19]	ResNet-101	-	81.52
Acfnet [367]	[ICCV19]	ResNet-101	-	81.60
CCNet [117]	[ICCV19]	ResNet-101	-	81.08
HANet [47]	[CVPR20]	ResNet-101	-	81.82
HRNet [312]	[TPAMI20]	HRNet-W48	92.12	81.96
OCRNet [365]	[ECCV20]	HRNet-W48	92.57	82.33
MaskFormer [42]	[NeurIPS21]	Swin-Small	92.96	82.57
DeepLabV3+		ResNet-101	93.31	83.02
Hssn OCRNet		HRNet-W48	93.92	83.37
MaskFormer		Swin-Small	**94.39**	**83.74**

Table 5.2 : **Hierarchical semantic segmentation results** (§5.3.2) on the `val` set of Cityscapes [49].

Method	Head	Torso	U-Arm	L-Arm	U-Leg	L-Leg	U-Body	L-Body	F-Body	B.G.	mIoU^3↑	mIoU^2↑	mIoU^1↑
DeepLabV3+ [32]	87.02	72.02	60.37	57.36	53.54	48.52	90.07	65.88	93.02	96.07	94.55	84.01	67.84
SPGNet [40]	87.67	71.41	61.69	60.35	52.62	48.80	-	-	-	95.98	-	-	68.36
PGN [95]	90.89	75.12	55.83	64.61	55.42	41.57	-	-	-	95.33	-	-	68.40
CNIF [326]	88.02	72.91	64.31	63.52	55.61	54.96	91.82	66.56	94.33	96.02	95.18	84.80	70.76
SemaTree [130]	89.15	74.76	63.90	63.95	57.53	54.62	92.36	67.13	95.11	96.84	95.98	85.44	71.59
HHP [330]	89.73	75.22	66.87	66.21	58.69	58.17	93.44	68.02	96.77	96.94	96.86	86.13	73.12
BGNet [376]	90.18	77.44	68.93	67.15	60.79	59.27	-	-	-	97.12	-	-	74.42
PCNet [375]	90.04	76.89	69.11	68.40	60.78	60.14	-	-	-	96.78	-	-	74.59
Hssn	**90.19**	**78.72**	**70.67**	**69.71**	**61.15**	**60.44**	**95.86**	**71.56**	**98.20**	**97.18**	**97.69**	**88.20**	**75.44**

Table 5.3 : **Hierarchical human parsing results** (§5.3.2) on PASCAL-Person-Part [342] `test`. All models use ResNet-101 as the backbone.

Method		Backbone	mIoU^3↑	mIoU^2↑	mIoU^1↑
SegNet [11]	[TPAMI17]	ResNet-101	-	-	18.17
FCN-8s [196]	[CVPR15]	ResNet-101	-	-	28.29
DeepLabV2 [29]	[CVPR17]	ResNet-101	-	-	41.64
Attention [31]	[CVPR16]	ResNet-101	-	-	42.92
MMAN [207]	[ECCV18]	ResNet-101	-	-	46.93
DeepLabV3+[32]	[ECCV18]	ResNet-101	88.13	83.97	52.28
CE2P [263]	[AAAI19]	ResNet-101	-	-	53.10
BraidNet [191]	[ACMMM19]	ResNet-101	-	-	54.42
SemaTree [130]	[ECCV20]	ResNet-101	90.78	87.12	54.73
BGNet [376]	[ECCV20]	ResNet-101	-	-	56.82
PCNet [375]	[CVPR20]	ResNet-101	-	-	57.03
CNIF [326]	[ICCV19]	ResNet-101	95.92	91.83	57.74
HRNet [312]	[TPAMI20]	HRNet-W48	95.53	91.21	57.23
OCRNet [365]	[ECCV20]	HRNet-W48	96.78	92.56	58.47
HHP [330]	[CVPR20]	ResNet-101	97.41	93.43	59.25
Hssn DeepLabV3+		ResNet-101	**98.86**	**94.75**	**60.37**

Table 5.4 : **Hierarchical human parsing results** (§5.3.2) on LIP `val`.

of training samples.

LIP [178]. In Table 5.4, we compare HSSN with state-of-the-art human semantic parsing models on LIP `val`. As observed, our model provides a considerable performance gain against the leading hierarchy-aware human parser (*i.e.*, HHP [330]) across all three levels (**1.12%/1.32%/1.45%**). These results support our motivation of exploiting structured label constraints and structured representation learning rather than only focusing on structured feature fusion.

5.3.3 Qualitative Results

Fig. 5.5 and Fig. 5.6 depict representative visual results on four datasets. As seen, HSSN yields more precise segmentation results in comparison with some top-

Figure 5.5 : **Visual results** (§5.3.3) on Mapillary Vistas 2.0 [228] `val` (left) and Cityscapes [49] `val` (right). Top: MaskFormer, Bottom: HSSN.

Figure 5.6 : **Visual results** (§5.3.3) on LIP [178] `val` (left) and PASCAL-Person-Part [342] `test` (right). Top: DeepLabV3+, Bottom: HSSN.

performing methods (*i.e.*, MaskFormer in Fig. 5.5 and DeepLabV3+ in Fig. 5.6), and shows strong robustness to various challenging scenarios with occlusions, small objects and densely arranged targets, *etc.* Moreover, as shown in the last column of Fig. 5.5, MaskFormer makes a severe mistake that misclassifies a part of background structure as `truck`. In contrast, benefiting from hierarchy-aware segmentation learning, HSSN naturally address the issue of mistake severity, *i.e.*, distinguish significantly different concepts with larger margins.

5.3.4 Diagnostic Experiment

To gain more insights into HSSN, we conduct a set of ablative studies on Mapillary Vistas 2.0 [228] and Pascal-Person-Part [342], with ResNet-101 as the backbone.

Key Component Analysis. First, we investigate the essential designs in HSSN, *i.e.*, hierarchical segmentation learning (§5.2.2) with $\mathcal{L}^{\text{FTM}}$ (*cf.* Eq. 5.6) and hierarchical representation learning (§5.2.2) with $\mathcal{L}^{\text{TT}}$ (*cf.* Eq. 5.7). The results are summarized in Table 5.5. The first row refers to a hierarchy-agnostic baseline that

$\mathcal{L}^{\text{FTM}}$	$\mathcal{L}^{\text{TT}}$	Mapillary Vistas 2.0			Pascal-Person-Part		
Eq. 5.6	Eq. 5.7	mIoU^3↑	mIoU^2↑	mIoU^1↑	mIoU^3↑	mIoU^2↑	mIoU^1↑
		81.86	68.17	37.43	93.58	83.04	67.84
✓		84.17	69.62	39.17	96.33	86.72	72.89
	✓	83.06	68.61	38.29	95.92	86.03	72.27
✓	✓	**85.27**	**71.40**	**40.16**	**97.69**	**88.20**	**75.44**

Table 5.5 : **Analysis of essential components** on Mapillary Vistas 2.0 [228] `val` and PASCAL-Person-Part [342] `test` (§5.3.4).

Loss	Mapillary Vistas 2.0			Pascal-Person-Part		
	mIoU^3↑	mIoU^2↑	mIoU^1↑	mIoU^3↑	mIoU^2↑	mIoU^1↑
CCE	81.86	68.17	37.43	93.58	83.04	67.84
BCE	81.56	67.61	37.26	93.12	82.55	67.38
Focal	82.63	68.48	38.09	94.07	83.66	68.42
TM	83.48	69.13	38.69	95.32	85.99	72.17
FTM	84.17	69.62	39.17	96.33	86.72	72.89
Full	**85.27**	**71.40**	**40.16**	**97.69**	**88.20**	**75.44**

Table 5.6 : **Analysis of focal tree-min loss $\mathcal{L}^{\text{FTM}}$** on Mapillary Vistas 2.0 [228] `val` and PASCAL-Person-Part [342] `test` (§5.3.4).

only concerns the leaf nodes and is trained using the categorical cross-entropy loss $\mathcal{L}^{\text{CCE}}$ (*cf.* Eq. 5.1). Three crucial conclusions can be drawn. **First**, our $\mathcal{L}^{\text{FTM}}$ leads to significant performance improvements against the baseline across all the metrics on both datasets. This evidences that our hierarchical segmentation learning strategy is able to produce hierarchy-coherent predictions. **Second**, we also observe compelling gains by incorporating $\mathcal{L}^{\text{TT}}$ into the baseline. This proves the importance of hierarchical representation learning. **Third**, our full model achieves the best performance by combining our $\mathcal{L}^{\text{FTM}}$ and $\mathcal{L}^{\text{TT}}$ together, confirming the necessity of joint hierarchical segmentation and embedding learning.

Focal Tree-Min Loss. We next examine the design of our focal tree-min loss

γ	Mapillary Vistas 2.0			Pascal-Person-Part		
Eq. 5.6	mIoU^3↑	mIoU^2↑	mIoU^1↑	mIoU^3↑	mIoU^2↑	mIoU^1↑
0	84.47	70.24	39.52	96.90	87.56	74.84
0.2	84.53	70.38	39.62	97.17	87.71	74.91
0.5	84.85	70.61	39.72	97.23	87.68	74.94
1.0	85.11	70.95	39.94	97.44	87.97	75.20
2.0	**85.27**	**71.40**	**40.16**	**97.69**	**88.20**	**75.44**
5.0	84.92	70.07	39.40	96.84	87.25	74.65

Table 5.7 : **Analysis of** γ for $\mathcal{L}^{\text{FTM}}$ (Eq. 5.6) on Mapillary Vistas 2.0 [228] `val` and PASCAL-Person-Part [342] `test` (§5.3.4).

Triplet	Margin	Mapillary Vistas 2.0			Pascal-Person-Part		
Loss	m	mIoU^3↑	mIoU^2↑	mIoU^1↑	mIoU^3↑	mIoU^2↑	mIoU^1↑
Vanilla	Constant	84.25	70.13	39.41	96.58	87.03	74.10
$\mathcal{L}^{\text{TT}}$	Constant	84.66	70.42	39.67	97.30	87.86	74.83
$\mathcal{L}^{\text{TT}}$	Hierarchy	**85.27**	**71.40**	**40.16**	**97.69**	**88.20**	**75.44**

Table 5.8 : **Analysis of different variants of** $\mathcal{L}^{\text{TT}}$ on Mapillary Vistas 2.0 [228] `val` and PASCAL-Person-Part [342] `test` (§5.3.4).

$\mathcal{L}^{\text{FTM}}$ (*cf.* Eq. 5.6). As shown in Table 5.6, we compare $\mathcal{L}^{\text{FTM}}$ with four different losses, *i.e.*, categorical cross-entropy loss $\mathcal{L}^{\text{CCE}}$ (*cf.* Eq. 5.1), binary cross-entropy loss $\mathcal{L}^{\text{BCE}}$ (*cf.* Eq. 5.2), focal loss [186], and our tree-min loss $\mathcal{L}^{\text{TM}}$ (*cf.* Eq. 5.5). We can find that our $\mathcal{L}^{\text{TM}}$ generates impressive results, and $\mathcal{L}^{\text{FTM}}$ is even better than $\mathcal{L}^{\text{TM}}$. Then, in Table 5.7, we analyze the impact of the focusing parameter γ in $\mathcal{L}^{\text{FTM}}$. As seen, the performance progressively improves as γ is increased, and the gain becomes marginal when $\gamma=2$. Hence, we choose $\gamma=2$ by default.

Tree-Triplet Loss. We further investigate the design of our tree-triplet loss $\mathcal{L}^{\text{TT}}$ (*cf.* Eq. 5.7). In Table 5.8, "Vanilla" refers to the vanilla triplet loss with a constant margin [267]. By constructing hierarchy-aware triplet samples, our tree-triplet loss $\mathcal{L}^{\text{TT}}$ (also with a constant margin) outperforms "Vanilla". The gains become larger

Distance	Mapillary Vistas 2.0			Pascal-Person-Part		
Measurement	mIoU^3↑	mIoU^2↑	mIoU^1↑	mIoU^3↑	mIoU^2↑	mIoU^1↑
Euclidean	84.23	70.02	39.33	96.28	86.73	73.88
Cosine	**85.27**	**71.40**	**40.16**	**97.69**	**88.20**	**75.44**

Table 5.9 : **Analysis of distance measure** for $\mathcal{L}^{\text{TT}}$ on Mapillary Vistas 2.0 [228] `val` and PASCAL-Person-Part [342] `test` (§5.3.4).

Method	mIoU↑	Memory (MB)↓		Time (Second)↓	
		Training	Inference	Training	Inference
DeepLabV3+ [32]	39.17	4428	1083	0.91	0.023
HSSN [167]	40.16	4897	1106	1.03	0.023

Table 5.10 : **Analysis of model efficiency** for HSSN on Mapillary Vistas 2.0 [228] `val`.

when further applying the hierarchy-induced margin constraint. These results confirm the designs of our tree-triplet loss. Finally, we assess the impact of the distance measurement $\langle \cdot, \cdot \rangle$ used in $\mathcal{L}^{\text{TT}}$. We study Cosine and Euclidean distances. Table 5.9 shows that Cosine distance performs much better than Euclidean distance, corroborating relevant observations in [229, 80, 265].

Efficiency Analysis. Finally, we proceed to study the efficiency of HSSN in terms of both memory consumption and running time, as presented in table 5.10. First, it can be observed that HSSN incurs approximately a 10% increase in memory usage and 13% additional running time during the training stage. This is deemed acceptable particularly in light of the **0.99%** improvement in performance. Second, as our proposed hierarchy-aware segmentation learning primarily focuses on the network training process and the hierarchy-aware pixel label assignment is executed in parallel during inference, our method introduces almost no additional memory and computational overhead when compared to the baseline method. The above analyses collectively state that HSSN not only achieve commendable performance

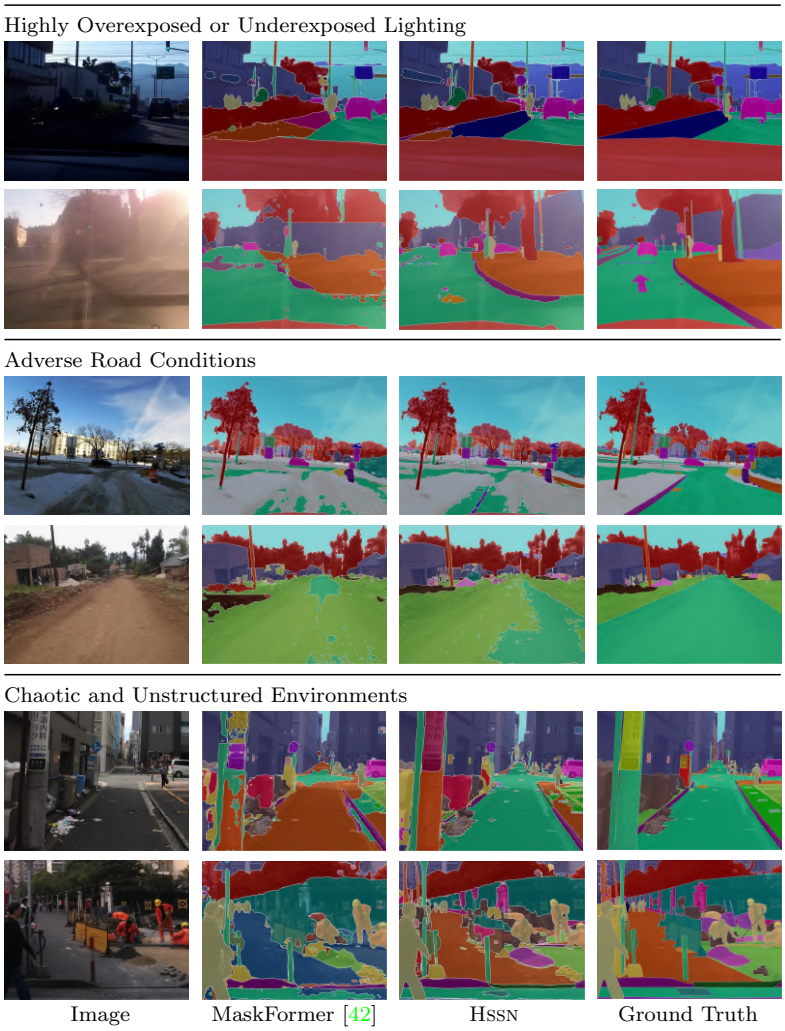


Figure 5.7 : **Representative failure cases** on Mapillary Vistas 2.0 [228] `val`.

but also exhibit high efficiency.

5.4 Failure Case Analysis

While our method significantly enhances the semantic segmentation performance by leveraging class hierarchies, it encounters difficulties in effectively addressing some extremely challenging scenarios. As depicted in Fig. 5.7, we provide a summary of the representative failure cases and identify the characteristic patterns that could potentially lead to inferior outcomes. Specifically, our investigation reveals that failure cases primarily manifest in the following scenarios: i) highly underexposed or overexposed lighting, making it difficult to discern objects; ii) adverse road condi-

tions, such as snowy or dirty roads; iii) chaotic and unstructured environments that are unfavorable to derive hierarchical cues. Despite facing difficulties in these challenging scenarios, our algorithm has demonstrated remarkable improvement over the baseline model. Moreover, the patterns of these failure cases offer valuable insight into the possible directions for future efforts.

5.5 Limitation

Our algorithm is currently designed to facilitate structured scene understanding under the close-world assumption. Nevertheless, expanding it to the incomplete/open-world setup, such as automatic construction of label hierarchies in the presence of novel semantic concepts, delivers an intriguing avenue for our future research. Additionally, the performance of our method when confronted with labels at different levels of granularity or noisy class hierarchies also remains a subject of ongoing investigation.

5.6 Conclusion

In this chapter, we presented HSSN, a structured solution for semantic segmentation. HSSN is capable of exploiting taxonomic semantic relations for structured scene parsing, by only slightly changing existing hierarchy-agnostic segmentation networks. By exploiting hierarchy properties as optimization criteria, hierarchical violation in the segmentation predictions can be explicitly penalized. Through hierarchy-induced margin separation, more effective pixel representations can be generated. We experimentally show that HSSN outperforms many existing segmentation models on four famous datasets. We wish this work to pave the way for future research on hierarchical semantic segmentation.

Chapter 6

Relational Scene Understanding: Modeling Interactional Structures in Human-Object Interaction Detection

Continuing prior investigation into hierarchical structure of semantic categories, this chapter extends the focus to explicit interactions between humans and objects, furthering our goal of achieving a richer, structured understanding of visual scenes.

6.1 Introduction

As a crucial topic in the field of visual scene understanding, human-object interaction (HOI) detection demands not only inferring the semantics and locations of entities but also should comprehend the ongoing events happening between them [360, 67]. Given the complexity and diversity of human activities in object-rich realistic scenes, this task presents challenges in long-tailed distributions and zero-shot discovery [182]. A set of studies seek to tackle these two issues by leveraging large-scale visual-linguistic models (*e.g.*, CLIP [255, 389]) which show strong generalization ability on dozens of tasks. Though strides made, it has been observed that models trained by aligning high-level text-image semantics face difficulties in discerning spatial locations [280], and struggle at compositionality [210] which is a fundamental ability for human to capture new concepts by combining known parts. In fact, both middle-level visual cues (*e.g.*, spatial relation) and compositionality are essential facets for HOI detection. The former can help deduce feasible interactions according to locations between instances, while compositionality contributes significantly to zero-shot generalization. For example, we can easily understand `human-`

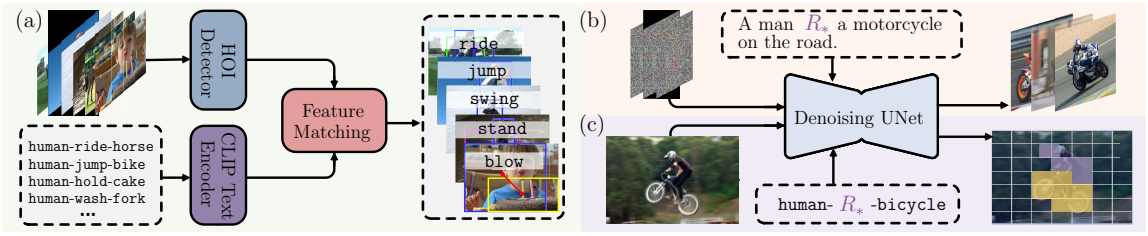


Figure 6.1 : Existing solutions utilize mere linguistic knowledge (a). Our solution utilizes both text-prompt image generation (b) and conditioned feature extraction (c) abilities of diffusion models for knowledge transfer.

`hold-horse` by composing `human-hold-dog` and class `horse` that have encountered previously.

In contrast, the text-to-image diffusion models [264, 188, 190, 97, 256, 373, 8, 262] also pre-trained on large-scale image-text pairs, are demonstrating superior capabilities outperforming models like CLIP. Concretely, they are able to generate diverse high-quality images conditioned on textual inputs, showing proficiency in understanding ***high-level semantics*** [251, 150]. In addition, the generated images convey reasonable shape, texture, layout, and structure, indicating the comprehension in ***mid/low-level visual concepts*** as generative models [348]. More importantly, the descriptions are typically organized in a compositional manner, with phrases such as "happy","near a bridge", or "hugged by a man" continually appended to objects like "a dog". This suggests that diffusion models inherently possess ***compositionality***, to systematically adapt to newly encountered user requirements by composing known visual concepts.

The above analysis motivates us to explore diffusion models for HOI detection. Nonetheless, to fully unlock the potential of diffusion models and accommodate the unique characteristic of HOI detection task, the following questions naturally arise: ❶ With diffusion models typically emphasizing instance generation, how to steer it to prioritize the relationships between humans and objects? ❷ How to transfer the

extensive knowledge obtained from large-scale pre-training in diffusion models to assist the recognition of interactions? To address ❶, we harness textual inversion [78] which conceptualizes a user-provided object by inverting it to a text embedding. However, this method focuses solely on instance objects. To facilitate a smooth shift from object-centric to *relation-centric* modeling, we devise a human-object relation inversion strategy grounded in the disentanglement of HOI. Concretely, given the HOI latent describing `human-action-object`, we build a cycle-consistency objective to reconstruct it from a intermediate relation latent derived from the original HOI latent. This reconstruction process is guided by a set of learnable relation embeddings as text prompts, for which we use the placeholder R_* to denote the textual form before encoded into embedding space. These relation embeddings further involves in a relation-centric contrastive learning to enhance the awareness of high-level relational semantics. To answer ❷, we leverage both the text promoted image generation and conditioned feature extraction abilities of diffusion models. We realize *relation-driven* image generation by compositionally organizing R_* with other linguistic elements to formulate new text prompts (Fig. 6.1(b)). This allows for the generation of novel interactions with unseen objects, and extends the training set for HOI detectors. Moreover, we directly utilize diffusion models as backbone to extract HOI-relevant features conditioned on R_* (Fig. 6.1(c)). After a single noise-free forward step, features distinct for each interaction can be obtained. Finally, to establish a loop for mutual boosting between above *relation-inspired* HOI detection and relation modeling, we devise an online update strategy to facilitate the continual evolving of relation embeddings during HOI detection learning.

Benefited from controllable image generation and knowledge transfer from diffusion models, our method named DiffusionHOI enjoys several appealing advantages: **First**, it steers diffusion models to focus on complex relationships rather than single objects in an efficient way. This offers a robust foundation for HOI

modeling. **Second**, from the perspective of relation-driven, it unlocks the image generation power of diffusion models tailored for the HOI detection task. This enriches the pool of training samples, particularly for long-tailed/unseen interaction classes. **Third**, the relation-inspired prompting improves both the flexibility and accuracy of HOI detectors. It adapts to each individual image to extract action or object related cues, while CLIP-based methods [182, 230] produce action/object features merely from texts (*i.e.*, Fig. 6.1(a)), remaining static and unresponsive to image content.

By embracing text-to-image diffusion models as well as facilitating relation-driven image generation and prompting, our method demonstrates superior performance. It surpasses all top-leading solutions on HICO-DET [27] and V-COCO [99], and sets new state-of-the-arts. In addition, it yields up to **6.43%** mAP improvements on SWiG-HOI [316] under the zero-shot HOI discovery setup. These promising performance evidences the great potential of integrating diffusion models for visual relation understanding. We hope this work could foster the broader exploration of large-scale pre-trained diffusion models on more computer vision tasks beyond mere image generation.

6.2 Methodology

6.2.1 Preliminary: Textual Inversion

Latent diffusion models [262] represent an evolution of diffusion models which offer significant enhancements in both computational and memory efficiency by executing denosing in the latent space. It comprises two primary components. The first is a pre-trained generator equipped with an encoder $\mathcal{E}$ to map the input image x into a latent vector $\boldsymbol{z} = \mathcal{E}(x)$, from which the original data can be reconstructed via a decoder $\mathcal{D}$ by $\hat{x} = \mathcal{D}(\boldsymbol{z}) \approx x$. The second is a diffusion model to generate latent codes $\boldsymbol{z}$ conditioned on user guidance y which can be text, image, *etc*The

latent codes then serve as inputs to $\mathcal{D}$ for image generation *w.r.t.* y. The training objective is given as:

$$\mathcal{L}_{\text{LDM}} := \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\Big[\|\epsilon - \epsilon_\theta(\boldsymbol{z}_t, t, c_\theta(y))\|_2^2\Big], \tag{6.1}$$

where c_θ is a conditioning model to encode y, $\boldsymbol{z}_t$ is the noised latent at time t, ϵ is sampled noise, ϵ_θ is the denoising network. Based on latent diffusion models, inversion-based diffusion [78] seeks to learn a text embedding v_* that accurately describes novel concepts in user provided images. This is achieved by optimizing v_* with Eq. 6.1 to iteratively reconstruct the latent code $\boldsymbol{z}$ of user provided images with text prompts y like "an image of S_*", where S_* is the placeholder of new concept:

$$v_* = \arg\min_v \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\Big[\|\epsilon - \epsilon_\theta(\boldsymbol{z}_t, t, c_\theta(y))\|_2^2\Big]. \tag{6.2}$$

As such, it enables image generation *w.r.t.* target concepts in diverse scenes by using the learned embedding v_* to replace the tokenized placeholder S_* in text prompts.

6.2.2 Inversion-Based HOI Modeling

Disentanglement-based Relation Embedding Learning. To facilitate above inversion technology for relation modeling, two options present: **i)** directly optimizing embeddings describing interactions (*i.e.*, `human-action-object`), which risks overfitting with limited samples for long-tailed categories and cannot generalize to novel concepts, and **ii)** learning `action` embeddings with diverse images sharing a common action but different objects, which seems feasible but poses significant convergence issues due to the complex content, and the optimization target cannot be fixed to actions but not other unrelated elements. In contrast, drawn from the compositional nature of HOI, we adopt a disentangled solution (*i.e.*, Fig. 6.2) where HOI triplets are broken into `human-action` and `object`. Here `human-action` is considered to describe the relation between `human` and `object`, as `action` is executed by and strictly adheres to `human` involved. Then, denoting the text describing `human-action` as R_*, encoded relation embeddings as $v_*^{\text{Rel}} = c_\theta(R_*)$, and the latent of one

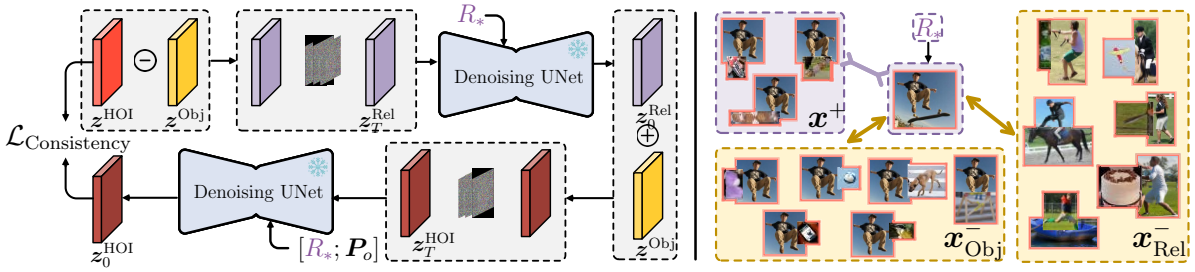


Figure 6.2 : (Left) Disentanglement-based cycle-consistency learning. (Right) Relation-centric contrastive learning.

happening HOI in image as $\boldsymbol{z}^{\text{HOI}}$, a relation latent $\boldsymbol{z}_0^{\text{Rel}}$ could be reconstructed (*i.e.*, denoising with ϵ_θ from time T to 0) by:

$$\epsilon_\theta((\boldsymbol{z}^{\text{HOI}} - \boldsymbol{z}^{\text{Obj}})_T, T, v_*^{\text{Rel}}) \rightarrow \boldsymbol{z}_0^{\text{Rel}}. \tag{6.3}$$

Here $(*)_T$ is the noised version at time T, and $\boldsymbol{z}^{\text{Obj}}$ is retrieved by encoding the cropped object from image with provided bounding box annotations. We consider $\boldsymbol{z}^{\text{HOI}} - \boldsymbol{z}^{\text{Obj}}$ is able to describe the `human-action` component by subtracting the `object` from `human-action-object`. Then, we can reconstruct the latent representing the complete HOI image by adding $\boldsymbol{z}^{\text{Obj}}$ back to $\boldsymbol{z}_0^{\text{Rel}}$:

$$\epsilon_\theta((\boldsymbol{z}_0^{\text{Rel}} + \boldsymbol{z}^{\text{Obj}})_T, T, [v_*^{\text{Rel}}; \boldsymbol{P}_o]) \rightarrow \boldsymbol{z}_0^{\text{HOI}}, \tag{6.4}$$

where $\boldsymbol{P}_o$ is the CLIP encoded text embedding of `object`, and it is combined with the relation embedding v_*^{Rel} to generate the prompt that describes the entire HOI image. In this way, with only one learnable relation embedding (*i.e.*, v_*^{Rel}), we build a cycle to generate relation latent $\boldsymbol{z}_0^{\text{Rel}}$ from the HOI image latent $\boldsymbol{z}^{\text{HOI}}$, and subsequently, the original HOI image latent is reconstructed from the generated relation latent. The learning of v_*^{Rel} can be supervised without human annotation, but just ensuring the consistency between the original HOI latent and the reconstructed one:

$$\mathcal{L}_{\text{Consistency}} = ||\ell_2(\boldsymbol{z}^{\text{HOI}}) - \ell_2(\boldsymbol{z}_0^{\text{HOI}})||_2^2, \tag{6.5}$$

where all latents are ℓ_2-normalized for improved training stability [364]. Through such a disentan- glement-based relation modeling and cycle-consistency training, the optimization objective become clearer and easier to learn. It enables using same `action` from different interactions to enhance the comprehension of a relation, and generalizing to new interactions by combining it with other `object`.

Relation-Centric Contrastive Learning. Eq. 6.5 is a pixel-level reconstruction loss which prioritizes aligning low-level cues. We supplement it with a relation-centric contrastive loss to enhance the awareness of high-level semantics. Instead of directly engaging learning with relation latents, we combine them with object latents to form new HOI latents, thus significantly enriching the diversity of samples:

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{z}_0^{\text{Rel}} + \boldsymbol{z}^{\text{Obj}}, & \boldsymbol{x}^{+} &= \boldsymbol{z}_0^{\text{Rel}} + \boldsymbol{p}^{\text{Obj}}, \\ \boldsymbol{x}_{\text{Obj}}^{-} &= \boldsymbol{z}_0^{\text{Rel}} + \boldsymbol{n}_k^{\text{Obj}}, & \boldsymbol{x}_{\text{Rel}}^{-} &= \boldsymbol{n}_{0,i}^{\text{Rel}} + \boldsymbol{s}_j^{\text{Obj}}, \end{aligned} \tag{6.6}$$

where $\boldsymbol{x}$ is the anchor sample, $\boldsymbol{x}^{+}$ is the positive sample composed of a different object latent $\boldsymbol{p}^{\text{Obj}}$ sharing the same class as $\boldsymbol{z}^{\text{Obj}}$. Conversely, $\boldsymbol{x}_{\text{Obj}}^{-}$ and $\boldsymbol{x}_{\text{Rel}}^{-}$ are negative samples, with $\boldsymbol{x}_{\text{Obj}}^{-}$ composed of a different class object latent $\boldsymbol{n}^{\text{Obj}}$ compared to $\boldsymbol{x}$, and $\boldsymbol{x}_{\text{Rel}}^{-}$ composed of any other relation latent $\boldsymbol{n}_0^{\text{Rel}}$ and arbitrary object latent $\boldsymbol{s}^{\text{Obj}}$. The final optimization objective is given as:

$$\mathcal{L}_{\text{Contrastive}} = -\log \frac{\exp(\boldsymbol{x}\cdot\boldsymbol{x}^{+}/\tau)}{\exp(\boldsymbol{x}\cdot\boldsymbol{x}^{+}/\tau) + \sum_k \exp(\boldsymbol{x}\cdot\boldsymbol{x}_{\text{Obj}}^{-}/\tau) + \sum_i \sum_j \exp(\boldsymbol{x}\cdot\boldsymbol{x}_{\text{Rel}}^{-}/\tau)}, \tag{6.7}$$

to optimize v_*^{Rel} which involves in reconstructing $\boldsymbol{z}_0^{\text{Rel}}$. $\tau = 0.07$ is the temperature parameter.

6.2.3 Relation-Driven Sample Generation

Text Prompts Preparation. We harness the captions provided in the MS COCO Caption dataset [37] to generate diverse prompts. Compared to text synthesized by GPT-4, these captions are more precise and closer to real visual scenes as they are

annotated by human subjects. The preparation initiates with a filtration where captions not containing pronouns indicating `human` (*e.g.*, man, woman, boy) or `action` words are removed. To further enrich the diversity of prompts, given two randomly selected sentences that share the same `action`, we exchange the clauses following the `action` word. Prompts are exclusively generated with GPT-4 only when actions or objects not present in COCO Caption. This results in 33,834 text prompts in total. Finally, `action` words in prompts are replaced with placeholders corresponding to learned relation embeddings, so as to empower the diffusion model with enhanced awareness of relation patterns between `human` and `object` during generation.

Image and Annotation Generation. Denoting text prompts as $\mathcal{P}=\{\mathcal{P}_1,\cdots,\mathcal{P}_N\}$, we aim to construct a dataset $\mathcal{X}=\{(\mathcal{I}_1,\mathcal{A}_1),\cdots,(\mathcal{I}_N,\mathcal{A}_N)\}$ where $\mathcal{I}_i\in\mathbb{R}^{H\times W\times 3}$ represents the synthesized image and $\mathcal{A}_i=\{\mathcal{B}_i^h,\mathcal{B}_i^o,\mathcal{C}_i^o,\mathcal{C}_i^a\}$ is the pseudo annotation containing bounding boxes $\mathcal{B}_i^h$ for `human`, $\mathcal{B}_i^o$ for `object`, and class labels $\mathcal{C}_i^o$ for `object`, $\mathcal{C}_i^a$ for `action`. For the generation of $\mathcal{I}_i$, the text prompts $\mathcal{P}_i$ is first encoded by CLIP text encoder to obtain the conditioning vector $\boldsymbol{P}_i=c_\theta(\mathcal{P}_i)\in\mathbb{R}^d$, where the placeholder string is directly replaced with relation embedding v_*^{Rel}. Then, a random sampled noise tensor $\boldsymbol{z}_T\in\mathbb{R}^{h\times w\times d}$ is iteratively denoised to yield a new latent $\boldsymbol{z}_0$. $\mathcal{I}_i$ is generated by a single pass through $\mathcal{D}$, *i.e.*, $\mathcal{I}_i=\mathcal{D}(\boldsymbol{z}_0)$. For the generation of $\mathcal{A}_i$, $\mathcal{C}_i^o$ and $\mathcal{C}_i^a$ can be easily determined by referring to the `action` and `object` words in $\mathcal{P}_i$, while $\mathcal{B}_i^h$ and $\mathcal{B}_i^o$ are derived from the cross-attention maps computed within the U-shape denoising network ϵ_θ. Specifically, to effectively tackle various input modalities, ϵ_θ is equipped with cross-attention mechanisms in each layer to inject $\boldsymbol{P}_i$ into $\boldsymbol{z}$ conforming to the similarity between them. For the l-th layer at the last denoising step 0, the cross-attention map is computed as: $\boldsymbol{M}_{i,0}^l=\texttt{softmax}(\boldsymbol{z}_0\cdot\boldsymbol{P}_i^\top/\sqrt{d})\in\mathbb{R}^{h\times w}$. According to prior work [348, 381], here $\boldsymbol{M}_{i,0}^l$ signifies the correspondence between text prompt $\mathcal{P}_i$ and regions in generated image. Thus, we explicitly concatenate words describing `human` and `object` with $\mathcal{P}_i$ (*i.e.*, $[\mathcal{P}_i;word_{\texttt{human}};word_{\texttt{object}}]$), re-

sulting in a new text embedding $\hat{\boldsymbol{P}}_i \in \mathbb{R}^{d\times 3}$ and corresponding cross-attention maps $\hat{\boldsymbol{M}}^l_{i,0} \in \mathbb{R}^{h\times w\times 3}$ where the last two items along the third dimension channel are probability maps of `human` and `object`. Finally, we leverage the implementation in weakly supervised object localization [46] to outline bounding boxes from these probability maps.

6.2.4 HOI Knowledge Transfer from Diffusion Models

While prior studies [315, 254, 60, 182] have investigated knowledge transfer from visual-linguistic models such as CLIP, they utilize visual knowledge solely during training. The prediction relies on a confined set of CLIP encoded word embeddings, which leads to limited knowledge transfer and rigid inference unresponsive to image content. In contrast, we propose directly leveraging diffusion models as the feature extractor and build HOI detector on this basis. Moreover, given the conditioning property of diffusion model, relation embeddings can serve as text prompts to guide the retrieval of interaction-relevant visual cues from images, further benefiting HOI detection.

HOI Detector Built Upon Diffusion Models. Pioneering studies [348, 381, 107] have empirically demonstrated that the output of frozen text-to-image diffusion models possesses rich visual features to tackle complex perception tasks. Next we illustrate how to build a HOI detector on this basis. As shown in Fig. 6.3, our method is a one-stage solution composed of: a visual encoder with diffusion models serving as the backbone, and a HOI decoder consisting of two parallel decoders for instance and interaction detection. For the visual encoder, given an image $\mathcal{I}$, it is encoded into latent space with the encoder $\mathcal{E}$ of a pre-trained generator (*e.g.*, VQGAN): $\boldsymbol{z} = \mathcal{E}(\mathcal{I})$. Then, $\boldsymbol{z}$ is fed into ϵ_θ through a single noise-free forward pass to derive text conditioned features: $\epsilon_\theta(\boldsymbol{z}, T, c_\theta(y)) \rightarrow \{\boldsymbol{z}^l_T\}^4_{l=1}$. All scales of features are aggregated with FPN [185], yielding $\boldsymbol{z}'_T$ in a downsampling factor of 32. The

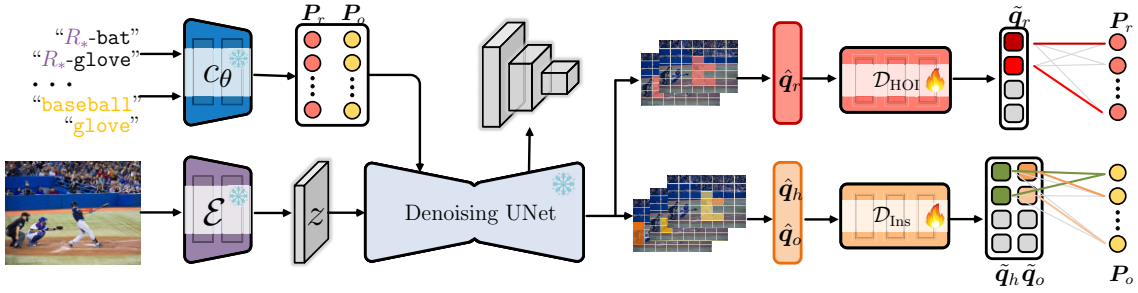


Figure 6.3 : The overall pipeline of DIFFUSIONHOI. See §6.2.4 for details.

architecture of HOI decoder is similar to GEN-VLKT [182]. Concretely, the instance decoder $\mathcal{D}_{\text{Ins}}$ employs a set of human queries $\{\boldsymbol{q}_h^i\}_{i=1}^{N_q}$ and object queries $\{\boldsymbol{q}_o^j\}_{j=1}^{N_q}$, and considers those at the same index (*i.e.*, $i=j$) as a pair to initialize queries $\boldsymbol{q}_r$ for the interaction decoder $\mathcal{D}_{\text{HOI}}$. In fact, the HOI decoder can be replaced with any other one-stage models. We do not claim the detector architecture as the contribution, but focus on how to derive HOI-relevant feature to assist in HOI detection.

Relation-Inspired HOI Detection. As the relation embeddings are optimized towards modeling the interactions between `human` and `object`, we use them as conditions to inspire the extraction of HOI-relevant cues. This can eliminate the potential domain gap between general-purpose diffusion models and the downstream HOI detection task. Specifically, all feasible HOI phrases (*e.g.*, "human feed horse") are encoded with CLIP text encoder into embedding space and concatenated together, with the `human-action` component replaced with learned relation embeddings (*e.g.*, "R_* horse"). This results in HOI prompts $\boldsymbol{P}_r \in \mathbb{R}^{N_r \times d_l}$ which further participates into the cross-attention in ϵ_θ via:

$$\boldsymbol{z}_T^l = \boldsymbol{z}_T^l + \boldsymbol{M}_r^l \cdot \boldsymbol{V}_{\boldsymbol{P}_r} \in \mathbb{R}^{h \times w \times d_l}, \qquad \boldsymbol{M}_r^l = \texttt{softmax}(\boldsymbol{z}_T^l \cdot \boldsymbol{K}_{\boldsymbol{P}_r}^\top / \sqrt{d}) \in \mathbb{R}^{h \times w \times N_r}, \tag{6.8}$$

where $\boldsymbol{K}_{\boldsymbol{P}_r}$ and $\boldsymbol{V}_{\boldsymbol{P}_r}$ are key and value embeddings projected from $\boldsymbol{P}_r$. As seen, $\boldsymbol{P}_r$ contributes to: **i)** encourage the denoising network ϵ_θ to extract visual features $\boldsymbol{z}_T^l$ *w.r.t.* HOI prompts, and **ii)** guide the derivative of cross-attention maps $\boldsymbol{M}_r^l$ in response to ongoing interactions in $\mathcal{I}$. The final interaction maps are computed

as the average value of $\{\boldsymbol{M}_r^l\}_{l=1}^4$. we also derive cross-attention maps for `human` $\boldsymbol{M}_h$ and `object` $\boldsymbol{M}_o$ in a similar way as $\boldsymbol{M}_r$, by without update to $\boldsymbol{z}_T^l$. Then, these cross-attention maps are used to initialize queries from the aggregated visual feature $\boldsymbol{z}_T'$ via mask pooling:

$$\begin{aligned}\hat{\boldsymbol{q}}_r &= \texttt{MaskPooling}(\boldsymbol{z}_T', \boldsymbol{M}_r^k),\\ \hat{\boldsymbol{q}}_o &= \texttt{MaskPooling}(\boldsymbol{z}_T', \boldsymbol{M}_o),\\ \hat{\boldsymbol{q}}_h &= \texttt{MaskPooling}(\boldsymbol{z}_T', \boldsymbol{M}_h).\end{aligned} \tag{6.9}$$

Note we conduct Hungarian matching between $\hat{\boldsymbol{q}}_r^k$ and $\hat{\boldsymbol{q}}_o^i + \hat{\boldsymbol{q}}_h^j$, so as to arrange HOI, and combined human-object queries that are most similar to the same index in their respective query lists. Following [348], the classification for interaction and instance are jointly supervised by:

$$\begin{aligned}\mathcal{L}_{\text{HOI}} &= \texttt{CE}(\texttt{softmax}(\tilde{\boldsymbol{q}}_r \cdot \boldsymbol{P}_r/\tau_r), y_r) + \texttt{CE}(\texttt{softmax}(\texttt{FFN}(\tilde{\boldsymbol{q}}_r)), y_r),\\ \mathcal{L}_{\text{Ins}} &= \texttt{CE}(\texttt{softmax}(\tilde{\boldsymbol{q}}_o \cdot \boldsymbol{P}_o/\tau_o), y_o) + \texttt{CE}(\texttt{softmax}(\texttt{FFN}(\tilde{\boldsymbol{q}}_o)), y_o),\end{aligned} \tag{6.10}$$

where y_r and y_o are ground truth for interaction and object categories, $\tilde{\boldsymbol{q}}_r$ and $\tilde{\boldsymbol{q}}_o$ are queries after decoding through $\mathcal{D}_{\text{HOI}}$ and $\mathcal{D}_{\text{Ins}}$, `CE` and `FFN` denote the cross entropy loss and feed-forward network. Beyond the score delivered by conventional linear classifier (*i.e.*, $\texttt{softmax}(\texttt{FFN}(\tilde{\boldsymbol{q}}_o))$), here $\texttt{softmax}(\tilde{\boldsymbol{q}} \cdot \boldsymbol{P}/\tau)$ with learnable parameters τ_r and τ_o computes the similarity between decoded queries and conditioning prompts, thereby facilitating the recognition for unseen categories.

Online Update for Relation Embedding. To enable the continual evolution of relation embeddings v_*^{Rel} throughout the supervised HOI detection learning, an additional loss considering the compositional nature of HOI is devised. Specifically, we concatenate all N_a relation embeddings into a new prompt $\boldsymbol{P}_a$, from which a set of relation query embeddings $\hat{\boldsymbol{q}}_a$ can be initialized in the same way as $\hat{\boldsymbol{q}}_o$ (*c.f.*, Eq. 6.9). In addition, another set of embeddings describing relations can be derived from $\tilde{\boldsymbol{q}}_r$ and $\tilde{\boldsymbol{q}}_o$ by: $\tilde{\boldsymbol{q}}_a = \tilde{\boldsymbol{q}}_r - \tilde{\boldsymbol{q}}_o$. The goal is to align $\hat{\boldsymbol{q}}_a$ directly derived from visual features with relation embeddings as conditions, and $\tilde{\boldsymbol{q}}_a$ computed from interaction

and object queries after decoding:

$$\mathcal{L}_{\text{Rel}} = ||\ell_2(\hat{\boldsymbol{q}}_a) - \ell_2(\tilde{\boldsymbol{q}}_a)||_2^2. \tag{6.11}$$

Here $\mathcal{L}_{\text{Rel}}$ solely optimizes v_*^{Rel} to render a mutual boost between HOI detection and relation embedding learning. Concretely, enhanced relation embeddings inspires improved HOI feature discovery, and in turn, the more precise query decoding benefits the update of relation embeddings.

6.2.5 Implementation Details

Network Architecture. DIFFUSIONHOI is built upon Stable Diffusion v1.5 with xFormers [157] installed. The denoising UNet ϵ_θ receives input latents at a downsampling factor of 1/8, with four encoder blocks output feature at a size of $1/2^{l+3}$ where l is the block index. For the final visual feature $\boldsymbol{z}'$ after FPN aggregation, it is interpolated to a size of 1/32 and then projected to 256 channels to enhance computing efficiency. Both $\mathcal{D}_{\text{HOI}}$ and $\mathcal{D}_{\text{Ins}}$ consist of six Transformer decoding layers with hidden dimension of 768. The query number N^q is uniformly set to 64 for both $\mathcal{D}_{\text{HOI}}$ and $\mathcal{D}_{\text{Ins}}$.

Training Objective. The inversion-based HOI modeling is jointly optimized by two embedding learning losses: $\mathcal{L}_{\text{Inversion}} = \mathcal{L}_{\text{Consistency}} + \lambda_1 \mathcal{L}_{\text{Contrastive}}$, where $\lambda_1 \in [0, 0.2]$ is scheduled following a cosine annealing policy. For HOI detection learning, we follow DETR [24] to match predictions and ground truths with Hungarian algorithm. Denoting the bounding box detection loss as $\mathcal{L}_{\text{Det}}$, the final training objective is given as: $\mathcal{L} = \mathcal{L}_{\text{HOI}} + \mathcal{L}_{\text{Ins}} + \mathcal{L}_{\text{Det}} + \lambda_2 \mathcal{L}_{\text{Rel}}$ where λ_2 is fixed to 0.5.

6.3 Experiment

6.3.1 Experimental Setup

Datasets. We conduct extensive experiments on three datasets.

- HICO-DET [27] is a large-scale HOI detection benchmark with 38,118/9,658 images for training/testing, respectively. This dataset includes 80 object categories as in MS-COCO [187] and 117 action categories, formulating a rich vocabulary of 600 human-object interactions in total.
- V-COCO [99] is a curated subset of MS-COCO [187] including 2,533/2,867/4,946 images in `train`/`val`/ `test` sets. It also contains 80 object categories from MS-COCO [187] and a much smaller set of 29 action classes, resulting in a total of 263 human-object interactions.
- SWiG-HOI [316] is assembled from SWiG [250] and DOH [270] with about 45,000/ 14,000 for training/testing. This dataset covers 406 human actions and 1,000 object categories.

Zero-Shot HOI Discovery. In accordance with prior research [110, 112, 111, 113, 182, 230], the zero-shot HOI discovery on HICO-DET [27] uses four setups: Rare First Unseen Combination (RF-UC), Non-rare First Unseen Combination (NF-UC), Unseen Verb (UV), and Unseen Object (UO). The RF-UC and NF-UC configurations excluded the 120 most frequent/infrequent interaction categories from the training sets for testing purposes only. The UV and UO setups reserve 20 verb classes and 12 object classes never encountered during training for testing. For SWiG-HOI [316], the test set includes approximately 5,500 interactions, with around 1,800 of them not present in the training set.

Evaluation Metric. Following conventions [284, 141, 182], we adopt mAP as metrics. For HICO-DET, we report performance according to Default and Known Object two setups. The former computes mAP across all testing images, while the latter is tailored for each object class. For each setup, the scores are reported in Full/Rare/Non-Rare three types. For V-COCO, we evaluate the performance under scenario 1 (S1) which contains all 29 actions and scenario 2 (S2) which excludes 4

actions interact with no objects. For zero-shot setup, the evaluation is divided into Seen/Unseen/Full three sets for HICO-DET, and Non-Rare/Rare/Unseen/Full four sets for SWiG-HOI.

Training and Testing. The diffusion model and CLIP text encoder are kept frozen during training. For inversion-based HOI modeling, the only learnable parameters are relation embeddings, which are updated for 40,000 steps using images sampled from HICO-DET. Following [78], we employ a base learning rate of $8e^{-2}$ with a batch size of 32. For HOI detection learning, we train the interaction decoder $\mathcal{D}_{\text{Ins}}$ and object decoder $\mathcal{D}_{\text{HOI}}$ for 60 epochs with a base learning rate of $1e^{-4}$ and batch size of 16, using both synthesized data and the target dataset. Subsequently, the model is trained only on the target dataset for an additional 30 epochs with a base learning rate of $1e^{-5}$. During inference, no data augmentation is used to ensure fair comparison. Following [386, 182], the inputs are resized to maximum of 1,333 pixels on long sides, and the shortest sides falls between 480 and 800 pixels.

Reproducibility. DiffusionHOI is implemented in PyTorch and trained on 8 Tesla A40 GPUs with 48GB memory per card. To ensure reproducibility, our full code will be released.

6.3.2 Comparison with State-of-the-Arts

Regular Setup. We first compare DiffusionHOI with top-leading solutions on HICO-DET [27] and V-COCO [99] under the regular setup. As shown in Table 6.1, for HICO-DET, our method achieves the best performance on both Default and Known Object setups. Notably, with the encoder of VQGAN as the backbone, it surpasses the previous SOTA, RemLR [23], which employs a similar level backbone (*i.e.*, ResNet-50) by **1.19%** and **2.64%** on the Full categories. Benefited from synthesized data and comprehensive knowledge transfer from diffusion models, the performance on Rare categories improves significantly, achieving higher scores than

on Non-Rare categories for the first time. Finally, with a more powerful VL model (*i.e.*, Stable unCLIP [256]) and backbone (*i.e.*, ViT-L), the performance is boosted to **42.54%** under the Default setup, surpassing nearly all existing work by a considerable margin. Please note that PViC [370] with Swin-L as the backbone leverages $\mathcal{H}$-Deform-DETR [131] as the detector which achieves 48.7 mAP on MS COCO [187] by running merely 12 epochs, significantly higher than DETR [24] which achieves 36.2 mAP by running 50 epochs.

Method	Backbone	VL Pretrain	Default			Known Object			V-COCO	
			Full	Rare	Non-Rare	Full	Rare	Non-Rare	AP^{S1}_{role}	AP^{S2}_{role}
iCAN[82][BMVC18]	R50	-	14.84	10.45	16.150	16.26	11.33	17.73	45.3	-
PPDM[181][CVPR20]	HG104	-	21.73	13.78	24.10	24.58	16.65	26.84	-	-
HOTR[141][CVPR21]	R50	-	23.46	16.21	25.60	-	-	-	55.2	64.4
QPIC[284][CVPR21]	R101	-	29.90	23.92	31.69	32.38	26.06	34.27	58.3	60.7
CDN[366][NeurIPS21]	R101	-	32.07	27.19	33.53	34.79	29.48	36.38	63.9	65.9
CPChoi[241][CVPR22]	R50	-	29.63	23.14	31.57	-	-	-	63.1	65.4
STIP[378][CVPR22]	R50	-	32.22	28.15	33.43	35.29	31.43	36.45	66.0	70.7
UPT[369][CVPR22]	R101	-	32.62	28.62	33.81	36.08	31.41	37.47	61.3	67.1
Iwin[292][ECCV22]	R101	-	32.79	27.84	35.40	35.84	28.74	36.09	60.9	-
MCPC[340][ECCV22]	R50	-	35.15	33.71	35.58	37.56	35.87	38.06	63.0	65.1
PViC[370] [ICCV23]	R50	-	34.69	32.14	35.45	38.14	35.38	38.97	62.8	67.8
PViC†[370] [ICCV23]	Swin-L	-	44.32	44.61	44.24	47.81	48.38	47.64	64.1	70.2
GEN-VLK[182][CVPR22]	R101	CLIP	34.95	31.18	36.08	38.22	34.36	39.37	63.6	65.9
HOICLIP[230][CVPR23]	R50	CLIP	34.69	31.12	35.74	37.61	34.47	38.54	63.5	64.8
CQL[344][CVPR23]	R50	CLIP	35.36	32.97	36.07	38.43	34.85	39.50	66.4	69.2
ViPLO[242][CVPR23]	ViT-B	CLIP	37.22	35.45	37.75	40.61	38.82	41.15	62.2	68.0
AGER[293][ICCV23]	R50	CLIP	36.75	33.53	37.71	39.84	35.58	40.2	65.7	69.7
RmLR[23][ICCV23]	R101	MobileBERT	37.41	28.81	39.97	38.69	31.27	40.91	64.2	70.2
ADA-CM[158][ICCV23]	ViT-L	CLIP	38.40	37.52	38.66	-	-	-	58.6	64.0
DiffusionHOI	VQGAN	Stable Diffusion	**38.12**	**38.93**	**37.84**	**40.93**	**42.87**	**40.04**	**66.8**	**70.9**
DiffusionHOI	ViT-L	Stable unCLIP	**42.54**	**42.95**	**42.35**	**44.91**	**45.18**	**44.83**	**67.1**	**71.1**

†: Models built upon advanced object dector, *i.e.*, $\mathcal{H}$-Deform-DETR [131].

Table 6.1 : Quantitative results for regular HOI detection on HICO-DET [27] and V-COCO [99].

Zero-Shot Setup. Next we investigate the effectiveness of DiffusionHOI under the zero-shot generalization setup. As shown in Table 6.2, our method yields remarkable performance across all four setups on HICO-DET. In particular, it surpasses the previous SOTA (*i.e.*, HOICLIP [230]) by **2.90%** under the RF-UC setup. This setup emphasizes compositional generalization which requires models to comprehend new types of interactions using known actions and objects. It aligns well with the strengths of text-to-image diffusion models to generate images conditioned on compositionally organized textual descriptions. Moreover, due to the effective knowledge transfer, DiffusionHOI also achieves satisfactory improvement under the UV and UO setups which focus on the recognition of novel actions and objects. Table 6.3 further confirms the exceptional ability of our method, showing **5.97%**/**8.23%** mAP improvements over CMD-SE [159] under Rare and Unseen two categories.

Model Efficiency. We compare the trainable parameter number and inference time in Table 6.4. As seen, DiffusionHOI demonstrates significantly fewer trainable parameters compared to the one-stage counterparts. This is attributed to our inversion-based HOI modeling, which avoids fine-tuning diffusion models like previous work [381], while effectively capturing task-specific properties. Regarding inference speed, even with stable diffusion for feature extraction, our method still achieves 9.49 FPS, a rate similar to two-stage models. This is due to the inference involving only one single forward pass, and the downsampling factor of stable diffusion from 1/8 to 1/64 is smaller than conventional backbones typically from 1/4 to 1/32. Moreover, thank to the flourishing community of stable diffusion, a variety of optimized inference solutions have emerged. By running at fp16 precision and using traced UNet, the FPS increases to 24.77, surpassing most one-stage methods.

Table 6.2 : Zero-shot generalization on HICO-DET [27].

Method	Type	Unseen	Seen	Full
ATL[111][CVPR21]	RF-UC	9.18	24.67	21.57
FCL[112][CVPR21]	RF-UC	13.16	24.23	22.01
SCL[113][ECCV22]	RF-UC	19.07	30.39	28.08
GEN-VLKT[182][CVPR22]	RF-UC	21.36	32.91	30.56
OpenCat[386][CVPR23]	RF-UC	21.46	33.86	31.38
HOICLIP[230][CVPR23]	RF-UC	25.53	34.85	32.99
DiffusionHOI	RF-UC	**32.06**	**36.77**	**35.89**
ATL[111][CVPR21]	NF-UC	18.25	18.78	18.67
FCL[112][CVPR21]	NF-UC	18.66	19.55	19.37
SCL[113][ECCV22]	NF-UC	21.73	25.00	24.34
GEN-VLKT[182][CVPR22]	NF-UC	25.05	23.38	23.71
OpenCat[386][CVPR23]	NF-UC	23.25	28.04	27.08
HOICLIP[230][CVPR23]	NF-UC	26.39	28.10	27.75
DiffusionHOI	NF-UC	**30.04**	**30.29**	**30.25**
ATL[111][CVPR21]	UO	5.05	14.69	13.08
FCL[112][CVPR21]	UO	15.54	20.74	19.87
GEN-VLKT[182][CVPR22]	UO	10.51	28.92	25.63
OpenCat[386][CVPR23]	UO	23.84	28.49	27.72
HOICLIP[230][CVPR23]	UO	16.20	30.99	28.53
DiffusionHOI	UO	**22.37**	**32.03**	**31.12**
GEN-VLKT[182][CVPR22]	UV	20.96	30.23	28.74
HOICLIP[230][CVPR23]	UV	24.30	32.19	31.09
DiffusionHOI	UV	**28.05**	**33.24**	**32.67**

Table 6.3 : Zero-shot generalization on SWiG-DET [316].

Method	Non-rare	Rare	Unseen	Full
QPIC[284][CVPR21]	16.95	10.84	6.21	11.12
THID[315][CVPR22]	17.67	12.82	10.04	13.26
CMD-SE[159][CVPR24]	21.46	14.64	10.70	15.26
DiffusionHOI	**25.59**	**20.61**	**18.93**	**21.69**

Table 6.4 : Comparison of parameters and running efficiency. * means applying accelerated technology.

Method	Backbone	Trainable Params (M)	FPS	HICO-DET
Two-stages Detectors:				
iCAN[82][BMVC18]	R50	39.8	6.23	14.84
DRG[81][ECCV20]	R50	46.1	6.05	19.26
STIP[378][CVPR22]	R50	50.4	7.12	32.22
ViPLO[242][CVPR23]	ViT-B	118.2	5.66	37.22
ADA-CM[158][ICCV23]	ViT-L	6.6	3.24	38.40
One-stages Detectors:				
PPDM[181][CVPR20]	HG104	194.9	17.58	21.73
HOTR[141][CVPR21]	R50	51.2	15.92	23.46
QPIC[284][CVPR21]	R50	41.9	17.41	29.07
CDN[366][NeurIPS21]	R50	42.1	16.24	31.78
GEN-VLKT[182][CVPR22]	R50	42.8	18.23	33.75
DiffusionHOI	VQ-GAN	27.6	9.49	38.12
*DiffusionHOI	VQ-GAN	27.6	24.77	38.12

6.3.3 Diagnostic Analysis

Key Component Analysis. We first examine the essential components of DiffusionHOI in Table 6.5. Here Baseline denotes HOI detector built upon stable diffusion without text prompting. Through jointly training with the synthesized data, both Default and RF-UC setups observe notable improvements (*e.g.*, up to **2.25%** and **3.32%** on Full categories). This verifies the effectiveness of our relation-driven HOI image generation strategy. in addition, after imposing relation embedding to

prompt the feature extraction and HOI detection processes, the performance boosts to **36.45%** and **34.25%** under two setups. Finally, after combining these two core components together, our DiffusionHOI delivers consistent improvements and sets new SOTA across all setups.

Conditioning Input. To assess the effectiveness of learned relational embeddings, we present the experimental results using different conditional inputs to stimulate HOI detection in Table 6.6. As seen, though action words offer limited improvement, they are far surpassed by relation embeddings which enables HOI-oriented feature extraction and enhance query initialization through cross-attention.

Relation Embedding Learning. Next we probe the impact of different strategies for relation embedding learning. The results regarding relation-inspired HOI detection are summarized in Table 6.7. It can be observed that textual inversion, which directly uses different images sharing the same action for relation embedding learning, is inferior to our cycle-consistency learning strategy that considers the disentanglement nature of HOI interactions. On this basis, the relation-centric contrastive learning and online update strategies consistently bring improvement in both setups.

Prompt for Dataset Generation. Finally we study the impact of data synthesized by different types of textual prompts in Table 6.8. As observed, data generated with purely textual description using plain action words like "The man at bat readies to swing at the pitch" gives negative improvement over baseline. The potentially indicates that diffusion models cannot understand the relations between human-object pairs and generate meaningful images when provided with straightforward textual description. In contrast, through relation modeling, data generated with relation embeddings to replace the plain action words provides high-quality samples for the training of HOI detectors.

Algorithm Component	Default			RF-UC		
	Full	Rare	Non-Rare	Full	Unseen	Seen
BASELINE	33.24	30.25	34.32	30.47	20.63	33.09
+ Synthesized Data *only*	35.49	36.27	35.02	33.79	28.22	34.85
+ Relation Prompting *only*	36.45	35.78	36.71	34.25	26.57	35.58
DIFFUSIONHOI	38.12	38.93	37.84	35.89	32.06	36.77

Table 6.5 : Detailed analysis of essential components of DIFFUSIONHOI on HICO-DET [27].

Learning Strategy	Default			RF-UC		
	Full	Rare	Non-Rare	Full	Unseen	Seen
Textual Inversion	34.03	32.17	34.61	30.93	21.55	33.45
Cycle-Consistency	35.23	34.56	35.46	32.96	24.24	34.54
+ Relation-Centric CL	35.94	35.06	36.32	33.72	25.73	35.17
+ Online Update	36.45	35.78	36.71	34.25	26.57	35.58

Table 6.7 : Analysis of relation embeddings with different learning strategies for relation-inspired HOI detection.

Conditioning Input	Default			RF-UC		
	Full	Rare	Non-Rare	Full	Unseen	Seen
-	33.24	30.25	34.32	30.47	20.63	33.09
Textual Description	33.71	30.98	34.73	30.72	21.29	33.24
Relation Embedding	36.45	35.78	36.71	34.25	26.57	35.58

Table 6.6 : Analysis of conditioning input for relation-inspired HOI detection on HICO-DET [27].

Training Set	Default			RF-UC		
	Full	Rare	Non-Rare	Full	Unseen	Seen
HICO-DET	33.24	30.25	34.32	30.47	20.63	33.09
+ *TD* Synthesized Data	32.54	30.04	33.49	30.12	20.55	32.57
+ *RE* Synthesized Data	35.49	36.27	35.02	33.79	28.22	34.85

Table 6.8 : Analysis of prompts for dataset generation. *TD*: textual description, *RE*: relation embedding.

Analysis on Training Cost. For our inversion-based HOI modeling to learn relation-centric embeddings, unlike the original textual inversion technology that learns text embeddings within the image space, we optimize relation embeddings within the latent space by reconstructing interaction features. This lead to reduced training costs. Consequently, the 117 relation embeddings in HICO-DET [27] can be learned within **5.7** hours (23 minutes per relation embedding) which is more efficient than textual inversion (*i.e.*, 32 minutes per embedding). For the main training of HOI detection on HICO-DET, since our method utilizes significantly fewer trainable parameters compared to existing work (*e.g.*, 27.6M *v.s.* 50.4M for STIP [378], 41.9M for QPIC [284], and 42.8M for GEN-VLKT [182] in Table 6.4), the training process can be completed in just **11.5** hours.

Figure 6.4 : Typical failure case on HICO-DET. Actions highlighted in red indicate missing predictions that should be detected, while text with ~~strikethrough~~ means wrong predictions that should be removed.

6.4 Failure Case

As shown in Fig. 6.4, we found that failure cases primarily manifest in the following scenarios: i) scenes featuring only partial human bodies, such as arm or leg, which introduces challenges for person detection; and ii) chaotic scenes teeming with people, which causes occlusion and difficulties in identifying interactions. Despite these challenges, DiffusionHOI has shown remarkable improvement over existing approaches. Additionally, the patterns of these failure cases provide valuable insights for future research.

6.5 Conclusion

In this chapter, we present DiffusionHOI, a new HOI detector built upon diffusion models. By explicitly modeling the relations between humans and objects in an inversion-based manner, we enable effective knowledge transfer from diffusion models while adapting unique characteristics of the HOI detection task. This is achieved

by: **i)** *relation-driven* image generation using diffusion models to enrich the training set with more HOI-oriented samples, and **ii)** *relation-inspired* HOI detection with learned relation embeddings as prompts to retrieve task-specific features from images, thereby enhancing the recognition of ongoing interactions. Extensive experiments demonstrate that DiffusionHOI excels in both regular or zero-shot setups and sets new SOTAs. We believe this work offers valuable insights into harnessing the potential of generative diffusion models for structured relation modeling.

Chapter 7

Neural-Logic Integration for Semantic Parsing: The LogicSeg Framework

The previous chapter details our efforts in relational modeling for structured scene understanding. While it demonstrated rich relational structures can be learned implicitly from data, this chapter explores how relations can be formally represented as symbolic rules and integrated with neural networks in a general manner, thereby bridging the gap between experience-driven perception and logic-induced reasoning.

7.1 Introduction

Interpreting high-level semantic concepts of visual stimuli is an integral aspect of human perception and cognition, and has been a subject of interest in computer vision for nearly as long as this discipline has existed. As an exemplar task of visual semantic interpretation, *semantic segmentation* aims to group pixels into different semantic units. Progress in this field has been notable since the seminal work of fully convolution networks (FCNs) [196] and been further advanced by the recent launch of fully attention networks (Transformer) [302].

Despite these technological strides, we still observe current prevalent segmentation systems lack in-depth reflection on some intrinsic nature of human cognition. **First**, standard segmentation systems simply assume the semantic concepts in the set of interest have no underlying relation and predict all these concepts *exclusively*. By contrast, humans interpret a scene by components. For example in Fig. 7.1, we can effortlessly recognize many pieces of `furniture`, such as `chair`s and

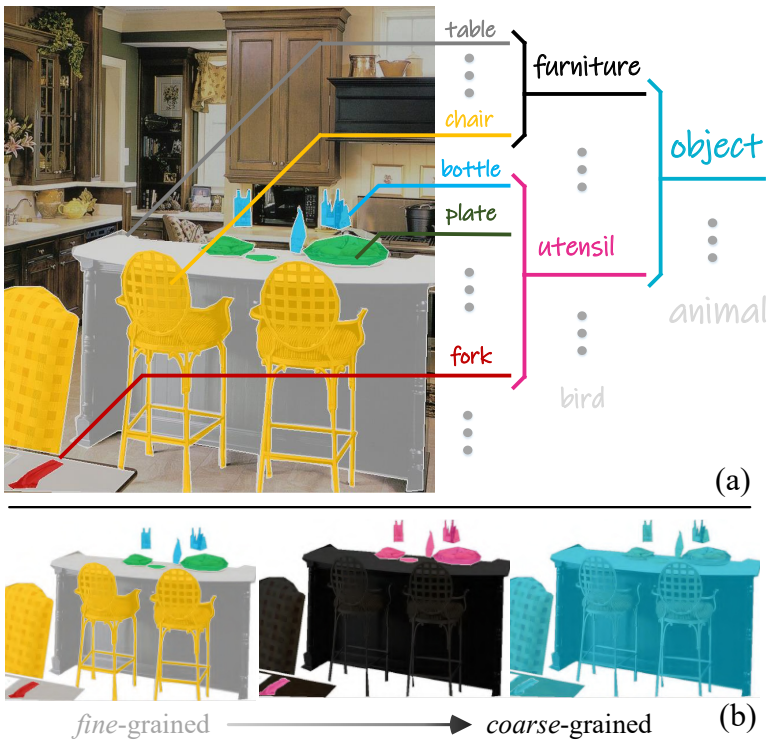


Figure 7.1 : (a) We humans abstract our perception in a structured manner, and conduct reasoning through symbol manipulation over such multi-level abstraction. (b) We aim to *holistically* interpret visual semantics, through the integration of both data-driven sub-symbolic learning and symbolic knowledge-based logic reasoning.

`tables`, and identify various `utensils`, *e.g.*, `bottles`, and `plates`. Such capacity of structured understanding of visual semantics is an innate aspect of human perception [20], complies with our way of the organization of knowledge [75, 358], and has a close relation to many meta-cognitive skills including *compositional generalization* (*i.e.*, making infinite use of finite means) [126], *systematicity* (*i.e.*, cognitive capacity comes in groups of related behaviours) [74], and *interpretability* (*i.e.*, interpreting complex concepts with simpler ones) [238, 274]. Despite its significance and ubiquity, surprisingly little has been done on the computational modeling of structured visual perception in the segmentation literature. Though exceptions exist [343, 179, 176, 167, 328], in general they are scattered, lacking systematic study.

Second, the latest semantic segmentation systems, label structure aware or not,

have developed a pure sub-symbolic learning approach. They enjoy the advantages of robust distributed representation of concept entities, but struggle with explicit reasoning with the relations among entities by discrete symbolic representations [371]. Nevertheless, studies in cognition suggest that our perception works at multiple levels of semantic abstraction [257], intertwined with logical reasoning through manipulation of symbolic knowledge/concepts [133]. For example, after recognizing many `utensil`s from Fig. 7.1, we know the scene is more likely a `kitchen`, rather than a `bathroom` or `gym`. This judgement comes as a result of reasoning with some abstract knowledge, such as "*utensils typically appear in the kitchen*" and "*utensils are seldom seen in the bathroom*," which are generalized from our daily experience. The judgement of the scene type may become a belief and in turn cause reallocation of our visual attention [135], hence driving us to find out more relevant details, such as small `fork`s.

Filling the gaps identified above calls for a fundamental paradigm shift: **i)** moving away from pixel-wise 'flat' classification towards semantic structure-aware parsing; and **ii)** moving away from the extreme of pure distributed representation learning towards an ambitious hybrid which combines both powerful sub-symbolic learning and principled symbolic reasoning. To embrace this change, we develop LogicSeg, a structured visual parser which exploits neural computing and symbolic logic in a neural-symbolic framework for holistic visual semantic learning and reasoning. In particular, given a set of hierarchically-organized semantic concepts as background knowledge and parsing target, we first use *first-order logic*, a powerful declarative language, to comprehensively specify relations among semantic classes. After *fuzzy logic* based relaxation, the logical formulae of hierarchy constraints can be grounded on data. During training, each logical constraint is converted into a differentiable loss function for gradient descent optimization. During inference, the logical constraints are involved into an iterative process, and calculated in matrix form. This not only

ensures the observance of the compositional semantic structure but also binds logic reasoning into network feed-forward prediction.

By accommodating logic-based symbolic rules into network training and inference, our LogicSeg **i)** blends statistical learning with symbolic reasoning, **ii)** obtains better performance, and **iii)** guarantees its parsing behavior compliant with the logically specified symbolic knowledge. We also remark that our study is relevant to a field of research called *neural-symbolic computing* (NSC) [84, 136, 324]. With the promise of integrating two critical cognitive abilities [298]: inductive learning (*i.e.*, the ability to learn general principles from experience) and deductive reasoning (*i.e.*, the ability to draw logical conclusions from what has been learned), NSC has long been a multi-disciplinary research focus and shown superiority in certain application scenarios, such as program generation [299, 233, 240], and question answering [303, 362]. This work unlocks the potential of NSC in visual semantic parsing – a fundamental, challenging, and large-scale vision task.

LogicSeg is a principled framework. It is fully compatible with existing segmentation network architectures, with only minor modification to the classification head and a plug-and-play logic-induced inference module. We perform experiments on four datasets covering wide application scenarios, including automated-driving (Mapillary Vistas 2.0 [228], Cityscapes [49]), object-centric (Pascal-Part [38]), and daily (ADE-20K [390]) scenes. Experimental results show that, on the top of various segmentation models (*i.e.*, DeepLabV3+ [32], Mask2Former [41]) and backbones (*i.e.*, ResNet-101 [106], Swin-T [194]), LogicSeg yields solid performance gains (**1.12**%-**3.29**% mIoU) and suppresses prior structured alternatives. The strong generalization and promising performance of LogicSeg evidence the great potential of integrating symbolic reasoning and sub-symbolic learning in machine perception.

7.2 Methodology

Task Setup and Notations. In this work we are interested in structured visual parsing [167] – a more challenging yet realistic setting for semantic segmentation – where both semantic concepts and their relations are considered in a form of a tree-shaped class hierarchy $\mathcal{T} = \langle \mathcal{V}, \mathcal{E} \rangle$. The node set $\mathcal{V} = \cup_{l=1}^{L} \mathcal{V}_l$ represents the classes/concepts at L abstraction levels. For instance in Fig. 7.2(a), the leaf nodes $\mathcal{V}_1$ are the finest classes (*e.g.*, `chair`, `pot`), while the internal nodes are higher-level concepts (*e.g.*, `furniture`, `utensil`), and the roots $\mathcal{V}_L$ are the most abstract ones (*e.g.*, `object`). The edge set $\mathcal{E}$ encodes relational knowledge among classes. For example, a directed edge $u \rightarrow v \in \mathcal{E}$ denotes a *part-of* relation between classes $u, v \in \mathcal{V}$ in *adjacent* levels (*e.g.*, `utensil`→`pot`). Given $\mathcal{T}$, the target goal is to assign each pixel a *valid* root-to-leaf path in $\mathcal{T}$. For instance, associating a pixel with `object`→`utensil`→`pot` is valid, yet with `object`→ `furniture`→`pot` is *invalid*. Thus standard semantic segmentation can be viewed as a specific case of such structured setting — only assigning pixels with one single class label from the leaf nodes $\mathcal{V}_1$ without considering the hierarchy.

Algorithmic Overview. LogicSeg is a unified, neural-logic learning and reasoning model for visual parsing, supported by large-scale data and the structured symbolic knowledge $\mathcal{T}$.

- From the ***neural*** aspect, LogicSeg is *model-agnostic*. After dense feature extraction, its classification head outputs a total of $|\mathcal{V}|$ *sigmoid*-normalized scores, *i.e.*, $\boldsymbol{s} \in [0,1]^{|\mathcal{V}|}$, over all the classes $\mathcal{V}$ for each pixel, like [167]. Here $|\cdot|$ counts its elements. A set of logic rules, derived from $\mathcal{T}$, are injected into network training and inference.
- From the ***logic*** aspect, LogicSeg uses *first-order logic* to express the complex and abstract relational knowledge in $\mathcal{T}$. The network is learnt as approximation of

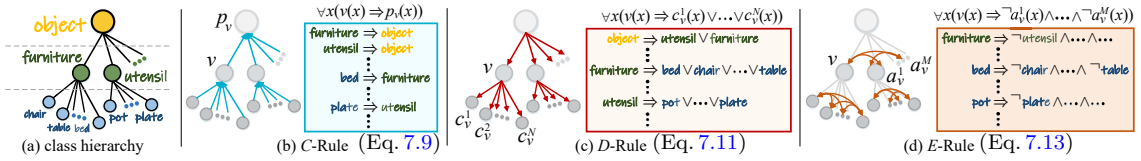


Figure 7.2 : Illustration of the (a) class hierarchy $\mathcal{T}$, and (b-d) abstract relational knowledge specified by first-order logic formulae (§7.2.1).

logic predicates by following the logical specifications. Once trained, it conducts iterative reasoning on the basis of logic rules.

After introducing our logic based visual relational knowledge representation (§7.2.1), we will elaborate on our logic-induced network training (§7.2.2) and inference (§7.2.3) strategies.

7.2.1 Parsing Visual Semantics with Logic Rules

We formalize our target task — *learning and reasoning visual semantics with logic* — as a triple $\langle \mathcal{T}, \mathcal{X}, \Pi \rangle$. $\mathcal{X}$ is a data collection, *i.e.*, $\mathcal{X} = \{(x_k, \boldsymbol{y}_k)\}_{k=1}^{K}$, where x_k is a pixel data point, and $\boldsymbol{y}_k \in \{0,1\}^{|\mathcal{V}|}$ is its groundtruth symbolic description in terms of the semantic hierarchy $\mathcal{T}$. Π is a set of hierarchy rules declaratively expressed by *first-order logic*, containing **i)** *constants*, *e.g.*, pixel samples $x_1, x_2, \cdots$; **ii)** *variables* ranging over constants, *e.g.*, x; and **iii)** *unary predicates*, one for each class $v \in \mathcal{V}$, denote the semantics of variables and return *true* and *false*, *e.g.*, $\texttt{bed}(x) = true$ states the fact that pixel x belongs to a bed. A logic rule/formula is a sequence of finite predicates with *connectives* (*i.e.*, $\wedge$, $\vee$, $\neg$, $\Rightarrow$) and *quantifiers* (*i.e.*, $\forall$, $\exists$), and organized in *prenex* form in our case. Concretely, Π is composed of three types of rules, *i.e.*, *composition*, *decomposition*, and *exclusion*, for comprehensively describing the structured symbolic knowledge $\mathcal{T}$.

- ***Composition Rule*** (C-rule) expresses our knowledge about the *composition* relations between semantic concepts, such as "*bed and chair are (subclasses of)*

furniture," in a form of:

$$\begin{aligned} \forall x(\texttt{bed}(x) \ &\Rightarrow \texttt{furniture}(x)), \\ \forall x(\texttt{chair}(x) &\Rightarrow \texttt{furniture}(x)), \end{aligned} \tag{7.1}$$

where `bed`, `chair`, `furniture` are predicates, and '$\phi \Rightarrow \varphi$' indicates φ is a logical consequence of antecedence ϕ.

Definition 7.2.1.1 (C-rule)**.** *If one class is labeled true, its superclass should be labeled true (Fig. 7.2(b)):*

$$\forall x(v(x) \Rightarrow p_v(x)), \tag{7.2}$$

where p_v is the parent node of v in $\mathcal{T}$, i.e., *$p_v \rightarrow v \in \mathcal{E}$ (the tree structure of $\mathcal{T}$ restricts each class to possess only one superclass). C-rule generalizes the famous tree-property [18, 90].*

• ***Decomposition Rule*** (D-rule) states our knowledge about the *decomposition* relations among semantic concepts, such as "*furniture is the superclass of bed, chair, $\cdots$, table*," via:

$$\begin{aligned} \forall x(\texttt{furniture}(x) \Rightarrow &\texttt{bed}(x) \vee \texttt{chair}(x) \vee \\ &\cdots \vee \texttt{tabel}(x)). \end{aligned} \tag{7.3}$$

Definition 7.2.1.2 (D-rule)**.** *If one class is labeled true, at least one of its subclasses should be labeled true (Fig. 7.2(c)):*

$$\forall x(v(x) \Rightarrow c_v^1(x) \vee c_v^2(x) \vee \cdots \vee c_v^N(x)), \tag{7.4}$$

where $c_v^n \in \mathcal{C}_v$ are all the child nodes of v in $\mathcal{T}$, i.e., *$v \rightarrow c_v^n \in \mathcal{E}$. C-rule and D-rule are not equivalent. For instance in Eq. 7.1, $\texttt{bed}(x)$ is sufficient but not necessary for $\texttt{furniture}(x)$: given the fact "x is furniture", we cannot conclude "x is bed".*

• ***Exclusion Rule*** (E-rule) specifies our knowledge about *mutual exclusion* relations between *sibling* concepts, such as "*a bed cannot be at the same time a chair*," in a form of:

$$\forall x(\texttt{bed}(x) \Rightarrow \neg\texttt{chair}(x)). \tag{7.5}$$

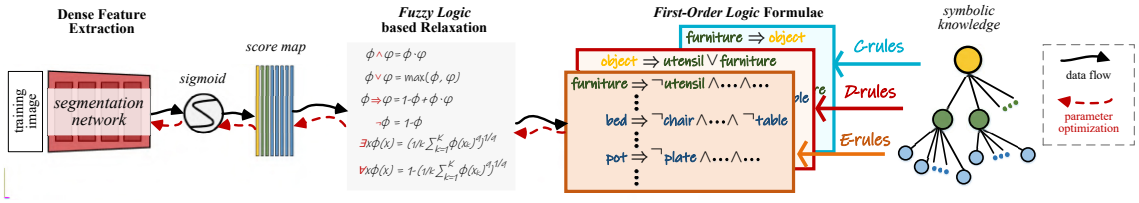


Figure 7.3 : Illustration of our logic-induced network training (§7.2.2). For clarity, the pixel-wise binary cross-entropy loss is omitted.

Definition 7.2.1.3 (E-rule)**.** *If one class is labeled true, all its sibling classes should be labeled false (Fig. 7.2(d)):*

$$\forall x(v(x) \Rightarrow \neg a_v^1(x) \wedge \neg a_v^2(x) \wedge \cdots \wedge \neg a_v^M(x)), \tag{7.6}$$

where $a_v^m \in \mathcal{A}_v$ are all the peer nodes of v in $\mathcal{T}$. Note that E-rule is ignored by many hierarchy-aware algorithms [167, 15, 333].

7.2.2 Logic-Induced Training

So far, we shown the logic rules Π provide LogicSeg a flexible language for comprehensively expressing the complex *meronymy* and *exclusion* relations among symbolic concepts in the hierarchy $\mathcal{T}$. Unfortunately, these rules are logic formulae working with variables (assuming a boolean value), and non-differentiable logic symbols (*e.g.*, $\forall$, $\Rightarrow$). This prevents the integration with end-to-end network learning. Inspired by [300, 9], a *fuzzy logic* based *grounding* process is adopted to interpret logic formulae as differentiable fuzzy relations on real numbers for neural computing (Fig. 7.3).

Fuzzy relaxation. Fuzzy logic is a form of soft probabilistic logic. It deals with reasoning that is approximate instead of fixed and exact; variables have a truth degree that ranges in $[0, 1]$: zero and one meaning that the variable is *false* and *true* with certainty, respectively [231]. Hence we can ground predicates onto segmentation network outputs. For instance, given a pixel sample x, corresponding network

prediction score w.r.t.class *bed* is a grounded predicate w.r.t.$\texttt{bed}(x)$. Logical connectives, *i.e.*, $\wedge, \vee, \neg, \Rightarrow$ are approximated with *fuzzy operators*, *i.e.*, *t-norm*, *t-conorm*, *fuzzy negation*, and *fuzzy implication*. As suggested by [300], we adopt the operators in *Goguen fuzzy logic* [101] and *Gödel fuzzy logic* [70]:

$$\begin{aligned} \phi \wedge \varphi = \phi \cdot \varphi, \qquad \phi \vee \varphi = \texttt{max}(\phi, \varphi), \\ \neg\phi = 1 - \phi, \qquad \phi \Rightarrow \varphi = 1 - \phi + \phi \cdot \varphi. \end{aligned} \tag{7.7}$$

The existential quantifier $\exists$ and universal quantifier $\forall$ are approximated in a form of generalized mean:

$$\begin{aligned} \exists x \phi(x) = \ & (\tfrac{1}{K}\textstyle\sum_{k=1}^{K} \phi(x_k)^q)^{\frac{1}{q}}, \\ \forall x \phi(x) = \ & 1 - (\tfrac{1}{K}\textstyle\sum_{k=1}^{K} (1-\phi(x_k))^q)^{\frac{1}{q}}, \end{aligned} \tag{7.8}$$

where $q \in \mathbb{Z}$. Please refer to [300, 9] for detailed discussion regarding the rationale behind such approximation of $\exists$ and $\forall$.

Logic Loss. With fuzzy relaxation, we are ready to convert our first-order logic rules Π into loss functions.

• ***C-rule Loss.*** For a non-root node $v \in \mathcal{V}/\mathcal{V}_L$, its corresponding C-rule (*cf.* Eq. 7.2) is grounded as:

$$\mathcal{G}_C(v) = 1 - \left(\tfrac{1}{K}\textstyle\sum_{k=1}^{K} (\boldsymbol{s}_k[v] - \boldsymbol{s}_k[v] \cdot \boldsymbol{s}_k[p_v])^q\right)^{\frac{1}{q}}, \tag{7.9}$$

where $\boldsymbol{s}_k[v] \in [0, 1]$ refers to the score (confidence) of x_k for class v. Then the C-rule based training objective is given as:

$$\mathcal{L}_C = \tfrac{1}{|\mathcal{V}|-|\mathcal{V}_L|}\textstyle\sum_{v \in \mathcal{V}/\mathcal{V}_L} 1 - \mathcal{G}_C(v). \tag{7.10}$$

• ***D-rule Loss.*** For a non-leaf node $v \in \mathcal{V}/\mathcal{V}_1$, its corresponding D-rule (*cf.* Eq. 7.4) is grounded as:

$$\mathcal{G}_D(v) = 1 - \left(\tfrac{1}{K}\sum_{k=1}^{K} (\boldsymbol{s}_k[v] - \boldsymbol{s}_k[v] \cdot \texttt{max}(\{\boldsymbol{s}_k[c_v^n]\}_n))^q\right)^{\frac{1}{q}}. \tag{7.11}$$

Similarly, our D-rule loss is given as:

$$\mathcal{L}_D = \tfrac{1}{|\mathcal{V}|-|\mathcal{V}_1|}\textstyle\sum_{v \in \mathcal{V}/\mathcal{V}_1} 1 - \mathcal{G}_D(v). \tag{7.12}$$

• ***E-rule Loss.*** During the grounding of E-rule (*cf.* Eq. 7.6), we first translate such *one-vs-all* exclusion statement to a semantically equivalent expression, *i.e.*, the aggregation of multiple *one-vs-one* exclusion ($\{(v(x) \Rightarrow \neg a_v^1(x)), \cdots, \{(v(x) \Rightarrow \neg a_v^M(x))\}$). Adopting such translation is to avoid the *sorites paradox*, *i.e.*, a long chain of only slightly unreliable deductions can be very unreliable [94] (*e.g.*, $0.9^{10} \approx 0.34$), happened during approximating a series of $\wedge$. Then, for each node $v \in \mathcal{V}$, its corresponding E-rule is grounded as:

$$\mathcal{G}_E(v) = 1 - \frac{1}{M}\sum_{m=1}^{M}\left(\frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{s}_k[v] \cdot \boldsymbol{s}_k[a_v^m])^q\right)^{\frac{1}{q}}. \tag{7.13}$$

Similarly, our E-rule loss is given as:

$$\mathcal{L}_E = \frac{1}{|\mathcal{V}|}\sum_{v \in \mathcal{V}} 1 - \mathcal{G}_E(v). \tag{7.14}$$

In this way, it is possible to backpropagate the gradient from the logic loss into the network. The network is essentially learned as neural predicates obeying the logical constraints. It is worth mentioning that, due to large-scale training, it is infeasible to compute the full semantics of $\forall$; batch-training can be viewed as sampling based approximation [300]. Our overall training target is organized as:

$$\mathcal{L} = \alpha(\mathcal{L}_C + \mathcal{L}_D + \mathcal{L}_E) + \frac{1}{K}\sum_{k=1}^{K}\mathcal{L}_{\text{BCE}}(\boldsymbol{s}_k, \boldsymbol{y}_k). \tag{7.15}$$

Here $\boldsymbol{y} \in \{0,1\}^{|\mathcal{V}|}$ is the groundtruth, $\mathcal{L}_{\text{BCE}}$ is the binary cross-entropy loss, and the coefficient is empirically set as $\alpha = 0.2$.

7.2.3 Logic-Induced Inference

We just showed that LogicSeg can approximate the predicates by integrating symbolic logic constraints into large-scale network training. However, during inference, there is no explicit way to ensure the alignment between the class hierarchy $\mathcal{T}$ and network prediction, neither sound reasoning with the logic rules Π. We thus put forward *logic-induced reasoning* (Fig. 7.4), where the logic rules Π are encapsulated into an iterative optimization process. Such process is non-learnable, based on only

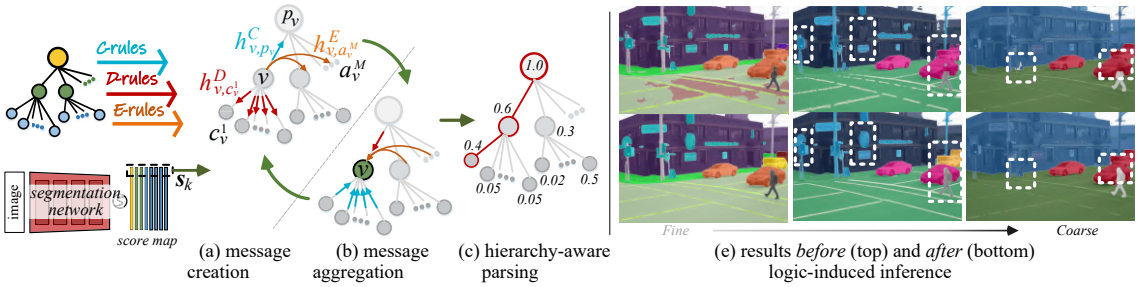


Figure 7.4 : Illustration of our logic-induced inference (§7.2.3). (a-b) Iterative reasoning is made by exchanging and absorbing messages between nodes, following the logic rules Π. For clarity, we only show the message creation (Eq. 7.16) and aggregation (Eq. 7.17) stages for one single node. (c) Structured parsing (Eq. 7.18) is conducted by selecting the top-scoring path $\mathcal{P}^*$ (highlighted in red) after logic-guided iterative reasoning. (d) With logic-induced inference, LogicSeg is able to generate more accurate and hierarchy-compliant predictions.

matrix operations and thus can be seamlessly embedded into network feed-forward inference, yielding an elegant yet compact neural-logic visual parser.

Our solution is built upon the classic *message passing* algorithm which is to estimate the marginal likelihood for a given tree structure by *iteratively* exchanging messages between nodes. Specifically, at each iteration, for each pixel sample x_k, node $v \in \mathcal{V}$ sends different types of messages to different neighboring nodes, according to the logic rules Π:

$$\begin{aligned}
C\text{-message: } h^C_{v,p_v} &= v(x_k) \Rightarrow p_v(x_k) \\
&= 1 - \boldsymbol{s}_k[v] + \boldsymbol{s}_k[v] \cdot \boldsymbol{s}_k[p_v], \\
D\text{-message: } h^D_{v,c^n_v} &= v(x_k) \Rightarrow c^1_v(x_k) \vee \cdots \vee c^N_v(x_k) \\
&= 1 - \boldsymbol{s}_k[v] + \boldsymbol{s}_k[v] \cdot \texttt{max}(\{\boldsymbol{s}_k[c^n_v]\}_n), \\
E\text{-message: } h^E_{v,a^m_v} &= -1 \cdot \big(v(x_k) \Rightarrow \neg a^1_v(x_k) \wedge \cdots \wedge \neg a^M_v(x_k)\big) \\
&= -\left(1 - \tfrac{1}{M} \textstyle\sum_{m=1}^{M} \boldsymbol{s}_k[v] \cdot \boldsymbol{s}_k[a^m_v]\right).
\end{aligned} \tag{7.16}$$

Node v is updated by aggregating the received messages:

$$\begin{aligned}\boldsymbol{s}_k[v] \leftarrow \boldsymbol{s}_k[v] + \tfrac{1}{N}\textstyle\sum_{c_v^n \in \mathcal{C}_v} \boldsymbol{s}_k[c_v^n] \cdot h^C_{c_v^n, v} + \boldsymbol{s}_k[p_v] \cdot h^D_{p_v, v} \\ + \tfrac{1}{M}\textstyle\sum_{a_v^m \in \mathcal{A}_v} \boldsymbol{s}_k[a_v^m] \cdot h^E_{a_v^m, v}.\end{aligned} \tag{7.17}$$

Each message (*cf.* Eq. 7.16) accounts for the certainty degree that v satisfies the corresponding logic rule (*cf.* §7.2.1) when being grounded on pixel data point x_k, with fuzzy logic based approximation (*cf.* §7.2.2). Intuitively, the more certainty a node meets the logic rules, the more message it can propagate to other nodes. Note that, v creates a *negative* message h^E_{v,a_v^m} to "suppress" other peer nodes due to their exclusion relations. In Eq. 7.17, the received messages are weighted by the confidence of the source nodes themselves – the grounded predicates, *i.e.*, $\boldsymbol{s}_k[c_v^n]$, $\boldsymbol{s}_k[p_v]$, and $\boldsymbol{s}_k[a_v^m]$. After each iteration, the score vector $\boldsymbol{s}_k$ is *softmax*-normalized per hierarchy level. Finally, each pixel x_k is associated with the top-scoring *root-to-leaf* path in the hierarchy $\mathcal{T}$ (red path in Fig. 7.4(c)):

$$\mathcal{P}^* = \{v_1^*, \cdots, v_L^*\} = \arg\max_{\mathcal{P} \subset \mathcal{T}} \textstyle\sum_{v^{\mathcal{P}} \in \mathcal{P}} \boldsymbol{s}_k[v^{\mathcal{P}}], \tag{7.18}$$

where $\mathcal{P} = \{v_1^{\mathcal{P}}, \cdots, v_L^{\mathcal{P}}\} \subset \mathcal{T}$ indicates a feasible root-to-leaf path in $\mathcal{T}$, *i.e.*, $\forall v_l^{\mathcal{P}}, v_{l-1}^{\mathcal{P}} \in \mathcal{P} \Rightarrow v_l^{\mathcal{P}} \rightarrow v_{l-1}^{P} \in \mathcal{E}$. It is easy to find that all the logic-induced inference steps (*cf.* Eq. 7.16-7.18) can be formulated in *matrix* form with only a couple of matrix multiplications (see corresponding pseudo-code in the supplementary). Hence it is efficient on GPU and can be straightforward injected into the network, making LogicSeg a fully-integrated neural-logic machine. In practice, 2-iteration message passing is enough for robust prediction. Through logic-induced reasoning (*cf.* Eq. 7.17) and hierarchy-aware parsing (*cf.* Eq. 7.18), LogicSeg is able to **i)** obtain *improved performance*, and **ii)** guarantee the parsing results to *respect the hierarchy* $\mathcal{T}$, with **iii)** only *negligible speed delay* (about 3.8%). See §7.3.4 for related experiments.

7.3 Experiment

7.3.1 Experimental Setup

Datasets. We conduct extensive experiments on four datasets, *i.e.*, Mapillary Vistas 2.0 [228], Cityscapes [49], Pascal-Part-108 [38], and ADE20K [390]. The four datasets are selected to cover the rich application scenarios of semantic segmentation, including urban street segmentation for automated driving (*i.e.*, [228, 49]), object part parsing (*i.e.*, [38]), and fine-grained understanding of daily scenes (*i.e.*, [390]), so as to comprehensively examine the utility of our algorithm.

- **Mapillary Vistas 2.0** is a large-scale urban scene dataset. It contains $18,000$/ $2,000$/$5,000$ images for `train`/`val`/ `test`. A three-level semantic hierarchy, covering 4/16/ 124 concepts, is officially provided for dense annotation.
- **Cityscapes** has $2,975$/500/$1,524$ finely annotated, urban street images for `train`/ `val`/`test`. The label hierarchy consists of 19 fine-gained concepts and 6 super-classes.
- **Pascal-Part-108** is the largest object part parsing dataset. It consists of $4,998$/ $5,105$ images for `train`/`test`. To establish the class hierarchy, we group 108 part-level labels into 20 object-level categories, as in [221, 96, 382, 273].
- **ADE20K** is a large-scale generic scene parsing dataset. It is divided into $20,210$/ $2,000$/$3,000$ images for `train`/`val`/`test`. It provides pixel-wise annotations for 150 fine-grained semantic classes, from which a three-layer label hierarchy (with 3/14/150 concepts) can be derived.

Evaluation Metric. We adopt the standard metric, mean intersection-over-union (mIoU), for evaluation. For detailed performance analysis, the score is reported for each hierarchy level l (denoted as mIoU^l), as suggested by [326, 167].

Base Models and Competitors. To demonstrate our wide benefit, we approach our algorithm on two famous segmentation architectures, *i.e.*, DeepLabV3+ [32]

and Mask2Former [41], with ResNet-101 [106] and Swin-T [194] backbones. For performance comparison, we involve several hierarchy-aware segmentation models [167, 221, 273], and view HSSN [167] as our major rival as it is a general framework that reports strong results over several datasets, instead of the others that are dedicated to specific dataset(s) or task setup(s). For comprehensive evaluations, we include a group of previous hierarchy-agnostic segmentation algorithms [365, 11, 343, 385, 279, 345], whose segmentation results on coarse-grained semantics are obtained by merging the predictions of the corresponding subclasses.

Training. For the sake of fairness, we follow the standard training setup in [29, 365, 42, 98, 177]. In particular, we train 240K/80K iterations for Mapillary Vistas 2.0/Cityscapes, with batch size 8 and crop size 512×1024; 60K/160K iterations for Pascal-Part-108/ADE20K, with batch size 16 and crop size 512×512. For data augmentation, the images are horizontally flipped and scaled with a ratio between 0.5 and 2.0 at random. For network optimization, SGD (with initial learning rate 1e-2, momentum 0.9, and weight decay 1e-4) and Adam (with initial learning rate 6e-5 and weight decay 0.01) are respectively used for CNN-based and neural attention-based models, where the learning rate is scheduled by the polynomial annealing rule. For network initialization, ImageNet [56] pre-trained weights are pre-loaded.

Testing. For Mapillary Vistas 2.0 and Cityscapes, we keep the original image aspect ratio but resize the short edge to 1024. Sliding window inference with the identical window shape as the training size is adopted to save memory. For ADE20K and Pascal-Part-108, the short edge is resized to 512 so as to enable one-time inference for the whole image. As in [117, 42, 365, 130], performance of all the models is reported at multiple scales ($\{0.5, 0.75, 1.0, 1.25, 1.5, 1.75\}$) with horizontal flipping.

Hyperparameters. We set $\alpha = 0.2$ for the loss coefficient (*cf.* Eq. 7.15), and $q = 5$ for logic quantifier approximation (*cf.* Eq. 7.8), as suggested by [9]. For network

Method	Backbone	mIoU^3↑	mIoU^2↑	mIoU^1↑
Seamless [249]	ResNet-101	84.23	70.24	38.82
OCRNet [365]	HRNet-W48	84.19	69.82	38.26
HMSANet [330]	HRNet-W48	84.63	70.71	39.53
Mask2Former [41]	Swin-S	88.81	74.98	43.49
Hssn [167]	ResNet-101	85.27	71.40	40.16
Hssn [167]	Swin-S	90.02	75.81	43.97
DeepLabV3+ [32]	ResNet-101	81.86	68.17	37.43
+ **LogicSeg**		**85.51** ↑3.65	**71.69** ↑3.42	**40.72** ↑3.29
MaskFormer [42]	Swin-S	87.93	73.88	42.16
+ **LogicSeg**		**90.35** ↑2.42	**76.61** ↑2.73	**45.12** ↑2.96

Table 7.1 : **Urban scene parsing results** (§7.3.2) on Mapillary Vistas 2.0 [228] `val` with a three-level label hierarchy of 4/16/124 concepts.

inference, we find 2 iterations of message passing are enough.

7.3.2 Quantitative Comparison Result

Mapillary Vistas 2.0 [228] `val`. From Table 7.1 we can observe that our approach provides notable performance gains over the baselines. For example, our algorithm promotes classic DeepLabV3+ [32] by **3.65**%/**3.42**%/**3.29**% over the three semantic levels. On top of MaskFormer [42], our algorithm further lifts the scores by **2.42**%/**2.73**%/**2.96**%, suppressing previous hierarchy-agnostic models, as well as Hssn [167] – a newly proposed hierarchy-aware segmentation model.

Cityscapes [49] `val`. Table 7.2 confirms again our compelling performance in challenging urban street scenes and wide benefits for different segmentation models, *i.e.*, **1.21**%/**1.12**% over DeepLabV3+, and **1.35**%/**1.28**% over MaskFormer. Though both encoding concept structures into segmentation, our algorithm greatly outperforms Hssn, suggesting the superiority of our logic reasoning framework.

Method	Backbone	mIoU^2↑	mIoU^1↑
DANet [76]	ResNet-101	91.83	81.52
CCNet [117]	ResNet-101	91.70	81.08
SETR [385]	ViT-L	92.86	82.75
SegMentor [279]	ViT-L	91.79	81.30
UperNet [343]	Swin-S	91.92	81.79
Mask2Former [41]	Swin-S	93.68	83.62
SegFormer [345]	MiT-B4	93.81	83.90
HSSN [167]	ResNet-101	93.31	83.02
HSSN [167]	Swin-S	94.39	83.74
DeepLabV3+ [32]	ResNet-101	92.16	82.08
+ **LogicSeg**		**93.37** ↑1.21	**83.20** ↑1.12
MaskFormer [42]	Swin-S	92.96	82.57
+ **LogicSeg**		**94.31** ↑1.35	**83.85** ↑1.28

Table 7.2 : **Urban scene parsing results** (§7.3.2) on Cityscapes [49] `val` with a two-level label hierarchy of 6/19 concepts.

Pascal-Part-108 [38] `test`**.** As illustrated by Table 7.3, our algorithms yields remarkable performance on explaining the compositionality of object-centric semantic structures. Specifically, our algorithm not only consistently boosts the performance of base segmentation models [32, 41], but also defeats two outstanding hierarchy-agnostic competitors [11, 273] as well as three structured alternatives [221, 273, 167].

ADE20K [390] `val`**.** Table 7.4 presents our parsing results in general scenes. With a relatively conservative baseline, *i.e.*, DeepLabV3+ [32], our algorithm earn **79.60**%, **59.04**%, and **48.46**%, in terms of mIoU^1, mIoU^2, and mIoU^3 respectively. It delivers a solid overtaking against Mask2Former [41], which is built upon a more advanced architecture. When applied to MaskFormer [42], our algorithm achieves **82.45**%/ **62.44**%/**52.82**%, pushing forward the state-of-the-art.

Method	Backbone	mIoU^2↑	mIoU^1↑
SegNet [11]	ResNet-101	59.81	36.42
FCN-8s [196]	ResNet-101	62.26	38.62
BSANet [382]	ResNet-101	69.37	47.36
GMNet [221]	ResNet-101	69.28	47.21
FLOAT [273]	ResNet-101	70.03	48.08
HSSN [167]	ResNet-101	72.91	48.32
HSSN [167]	Swin-S	77.01	54.79
DeepLabV3+ [32]	ResNet-101	70.86	46.54
+ **LogicSeg**		**73.68** ↑2.82	**49.13** ↑2.69
MaskFormer [42]	Swin-S	75.78	53.07
+ **LogicSeg**		**77.92** ↑2.14	**55.53** ↑2.46

Table 7.3 : **Object part parsing results** (§7.3.2) on PASCAL-Part-108 [38] `test` with a two-level label hierarchy of 20/108 concepts.

Method	Backbone	mIoU^3↑	mIoU^2↑	mIoU^1↑
OCRNet [365]	HRNet-W48	76.33	55.76	44.92
UperNet [343]	Swin-S	78.90	59.17	49.47
SegMentor [279]	ViT-S	77.32	57.18	46.82
K-Net [374]	Swin-S	79.11	59.38	49.95
SegFormer [345]	MiT-B4	79.85	60.24	51.08
Mask2Former [41]	Swin-S	80.46	61.15	52.43
HSSN [167]	ResNet-101	79.23	58.52	47.69
HSSN [167]	Swin-S	82.59	62.56	52.37
DeepLabV3+ [32]	ResNet-101	77.24	56.87	46.43
+ **LogicSeg**		**79.60** ↑2.36	**59.04** ↑2.17	**48.46** ↑2.03
MaskFormer [42]	Swin-S	79.89	60.32	51.04
+ **LogicSeg**		**82.45** ↑2.56	**62.44** ↑2.12	**52.82** ↑1.78

Table 7.4 : **Generic scene parsing results** (§7.3.2) on ADE20K [390] `val` with a three-level label hierarchy of 3/14/150 concepts.

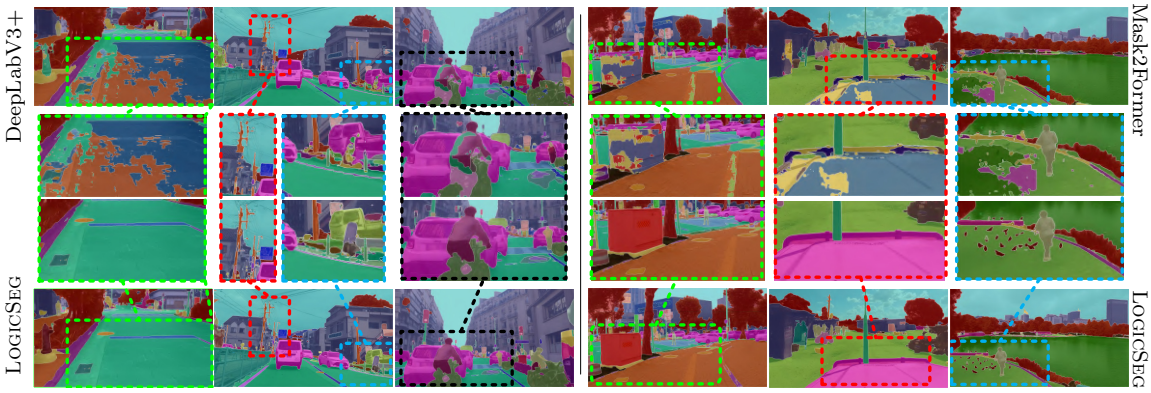


Figure 7.5 : **Visual results** (§7.3.3) on Mapillary Vistas 2.0 [228]. *Left*: DeepLabV3+ [32] *vs* LogicSeg; *Right*: Mask2Former [41] *vs* LogicSeg.

Taking together, our extensive benchmarking results provide solid evidence that our algorithm successfully unlocks the power of logic reasoning in large-scale visual parsing, and owns broad applicability across various task scenarios, segmentation architectures, and backbone networks.

7.3.3 Qualitative Comparison Result

Fig. 7.5 visualizes qualitative comparisons of LogicSeg against DeepLabV3+ [32] (*left*) and Mask2Former [41] (*right*) on Mapillary Vistas 2.0 dataset [228]. As seen, with the help of symbolic reasoning, LogicSeg can generate higher-quality predictions even in challenging scenarios.

7.3.4 Diagnostic Experiment

For thorough evaluation, we perform a series of ablative studies on Mapillary Vistas 2.0 [228] `val`. All variants are based on DeepLabV3+ [32] with ResNet-101 [106] backbone.

Logic-Induced Training. We first study the effectiveness of our logic-induced training strategy (§7.2.2) in Table 7.5a. 1^{st} row reports the results of our baseline model – DeepLabV3+. 2^{nd}, 3^{rd}, and 4^{th} rows respectively list the scores obtained by

$\mathcal{L}_C$ Eq. 7.9	$\mathcal{L}_D$ Eq. 7.11	$\mathcal{L}_E$ Eq. 7.13	mIoU^3↑	mIoU^2↑ (multi scale)	mIoU^1↑	Training Speed (min/epoch)
			81.86	68.17	37.43	45.62
✓			83.56	69.74	38.71	46.35 +1.60%
	✓		84.08	69.97	38.98	46.13 +1.12%
		✓	83.42	69.60	38.43	46.72 +2.41%
✓	✓	✓	**85.51**	**71.69**	**40.72**	47.67 +4.51%

(a) logic loss (§7.2.2)

# Iter.	mIoU^3↑	mIoU^2↑ (multi scale)	mIoU^1↑	Inference Speed (fps)
0	84.62	70.95	40.18	3.44
1	85.23	71.46	40.56	3.37 -2.03%
2	**85.51**	**71.69**	**40.72**	3.31 -3.78%
3	84.84	71.12	40.29	3.25 -5.52%
4	84.49	70.84	40.03	3.20 -6.98%

(b) iteration of message passing (§7.2.3)

q	mIoU^3↑	mIoU^2↑ (multi scale)	mIoU^1↑
1	83.83	70.22	38.77
3	84.65	71.15	40.09
5	**85.51**	**71.69**	**40.72**
8	84.47	71.03	39.74
10	83.52	69.88	38.25

(c) aggregation coefficient for $\forall$ (Eq. 7.8)

Table 7.5 : **Ablative studies** on Mapillary Vistas 2.0 [228] `val` (§7.3.4). All variants are based on DeepLabV3+ [32] with ResNet-101 [106] backbone.

individually adding our C-rule loss $\mathcal{L}_C$ (*cf.* Eq. 7.10), D-rule loss $\mathcal{L}_D$ (*cf.* Eq. 7.12), and E-rule loss $\mathcal{L}_E$ (*cf.* Eq. 7.14). The last row gives the performance of our full loss $\mathcal{L}$ (*cf.* Eq. 7.15). We can find that: **i)** Taking each of our logic losses into consideration can provide consistent performance gains. This reveals that different logic rules can describe different properties of semantic structure and verify that the segmentation model can indeed benefit from our proposed logic losses. **ii)** Combing all three logic losses together results in the best performance. This suggests that our logic rules provide a comprehensive description of the relational knowledge in the semantic hierarchy $\mathcal{T}$, and supports our core idea that exploiting symbolic knowledge is crucial for visual semantic interpretation and can boost sub-symbolic learning.

Training Speed. As shown in the last column of Table 7.5a, our logic-induced training regime causes a trivial delay (∼5.0%).

Logic-Induced Inference. We next investigate the impact of our logic-induced inference strategy (§7.2.3) in Table 7.5b. 1st row reports the results of network feed-forward output. The rest rows give the scores obtained with different iterations of message passing (*cf.* Eq. 7.17). These results demonstrate the efficacy of our strategy and the necessity of incorporating logic reasoning into network inference. We accordingly set 2-iteration as the default to pursue the best performance.

Inference Speed. We also report inference speed (fps) in Table 7.5b. As seen, our logic-induced inference strategy only slows the speed slightly during model deployment (∼3.8%).

Aggregation Coefficient. For the approximation of ∀ quantifier (*cf.* Eq. 7.8), we adopt the generalized mean for stable training, as suggested by [9]. Basically, a higher coefficient q renders ∀ a stronger focus on outliers. For completeness, the results with different values of q are reported in Table 7.5c.

7.4 Conclusion

The creation of intelligent systems that integrate the fundamental cognitive abilities of reasoning and learning has long been viewed as a core challenge for AI [298]. While the community recently witnessed great advances in high-level perception tasks such as visual semantic interpretation, top-leading solutions are purely driven by sub-symbolic learning, far from such effective integration. In this chapter, we represent an innovative and solid attempt towards closing this gap. By embedding symbolic logic into both network training and inference, a structured and powerful visual semantic parser is delivered. We hope this work can stimulate our community to rethink current *de facto*, sub-symbolic paradigm and investigate new methodologies, from the perspective of achieving a better understanding of human and machine intelligence.

Chapter 8

Neural-Logic Integration for HOI Detection: The LogicHOI Framework

In the previous chapter, we demonstrated how explicit hierarchical semantic knowledge could be injected into neural networks for visual parsing. This chapter extends this neuro-symbolic paradigm to a different facet of visual understanding, where commonsense knowledge regarding object affordances and proxemics is utilized to reason about the plausibility of human-object interactions.

8.1 Introduction

The main purpose of human-object interaction (HOI) detection is to interpret the intricate relationships between human and other objects within a given scene [360]. Rather than traditional visual perception tasks that focus on the *recognition* of objects or individual actions, HOI detection places a greater emphasis on *reasoning* over entities [252]. With the prosperity of Transformer-based object detector (*e.g.*, DETR [24]), current top-leading solutions for HOI detection typically adopt a Transformer [302]-based architecture. In these work, an interaction decoder receives proposed human-object pairs [36, 141, 284, 402, 366, 142, 369, 391, 182, 189, 388] as inputs, and then infers the interactions happening between them. Though achieving promising performance, this proposal-then-classification paradigm suffers from several limitations: **first**, the human-object pairs are often raised by a proposal network composed of simple MLP layers [378, 252, 100, 308, 347, 392, 175, 296, 310, 81, 110, 174, 144, 173, 192, 368, 93] or simultaneously constructed during the detection of objects [141, 284, 402, 366, 142, 369, 391, 182], lacking comprehensive exploration

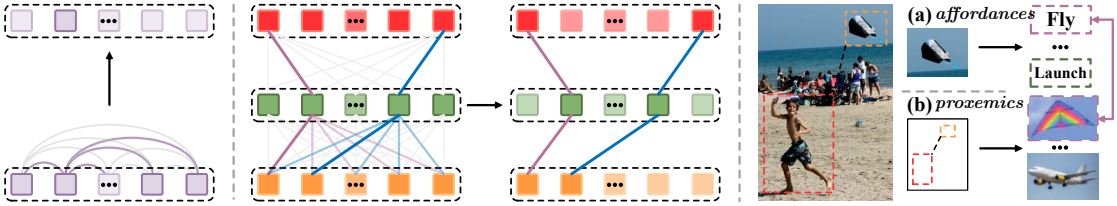


Figure 8.1 : **Left**: self-attention aggregates information across pre-composed interaction (▢) *queries*. **Middle**: in contrast, our proposed *triplet-reasoning* attention traverses over human (■), action (■), and object (■) *queries* to find plausible interactions. **Right**: logic-induced *affordances* and *proxemics* property learning.

on the potential combinations between humans and objects. **Second**, current solutions primarily optimize for known concepts, neglecting a vast number of feasible human-object combinations never encountered during training. This oversight leads to poor zero-shot generalization ability [113]. **Third**, existing approaches are purely neural-based. While effective at pattern recognition, they struggle with handling tasks that require advanced commonsense and spatial reasoning abilities.

To tackle above issues, we propose to formulate HOI detection through the integration of compositional learning and neural-logic computing. Concretely, compositionality reflects the computational paradigm of how human comprehend and create new concepts, wherein complex information encoded in large structures can be systematically understood by composing it from smaller, simpler structures [127, 238]. This principle aligns closely with the nature of HOI, which consists of humans, actions, and objects three base elements. However, while deep neural networks excel at learning distributed representation from raw data, they lack explicit mechanisms to represent and manipulate compositional structures such as casual dependencies or relational constructs [73, 53]. To bridge this gap, neural-logic computing [324, 84, 83] offers a promising avenue to unify the robust, flexible learning capabilities of neural networks, with the structured, interpretable, and compositional reasoning strengths

of symbolic systems.

In light of the analysis above, we present LogicHOI, which enjoys both the robust distributed representation of neural networks, as well as the compositional manipulation and logic reasoning abilities of symbolic systems [214, 54], to tackle the recognition of complex visual interactions. Specifically, to facilitate compositional learning, we decouple the triplet prediction into the detection of `human`, `action`, `object`, using three parallel branches. These detected elements are then used as the inputs to the interaction decoder. We revise the attention mechanism in the Transformer-based interaction decoder, which is originally designed to decode embeddings of pre-composed human-object pairs (Fig. 8.1: left). Instead, we adapt it to operate in a triplet manner over ⟨`human`, `action`, `object`⟩ three elements, leading to a *triplet-reasoning attention* (Fig. 8.1: middle). During decoding, the associations between entities involved in interactions are enhanced, while connections among entities with no interactions are diminished. Moreover, we explore two HOI properties, namely *affordances* and *proxemics*, which define the causal relationships between actions/objects and interactions, as well as between spatial relationships and interactions, to inform and mentor the compositional learning process (Fig. 8.1: right). These properties are expressed in *first-order logic* formulae to explicitly specify the logical manipulations among entities. To encourage the model predictions aligning with the specified commonsense and spatial knowledge, these logic rules are converted from discrete symbolic space to continuous data space through *fuzzy logic*-based relaxation, and then serve as the optimization objectives for the neural networks.

To comprehensively evaluate our proposed methods, we experiment them on three gold-standard HOI datasets (*i.e.*, V-COCO [99], HICO-DET [27], and SWIG-HOI [316]), where we achieve **35.47%**, **65.0%**, and **24.95%** overall mAP scores, setting new state-of-the-arts. We also study the performance under the zero-shot setup from different perspectives. As expected, our algorithm consistently delivers

remarkable improvements, up to **+8.38%** mAP under the *unseen object* setup, outperforming all competitors by a large margin.

8.2 Our Approach

In this section, we first introduce the pipeline of LogicHOI, which consists of the proposed triplet-reasoning attention mechanism (§8.2.1), and then present the logic-induced HOI detection learning approach (§8.2.2). Finally, we provide the implementation details (§8.2.3).

8.2.1 HOI Detection via Triplet-Reasoning Attention

Enlightened by the success of DETR [24], recent state-of-the-art approaches [36, 141, 284, 402, 366, 142, 369, 391, 182, 189, 388] for HOI detection typically adopt a Transformer-based encoder-decoder architecture. Specifically, given an interaction decoder containing multiple self-attention layers and the input matrix $\boldsymbol{X}$, the *query*, *key*, *value* embeddings (*i.e.*, $\boldsymbol{F}^{\{q,k,v\}} \in \mathbb{R}^{N\times D}$) can be constructed from the unified embedding of human-object pairs (*i.e.*, $\boldsymbol{Q}^u$) by:

$$\boldsymbol{F}^{\{q,k,v\}} = (\boldsymbol{X}+\boldsymbol{Q}^u)\cdot\boldsymbol{W}^{\{q,k,v\}}, \tag{8.1}$$

where $\boldsymbol{W}^{\{q,k,v\}} \in \mathbb{R}^{D\times D}$ represents the parameter matrix and $\boldsymbol{Q}^u$ can be obtained either from the feature of the union bounding box of detected human and object [378] or directly concatenating their respective embeddings [182]. Then $\boldsymbol{X}$ is updated through a self-attention layer by:

$$\boldsymbol{X}'_i = \boldsymbol{W}^{\text{attn}}\cdot\textstyle\sum_{n=1}^{N}\text{softmax}(\boldsymbol{F}_i^q\cdot\boldsymbol{F}_n^k/\sqrt{D})\cdot\boldsymbol{F}_n^v, \tag{8.2}$$

where $\boldsymbol{W}^{\text{attn}}$ represents the attention weight, and we adopt the single-head variant of self-attention for simplification. Note that under this scheme, the attention score is computed over the embeddings of limited pre-composed human-object pairs. This

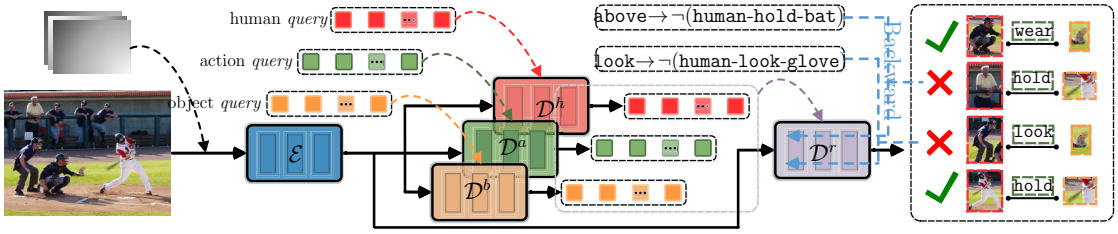


Figure 8.2 : Overview of LogicHOI. We first retrieve human, action, and object *queries* by $\mathcal{D}^h$, $\mathcal{D}^a$, and $\mathcal{D}^o$, respectively. Then $\mathcal{D}^r$ take them as input to combine and explore potential interaction triplets. Finally, this compositional learning process is guided by *affordances* and *proxemics* properties, resulting in a knowledge-informed HOI detection framework.

raises concerns that certain positive human-object pairs may be discarded, and novel combinations cannot be effectively captured during decoding.

To tackle this, we propose solving HOI detection from a compositional perspective. As shown in Fig. 8.2, given the extracted visual features $\boldsymbol{V}$ from backbone, we employ three parallel Transformer decoders $\mathcal{D}^{\{h,a,o\}}$ to extract corresponding human, action, and object features via:

$$\boldsymbol{Q}^{\{h,a,o\}} = \mathcal{D}^{\{h,a,o\}}(\boldsymbol{V}, \boldsymbol{Q}^{\{h,a,o\}}), \tag{8.3}$$

where $\boldsymbol{Q}^{\{h,a,o\}} \in \mathbb{R}^{N_{\{h,a,o\}} \times D}$ are three sets of randomly initialized learnable queries, and $N_h = N_a = N_o$ represents the number of queries per entity type with the superscripts retained for clarity. These query embeddings are then processed through distinct linear layers to deliver element-wise predictions. The human and object predictions are supervised with entity-level category and bounding box annotations, while the action predictions rely on image-level annotations, which indicates all types of actions occurring in the image. Following this, an interaction decoder $\mathcal{D}^r$ composed by multiple Transformer layers is adopted. Notably, the traditional self-attention is replaced by our proposed *triplet-reasoning attention*, which enables

direct attention over the three base elements (*i.e.*, human, action, and object) that constitute an interaction. Specifically, in *triplet-reasoning* attention, in contrast to Eq. 8.1, the shared `human-action` and `action-object` embeddings are explicitly established via:

$$\begin{aligned} \boldsymbol{Q}^{h\text{-}a} &= (\boldsymbol{X} + \boldsymbol{Q}^{h} + \boldsymbol{Q}^{a}) \in \mathbb{R}^{N_h \times N_a \times D}, \\ \boldsymbol{Q}^{a\text{-}o} &= (\boldsymbol{X} + \boldsymbol{Q}^{a} + \boldsymbol{Q}^{o}) \in \mathbb{R}^{N_a \times N_o \times D}, \end{aligned} \tag{8.4}$$

where the human *queries* $\boldsymbol{Q}^{h} \in \mathbb{R}^{N_h \times D}$ and action *queries* $\boldsymbol{Q}^{a} \in \mathbb{R}^{N_a \times D}$ are first expanded to $\mathbb{R}^{N_h \times 1 \times D}$ and $\mathbb{R}^{1 \times N_a \times D}$, respectively. In this way, $\boldsymbol{Q}^{h} + \boldsymbol{Q}^{a}$ associates different `human` and `action` entities, resulting in $N_h \times N_a$ `human-action` pairs in total. Similarly, given $\boldsymbol{Q}^{a} \in \mathbb{R}^{N_a \times 1 \times D}$ and $\boldsymbol{Q}^{o} \in \mathbb{R}^{1 \times N_o \times D}$, $N_a \times N_o$ viable `action-object` pairs are risen by $\boldsymbol{Q}^{a} + \boldsymbol{Q}^{o}$. The *query*, *key*, *value* embeddings, $\boldsymbol{F}^{q}$, $\boldsymbol{F}^{k}$, $\boldsymbol{F}^{v}$ in *triplet-reasoning attention* are then computed as:

$$\begin{aligned} \boldsymbol{F}^{q} &= \boldsymbol{Q}^{h\text{-}a} \cdot \boldsymbol{W}^{q} \in \mathbb{R}^{N_h \times N_a \times D}, \\ \boldsymbol{F}^{k} &= \boldsymbol{Q}^{a\text{-}o} \cdot \boldsymbol{W}^{k} \in \mathbb{R}^{N_a \times N_o \times D}, \\ \boldsymbol{F}^{v} &= \boldsymbol{W}_{h}^{v} \cdot \boldsymbol{Q}^{h\text{-}a} \odot \boldsymbol{Q}^{a\text{-}o} \cdot \boldsymbol{W}_{o}^{v} \in \mathbb{R}^{N_h \times N_a \times N_o \times D}, \end{aligned} \tag{8.5}$$

where $\odot$ is the element-wise production. For the *value* embedding $\boldsymbol{F}^{v}$, it encodes the representation of all $N_h \times N_a \times N_o$ potential interactions, with each element (*e.g.*, $\boldsymbol{F}^{v}_{inj}$) derived from $\boldsymbol{F}^{q}_{in}$ and $\boldsymbol{F}^{k}_{nj}$. This corresponds to the triplet combination of i-th human query $\boldsymbol{Q}^{h}_{i}$, n-th action query $\boldsymbol{Q}^{a}_{n}$, and j-th object query $\boldsymbol{Q}^{o}_{j}$. The input matrix $\boldsymbol{X}$ is then updated by:

$$\boldsymbol{X}'_{ij} = \boldsymbol{W}^{v'} \cdot \textstyle\sum_{n=1}^{N_a} \mathrm{softmax}(\boldsymbol{F}^{q}_{in} \cdot \boldsymbol{F}^{k}_{nj} / \sqrt{D}) \cdot \boldsymbol{F}^{v}_{inj}, \tag{8.6}$$

where $\boldsymbol{X}'$ denotes the refined output of *triplet-reasoning* attention. Unlike *self-attention* (*cf.*, Eq. 8.2) which independently processes interaction queries, *triplet-reasoning* attention (*cf.*, Eq. 8.6) stretches out edges between every `human-action` and `action-object` pairs sharing the same action query, to capture plausible ⟨`human`, `action`, `object`⟩ triplets in a compositional learning manner. The final output $\boldsymbol{Y}$ of the in-

teraction decoder $\mathcal{D}^r$ is computed as:

$$\boldsymbol{Y} = \mathcal{D}^r(\boldsymbol{V}, \boldsymbol{Q}^h, \boldsymbol{Q}^a, \boldsymbol{Q}^o) \in \mathbb{R}^{N_h \times N_o \times D}. \tag{8.7}$$

Here $\mathcal{D}^r$ iteratively refines and filters triplet combinations through the layer-wise inference within the Transformer, to predict interaction categories for $N_h \times N_o$ `human-object` pairs. We set N_h, N_a, N_o to a relatively small number (*e.g.*, 32) to balance computational efficiency and accuracy, as larger number of queries exacerbate the imbalance between positive and negative samples. To further enhance efficiency, we filter out low-scoring human, object, and action queries and keep only half of them before feeding into $\mathcal{D}^r$.

8.2.2 Logic-Guided HOI Detection Learning

To explicitly supervise the compositional learning process above, we propose leveraging the *affordances* and *proxemics* properties embedded between human, action, and objects, for HOI detection. Here *affordances* refers to the property that given an determined object, only a partial number of actions can facilitate it, and vice versa. For instance, the observation of a `kite` may give rise to actions such as `launch` or `fly`, while other choices such as `repair`, `throw` are implausible. Similarly, the observation of one man is `throw` something, can only lead us to infer the objects as small things such as `ball` or `cup`. Meanwhile, *proxemics* describes the relative spatial relationships between humans and objects, *e.g.*, when something is positioned `above` a human, the candidate actions are restricted to `airplane`, `kite`, *etc.* In this work, a total of five positional relationships are considered, which are `above` (*e.g.*, kite `above` human), `below` (*e.g.*, skateboard `below` human), `around` (*e.g.*, giraffe `around` human), `within` (*e.g.*, handbag `within` human), `containing` (*e.g.*, human `smiling`). We compose these two kinds of properties and state them as first-order logical formulas, so as to instruct the interaction prediction process of LogicHOI. Specifically, given one `human-object` pair x that is potential to have

interactions, the predicated `action` $\mathcal{V}(x)$ of this pair, and the spatial relationship $\mathcal{P}(x)$ between them, the following casual relations can be derived:

$$\forall x(\mathcal{V}(x) \wedge \mathcal{P}(x) \rightarrow \neg\, \mathcal{R}_1(x) \wedge \cdots \wedge \neg\, \mathcal{R}_M(x)), \tag{8.8}$$

where $\{\mathcal{R}_1, \cdots, \mathcal{R}_M\}$ are infeasible interactions. In first-order logic, x is the *variables* to represent arbitrary elements; $\mathcal{V}, \mathcal{P}$, and $\mathcal{R}_m$ are *predicates* to represent properties of objects or relations between objects; $\wedge$, $\neg$, and $\rightarrow$ are *logical connectives* to define the relationships between statements; $\forall$ is the *quantifiers* to create general statements about entire sets of objects. Eq. 8.8 states that, for instance, if the v is `launch`, and p is `above`, then the predicated interaction can not be `human-throw-ball` which violates the pre-defined action, and `human-launch-boat` which violates the spatial relation. Similarly, given the `object` category $\mathcal{O}$ and `position` relationship $\mathcal{P}$, we shall have:

$$\forall x(\mathcal{O}(x) \wedge \mathcal{P}(x) \rightarrow \neg\, \mathcal{R}_1(x) \wedge \cdots \wedge \neg\, \mathcal{R}_N(x)), \tag{8.9}$$

which indicates that, for example, if a skateboard is standing `around` a person, the interaction cannot be `human-skate-skateboard`. Eq. 8.8 and Eq. 8.9 clearly define the *affordances* and *proxemics* properties. However, all these formulae are expressed in first-order logic, which strictly requires every statement to be either true or false in binary. This does not align with real-world reasoning which often involves vague or uncertain information. Additionally, first-order logic is not differentiable, making it incompatible with neural networks. To tackle this, we employ fuzzy logic [300], whose value range typically spans $[0, 1]$ to represent degrees of truth. A value closer to 1 indicates a higher degree of truth. For logical connectives (*e.g.*, $\rightarrow$, $\neg$, $\vee$, $\wedge$), they can be relaxed to functions working on continuous variables via:

$$\begin{aligned} \psi \rightarrow \phi &= 1 - \psi + \psi \cdot \phi, & \neg\psi &= 1 - \psi, \\ \psi \vee \phi &= \max(\psi, \phi), & \psi \wedge \phi &= \psi \cdot \phi. \end{aligned} \tag{8.10}$$

Similarly, the *quantifier* are implemented in a generalized-mean manner following [9]:

$$\begin{aligned}\exists x(\psi(x)) &= (\tfrac{1}{K}\textstyle\sum_{k=1}^{K}\psi(x_k)^q)^{\frac{1}{q}},\\ \forall x(\psi(x)) &= 1-(\tfrac{1}{K}\textstyle\sum_{k=1}^{K}(1-\psi(x_k))^q)^{\frac{1}{q}}.\end{aligned} \tag{8.11}$$

For the *variable* x, it is instantiated with a specific human-object pairs k in the image. Then, given sample k, the *predicate* $\mathcal{V},\mathcal{O},\mathcal{R}$ are grounded to the category-wise scores output by action decoder $\mathcal{D}^a$ (*i.e.*, $s_k[\mathcal{V}]$), object decoder $\mathcal{D}^o$ (*i.e.*, $s_k[\mathcal{O}]$), and interaction decoder $\mathcal{D}^r$ (*i.e.*, $s_k[\mathcal{R}]$). As such, Eq. 8.8 is ready to be grounded into sub-symbolic space as:

$$\mathcal{G}_{\mathcal{V},\mathcal{P}} = 1-\tfrac{1}{M}\textstyle\sum_{m=1}^{M}(\tfrac{1}{K}\sum_{k=1}^{K}(s_k[\mathcal{V}]\cdot s_k[\mathcal{R}_m])), \tag{8.12}$$

where K refers to the number of all human-object pairs in the training set. Since it is impractical to include all samples in one training step, we relax K to samples in a mini-batch. For the position *predicate* $\mathcal{P}$, the spatial relation between humans and objects is predetermined and can be effortlessly inferred from the box predictions. Thus, we omit $\mathcal{P}(x)$ when grounded into Eq. 8.12. Similarly, Eq. 8.9 can be grounded into:

$$\mathcal{G}_{\mathcal{O},\mathcal{P}} = 1-\tfrac{1}{N}\textstyle\sum_{n=1}^{N}(\tfrac{1}{K}\sum_{k=1}^{K}(s_k[\mathcal{O}]\cdot s_k[\mathcal{R}_n])). \tag{8.13}$$

For $\mathcal{G}_{\mathcal{V},\mathcal{P}}$, it scores the satisfaction of predictions to rules defined in Eq. 8.8. For example, given a high probability of action `ride` (*i.e.*, a high value of $s_k[\mathcal{V}]$) and the position relationship `above`, if the probability of infeasible interactions (*e.g.*, `human-feed-fish`) is also high, then $\mathcal{G}_{\mathcal{V},\mathcal{P}}$ would receive a low value. On the other hand, $\mathcal{G}_{\mathcal{O},\mathcal{P}}$ scores the satisfaction of predictions to Eq. 8.9 with given position and objects. Both of them can be converted into optimization objectives to supervise the training of models via:

$$\begin{aligned}\mathcal{L}_{\mathcal{V},\mathcal{P}} &= 1-\mathcal{G}_{\mathcal{V},\mathcal{P}}\\ &= \tfrac{1}{M}\textstyle\sum_{m=1}^{M}(\tfrac{1}{K}\sum_{k=1}^{K}(s_k[\mathcal{V}]\cdot s_k[\mathcal{R}_m])),\end{aligned} \tag{8.14}$$

$$\begin{aligned}\mathcal{L}_{\mathcal{O},\mathcal{P}} &= 1 - \mathcal{G}_{\mathcal{O},\mathcal{P}} \\ &= \tfrac{1}{N}\textstyle\sum_{n=1}^{N}(\tfrac{1}{K}\sum_{k=1}^{K}(s_k[\mathcal{O}]\cdot s_k[\mathcal{R}_n])).\end{aligned} \tag{8.15}$$

Though several studies have explored the *affordances* and *proxemics* properties to a limited extent [144, 369, 119], they primarily adopt a statistical perspective, *e.g.*, computing the distribution of co-occurrence of actions and objects to reformulate the predictions [144], enhancing spatial awareness by incorporating positional encodings into network features [369, 378], or introducing a two-path feature generator [119] which increases the number of network parameters. In contrast, by embedding clear, logical constraints into the learning process, our approach makes more interpretable predictions and becomes robust against implausible interactions. Moreover, while prior work relies on extra hand-crafted modules to manage different kinds of properties, our logical framework can flexibly incorporate new rules as needed, without modifying the network architecture.

8.2.3 Implementation Details

Rule Construction. The rule base is constructed based on HowTo100M [223], a huge collection of YouTube instructional videos involving interactions with the physical world (*e.g.*, making peanut butter, pruning a tree) and contains 23,611 visual tasks in total. For our purposes, we utilize only the text queries (*e.g.*, "how to paint furniture") which outline feasible objects (*i.e.*, furniture) for an action (*i.e.*, paint) to be performed by human. Then the spatial relations between humans and objects for a specific interaction (*e.g.*, human-paint-furniture) is determined using LLAMA2-7B [290], with responses constrained to five spatial relations defined below.

Spatial Relation Definition. Let the bounding boxes for objects and humans be denoted as $(c_x^o, c_y^o, l_x^o, l_y^o)$ and $(c_x^h, c_y^h, l_x^h, l_y^h)$, respectively, where (c_x, c_y) represents the central coordinates of the bounding box, and l_x and l_y correspond to its width and height. The five spatial relations utilized in this work, as illustrated in Fig. 8.3, are

defined as:

- `Above`: An object is positioned at a higher location relative to a person, such as a bird or an airplane flying overhead. This condition is satisfied if: $c_y^o > c_y^h + \lambda \cdot l_y^h$.
- `Below`: An object is located at a lower position relative to a person, such as a person riding a bicycle. This condition holds if: $c_y^o < c_y^h - \lambda \cdot l_y^h$.
- `Around`: An object exhibits significant overlap with either the left or right side of a person, such as a car positioned behind them. This condition is met if: $(c_y^h - \lambda \cdot l_y^h < c_y^o < c_y^h + \lambda \cdot l_y^h) \wedge ((c_x^o < c_x^h - \lambda \cdot l_x^h) \vee (c_x^o > c_x^h + \lambda \cdot l_x^h))$.
- `Within`: The bounding box of a human completely encloses an object, such as a person holding a phone. This condition is satisfied if: $(c_y^h - \lambda \cdot l_y^h < c_y^o < c_y^h + \lambda \cdot l_y^h) \wedge (c_x^h - \lambda \cdot l_x^h < c_x^o < c_x^h + \lambda \cdot l_x^h)$.
- `Containing`: This relation applies when a person exhibits body motions without interacting with any object, such as in an image of a person smiling. It also applies when a person interacts with an object that is not explicitly detected. In this case, only one bounding box represents the person, and no object bounding box is specified.

Here λ ($= 0.8$) is a hyperparameter to control and relax the margins between different spatial relations.

Network Architecture. To ensure a fair comparison with existing Transformer-based approaches [36, 141, 284, 402, 366, 142, 369, 391, 182, 189, 388], we adopt ResNet-50 [106] and Swin-L [194] as the backbone. The visual encoder $\mathcal{E}$ consists of six Transformer encoder layers, while the human, object, and action decoders, $\mathcal{D}^h$, $\mathcal{D}^o$, and $\mathcal{D}^a$, are each implemented using three Transformer decoder layers. Similarly, the interaction decoder $\mathcal{D}^r$ is constructed with three Transformer decoder layers; however, we replace *self-attention* with our proposed *triplet-reasoning attention* to enhance relational reasoning. To balance efficiency and expressiveness, we

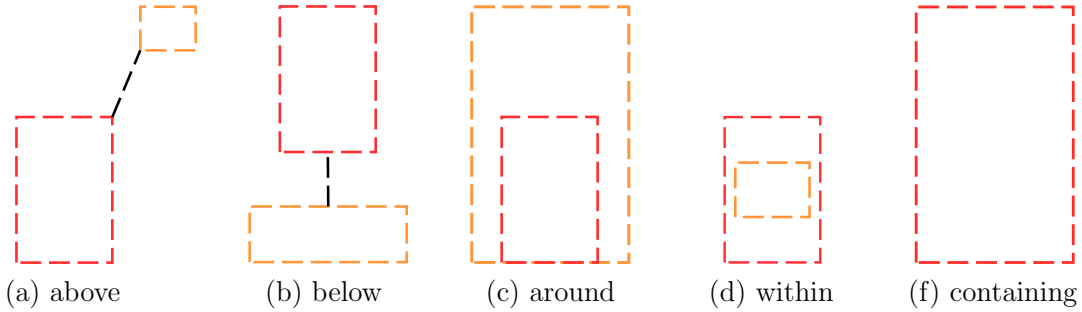


Figure 8.3 : Illustration of five spatial relationships between humans ( ) and objects ( ) used in LogicHOI.

set the number of human, object, and action *queries* (N_h, N_o, N_a) to 32, and the hidden dimension of all modules to $D = 768$. To improve inference efficiency, we adopt the guided embedding approach proposed in [182], merging the human and object decoders ($\mathcal{D}^h$ and $\mathcal{D}^o$) into a single Transformer decoder to jointly predict human and object instances. To make the Transformer-based interaction decoder spatiality-aware, the human and object embeddings retrieved from Eq. 8.3 are concatenated with sinusoidal positional encodings generated from predicted bounding boxes. An auxiliary loss is applied to the intermediate outputs of each decoder layer which contributes to enhanced performance by improving the decoding process. As state-of-the-art approaches [315, 119, 182] typically leverage large-scale vision-language pre-trained models to improve the zero-shot generation ability, we follow this setup and adopt the ViT-B/32 variant of CLIP [255] to enhance object and action detection. The temperature τ used in Gumbel-softmax is set to 1.4 at the start of training, and gradually decreases to 0.7.

Training Objectives. The training objective for LogicHOI is composed of two parts, one is the HOI detection loss (*i.e.*, $\mathcal{L}_{\text{HOI}}$) and the other is for logic-induced

property learning (*i.e.*, $\mathcal{L}_{\text{LOG}}$):

$$\mathcal{L} = \mathcal{L}_{\text{HOI}} + \alpha\mathcal{L}_{\text{LOG}}, \qquad \mathcal{L}_{\text{LOG}} = \mathcal{L}_{\mathcal{V},\mathcal{P}} + \mathcal{L}_{\mathcal{O},\mathcal{P}}. \tag{8.16}$$

Here α is set to 0.2 empirically. Note that $\mathcal{L}_{\text{LOG}}$ solely updates the parameters of the interaction decoder $\mathcal{D}^r$, but not the entire network to prevent over-fitting. For $\mathcal{L}_{\text{HOI}}$, it is composed of human/object (*i.e.*, output of $\mathcal{D}^h$ and $\mathcal{D}^o$, respectively) detection loss, action (*i.e.*, output of $\mathcal{D}^a$) classification loss as well as interaction (*i.e.*, output of $\mathcal{D}^r$) classification loss. The detection loss is implemented in accordance with DETR [24]. Specifically, we adopt the cross entropy loss for object classification, and the smooth ℓ_1 loss as well as the generalized intersection over union (GIoU) loss for bounding box regression during training.

8.3 Experiments

8.3.1 Experimental Setup

Datasets. We conduct experiments on three widely-used HOI detection benchmarks:

- V-COCO [99] is a carefully curated subset of MS-COCO [187] which contains 10,346 images (5,400 for training and 4,946 for testing). There are 263 human-object interactions annotated in this dataset in total, which are derived from 80 object categories and 29 action categories.

- HICO-DET [27] consists of 47,776 images, with 38,118 for training and 9,658 designated for testing. It has 80 object categories identical to those in V-COCO and 117 action categories, consequently encompassing a comprehensive collection of 600 unique human-object interactions.

- SWIG-HOI [316] is a open-set dataset comprising 1,000 object categories interacting with 406 human actions, which results in a long-tailed distribution for

interactions. It includes around 45,000/14,000 images for training/testing.

Evaluation Metric. Following conventions [93, 252, 141], the mean Average Precision (mAP) is adopted for evaluation. Specifically, for V-COCO, we report the mAP scores under both scenario 1 (S1) which includes all of the 29 action categories and scenario 2 (S2) which excludes 4 human body motions without interaction to any objects. For HICO-DET, we perform evaluation across three category sets: all 600 HOI categories (Full), 138 HOI categories with less than 10 training instances (Rare), and the remaining 462 HOI categories (Non-Rare). Moreover, the mAP scores are calculated in two separate setups: **i)** the Default setup computing the mAP on all testing images, and **ii)** the Known Object setup measuring the AP for each object independently within the subset of images containing this object. For SWIG-HOI, we report mAP across Non-Rare/Rare/Unseen/Full four sets. An interaction is considered as Non-Rare if it has more than 10 training samples, Rare with at least one but fewer than 10 training samples, and Unseen with no training samples.

Zero-Shot HOI Detection. For HICO-DET, we follow the setup in previous work [182, 113] to carry on zero-shot generalization experiments, resulting in four setups: Rare First Unseen Combination (RF-UC), Non-rare First Unseen Combination (NF-UC), Unseen Verb (UV) and Unseen Object (UO) on HICO-DET. Specifically, the RF and NF strategies in the UC setting indicate selecting 120 most frequent and infrequent interaction categories for testing, respectively. In the UO setting, we choose 12 objects from 80 objects that are previously unseen in the training set, while in the UV setting, we exclude 20 verbs from a total of 117 verbs during training and only use them at testing. For SWIG-HOI, the test set includes approximately 5,500 interactions, of which around 1,800 not present in the training set (*i.e.*, Unseen).

Training. To ensure fair comparison with existing work [36, 141, 284, 402, 366, 142, 369, 391, 182, 189, 388], we initialize LogicHOI with weights of DETR [24] pre-trained on MS-COCO. Subsequently, we train the model for 90 epochs on HICO-DET/V-COCO, and 60 epochs on SWIG-HOI, using the AdamW optimizer with a batch size of 16 and a base learning rate of $1e^{-4}$. Training is conducted on four GeForce RTX 3090 GPUs. The learning rate is scheduled following a step policy, decayed by a factor of 0.1 after completing two-thirds of the total training epochs. In line with [24, 60, 141], the random scaling augmentation is adopted, where training images are resized to a maximum size of 1333 for the long edge, and minimum size of 400 for the short edge.

Testing. For fairness, we do not implement any data augmentation during testing. Specifically, we first select K interactions with the highest scores and further filter them by applying NMS to retrieve the final predictions. Following the convention [284, 182, 60, 391, 388], K is set to 100.

8.3.2 Zero-Shot HOI Detection

Results on HICO-DET. The comparisons of LogicHOI against several top-leading zero-shot HOI detection models on HICO-DET are presented in Table 8.1.

- **Unseen Combination.** As seen, LogicHOI provides a comparable performance gain against existing methods. In particular, it outperforms GEN-VLKT [182] by **4.61%** and **1.79%** in terms of mAP on *unseen* categories for RF and NF scenarios. These numerical results substantiate our motivation of empowering models with the compositional learning ability and guide the training process in a logic-induced manner, rather than solely rely on category cross entropy loss.

- **Unseen Object.** LogicHOI also demonstrates superior performance in the UO setup. Concretely, it surpasses GEN-VLKT [182] by **5.16%** mAP on *unseen* cat-

Method	Type	Unseen	Seen	Full
GEN-VLKT[182][CVPR22]	RF-UC	21.36	32.91	30.56
SCL[113][ECCV22]	RF-UC	19.07	30.39	28.08
OpenCat[386][CVPR23]	RF-UC	21.46	33.86	31.38
HOICLIP[230][CVPR23]	RF-UC	25.53	34.85	32.99
CLIP4HOI[212][NeurIPS23]	RF-UC	28.47	35.48	34.08
LogicHOI[166][NeurIPS23]	RF-UC	25.97	34.93	33.17
GEN-VLKT[182][CVPR22]	NF-UC	25.05	23.38	23.71
OpenCat[386][CVPR23]	RF-UC	23.25	28.04	27.08
HOICLIP[230][CVPR23]	NF-UC	26.39	28.10	27.75
CLIP4HOI[212][NeurIPS23]	NF-UC	**31.44**	28.26	28.90
LogicHOI[166][NeurIPS23]	NF-UC	26.84	27.86	27.95

(a) RF-UC & NF-UC

Method	Type	Unseen	Seen	Full
ATL[111][CVPR21]	UO	5.05	14.69	13.08
FCL[112][CVPR21]	UO	0.00	13.71	11.43
GEN-VLKT[182][CVPR22]	UO	10.51	28.92	25.63
OpenCat[386][CVPR23]	UO	**23.84**	28.49	27.72
HOICLIP[230][CVPR23]	UO	16.20	30.99	28.53
LogicHOI[166][NeurIPS23]	UO	15.67	30.42	28.23
GEN-VLKT[182][CVPR22]	UV	20.96	30.23	28.74
EoID[338][AAAI23]	UV	22.71	30.73	29.61
HOICLIP[230][CVPR23]	UV	24.30	32.19	31.09
CLIP4HOI[212][NeurIPS23]	UV	26.02	31.14	30.42
LogicHOI[166][NeurIPS23]	UV	24.57	31.88	30.77

(b) UO & UV

Table 8.1 : Comparison of zero-shot generalization with state-of-the-arts on HICO-DET [27] `test`. See §8.3.2 for details.

egories and achieves **28.23%** overall scores. For OpenCat [386], it undergoes pretraining on five datasets, which utilizes significantly larger number of training samples than our approaches.

- **Unseen Verb.** Our approaches achieves dominant results under the UV setting, surpassing other competitors across all metrics. Notably, LogicHOI yields **24.57%** *unseen* mAP scores, while the corresponding performance for the existing methods (*e.g.*, GEN-VLKT [182], EoID [338], and HOICLIP [230]) are 20.06%, 22.71%, and 24.30%, presenting an improvement up to **3.61%**, **1.86%**, and **0.27%** mAP scores.

All of the above confirm the effectiveness of our proposed compositional neural-logic learning framework, which is informed by *affordances* and *proxemics* knowledge to address novel challenges that was never encountered before.

Method	Non-rare	Rare	Unseen	Full
QPIC [284] [CVPR21]	16.95	10.84	6.21	11.12
THID [315] [CVPR22]	17.67	12.82	10.04	13.26
CMD-SE [159] [CVPR24]	21.46	14.64	10.70	15.26
LogicHOI [166] [NeurIPS23]	24.95	19.47	16.24	20.62

Table 8.2 : Quantitative results for zero-shot generalization on SWIG-HOI [316] `test`. See §8.3.2 for details.

Results on SWIG-HOI. As shown in Table 8.2, LogicHOI demonstrate impressive performance across all four setups, achieving significant improvements of **5.36%** mAP for overall performance. It is noteworthy that LogicHOI demonstrates exceptional accuracy in handling unseen categories, significantly outperforming prior work (*i.e.*, **16.24%** *vs.* 10.70% mAP for CMD-SE [159]). These results reinforces our belief that integrating compositional learning with neural-logic computing is imperative and indispensable.

8.3.3 Regular HOI Detection

Results on HICO-DET. In Table 8.3, we present the results of our methods and other top-performing models under the normal HOI detection setup on HICO-DET [27] `test`. LogicHOI demonstrates a comparable performance to previous state-of-the-art [182], with **35.46%**/ **32.03%**/**36.22%** mAP scores for Full, Rare, and Non-Rare categories under the Default setup. Similar trends can be observed after replacing with the backbone with Transformer-based ones (*i.e.*, Swin-L), where LogicHOI yields the best performance across all metrics. This verifies the general effectiveness of our approaches.

Results on V-COCO. As indicated by the last two columns of Table 8.3, we also compare LogicHOI with competitive models on V-COCO [99] `test`. Despite the

relatively smaller number of images and HOI categories in this dataset, our method still yields promising results, showcasing its effectiveness. In particular, it achieves a mean mAP score of **65.0%** across two scenarios. Furthermore,when using Swin-L as the backbone, LogicHOI outperforms all counterparts and establishes new state-of-the-arts under both scenarios.

8.3.4 Qualitative HOI Detection Result

We provide qualitative results of LogicHOI, highlighting both success and failure cases in Fig. 8.4. It can be observed that our methods demonstrate remarkable improvements in HOI detection across a wide range of scenarios. This is attributed to the integration of triplet reasoning and logic-guided knowledge learning, which enables our models to effectively capture complex relationships between humans and objects and enhance detection accuracy. However, certain challenges remain. Specifically, as shown in the last column, our models face difficulties when dealing with highly ambiguous relations, such as instances where a frisbee is held by a human in a strange pose. The complex spatial arrangement and occlusion make it challenging for the model to accurately infer the correct HOI. Additionally, our approaches may be inefficient when it needs to deduce additional contextual cues. For example, when a chair is partially occluded by a human, our model struggles to correctly recognize the interaction between the two entities due to the lack of complete visual information.

8.3.5 Diagnostic Experiment

For in-depth analysis, we perform a series of ablative studies on HICO-DET [27] `test` with ResNet-50 as the backbone.

Key Component Analysis. We first examine the effectiveness of essential designs of LogicHOI, *i.e.*, *triplet-reasoning* attention (TRA) and logic-guided HOI

Figure 8.4 : Successful and failure cases selected from V-COCO [99] and HICO-DET [27].

detection learning (LGL). The results are summarized in Table 8.4, from which three key conclusions can be drawn. First, our proposed *triplet-reasoning* attention significantly enhances performance against the baseline across all the metrics. Notably, TRA achieves **4.53%** mAP improvement on Rare categories, demonstrating the ability of compositional learning to explore entity combinations and generate feasible predictions. Second, we also observe compelling gains after incorporating LGL into the baseline, even with standard self-attention, affirming its versatility. Third, our full model LogicHOI achieves satisfactory performance, which confirms the complementarity and effectiveness of TRA and LGL.

Triplet-Reasoning Attention. We further investigate the impact of different configurations on the number of queries and layers used in the interaction decoder $\mathcal{D}^r$, which directly influence the capability of *triplet-reasoning* attention. As shown in Table 8.7, LogicHOI achieves similar performance when L is larger than 2. For efficiency, we set $L = 3$ which is the smallest among existing work [36, 141, 284, 402, 366, 142, 369, 391, 182, 189, 388]. Table 8.6 summarizes the results for

Method	Backbone	Default			Known Object			AP^{S1}_{role}	AP^{S2}_{role}
		Full	Rare	Non-Rare	Full	Rare	Non-Rare		
AS-Net[36][CVPR21]	R50	28.87	24.25	30.25	31.74	27.07	33.14	53.9	-
QPIC[284][CVPR21]	R50	29.07	21.85	31.23	31.68	24.14	33.93	58.8	61.0
CDN[366][NeurIPS21]	R50	31.78	27.55	33.05	34.53	29.73	35.96	62.3	64.4
SSRT[119][CVPR22]	R50	30.36	25.42	31.83	-	-	-	63.7	65.9
UPT[369][CVPR22]	R50	31.66	25.94	33.36	35.05	29.27	36.77	59.0	64.5
CTAN[60][CVPR22]	R50	31.71	24.82	33.77	33.96	26.37	36.23	60.1	
Iwin[292][ECCV22]	R50-FPN	32.03	27.62	34.14	35.17	28.79	35.91	60.5	-
STIP[378][CVPR22]	R50	32.22	28.15	33.43	35.29	31.43	36.45	65.1	69.7
DOQ[254][CVPR22]	R50	33.28	29.19	34.50	-	-	-	63.5	-
IF-HOI[189][CVPR22]	R50	33.51	30.30	34.46	36.28	33.16	37.21	63.0	65.2
GEN-VLK[182][CVPR22]	R50	33.75	29.25	35.10	37.80	34.76	38.71	62.4	64.4
HOICLIP[230][CVPR23]	R50	34.69	31.12	35.74	37.61	34.47	38.54	63.5	64.8
PViC[370][ICCV23]	R50	34.69	32.14	35.45	38.14	35.38	38.97	62.8	67.8
LogicHOI[166][NeurIPS23]	R50	**35.47**	**32.03**	**36.22**	**38.21**	**35.29**	**39.03**	64.4	65.6
FGAHOI[209][PAMI24]	Swin-L	37.18	30.71	39.11	38.93	31.93	41.02	-	-
ADA-CM[158][ICCV23]	ViT-L	38.40	37.52	38.66	-	-	-	58.6	64.0
PViC[370][ICCV23]	Swin-L	44.32	44.61	44.24	47.81	48.38	47.64	61.7	68.0
LogicHOI[166][NeurIPS23]	Swin-L	**45.53**	**46.14**	**45.20**	**48.96**	**49.91**	**48.63**	**66.4**	68.9

Table 8.3 : Quantitative results on HICO-DET [27] `test` and V-COCO [99] `test`.

TRA	LRL	Full	Rare	Non-Rare
		31.87	26.14	33.29
✓		34.32	30.67	35.19
	✓	33.26	29.53	34.56
✓	✓	**35.47** ↑3.60	**32.03** ↑5.89	**36.22** ↑2.93

Table 8.4 : Analysis of essential components of LogicHOI on HICO-DET [27].

Setting	RF-UC	UO	UV
TRA	24.01	13.26	23.14
$+\mathcal{L}_{v,p}$	25.22	15.32	23.68
$+\mathcal{L}_{o,p}$	25.34	13.91	24.29
LogicHOI	**25.97** ↑1.96	**15.67** ↑2.41	**24.57** ↑1.43

Table 8.5 : Analysis of LRL under the zero-shot setup of *unseen* categories.

# of layers (L)	Full	Rare	Non-Rare
2	34.61	30.72	35.54
3	**35.47**	**32.03**	**36.22**
4	35.37	31.96	36.09
6	35.61	32.13	36.39

Table 8.6 : Analysis of interaction decoder layer number on HICO-DET [27].

# of queries (N)	Full	Rare	Non-Rare
16	35.06	31.36	35.94
32	**35.47**	**32.03**	**36.22**
64	35.26	31.65	36.06
128	34.67	30.98	35.53

Table 8.7 : Analysis of number of the *queries* on HICO-DET [27].

different numbers of queries assigned to humans, objects, and actions. Note that the number of queries for these three categories is identical and denoted as N. The best performance is obtained at $N = 32$ and more queries lead to inferior performance. This may be because large number of human-object combinations with most of them being infeasible render negative impacts to the compositional learning process.

Logic-Guided Learning. We guide the compositional learning process with two logic-induced training objectives. Table 8.5 reports the scores of *unseen* categories under three zero-shot setups on HICO-DET. As seen, the contributions of $\mathcal{L}_{v,p}$ and $\mathcal{L}_{o,p}$ are approximately equal in the RF-UC setup, since during training, all actions and objects can be seen and utilized to guide reasoning. On the other hand, under the UO and UV setups, the improvements heavily rely on $\mathcal{L}_{v,p}$ and $\mathcal{L}_{o,p}$ respectively, while the other one brings minor enhancements. Finally, the combination of both leads to LogicHOI, which sets the new state-of-the-arts on all zero-shot scenarios.

Runtime Analysis. The computational complexity of our *triplet-reasoning attention* is squared compared to *self-attention*. Towards this, we make some specific designs: **i)** both the number of *queries* and Transformer decoder layers of our method are the smallest when compared to existing work [36, 141, 284, 402, 366, 142, 369, 391, 182, 189, 388], **ii)** as specified in §8.2.2, we filter the human, action, object

Method	Backbone	Params	FLOPs	FPS
Two-stages Detectors:				
iCAN[82][BMVC18]	R50	39.8	-	5.99
DRG[81][ECCV20]	R50-FPN	46.1	-	6.05
STIP[378][CVPR22]	R50	50.4	-	6.78
One-stages Detectors:				
PPDM[181][CVPR20]	HG104	194.9	-	17.14
HOTR[141][CVPR21]	R50	51.2	90.78	15.18
CDN-S[366][NeurIPS21]	R50	42.1	-	15.54
GEN-VLK$_s$[182][CVPR22]	R50	42.8	86.74	18.69
LogicHOI[166][NeurIPS23]	R50	49.8	89.65	16.84

Table 8.8 : Analysis of parameters and running efficiency.

queries and only keep half of them for efficiency, and **iii)** *triplet-reasoning attention* introduces few additional parameters. As summarized in Table 8.8, above facets make LogicHOI even smaller in terms of FLOPs and faster in terms of inference speed compared to most existing work.

8.4 Limitation

It is important to acknowledge a limitation regarding the scale of validation within our study. The number of interactions included in the dataset for model evaluation is limited to fewer than 600 instances. This constrained sample size falls short of capturing the full spectrum of interactions that take place in real-world scenarios. Consequently, the exploration of applications related to object and interaction detection in more complex and diverse situations may be hindered.

8.5 Conclusion

In this chapter, we propose LogicHOI a high-performance neuro-symbolic model for HOI detection. Unlike existing methods relying on predetermined human-object pairs, our approaches enable the exploration of novel combinations of entities during decoding, which improves the detection performance as well as the zero-shot generalization capabilities. This is achieved by: **i)** modifying the *self-attention* mechanism in vanilla Transformer to reason over ⟨`human`, `action`, `object`⟩ triplets; and **ii)** incorporating *affordances* and *proxemics* properties as constraints to guide the learning process in a logic-induced manner. Experiments on three gold-standard HOI datasets demonstrates the superiority of our approaches against existing methods. Our work opens a new avenue for HOI detection from the perspective of combining compositional learning with neural-logic computing, and we hope it could inspire future research in this direction.

Chapter 9

Conclusion and Future Works

9.1 Summary

This thesis presents a comprehensive investigation into developing deep learning algorithms with structured visual perception capabilities, moving beyond isolated interpretation of semantic categories towards a holistic understanding of visual environments. Our research is advanced through three interconnected and progressively sophisticated perspectives.

Temporal Structure Modeling. The initial effort of our work focused on modeling structures inherent in dynamic visual information. This involved leveraging visual correspondence across temporal observations to uncover how objects and scenes evolve over time, primarily through self-supervised learning from unlabeled videos. In Chapter 3, we introduced a locality-aware inter-and intra-video reconstruction framework to enhance self-supervised temporal correspondence learning. This aimed to learn robust representations of temporal dynamics directly from real-world data. Further, in Chapter 4, we integrated mask embedding learning into the correspondence-driven paradigm. Through a self-taught mechanism that alternates between spacetime pixel clustering for pseudo-mask generation and mask-embedded segmentation learning, this work enables more accurate and robust object tracking in unlabeled videos by explicitly modeling the temporal structures of target objects.

Structured Scene Understanding. Extending from the temporal to the spatial dimension, the thesis then delved into the structural organization within visual scenes. This involved understanding not just individual components but also their

inter-relations and compositions. In Chapter 5, we tackled hierarchical semantic segmentation, reformulating it to allow existing networks to recognize and leverage predefined class hierarchies (*e.g.*, a "car" is a "vehicle"). This work aimed to capture how semantic concepts relate to one another in a structured manner. Further, in Chapter 6, we addressed the complex task of Human-Object Interaction detection by innovatively employing large-scale diffusion models. By steering these generative models to model relations and generate diverse training data, we sought to enable a deeper comprehension of how humans and objects are organized and interact, crucial for robust generalization to novel configurations.

Structured Knowledge Integration. Finally, moving beyond tailored considerations for specific structural types, the research investigated a general principle for realizing structured visual perception through explicit knowledge integration. This perspective, explored in the burgeoning field of neuro-symbolic computing, aimed to inform visual perception models with explicit symbolic knowledge, such as commonsense or domain-specific constraints. Chapter 7 demonstrated this by representing hierarchical semantic knowledge using first-order logic, translating these rules into differentiable losses to guide network training and inference for more robust and interpretable segmentation. Building on this, Chapter 8 incorporated commonsense knowledge about object affordances and proxemics into HOI detection, again using logical rules relaxed into continuous constraints. This approach aimed to create visual understanding models capable of reasoning about interrelations and handling complex situations through structured reasoning rather than purely empirical pattern matching.

Overall, this body of work represents a dedicated effort to equip machines with a more profound, human-like visual intelligence. By systematically addressing temporal dynamics, spatial-relational structures, and the general integration of explicit knowledge, we strive towards creating systems that can perceive, interpret, and

reason about the visual world in a truly structured manner.

9.2 Future Directions

Moving forward, the insights and methodologies developed in this thesis pave the way for several exciting avenues of future research. I am broadly interested in advancing the capacity of AI systems to perceive, reason about, and ultimately learn the inherent structure within the visual world in a more autonomous, robust, and interpretable manner. Within this overarching ambition, I identify the following key research directions that extend my past and ongoing work:

Leveraging Foundation Models for Visual Structure Learning. The remarkable success of multimodal foundation models stems from their ability to learn rich statistical correlations from web-scale data. However, these models often function as black boxes and can struggle with systematic generalization, fine-grained reasoning, and logical consistency, frequently producing plausible yet incorrect hallucinations. The explicit structural representations developed in this thesis provide a principled means to augment such models and mitigate these limitations. Speficically, the neural-logic frameworks introduced in Chapter 7 (LogicSeg) and Chapter 8 (LogicHOI) can be adapted to function as integrated reasoning layers within multimodal architectures. Instead of relying solely on the emergent reasoning abilities of transformers, these frameworks could operate directly on fused multimodal embeddings. For example, before generating a final textual description or a set of bounding boxes, the model's latent representations could be processed through a differentiable logic module that enforces consistency with commonsense constraints—such as object affordances, spatial proxemics, or semantic hierarchies—by modulating the latent activations. This integration would guide the model to produce outputs that are not only statistically probable but also logically coherent, enhancing its ability to generalize to novel compositions. Although multimodal models are good at recog-

nizing objects and broad scene layouts, their grasp of precise spatial and temporal relationships remains limited. The methods developed in this thesis can be reformulated as structure-aware training objectives to fine-tune such models toward more structured perception. The relational modeling approaches from Part II (Chapters 5–6) can introduce hierarchy-consistent loss functions—for example, integrating the hierarchical consistency loss from Chapter 5 into a segmentation-based VLM fine-tuning pipeline. This would penalize predictions that violate semantic structure, such as labeling a region as wheel without simultaneously recognizing it as part of a vehicle. Likewise, the self-supervised objectives for learning temporal correspondence from Part I (Chapters 3–4) could be employed to pre-train or adapt the vision encoder of a multimodal model using large-scale unlabeled video data. This would endow the model with a more grounded understanding of motion, causality, and object permanence, providing a stronger perceptual foundation for high-level video reasoning and multimodal temporal inference.

Autonomous Discovery of Visual Structures. While approaches in this thesis (*e.g.*, LogicSeg) rely on predefined semantic hierarchies, a truly intelligent system should be able to discover these structures autonomously. Foundation models, endowed with vast world knowledge, are ideally suited for this purpose. Instead of manually specifying a class taxonomy, one could prompt a foundation model with a set of visual exemplars and ask it to propose a plausible hierarchy. The model might autonomously produce structures such as vehicle→car→wheel or vehicle→two-wheeler→motorcycle. Such emergent structures could then be used as symbolic backbones for structured perception frameworks, forming a self-reinforcing loop where foundation models hypothesize structure, and structured perception methods enforce and refine it. Furthermore, the commonsense rules used in LogicHOI were manually formulated. Future systems could instead mine such rules automatically by querying LMMs. For instance: “What actions are possible with

a kite?" or "What is the typical spatial relationship between a human and a surfboard when surfing?" The responses could be parsed into probabilistic logic rules, yielding a dynamic and scalable knowledge base that evolves alongside the model's understanding. This would transform current frameworks from static, rule-driven systems into self-learning neuro-symbolic agents capable of acquiring and reasoning with new knowledge autonomously.

Advancing Interpretability through Learned Structures. As AI systems move into safety-critical domains such as autonomous driving and medical imaging, ensuring that predictions are not only accurate but also interpretable and trustworthy becomes essential. Structured perception provides a direct pathway toward this goal. Specifically, rather than relying on post-hoc explanation tools (*e.g.*, saliency maps), future models can be designed to be interpretable by construction. The structures learned in this thesis provide a natural explanatory medium. For LogicSeg, a model could justify its output by stating "This pixel was classified as wheel and, by the logical rule C, must also belong to vehicle." For LogicHOI, the system might explain "The interaction human rides horse was predicted because the human is positioned above the horse, consistent with the learned proxemics rule for ride." Conversely, it could reject implausible interactions such as human holds car by appealing to learned affordance constraints. Moreover, the current focus on relational structures can be extended toward causal representations of visual scenes. Leveraging large-scale video data through the temporal-correspondence frameworks in Part I, future systems could learn intuitive physics and event causality. For example, understanding that "a hand releasing a ball" (cause) precedes "the ball falling" (effect). Integrating such causal knowledge into neuro-symbolic frameworks would enable models not only to describe what is happening, but to reason about why and what might happen next, which is a crucial step toward genuine visual intelligence.

9.3 Final Remarks

My long-term research vision is to develop AI systems that hold human-like visual intelligence. I believe that by endowing machines with the ability to see beyond mere collections of pixels, to grasp the relationships, dynamics, and underlying logic within visual scenes, we can unlock new frontiers in artificial intelligence, leading to systems that are not only more capable, robust, and generalizable, but also more closely aligned with human understanding.

The journey detailed in this thesis, from modeling temporal dynamics to understanding structured scenes and integrating explicit knowledge, represents foundational steps towards this vision. I also envision a near future where AI can autonomously learn and reason with diverse structural representations, and continuously refine its understanding of the world from the vast mount of visual data it encounters. This pursuit is driven by the belief that structured visual perception is a cornerstone of true visual intelligence.

This shift towards structured perception represent a solution that challenges the current AI paradigm: **i)** by making the relational and logical constraints explicit, the reasoning of model becomes inspectable, where its decisions are no longer solely the output of a black box, but are grounded in a structure that can be audited and understood by humans; **ii)** by forcing models to learn and reason with explicit structures, we move them away from memorizing patterns and toward a more human-like ability to compose known concepts into new, coherent thoughts; **iii)** an AI system that understands the rules of the visual world should not need to see a million examples to learn a new concept, since structured priors serve as a powerful form of inductive bias, enabling more efficient learning from limited data; **iv)** this research, therefore, also serves as a bridge between artificial intelligence and computational cognitive science. By building systems that attempt to model the hierarchical, re-

lational, and logical way that humans perceive the world, we are creating testable, computational hypotheses about the nature of cognition itself.

As AI continues to evolve and become more deeply integrated into society, it is essential that these advancements serve to enhance human capabilities and well-being. The development of AI systems capable of robustly interpreting complex visual environments holds tremendous potential, ranging from enabling more reliable autonomous systems to scientific discovery and creative innovation. My aspiration is that, by striving to build AI that can "see" and "understand" the world in a structured, more human-like way, we not only push the boundaries of machine intelligence but also gain deeper insights into the remarkable capacities of human visual perception and cognition. Ultimately, the goal is not limited to replicate human intelligence, but advancing toward the creation of complementary intelligent systems that augment our abilities and enrich our interaction with the world.

Bibliography

[1] Aishwarya Agrawal, Aniruddha Kembhavi, Dhruv Batra, and Devi Parikh. C-vqa: A compositional split of the visual question answering (vqa) v1. 0 dataset. *arXiv preprint arXiv:1704.08243*, 2017.

[2] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *ICCV*, 2015.

[3] Ekin Akyürek, Afra Feyza Akyürek, and Jacob Andreas. Learning to recombine and resample data for compositional generalization. In *ICLR*, 2021.

[4] Forough Arabshahi, Sameer Singh, and Animashree Anandkumar. Combining symbolic expressions and black-box function evaluations in neural programs. In *ICLR*, 2018.

[5] Nikita Araslanov, Simone Schaub-Meyer, and Stefan Roth. Dense unsupervised learning for video segmentation. In *NeurIPS*, 2021.

[6] Yuval Atzmon, Jonathan Berant, Vahid Kezami, Amir Globerson, and Gal Chechik. Learning to generalize to new compositions in image understanding. *arXiv preprint arXiv:1608.07639*, 2016.

[7] S Avinash Ramakanth and R Venkatesh Babu. Seamseg: Video object segmentation using patch seams. In *CVPR*, 2014.

[8] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022.

[9] Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic tensor networks. *Artificial Intelligence*, 303:103649, 2022.

[10] Vijay Badrinarayanan, Fabio Galasso, and Roberto Cipolla. Label propagation in video sequences. In *CVPR*, 2010.

[11] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE TPAMI*, 39(12):2481–2495, 2017.

[12] Dzmitry Bahdanau, Shikhar Murty, Michael Noukhovitch, Thien Huu Nguyen, Harm de Vries, and Aaron Courville. Systematic generalization: What is required and can it be learned? In *ICLR*, 2019.

[13] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 92(1):1–31, 2011.

[14] Ann Marie Barry. *Visual intelligence: Perception, image, and manipulation in visual communication*. Suny Press, 1997.

[15] Luca Bertinetto, Romain Mueller, Konstantinos Tertikas, Sina Samangooei, and Nicholas A Lord. Making better mistakes: Leveraging class hierarchies with deep networks. In *CVPR*, 2020.

[16] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *ECCV*, 2016.

[17] Goutam Bhat, Felix Järemo Lawin, Martin Danelljan, Andreas Robinson, Michael Felsberg, Luc Van Gool, and Radu Timofte. Learning what to learn for video object segmentation. In *ECCV*, 2020.

[18] Wei Bi and James T Kwok. Multilabel classification on tree-and dag-structured hierarchies. In *ICML*, 2011.

[19] Zhangxing Bian, Allan Jabri, Alexei A Efros, and Andrew Owens. Learning pixel trajectories with multiscale contrastive random walks. *arXiv preprint*

arXiv:2201.08379, 2022.

[20] Johannes Bill, Hrag Pailian, Samuel J Gershman, and Jan Drugowitsch. Hierarchical structure is employed by humans during visual motion perception. *Proceedings of the National Academy of Sciences*, 117(39):24581–24589, 2020.

[21] Rodney A Brooks. Intelligence without representation. *Artificial intelligence*, 47(1-3):139–159, 1991.

[22] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *CVPR*, 2017.

[23] Yichao Cao, Qingfei Tang, Feng Yang, Xiu Su, Shan You, Xiaobo Lu, and Chang Xu. Re-mine, learn and reason: Exploring the cross-modal semantic correlations for language-guided hoi detection. In *ICCV*, 2023.

[24] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[25] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

[26] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

[27] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, 2018.

[28] Guikun Chen, Jin Li, and Wenguan Wang. Scene graph generation with role-playing large language models. In *NeurIPS*, 2024.

[29] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional

nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 2017.

[30] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[31] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *CVPR*, 2016.

[32] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018.

[33] Minghan Chen, Guikun Chen, Wenguan Wang, and Yi Yang. Hydra-sgg: Hybrid relation assignment for one-stage scene graph generation. *arXiv preprint arXiv:2409.10262*, 2024.

[34] Mu Chen, Liulei Li, Wenguan Wang, Ruijie Quan, and Yi Yang. General and task-oriented video segmentation. In *ECCV*, 2024.

[35] Mu Chen, Liulei Li, Wenguan Wang, and Yi Yang. Diffvsgg: Diffusion-driven online video scene graph generation. In *CVPR*, 2025.

[36] Mingfei Chen, Yue Liao, Si Liu, Zhiyuan Chen, Fei Wang, and Chen Qian. Reformulating hoi detection as adaptive set prediction. In *CVPR*, 2021.

[37] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

[38] 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 *CVPR*, 2014.

[39] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *CVPR*, 2018.

[40] Bowen Cheng, Liang-Chieh Chen, Yunchao Wei, Yukun Zhu, Zilong Huang, Jinjun Xiong, Thomas S Huang, Wen-Mei Hwu, and Honghui Shi. Spgnet: Semantic prediction guidance for scene parsing. In *ICCV*, 2019.

[41] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022.

[42] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021.

[43] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *CVPR*, 2021.

[44] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. In *NeurIPS*, 2021.

[45] Yangming Cheng, Liulei Li, Yuanyou Xu, Xiaodi Li, Zongxin Yang, Wenguan Wang, and Yi Yang. Segment and track anything. *arXiv preprint arXiv:2305.06558*, 2023.

[46] Junsuk Choe, Seong Joon Oh, Seungho Lee, Sanghyuk Chun, Zeynep Akata, and Hyunjung Shim. Evaluating weakly supervised object localization methods right. In *CVPR*, 2020.

[47] Sungha Choi, Joanne T Kim, and Jaegul Choo. Cars can't fly up in the sky: Improving urban-scene segmentation via height-driven attention networks. In *CVPR*, 2020.

[48] Hai Ci, Chunyu Wang, and Yizhou Wang. Video object segmentation by learning location-sensitive embeddings. In *ECCV*, 2018.

[49] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus

Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[50] Miles Cranmer, Alvaro Sanchez Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho. Discovering symbolic models from deep learning with inductive biases. In *NeurIPS*, 2020.

[51] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, 2013.

[52] Hanjun Dai, Chengtao Li, Connor Coley, Bo Dai, and Le Song. Retrosynthesis prediction with conditional graph logic network. In *NeurIPS*, 2019.

[53] Wang-Zhou Dai, Qiuling Xu, Yang Yu, and Zhi-Hua Zhou. Bridging machine learning and logical reasoning by abductive learning. In *NeurIPS*, 2019.

[54] Luc De Raedt, Sebastijan Dumančić, Robin Manhaeve, and Giuseppe Marra. From statistical relational to neural-symbolic artificial intelligence. In *IJCAI*, 2021.

[55] Ofer Dekel, Joseph Keshet, and Yoram Singer. Large margin hierarchical classification. In *ICML*, 2004.

[56] 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 *ECCV*, 2014.

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

[58] Michelangelo Diligenti, Marco Gori, and Claudio Sacca. Semantic-based regularization for learning and inference. *Artificial Intelligence*, 244:143–165, 2017.

[59] Yuhang Ding, Liulei Li, Wenguan Wang, and Yi Yang. Clustering propagation for universal medical image segmentation. In *CVPR*, 2024.

[60] Leizhen Dong, Zhimin Li, Kunlun Xu, Zhijun Zhang, Luxin Yan, Sheng Zhong, and Xu Zou. Category-aware transformer network for better human-object interaction detection. In *CVPR*, 2022.

[61] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G Heigold, S Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020.

[62] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015.

[63] Brendan Duke, Abdalla Ahmed, Christian Wolf, Parham Aarabi, and Graham W Taylor. Sstvos: Sparse spatiotemporal transformers for video object segmentation. In *CVPR*, 2021.

[64] Pavel Dvurechensky, Alexander Gasnikov, and Alexey Kroshnin. Computational optimal transport: Complexity by accelerated gradient descent is better than by sinkhorn's algorithm. In *ICLR*, 2018.

[65] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *CVPR*, 2019.

[66] Alon Faktor and Michal Irani. Video segmentation by non-local consensus voting. In *BMVC*, 2014.

[67] Lifeng Fan, Wenguan Wang, Siyuan Huang, Xinyu Tang, and Song-Chun Zhu. Understanding human gaze communication by spatio-temporal graph reasoning. In *ICCV*, 2019.

[68] Qingnan Fan, Fan Zhong, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Jumpcut: non-successive mask transfer and interpolation for video cutout. *IEEE TOG*, 34(6):195–1, 2015.

[69] Hao-Shu Fang, Yichen Xie, Dian Shao, and Cewu Lu. Dirv: Dense interaction region voting for end-to-end human-object interaction detection. In *AAAI*, 2021.

[70] Solomon Feferman, John W Dawson Jr, Stephen C Kleene, Gregory H Moore, Robert M Solovay, and Jean Van Heijenoort. *Kurt Godel: collected works. Vol. 1: Publications 1929-1936*. Oxford University Press, Inc., 1986.

[71] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.

[72] Marc Fischer, Mislav Balunovic, Dana Drachsler-Cohen, Timon Gehr, Ce Zhang, and Martin Vechev. Dl2: Training and querying neural networks with logic. In *ICML*, 2019.

[73] Peter A Flach and Antonis C Kakas. *Abduction and Induction: Essays on their relation and integration*, volume 18. Springer Science & Business Media, 2000.

[74] Jerry A Fodor and Zenon W Pylyshyn. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71, 1988.

[75] Steven M Frankland and Joshua D Greene. Concepts and compositionality: in search of the brain's language of thought. *Annual Review of Psychology*, 71:273–303, 2020.

[76] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *CVPR*, 2019.

[77] Daniel Furrer, Marc van Zee, Nathan Scales, and Nathanael Schärli. Compositional generalization in semantic parsing: Pre-training vs. specialized architectures. *arXiv preprint arXiv:2007.08970*, 2020.

[78] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano,

Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*, 2022.

[79] Chuang Gan, Boqing Gong, Kun Liu, Hao Su, and Leonidas J Guibas. Geometry guided convolutional neural networks for self-supervised video representation learning. In *CVPR*, 2018.

[80] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *ICML*, 2018.

[81] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. Drg: Dual relation graph for human-object interaction detection. In *ECCV*, 2020.

[82] Chen Gao, Yuliang Zou, and Jia-Bin Huang. ican: Instance-centric attention network for human-object interaction detection. In *BMVC*, 2018.

[83] Artur d'Avila Garcez, Sebastian Bader, Howard Bowman, Luis C Lamb, Leo de Penning, BV Illuminoo, Hoifung Poon, and COPPE Gerson Zaverucha. Neural-symbolic learning and reasoning: A survey and interpretation. *Neuro-Symbolic Artificial Intelligence: The State of the Art*, 342:1, 2022.

[84] Artur d'Avila Garcez, Marco Gori, Luis C Lamb, Luciano Serafini, Michael Spranger, and Son N Tran. Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. *arXiv preprint arXiv:1905.06088*, 2019.

[85] Artur S d'Avila Garcez, Tarek R Besold, Luc De Raedt, Peter Földiak, Pascal Hitzler, Thomas Icard, Kai-Uwe Kühnberger, Luis C Lamb, Risto Miikkulainen, and Daniel L Silver. Neural-symbolic learning and reasoning: Contributions and challenges. In *AAAI*, 2015.

[86] Vivien Sainte Fare Garnot and Loic Landrieu. Leveraging class hierarchies with metric-guided prototype learning. *arXiv preprint arXiv:2007.03047*, 2020.

[87] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *2011 IEEE Intelligent Vehicles Symposium*, 2011.

[88] Sergei Gepshtein and Michael Kubovy. The emergence of visual objects in space–time. *Proceedings of the National Academy of Sciences*, 97(14):8186–8191, 2000.

[89] Ross Girshick. Fast r-cnn. In *ICCV*, 2015.

[90] Eleonora Giunchiglia and Thomas Lukasiewicz. Coherent hierarchical multi-label classification networks. In *NeurIPS*, 2020.

[91] Eleonora Giunchiglia and Thomas Lukasiewicz. Multi-label classification neural networks with hard logical constraints. *Journal of Artificial Intelligence Research*, 72:759–818, 2021.

[92] Eleonora Giunchiglia, Mihaela Catalina Stoian, and Thomas Lukasiewicz. Deep learning with logical constraints. In *IJCAI*, 2022.

[93] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *CVPR*, 2018.

[94] Joseph A Goguen. The logic of inexact concepts. *Synthese*, pages 325–373, 1969.

[95] Ke Gong, Yiming Gao, Xiaodan Liang, Xiaohui Shen, Meng Wang, and Liang Lin. Graphonomy: Universal human parsing via graph transfer learning. In *CVPR*, 2019.

[96] Abel Gonzalez-Garcia, Davide Modolo, and Vittorio Ferrari. Do semantic parts emerge in convolutional neural networks? *IJCV*, 126(5):476–494, 2018.

[97] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *CVPR*, 2022.

[98] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. In *NeurIPS*, 2022.

[99] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015.

[100] Tanmay Gupta, Alexander Schwing, and Derek Hoiem. No-frills human-object interaction detection: Factorization, layout encodings, and training techniques. In *ICCV*, 2019.

[101] Petr Hájek. *Metamathematics of fuzzy logic*, volume 4. Springer Science & Business Media, 2013.

[102] Feng Han and Song-Chun Zhu. Bottom-up/top-down image parsing with attribute grammar. *IEEE TPAMI*, 31(1):59–73, 2008.

[103] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020.

[104] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *NeurIPS*, 2020.

[105] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[106] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[107] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2022.

[108] Jonathan Herzig and Jonathan Berant. Span-based semantic parsing for compositional generalization. In *ACL*, 2021.

[109] Robert F Hess and Ian E Holliday. The coding of spatial position by the human visual system: effects of spatial scale and contrast. *Vision Research*, 32(6):1085–1097, 1992.

[110] Zhi Hou, Xiaojiang Peng, Yu Qiao, and Dacheng Tao. Visual compositional learning for human-object interaction detection. In *ECCV*, 2020.

[111] Zhi Hou, Baosheng Yu, Yu Qiao, Xiaojiang Peng, and Dacheng Tao. Affordance transfer learning for human-object interaction detection. In *CVPR*, 2021.

[112] Zhi Hou, Baosheng Yu, Yu Qiao, Xiaojiang Peng, and Dacheng Tao. Detecting human-object interaction via fabricated compositional learning. In *CVPR*, 2021.

[113] Zhi Hou, Baosheng Yu, and Dacheng Tao. Discovering human-object interaction concepts via self-compositional learning. In *ECCV*, 2022.

[114] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G Schwing. Videomatch: Matching based video object segmentation. In *ECCV*, 2018.

[115] Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. Harnessing deep neural networks with logic rules. In *ACL*, 2016.

[116] Zhengdong Hu, Yifan Sun, and Yi Yang. Switch to generalize: Domain-switch learning for cross-domain few-shot classification. In *ICLR*, 2022.

[117] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019.

[118] Jarmo Hurri and Aapo Hyvärinen. Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Computation*, 15(3):663–691, 2003.

[119] ASM Iftekhar, Hao Chen, Kaustav Kundu, Xinyu Li, Joseph Tighe, and Davide Modolo. What to look at and where: Semantic and spatial refined transformer for detecting human-object interactions. In *CVPR*, 2022.

[120] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017.

[121] Md Amirul Islam, Sen Jia, and Neil DB Bruce. How much position information do convolutional neural networks encode? In *ICLR*, 2020.

[122] Allan Jabri, Andrew Owens, and Alexei A Efros. Space-time correspondence as a contrastive random walk. In *NeurIPS*, 2020.

[123] Suyog Dutt Jain and Kristen Grauman. Supervoxel-consistent foreground propagation in video. In *ECCV*, 2014.

[124] Varun Jampani, Raghudeep Gadde, and Peter V Gehler. Video propagation networks. In *CVPR*, 2017.

[125] Joel Janai, Fatma Guney, Anurag Ranjan, Michael Black, and Andreas Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *ECCV*, 2018.

[126] Theo MV Janssen et al. Compositionality: Its historic context. *The Oxford handbook of compositionality*, pages 19–46, 2012.

[127] Theo MV Janssen and Barbara H Partee. Compositionality. In *Handbook of logic and language*, pages 417–473. 1997.

[128] Sangryul Jeon, Dongbo Min, Seungryong Kim, and Kwanghoon Sohn. Mining better samples for contrastive learning of temporal correspondence. In *CVPR*, 2021.

[129] Hueihan Jhuang, Juergen Gall, Silvia Zuffi, Cordelia Schmid, and Michael J Black. Towards understanding action recognition. In *ICCV*, 2013.

[130] 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 *ECCV*, 2020.

[131] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. In *CVPR*, 2023.

[132] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017.

[133] Daniel Kahneman. *Thinking, fast and slow*. Macmillan, 2011.

[134] Daniel Kaiser, Genevieve L Quek, Radoslaw M Cichy, and Marius V Peelen. Object vision in a structured world. *Trends in cognitive sciences*, 23(8):672–685, 2019.

[135] Fumi Katsuki and Christos Constantinidis. Bottom-up and top-down attention: different processes and overlapping neural systems. *The Neuroscientist*, 20(5):509–521, 2014.

[136] Henry Kautz. The third AI summer: AAAI robert s. engelmore memorial lecture. *AI Magazine*, 43(1):93–104, 2022.

[137] Osman Semih Kayhan and Jan C van Gemert. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In *CVPR*, 2020.

[138] Tsung-Wei Ke, Jyh-Jing Hwang, Yunhui Guo, Xudong Wang, and Stella X Yu. Unsupervised hierarchical semantic segmentation with multiview cosegmentation and clustering transformers. In *CVPR*, 2022.

[139] Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, et al. Measuring compositional generalization: A comprehensive

method on realistic data. In *ICLR*, 2020.

[140] Bumsoo Kim, Taeho Choi, Jaewoo Kang, and Hyunwoo J Kim. Uniondet: Union-level detector towards real-time human-object interaction detection. In *ECCV*, 2020.

[141] Bumsoo Kim, Junhyun Lee, Jaewoo Kang, Eun-Sol Kim, and Hyunwoo J Kim. Hotr: End-to-end human-object interaction detection with transformers. In *CVPR*, 2021.

[142] Bumsoo Kim, Jonghwan Mun, Kyoung-Woon On, Minchul Shin, Junhyun Lee, and Eun-Sol Kim. Mstr: Multi-scale transformer for end-to-end human-object interaction detection. In *CVPR*, 2022.

[143] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.

[144] Dong-Jin Kim, Xiao Sun, Jinsoo Choi, Stephen Lin, and In So Kweon. Detecting human-object interactions with action co-occurrence priors. In *ECCV*, 2020.

[145] Najoung Kim and Tal Linzen. Cogs: A compositional generalization challenge based on semantic interpretation. In *EMNLP*, 2020.

[146] Youngeun Kim, Seokeon Choi, Hankyeol Lee, Taekyung Kim, and Changick Kim. Rpm-net: Robust pixel-level matching networks for self-supervised video object segmentation. In *WACV*, 2020.

[147] Filippos Kokkinos and Iasonas Kokkinos. To the point: correspondence-driven monocular 3d category reconstruction. In *NeurIPS*, 2021.

[148] Daphne Koller and Mehran Sahami. Hierarchically classifying documents using very few words. In *ICML*, 1997.

[149] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[150] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *ICLR*, 2022.

[151] Zihang Lai, Erika Lu, and Weidi Xie. Mast: A memory-augmented self-supervised tracker. In *CVPR*, 2020.

[152] Zihang Lai and Weidi Xie. Self-supervised learning for video correspondence flow. In *BMVC*, 2019.

[153] Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *ICML*, 2018.

[154] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

[155] Guillaume Lample and François Charton. Deep learning for symbolic mathematics. In *ICLR*, 2019.

[156] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[157] Benjamin Lefaudeux, Francisco Massa, Diana Liskovich, Wenhan Xiong, Vittorio Caggiano, Sean Naren, Min Xu, Jieru Hu, Marta Tintore, Susan Zhang, Patrick Labatut, Daniel Haziza, Luca Wehrstedt, Jeremy Reizenstein, and Grigory Sizov. xformers: A modular and hackable transformer modelling library. https://github.com/facebookresearch/xformers, 2022.

[158] Ting Lei, Fabian Caba, Qingchao Chen, Hailin Jin, Yuxin Peng, and Yang Liu. Efficient adaptive human-object interaction detection with concept-guided memory. In *ICCV*, 2023.

[159] Ting Lei, Shaofeng Yin, and Yang Liu. Exploring the potential of large foundation models for open-vocabulary hoi detection. In *CVPR*, 2024.

[160] Bergen Leon, Timothy O'Donnell, and Dzmitry Bahdanau. Systematic generalization with edge transformers. In *NeurIPS*, 2021.

[161] Hanchao Li, Pengfei Xiong, Jie An, and Lingxue Wang. Pyramid attention network for semantic segmentation. *arXiv preprint arXiv:1805.10180*, 2018.

[162] Liulei Li, Wenguan Wang, and Yi Yang. Logicseg: Parsing visual semantics with neural logic learning and reasoning. In *ICCV*, 2023.

[163] Liulei Li, Wenguan Wang, and Yi Yang. Human-object interaction detection collaborated with large relation-driven diffusion models. In *NeurIPS*, 2024.

[164] Liulei Li, Wenguan Wang, Tianfei Zhou, Jianwu Li, and Yi Yang. Unified mask embedding and correspondence learning for self-supervised video segmentation. In *CVPR*, 2023.

[165] Liulei Li, Wenguan Wang, Tianfei Zhou, Ruijie Quan, and Yi Yang. Semantic hierarchy-aware segmentation. *IEEE TPAMI*, 46(4):2123–2138, 2023.

[166] Liulei Li, Jianan Wei, Wenguan Wang, and Yi Yang. Neural-logic human-object interaction detection. In *NeurIPS*, 2023.

[167] Liulei Li, Tianfei Zhou, Wenguan Wang, Jianwu Li, and Yi Yang. Deep hierarchical semantic segmentation. In *CVPR*, 2022.

[168] Liulei Li, Tianfei Zhou, Wenguan Wang, Lu Yang, Jianwu Li, and Yi Yang. Locality-aware inter-and intra-video reconstruction for self-supervised correspondence learning. In *CVPR*, 2022.

[169] Li-Jia Li, Richard Socher, and Li Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *CVPR*, 2009.

[170] Qing Li, Siyuan Huang, Yining Hong, Yixin Chen, Ying Nian Wu, and Song-Chun Zhu. Closed loop neural-symbolic learning via integrating neural perception, grammar parsing, and symbolic reasoning. In *ICML*, 2020.

[171] Shan Li, Lu Yang, Pu Cao, Liulei Li, and Huadong Ma. Frequency-based matcher for long-tailed semantic segmentation. *IEEE TMM*, 26:10395–10405, 2024.

[172] Xueting Li, Sifei Liu, Shalini De Mello, Xiaolong Wang, Jan Kautz, and Ming-Hsuan Yang. Joint-task self-supervised learning for temporal correspondence. In *NeurIPS*, 2019.

[173] Yong-Lu Li, Xinpeng Liu, Han Lu, Shiyi Wang, Junqi Liu, Jiefeng Li, and Cewu Lu. Detailed 2d-3d joint representation for human-object interaction. In *CVPR*, 2020.

[174] Yong-Lu Li, Xinpeng Liu, Xiaoqian Wu, Yizhuo Li, and Cewu Lu. Hoi analysis: Integrating and decomposing human-object interaction. In *NeurIPS*, 2020.

[175] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*, 2019.

[176] Zhiheng Li, Wenxuan Bao, Jiayang Zheng, and Chenliang Xu. Deep grouping model for unified perceptual parsing. In *CVPR*, 2020.

[177] Chen Liang, Wenguan Wang, Jiaxu Miao, and Yi Yang. Gmmseg: Gaussian mixture based generative semantic segmentation models. In *NeurIPS*, 2022.

[178] Xiaodan Liang, Ke Gong, Xiaohui Shen, and Liang Lin. Look into person: Joint body parsing & pose estimation network and a new benchmark. *IEEE TPAMI*, 41(4):871–885, 2018.

[179] Xiaodan Liang, Hongfei Zhou, and Eric Xing. Dynamic-structured semantic propagation network. In *CVPR*, 2018.

[180] Yongqing Liang, Xin Li, Navid Jafari, and Qin Chen. Video object segmentation with adaptive feature bank and uncertain-region refinement. In *NeurIPS*,

2020.

[181] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, Chen Qian, and Jiashi Feng. Ppdm: Parallel point detection and matching for real-time human-object interaction detection. In *CVPR*, 2020.

[182] Yue Liao, Aixi Zhang, Miao Lu, Yongliang Wang, Xiaobo Li, and Si Liu. Gen-vlkt: Simplify association and enhance interaction understanding for hoi detection. In *CVPR*, 2022.

[183] Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *EMNLP*, 2019.

[184] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023.

[185] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[186] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

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

[188] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *ECCV*, 2022.

[189] Xinpeng Liu, Yong-Lu Li, Xiaoqian Wu, Yu-Wing Tai, Cewu Lu, and Chi-Keung Tang. Interactiveness field in human-object interactions. In *CVPR*,

2022.

[190] Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. More control for free! image synthesis with semantic diffusion guidance. In *WACV*, 2023.

[191] Xinchen Liu, Meng Zhang, Wu Liu, Jingkuan Song, and Tao Mei. Braidnet: Braiding semantics and details for accurate human parsing. In *ACM MM*, 2019.

[192] Yang Liu, Qingchao Chen, and Andrew Zisserman. Amplifying key cues for human-object-interaction detection. In *ECCV*, 2020.

[193] Yuan Liu, Ruoteng Li, Yu Cheng, Robby T Tan, and Xiubao Sui. Object tracking using spatio-temporal networks for future prediction location. In *ECCV*, 2020.

[194] 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 *ICCV*, 2021.

[195] Margaret S Livingstone and David H Hubel. Psychophysical evidence for separate channels for the perception of form, color, movement, and depth. *Journal of Neuroscience*, 7(11):3416–3468, 1987.

[196] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

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

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

[199] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint*

arXiv:1605.08104, 2016.

[200] João Loula, Marco Baroni, and Brenden Lake. Rearranging the familiar: Testing compositional generalization in recurrent networks. In *EMNLP*, 2018.

[201] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. In *CVPR*, 2018.

[202] Xiankai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. In *ECCV*, 2020.

[203] Xiankai Lu, Wenguan Wang, Jianbing Shen, David Crandall, and Luc Van Gool. Segmenting objects from relational visual data. *IEEE TPAMI*, 44(11):7885–7897, 2021.

[204] Xiankai Lu, Wenguan Wang, Jianbing Shen, Yu-Wing Tai, David J Crandall, and Steven CH Hoi. Learning video object segmentation from unlabeled videos. In *CVPR*, 2020.

[205] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *ACCV*, 2018.

[206] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. In *NeurIPS*, 2023.

[207] Yawei Luo, Zhedong Zheng, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Macro-micro adversarial network for human parsing. In *ECCV*, 2018.

[208] Daoming Lyu, Fangkai Yang, Bo Liu, and Steven Gustafson. Sdrl: interpretable and data-efficient deep reinforcement learning leveraging symbolic planning. In *AAAI*, 2019.

[209] Shuailei Ma, Yuefeng Wang, Shanze Wang, and Ying Wei. Fgahoi: Fine-grained anchors for human-object interaction detection. *IEEE TPAMI*, 46(4):2415–2429, 2023.

[210] Zixian Ma, Jerry Hong, Mustafa Omer Gul, Mona Gandhi, Irena Gao, and Ranjay Krishna. Crepe: Can vision-language foundation models reason compositionally? In *CVPR*, 2023.

[211] K-K Maninis, Sergi Caelles, Yuhua Chen, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. Video object segmentation without temporal information. *IEEE TPAMI*, 2018.

[212] Yunyao Mao, Jiajun Deng, Wengang Zhou, Li Li, Yao Fang, and Houqiang Li. Clip4hoi: towards adapting clip for practical zero-shot hoi detection. In *NeurIPS*, 2023.

[213] Gary Marcus. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631*, 2018.

[214] Yoshihiro Maruyama. Symbolic and statistical theories of cognition: towards integrated artificial intelligence. In *International Conference on Software Engineering and Formal Methods*, 2020.

[215] Andrew McCallum, Ronald Rosenfeldy, Tom Mitchelly, and Andrew Y Ngz. Improving text classification by shrinkage in a hierarchy of classes. In *ICML*, 1998.

[216] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133, 1943.

[217] Tim Meinhardt and Laura Leal-Taixé. Make one-shot video object segmentation efficient again. In *NeurIPS*, 2020.

[218] Simon Meister, Junhwa Hur, and Stefan Roth. Unflow: Unsupervised learning

of optical flow with a bidirectional census loss. In *AAAI*, 2018.

[219] Panagiotis Meletis and Gijs Dubbelman. Training of convolutional networks on multiple heterogeneous datasets for street scene semantic segmentation. In *IEEE Intelligent Vehicles Symposium*, 2018.

[220] Jiaxu Miao, Yunchao Wei, and Yi Yang. Memory aggregation networks for efficient interactive video object segmentation. In *CVPR*, 2020.

[221] Umberto Michieli, Edoardo Borsato, Luca Rossi, and Pietro Zanuttigh. Gmnet: Graph matching network for large scale part semantic segmentation in the wild. In *ECCV*, 2020.

[222] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020.

[223] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.

[224] David Milner and Mel Goodale. *The visual brain in action*, volume 27. Oup Oxford, 2006.

[225] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.

[226] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *ICML*, 2009.

[227] Lili Mou, Zhengdong Lu, Hang Li, and Zhi Jin. Coupling distributed and symbolic execution for natural language queries. In *ICML*, 2017.

[228] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017.

[229] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*, 2017.

[230] Shan Ning, Longtian Qiu, Yongfei Liu, and Xuming He. Hoiclip: Efficient knowledge transfer for hoi detection with vision-language models. In *CVPR*, 2023.

[231] Vilém Novák, Irina Perfilieva, and Jiri Mockor. *Mathematical principles of fuzzy logic*, volume 517. Springer Science & Business Media, 2012.

[232] David Novotny, Samuel Albanie, Diane Larlus, and Andrea Vedaldi. Self-supervised learning of geometrically stable features through probabilistic introspection. In *CVPR*, 2018.

[233] Maxwell Nye, Armando Solar-Lezama, Josh Tenenbaum, and Brenden M Lake. Learning compositional rules via neural program synthesis. In *NeurIPS*, 2020.

[234] Pedro O O Pinheiro, Amjad Almahairi, Ryan Benmalek, Florian Golemo, and Aaron C Courville. Unsupervised learning of dense visual representations. *NeurIPS*, 2020.

[235] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *CVPR*, 2018.

[236] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019.

[237] Lauri Oksama and Jukka Hyönä. Position tracking and identity tracking are separate systems: Evidence from eye movements. *Cognition*, 146:393–409, 2016.

[238] Peter Pagin and Dag Westerståhl. Compositionality i: Definitions and variants. *Philosophy Compass*, 5(3):250–264, 2010.

[239] George Pantazopoulos, Alessandro Suglia, and Arash Eshghi. Combine to

describe: Evaluating compositional generalization in image captioning. In *ACL*, 2022.

[240] Emilio Parisotto, Abdel-rahman Mohamed, Rishabh Singh, Lihong Li, Dengyong Zhou, and Pushmeet Kohli. Neuro-symbolic program synthesis. In *ICLR*, 2017.

[241] Jihwan Park, SeungJun Lee, Hwan Heo, Hyeong Kyu Choi, and Hyunwoo J Kim. Consistency learning via decoding path augmentation for transformers in human object interaction detection. In *CVPR*, 2022.

[242] Jeeseung Park, Jin-Woo Park, and Jong-Seok Lee. Viplo: Vision transformer based pose-conditioned self-loop graph for human-object interaction detection. In *CVPR*, 2023.

[243] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017.

[244] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, 2017.

[245] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016.

[246] Tony A Plate. Holographic reduced representations. *IEEE TNN*, 6(3):623–641, 1995.

[247] Jordan B Pollack. Recursive distributed representations. *Artificial Intelligence*, 46(1-2):77–105, 1990.

[248] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2022.

[249] Lorenzo Porzi, Samuel Rota Bulo, Aleksander Colovic, and Peter Kontschieder. Seamless scene segmentation. In *CVPR*, 2019.

[250] Sarah Pratt, Mark Yatskar, Luca Weihs, Ali Farhadi, and Aniruddha Kembhavi. Grounded situation recognition. In *ECCV*, 2020.

[251] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *CVPR*, 2022.

[252] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018.

[253] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*, 2021.

[254] Xian Qu, Changxing Ding, Xingao Li, Xubin Zhong, and Dacheng Tao. Distillation using oracle queries for transformer-based human-object interaction detection. In *CVPR*, 2022.

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

[256] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[257] Stephen K Reed. A taxonomic analysis of abstraction. *Perspectives on Psychological Science*, 11(6):817–837, 2016.

[258] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: To-

wards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[259] Lynn C Robertson and Marvin R Lamb. Neuropsychological contributions to theories of part/whole organization. *Cognitive psychology*, 23(2):299–330, 1991.

[260] Andreas Robinson, Felix Jaremo Lawin, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. Learning fast and robust target models for video object segmentation. In *CVPR*, 2020.

[261] Tim Rocktäschel and Sebastian Riedel. End-to-end differentiable proving. In *NeurIPS*, 2017.

[262] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[263] Tao Ruan, Ting Liu, Zilong Huang, Yunchao Wei, Shikui Wei, and Yao Zhao. Devil in the details: Towards accurate single and multiple human parsing. In *AAAI*, 2019.

[264] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

[265] Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *ICML*, 2018.

[266] Scott Satkin and Martial Hebert. 3dnn: Viewpoint invariant 3d geometry matching for scene understanding. In *ICCV*, 2013.

[267] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[268] Marwin HS Segler, Mike Preuss, and Mark P Waller. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature*, 555(7698):604–610, 2018.

[269] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *ECCV*, 2020.

[270] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *CVPR*, 2020.

[271] Lokendra Shastri and Venkat Ajjanagadde. From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences*, 16(3):417–451, 1993.

[272] Peter Shaw, Ming-Wei Chang, Panupong Pasupat, and Kristina Toutanova. Compositional generalization and natural language variation: Can a semantic parsing approach handle both? In *ACL*, 2021.

[273] Rishubh Singh, Pranav Gupta, Pradeep Shenoy, and Ravikiran Sarvadevabhatla. Float: Factorized learning of object attributes for improved multi-object multi-part scene parsing. In *CVPR*, 2022.

[274] Paul Smolensky, R Thomas McCoy, Roland Fernandez, Matthew Goldrick, and Jianfeng Gao. *Neurocompositional computing in human and machine intelligence: A tutorial*. Microsoft Technical Report MSR-TR-2022, 2022.

[275] Jeany Son. Contrastive learning for space-time correspondence via self-cycle consistency. In *CVPR*, 2022.

[276] Jie Song, Limin Wang, Luc Van Gool, and Otmar Hilliges. Thin-slicing network: A deep structured model for pose estimation in videos. In *CVPR*, 2017.

[277] Elizabeth S Spelke and Katherine D Kinzler. Core knowledge. *Developmental science*, 10(1):89–96, 2007.

[278] Russell Stewart and Stefano Ermon. Label-free supervision of neural networks with physics and domain knowledge. In *AAAI*, 2017.

[279] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *ICCV*, 2021.

[280] Sanjay Subramanian, William Merrill, Trevor Darrell, Matt Gardner, Sameer Singh, and Anna Rohrbach. Reclip: A strong zero-shot baseline for referring expression comprehension. In *ACL*, 2022.

[281] Erik B Sudderth, Antonio Torralba, William T Freeman, and Alan S Willsky. Learning hierarchical models of scenes, objects, and parts. In *ICCV*, 2005.

[282] Erik B Sudderth, Antonio Torralba, William T Freeman, and Alan S Willsky. Describing visual scenes using transformed objects and parts. *IJCV*, 77(1-3):291–330, 2008.

[283] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019.

[284] Masato Tamura, Hiroki Ohashi, and Tomoaki Yoshinaga. Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information. In *CVPR*, 2021.

[285] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *arXiv preprint arXiv:2306.03881*, 2023.

[286] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017.

[287] Komal Teru, Etienne Denis, and Will Hamilton. Inductive relation prediction by subgraph reasoning. In *ICML*, 2020.

[288] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object frames by dense equivariant image labelling. In *NeurIPS*, 2017.

[289] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *ICCV*, 2017.

[290] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[291] Geoffrey G Towell and Jude W Shavlik. Knowledge-based artificial neural networks. *Artificial intelligence*, 70(1-2):119–165, 1994.

[292] Danyang Tu, Xiongkuo Min, Huiyu Duan, Guodong Guo, Guangtao Zhai, and Wei Shen. Iwin: Human-object interaction detection via transformer with irregular windows. In *ECCV*, 2022.

[293] Danyang Tu, Wei Sun, Guangtao Zhai, and Wei Shen. Agglomerative transformer for human-object interaction detection. In *ICCV*, 2023.

[294] Zhuowen Tu, Xiangrong Chen, Alan L Yuille, and Song-Chun Zhu. Image parsing: Unifying segmentation, detection, and recognition. *IJCV*, 63(2):113–140, 2005.

[295] Hsiao-Yu Fish Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *NeurIPS*, 2017.

[296] Oytun Ulutan, ASM Iftekhar, and Bangalore S Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. In *CVPR*, 2020.

[297] Giorgio Valentini. True path rule hierarchical ensembles for genome-wide gene function prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(3):832–847, 2010.

[298] Leslie G Valiant. Three problems in computer science. *Journal of the ACM*, 50(1):96–99, 2003.

[299] Lazar Valkov, Dipak Chaudhari, Akash Srivastava, Charles Sutton, and Swarat Chaudhuri. Houdini: Lifelong learning as program synthesis. In *NeurIPS*, 2018.

[300] Emile van Krieken, Erman Acar, and Frank van Harmelen. Analyzing differentiable fuzzy logic operators. *Artificial Intelligence*, 302:103602, 2022.

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

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

[303] Ramakrishna Vedantam, Karan Desai, Stefan Lee, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Probabilistic neural symbolic models for interpretable visual question answering. In *ICLR*, 2019.

[304] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. Order-embeddings of images and language. In *ICLR*, 2016.

[305] Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *CVPR*, 2019.

[306] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. In *CVPR*, 2016.

[307] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *ECCV*, 2018.

[308] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware

multi-level feature network for human object interaction detection. In *ICCV*, 2019.

[309] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *CVPR*, 2023.

[310] Hai Wang, Wei-shi Zheng, and Ling Yingbiao. Contextual heterogeneous graph network for human-object interaction detection. In *ECCV*, 2020.

[311] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *CVPR*, 2019.

[312] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE TPAMI*, 43(10):3349–3364, 2020.

[313] Ning Wang, Yibing Song, Chao Ma, Wengang Zhou, Wei Liu, and Houqiang Li. Unsupervised deep tracking. In *CVPR*, 2019.

[314] Ning Wang, Wengang Zhou, and Houqiang Li. Contrastive transformation for self-supervised correspondence learning. In *AAAI*, 2021.

[315] Suchen Wang, Yueqi Duan, Henghui Ding, Yap-Peng Tan, Kim-Hui Yap, and Junsong Yuan. Learning transferable human-object interaction detector with natural language supervision. In *CVPR*, 2022.

[316] Suchen Wang, Kim-Hui Yap, Henghui Ding, Jiyan Wu, Junsong Yuan, and Yap-Peng Tan. Discovering human interactions with large-vocabulary objects via query and multi-scale detection. In *ICCV*, 2021.

[317] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using in-

teraction points. In *CVPR*, 2020.

[318] Wenguan Wang, Cheng Han, Tianfei Zhou, and Dongfang Liu. Visual recognition with deep nearest centroids. In *ICLR*, 2023.

[319] Wenguan Wang, Jianbing Shen, Xiankai Lu, Steven CH Hoi, and Haibin Ling. Paying attention to video object pattern understanding. *IEEE TPAMI*, 43(7):2413–2428, 2020.

[320] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *CVPR*, 2015.

[321] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Selective video object cutout. *IEEE TIP*, 26(12):5645–5655, 2017.

[322] Wenguan Wang, Jianbing Shen, Fatih Porikli, and Ruigang Yang. Semi-supervised video object segmentation with super-trajectories. *IEEE TPAMI*, 41(4):985–998, 2018.

[323] Wenguan Wang, Jianbing Shen, Jianwen Xie, and Fatih Porikli. Super-trajectory for video segmentation. In *ICCV*, 2017.

[324] Wenguan Wang and Yi Yang. Towards data-and knowledge-driven artificial intelligence: A survey on neuro-symbolic computing. *IEEE TPAMI*, 47(2):878–899, 2024.

[325] Wenguan Wang, Yi Yang, and Yunhe Pan. Visual knowledge in the big model era: Retrospect and prospect. *Frontiers of Information Technology & Electronic Engineering*, 26(1):1–19, 2025.

[326] Wenguan Wang, Zhijie Zhang, Siyuan Qi, Jianbing Shen, Yanwei Pang, and Ling Shao. Learning compositional neural information fusion for human parsing. In *ICCV*, 2019.

[327] Wenguan Wang, Tianfei Zhou, Fatih Porikli, David Crandall, and Luc Van Gool. A survey on deep learning technique for video segmentation. *arXiv*

preprint arXiv:2107.01153, 2021.

[328] Wenguan Wang, Tianfei Zhou, Siyuan Qi, Jianbing Shen, and Song-Chun Zhu. Hierarchical human semantic parsing with comprehensive part-relation modeling. *IEEE TPAMI*, 44(7):3508–3522, 2021.

[329] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *ICCV*, 2021.

[330] Wenguan Wang, Hailong Zhu, Jifeng Dai, Yanwei Pang, Jianbing Shen, and Ling Shao. Hierarchical human parsing with typed part-relation reasoning. In *CVPR*, 2020.

[331] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019.

[332] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. In *NeurIPS*, 2020.

[333] Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. Hierarchical multi-label classification networks. In *ICML*, 2018.

[334] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.

[335] Jianan Wei, Tianfei Zhou, Yi Yang, and Wenguan Wang. Nonverbal interaction detection. In *ECCV*, 2024.

[336] Spencer Whitehead, Hui Wu, Heng Ji, Rogerio Feris, and Kate Saenko. Separating skills and concepts for novel visual question answering. In *CVPR*, 2021.

[337] Haiping Wu and Xiaolong Wang. Contrastive learning of image representations with cross-video cycle-consistency. In *ICCV*, 2021.

[338] Mingrui Wu, Jiaxin Gu, Yunhang Shen, Mingbao Lin, Chao Chen, and Xiaoshuai Sun. End-to-end zero-shot hoi detection via vision and language knowledge distillation. In *AAAI*, 2023.

[339] Ruizheng Wu, Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Memory selection network for video propagation. In *ECCV*, 2020.

[340] Xiaoqian Wu, Yong-Lu Li, Xinpeng Liu, Junyi Zhang, Yuzhe Wu, and Cewu Lu. Mining cross-person cues for body-part interactiveness learning in hoi detection. In *ECCV*, 2022.

[341] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[342] Fangting Xia, Peng Wang, Xianjie Chen, and Alan L Yuille. Joint multi-person pose estimation and semantic part segmentation. In *CVPR*, 2017.

[343] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018.

[344] Chi Xie, Fangao Zeng, Yue Hu, Shuang Liang, and Yichen Wei. Category query learning for human-object interaction classification. In *CVPR*, 2023.

[345] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021.

[346] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Shengping Zhang, and Wenxiu Sun. Efficient regional memory network for video object segmentation. In *CVPR*, 2021.

[347] Bingjie Xu, Yongkang Wong, Junnan Li, Qi Zhao, and Mohan S Kankanhalli. Learning to detect human-object interactions with knowledge. In *CVPR*, 2019.

[348] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image

diffusion models. In *CVPR*, 2023.

[349] Jiarui Xu and Xiaolong Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *ICCV*, 2021.

[350] Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Broeck. A semantic loss function for deep learning with symbolic knowledge. In *ICML*, 2018.

[351] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *ECCV*, 2018.

[352] Yichao Yan, Ning Zhuang, Jian Zhang, Minghao Xu, Qiang Zhang, Zhang Zheng, Shuo Cheng, Qi Tian, Xiaokang Yang, Wenjun Zhang, et al. Fine-grained video captioning via graph-based multi-granularity interaction learning. *IEEE TPAMI*, 44(2):666–683, 2019.

[353] Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *ICCV*, 2021.

[354] Fangkai Yang, Daoming Lyu, Bo Liu, and Steven Gustafson. Peorl: Integrating symbolic planning and hierarchical reinforcement learning for robust decision-making. In *IJCAI*, 2018.

[355] Lu Yang, Liulei Li, Xueshi Xin, Yifan Sun, Qing Song, and Wenguan Wang. Large-scale person detection and localization using overhead fisheye cameras. In *ICCV*, 2023.

[356] Xingyi Yang and Xinchao Wang. Diffusion model as representation learner. In *ICCV*, 2023.

[357] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE TPAMI*, 35(12):2878–2890, 2012.

[358] Yi Yang, Yueting Zhuang, and Yunhe Pan. Multiple knowledge representation

for big data artificial intelligence: framework, applications, and case studies. *Frontiers of Information Technology & Electronic Engineering*, pages 1551–1558, 2021.

[359] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *IEEE TPAMI*, 44(9):4701–4712, 2021.

[360] Bangpeng Yao and Li Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010.

[361] Jian Yao, Sanja Fidler, and Raquel Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*, 2012.

[362] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *NeurIPS*, 2018.

[363] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*, 2018.

[364] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. In *ICLR*, 2022.

[365] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *ECCV*, 2020.

[366] Aixi Zhang, Yue Liao, Si Liu, Miao Lu, Yongliang Wang, Chen Gao, and Xiaobo Li. Mining the benefits of two-stage and one-stage hoi detection. In *NeurIPS*, 2021.

[367] Fan Zhang, Yanqin Chen, Zhihang Li, Zhibin Hong, Jingtuo Liu, Feifei Ma, Junyu Han, and Errui Ding. Acfnet: Attentional class feature network for

semantic segmentation. In *ICCV*, 2019.

[368] Frederic Z Zhang, Dylan Campbell, and Stephen Gould. Spatially conditioned graphs for detecting human-object interactions. In *ICCV*, 2021.

[369] Frederic Z Zhang, Dylan Campbell, and Stephen Gould. Efficient two-stage detection of human-object interactions with a novel unary-pairwise transformer. In *CVPR*, 2022.

[370] Frederic Z Zhang, Yuhui Yuan, Dylan Campbell, Zhuoyao Zhong, and Stephen Gould. Exploring predicate visual context in detecting of human-object interactions. In *ICCV*, 2023.

[371] Jing Zhang, Bo Chen, Lingxi Zhang, Xirui Ke, and Haipeng Ding. Neural, symbolic and neural-symbolic reasoning on knowledge graphs. *AI Open*, 2:14–35, 2021.

[372] 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. In *NeurIPS*, 2023.

[373] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. In *ICLR*, 2022.

[374] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. In *NeurIPS*, 2021.

[375] Xiaomei Zhang, Yingying Chen, Bingke Zhu, Jinqiao Wang, and Ming Tang. Part-aware context network for human parsing. In *CVPR*.

[376] Xiaomei Zhang, Yingying Chen, Bingke Zhu, Jinqiao Wang, and Ming Tang. Blended grammar network for human parsing. In *ECCV*, 2020.

[377] Yurong Zhang, Liulei Li, Wenguan Wang, Rong Xie, Li Song, and Wenjun Zhang. Boosting video object segmentation via space-time correspondence

learning. In *CVPR*, 2023.

[378] Yong Zhang, Yingwei Pan, Ting Yao, Rui Huang, Tao Mei, and Chang-Wen Chen. Exploring structure-aware transformer over interaction proposals for human-object interaction detection. In *CVPR*, 2022.

[379] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[380] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *ECCV*, 2018.

[381] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *ICCV*, 2023.

[382] Yifan Zhao, Jia Li, Yu Zhang, and Yonghong Tian. Multi-class part parsing with joint boundary-semantic awareness. In *ICCV*, 2019.

[383] Zixu Zhao, Yueming Jin, and Pheng-Ann Heng. Modelling neighbor relation in joint space-time graph for video correspondence learning. In *ICCV*, 2021.

[384] Jilai Zheng, Chao Ma, Houwen Peng, and Xiaokang Yang. Learning to track objects from unlabeled videos. In *ICCV*, 2021.

[385] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, 2021.

[386] Sipeng Zheng, Boshen Xu, and Qin Jin. Open-category human-object interaction pre-training via language modeling framework. In *CVPR*, 2023.

[387] Xiushan Nie Dongfang Liu Yilong Yin Wenguan Wang Zheyun Qin, Xiankai Lu. Coarse-to-fine video instance segmentation with factorized condi-

tional appearance flows. *IEEE/CAA Journal of Automatica Sinica*, 10(5):1–17, 2023.

[388] Xubin Zhong, Changxing Ding, Zijian Li, and Shaoli Huang. Towards hard-positive query mining for detr-based human-object interaction detection. In *ECCV*, 2022.

[389] Bo Zhou, Liulei Li, Yujia Wang, Huafeng Liu, Yazhou Yao, and Wenguan Wang. Unialign: Scaling multimodal alignment within one unified model. In *CVPR*, 2025.

[390] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017.

[391] Desen Zhou, Zhichao Liu, Jian Wang, Leshan Wang, Tao Hu, Errui Ding, and Jingdong Wang. Human-object interaction detection via disentangled transformer. In *CVPR*, 2022.

[392] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *ICCV*, 2019.

[393] Qixian Zhou, Xiaodan Liang, Ke Gong, and Liang Lin. Adaptive temporal encoding network for video instance-level human parsing. In *ACM MM*, 2018.

[394] Tianfei Zhou, Liulei Li, Gustav Bredell, Jianwu Li, and Ender Konukoglu. Quality-aware memory network for interactive volumetric image segmentation. In *MICCAI*, 2021.

[395] Tianfei Zhou, Liulei Li, Gustav Bredell, Jianwu Li, Jan Unkelbach, and Ender Konukoglu. Volumetric memory network for interactive medical image segmentation. *Medical Image Analysis*, 83:102599, 2023.

[396] Tianfei Zhou, Liulei Li, Xueyi Li, Chun-Mei Feng, Jianwu Li, and Ling Shao. Group-wise learning for weakly supervised semantic segmentation. *IEEE TIP*, 31:799–811, 2021.

[397] Tianfei Zhou, Siyuan Qi, Wenguan Wang, Jianbing Shen, and Song-Chun Zhu. Cascaded parsing of human-object interaction recognition. *IEEE TPAMI*, 44(6):2827–2840, 2021.

[398] Tianfei Zhou, Wenguan Wang, Ender Konukoglu, and Luc Van Gool. Rethinking semantic segmentation: A prototype view. In *CVPR*, 2022.

[399] Tianfei Zhou, Wenguan Wang, Siyuan Qi, Haibin Ling, and Jianbing Shen. Cascaded human-object interaction recognition. In *CVPR*, 2020.

[400] Bingke Zhu, Yingying Chen, Ming Tang, and Jinqiao Wang. Progressive cognitive human parsing. In *AAAI*, 2018.

[401] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.

[402] Cheng Zou, Bohan Wang, Yue Hu, Junqi Liu, Qian Wu, Yu Zhao, Boxun Li, Chenguang Zhang, Chi Zhang, Yichen Wei, et al. End-to-end human object interaction detection with hoi transformer. In *CVPR*, 2021.

[403] Yuliang Zou, Zelun Luo, and Jia-Bin Huang. Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *ECCV*, 2018.