Towards Comprehensive Visual Understanding via Deep Neural Networks

by

Mu Chen

Thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy
under the supervision of **Prof. Yi Yang**

at the
Australian Artificial Intelligence Institute
Faculty of Engineering and Information Technology
University of Technology Sydney

April 2025

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, Mu Chen, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy (PhD) in the Faculty of Engineering and Information Technology 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 is supported by the Australian Government Research Training Program.

Production Note:
Signature removed prior to publication.

Signed: _____

Date: 9/7/2024

Abstract

Towards Comprehensive Visual Understanding via Deep Neural Networks

by

Mu Chen

Deep neural networks (DNNs) have made significant advancements in visual scene understanding, demonstrating great potential for applications in downstream tasks such as autonomous driving, robotic navigation, and human-computer interaction. Despite these successes, generalization ability remains a major obstacle on the path to comprehensive visual understanding, particularly when dealing with **i)** diverse scenes, as well as **ii)** diverse semantic structures within those scenes. Existing work typically requires extensive annotation for different scenes (domains) and separates the understanding of semantic targets into distinct tasks, designing meticulous networks and corresponding optimization for each. This poses challenges from two perspectives: **i)** generalizing from one domain to another, and **ii)** generalizing from one task to another. To adapt an existing model to various domains (challenge **i)**), this thesis proposes a self-supervised learning framework to learn generalizable structural representations, and a multi-task learning framework to extract transferable knowledge from multi-modalities. To enhance a model's ability to process various semantic structures (challenge **ii)**), this thesis introduces a holistic disentanglement and modeling for segment targets under an identical framework. Extensive experiments are conducted to verify the effectiveness of the proposed methods on scene understanding tasks, including Unsupervised Domain Adaptation (UDA), Exemplar-guided Video Segmentation (EVS), Video Instance Segmentation (VIS), Video Semantic Segmentation (VSS), Video Panoptic Segmentation (VPS), and Human-Object Interaction Detection (HOI Detection).

Dissertation directed by Professor Yi Yang

School of Computer Science

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Yi Yang, for his encouragement, guidance, and patience throughout my Ph.D. journey. Pursuing a Ph.D. under his supervision has been the most significant and fortunate decision I have ever made, one that has undoubtedly transformed my academic and personal trajectory. I am incredibly grateful for his tremendous support on any research topics that I was excited about. I also deeply appreciate his kind assistance in both my career and personal life.

I would also like to give my sincere gratitude to my co-supervisor, Professor Wenguan Wang, for his valuable guidance and inspirational advice on my research. His research passion and academic taste have profoundly influenced me. I have gradually grown into a professional researcher through every insightful conversation with him.

I extend my sincere thanks to my mentor, Dr. Zhedong Zheng, for his thoughtful guidance and substantial support in the early years of my research journey. He introduced me to the world of computer vision research and guided me in completing my first submission.

Thanks to all my colleagues and cherished friends in the ReLER lab. I was really fortunate to work with them and participate in illuminating discussions with these smart people. Special thanks to Mr. Liulei Li, for his generous assistance. I benefit greatly from his enlightening discussions and constructive suggestions.

Lastly, I would like to thank my parents Mr. Xiangyang Chen and Ms. Xiaoqin Mu, my grandparents Mr. Xuezhi Mu and Ms. Linglan Zeng, for their unconditional love throughout the years.

Mu Chen

Sydney, Australia, 2024

List of Publications

- **M. Chen**, Z. Zheng, Y. Yang, and T. Chua. "Pipa: Pixel-and patch-wise self-supervised learning for domain adaptative semantic segmentation," In *Proceedings of the 31st ACM International Conference on Multimedia.* (ACM MM 2023 Poster)
- **M. Chen**, Z. Zheng, Y. Yang. "Transferring to Real-World Layouts: A Depth-aware Framework for Scene Adaptation," In *Proceedings of the 32st ACM International Conference on Multimedia.* (ACM MM 2024 Oral Presentation)
- **M. Chen**, L. Liu, R. Quan, W. Wang, Y. Yang. "GvSeg: General and Task-oriented Video Segmentation," In *Proceedings of the 34th European Conference on Computer Vision.* (ECCV 2024)
- **M. Chen**, M. Chen, Y. Yang. "UAHOI: Uncertainty-aware Robust Interaction Learning for HOI Detection," In *Computer Vision and Image Understanding.* (CVIU)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Visual understanding is the perception of objects, actions, and the semantic relationships between them [13–18]. It emphasizes recognition, learning, and reasoning and can benefit a wide range of downstream tasks, such as autonomous driving, robotics, video surveillance, and augmented reality. Deep neural networks (DNNs) have revolutionized visual understanding, achieving groundbreaking advancements in both academic research and practical applications. Despite these successes, the generalization capability of DNNs remains a significant challenge. In practical applications, DNNs often exhibit poor generalization when confronted with varying data distributions and complex environments. Current research typically ❶ annotates large amounts of data for specific scenarios and ❷ designs specific network architectures for the semantic structures of particular interest within those scenarios to achieve high-precision scene parsing. For ❶, it is common practice to annotate datasets for scenarios such as nighttime [19], adverse weather conditions [20], or different cities [21]. However, DNNs are data-hungry, and meeting the annotation requirements is challenging. For instance, annotating a single static scene for pixel-level tasks in the Cityscapes dataset [22] takes 1.5 hours, while annotating images under adverse weather conditions can take up to 3.3 hours [20]. The situation is even more difficult for dynamic scene [1], making it impractical in real-world applications. For ❷, existing work designs visual understanding tasks such as semantic segmentation, instance segmentation, and panoptic segmentation based on different semantic targets. This allows for a multi-level understanding of the same scene, but these tasks employ distinct architectures, loss functions, and training procedures, leading to duplicate research and repeated optimization efforts.

To address these limitations, this thesis conducts an in-depth study from two critical perspectives, seeking for comprehensive scene understanding: ❶ enhancing the generalization of the learned model to other domains, and ❷ faclitating research endeavors devoting on one task to another.

For ❶ generalization from one domain to another: this thesis investigates the under-explored intra-domain knowledge, and extracts the domain-invariant representation under a multi-grained self-supervised learning framework. Additionally, it further minines the domain-invariant feature learing with the help of information from other modalities, ultimately achieving a multi-task learning framework.

For ❷ generalization from one task to another: this thesis recognizes the key factors that constitute segment targets and leavrage these key factors to build an identical model, enabling seamless accommodation of task-specific properties into a generalist framework.

The contributions of the thesis are listed as follows:

- This thesis explores the learning of domain-invariant representation through a pixel- and patch-level self-supervised learning framework. Taking a step further, it leverages rich depth information and proposes a depth-aware multi-task learning framework to improve the model's generalization across different domains. Extensive experiments on unsupervised domain adaptation benchmarks verify the effectiveness of the proposed methods.
- This thesis conducts an in-depth analysis of inherent properties of moving targets within scenes and proposes task-oriented property accommodation to handle various video tasks under a unified architecture. Experiments across multiple video segmentation benchmarks demonstrate its generalization ability and superior performance.
- This thesis dynamically adjusts confidence thresholds via uncertainty prediction to handle complex interactions, achieving a more robust visual scene understanding for more high-level semantic understanding tasks.

The following introduces the background, motivation and developed methodologies for comprehensive visual understanding.

1.1 Generalize from One Domain to Another

Generalizing from one domain to another is one of the major challenges in achieving comprehensive visual understanding using deep neural networks (DNNs). DNNs are proficient in feature extraction and representation learning, capturing rich feature representations through multiple layers of non-linear transformations. This capability gives DNNs a substantial advantage in handling complex visual tasks, fundamentally revolutionizing traditional methods of visual understanding. However, these networks are notoriously data-hungry, typically requiring extensive training datasets with pixel-level annotations, which are challenging to obtain in real-world scenarios. This limitation is particularly pronounced in segmentation tasks that necessitate pixel-level annotations. To address the scarcity of training data, one effective strategy is to utilize synthetic data with annotations generated via computer graphics. Nonetheless, significant domain discrepancies exist between synthetic and real-world images, particularly concerning illumination, weather conditions, and camera parameters. To overcome these discrepancies, researchers employ unsupervised domain adaptation (UDA) techniques, which transfer knowledge from labeled source domains to unlabeled target domains.

Chapter 3 delves into unsupervised domain adaptation (UDA) for semantic segmentation and proposes a novel self-supervised learning framework. Semantic segmentation serves as a foundational task for visual understanding, which enables detailed scene interpretation by classifying each pixel in an image into a predefined category. This granular level of understanding is crucial for accurately perceiving the structure and composition of visual scenes, facilitating higher-level reasoning and decision-making processes. The analysis of existing self-supervised learning UDA methods reveals two key issues: **i)** the high-level representations they produce lack sufficient contextual information, which is vital for scene understanding; and **ii)** self-sueprvised learning at the patch level can prevent the model from ignoring context entirely. Based on these insights, this chapter investigates prediction consistency across different regions. Specifically, incorporating patch-level contrastive learning leads to larger receptive field, making the approach more suitable for segmentation tasks that demand robust contextual information. Consequently, this chapter introduces a multi-grained Pixel- and Patch-wise self-supervised learning framework. Experiments are conducted using synthetic datasets SYNTHIA [23] and GTA-5 [24] as source domains and Cityscapes [22] as target domain. The proposed framework demonstrates superior performance compared to existing UDA methods.

Chapter 4 delves deeper into Unsupervised Domain Adaptation (UDA) through a multi-task learning approach. The key insight is derived from the observation that existing work typically employs the well-known class-mix technique to address the domain shift problem. In particular, cross-domain mixing involves copying regions corresponding to certain categories from a source domain image and pasting them onto an unlabeled target domain image. However, this straightforward strategy often results in placing a large number of objects at unrealistic depth positions. This issue arises because each category has its own positional distribution. Such artificially crafted training data compromise contextual learning, leading to sub-optimal performance, especially for small objects. Semantic categories can be effectively separated using depth maps. By introducing depth information, it is possible to ensure that the cross-domain mix conforms to the realistic distribution of categories, thereby improving performance. Additionally, multi-modal data can enhance the learning of deep representations. Thus, a multi-task learning framework is proposed that encourages the network to optimize fused multi-task features in an end-to-end manner. This method achieves competitive accuracy on two commonly used scene adaptation benchmarks, with particularly notable improvements in minor categories.

1.2 Generalize from One Task to Another

Generalizing from one task to another is also a major challenge in achieving comprehensive scene understanding. Depending on the objects of interest within the scene, existing scene understanding research often defines different task settings, breaking down the complex issue of scene understanding into smaller sub-tasks. While optimizing models for specific tasks can improve their performance on those tasks, such optimizations often fail to generalize across different tasks, leading to redundant research efforts.

Chapter 5 analyzes key factors that constitute segmentation targets and explores how to leverage these factors to make a generalizable model more sensitive to specific tasks, thus remaining competitive in each task. A generalist video segmentation framework is proposed that achieves holistic disentanglement and modeling of segmentation targets. This approach allows network optimization techniques to be effective across different tasks, substantially elevating both accuracy and robustness in several video segmentation benchmarks.

The above three chapters address the fundamental aspects (e.g., individual entities) of visual understanding.

Chapter 6 further investigates the interaction between these entities to achieve a higher-level and more holistic understanding of the scene. This chapter emphasizes Human-Object Interaction Detection (HOI Detection), which extends from the detection of objects to include their relationships, prompting a deeper understanding of high-level semantic comprehension. UAHOI (Uncertainty-aware Robust Human-Object Interaction Learning) is introduced as a novel method that leverages uncertainty estimation to dynamically adjust interaction prediction thresholds in HOI detection tasks. This approach incorporates uncertainty modeling to refine decision-making processes, enabling the model to set confidence thresholds based on predicted uncertainty for each interaction. Specifically, the variance in predictions is used as an uncertainty measure for both human/object bounding boxes and interactions, reflecting the model's confidence in its outputs. This variance is integrated into the optimization target, improving bounding box accuracy and ensuring significant interactions are not missed due to artificially low confidence thresholds. Comprehensive evaluations on two standard human-object interaction datasets, HICO-DET and V-COCO, demonstrate that this method significantly outperforms existing state-of-the-art approaches.

1.3 Thesis Outline

The thesis is organized as follows:

Chapter 2 presents a comprehensive literature review of related topics.

Chapter 3 and Chapter 4 sequentially explore the generalization problem from one domain to another for more comprehensive visual understanding. Chapter 3 focuses on domain adaptation and proposes a multi-grained self-supervised framework. Chapter 4 further investigates semantic segmentation domain adaptation and introduces a depth-aware multi-task leraning framework.

Chapter 5 shifts the investigation towards the generalization problem from one task to another and proposes a generalist video segmentation framework.

Chapter 6 investigates more high-level semantic understanding task HOI Detection and proposes an uncertainty-aware framework.

Finally, 7 summerizes the contributions and discusses potential future directions.

Chapter 2

Literature Review

This chapter provides a thorough literature review of related work in comprehensive visual understanding, encompassing various methodologies and advancements in image-level and video-level tasks, as well as domain adaptive scene understanding, highlighting the significant progress and ongoing challenges in the field.

2.1 Image-level Visual Understanding

Image-level visual understanding typically encompasses various tasks, ranging from fundamental segmentation which segments meaningful parts of a scene to more high-level and complex tasks such as human-object interaction detection which recognizes and localizes the interaction between human and objects inside a scene.

2.1.1 Image segmentation

Traditional image segmentation. Traditional image segmentation tasks are typically categorized based on the segmentation targets within a scene. These categories include semantic segmentation, instance segmentation, and panoptic segmentation. Semantic segmentation [25–28] assigns a class label to each pixel in the image, grouping pixels with the same label into regions. Instance segmentation [29–32] identifies and delineates each distinct object instance within the image, assigning unique labels to different instances of the same class. Panoptic segmentation [33–37] combines semantic and instance segmentation, providing both class labels for each pixel and instance labels for each object.

Query-based image segmentation. Image segmentation has witnessed substantial progress with top-performing approaches [32, 38–41] primarily falling into the query-based paradigm. Such paradigm directly models targets by introducing a set of learnable embeddings as queries to search for objects of interest and subsequently decode masks from image features. Inspired by DETR [42], the latest research [40, 41, 43] takes this paradigm a step further by harnessing the Transformer architecture. With a series of studies on query-based segmentation, recent works [39, 41, 44, 45] are moving towards universal image segmentation, which aims to develop a unified architecture to address various segmentation tasks.

2.1.2 Human-Object Interaction (HOI) Detection

Traditional HOI Detection. Human-Object Interaction (HOI) detection provides numerous high-level intricate relationships between humans and objects, gradually serving as the foundation of many computer vision applications [16, 46]. Traditional Human-Object Interaction (HOI) detection methods can typically be divided into two categories: two-stage methods [47–59] and one-stage methods [60–63]. Two-stage HOI detection methods rely on an off-the-shelf object detector to extract bounding boxes and class labels for humans and objects in the first stage. In the second stage, they model interactions for each human-object pair via a multi-stream network. For example, [47] propose leveraging the Graph Parsing Neural Network (GPNN) to incorporate structural knowledge into HOI detection. Similarly, [48] introduces a streamlined factorized model that utilizes insights from pre-trained object detectors. Subsequent research often involves integrating additional contextual and relational information to enhance performance further. However, two-stage methods find it challenging to identify human-object pairs among a large number of permutations and heavily rely on the detection results, suffering from low efficiency and effectiveness.[64–68]. By introducing anchor points to associate humans and objects, single-stage methods detect pairs likely to interact and their interactions simultaneously. This approach, which handles instance detection and interaction point prediction branches in parallel, has made impressive progress in HOI detection. For instance, [60] utilizes a union-level detector to directly capture the region of interaction, enhancing focus on interaction-specific areas. Meanwhile, [61] employs point detection branches that concurrently predict points for both the human/object and their interactions. This method not only implicitly provides context

but also offers regularization for the detection of humans and objects, improving overall accuracy and context relevance.

End-to-End HOI Detection. Inspired by DETR [42], recent work [69–80] modify HOI as a set-prediction problem by generating a set of HOI triplets. Early approaches [71, 72] simply migrated the Transformer decoder to HOI tasks, using a single decoder to couple human-object detection and interaction classification, achieving end-to-end training. [71] replaces manually defined location-of-interest with a transformer-based feature extractor, enhancing feature representation capabilities. [72] addresses HOI detection using an end-to-end approach, which eliminates the reliance on hand-designed components, streamlining the detection process. However, due to the significant differences between the two tasks, learning a unified instance-interaction representation proved challenging. Therefore, subsequent works [69, 70, 73] gradually shifted towards using separate decoders for instance prediction and interaction prediction, allowing the model to fully focus on the differences between instance and interaction prediction areas. To further enhance performance, other methods have introduced language [81, 82] and logic-reasoning [83] to explore the relationships between humans, objects, and their interactions more deeply. Moreover, scene graph generation (SGG) is another high-level semantic understanding task closely related to HOI detection, which is advanecd with the help of techniques such as noisy label correction [84], LLMs [85] and compositional augmentation [86]. Both HOI Detection and SGG aim to capture and understand complex relationships between objects within a scene, but scene graph generation focuses on identifying objects and their pairwise relationships to create a structured graph representation, while HOI detection specifically targets interactions between humans and objects.

2.2 Video-level Visual Understanding

2.2.1 Task-specific Video Segmentation

Traditional video segmentation methods, similar to image-level segmentation, typically design separate frameworks for each task and are divided into independent tasks based on the different segmentation targets in the video scene.

Exemplar-guided Video Segmentation (EVS). Exemplar-guided Video Segmentation (EVS). Given the hint which can be mask, bounding box, or point at one video frame, EVS

aims to propagate the mask-level predictions to subsequent frames [9, 87]. Therefore, the standard video object segmentation (VOS) task can be viewed as a specific instance of EVS – mask-guided video segmentation. Recent promising solutions for the mask-guided task mainly implemented in a *matching-based* manner which classifies pixels in current frame according to the feature similarities of target objects in reference frames [88–93, 93–106]. To solve the bounding box and point-guided tasks, current solutions typically have to regress a pseudo ground-truth mask via pre-processing [9, 87].

Video Instance Segmentation (VIS). Extending beyond detecting and segmenting instances within images, VIS further engages in the active tracking of individual objects across video frames. According to the process of video sequences, existing solutions for VIS fall into three categories [107]: *online*, *semi-online*, and *offline*. The *online* methods take each frame as inputs and associate instances through hand-designed rules [2, 108–110], integrating learnable matching algorithms [111–116], or deploying query matching frameworks [117–122]. The *semi-online* solutions typically divide long videos into clips and model the representations of instances by leveraging rich spatio-temporal information [123–126]. Conversely, *offline* methods predict the instance sequence for an entire video in a single step [116, 127–131] which require a growing amount of GPU memory as the video length extends, limiting their application in real-world scenarios.

Video Semantic Segmentation (VSS). Building upon the principle of semantic segmentation [16, 132–138], VSS extends this concept to video sequences, so as to capture the evolution of scenes and objects over time. Existing solutions can generally be classified into two main paradigms. The *motion-based* approaches [139–143] employ optical flow to model dynamic scenes. Though workable in certain scenarios, they rely heavily on the accuracy of flow maps and are prone to error accumulation [144]. On the other hand, the *attention-based* methods take advantage of the attention mechanism [145–147] or Transformer [148, 149] to aggregate temporal cues. This contributes to improved coherence among predictions of individual frames.

Video Panoptic Segmentation (VPS). With the emergence of seminal work [150], there has been a research trend [151–157] dedicated to unifying video instance and semantic segmentation. Though showing the promise of general video segmentation, the early work [151–153] utilizes task-specific heads to handle instance and semantic segmentation

separately, and assembles the panoptic predictions through post-processing. Recent algorithms typically leverage unified queries for the detection and tracking of both *thing* and *stuff* objects [154–157].

2.2.2 General Video Segmentation (GVS)

In order to address the limitations of task-specific models that lack the flexibility to generalize across different tasks and result in redundant research efforts, GVS aims at an all-inclusive solution for multiple video segmentation tasks. A limited number of studies [39, 87, 126, 158–161] have ventured in this direction. However, [39, 126, 158] exhibits inferior performance compared to dedicated, task-specific methods. [87] achieves remarkable results but requires extensive pre-training on various large-scale, pixel-level annotated datasets.

2.3 Domain Adaptive Scene Understanding

2.3.1 Unsupervised Domain Adaptation

Unsupervised domain adaptation (UDA) aims to train a model on a label-rich source domain and adapt the model to a label-scarce target domain. Some methods propose learning the domain-invariant knowledge by aligning the source and target distribution at different levels. For instance, AdaptSegNet [162], ADVENT [163], and CLAN [164] adversarially align the distributions in the feature space. CyCADA [165] diminishes the domain shift at both pixel-level and feature-level representation. DALN [166] proposes a discriminator-free adversarial learning network and leverages the predicted discriminative information for feature alignment. Both Wu *et al.*[167] and Yue *et al.* [168] learn domain-invariant features by transferring the input images into different styles, such as rainy and foggy, while Zhao *et al.* [169] and Zhang *et al.* [170] diversify the feature distribution via normalization and adding noise respectively. Another line of work refines pseudo-labels gradually under the iterative self-training framework, yielding competitive results. Following the motivation of generating highly reliable pseudo labels for further model optimization, CBST [171] adopts class-specific thresholds on top of self-training to improve the generated labels. Feng *et al.*[172] acquire pseudo labels with high precision by leveraging the group information. PyCDA [173] constructs pseudo-labels in various scales to further

improve the training. Zheng *et al.*[174] introduce memory regularization to generate consistent pseudo labels. Other works propose either confidence regularization [175, 176] or category-aware rectification [177, 178] to improve the quality of pseudo labels. DACS [179] proposes a domain-mixed self-training pipeline to mix cross-domain images during training, avoiding training instabilities. Kim *et al.*[180], Li *et al.*[181] and Wang *et al.*[182] combine adversarial and self-training for further improvement. Chen *et al.*[183] establish a deliberated domain bridging (DDB) that aligns and interacts with the source and target domain in the intermediate space. SePiCo [184] and PiPa [16] adopt contrastive learning to align the domains. Liu *et al.*[185] addresses the label shift problem by adopting class-level feature alignment for conditional distribution alignment. Researchers also attempted to perform entropy minimization [163, 186], and image translation [187, 188]. consistency regularization[189–192]. Recent multi-target domain adaptation (MTDA) methods enable a single model to adapt a labeled source domain to multiple unlabeled target domains [193–195].

2.3.2 Self-supervised Learning

Self-supervised learning is a method where models learn from unlabeled data by predicting parts of the input from other parts. It leverages data's intrinsic structure to generate labels. Contrastive learning,a subset of self-supervised learning, is one of the most prominent self-supervised representation learning methods [196–200], which contrasts similar (positive) data pairs against dissimilar (negative) pairs, thus learning discriminative feature representations. For instance, Wu *et al.* [197] learn feature representations at the instance level. He *et al.* [200] match encoded features to a dynamic dictionary which is updated with a momentum strategy. Chen *et al.*[199] proposes to engender negative samples from large mini-batches. In the domain adaptative image classification, contrastive learning is utilized to align feature space of different domains [201, 202].

A few recent studies utilize contrastive learning to improve the performance of semantic segmentation task [184, 203–207]. For example, Wang *et al.*[203] have designed and optimized a self-supervised learning framework for better visual pre-training. Gansbeke *et al.*[204] applies contrastive learning between features from different saliency masks in an unsupervised setting. Recently, Huang *et al.*[208] tackles UDA by considering instance

contrastive learning as a dictionary look-up operation, allowing learning of category-discriminative feature representations. Xie *et al.*[209] presents a semantic prototype-based contrastive learning method for fine-grained class alignment. Other works explore contrastive learning either in a fully supervised manner [184, 205] or in a semi-supervised manner [210–212]. For example, Wang *et al.*[205] uses pixel contrast in a fully supervised manner in semantic segmentation. But most methods above either target image-wise instance separation or tend to learn pixel correspondence alone.

2.3.3 Multi-task Learning

Semantic segmentation and geometric information are shown to be highly correlated [213–219]. Recently depth estimation has been increasingly used to improve the learning of semantics within the context of multi-task learning, but the depth information should be exploited more precisely to help the domain adaptation. SPIGAN [220] pioneered the use of geometric information as an additional supervision by regularizing the generator with an auxiliary depth regression task. DADA [221] introduces an adversarial training framework based on the fusion of semantic and depth predictions to facilitate the adaptation. GIO-Ada [222] leverages the geometric information on both the input level and output level to reduce domain shift. CTRL [223] encodes task dependencies between the semantic and depth predictions to capture the cross-task relationships. CorDA [224] bridges the domain gap by utilizing self-supervised depth estimation on both domains. Wu *et al.* [225] propose to further support semantic segmentation by depth distribution density.

Chapter 3

PiPa: Pixel- and Patch-wise Self-supervised Learning for Domain Adaptative Semantic Segmentation

3.1 Introduction

Unsupervised Domain Adaptation (UDA) aims to enhance the generalization of the learned model to other domains. The domain-invariant knowledge is transferred from the model trained on labeled source domain, *e.g.*, video game, to unlabeled target domains, *e.g.*, real-world scenarios, saving annotation expenses. Prevailing models, *e.g.*, Convolutional Neural Networks (CNNs) [226, 227] and Visual Transformers [228, 229], have achieved significant progress in computer vision applications [230–232]. But such networks are data-hungry, which usually require large-scale training datasets with pixel-level annotations. The annotation prerequisites are hard to meet in real-world scenarios. To address the shortage in the training data, one straightforward idea is to access the abundant synthetic data and the corresponding pixel-level annotations generated by computer graphics [233, 234]. However, there exist domain gaps between synthetic images and real-world images in terms of illumination, weather, and camera hyper-parameters [167, 235, 235, 236]. To minimize such a gap, researchers resort to unsupervised domain adaptation (UDA) to transfer the knowledge from labeled source-domain data to the unlabeled target-domain environment.

The key idea underpinning UDA is to learn the shared domain-invariant knowledge. One line of works, therefore, investigates techniques to mitigate the discrepancy of data distribution between source domain and target domain at different levels, such as pixel level [165, 167, 181, 237], feature level [238, 239], and prediction level [162, 163, 240, 241]. These inter-domain alignment approaches have achieved significant improvement compared to basic source-only methods, but usually overlook the intra-domain knowledge.

Another potential paradigm to address the lack of training data is self-supervised learning, which mines the visual knowledge from unlabeled data. One common optimization objective is to learn invariant representation against various augmentations, such as rotation [242], colorization [243], mixup [244] and random erasing [245]. Prior UDA works [174, 176] explored self-supervised methods to mine the domain-invariant knowledge, but the pipelines are relatively simple and only consider the prediction consistency against dropout or different network depths. Recent Segmentation and UDA work [184, 205] adopt contrastive learning methods, showing great performance. However, they focus only on pixel-level contrast without a context-aware design. We analyze existing contrastive learning methods and observe that (1) the high-level representation produced by them does not capture enough contextual information which is crucial in segmentation tasks. (2) performing contrastive learning at patch-level could prevent the model from degrading into totally ignoring the contexts. In light of the above observation, we explore the prediction consistency and contrastive learning at different effect regions. The consideration of patch-level has resulted in a larger receptive field, which makes it more suitable for segmentation tasks that require stronger contextual information. Therefore, we introduce a multi-grained Pixel- and Patch-wise self-supervised learning framework.

As the name implies, PiPa explores the pixel-to-pixel and patch-to-patch relation for regularizing the segmentation feature space. Our approach is based on two implicit priors: (1) the feature of the same-class pixels should be kept consistent with the category prototype; and (2) the feature within a patch should maintain robustness against different contexts. As shown in Figure 3.1, image pixels are mapped into an embedding space (Figure 3.1 (b) and (d)). For the **pixel-wise contrast**, we explicitly facilitate discriminative feature learning by pulling pixel embeddings of the same category closer while pushing those of different categories away (Figure 3.1 (b)).

Considering the **patch-wise contrast**, we randomly crop two image patches with an overlapping region (the yellow region in Figure 3.1 (c) and (d)) from an unlabeled image.

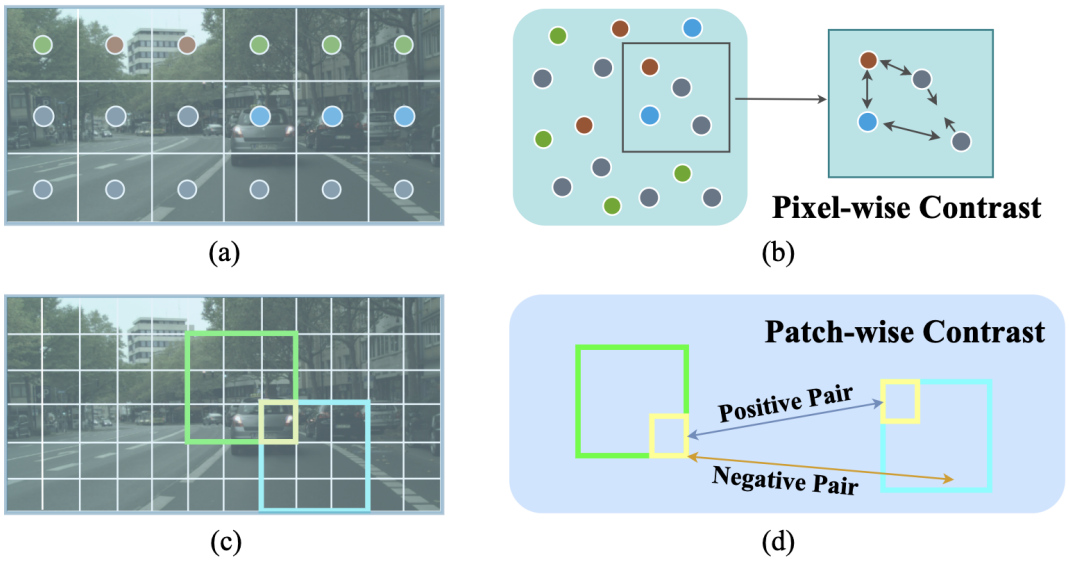

FIGURE 3.1: Different from existing works, we focus on mining the intra-domain knowledge, and argue that the contextual structure between pixels and patches can facilitate the model learning the domain-invariant knowledge in a self-supervised manner. In particular, our proposed training framework: (1) motivates intra-class compactness and inter-class dispersion by pulling closer the pixel-wise intra-class features and pushing away inter-class features within the image (see a,b at the top row); and (2) maintains the local patch consistency against different contexts, such as the yellow local patch in the green and the blue patch (see the bottom row c,d). Albeit simple, the proposed learning method is compatible with other existing methods to further boost performance.

The overlapping region of the two patches should not lose its spatial information and maintain the prediction consistency even against two different contexts. The proposed method is orthogonal to other existing domain-alignment works. We re-implement two competitive baselines, and show that our framework consistently improves the segmentation accuracy over other existing works. Our contributions are as follows:

(1) Different from existing works on inter-domain alignment, we focus on mining domain-invariant knowledge from the original domain in a self-supervised manner. We propose a unified Pixel- and Patch-wise self-supervised learning framework to harness both pixel- and patch-wise consistency against different contexts, which is well-aligned with the segmentation task.

(2) Our self-supervised learning method does not require extra annotations, and is compatible with other existing UDA frameworks. The effectiveness of PiPa has been tested

by extensive ablation studies, and it achieves competitive accuracy on two commonly used UDA benchmarks, namely 75.6 mIoU on GTA→Cityscapes and 68.2 mIoU on Synthia→Cityscapes.

3.2 Methodology

We first introduce the problem definition and conventional segmentation losses for semantic segmentation domain adaptation. Then we shed light on the proposed component of our framework PiPa, *i.e.*, Pixel-wise Contrast and Patch-wise Contrast, both of which work on local regions to mine the inherent contextual structures. We finally also raise a discussion on the mechanism of the proposed method.

3.2.1 Problem Statement

As shown in Figure 3.2, given the source-domain synthetic data $X^S = \left\{x_u^S\right\}_{u=1}^U$ labeled by $Y^S = \left\{y_u^S\right\}_{u=1}^U$ and the unlabelled target-domain real-world data $X^T = \left\{x_v^T\right\}_{v=1}^V$, where U and V are the numbers of images in the source and target domain, respectively. The label Y^S belongs to C categories. Domain adaptive semantic segmentation intends to learn a mapping function that projects the input data X^T to the segmentation prediction Y^T in the target domain.

Basic Segmentation Losses. Similar to existing works [174, 175], we learn the basic source-domain knowledge by adopting the segmentation loss on the source domain as:

$$\mathcal{L}_{ce}^S = \mathbb{E}\left[-p_u^S \log h_{cls}(g_\theta\left(x_u^S\right))\right], \tag{3.1}$$

where p_u^S is the one-hot vector of the label y_u^S, and the value $p_u^S(c)$ equals to 1 if $c == y_u^S$ otherwise 0. We harness the visual backbone g_θ, and 2-layer multilayer perceptrons (MLPs) h_{cls} for segmentation category prediction.

To mine the knowledge from the target domain, we generate pseudo labels $\bar{Y}^T = \{\bar{y_v}^T\}$ for the target domain data X^T by a teacher network $g_{\bar{\theta}}$ [179, 192], where $\bar{y_v}^T = argmax(h_{cls}g_{\bar{\theta}}(x_v^T))$. In practice, the teacher network $g_{\bar{\theta}}$ is set as the exponential moving average of the weights of the student network g_θ after each training iteration [246, 247]. Considering that there are no labels for the target-domain data, the network g_θ is trained on the pseudo label $\bar{y_v}^T$

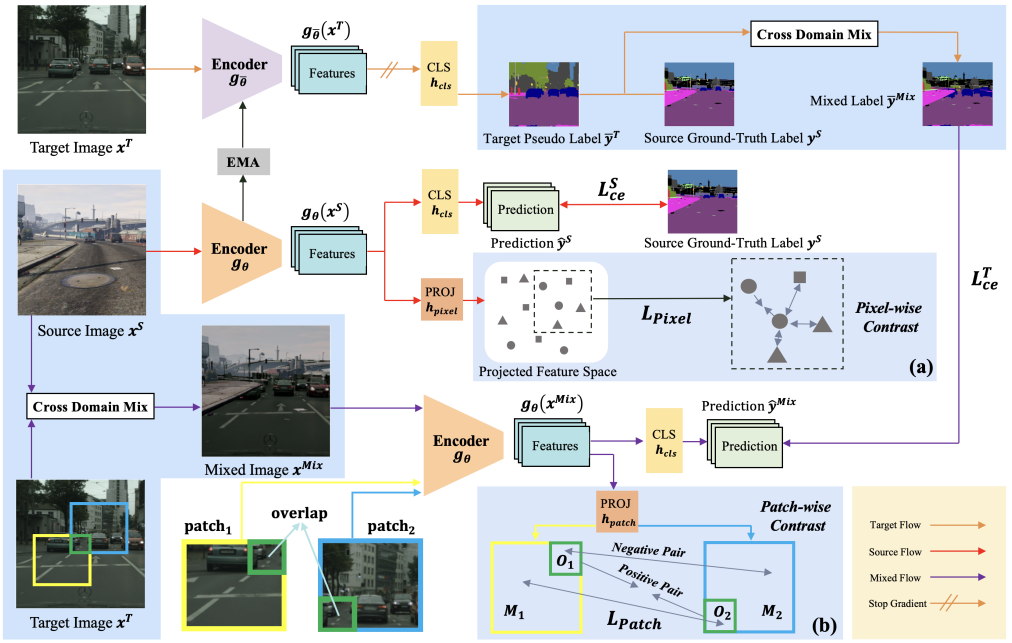

FIGURE 3.2: A brief illustration of our unified multi-grained self-supervised learning Framework (PiPa). Overall, PiPa is based on a teacher-student architecture. The teacher network is randomly initialized and the student network is pretrained on ImageNet1k. Then the teacher network is utilized to generate target pseudo labels $\bar{Y}^T$ and the weights are updated by the weights of the student network. Given the labeled source data $\{(x^S, y^S)\}$, we calculate the segmentation prediction $\hat{y}^S$ with the backbone g_θ and the classification head h_{cls}, supervised by the basic segmentation loss L_{ce}^S. During training, we leverage the moving averaged model $g_{\bar{\theta}}$ to estimate the pseudo label $\bar{y}^T$ to craft the mixed label $\bar{y}^{Mix}$ based on the category. According to the mixed label, we copy the corresponding regions as the mixed data x^{Mix}. We also deploy the model g_θ and the head h_{cls} to obtain the mixed prediction $\hat{y}^{Mix}$ supervised by L_{ce}^T. Except for the above-mentioned basic segmentation losses, we revisit current pixel contrast and propose a unified multi-grained Contrast. In **(a)**, we regularize the pixel embedding space by computing pixel-to-pixel contrast: impelling positive-pair embeddings closer, and pushing away the negative embeddings. In **(b)**, we regularize the patch-wise consistency between projected patch $\mathbf{O}_1$ and $\mathbf{O}_2$. Similarly, we harness the patch-wise contrast, which pulls positive pair, *i.e.*, two features at the same location of $\mathbf{O}_1$ and $\mathbf{O}_2$ closer, while pushing negative pairs apart, *i.e.*, any two features in $\mathbf{M}_1 \cup \mathbf{M}_2$ at different locations. During inference, we drop the two projection heads h_{patch} and h_{pixel} and only keep g_θ and h_{cls}.

generated by the teacher model $g_{\bar{\theta}}$. Therefore, the segmentation loss can be formulated as:

$$\mathcal{L}_{ce}^{T} = \mathbb{E}\left[-\bar{p_v}^{T} \log h_{cls}(g_\theta\left(x_v^T\right))\right], \tag{3.2}$$

where $\bar{p_v}^{T}$ is the one-hot vector of the pseudo label $\bar{y_v}^{T}$. We observe that pseudo labels inevitably introduce noise considering the data distribution discrepancy between two domains. Therefore, we set a threshold that only the pixels whose prediction confidence is higher than the threshold are accounted for the loss. In practice, we also follow [179, 248] to mix images from both domains to facilitate stable training. Specifically, the label $\bar{y}^{Mix}$ is generated by copying the random 50% categories in y^S and pasting such class areas to the target-domain pseudo label $\bar{y}^T$. Similarly, we also paste the corresponding pixel area in x^S to the target-domain input x^T as x^{Mix}. Therefore, the target-domain segmentation loss is updated as:

$$\mathcal{L}_{ce}^{T} = \mathbb{E}\left[-\bar{p_v}^{Mix} \log h_{cls}(g_\theta\left(x_v^{Mix}\right))\right], \tag{3.3}$$

where $\bar{p_v}^{Mix}$ is the probability vector of the mixed label $\bar{y_v}^{Mix}$. Since we deploy the copy-and-paste strategy instead of the conventional mixup [249], the mixed labels are still one-hot.

3.2.2 Multi-grained Contrast in different effect regions.

We note that the above-mentioned segmentation loss does not explicitly consider the inherent context within the image, which is crucial to the local-focused segmentation task. Therefore, we study the feasibility of self-supervised learning in mining intra-domain knowledge for domain adaptive semantic segmentation tasks. In this chapter, we revisit the current pixel-wise contrast in semantic segmentation [205] and explore the joint training mechanism of contrastive learning on both pixel- and patch-level effect regions. To this end, we introduce a unified multi-grained contrast including patch-wise contrast to enhance the consistency within a local patch.

In the **pixel-wise** effect region, given the labels of each pixel y^S, we regard image pixels of the same class C as positive samples and the rest pixels in x^S belonging to the other classes are the negative samples. The pixel-wise contrastive loss can be derived as:

$$\mathcal{L}_{\text{Pixel}} = -\sum_{C(i)=C(j)} \log \frac{r\left(e_i, e_j\right)}{\sum_{k=1}^{N_{pixel}} r\left(e_i, e_k\right)}, \tag{3.4}$$

where e is the feature map extracted by the projection head $e = h_{pixel}g_\theta(x)$, and N_{pixel} is the number of pixels. e_i denotes the i-th feature on the feature map e. r denotes the similarity between the two pixel features. In particular, we deploy the exponential cosine similarity $r\left(e_i, e_j\right) = \exp\left(s\left(e_i, e_j\right)/\tau\right)$, where s is cosine similarity between two pixel features e_i and e_j, and τ is the temperature. As shown in Figure 3.2, with the guide of pixel-wise contrastive loss, the pixel embeddings of the same class are pulled close and those of the other classes are pushed apart, which promotes intra-class compactness and inter-class separability.

In the **patch-wise** effect region, in particular, given unlabeled target image x^T, we also leverage the network g_θ to extract the feature map of two partially overlapping patches. The cropped examples are shown at the bottom of Figure 3.2. We deploy an independent head h_{patch} with 2-layer MLPs to further project the output feature maps to the embedding space for comparison. As shown in Figure 2 module (b), overlapping region $\mathbf{O}_1$ and $\mathbf{O}_2$ denote the same green area in the original image. In practice, we first randomly select the region $\mathbf{O}$ and then sample two neighbor patches $\mathbf{M}$ covering $\mathbf{O}$. We use $\mathbf{M}$ to denote the entire patch **including** $\mathbf{O}$. We argue that the output features of the overlapping region should be invariant to the contexts. Therefore, we encourage that each feature in $\mathbf{O}_1$ to be consistent with the corresponding feature of the same location in $\mathbf{O}_2$. Similar to pixel-wise contrast, as shown in Figure 3.2 module (b), we regard two features at the same position of $\mathbf{O}_1$ and $\mathbf{O}_2$ as positive pair, and any two features in $\mathbf{M}_1$ and $\mathbf{M}_2$ at different positions of the original image are treated as a negative pair. Given a target-domain input x^T, the patch-wise contrast loss can be formulated as:

$$\mathcal{L}_{\text{Patch}} = -\sum_{O_1(i)=O_2(j)} \log \frac{r\left(f_i, f_j\right)}{\sum_{k=1}^{N_{patch}} r\left(f_i, f_k\right)}, \tag{3.5}$$

where f is the feature map extracted by the projection head $f = h_{patch}g_\theta(x)$, and N_{patch} is the number of pixels in $M_1 \cup M_2$. i is the pixel index in the patch $\mathbf{M}_1$, and j is for $\mathbf{M}_2$. $\mathbf{O}_1(i)$ denotes the location in the overlapping region $\mathbf{O}_1$. $\mathbf{O}_1(i) = \mathbf{O}_2(j)$ denotes i and j are the same pixel (location) in the original image, as shown in Figure 3.4(b). f_i denotes i-th feature in the map. Similarly, r denotes the exponential function of the cosine similarity as the one in pixel contrast. It is worth noting that we also enlarge the negative sample pool. In practice, the rest feature f_k not only comes from the union set $M_1 \cup M_2$, but also from other training images within the current batch.

Algorithm 1 PiPa algorithm

Input: Source-domain data X^S, Source-domain labels Y^S, Target domain data X^T, segmentation network that contains segmentation encoder g_θ, classification head h_{cls}, pixel projection head h_{pixel}, patch projection head h_{patch}, the total iteration number T_{total}.

Initialize network parameter θ with ImageNet pre-trained parameters. Initialize teacher network $\bar{\theta}$ randomly
for iteration = 1 to T_{total} **do**
  $x^S,\ y^S \sim U.$
  $x^T \sim V.$
  $\bar{y}^T \leftarrow argmax\left(h_{cls}\left(g_{\bar{\theta}}\left(x^T\right)\right)\right).$
  $x^{Mix},\ \bar{y}^{Mix} \leftarrow$ Augmentation and pseudo label from mixing x^S, y^S, x^T and $\bar{y}^T$.
  Compute predicti
  $\hat{y}^S \leftarrow argmax\left(h_{cls}\left(g_\theta\left(x^S\right)\right)\right),$
  $\hat{y}^{Mix} \leftarrow argmax\left(h_{cls}\left(g_\theta\left(x^{Mix}\right)\right)\right).$
  Compute loss for the mini-batch:
  $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{Pixel}} + \mathcal{L}_{\text{Patch}}.$
  Compute $\nabla_\theta \mathcal{L}_{\text{total}}$ by backpropagation.
  Perform stochastic gradient descent.
  Update teacher network $\bar{\theta}$ with θ.
end for
return student network g_θ and classification head h_{cls}.

3.2.3 Total Loss.

The overall training objective is the combination of pixel-level cross-entropy loss and the proposed PiPa:

$$\mathcal{L}_{total} = \mathcal{L}_{\text{ce}}^S + \mathcal{L}_{\text{ce}}^T + \alpha\mathcal{L}_{\text{Pixel}} + \beta\mathcal{L}_{\text{Patch}}, \tag{3.6}$$

where α and β are the weights for pixel-wise contrast $\mathcal{L}_{\text{Pixel}}$ and patch-wise contrast $\mathcal{L}_{\text{Patch}}$, respectively. We summarize the pipeline of PiPa via an Algorithm below.

3.2.4 Discussion.

1. Correlation between Pixel and Patch Contrast. Both pixel and patch contrast are derived from instance-level contrastive learning and share a common underlying idea,

i.e., contrast, but they work at different effect regions, *i.e.*, pixel-wise and patch-wise. The pixel contrast explores the pixel-to-pixel category correlation over the whole image, while patch-wise contrast imposes regularization on the semantic patches from a local perspective. Therefore, the two kinds of contrast are complementary and can work in a unified way to mine the intra-domain inherent context within the data.

2. What is the advantage of the proposed framework? Traditional UDA methods focus on learning shared inter-domain knowledge. Differently, we are motivated by the objectives of UDA semantic segmentation in a bottom-up manner, and thus leverage rich pixel correlations in the training data to facilitate intra-domain knowledge learning. By explicitly regularizing the feature space via PiPa, we enable the model to explore the inherent intra-domain context in a self-supervised setting, *i.e.*, pixel-wise and patch-wise, without extra parameters or annotations. Therefore, PiPa could be effortlessly incorporated into existing UDA approaches to achieve better results without extra overhead during testing.

3. Difference from conventional contrastive learning. Conventional contrastive learning methods typically tend to perform contrast in the instance or pixel level alone [197, 205, 208]. We formulate pixel- and patch-wise contrast in a similar format but focus on the local effect regions within the images, which is well aligned with the local-focused segmentation task. We show that the proposed local contrast, *i.e.*, pixel- and patch-wise contrasts, regularizes the domain adaptation training and guides the model to shed more light on the intra-domain context. Our experiment also verifies this point that pixel- and patch-wise contrast facilitates smooth edges between different categories and yields a higher accuracy on small objects.

3.3 Experiment

3.3.1 Experimental Setup

Datasets. We evaluate the proposed method on GTA → Cityscapes and SYNTHIA → Cityscapes, following common UDA protocols [5, 179, 189, 224, 248]. The target dataset Cityscapes, collected from the real-world street-view images, contains 2,975 unlabeled images for training, 500 images for validation, and 1525 images for testing. We report the results on Cityscapes validation set for comparisons.

Structure Details. Following recent SOTA UDA setting [184, 192, 248], our network consists of a SegFormer MiT-B5 backbone [248, 250] pretrained on ImageNet-1k [251] and several MLP-based heads, *i.e.*, h_{cls} , h_{pixel} and h_{patch} , which contains two fully-connected (fc) layers and ReLU activation between two fc layers. Note that the self-supervised projection heads h_{pixel} and h_{patch} are only applied at training time and are removed during inference, which does not introduce extra computational costs in deployment.

Implementation details. We train the network with batch size 2 for 60k iterations with a single NVIDIA RTX 6000 GPU. We adopt AdamW [252] as the optimizer, a learning rate of 6×10^{-5}, a linear learning rate warmup of 1.5k iterations and the weight decay of 0.01. Following [184, 192], the input image is resized to 1280×720 for GTA and 1280×760 for SYNTHIA, with a random crop size of 640×640. For the patch-wise contrast, we randomly resize the input images by a ratio between 0.5 and 2, and then randomly crop two patches of the size 720×720 from the resized image and ensure the Intersection-over-Union(IoU) value of the two patches between 0.1 and 1. We utilize the same data augmentation *e.g.*, color jitter, Gaussian blur and ClassMix [253] and empirically set pseudo labels threshold 0.968 following [179]. The exponential moving average parameter of the teacher network is 0.999. The hyperparameters of the loss function are chosen empirically $\alpha = \beta = 0.1$. The code is based on Pytorch [254].

3.3.2 Results Comparison

We compare PiPa with several competitive UDA methods on GTA $\rightarrow$ Cityscapes and SYNTHIA $\rightarrow$ Cityscapes, respectively. The quantitative comparisons are shown in Table 3.1 and Table 3.2, respectively. We also show the visual difference between the proposed method and the other two strong Transformer-based methods [5, 248] in Figure 3.3.

GTA $\rightarrow$ Cityscapes. Generally, our PiPa yields a significant improvement over the transformer-based models DAFormer[248] and HRDA[5], as shown in Table 3.1. Particularly, PiPa achieves 71.7 mIoU, which outperforms DAFormer by a considerable margin of +3.4 mIoU (with the trasformer backbone). Additionally, when applying PiPa to HRDA, which is a strong baseline that adopts high-resolution crops, we increase +1.8 mIoU and achieve the state-of-the-art performance of 75.6 mIoU, verifying the effectiveness of the proposed method that introduces a unified and multi-grained self-supervised learning algorithm in UDA task. Furthermore, PiPa achieves leading IoU of almost all classes on GTA $\rightarrow$ Cityscapes, including several small-scale objectives such as Fence, Pole, Wall and Training

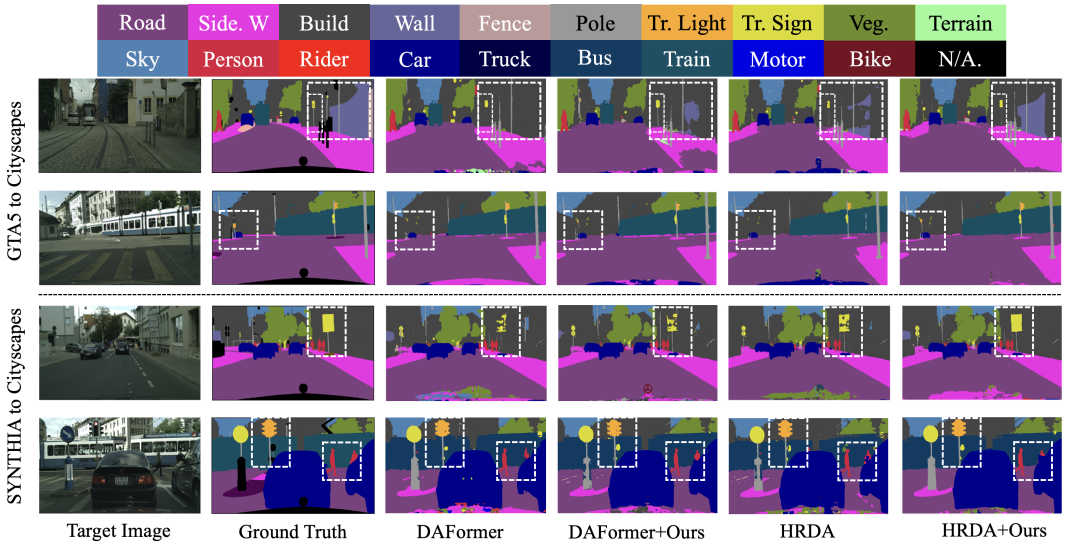

FIGURE 3.3: Qualitative results on GTA → Cityscapes and SYNTHIA → Cityscapes. From left to right: Target Image, Ground Truth, the visual results predicted by DAFormer, DAFormer + Ours (PiPa), HRDA, HRDA + Ours (PiPa). We deploy the white dash boxes to highlight different prediction parts.

Sign. Particularly, we increase the IoU of the Fence by +6.2 from 51.5 to 57.7 IoU. The IoU performance of PiPa verifies our motivation that the exploration of the inherent structures of intra-domain images indeed helps category recognition, especially for challenging small objectives. It is worth noting that CLUDA combines cross-domain contrastive learning with class-specific enhancement, which helps it better capture inter-class differences and achieve superior performance in several sub-categories.

SYNTHIA → Cityscapes. As revealed in Table 3.2, PiPa also achieves remarkable mIoU and mIoU* (13 most common categories) performance on SYNTHIA → Cityscapes, increasing +2.5 and +2.4 mIoU compared with DAFormer [248] and HRDA [5], respectively.

Qualitative results. In Figure 3.3, we visualize the segmentation results and the comparison with previous strong methods DAFormer [248], HRDA [5], and the ground truth on both GTA → Cityscapes and SYNTHIA → Cityscapes benchmarks. The results highlighted by white dash boxes show that PiPa is capable of segmenting minor categories such as 'wall', 'traffic sign' and 'traffic light'. It is also noticeable that PiPa predicts smoother edges between different categories, *e.g.*, 'person' in the fourth row of Figure 3.3. We think it is because the proposed method explicitly encourages patch-wise consistency against different contexts, which facilitates the prediction robustness on edges.

TABLE 3.1: [

Quantitative comparison with previous UDA methods on GTA → Cityscapes.]Quantitative comparison with previous UDA methods on GTA → Cityscapes. We present pre-class IoU and mIoU. The best accuracy in every column is in **bold**.

Method	**Road**	**SW**	**Build**	**Wall**	**Fence**	**Pole**	**TL**	**TS**	**Veg.**	**Terrain**	**Sky**	**PR**	**Rider**	**Car**	**Truck**	**Bus**	**Train**	**Motor**	**Bike**	**mIoU**
AdaptSegNet [162]	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
CyCADA [165]	86.7	35.6	80.1	19.8	17.5	38.0	39.9	41.5	82.7	27.9	73.6	64.9	19.0	65.0	12.0	28.6	4.5	31.1	42.0	42.7
CLAN [164]	87.0	27.1	79.6	27.3	23.3	28.3	35.5	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	31.9	31.4	43.2
SP-Adv [255]	86.2	38.4	80.8	25.5	20.5	32.8	33.4	28.2	85.5	36.1	80.2	60.3	28.6	78.7	27.3	36.1	4.6	31.6	28.4	44.3
MaxSquare [186]	88.1	27.7	80.8	28.7	19.8	24.9	34.0	17.8	83.6	34.7	76.0	58.6	28.6	84.1	37.8	43.1	7.2	32.3	34.2	44.3
ASA [256]	89.2	27.8	81.3	25.3	22.7	28.7	36.5	19.6	83.8	31.4	77.1	59.2	29.8	84.3	33.2	45.6	16.9	34.5	30.8	45.1
AdvEnt [163]	89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5
MRNet [174]	89.1	23.9	82.2	19.5	20.1	33.5	42.2	39.1	85.3	33.7	76.4	60.2	33.7	86.0	36.1	43.3	5.9	22.8	30.8	45.5
APODA [257]	85.6	32.8	79.0	29.5	25.5	26.8	34.6	19.9	83.7	40.6	77.9	59.2	28.3	84.6	34.6	49.2	8.0	32.6	39.6	45.9
CBST [171]	91.8	53.5	80.5	32.7	21.0	34.0	28.9	20.4	83.9	34.2	80.9	53.1	24.0	82.7	30.3	35.9	16.0	25.9	42.8	45.9
APODA [257]	85.6	32.8	79.0	29.5	25.5	26.8	34.6	19.9	83.7	40.6	77.9	59.2	28.3	84.6	34.6	49.2	8.0	32.6	39.6	45.9
PatchAlign [240]	92.3	51.9	82.1	29.2	25.1	24.5	33.8	33.0	82.4	32.8	82.2	58.6	27.2	84.3	33.4	46.3	2.2	29.5	32.3	46.5
MRKLD [175]	91.0	55.4	80.0	33.7	21.4	37.3	32.9	24.5	85.0	34.1	80.8	57.7	24.6	84.1	27.8	30.1	26.9	26.0	42.3	47.1
BL [181]	91.0	44.7	84.2	34.6	27.6	30.2	36.0	36.0	85.0	43.6	83.0	58.6	31.6	83.3	35.3	49.7	3.3	28.8	35.6	48.5
DT [232]	90.6	44.7	84.8	34.3	28.7	31.6	35.0	37.6	84.7	43.3	85.3	57.0	31.5	83.8	42.6	48.5	1.9	30.4	39.0	49.2
FADA [182]	91.0	50.6	86.0	43.4	29.8	36.8	43.4	25.0	86.8	38.3	87.4	64.0	38.0	85.2	31.6	46.1	6.5	25.4	37.1	50.1
Uncertainty [176]	90.4	31.2	85.1	36.9	25.6	37.5	48.8	48.5	85.3	34.8	81.1	64.4	36.8	86.3	34.9	52.2	1.7	29.0	44.6	50.3
FDA [237]	92.5	53.3	82.4	26.5	27.6	36.4	40.6	38.9	82.3	39.8	78.0	62.6	34.4	84.9	34.1	53.1	16.9	27.7	46.4	50.5
Adaboost [247]	90.7	35.9	85.7	40.1	27.8	39.0	49.0	48.4	85.9	35.1	85.1	63.1	34.4	86.8	38.3	49.5	0.2	26.5	45.3	50.9
SPCL [209]	90.3	50.3	85.7	45.3	28.4	36.8	42.2	22.3	85.1	43.6	87.2	62.8	39.0	87.8	41.3	53.9	17.7	35.9	33.8	52.1
DACS [179]	89.9	39.7	87.9	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
CorDA [224]	94.7	63.1	87.6	30.7	40.6	40.2	47.8	51.6	87.6	47.0	89.7	66.7	35.9	90.2	48.9	57.5	0.0	39.8	56.0	56.6
BAPA [258]	94.4	61.0	88.0	26.8	39.9	38.3	46.1	55.3	87.8	46.1	89.4	68.8	40.0	90.2	60.4	59.0	0.0	45.1	54.2	57.4
ProDA [177]	87.8	56.0	79.7	46.3	44.8	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	59.4	1.0	48.9	56.4	57.5
CaCo [208]	93.8	64.1	85.7	43.7	42.2	46.1	50.1	54.0	88.7	47.0	86.5	68.1	2.9	88.0	43.4	60.1	31.5	46.1	60.9	58.0
PiPa (CNN)	95.1	71.3	87.7	44.2	42.0	43.5	52.1	63.3	87.8	44.0	87.5	72.3	44.2	89.3	59.9	59.4	2.1	47.2	48.9	60.1
DAFormer [248]	95.7	70.2	89.4	53.5	48.1	49.6	55.8	59.4	89.9	47.9	92.5	72.2	44.7	92.3	74.5	78.2	65.1	55.9	61.8	68.3
CAMix [192]	96.0	73.1	89.5	53.9	50.8	51.7	58.7	64.9	90.0	51.2	92.2	71.8	44.0	92.8	78.7	82.3	70.9	54.1	64.3	70.0
DAFormer [248] + PiPa	96.1	72.0	90.3	56.6	52.0	55.1	61.8	63.7	90.8	52.6	93.6	74.3	43.6	93.5	78.4	84.2	77.3	59.9	66.7	71.7
HRDA [5]	96.4	74.4	91.0	61.6	51.5	57.1	63.9	69.3	91.3	48.4	94.2	79.0	52.9	93.9	84.1	85.7	75.9	63.9	67.5	73.8
CLUDA [259]	**97.1**	**78.0**	91.0	60.3	55.3	56.3	64.3	71.5	91.2	51.1	94.7	78.4	52.9	**94.5**	82.8	86.5	73.0	**64.2**	**69.7**	74.4
HRDA [5] + PiPa	96.8	76.3	**91.6**	**63.0**	**57.7**	**60.0**	**65.4**	**72.6**	**91.7**	**51.8**	**94.8**	**79.7**	**56.4**	94.4	**85.9**	**88.4**	**78.9**	63.5	67.2	**75.6**

Effect of Pixel-wise Contrast and Patch-wise Contrast. We evaluate the effectiveness of the two primary components, *i.e.*, Pixel-wise Contrast and Patch-wise Contrast in the proposed PiPa and investigate how the combination of two contrasts contributes to the final performance on GTA → Cityscapes. For a fair comparison, we apply the same experimental settings and hyperparameters. We first reproduce the baseline DAFormer [248], which yields a competitive mIoU of 68.4. As shown in the Table 3.3, we could observe: (1) Both Patch Contrast and Pixel Contrast individually could lead to +1.4 mIoU and +2.3 mIoU improvement respectively, verifying the effectiveness of exploring the inherent

TABLE 3.2: Quantitative comparison with previous UDA methods on SYNTHIA → Cityscapes. We present pre-class IoU, mIoU and mIoU*. mIoU and mIoU* are averaged over 16 and 13 categories, respectively. The best accuracy in every column is in **bold**.

Method	Road	SW	Build	Wall*	Fence*	Pole*	TL	TS	Veg.	Sky	PR	Rider	Car	Bus	Motor	Bike	mIoU*	mIoU
MaxSquare [186]	77.4	34.0	78.7	5.6	0.2	27.7	5.8	9.8	80.7	83.2	58.5	20.5	74.1	32.1	11.0	29.9	45.8	39.3
SIBAN [238]	82.5	24.0	79.4	–	–	–	16.5	12.7	79.2	82.8	58.3	18.0	79.3	25.3	17.6	25.9	46.3	–
PatchAlign [240]	82.4	38.0	78.6	8.7	0.6	26.0	3.9	11.1	75.5	84.6	53.5	21.6	71.4	32.6	19.3	31.7	46.5	40.0
AdaptSegNet [162]	84.3	42.7	77.5	–	–	–	4.7	7.0	77.9	82.5	54.3	21.0	72.3	32.2	18.9	32.3	46.7	–
CLAN [164]	81.3	37.0	80.1	–	–	–	16.1	13.7	78.2	81.5	53.4	21.2	73.0	32.9	22.6	30.7	47.8	–
SP-Adv [255]	84.8	35.8	78.6	–	–	–	6.2	15.6	80.5	82.0	66.5	22.7	74.3	34.1	19.2	27.3	48.3	–
AdvEnt [163]	85.6	42.2	79.7	8.7	0.4	25.9	5.4	8.1	80.4	84.1	57.9	23.8	73.3	36.4	14.2	33.0	48.0	41.2
ASA [256]	**91.2**	48.5	80.4	3.7	0.3	21.7	5.5	5.2	79.5	83.6	56.4	21.0	80.3	36.2	20.0	32.9	49.3	41.7
CBST [171]	68.0	29.9	76.3	10.8	1.4	33.9	22.8	29.5	77.6	78.3	60.6	28.3	81.6	23.5	18.8	39.8	48.9	42.6
DADA [221]	89.2	44.8	81.4	6.8	0.3	26.2	8.6	11.1	81.8	84.0	54.7	19.3	79.7	40.7	14.0	38.8	49.8	42.6
MRNet [174]	82.0	36.5	80.4	4.2	0.4	33.7	18.0	13.4	81.1	80.8	61.3	21.7	84.4	32.4	14.8	45.7	50.2	43.2
MRKLD [175]	67.7	32.2	73.9	10.7	1.6	37.4	22.2	31.2	80.8	80.5	60.8	29.1	82.8	25.0	19.4	45.3	50.1	43.8
CCM [260]	79.6	36.4	80.6	13.3	0.3	25.5	22.4	14.9	81.8	77.4	56.8	25.9	80.7	45.3	29.9	52.0	52.9	45.2
Uncertainty [176]	87.6	41.9	83.1	14.7	1.7	36.2	31.3	19.9	81.6	80.6	63.0	21.8	86.2	40.7	23.6	53.1	54.9	47.9
BL [181]	86.0	46.7	80.3	–	–	–	14.1	11.6	79.2	81.3	54.1	27.9	73.7	42.2	25.7	45.3	51.4	–
DT [232]	83.0	44.0	80.3	–	–	–	17.1	15.8	80.5	81.8	59.9	33.1	70.2	37.3	28.5	45.8	52.1	–
APODA [257]	86.4	41.3	79.3	–	–	–	22.6	17.3	80.3	81.6	56.9	21.0	84.1	49.1	24.6	45.7	53.1	–
Adaboost [247]	85.6	43.9	83.9	19.2	1.7	38.0	37.9	19.6	85.5	88.4	64.1	25.7	86.6	43.9	31.2	51.3	57.5	50.4
DAFormer [248]	84.5	40.7	88.4	41.5	6.5	50.0	55.0	54.6	86.0	89.8	73.2	48.2	87.2	53.2	53.9	61.7	67.4	60.9
CAMix [192]	87.4	47.5	88.8	–	–	–	55.2	55.4	87.0	91.7	72.0	49.3	86.9	57.0	57.5	63.6	69.2	–
DAFormer [248] + PiPa	87.9	48.9	88.7	45.1	4.5	53.1	59.1	58.8	87.8	92.2	75.7	49.6	88.8	53.5	58.0	62.8	70.1	63.4
HRDA [5]	85.2	47.7	88.8	49.5	4.8	57.2	65.7	60.9	85.3	92.9	79.4	52.8	89.0	64.7	63.9	64.9	72.4	65.8
CLUDA [259]	87.7	46.9	**90.2**	49.0	**7.9**	**59.5**	66.9	58.5	88.3	94.6	**80.1**	57.1	89.8	68.2	**65.5**	65.8	73.8	67.2
HRDA [5] + PiPa	**88.6**	**50.1**	90.0	**53.8**	7.7	58.1	**67.2**	**63.1**	**88.5**	**94.5**	79.7	**57.6**	**90.8**	**70.2**	65.1	**66.9**	**74.8**	**68.2**

contextual knowledge. (2) The two kinds of contrasts are complementary to each other. The proposed method successfully mines the multi-level knowledge by combining the two kinds of contrast. When applying both losses, our PiPa further improves the network performance to 71.7 mIoU, surpassing the model that deploys only one kind of contrast by a clear margin. The second baseline model is HRDA [5]. The observation is consistent with DAFormer. Using either pixel or patch loss could increase the performance, but jointly training them in a unified framework leads to the best results. Since HRDA introduces High Resolution (HR) and Low Resolution (LR) features, to effectively introduce Pixel-wise contrast and Patch-wise contrast in HRDA [5], we conducted experiments on both HR and LR features as shown in Table 3.4. It is shown that training with HR features results in higher performance.

TABLE 3.3: Ablation study on the effect of Pixel-wise Contrast and Patch-wise Contrast on GTA → Cityscapes based on two competitive baselines DAFormer[248] and HRDA[5].

Method	$\mathcal{L}_{\text{Pixel}}$	$\mathcal{L}_{\text{Patch}}$	mIoU	ΔmIoU
DAFormer[248]			68.4	–
Patch Contrast		✓	69.8	+1.4
Pixel Contrast	✓		70.7	+2.3
PiPa	✓	✓	71.7	+3.3
HRDA[5]			73.8	–
Patch Contrast		✓	74.7	+0.9
Pixel Contrast	✓		74.9	+1.1
PiPa	✓	✓	75.6	+1.8

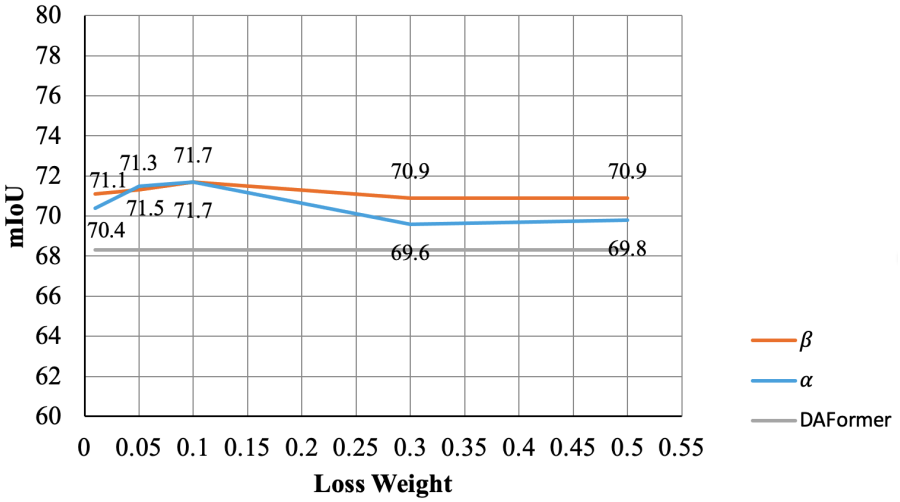

FIGURE 3.4: Ablation study on Loss Weights α and β.

Effect of the loss weight. We conduct loss weight sensitivity analysis on GTA → Cityscapes. Specifically, we change the weights α and β of the two kinds of contrasts in Eq 3.6, respectively. As shown in Figure 3.4, we can observe that both pixel-wise and patch-wise contrast are not sensitive to the relative weight. PiPa keeps outperforming the competitive DAFormer baseline of 68.3 mIoU in all compositions of loss weights. When applying the proposed method to an unseen environment, $\alpha = 0.1, \beta = 0.1$ can be a good initial weight to start.

Effect of the patch crop size. For the patch contrast, the size of the patch also affects

the number of negative pixels and training difficulty. As shown in Table 3.5, we gradually increase the patch size. We observe that larger patch generally obtain better performance since it contains more diverse contexts. There are two main advantages when increasing the patch size: (1) In larger patches, we could include more "hard negative" pixels for contrastive learning; (2) In larger patches, we have a larger receptive field, which could include contextual cues for bigger objects, such as trains. It is also worth noting that if the patch size is too large (like 960), the overlapping area can be larger than the non-overlapping area, which also may compromise the training.

Table 3.4: Effect of different crop types in HRDA [5].

Method	mIoU
LR Crops	75.1
HR Crops	75.6

Table 3.5: Effect of the patch crop size.

Crop Size	mIoU
480×480	70.4
600×600	71.0
720×720	71.7
900×900	70.9

Sensitivity of the pseudo label threshold. Since the target annotation is not available in unsupervised domain adaptation, a hard threshold beta is used to eliminate low-confidence pixel predictions from the predicted label. We conducted additional experiments on the threshold and found that within the range of 0.9-0.99, the DAFormer + PiPa results were not sensitive to the beta in Table 3.6. We set the threshold to 0.968 to obtain optimal results following previous self-training works [179, 248].

Multi source domain setting. By incorporating multi-source domain data, the model can be trained to be more robust to the unlabelled target environment [261, 262]. We first adopt previous work MADAN [261] as our baseline, which reaches 41.4 mIoU on GTA5 + SYNTHIA $\rightarrow$ Cityscapes. MADAN + PiPa increases the performance to 44.1 mIoU. Then we adopt a self-training baseline DACS [179], which achieves a mIoU of 52.1 (Only GTA) as shown in Table 3.7. By incorporating additional source-domain data, the model's performance improves to 54.2 mIoU. Our proposed method further improves

TABLE 3.6: Sensitivity analysis of the pseudo label threshold.

Threshold	0.6	0.7	0.8	0.9	0.95	0.968	0.99
mIoU	66.3	68.9	69.4	70.8	71.2	71.7	71.4

TABLE 3.7: Results on GTA5 + SYNTHIA → Cityscapes.

Base	Multi Src.	Multi Src + PiPa
52.1	54.2	56.1

the model's performance, increasing the mIoU from 54.2 to 56.1 mIoU, demonstrating consistent improvement over various baselines.

Ablation study on Normal-to-Adverse setting. ACDC is a large dataset with 4,006 images containing four common adverse conditions: fog, nighttime, rain and snow. In Cityscapes → ACDC, the knowledge is transferred from the source domain under normal visual conditions, *i.e.*, at daytime and in clear weather to adverse visual conditions. The quantitative comparisons are shown in Table 3.8. We can observe that our PiPa yields a significant improvement over the previous methods. Particularly, PiPa achieves 58.6 mIoU, which outperforms DAFormer by +3.2 mIoU, which demonstrates the competitive generalization ability of PiPa in adverse visual conditions. When plugging on recent works MIC [263] and Refign [264], PiPa shows consistent improvement.

Oxford RobotCar dataset [6] contains 894 training images with 9 classes and is collected during rainy and cloudy weather conditions, presenting a challenge due to the noisy variants introduced by such illumination conditions. We observe that the proposed method also has achieved the competitive results on Cityscapes → Oxford-Robot based on MRNet [174] and Uncertainty [176], reaching 1.8 and 2.1 mIoU increase respectively.

Ablation study on CNN-based architectures. In addition to Vision Transformer-based DA architectures, we also evaluate our PiPa on the DeepLabV2 [227] baseline with ResNet-101 backbone [10]. We do not pursue the SOTA performance here, but to demonstrate the relative improvement by plugging PiPa. Therefore, we do not search optimal hyper-parameters but follow the common setting. In Table 3.10, we show the adaptation performance of the baseline and our PiPa on GTA5 → Cityscapes. We also provide the performance of the DeepLabV2 trained merely on the source domain data, *i.e.*, Src-Only. It can be observed that PiPa improves the UDA baseline performance

TABLE 3.8: Quantitative comparison with previous UDA methods on Cityscapes → ACDC. The performance is provided as mIoU in % and the best result is in **bold**.

Method	Architecture	mIoU
ADVENT [163]	DeepLabv2	32.7
AdaptSegNet [162]	DeepLabv2	32.7
BDL [181]	DeepLabv2	37.7
CLAN [164]	DeepLabv2	39.0
FDA [237]	DeepLabv2	45.7
MGCDA [265]	DeepLabv2	48.7
DANNet [266]	DeepLabv2	50.0
DAFormer [248]	Transformer	55.4
DAFormer [248] + PiPa	Transformer	**58.6 (+3.2)**
MIC [263]	Transformer	59.2
MIC [263] + PiPa	Transformer	**61.1 (+1.9)**
Refign [264]	Transformer	65.5
Refign [264] + PiPa	Transformer	**66.4 (+0.9)**

TABLE 3.9: Quantitative Results on Cityscapes → Oxford-Robot [6]. The performance is provided as mIoU in % and the best result is in **bold**.

Method	road	sidewalk	building	light	sign	sky	person	automobile	two-wheel	mIoU
MRNet [174]	95.9	73.5	86.2	69.3	31.9	87.3	57.9	88.8	61.5	72.5
MRNet + PiPa	**96.9**	75.1	88.0	69.9	36.5	88.8	61.5	89.1	**63.1**	74.3
Uncertainty [176]	95.9	73.7	87.4	72.8	**43.1**	88.6	61.7	89.6	57.0	74.4
Uncertainty + PiPa	96.0	**76.2**	**93.3**	**73.3**	42.5	**90.9**	**65.4**	**91.1**	59.5	**76.5**

of DeepLabV2 by a large margin from 54.2 mIoU to 60.1 mIoU accuracy, still remains competitive.

Further experimental results on advanced architecture. We then apply our PiPa on the advanced method MIC [263]. MIC + PiPa achieves 77.3 mIoU (1.4 higher than MIC) on GTA-Cityscapes and 68.9 mIoU (1.6 higher than MIC) on SYNTHIA-Cityscapes, showing consistent improvement. The results are shown in Table 3.11.

TABLE 3.10: Quantitative result on a CNN-based architecture. The performance is provided as mIoU in %.

Src-Only	Baseline	Baseline+PiPa
34.3	54.2	60.1 (+5.9)

TABLE 3.11: Further study on advanced architecture. The performance is provided as mIoU in %.

Dataset	GTA-Cityscapes	SYNTHIA-Cityscapes
MIC [263]	75.9	67.3
MIC [263] + PiPa	77.3	68.9

3.4 Conclusion

In this chapter, we focus on the exploration of intra-domain knowledge, such as context correlation inside an image for the semantic segmentation domain adaptation. We target to learn a feature space that enables discriminative pixel-wise features and the robust feature learning of the overlapping patch against variant contexts. To this end, we propose PiPa, a unified pixel- and patch-wise self-supervised learning framework, which introduces pixel-level and patch-level contrast learning to UDA. PiPa encourages the model to mine the inherent contextual feature, which is domain invariant. Experiments show that PiPa outperforms the state-of-the-art approaches and yields a competitive 75.6 mIoU on GTA→Cityscapes and 67.4 mIoU on Synthia→Cityscapes. Since PiPa does not introduce extra parameters or annotations, it can be combined with other existing methods to further facilitate the intra-domain knowledge learning. In the future, we will continue to study the proposed PiPa on relevant tasks, such as domain adaptive video segmentation and open-set adaptation *etc.*

Chapter 4

Transferring to Real-World Layouts: A Depth-aware Framework for Scene Adaptation

4.1 Introduction

Semantic segmentation refers to the task of assigning pixel-level category labels in an image, which has achieved significant progress in the last few years [226, 227, 250, 267]. It is worth noting that prevailing models usually require large-scale training datasets with high-quality annotations, such as ADE20K [268], to achieve good performance and but such pixel-level annotations in real-world are usually unaffordable and time-consuming [269]. One straightforward idea is to train networks with synthetic data so that the pixel-level annotations are easier to obtain [23, 233]. However, the network trained with synthetic data usually results in poor scalability when being deployed to a real-world environment due to multiple factors, such as weather, illumination, and road design. Therefore, researchers resort to unsupervised domain adaptation (UDA) to further tackle the variance between domains. One branch of UDA methods attempts to mitigate the domain shift by aligning the domain distributions [162, 164, 165, 167, 270]. Another potential paradigm to heal the domain shift is self-training [171, 175, 176, 237, 271], which recursively refine the target pseudo-labels. Taking one step further, recent DACS [179] and follow-up works [5, 16, 184, 207, 224, 248, 272] combine self-training and ClassMix [253] to mix images from both source and target domain. In this way, these works could craft highly

perturbed samples to assist training by facilitating learning shared knowledge between two domains. Specifically, cross-domain mixing aims to copy the corresponding regions of certain categories from a source domain image and paste them onto an unlabelled target domain image. We note that such a vanilla strategy leads to pasting a large amount of objects to the unrealistic depth position. It is because that every category has its own position distribution. For instance, the background classes such as "sky" and "vegetation" usually appear farther away, while the classes that occupy a small number of pixels such as "traffic signs" and "pole", usually appear closer as shown in Figure 4.1 (a). Such crafted training data compromise contextual learning, leading to sub-optimal location prediction performance, especially for small objects.

To address these limitations, we observe the real-world depth distribution and find that semantic categories are easily separated (disentangled) in the depth map since they follow a similar distribution under certain scenarios, *e.g.*, urban. Therefore, we propose a new depth-aware framework, which contains Depth Contextual Filter (DCF) and a cross-task encoder. In particular, DCF removes unrealistic classes mixed with the real-world target training samples based on the depth information. On the other hand, multi-modal data could improve the performance of deep representations and the effective use of the deep multi-task features to facilitate the final predictions is crucial. The proposed cross-task encoder contains two specific heads to generate intermediate features for each task and an Adaptive Feature Optimization module (AFO). AFO encourages the network to optimize the fused multi-task features in an end-to-end manner. Specifically, the proposed AFO adopts a series of transformer blocks to capture the information that is crucial to distinguish different categories and assigns high weights to discriminative features and vice versa.

The main contributions are as follows:

(1) We propose a simple Depth-Guided Contextual Filter (DCF) to explicitly leverage the key semantic categories distribution hidden in the depth map, enhancing the realism of cross-domain information mixing and refining the cross-domain layout mixing.

(2) We propose an Adaptive Feature Optimization module (AFO) that enables the cross-task encoder to exploit the discriminative depth information and embed it with the visual feature which jointly facilitates semantic segmentation and pseudo depth estimation.

(3) Albeit simple, the effectiveness of our proposed methods has been verified by extensive ablation studies. Despite the pseudo depth, our method still achieves competitive

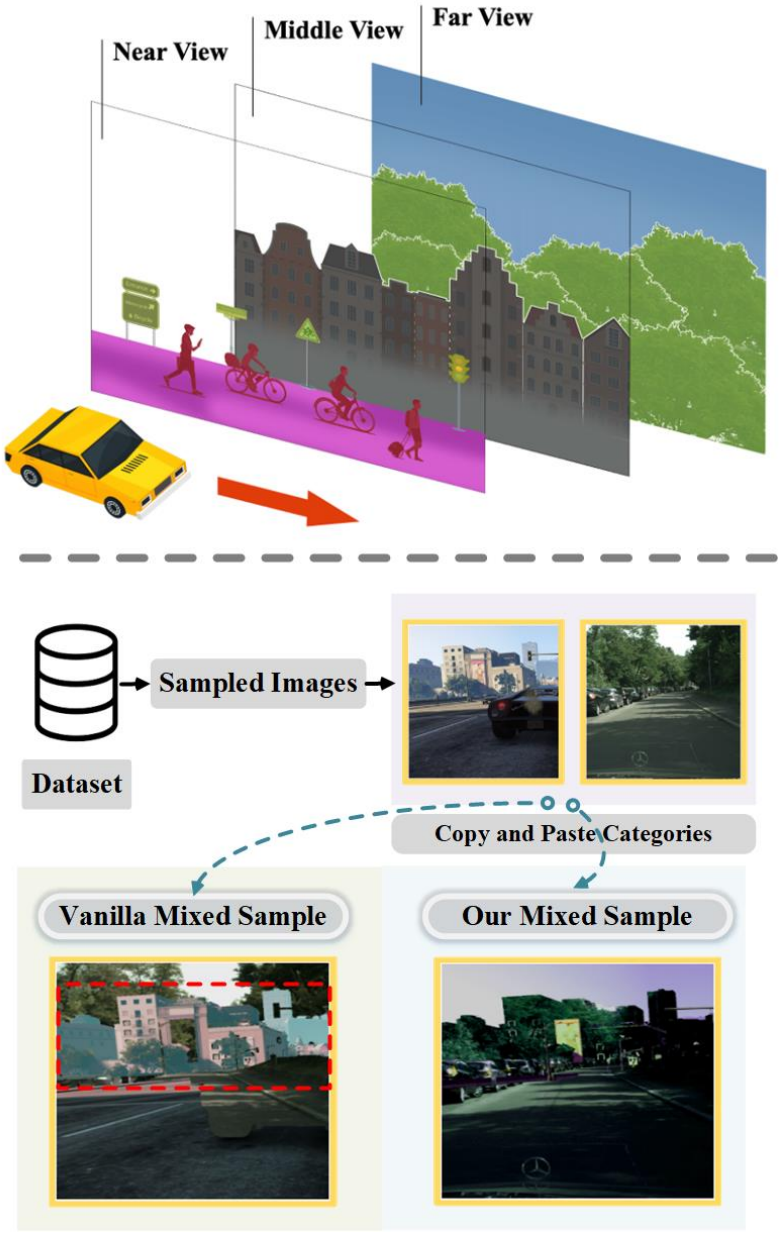

FIGURE 4.1: (a) Considering the driving scenario, we observe that the object location is relatively stable according to the distance from the camera. Therefore, we propose a Depth-guided Contextual Filter (DCF) which is aware of the semantic categories distribution in terms of Near, Middle, and Far view to facilitate cross-domain mixing. (b) Since we explicitly take the semantic layout into consideration, our method achieves more realistic mixed samples compared to the competitive MIC (Vanilla Mixed Sample) [272]. As shown in the red dotted box, "new" buildings are pasted before the parked cars.

accuracy on two commonly used scene adaptation benchmarks, namely 77.7 mIoU on GTA→Cityscapes and 69.3 mIoU on Synthia→Cityscapes.

4.2 Methodology

4.2.1 Problem Statement

In a typical Unsupervised Domain Adaptation (UDA) scenario, we have a source domain, denoted S, which consists of abundant labeled synthetic data. On the other hand, the target domain, represented by T, contains unlabeled real-world data. For example, we have labeled training samples $\left(\mathbf{x}^S, \mathbf{y}^S, \mathbf{z}^S \sim \mathbf{X}^S, \mathbf{Y}^S, \mathbf{Z}^S\right)$ in the source domain, where $\mathbf{x}^S, \mathbf{y}^S$ are the training image and the corresponding ground truth for semantic segmentation. $\mathbf{z}^S$ is the label for the depth estimation task. Similarly, we have unlabeled target images sampled from target domain data $\left(\mathbf{x}^T, \mathbf{z}^T \sim \mathbf{X}^T, \mathbf{Z}^T\right)$, where $\mathbf{x}^T$ is the unlabeled sample in the target domain and $\mathbf{z}^T$ is the label for the depth estimation task. Since depth annotation is not supported by common public datasets, we adopt pseudo depth that can be easily generated by the off-the-shelf model [273].

4.2.2 Depth-guided Contextual Filter

In UDA, recent works Recent UDA works [5, 16, 224, 248, 253, 272] often employ pixel mixing to create cross-domain augmented samples. The basic idea is straightforward: take a portion of pixels from a source domain image and transplant them onto an equivalent area in a target domain image. However, this simple approach faces challenges due to the inherent differences in structure and layout between source and target domain data. To decrease noisy signals and simulate augmented training samples with real-world layouts, we propose Depth-guided Contextual Filter (DCF) to reduce the noisy pixels that are naively mixed across domains. The implementation of DCF is represented as pseudo-code in Algorithm below, where the image $\mathbf{x}^S$ and the corresponding semantic labels $\mathbf{y}^S$ are sampled from source domain data. The image $\mathbf{x}^T$ and the depth label $\mathbf{z}^T$ are from target domain data. Pseudo label $\hat{\mathbf{y}}^T$ is then generated as $\hat{\mathbf{y}}^T = \mathcal{F}_\theta\left(\mathbf{x}^T\right)$, where $\mathcal{F}_\theta$ is a pre-trained semantic network. In practice, $\mathcal{F}_\theta$ usually has been trained on the source domain dataset via supervised learning.

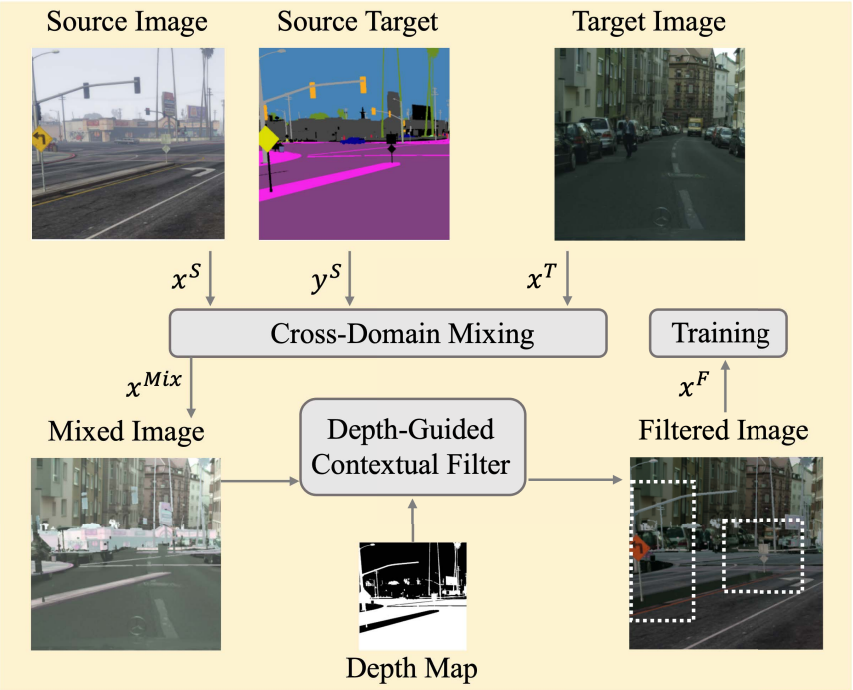

FIGURE 4.2: Source domain images x^S and x^T are mixed together, using the ground truth label y^S. The mixed images are de-noised by our proposed Depth-guided Contextual Filter (DCF) and then trained by the network. We illustrate DCF with a set of practical sample. As illustrated, the unrealistic "Building" pixels from the source image are mixed pasted to the target image, leading to a noisy mixed sample. The proposed DCF removes these pixels and maintain mixed pixels of "Traffic Sign" and "Pole" shown in the white dotted boxes, enhancing the realism of cross-domain mixing. (Best viewed when zooming in.)

Based on the hypothesis that most semantic categories usually fall under a finite depth range, we introduce DCF, which divides the target depth map $\mathbf{z}^T$ into a few discrete depth intervals $(\Delta z_1, ..., \Delta z_n)$. For a given real-world target input image $\mathbf{x}^T$ combined with the pseudo label $\hat{\mathbf{y}}^T$ and target depth map $\mathbf{z}^T$, the density value at each depth interval $(\Delta z_1, ..., \Delta z_n)$ for each class $i \in (1, \ldots, C)$ can be counted and normalized as a probability. We denote the density value for class i at the depth interval Δz_1 as $p_i(\Delta z_1)$. All the density values make up the depth distribution in the target domain image. Then we randomly select half of the categories on the source images to paste on the target domain image. In practice, we apply a binary mask $\mathcal{M}$ to denote the corresponding pixels. Then naive

Algorithm 1 Depth-guided Contextual Filter Algorithm with Cross-Image Mixing and Self Training

Input: Source domain: $(\mathbf{x}^S, \mathbf{y}^S, \mathbf{z}^S \sim \mathbf{X}^S, \mathbf{Y}^S, \mathbf{Z}^S)$, Target domain: $(\mathbf{x}^T, \mathbf{z}^T \sim \mathbf{X}^T, \mathbf{Z}^T)$. Semantic network $\mathcal{F}_\theta$.

Initialize network parameters θ randomly.
for iteration = 1 to n **do**
  $\hat{\mathbf{y}}^T \leftarrow \mathcal{F}_\theta\left(\mathbf{x}^T\right)$, Generate pseudo label
  Pre-calculate the density value $\mathbf{p}$ for each class i at each depth interval from the target depth map $\mathbf{z}^T$,
  $\hat{\mathbf{y}}^M \leftarrow \mathcal{M} \odot \mathbf{y}^S + (1 - \mathcal{M}) \odot \hat{\mathbf{y}}^T$, Randomly select 50% categories and copy the category ground truth label from the source image to target pseudo label
  $\mathbf{x}^M \leftarrow \mathcal{M} \odot \mathbf{x}^S + (1 - \mathcal{M}) \odot \mathbf{x}^T$, Copy the corresponding category region from the source image to the target image
  Re-calculate the density value $\hat{\mathbf{p}}$ after the mixing,
  Calculate the depth density distribution difference before and after mixing,
  Filter the category once the difference exceeds the threshold,
  Re-generate the depth-aware binary mask $\mathcal{M}^{DCF}$,
  $\hat{\mathbf{y}}^F \leftarrow \mathcal{M}^{\mathcal{DCF}} \odot \mathbf{y}^S + \left(1 - \mathcal{M}^{\mathcal{DCF}}\right) \odot \hat{\mathbf{y}}^T$, Generate the filtered training samples with new DCF mask
  $\mathbf{x}^F \leftarrow \mathcal{M}^{\mathcal{DCF}} \odot \mathbf{x}^S + \left(1 - \mathcal{M}^{\mathcal{DCF}}\right) \odot \mathbf{x}^T$,
  Compute predic
  $\bar{\mathbf{y}}^S \leftarrow argmax\left(\mathcal{F}_\theta\left(\mathbf{x}^S\right)\right)$,
  $\bar{\mathbf{y}}^F \leftarrow argmax\left(\mathcal{F}_\theta\left(\mathbf{x}^F\right)\right)$,
  Compute loss for the batch:
  $\ell \leftarrow \mathcal{L}\left(\bar{\mathbf{y}}^S, \mathbf{y}^S, \bar{\mathbf{y}}^F, \hat{\mathbf{y}}^F\right)$.
  Compute $\nabla_\theta \ell$ by backpropagation.
  Perform stochastic gradient descent.
end for
return $\mathcal{F}_\theta$

cross-domain mixed image $\mathbf{x}^{Mix}$ and the mixed label $\hat{\mathbf{y}}^{Mix}$ can be formulated as:

$$\mathbf{x}^{Mix} = \mathcal{M} \odot \mathbf{x}^S + (1 - \mathcal{M}) \odot \mathbf{x}^T, \tag{4.1}$$

$$\hat{\mathbf{y}}^{Mix} = \mathcal{M} \odot \mathbf{y}^S + (1 - \mathcal{M}) \odot \hat{\mathbf{y}}^T, \tag{4.2}$$

where $\odot$ denotes the element-wise multiplication of between the mask and the image. The naively mixed images are visualized in Figure 4.2. It could be observed that due to the depth distribution difference between two domains, pixels of "Building" category are mixed

from the source domain to the target domain, creating unrealistic images. Training with such training samples will compromise contextual learning. Therefore, we propose to filter the pixels that do not match the depth density distribution in the mixed image. After the naive mixing, we re-calculate the density value for each class at each depth interval. For example, the new density value for class i at the depth interval Δz_1 is denoted as $\hat{p}_i\left(\Delta z_1\right)$. Then we calculate the depth density distribution difference for each pasted category and denote the difference for class i at the depth interval Δz_1 as $\Delta p_i(\Delta z_1) = |p_i(\Delta z_1) - \hat{p}_i(\Delta z_1)|$. Once $\Delta p_i(\Delta z_1)$ exceeds the threshold of that category i, these pasted pixels are removed. After performing DCF, we confirm the final realistic pixels to be mixed and construct a depth-aware binary mask $\mathcal{M}^{\mathcal{DCF}}$, which is changed dynamically based on the depth layout of the current target image.

The filtered mixing samples are then generated. In practice, we directly apply the updated depth-aware mask to replace the original mask. Therefore, the new mixed sample and the label are as follows:

$$\mathbf{x}^F = \mathcal{M}^{\mathcal{DCF}} \odot \mathbf{x}^S + \left(1 - \mathcal{M}^{\mathcal{DCF}}\right) \odot \mathbf{x}^T, \tag{4.3}$$

$$\hat{\mathbf{y}}^F = \mathcal{M}^{\mathcal{DCF}} \odot \mathbf{y}^S + \left(1 - \mathcal{M}^{\mathcal{DCF}}\right) \odot \hat{\mathbf{y}}^T. \tag{4.4}$$

Because large objects such as “sky” and “terrain” usually aggregate and occupy a large amount of pixels and small objects only occupy a small amount of pixels in a certain depth range, we set different filtering thresholds for each category. DCF uses pseudo semantic labels for the target domain as there is no ground truth available. Since the label prediction is not stable in the early stage, we apply a warmup strategy to perform DCF after 10,000 iterations. Examples of the input images, naively mixed samples and filtered samples are presented in Figure 4.2. The sample after the process of the DCF module has the pixels from the source domain that match the depth distribution of the target domain, helping the network to better deal with the domain gap.

4.2.3 Multi-task Scene Adaptation Framework

In order to exploit the relation between segmentation and depth learning, we introduce a multi-task scene adaptation framework including a high resolution semantic encoder, and a cross-task shared encoder with a feature optimization module, which is depicted in Figure

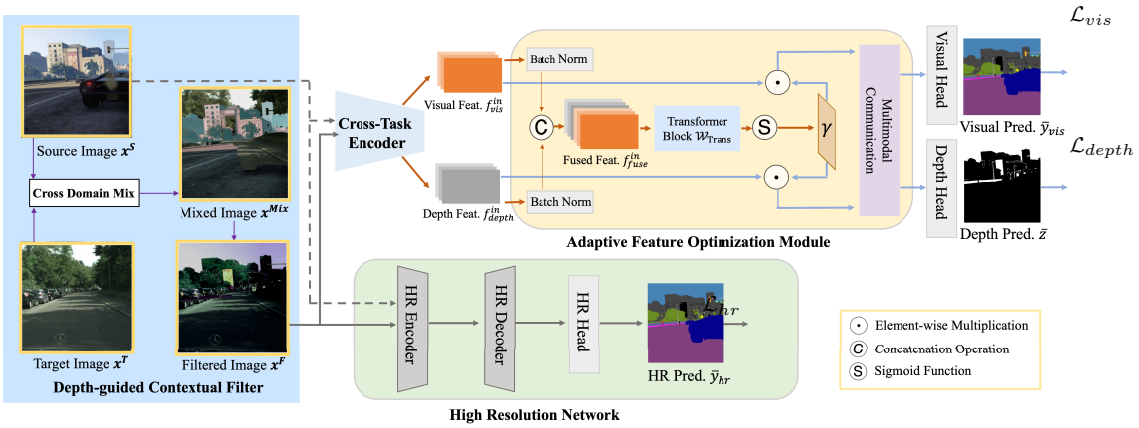

FIGURE 4.3: The proposed multi-task learning framework. The input images x^F are mixed from the source image x^S and target domain x^T according to the depth (Please refer to Figure 4.2). Then we are fed x^S and x^F into the high resolution encoder to generate high resolution predictions. To enhance multi-modal learning, the visual and depth feature created by the cross-task encoder are fused and fed into the proposed Adaptive Feature Optimization module (AFO) for multimodal communication. Finally, the multimodal communication via several transformer blocks incorporates and optimizes the fusion of depth information, improving the final visual predictions.

4.3. The proposed framework incorporates and optimizes the fusion of depth information for improving the final semantic predictions.

High Resolution Semantic Prediction. Most supervised methods use high resolution images for training, but common scene adaptation methods usually use random crops of the image that is half of the full resolution. To reduce the domain gap between scene adaptation and supervised learning while maintaining the GPU memory consumption, we adopt a high-resolution encoder to encode HR image crops into deep HR features. Then a semantic decoder is used to generate the HR semantic predictions $\bar{\mathbf{y}}_{hr}$. We adopt the cross entropy loss for semantic segmentation:

$$\mathcal{L}_{hr}^{S}\left(\mathbf{x}^{S},\mathbf{y}^{S}\right)=\mathbb{E}\left[-\mathbf{y}^{S}\log\bar{\mathbf{y}}_{hr}^{S}\right], \tag{4.5}$$

$$\mathcal{L}_{hr}^{F}\left(\mathbf{x}^{F},\mathbf{y}^{F}\right)=\mathbb{E}\left[-\hat{\mathbf{y}}^{F}\log\bar{\mathbf{y}}_{hr}^{F}\right], \tag{4.6}$$

where $\bar{\mathbf{y}}_{hr}^{S}$ and $\bar{\mathbf{y}}_{hr}^{T}$ are high resolution semantic predictions. $\mathbf{y}^S$ is the one-hot semantic label for the source domain and $\hat{\mathbf{y}}^F$ is the one-hot pseudo label for the depth-aware fused

domain.

Adaptive Feature Optimization. In addition to the high resolution encoder, We use another cross-task encoder to encode input images which are shared for both tasks. Depth maps are rich in spatial depth information, but a naive concatenation of depth information directly to visual information causes some interference, e.g. categories at similar depth positions are already well distinguished by visual information, and attention mechanisms can help the network to select the crucial part of the multitask information. In the proposed multi-task learning framework, the visual semantic feature and depth feature is generated by a visual head and a depth head, respectively. As shown in Figure 4.3, after applying batch normalization, an Adaptive Feature Optimization module then concatenates the normalized input visual feature and the input depth feature to create a fused multi-task feature by concatenation as $f_{fuse}^{in} = \text{CONCAT}\left(f_{vis}^{in}, f_{depth}^{in}\right)$. The fused feature is then fed into a series of transformer blocks to capture the key information between the two tasks. The attention mechanism adaptively adjusts the extent to which depth features are embedded in visual features:

$$f_{fuse}^{out} = \mathcal{W}_{Trans}\left(f_{fuse}^{in}\right), \tag{4.7}$$

where $\mathcal{W}_{Trans}$ is the transformer parameter. The learned output of the transformer blocks is a weight map γ which is multiplied back to the input visual feature and depth feature resulting in an optimized feature as:

$$\boldsymbol{\gamma} = \boldsymbol{\sigma}\left(\mathcal{W}_{Conv} \otimes f_{fuse}^{out}\right), \tag{4.8}$$

where $\mathcal{W}_{Conv}$ denotes the convolution parameter, $\otimes$ denotes the convolution operation and $\boldsymbol{\sigma}$ represents the sigmoid function. The weight matrix $\boldsymbol{\gamma}$ performs adaptive optimization of the muti-task features. Then, the fused feature f_{fuse}^{out} is fed into different decoders for predicting different final tasks, *i.e.*, the visual and the depth task. The output features are essentially multimodal features containing crucial depth information:

$$f_{vis}^{out} = f_{vis}^{in} \odot \boldsymbol{\gamma}, \quad f_{depth}^{out} = f_{depth}^{in} \odot \boldsymbol{\gamma}, \tag{4.9}$$

where $\odot$ represents element-wise multiplication. The optimized visual and depth feature is then fed into the multimodal communication module for further processing. The multimodal communication module refines the learning of key information between two tasks by iterative

use of transformer blocks. the inference is merely based on the visual input when the feature optimization is fished. The final semantic prediction $\bar{\mathbf{y}}_{vis}^{S}$ and depth prediction $\bar{\mathbf{z}}^{S}$ can be generated from the final visual feature f_{vis}^{final} and depth feature f_{depth}^{final} by the visual head and depth head . Similar to the high resolution predictions, we use the cross entropy loss for the semantic loss calculation:

$$\mathcal{L}_{vis}^{S}\left(\mathbf{x}^{S},\mathbf{y}^{S}\right)=\mathbb{E}\left[-\mathbf{y}^{S}\log\bar{\mathbf{y}}_{vis}^{S}\right], \tag{4.10}$$

$$\mathcal{L}_{vis}^{F}\left(\mathbf{x}^{F},\mathbf{y}^{F}\right)=\mathbb{E}\left[-\hat{\mathbf{y}}^{F}\log\bar{\mathbf{y}}_{vis}^{F}\right]. \tag{4.11}$$

We also employ the berHu loss for depth regression at source domain:

$$\mathcal{L}_{depth}^{S}\left(\mathbf{z}^{S}\right)=\mathbb{E}\left[\mathrm{berHu}\left(\bar{\mathbf{z}}^{S}-\mathbf{z}^{S}\right)\right], \tag{4.12}$$

where $\bar{z}$ and z are predicted and ground truth semantic maps. Following [221, 223], we deploy the reversed Huber criterion [274], which is defined as :

$$\begin{aligned}\mathrm{ber}\,Hu\left(e_z\right)&=\begin{cases}\left|e_z\right|, & \left|e_z\right|\leq H\\ \frac{(e_z)^2+H^2}{2H}, & \left|e_z\right|>H\end{cases}\\ H&=0.2\max\left(\left|e_z\right|\right),\end{aligned} \tag{4.13}$$

where H is a positive threshold and we set it to 0.2 of the maximum depth residual. Finally, the overall loss function is:

$$\mathcal{L}=\mathcal{L}_{hr}^{S}+\mathcal{L}_{vis}^{S}+\lambda_{depth}\mathcal{L}_{depth}^{S}+\mathcal{L}_{hr}^{F}+\mathcal{L}_{vis}^{F}, \tag{4.14}$$

where hyperparameter λ_{depth} is the loss weight. Considering that our main task is semantic segmentation and the depth estimation is the auxiliary task, we empirically $\lambda_{depth}=0.1\times10^{-2}$. We also designed the ablation studies to change the weight of depth task λ_{depth} to the level of 10^{-1} or 10^{-3}.

4.3 Experiment

4.3.1 Experimental Setup

Datasets. We evaluate the proposed framework on two scene adaptation settings, *i.e.*, GTA → Cityscapes and SYNTHIA → Cityscapes, following common protocols [5, 179, 189, 224, 248, 272]. Particularly, the GTA5 dataset [233] is the synthetic dataset collected from a video game, which contains 24,966 images annotated by 19 classes. Following [224], we adopt depth information generated by Monodepth2 [273] model which is trained merely on GTA image sequences. SYNTHIA [23] is a synthetic urban scene dataset with 9,400 training images and 16 classes. Simulated depth information provided by SYNTHIA is adopted. GTA and SYNTHIA serve as source domain datasets. The target domain dataset is Cityscapes, which is collected from real-world street-view images. Cityscapes contains 2,975 unlabeled training images and 500 validation images. The resolution of Cityscapes is 2048 × 1024 and the common protocol downscales the size to 1024 × 512 to save memory. Following [224], the stereo depth estimation from [19] is used. We leverage the Intersection Over Union (IoU) for per-class performance and the mean Intersection over Union (mIoU) over all classes to report the result. The code is based on Pytorch [254].

Experimental Setup. We adopt DAFormer [248] network with MiT-B5 backbone [250] for the high resolution encoder and DeepLabV2 network with ResNet-101 backbone for the cross-task encoder to reduce the memory consumption. All backbones are initialized with ImageNet pretraining. Our training procedure is based on self-training methods with cross-domain mixing [5, 179, 248, 272] and enhanced by our proposed Depth-guided Contextual Filter. Following [5, 179], the input image resolution is half of the full resolution for the cross-task encoder and full resolution for high resolution encoder. We utilize the same data augmentation, *e.g.*, color jitter and Gaussian blur and empirically set pseudo labels threshold 0.968 following [179]. We train the network with batch size 2 for 40k iterations on a Tesla V100 GPU.

Data Resolution. Our proposed depth-aware multi-task framework contains a high resolution encoder and a cross-task encoder with an adaptive feature optimization module (AFO). Previous works [179, 181, 240] downsample Cityscapes to 1024 × and GTA to 1280 × 720. Following [5], for the high resolution encoder, we resize GTA to 2560 × 1440 and SYNTHIA to 2560 x 1520. Then the crop size is 1024 × 1024. In addition, SegFormer [250] MLP decoder with an embedding dimension of 256 is used for the high resolution branch.

For the cross-task encoder branch, we follow common UDA methods [179, 248] to adopt 1024 × 512 pixels (half of the full resolution) for Cityscapes, 1280 × 760 for SYNTHIA and 1280 × 720 for GTA. In addition, a 512 × 512 random crop is extracted.

4.3.2 Results Comparison

Results on GTA→Cityscapes. We show our results on GTA → Cityscapes in Table 4.1 and highlight the best results in bold. It could be observed that our method yields significant performance improvement over the state-of-the-art method MIC [272] from 75.9 mIoU to 77.7 mIoU. Usually, classes that occupy a small number of pixels are difficult to adapt and have a comparably low IoU performance. However, our method demonstrates competitive IoU improvement on most categories especially on small objects such as +5.7 on "Rider", +5.4 on "Fence", +5.2 on "Wall", +4.4 on "Traffic Sign" and +3.4 on "Pole". The result shows the effectiveness of the proposed contextual filter and cross-task learning framework in the contextual learning. Our method also increases the mIoU performance of classes that aggregate and occupy a large amount of pixels in an image by a smaller margin such as +1.8 on "Pedestrain" and +1.1 on "Bike", probably because the rich texture and color information contained in the visual feature already has the ability to recognize these relatively easier classes. The above observations are also qualitatively reflected in Figure 4.4, where we visualize the segmentation results of the proposed method and the comparison with previous strong transformer-based methods HRDA [5], and MIC [272]. The qualitative results highlighted by white dash boxes show that the proposed method largely improved the prediction quality of challenging small object "Traffic Sign" and large category "Terrain".

Results on Synthia→Cityscapes. We show our results on SYNTHIA → Cityscapes in Table 4.2 and the results show the consistent performance improvement of our method, increasing from 67.3 to 69.3 (+2.0 mIoU) compared to the state-of-the-art method MIC [272]. Especially, our method significantly increases the IoU performance of the challenging class "SideWalk" from 50.5 to 63.1 (+12.6 mIoU). It is also noticeable that our method remains competitive in segmenting most individual classes and yields a significant increase of +6.8 on "Road", +6.6 on "Bus", +3.9 on "Pole", +3.7 on "Road", +3.2 on "Wall" and +2.9 on "Truck".

Ablation Study on Different Scene Adaptation Frameworks. We combine our method with different scene adaptation architectures on GTA→Cityscapes. Table 4.3 shows

TABLE 4.1: Quantitative comparison with previous UDA methods on GTA → Cityscapes. We present per-class IoU and mIoU. The best accuracy in every column is in **bold**. Our results are averaged over 3 random seeds.

Method	**Road**	**SW**	**Build**	**Wall**	**Fence**	**Pole**	**TL**	**TS**	**Veg.**	**Terrain**	**Sky**	**PR**	**Rider**	**Car**	**Truck**	**Bus**	**Train**	**Motor**	**Bike**	**mIoU**
AdaptSegNet [162]	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
CyCADA [165]	86.7	35.6	80.1	19.8	17.5	38.0	39.9	41.5	82.7	27.9	73.6	64.9	19.0	65.0	12.0	28.6	4.5	31.1	42.0	42.7
CLAN [164]	87.0	27.1	79.6	27.3	23.3	28.3	35.5	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	31.9	31.4	43.2
SP-Adv [255]	86.2	38.4	80.8	25.5	20.5	32.8	33.4	28.2	85.5	36.1	80.2	60.3	28.6	78.7	27.3	36.1	4.6	31.6	28.4	44.3
MaxSquare [186]	88.1	27.7	80.8	28.7	19.8	24.9	34.0	17.8	83.6	34.7	76.0	58.6	28.6	84.1	37.8	43.1	7.2	32.3	34.2	44.3
ASA [256]	89.2	27.8	81.3	25.3	22.7	28.7	36.5	19.6	83.8	31.4	77.1	59.2	29.8	84.3	33.2	45.6	16.9	34.5	30.8	45.1
AdvEnt [163]	89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5
MRNet [174]	89.1	23.9	82.2	19.5	20.1	33.5	42.2	39.1	85.3	33.7	76.4	60.2	33.7	86.0	36.1	43.3	5.9	22.8	30.8	45.5
APODA [257]	85.6	32.8	79.0	29.5	25.5	26.8	34.6	19.9	83.7	40.6	77.9	59.2	28.3	84.6	34.6	49.2	8.0	32.6	39.6	45.9
CBST [171]	91.8	53.5	80.5	32.7	21.0	34.0	28.9	20.4	83.9	34.2	80.9	53.1	24.0	82.7	30.3	35.9	16.0	25.9	42.8	45.9
MRKLD [175]	91.0	55.4	80.0	33.7	21.4	37.3	32.9	24.5	85.0	34.1	80.8	57.7	24.6	84.1	27.8	30.1	26.9	26.0	42.3	47.1
FADA [182]	91.0	50.6	86.0	43.4	29.8	36.8	43.4	25.0	86.8	38.3	87.4	64.0	38.0	85.2	31.6	46.1	6.5	25.4	37.1	50.1
Uncertainty [176]	90.4	31.2	85.1	36.9	25.6	37.5	48.8	48.5	85.3	34.8	81.1	64.4	36.8	86.3	34.9	52.2	1.7	29.0	44.6	50.3
FDA [237]	92.5	53.3	82.4	26.5	27.6	36.4	40.6	38.9	82.3	39.8	78.0	62.6	34.4	84.9	34.1	53.1	16.9	27.7	46.4	50.5
Adaboost [247]	90.7	35.9	85.7	40.1	27.8	39.0	49.0	48.4	85.9	35.1	85.1	63.1	34.4	86.8	38.3	49.5	0.2	26.5	45.3	50.9
DACS [179]	89.9	39.7	87.9	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
BAPA [258]	94.4	61.0	88.0	26.8	39.9	38.3	46.1	55.3	87.8	46.1	89.4	68.8	40.0	90.2	60.4	59.0	0.0	45.1	54.2	57.4
ProDA [177]	87.8	56.0	79.7	46.3	44.8	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	59.4	1.0	48.9	56.4	57.5
CaCo [208]	93.8	64.1	85.7	43.7	42.2	46.1	50.1	54.0	88.7	47.0	86.5	68.1	2.9	88.0	43.4	60.1	31.5	46.1	60.9	58.0
DAFormer [248]	95.7	70.2	89.4	53.5	48.1	49.6	55.8	59.4	89.9	47.9	92.5	72.2	44.7	92.3	74.5	78.2	65.1	55.9	61.8	68.3
CAMix [192]	96.0	73.1	89.5	53.9	50.8	51.7	58.7	64.9	90.0	51.2	92.2	71.8	44.0	92.8	78.7	82.3	70.9	54.1	64.3	70.0
HRDA [5]	96.4	74.4	91.0	61.6	51.5	57.1	63.9	69.3	91.3	48.4	94.2	79.0	52.9	93.9	84.1	85.7	75.9	63.9	67.5	73.8
MIC [272]	97.4	80.1	91.7	61.2	56.9	59.7	66.0	71.3	91.7	51.4	**94.3**	79.8	56.1	94.6	85.4	**90.3**	80.4	64.5	68.5	75.9
CorDA† [224]	94.7	63.1	87.6	30.7	40.6	40.2	47.8	51.6	87.6	47.0	89.7	66.7	35.9	90.2	48.9	57.5	0.0	39.8	56.0	56.6
FAFS† [275]	93.4	60.7	88.0	43.5	32.1	40.3	54.3	53.0	88.2	44.5	90.0	69.5	35.8	88.7	34.1	53.9	41.3	51.7	54.7	58.8
DBST† [275]	94.3	60.0	87.9	50.5	43.0	42.6	50.8	51.3	88.0	45.9	89.7	68.9	41.8	88.0	45.8	63.8	0.0	50.0	55.8	58.8
Ours†	**97.5**	**80.7**	**92.1**	**66.4**	**62.3**	**63.1**	**67.7**	**75.7**	**91.8**	**52.4**	93.9	**81.6**	**61.8**	**94.7**	**88.3**	90.0	**81.2**	**65.8**	**69.6**	**77.7 ± 0.3**

†: Training with depth data.

that our method achieves consistent and significant improvements across different methods with different network architectures. Firstly, our method improves the state-of-the-art performance by +1.8 mIoU. Then we evaluate the proposed method on two strong methods based on transformer backbone, yielding +3.2 mIoU and +2.3 mIoU performance increase on DAFormer [248] and HRDA [5], respectively. Secondly, we evaluate our method on DeepLabV2 [227] architecture with ResNet-101 [10] backbone. We show that we improve the performance of the CNN-based cross-domain mixing method, *i.e.*, DACS by +4.1 mIoU. The ablation study verifies the effectiveness of our method in leveraging depth information to enhance cross-domain mixing not only on Transformer-based networks but also on CNN-based architecture.

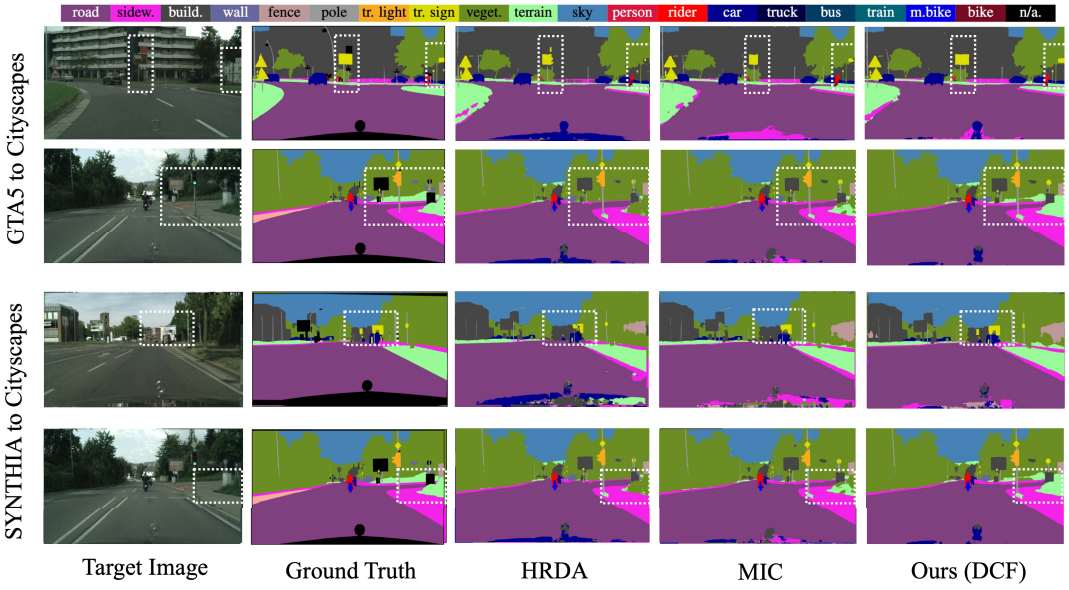

FIGURE 4.4: Qualitative results on GTA → Cityscapes. From left to right: Target Image, Ground Truth, the visual results predicted by HRDA, MIC and Ours. We highlight prediction differences in white dash boxes. The proposed method could predict clear edges.

Ablation Study on Different Components of the Proposed Method. In order to verify the effectiveness of our proposed components, we train four different models from M1 to M4 and show the result in Table 4.4. "ST Base" means the self training baseline with semantic segmentation branch and depth regression branch. "Naive Mix" denotes the cross-domain mixing strategy. "DCF" represents the proposed depth-aware mixing (Depth-guided Contextual Filter). "AFO" denotes the proposed Adaptive Feature Optimization module and we used two different method to perform AFO. Firstly, we leverage channel attention (CA) that could select useful information along the channel dimension to perform the feature optimization. In this method, the fused feature is adaptively optimized by SENet [277], the output is a weighted vector which is multiplied back to the visual and depth feature. We leavrage "AFO (CA)" to denote this method. Secondly, we leverage the iterative use of transformer block to adaptively optimize the multi-task feature. In this case, the output of the transformer block is a weighted map. The Multimodal Communication (MMC) module is then used to incorporate rich knowledge from the depth prediction. We denote this method as "AFO (Trans + MMC)". M1 is the self training baseline with depth regression based on DAFormer architecture. M2 adds the cross-domain mixing

TABLE 4.2: Quantitative comparison with previous UDA methods on SYNTHIA → Cityscapes. We present per-class IoU, mIoU, and mIoU*. mIoU and mIoU* are averaged over 16 and 13 categories, respectively. The best accuracy in every column is in **bold**. Our results are averaged over 3 random seeds.

Method	**Road**	**SW**	**Build**	**Wall***	**Fence***	**Pole***	**TL**	**TS**	**Veg.**	**Sky**	**PR**	**Rider**	**Car**	**Bus**	**Motor**	**Bike**	**mIoU***	**mIoU**
MaxSquare [186]	77.4	34.0	78.7	5.6	0.2	27.7	5.8	9.8	80.7	83.2	58.5	20.5	74.1	32.1	11.0	29.9	45.8	39.3
SIBAN [238]	82.5	24.0	79.4	–	–	–	16.5	12.7	79.2	82.8	58.3	18.0	79.3	25.3	17.6	25.9	46.3	–
PatchAlign [240]	82.4	38.0	78.6	8.7	0.6	26.0	3.9	11.1	75.5	84.6	53.5	21.6	71.4	32.6	19.3	31.7	46.5	40.0
AdaptSegNet [162]	84.3	42.7	77.5	–	–	–	4.7	7.0	77.9	82.5	54.3	21.0	72.3	32.2	18.9	32.3	46.7	–
CLAN [164]	81.3	37.0	80.1	–	–	–	16.1	13.7	78.2	81.5	53.4	21.2	73.0	32.9	22.6	30.7	47.8	–
SP-Adv [255]	84.8	35.8	78.6	–	–	–	6.2	15.6	80.5	82.0	66.5	22.7	74.3	34.1	19.2	27.3	48.3	–
AdvEnt [163]	85.6	42.2	79.7	8.7	0.4	25.9	5.4	8.1	80.4	84.1	57.9	23.8	73.3	36.4	14.2	33.0	48.0	41.2
ASA [256]	**91.2**	48.5	80.4	3.7	0.3	21.7	5.5	5.2	79.5	83.6	56.4	21.0	80.3	36.2	20.0	32.9	49.3	41.7
CBST [171]	68.0	29.9	76.3	10.8	1.4	33.9	22.8	29.5	77.6	78.3	60.6	28.3	81.6	23.5	18.8	39.8	48.9	42.6
MRNet [174]	82.0	36.5	80.4	4.2	0.4	33.7	18.0	13.4	81.1	80.8	61.3	21.7	84.4	32.4	14.8	45.7	50.2	43.2
MRKLD [175]	67.7	32.2	73.9	10.7	1.6	37.4	22.2	31.2	80.8	80.5	60.8	29.1	82.8	25.0	19.4	45.3	50.1	43.8
CCM [260]	79.6	36.4	80.6	13.3	0.3	25.5	22.4	14.9	81.8	77.4	56.8	25.9	80.7	45.3	29.9	52.0	52.9	45.2
Uncertainty [176]	87.6	41.9	83.1	14.7	1.7	36.2	31.3	19.9	81.6	80.6	63.0	21.8	86.2	40.7	23.6	53.1	54.9	47.9
BL [181]	86.0	46.7	80.3	–	–	–	14.1	11.6	79.2	81.3	54.1	27.9	73.7	42.2	25.7	45.3	51.4	–
DT [232]	83.0	44.0	80.3	–	–	–	17.1	15.8	80.5	81.8	59.9	33.1	70.2	37.3	28.5	45.8	52.1	–
IAST [276]	81.9	41.5	83.3	17.7	4.6	32.3	30.9	28.8	83.4	85.0	65.5	30.8	86.5	38.2	33.1	52.7	49.8	–
DAFormer [248]	84.5	40.7	88.4	41.5	6.5	50.0	55.0	54.6	86.0	89.8	73.2	48.2	87.2	53.2	53.9	61.7	67.4	60.9
CAMix [192]	87.4	47.5	88.8	–	–	–	55.2	55.4	87.0	91.7	72.0	49.3	86.9	57.0	57.5	63.6	69.2	–
HRDA [5]	85.2	47.7	88.8	49.5	4.8	57.2	65.7	60.9	85.3	92.9	79.4	52.8	89.0	64.7	63.9	64.9	72.4	65.8
MIC [272]	86.6	50.5	89.3	47.9	7.8	59.4	66.7	63.4	87.1	**94.6**	**81.0**	**58.9**	90.1	61.9	**67.1**	64.3	74.0	67.3
DADA [221]	89.2	44.8	81.4	6.8	0.3	26.2	8.6	11.1	81.8	84.0	54.7	19.3	79.7	40.7	14.0	38.8	49.8	42.6
CorDA† [224]	93.3	61.6	85.3	19.6	5.1	37.8	36.6	42.8	84.9	90.4	69.7	41.8	85.6	38.4	32.6	53.9	62.8	55.0
Ours†	**93.4**	**63.1**	**89.8**	**51.1**	**9.1**	**61.4**	**66.9**	**64.0**	**88.0**	94.5	80.9	56.6	**90.9**	**68.5**	63.7	**66.6**	**75.9**	**65.9 ± 0.2**

†: Training with depth data.

strategy for improvement and shows a competitive result of 76.0 mIoU. M3 is the model with the Depth-guided Contextual Filter, increasing the performance from 76.0 to 77.1 mIoU (+1.1 mIoU), which demonstrates the effectiveness of transferring the mixed training images to real-world layout with the help of the depth information. M4 adds the multi-task framework that leverages Channel Attention (CA) mechanism to fuse the discriminative depth feature into the visual feature. The segmentation result is increased by a small margin (+0.2 mIoU), which means CA could help the network to adaptively learn to focus or to ignore information from the auxiliary task to some extent. M5 is our proposed depth-aware multi-task model with both Depth-guided Contextual Filter and Adaptive

TABLE 4.3: Compatibility of the proposed method on different UDA methods and backbones on GTA→Cityscapes. Our results are averaged over 3 random seeds.

Backbone	UDA Method	w/o	w/	Diff.
DeepLabV2 [227]	DACS [179]	52.1	56.2	+4.1
DAFormer [248]	DAFormer [248]	68.3	71.5	+3.2
DAFormer [248]	HRDA [5]	73.8	76.1	+2.3
DAFormer [248]	MIC [272]	75.9	77.7 ± 0.3	+1.8

TABLE 4.4: Ablation study of different components of our proposed framework on GTA→Cityscapes. The results are averaged over 3 random seeds.

Method	ST Base.	Naive Mix.	DCF.	AFO. (CA)	AFO. (Trans + MMC)	mIoU↑
M1	✓					73.1
M2	✓	✓				76.0
M3	✓	✓	✓			77.1
M4	✓	✓	✓	✓		77.3
M5	✓	✓	✓		✓	77.7 ± 0.3

Feature Optimization (AFO) module. Compared to M3, M5 has a mIoU increase of +0.6 from 77.1 to 77.7, which shows the effectiveness of multi-modal feature optimization using transformers to facilitate contextual learning.

Ablation study on GTA+SYNTHIA → Cityscapes. We evaluate the proposed method on multi-source domains setting and report the quantitative result on GTA+SYNTHIA → Cityscapes. With multi-source domain data, the model can be trained more robust to the unlabelled target environment. We adopt DACS [179] as our baseline with 52.1 mIoU (Only GTA) performance shown in Table 4.5. With more source-domain data, the model yields a better result of 54.2 mIoU. Then, we can observe that our method yields a larger improvement from 54.2 to 56.7 mIoU, demonstrating that the proposed model could adapt multi-domain depth to the target domain and hence increase performance.

TABLE 4.5: Quantitative results on GTA+SYNTHIA → to Cityscapes. The performance is provided as mIoU in %.

Baseline (Single Source)	Multi Source	Multi Source + Depth
52.1	54.2	56.7

4.4 Conclusion

In this chapter, we introduce a new depth-aware scene adaptation framework that effectively leverages the guidance of depth to enhance data augmentation and contextual learning. The proposed framework not only explicitly refines the cross-domain mixing by stimulating real-world layouts with the guidance of depth distributions of objects, but also introduced a cross-task encoder that adaptively optimizes the multi-task feature and focused on the discriminative depth feature to help contextual learning. By integrating our depth-aware framework into existing self-training methods based on either transformer or CNN, we achieve state-of-the-art performance on two widely used benchmarks and a significant improvement on small-scale categories. Extensive experimental results verify our motivation to transfer the training images to real-world layouts and demonstrate the effectiveness of our multi-task framework in improving scene adaptation performance.

Chapter 5

GvSeg: General and Task-oriented Video Segmentation

5.1 Introduction

Identifying target objects and then inferring their spatial locations over time in a pixel observation constitute fundamental challenges in computer vision [144]. Depending on discriminating unique instances or semantics associated with targets, exemplary tasks include: *exemplar-guided* video segmentation (EVS) that tracks objects with given annotations at the first frame, video *instance* segmentation (VIS), video *semantic* segmentation (VSS), and video *panoptic* segmentation (VPS) which entails the delineation of foreground instance tracklets, while simultaneously assigning semantic labels to each video pixel. Prevalent work primarily adheres to discrete technical protocols customized for each task, showcasing promising results [2, 7, 89, 92, 97, 117, 129–131, 145–147, 150, 151, 278–283]. Nevertheless, these approaches necessitate meticulous architectural designs for each unique task, thereby posing challenges in facilitating research endeavors devoting on one task to another. Recently, there have been efforts in shifting the above *task-specific* paradigm to a *general* solution that can be applied across multiple distinct tasks [87, 126, 158–160]. However, one concern naturally arises that such a highly homogenized framework would overlook the diversity between tasks, potentially leading to suboptimal performance. For instance, the segmenting and tracking of objects like *human* prioritize *instance discrimination* in VIS but lean towards *semantic recognition* in VSS. However, prior general approaches adopt exactly

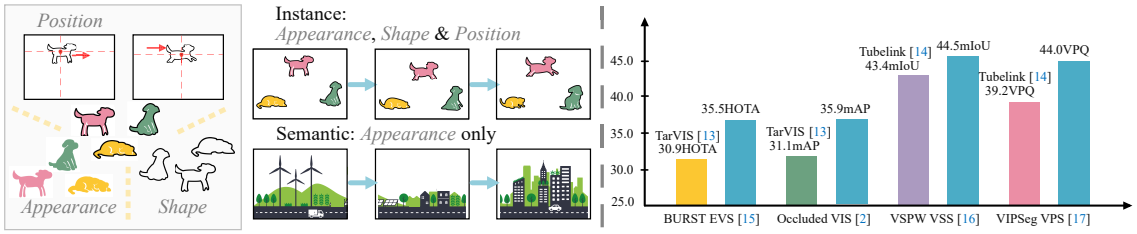

FIGURE 5.1: (a) We render holistic modeling on segment targets by disentangling them into appearance, shape and position. (b) By adjusting the involvement of the above three factors into tracking and segmentation according to task requirement, GVSEG achieves remarkable improvement compared to prior top-leading general solutions.

same query initialization, matching and space-time learning strategies [126, 158, 160], lacking tailored differentiation within the algorithm design that caters to the specific properties of individual tasks.

In this chapter, we present GVSEG, a **g**eneral **v**ideo **seg**mentation framework to address EVS, VIS, VSS, and VPS that can seamlessly accommodate ***task-oriented*** properties into the learning and inference process, while maintaining an ***identical*** architectural design. To achieve this, we rethink video segmentation in two aspects: ❶ what are the key factors that constitute segment targets (*i.e.*, *instance*, *thing*, and *stuff*), and ❷ how to leverage these key factors to build a unique sequential observation for each specific task within a general model. To address ❶, we delve deeply into the mechanism of how individuals can effectively discriminate moving instances or background stuff. The most intuitive answer in this regard is appearance, aligning with current video solutions where binary masks are classified solely based on visual representations (*i.e.*, **appearance**) [107, 117, 119, 284]. However, human perception extends beyond mere appearance [46, 285, 286]. For instance, we can also recognize moving entities such as cats in low-light conditions by referring to sketches (*i.e.*, **shape**), and distinguish distinct instances on the basis of respective spatial locations (*i.e.*, **position**), even in fast motion. Therefore, it is noteworthy that the instances to be segmented usually carry rich cues encompassing not only appearance but also position and shape characteristics. In light of the analysis above, we could assert three significant observations that contribute to the resolution of ❷: **First**, it becomes evident that current solutions downplay the importance of position and consistently ignore shape, in favor of solely appearance-based discrimination. To tackle this, we derive a *shape-position*

descriptor for each object, followed by encoding them into the cross-frame query matching process to enable the participation of three key factors in discriminating corresponding instances across the entire video. **Second**, it is crucial to acknowledge that the engagement of appearance, position, and shape cues should be adjusted in accordance with the task requirements. In current general solutions, all queries are roughly initialized as empty and matched in the same manner. However, for semantic classes VSS and background *stuff* in VPS, there is no instance discrimination and overly emphasize shape/location cues would harm the generalization of the model to various targets with the same semantics. Concerning this, we advocate for a tailored query initialization and object association strategies for each task by adjusting the relative contribution of three key elements. **Third**, owing to the absence of disentanglement on segment targets, the widely used temporal contrastive learning [117, 118, 158, 160] strategy for object association in current solutions is deemed suboptimal. Concretely, prior work empirically chooses objects in nearby frames as positive samples, remaining unaware of why excluding the same instance in distant frames. In fact, entities moving in long temporal range may display similar **appearance**, but undergo strong **shape** distortion, rendering them unsuitable as positive samples for instance discrimination. Therefore, we devise a task-oriented sampling strategy that caters to *thing* and *stuff*, where instance examples are selectively sampled from the entire video by referring to shape similarity and location distance. This not only makes full use of the pre-defined *shape-position descriptors*, but also recollects valuable samples that were arbitrarily discarded in prior work. In a similar spirit, the *stuff* examples are gathered from the whole dataset which renders rich semantic description for each semantic class. Through an in-depth analysis of the essential elements that compose segmentation targets and subsequently derive task-oriented insights, our work exhibits several compelling facets: **First**, it not only recognizes but also effectively harnesses the unique nature of each task, enabling seamless accommodation of task-specific properties into segmentation models. **Second**, all of our designs are architecture-agnostic, preserving a uniform structural to efficiently address task diversity. **Third**, GvSeg substantially attains remarkable performance on each task. Notably, it surpasses existing general solutions by **4.6%** HOTA on BURST [9], **1.3%** AP on YouTube-VIS 2021 [2], **4.8%** AP on Occluded-VIS [8], **1.1%** mIoU on VSPW [3], **4.8%** VPQ on VIPSeg [1], establishing new SOTA.

5.2 Methodology

5.2.1 Problem Statement

Video segmentation seeks to partition a video clip $V \in \mathbb{R}^{THW \times 3}$ containing T frames of size $H \times W$ into K non-overlap tubes linked along the time axis:

$$\{Y_k\}_{k=1}^{K} = \{(M_k, c_k)\}_{k=1}^{K}, \tag{5.1}$$

where each tube mask $M_k \in \{0,1\}^{T \times H \times W}$ is labeled with a category $c_k \in \{1, \cdots, C\}$. The value of K varies across tasks: in VSS, it is consistent with the number of predefined semantic categories; in EVS and VIS, it is adjusted in response to the instance count; and in VPS, it is the sum of *stuff* categories and *thing* entities.

5.2.2 Tracking by Query Matching

Inspired by the success of *query-based* object detectors, [117, 118, 158] propose to associate instances based on the query embeddings. Specifically, given a set of N randomly initialized queries $\{\boldsymbol{q}_n^t\}_{n=1}^{N}$, we can derive the object-centric representation $\{\hat{\boldsymbol{q}}_n^t\}_{n=1}^{N}$ for frame V^t by:

$$\{\hat{\boldsymbol{q}}_n^t\}_{n=1}^{N} = \mathcal{D}(\mathcal{E}(V^t), \{\boldsymbol{q}_n^t\}_{n=1}^{N}), \tag{5.2}$$

where $\mathcal{E}$ and $\mathcal{D}$ are the Transformer encoder and decoder. Here $\hat{\boldsymbol{q}}_n^t$ refines rich appearance representation for a specific object. The tracking is done by applying Hungarian Matching on the affinity matrix $\mathcal{S}_{ij} = \texttt{cosine}(\hat{\boldsymbol{q}}_i^t, \hat{\boldsymbol{q}}_j^{t+1})$ computed between $\hat{\boldsymbol{q}}_i^t$ and $\hat{\boldsymbol{q}}_j^{t+1}$ of two successive frame V^t and V^{t+1}. As such, instances exhibiting identical attributes across the video sequence are linked automatically.

5.2.3 GvSeg: Task-Oriented Property Accommodation Framework

GvSeg seeks to advance general video segmentation through controllable emphasis on instance discrimination and semantic comprehension according to task requirements. Concretely, we first devise a new shape-position descriptor to accurately reveal the shape and location of targets. Then, by adjusting the engagement of above shape-position descriptor during cross-frame query matching, we could realize controllable association for instance

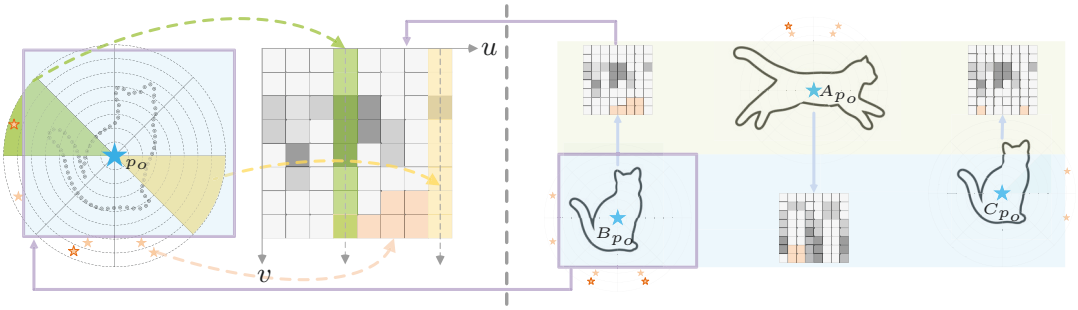

FIGURE 5.2: Illustration of **shape-position descriptor.**

and background stuff, respectively. Finally, we give an analysis on the limitation of current temporal contrastive learning and devise a task-oriented sampling strategy to tackle encountered issues.

Shape-Position Descriptor. Inspired by shape context [287], a shape-position descriptor is constructed to represent the spatial distribution and shape of target objects. First, it describes shape cues by encoding the relative geometric relationships of points in object contours relative to the object center. As shown in Fig. 5.2, given the contour $G \in \{0,1\}^{H \times W}$ of a target object which can be easily derived from masks, a set P with M anchor points (*i.e.*, ✌) are evenly sampled:

$$\mathcal{P} = \{p_m = (x, y) \,|\, G(x, y) = 1,\, 1 \leq m \leq M\}. \tag{5.3}$$

Above anchor points are transformed into polar coordinates with the central point p_o of targets (*i.e.*, ★) as the reference point. The polar coordinate is a histogram divided into a grid of $u \times v$ bins with u angle divisions and v radius divisions. Next we calculate the number of anchor points falling within each bin:

$$\boldsymbol{H}_{i,j} = \sum_{m=1}^{M} \left\{ \begin{array}{ll} \frac{}{\sqrt{d_{\text{model}}}} & \text{if} \quad |\theta_m - \hat{\theta}_i| \leq \frac{\Delta\theta}{2} \quad \text{and} \quad |r_m - \hat{r}_j| \leq \frac{\Delta r}{2} \\ 0 & \text{otherwise} \end{array} \right\}, \tag{5.4}$$

where $\Delta\theta$, Δr, and $(\hat{\theta}_i, \hat{r}_j)$ are the angle span, ra dius span, and center point of each bin, (θ_m, r_m) is the polar coordinate of anchor point p_m, d_{model} is the embedding dimension of model. As such, $\boldsymbol{H}$ expresses the spatial configuration of contour G relative to center point (*i.e.*, p_o) in a compact and robust way. As depicted in Fig. 5.2, instances with different shapes (*i.e.*, target A and B) present varying distributions of $\boldsymbol{H}$ which demonstrates the

capability to encode the shape cues of target objects. Moreover, we equip $\boldsymbol{H}$ with the ability to account for the relative spatial location of target objects by setting $\boldsymbol{H}_{i,j} = -1/\sqrt{d_{\text{model}}}$ if the center point of a bin (*i.e.*, ☆) falls outside of masks. Therefore, instances with similar shapes but different locations (*i.e.*, target B and C) would yield similar distribution of positive values, but distinct distribution of negative values, effectively evolving above shape descriptor into a **shape-position** descriptor.

Shape- and Position-Aware (SPA) Query Matching. Given the above analysis, a set of shape-position descriptors $\{\boldsymbol{H}_k\}_{k=1}^{K}$ could be derived from each object k within the mask. We then aim to facilitate the awareness of shape-position cues for object association between frames, by integrating such descriptors into the query matching process. To achieve this, we draw inspiration from the absolute position encoding (APE) which is widely adopted in Transformer [288]. Specifically, during mask decoding, N query embeddings $\{\boldsymbol{q}_n\}_{n=1}^{N}$ is interacting with the backbone feature $\boldsymbol{F}$ to retrieve object-centric feature in each decoder layer by:

$$\boldsymbol{q}^l = \texttt{CrossAttn}(\boldsymbol{q}^{l-1}, \boldsymbol{F}), \quad \boldsymbol{q}^l = \texttt{SelfAttn}(\boldsymbol{q}^l, \boldsymbol{q}^l) \tag{5.5}$$

Where l is the layer index. Typically, a Hungarian Matching matrix $\mathbb{1}^l \in \{0,1\}^{N\times K}$ between N predictions generated from query embeddings and K ground truth objects can be derived from each decoding layer. Following the principle of APE, where the position encodings $\mathbf{P}$ is integrated into $\boldsymbol{q}$: $\boldsymbol{q} \leftarrow \boldsymbol{q} + \mathbf{P}$, we assign $\{\boldsymbol{H}_k\}_{k=1}^{K}$ to K elements in $\boldsymbol{q}$ that corresponds to the object described in ground truth by referring to $\mathbb{1}^{l-1}$ produced from prior decoding layer: $\boldsymbol{q}^l \leftarrow \boldsymbol{q}^l + \mathbb{1}^{l-1} \cdot \boldsymbol{H}$ before conducting `SelfAttn`. Note the K elements in $\{\boldsymbol{H}_k\}_{k=1}^{K}$ are flattened and bilinearly interpolated to size d_{model}, and then stacked together to get $\boldsymbol{H} \in \mathbb{R}^{K\times d_{\text{model}}}$. In this way, the query embeddings can **i)** well attend to and discriminate corresponding objects by injecting the descriptors into `SelfAttn`, and **ii)** be aware to shape-position cues after mask decoding (*i.e.*, $\hat{\boldsymbol{q}}$ in Eq. 5.2). To further reinforce the consideration to shape and position of targets in $\hat{\boldsymbol{q}}$, we compile $\boldsymbol{H}$ into the affinity-based query matching between two adjacent frames:

$$\mathcal{S}_{ij} = \texttt{cosine}(\hat{\boldsymbol{q}}_i^t + \boldsymbol{H}_i^t, \hat{\boldsymbol{q}}_j^{t+1} + \boldsymbol{H}_j^{t+1}). \tag{5.6}$$

As such, each query embedding is seamlessly incorporated with the unique attributes of corresponding objects, thereby endowing them with a heightened sensitivity to specific targets when matching with other frames afterward. The related algorithm is shown below:

Algorithm 1 Pseudo-code of Shape- and Position-Aware Query Matching in a PyTorch-like style.

```
"""
inter_preds: intermediate mask predictions of the
    mask decoding process.
output_preds: output mask predictions after the
    mask decoding process.
SPD: construction of shape-position descriptor
    from mask predictions.
feats: intermediate pixel features from last
    transformer encoder layer.
o: object-centric query embedding after mask
    decoding process.
o_hat: shape- and position-aware query embedding.
cur_rep: object-centric representation of current
    frame.
ref_rep: object-centric representation of previous
     frame.
HM_assignment: hungarian matching algorithm.
"""

# Integrating Shape-Position Descriptor to enable
# shape- and position-aware mask decoding.
def transformer decoder layer(feats, inter_preds):

  #k x H x W
  spd = SPD(inter_preds)
  #D x H x W
  k = nn.Linear(feats)
  #D x H x W
  v = nn.Linear(feats)

  #compute attention map (Eq. 5)
  #N x H x W
  A = torch.matmul(q, k.transpose())

  #integrate descriptors (Eq. 6)
  #N x H x W
  A_hat = A + spd

  #N x H x W
  A_hat = torch.nn.functional.softmax(A_hat)
  #N x D
  output_q = torch.matmul(A_hat, v)

  return output_q
```

```
# Reformulate query update to establish a
# shape- and position-aware object association
def query_update(output_preds, o):

   #k x D
   spd = SPD(output_preds)

   #integrate descriptors (Eq. 7)
   #N x D
   o_hat = o + spd

   return updated_o

# Perform query matching process that harness
# appearance, shape, and position
def query_match(cur_rep, ref_rep):

   #generate shape- and position-aware object-
       centric representation
   new_cur_rep = query_update(cur_rep)
   new_ref_rep = query_update(ref_rep)

   #calculate affinity matrix between
   #current frame and last frame
   cos_sim = torch.matmul(new_cur_rep, new_ref_rep.
       transpose())
   C = 1 - cos_sim

   #apply hungarian matching on affinity
   #matrix and return matching results
   indices = HM_assignment(C)

   return indices
```

Task-Oriented Query Initialization & Object Association. To orient the model towards specific tasks, existing work usually employs dedicated queries (*i.e.*, *stuff*/*thing* query) for semantic/instance segmentation [155, 289], and process them parallel by modifying the model into a two-path architecture. In contrast, GvSeg smartly addresses this challenge by dynamically adjusting the involvement of three key constitutes, *i.e.*, **appearance**, **shape**, and **position** within the query initialization and object association according to task requirements.

• **EVS** underscores the utilization of given hints to guide the segmentation of subsequent frames. To flexibly unleash the potential of different kinds of hints under the *track by query matching* paradigm, we propose to initialize the query embeddings from backbone features sampled within hinted regions. Specifically, for the point-guided task which provides a single point $p_k = (x, y)$ to indicate the target object, the backbone feature at corresponding location can be sampled by:

$$\boldsymbol{f}_k = \texttt{sample}(\boldsymbol{F},\ p_k), \tag{5.7}$$

where the implementation of `sample` follows PointRent [290]. Then, the query embedding is

initialized with f_k: $\bar{\boldsymbol{q}}_k = \texttt{FFN}(\boldsymbol{f}_k)$ to fulfill the guidance ability of given exemplars where FFN is a feed-forward network. For the mask and box guided tasks, we sample multiple f_k and average them to get the feature that comprehensively describe target objects. Finally, SPA query matching is applied to enhance instance discrimination during the object association between frames.

• **VIS** emphasizes the tracking of instances which usually exhibits unique attributes for discrimination. To encode these instance-specific properties (*e.g.*, location, appearance) into query embeddings, we follow [32] to initialize $\boldsymbol{q} \in \mathbb{R}^{N \times D}$ from the backbone features. Concretely, we partition the backbone features into $S \times S$ grids and flatten them, resulting in $\{\boldsymbol{F}_i\}_{i=1}^{S \times S}$. We then randomly select N elements from this set for the initialization of queries and obtain $\{\bar{\boldsymbol{q}}_i\}_{i=1}^{N}$:

$$[\bar{\boldsymbol{q}}_0; \cdots; \bar{\boldsymbol{q}}_N] = \texttt{FFN}(\boldsymbol{F}). \tag{5.8}$$

As such, queries could involve appearance and location cues for diverse instances present in the frame. Similarly to EVS, we apply SPA query matching for object association to enable more precise instance discrimination across the entire video.

• **VSS** prioritizes semantic understanding of each class. Therefore, to enhance the thorough grasp of semantics, we continuously collect the query embeddings corresponding to each semantic class during training. More precisely, given N queries $\boldsymbol{q} \in \mathbb{R}^{N \times D}$, we gather K entities from them based on the bipartite matching results $\mathbb{1} \in \{0, 1\}^{K \times N}$ between predictions generated from $\boldsymbol{q}$ and ground truth:

$$\bar{\boldsymbol{q}} = \mathbb{1} \odot \boldsymbol{q} \in \mathbb{R}^{K \times D}. \tag{5.9}$$

Here $\bar{\boldsymbol{q}}$ encodes the semantic-specific properties for each class, and we momentously update it in each training step to approximate the global representation of semantic classes over the entire dataset. During inference, we initialize object queries for each frame from $\bar{\boldsymbol{q}}$. Note we do not apply SPA query matching for VSS, as shape and location cues would harm semantic-level tracking.

• **VPS** integrates both instance-discrimination for foreground *thing* classes and semantic interpretation for background *stuff* categories. We thus combine the query initialization and association strategies used in VIS and VSS, to facilitate the effective recognition and tracking for *thing* and *stuff* classes, respectively.The related algorithm is shown below:

Algorithm 2 Task-Smart Query Initialization & Object Association in a PyTorch-like style.

```
"""
feats: backbone features
FP: feats processing (partition and flatten)
FFN: feedforward network
global_memory: global representation of the entire
     dataset
"""

# VIS Query-Initialization
def initialize_queries_vis(feats):

   #process backbone features
   grid_feats = FP(feats)

   #initialize queries
   init_queries = FFN(grid_feats)

   return init_queries

# VSS Query-Initialization
def initialize_queries_vss(feats):

   #initialize queries
   init_queries = FFN(global_memory)

   return init_queries
```

Task-Oriented Temporal Contrastive Learning. The performance of current *track by query matching-based* solutions depends significantly on the temporal contrastive learning (TCL) between frames. Given a key frame, prior methods [118, 158, 160] typically select reference frames from the temporal neighborhood, while ignoring all other frames. This leads to limited positive/negative samples for effective contrastive learning which relies on a substantial quantity of samples to achieve optimal performance. To maximize the usage of these discarded samples, we devise a smart sampling strategy that caters to individual tasks and addresses the challenge of accurately distinguishing the positive ones from them. Specifically, for tasks leaning towards instance discrimination (*i.e.*, VIS, EVS and *thing* in VPS), it is essential to note that not all identical instances in the same video are suitable as positive samples. This is due to the strong variations in shape and spatial location among instances, which can disrupt the local consistency between the same instance at nearby frames that usually manifest similar shape and position. To tackle this, in contrast to existing work arbitrarily discards samples in distant frames, we sample examples across the whole video by measuring the shape and location similarity. The variation of shape-position descriptors (*i.e.*, ΔH) belonging to the same instance but at

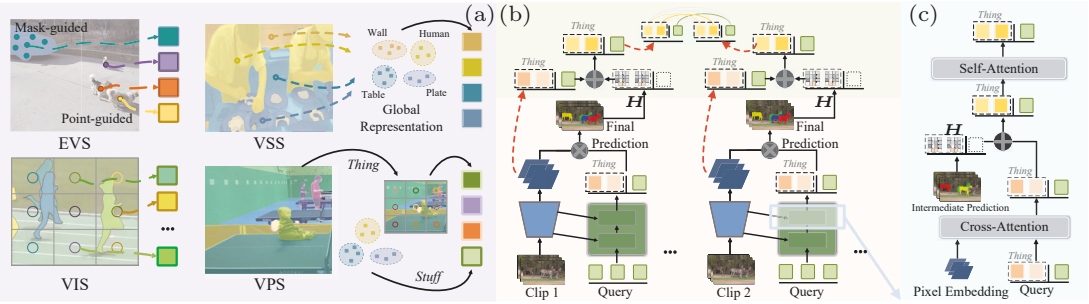

FIGURE 5.3: (a) Task-oriented queries initialization. (b) Task-oriented object association tailored *thing* and *stuff* objects. (c) Shape- and position-aware query matching.

frame V^t and V^{t+n} is computed via:

$$\Delta H = \frac{\|\boldsymbol{H}^{t+n} - \boldsymbol{H}^{t}\|_2}{\|\boldsymbol{H}^{t}\|_2}. \tag{5.10}$$

We set a threshold $\tau = 0.2$ and consider the query embedding associated with $\boldsymbol{H}^{t+n}$ as a positive example if ΔH is smaller than τ; otherwise, it is deemed negative. As such, we involve distant frames into the reference set which enriches the diversity of samples and bolsters the robustness of TCL. On the other hand, for VSS and background *stuff* classes in VPS, samples are relaxed to select from the whole training set, as larger mount of entities with diverse appearance, shape, and location will improve the grasp of semantics. To implement this, we maintain a first-in-first-out queue $\mathcal{Q}$ that contains $N_{\mathcal{Q}}$ queries for each pre-defined semantic class. Elements in $\mathcal{Q}$ will engage in TCL and be updated with new samples at each training step. We set $N_{\mathcal{Q}}$ to a relatively small number (*e.g.*, 100), which incurs negotiable cost in training time but considerable improvement in performance. The related algorithm is shown below:

Algorithm 3 Task-Smart Temporal Contrastive Learning in a PyTorch-like style.

```
"""
q_t: query embeddings for the current frame t
H_t: shape-position descriptor associated with q_t
q_ref: query embeddings for the reference frame
    t_ref
H_ref: shape-position descriptor associated with
    q_ref
queue: a set of queue embeddings selected from the
     whole training set
"""

#computation of Delta_H
def com delta H(H_t, H_ref):

  Delta_H = torch.norm(H_t-H_ref,p=2) /
       torch.norm(H_t,p=2)

  return Delta_H

#task-smart temporal contrastive learning (TCL)
#for vis

#hyperparameter
tau = 0.2

def TCL vis(q_t, q_ref, H_t, H_ref):

  Delta_H = com delta H(H_t, H_ref)

  if Delta_H < tau
      #compute contrastive loss
    loss = TCL loss(q_t, q_ref)

  return loss

#task-smart temporal contrastive learning (TCL)
#for vss

def TCL vss(q_t, queue):

  #compute contrastive loss
  loss = TCL loss(q_t, queue)

  return loss
```

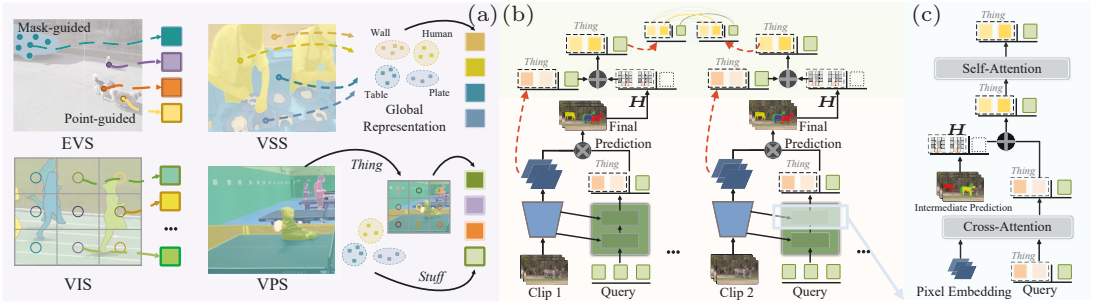

FIGURE 5.4: Illustration of **task-oriented temporal contrastive learning**. Prior work considers solely *instance* objects, and samples are restricted within neighbor frames. In UVSEG, *instance* & *thing* samples are collected from the whole video according to shape and location similarity, while *semantic* & *stuff* samples are gathered from the entire training set to capture diver shapes and appearances of each semantic class.

5.3 Experiment

5.3.1 Experimental Setup

Network Configuration. GVSEG is a semi-online generalist video segmentation framework built upon the *tracking by query matching* paradigm [117]. It comprises an *image-level segmenter* to extract frame-level queries, and an object associator to match query embeddings across frames. The *image-level segmenter* is implemented as Mask2Former [39] with both ResNet-50 [10] and Swin-L [229] as the backbone. Given the most recent work typically adopts clip-level inputs for richer temporal cues [107, 131, 160], in alignment with this trend, GVSEG takes a clip containing three frames as input each time. The size of points set $\mathcal{P}$ derived from object contour is fixed to 200 to make the shape-position descriptor effectively characterize objects of varying scales. We employ $u = 36$ angle divisions and $v = 12$ radius divisions to capture point distribution in a finer granularity. GvSeg is implemented on top of detectorn2 [291]. During training, for YouTube-VIS/VOS, the input frames are randomly cropped to ensure that the longer side is at most 768p/1024p for ResNet/Swin backbones, respectively. The shorter side is resized to at least 240p/360p and at most 480p/600p for ResNet/Swin. For OVIS/VSPW/VIPSeg/KITTI/BURST, we resize the

input frame so that the shorter side is at least 480p and at most 800p and the longer side is at most 1333p.

Training. Following the standard protocols [43, 100, 126, 131, 160] in video segmentation, the maximum training iteration is set to 10K for OVIS/VSPW/VIPSeg/KITTI and 15K for YouTube-VOS$_{18}$/YouTube-VIS$_{21}$ with a mini-batch size of 16. The AdamW optimizer with initial learning rate 0.001 is adopted. The learning rate is scheduled following a step policy, decayed by a factor of 10 at 7K/11K for 10K/15K total training steps, respectively. Following existing solutions [107, 118, 119, 130], we generate pseudo videos from MS COCO [292] as training samples for YouTube-VOS$_{18}$/YouTube-VIS$_{21}$ while no additional data is used for other benchmarks. We use standard data augmentations, *i.e.*, flipping, random scaling and cropping. The *frame segmenter* is initialized with weights pre-trained on MS COCO.

Testing. The evaluation process follows existing work [87, 100, 160, 293] and adopts no test-time augmentation to ensure a fair comparison. For YouTube-VOS$_{18}$/YouTube-VIS$_{21}$, videos are resized to 360p/480p for ResNet/Swin backbones. For OVIS/VSPW/VIPSeg/KITTI/BURST, videos are tested at a resolution of 720p.

5.3.2 Results Comparison

5.3.2.1 Results for Video Panoptic Segmentation

Dataset. VIPSeg [1] provides 2,806/323 videos in `train`/`test` splits which covers 232 real-world scenarios and 58/66 thing/stuff classes. KITTI-STEP [7] is an urban street-view dataset with 12/9 videos for `train`/`val`. It includes 19 semantic classes, with two of them (*pedestrians* and *cars*) having tracking IDs.

Evaluation Metric. Following conventions [1, 7, 126], we adopt VPQ and STQ as metrics. VPQ computes the average panoptic quality from tube IoU across a span of several frames. For VIPSeg [1], we further report the VPQ scores for *thing* and *stuff* classes (*i.e.*, VPQ$^{\text{Th}}$ and VPQ$^{\text{St}}$). For KITTI-VPS [7], we divide STQ into segmentation quality (SQ) and association quality (AQ) which evaluate the pixel-level tracking and segmentation performance in a video clip.

Performance. As illustrated by Table 5.1, GvSeg achieves dominant results on VIPSeg [1], presenting an improvement up to **6.1%/5.6%** in terms of VPQ/STQ over the SOTA [160]

TABLE 5.1: Quantitative results for VPS on VIPSeg [1] and KITTI-STEP [7], and VSS on VSPW [3].

Method	Backbone	General Solution	VIPSeg `val`				KITTI-STEP `val`				VSPW `val`		
			VPQ	VPQ^{Th}	VPQ^{St}	STQ	VPQ	STQ	AQ	SQ	mIoU	mVC_8	mVC_{16}
VPSNet [150]	R-50	✗	14.0	14.0	14.2	20.8	0.43	0.56	0.52	0.61	-	-	-
Mask-Prop [150]	R-50	✗	-	-	-	-	-	0.67	0.63	0.71	-	-	-
MotionLab [7]	R-50	✗	-	-	-	-	0.40	0.58	0.51	0.67	-	-	-
SiamTrack [151]	R-50	✗	17.2	17.3	17.3	21.1	-	-	-	-	-	-	-
TCB [3]	R-101	✗	-	-	-	-	-	-	-	-	37.5	86.9	82.1
DVIS [161]	R-50	✗	43.2	43.6	42.8	42.8	-	-	-	-	-	-	-
Mask2Former [39]	R-50	✓	-	-	-	-	-	-	-	-	38.4	87.5	82.5
TubeFormer [126]	R-50	✓	26.9	-	-	38.6	0.51	0.70	0.64	0.76	-	-	-
Video K-Net [158]	R-50	✓	26.1	-	-	31.5	0.46	0.71	0.70	0.71	37.9	87.0	82.1
TarVIS [87]	R-50	✓	33.5	39.2	28.5	43.1	-	0.70	0.70	0.69	-	-	-
DEVA [294]	R-50	✓	38.3	-	-	41.5	-	-	-	-	-	-	-
Tube-Link [160]	R-50	✓	39.2	-	-	39.5	0.51	0.68	0.67	0.69	43.4	89.2	85.4
GvSeg	R-50	✓	**45.3**	**45.7**	**43.5**	**46.1**	**0.53**	**0.72**	**0.71**	**0.73**	**45.1**	**90.9**	**87.0**
CFFM [147]	MiT-B5	✗	-	-	-	-	-	-	-	-	49.3	90.8	87.1
MRCFA [149]	MiT-B2	✗	-	-	-	-	-	-	-	-	49.9	90.9	87.4
DVIS [161]	Swin-L	✗	57.6	59.9	55.5	55.3	-	-	-	-	-	-	-
Video K-Net [158]	Swin-B	✓	-	-	-	-	-	-	-	-	57.2	90.1	87.8
$TarVIS^{\dagger}$ [87]	Swin-L	✓	48.0	58.2	39.0	52.9	-	-	-	-	-	-	-
DEVA [294]	Swin-L	✓	52.2	-	-	52.2	-	-	-	-	-	-	-
Tube-Link [160]	Swin-B	✓	50.4	-	-	49.4	0.56	0.72	0.69	0.74	62.3	91.4	89.3
GvSeg	Swin-B	✓	**56.4**	**58.4**	**53.7**	**53.5**	**0.58**	**0.75**	**0.75**	**0.74**	**63.6**	**92.1**	**89.7**
GvSeg	Swin-L	✓	**58.5**	**60.2**	**56.7**	**56.1**	-	-	-	-	**65.8**	**94.2**	**92.3**

with ResNet-50 as backbone. This reinforces our belief that accommodating task-oriented property into general video segmentation is imperative. Such an assertion gets further support on KITTI-STEP [7] that GVSEG outperforms all existing solutions by significant margins in STQ and AQ, which focus more on the coherent association of identical objects.

5.3.2.2 Results for Video Semantic Segmentation

Dataset. VSPW [3] has $2,806/343$ in-the-wild videos with $198,224/24,502$ frames for `train`/`val`, and provides pixel-level annotations for 124 semantic categories.

TABLE 5.2: Quantitative results for VIS on OVIS [8] and YouTube-VIS$_{21}$ [2].

Method	Backbone	General Solution	Occluded-VIS `val`					Youtube-VIS$_{21}$ `val`				
			AP	AP$_{50}$	AP$_{75}$	AR$_1$	AR$_{10}$	AP	AP$_{50}$	AP$_{75}$	AR$_1$	AR$_{10}$
SipMask [108]	R-50	✗	10.2	24.7	7.8	7.9	15.8	31.7	52.5	34.0	30.8	37.8
InsPro [119]	R-50	✗	-	-	-	-	-	37.6	58.7	0.9	32.7	41.4
SeqFormer [130]	R-50	✗	-	-	-	-	-	40.5	62.4	43.7	36.1	48.1
VITA [131]	R-50	✗	19.6	41.2	17.4	11.7	26.0	45.7	67.4	49.5	40.9	53.6
MinVIS [117]	R-50	✗	25.0	45.5	24.0	13.9	29.7	44.2	66.0	48.1	39.2	51.7
IDOL [118]	R-50	✗	30.2	51.3	30.0	15.0	37.5	43.9	68.0	49.6	38.0	50.9
MDQE [122]	R-50	✗	33.0	57.4	32.2	15.4	38.4	44.5	67.1	48.7	37.9	49.8
DVIS [161]	R-50	✗	34.1	59.8	32.3	15.9	41.1	-	-	-	-	-
GenVIS [107]	R-50	✗	34.5	59.4	35.0	16.6	38.3	47.1	67.5	51.5	41.6	54.7
TCOVIS [295]	R-50	✗	35.3	60.7	36.6	15.7	39.5	49.5	71.2	53.8	41.3	55.9
CTVIS [296]	R-50	✗	35.5	60.8	34.9	16.1	41.9	50.1	73.7	54.7	41.8	59.5
TubeFormer [126]	R-50	✓	-	-	-	-	-	41.2	60.4	44.7	40.4	54.0
CAROQ [159]	R-50	✓	25.8	47.9	25.4	14.2	33.9	43.3	64.9	47.1	39.3	52.7
TarVIS [87]	R-50	✓	31.1	52.5	30.4	15.9	39.9	48.3	69.6	53.2	40.5	55.9
Tube-Link [160]	R-50	✓	29.5	51.5	30.2	15.5	34.5	47.9	70.0	50.2	42.3	55.2
GvSeg	R-50	✓	**36.9**	**60.6**	**38.9**	**17.1**	**41.0**	**50.5**	**73.0**	**54.1**	**43.7**	**57.8**
GenVIS [107]	Swin-L	✗	45.4	69.2	47.8	18.9	49.0	59.6	80.9	65.8	48.7	65.0
TCOVIS [295]	Swin-L	✗	46.7	70.9	49.5	19.1	50.8	61.3	82.9	68.0	48.6	65.1
CTVIS [296]	Swin-L	✗	46.9	71.5	47.5	19.1	52.1	61.2	84.0	68.8	48.0	65.8
CAROQ [159]	Swin-L	✓	-	-	-	-	-	54.5	75.4	60.5	45.5	61.4
TarVIS [87]	Swin-L	✓	43.2	67.8	44.6	18.0	50.4	60.2	81.4	67.6	47.6	64.8
Tube-Link [160]	Swin-L	✓	-	-	-	-	-	58.4	79.4	64.3	47.5	63.6
GvSeg	Swin-L	✓	**50.8**	**75.8**	**53.0**	**20.1**	**55.7**	**61.4**	**83.4**	**70.3**	**48.0**	**66.6**

Evaluation Metric. Following the standard evaluation protocol [3, 160], we adopt the mean Intersection-over-Union (mIoU), and mean video consistency (mVC) which evaluates the category consistency among a video clip containing 8/16 frames (*i.e.*, mVC$_8$ and mVC$_{16}$) as metrics.

Performance. As shown in Table 5.1, based on ResNet-50, GVSEG outperforms all competitors and achieves **45.1%** mIoU. In particular, the **90.9%**/**87.0%** scores in terms of mVC$_8$/mVC$_{16}$ are comparable to MRCFA [149] which utilizes Swin-B as the backbone and yields much higher mIoU. This suggests that, benefited by task-oriented temporal

contrast learning, GvSeg can produce more consistent prediction across frames. When integrated with Swin-B, GvSeg demonstrates **1.3%** gains over Tube-Link [160], confirming the superiority of our approach.

5.3.2.3 Results for Video Instance Segmentation

Dataset. Occluded VIS [8] is specifically designed to tackle the challenging scenario of object occlusions. It consists of 607/140 long videos with up to 292 frames for `train/val` and spans 25 object categories with a high density of instances. YouTube-VIS$_{21}$ [2] comprises 2,985/421 high resolution videos for `train/val`. It extensively covers 40 object classes with 8,171 unique instances.

Evaluation Metric. Following the official setup [2, 8], we report the mean average precision (mAP) by averaging multiple IoU scores with thresholds from 0.5 to 0.95 at step 0.05, and the average recall (AR) given 1/10 segmented instances per video (*i.e.*, AR$_1$, AR$_{10}$). AP$_{50}$ and AP$_{75}$ with IoU thresholds at 0.5 and 0.75 are also employed for further analysis.

Performance. From Table 5.2 we can observe that GvSeg provides a considerable performance gain over existing methods on Occluded-VIS [8]. Notably, it outperforms the prior specalized/general solution SOTA CTVIS [296]/TarVIS [87] by **1.4%/5.8%** in terms of mAP with ResNet-50 as the backbone. When adopting Swin-L, GvSeg showcases far better performance, achieving up to **50.8%** mAP which earns an impressive **3.9%** improvement against CTVIS. Moreover, we report performance on YouTube-VIS$_{21}$ [2]. As seen, GvSeg surpasses the main rival (*i.e.*, TarVIS), by **2.2%/1.2%** with ResNet-50/Swin-L as backbone.

Additional Quantitative Results for VIS. We provide additional results on YouTube-VIS$_{19}$ in Table 5.3. YouTube-VIS$_{19}$ consists of 2,238/343 videos for `train/val`. Following official setting [2, 8], we adopt mean average precision (mAP) and average recall (AR) as evaluation metrics. The training settings remain consistent with those used for YouTube-VIS$_{21}$. Our observations indicate that GvSeg consistently outperforms previous state-of-the-art methods in terms of mAP and AR.

Method	Backbone	Gen. Sol	mAP	AP_{50}	AP_{75}	AR_1	AR_{10}
MaskTrack [2]	R-50	✗	30.3	51.1	32.6	31.0	35.5
SipMask [108]	R-50	✗	33.7	54.1	35.8	35.4	40.1
CrossVIS [110]	R-50	✗	36.3	56.8	38.9	35.6	40.7
InsPro [119]	R-50	✗	37.6	58.7	0.9	32.7	41.4
VISOLO [111]	R-50	✗	38.6	56.3	43.7	35.7	42.5
InstMove [121]	R-50	✗	40.6	67.2	45.1	35.0	48.2
SeqFormer [130]	R-50	✗	47.4	69.8	51.8	45.4	54.8
MinVIS [117]	R-50	✗	47.4	69.0	52.1	45.7	55.7
IDOL [118]	R-50	✗	49.5	74.0	52.9	47.7	58.7
VITA [131]	R-50	✗	49.8	72.6	54.5	49.4	61.0
GenVIS [107]	R-50	✗	50.0	71.5	54.6	49.5	59.7
TCOVIS [295]	R-50	✗	49.5	71.2	53.8	41.3	55.9
CTVIS [296]	R-50	✗	50.1	73.7	54.7	41.8	59.5
Mask2Former [43]	R-50	✓	46.4	68.0	50.0	-	-
CAROQ [159]	R-50	✓	46.7	70.4	50.9	45.7	55.9
TubeFormer [126]	R-50	✓	47.5	68.7	52.1	50.2	59.0
Tube-Link [160]	R-50	✓	52.8	75.4	56.5	49.3	59.9
GvSeg	R-50	✓	**54.9**	**76.6**	**60.1**	**50.6**	**63.0**

TABLE 5.3: **Quantitative results** on YouTube-VIS_{19} [2] `val`.

5.3.2.4 Results for Exemplar-guided Video Segmentation

Dataset. YouTube-VOS_{18} [4] includes 3, 471/474 videos for `train/val`. The videos are sampled at 30 FPS and annotated per 5 frame with multiple objects. BURST [9] contains 500/993/1, 421 videos for `train/val/test`. It provides mask/point/bounding box as exemplars and averages over 1000 frames per video.

Evaluation Metric. For YouTube-VOS_{18}, we report region similarity ($\mathcal{J}$) and contour accuracy ($\mathcal{F}$) at *seen* and *unseen* classes. For BURST, we assess higher order tracking accuracy [299] on common (H_{com}) and uncommon (H_{unc}) classes.

TABLE 5.4: Quantitative results for EVS on YouTube-VOS$_{18}$ [4], and BURST [9].

Method	Backbone	General Solution	YouTube-VOS$_{18}$ `val` (Mask-guide)					BURST `val` (Point-guide)		
			$\mathcal{G}$	$\mathcal{J}_s$	$\mathcal{F}_s$	$\mathcal{J}_u$	$\mathcal{F}_u$	H_{all}	H_{com}	H_{unc}
Box Tracker [297]	R-50	✗	-	-	-	-	-	12.7	31.7	7.9
STCN [100]	R-50	✗	83.0	81.9	86.5	77.9	85.7	24.4	44.0	19.5
XMem [101]	R-50	✗	85.7	84.6	89.3	80.2	88.7	32.3	47.5	28.6
UNINEXT [298]	R-50	✓	77.0	76.8	81.0	70.8	79.4	-	-	-
TarVIS [87]	R-50	✓	79.2	79.7	84.2	72.9	79.9	30.9	43.2	27.8
GvSeg	R-50	✓	**82.5**	**81.9**	**87.0**	**76.4**	**84.7**	**36.9**	**50.6**	**33.7**
UNINEXT [298]	ConvNeXt-L	✓	78.1	79.1	83.5	71.0	78.9	-	-	-
TarVIS [87]	Swin-L	✓	82.1	82.3	86.5	76.1	83.5	37.5	51.7	34.0
GvSeg	Swin-L	✓	**84.8**	**83.1**	**88.3**	**79.5**	**88.2**	**41.8**	**56.5**	**37.3**

Performance. To make fair comparison with existing work which usually tests on BURST without training, we train GvSeg on YouTube-VOS$_{18}$ and randomly adopt mask or point exemplars as the guidance. Then the performance is evaluated with mask exemplar on YouTube-VOS$_{18}$ and point exemplar on BURST. As shown in Table 5.4, GvSeg yields satisfactory performance on YouTube-VOS$_{18}$, *i.e.*, surpassing the general counterpart (*i.e.*, TarVIS [87]) by **3.3%/2.7%** in terms of $\mathcal{G}$ score with ResNet-50/Swin-L as the backbone. We also provide the point-guided segmentation results on BURST. As seen, GvSeg surpasses current solutions by a large margin across all metrics. For instance, When compared with task-specialized approaches (*e.g.*, XMem [101]), our approach still earns **4.6%** improvement. Note existing work has to adopt an additional offline model for mask prediction with given points, while our method natively supports points as the exemplar, contributing to the superiority in both efficiency and effectiveness.

5.3.2.5 Qualitative Results

In Fig. 5.5, we visualize the comparisons of GvSeg against the top-leading methods on four different tasks (*i.e.*, VPS, VIS, VSS, and EVS). As seen, GvSeg gives more precise and consistent predictions in challenging scenarios.

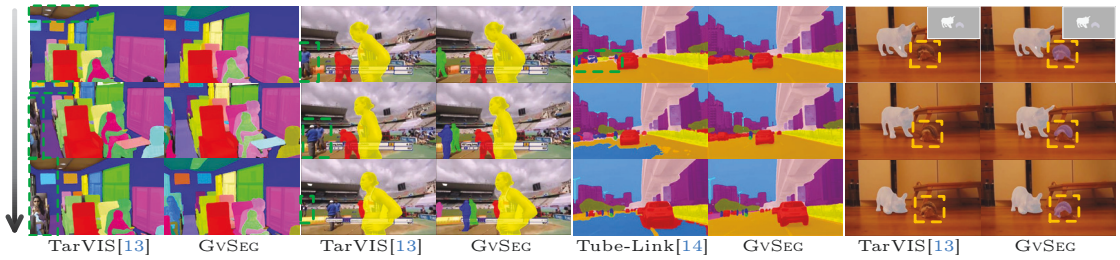

FIGURE 5.5: **Visual comparison results** on VIPSeg-VPS [1], YouTube-VIS$_{21}$ [2], VSPW-VSS [3] and YouTube-VOS$_{18}$ [4].

5.3.2.6 Diagnostic Experiment

For more detailed analysis, we conduct a set of ablative studies on VIPSeg-VPS [1] with ResNet-50 as the backbone.

Key Component Analysis. We investigate the improvements brought by each component of GvSeg in Table 5.5a where 'SPA' indicates 'shape-position aware'. First, it can be observed that SPA query matching brings a considerable improvement over the Baseline, *i.e.*, **1.8%/1.2%** concerning VPQ and STQ. This verifies our modeling of segment targets by disentangling them into appearance, shape, and position. Moreover, the adoption of task-oriented strategies for query initialization, object association, and temporal contrastive learning (TCL) elevates the results to a new level. Finally, we combine all these designs together which results in GvSeg and obtains the optimal performance. This confirms the compatibility of each component and the effectiveness of our whole algorithm.

Matching Threshold & Queue Length. The results with different threshold τ and queue length $N_{\mathcal{Q}}$ utilized in task-oriented TCL are reported in Table 5.5b. Though larger size of samples in the queue contributes to higher scores, we remain $N_{\mathcal{Q}}$ to 100 which gives nearly no impact in training speed and memory usage.

Histogram Size. In Table 5.5c, we investigate the impact of the number of bins within the polar-style histogram for building position-shape descriptor. As seen, there is minor change in performance if $u \times v$ is large enough (*e.g.*, > 200) to capture the fine-grained variation in shape and location.

Task-Oriented Object Association. We probe the impact of integrating distinct cues into object association in Table 5.5d. By comparing *Row* #2 to #1 we can observe that

TABLE 5.5: A set of ablative studies on VIPSeg-VPS [1] `val` with ResNet-50 [10] as the backbone. The adopted settings are marked in red.

(A) Component analysis

Component	VPQ ↑	STQ ↑
Baseline	37.3	38.5
+ SPA query matching	39.1	39.7
+ Task-oriented init.&asso.	41.4	41.9
+ Task-oriented TCL	42.5	43.2
GvSeg	**45.3**	**46.1**

(B) Task-oriented TCL

τ	$N_{\mathcal{Q}}$	VPQ ↑	STQ ↑
0.1	100	44.6	45.1
0.2	**100**	**45.3**	**46.1**
0.2	200	45.4	46.3
0.3	100	44.9	45.6
0.3	200	45.0	45.8

(C) Shape-position descriptor

Angle u	Radius v	VPQ ↑	STQ ↑
12	6	44.4	45.0
24	12	44.9	45.5
36	**12**	**45.3**	**46.1**
36	18	45.2	46.2
48	12	45.3	46.0

(D) Task-oriented query association

#	*Thing*		*Stuff*		VPQ ↑	STQ ↑
	Appear.	Shape & Pos.	Appear.	Shape & Pos.		
1	✓		✓		43.4	44.3
2	✓	✓	✓		**45.3**	**46.1**
3	✓		✓	✓	43.0	44.0
4	✓	✓	✓	✓	44.2	44.6

(E) Task-oriented example sampling

#	*Thing*		*Stuff*		VPQ ↑	STQ ↑
	Frame	Video	Frame	Dataset		
1	✓		✓		41.4	41.9
2	✓			✓	43.7	44.5
3		✓	✓		44.3	45.1
4		✓		✓	**45.3**	**46.1**

considering shape and position can boost the performance for *thing* objects. In stark contrast, the inclusion of these cues causes negative impacts and yields less favorable results for *stuff* objects. This proves the necessity and urgency to cater to the task-oriented property which emphasizes more on *instance discrimination* or *semantic understanding*.

Task-Oriented Example Sampling. To determine the contribution of our devised example sampling strategy utilized in TCL, we examine the performance *thing* and *stuff* categories in Table 5.5e where 'Frame' refers to selecting samples from nearby frames, 'Video' indicates gathering samples across the entire video based on shape-position descriptor for instance discrimination, and 'Dataset' means storing samples in a queue to enhance the comprehension of semantics. As seen, both 'Video' and 'Dataset' level sampling for *thing* and *stuff* classes boost the scores significantly. This verifies our core insight that current sampling strategy in TCL in sub-optimal, and we can improve it by rendering a more holistic modeling on segment targets to select richer and more suitable samples.

5.4 Discussion

Broader Impact. Understanding of visual scenes is a primary goal of computer vision. On the positive side, GVSEG represents generalist video segmentation framework for EVS, VIS, VSS, and VPS which provides insight towards designing a universal model capable of addressing a broader spectrum of vision-related tasks. The disentanglement of task-specific

properties of moving objects can benefit the wide application scenarios in video tasks such as video object detection (VOD) and Multi-Object Tracking and Segmentation (MOTS). On the negative side, it's essential to acknowledge potential operational challenges our method may face in real-world applications. As a proactive step to mitigate any adverse effects on individuals and society, we advise the establishment of a robust security protocol which help ensure the safety and well-being of users and the broader community in case of any unforeseen issues.

5.5 Conclusion

We present GvSeg, the first generalist video segmentation solution that accommodates task-oriented properties into model learning. To achieve this, we first render a holistic investigation on segment targets by disentangling them into three essential constitutes: appearance, shape, and position. Then, by adjusting the involvement of these three key elements in query initialization and object association, we realize customizable prioritization of *instance discrimination* or *semantic understanding* to address different tasks. Moreover, task-oriented temporal contrastive learning is proposed to accumulate a diverse range of informative samples that considers both local consistency and semantic understanding properties for tracking instances and semantic/background classes, respectively. In this manner, GvSeg offers tailored consideration for each individual task and consistently obtains top-leading results in four video segmentation tasks.

Chapter 6

UAHOI: Uncertainty-aware Robust Interaction Learning for HOI Detection

6.1 Introduction

Human-Object Interaction Detection (HOI Detection) aims to localize and recognize HOI triplets in the format of <human, verb, object> from static images [300, 301]. This field stems from detection of objects to include their relationships, prompting a deeper understanding on high-level semantic comprehension. HOI Detection has attracted considerable attention for its great potential in numerous high-level visual understanding tasks, including video question answering [302], video captioning [303, 304], activity recognition [305], and syntia-to-reality translation [137].

Traditional methods [65–68] typically adopt either two-stage or one-stage pipeline, where the former detects instances first and then enumerates human-object paris to identify their interactions, and the latter attempts to do both simultaneously. However, these methods struggle with modeling the complex, long-range dependencies between humans and objects due to the localized nature of convolutional operations—a limitation that transformer-based methods address by capturing intricate interrelations across entire scenes. Recent advancements [69–71, 83] have predominantly embraced the encoder-decoder framework pioneered by detection transformers (DETR) [42], initializing learnable queries randomly, and subsequently decode the object queries into detailed triplets of

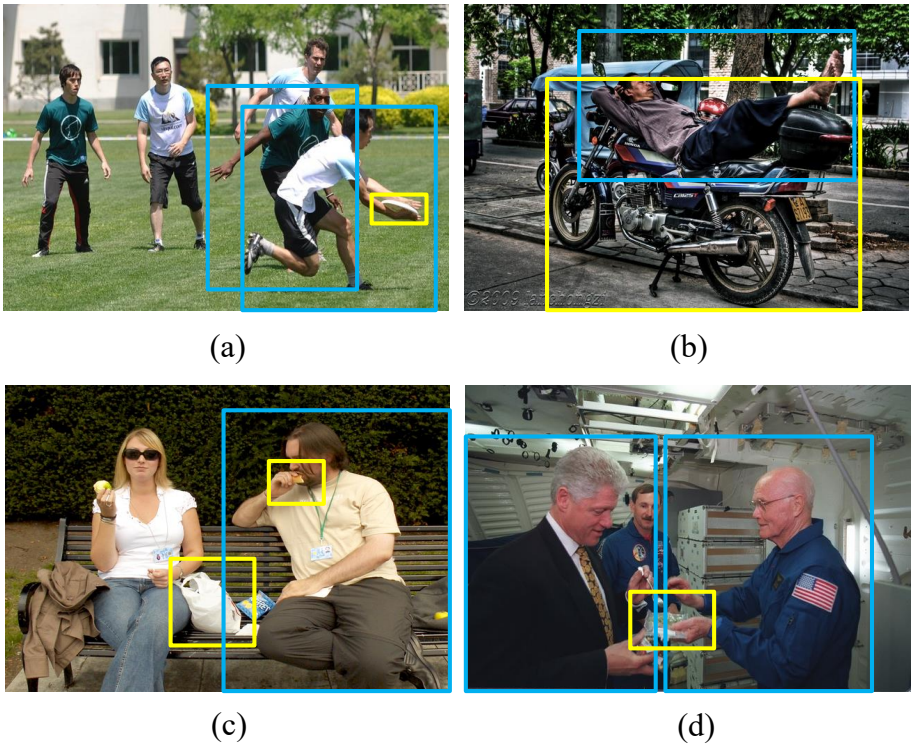

FIGURE 6.1: Common challenges of current HOI Detection methods in complex scene. The human/object bounding boxes are shown in blue/yellow.

human-object interactions. These methods offer enhanced accuracy by capturing global contexts and intricate interrelations, and simplify the architecture by eliminating the need for extra hand-designed components.

Despite the notable advances, several challenges still persist. As shown in Fig 6.1, **firstly**, among the wide range of predicates, some interactions may not be directly manifested through visual signals, often involving subtle movements and non-physical connections. For example, the interaction looking in a certain direction in (c) may easily be overlooked. And a man lying on a mortobike could be easily recognized as a man riding a mortobike (b). **Secondly**, in a multi-human scene, a person who may be imagining or planning to interact with an object without yet taking any action can easily be mistaken for having already performed the action, such as the unselected background individuals in images (a) and (d). **Thirdly**, the same type of interaction can appear very differently in different contexts. For instance, the action of "grabbing" can vary significantly when interacting with different objects, depending on the object's size, shape, weight, and other characteristics. In such

complex scenarios, current models typically assign lower confidence to interactions, which affect the performance.

To address these issues, conventional two-stage methods involves assisstance from extra communication signal [94, 134, 306–308], and language [50, 64], but the models still tend to focus on inaccurate regions. More recent approaches independently process instance detection and interaction classification using two separate decoders, which operate either in parallel [76] or cascaded [69, 73] mode. For example, [73] applied two cascade decoders, one for generating human-object pairs and another for dedicated interaction classification of each pair, which helps the model to determine which regions inside a scene to concentrate on. Taking a step further, Unary [75] separately encodes human and object instances, enhancing the output features with additional transformer layers for more accurate HOI classification. [76] disentangle both encoder and decoder to enhance the learning process for two distinct subtasks: identifying human-object instances and accurately classifying interactions, which necessitates learning representations attentive to varied regions. While the disentanglement of subtasks allows individual modules to concentrate on their specific tasks, thereby boosting overall performance, these methods often require additional heuristic thresholding when deciding which interactions to retain. For simple interactions, it is relatively easy to obtain good interaction predictions with high confidence scores in their predicted categories. However, for more complex interactions, confidence scores may be lower. In such cases, finding a suitable threshold manually to disregard low-confidence predictions could lead to overlooking correct interactions, representing a persistent challenge in HOI detection. Specifically, determining the optimal threshold value is challenging across different categories, and estimiting a value in advance is more complex. For overt interactions like "riding," where a person is physically mounted on an object such as a bicycle or horse, confidence scores are typically high. In contrast, subtle interactions like "reading", characterized by the presence of a book and the direction of a person's gaze, often yield lower overall confidence scores. A high threshold in such cases might cause the model to ignore these interactions.

We propose a novel method UAHOI, Uncertainty-aware Robust Human-Object Interaction Learning that utilizes uncertainty estimation to dynamically adjust the threshold for interaction predictions in the HOI detection task. This approach integrates uncertainty modeling to refine the decision-making process, enabling the model to adjust its confidence thresholds based on the predicted uncertainty associated with each interaction. Specifically,

we utilize the variance in predictions as a measure of uncertainty for both human/object bounding boxes and interaction, which reflects the model's confidence in their outputs. The variance is directly incorporated into our optimization target, enhancing the accuracy of bounding box predictions and ensuring that significant interactions are not overlooked due to artificially low confidence thresholds. Such adaptive handling of complex interactions increases the robustness of the HOI detection model. UAHOI handles complex interactions in an adaptive manner, enhancing the accuracy of the bounding box predictions and preventing important interactions from being overlooked due to artificially low confidence thresholds, thereby increasing the robustness of HOI Detection model. We conducted a comprehensive evaluation on two standard human-object interaction datasets HICO-DET [11] and V-COCO [12], and our experiments demonstrate a significant improvement over the existing state-of-the-art methods. Specifically, UAHOI achieved 34.19 mAP on HICO-DET and 62.6 mAP on V-COCO.

6.2 Methodology

6.2.1 Problem Statement

Since DETR [42], object detection has been investigated as a set prediction problem. DETR employs a transformer encoder-decoder architecture to transform N positional encodings into N predictions, encompassing both object class and bounding box coordinates. Similar to object detection, recent advancements[69–80] have adeptly harnessed the Encoder-Decoder framework, integrating the Transformer architecture to better capture complex dependencies between humans and objects. This integration significantly enhances model precision and deepens understanding of interactions within a scene [70, 72]. In our approach, as is shown in Fig 6.2, we implement an encoder-decoder structure with a shared encoder alongside two parallel decoders: one for instance localization and another for interaction recognition. This design helps to eliminate the issue of redundant predictions. [70]. In detail, the feature $\boldsymbol{f} \in \mathbb{R}^{D \times H \times W}$ is extracted from the input image $\boldsymbol{x}$ via a CNN backbone, where H and W are the size of the input image, and D is the number of channel. Combined with positional embedding p, the feature $\boldsymbol{f}$ containing semantic concepts is flattened to construct a sequence of length $H \times W$ and then fed into the image encoder. We adopt ResNet as our backbone. Each image encoder layer consists of a multi-head self-attention (MHSA) module and a feed-forward network, which refine the feature representation

sequentially. After processing by image encoder, the resulting encoded features, denoted as $\boldsymbol{f}_{en} \in \mathbb{R}^{D' \times H' \times W'}$, are split and fed to the instance localization decoder and interaction recognition decoder. The instance localization decoder identifies and localizes objects within the scene, and the interaction recognition decoder analyzes the interaction between detected objects and humans, aiming to understand their mutual interactions. Object queries $\boldsymbol{Q}_{obj} = \left\{\boldsymbol{q}_i \mid \boldsymbol{q}_i \in \mathbb{R}^d\right\}_{i=1}^{H}$ and Interaction queries $\boldsymbol{Q}_{inter} = \left\{\boldsymbol{q}_i \mid \boldsymbol{q}_i \in \mathbb{R}^d\right\}_{i=1}^{H}$ are initialized randomly, and then learnt via cross-attention layers. H is the number of queries and d is the query dimension. The decoding process for localization and interaction recognition could be formulated by:

$$\boldsymbol{L} = \text{Decoder}_{loc}\left(\boldsymbol{f}_{en}, \boldsymbol{Q}_{obj}\right) \in \mathbb{R}^{H \times 4}, \tag{6.1}$$

$$\boldsymbol{I} = \text{Decoder}_{\text{inter}}\ \left(\boldsymbol{f}_{\text{en}}\ , \boldsymbol{Q}_{\text{inter}}\ \right) \in \mathbb{R}^{H \times C}. \tag{6.2}$$

where $\boldsymbol{Q}_{obj}$ and $\boldsymbol{Q}_{\text{inter}}$ are sets of initialized queries for object detection and interaction recognition, respectively, refined through cross-attention mechanism.

With the learned HOI queries, the HOI pair could be decoded by several MLP branches. Specifically, we adopt three MLP branches designed to output the confidence levels for the human, object, and their interaction, respectively, each employing a softmax function to ensure probabilistic outputs. The addition of these branches allows for a more granular understanding of the HOI dynamics by providing individual confidence scores that reflect the certainty of each element's involvement in the interaction. The output embedding is then decoded into specific HOI instance via several multiple Multi-Layer Perceptron (MLP) layers. In detail, three separate MLP branches are designed to predict the confidence levels for the human, object, and interaction, respectively. Each branch employs a softmax function to generate probabilistic outputs. The human and object branches are denoted by orange color in Figure 6.2. For human branch, two values with confidence are outputed to indicate the likelihoods of foreground and background presence. Regarding object and interaction branches, the output scores including all categories of objects or actions and another one category for the background. UAHOI adopts two Fully Feed-Forward Networks (FFN) layers to predict the bounding boxes of the human and the object. The bounding box consists of four values to represent each coordinate for precise localization within the visual scene.

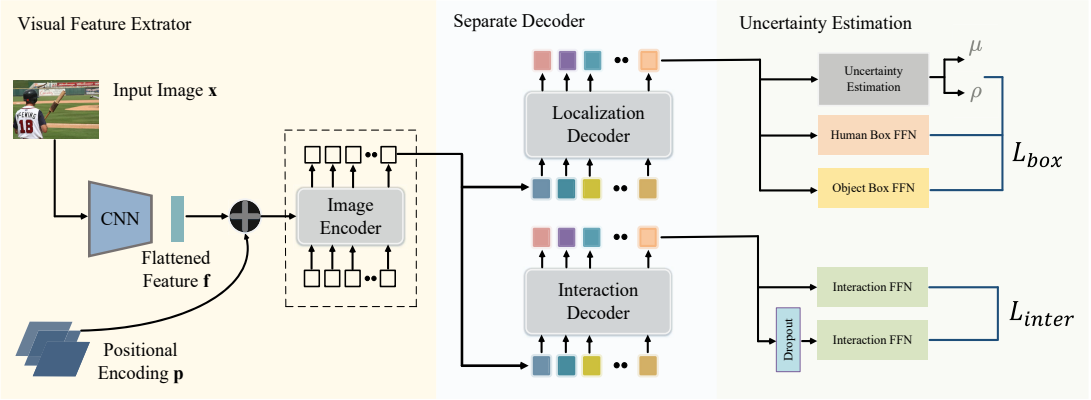

FIGURE 6.2: Overall framework of our UAHOI. UAHOI consists of three components: Visual Feature Extrator, Parallel Decoder and Uncertainty Estimation module. Visual features are firstly extracted by CNN and shared Transformer Encoder. Then, the Localization Decoder and Interaction Decoder run n parallelto extract human/object bounding boxes and interaction class. Lastly, the proposed Uncertainty-aware Instance Localization and Interaction Refinement module are used to perform uncertainty regularization.

6.2.2 Uncertainty-aware Instance Localization

Firstly, when estimating localization uncertainty, we consider the bounding boxes for both humans, denoted by $C_{human} \in (l_h, r_h, t_h, b_h)$ and objects, represented as $C_{object} \in (l_o, r_o, t_o, b_o)$. This representation allows us to explore inherent uncertainty in the prediction of bounding box coordinates, which is particularly useful in complex scenes where occlusion or interaction may obscure part of the subjects. We employ a dedicated network to compute the standard deviations of these distributions, providing a measurable and quantifiable uncertainty which enhances the precision in object boundary detection This methodology not only improves accuracy but also increases the reliability of localizations by effectively capturing and quantifying the inherent uncertainties associated with positional offsets. To accurately delineate the object's boundary, it is essential to account for the four directional offsets of the human/object bounding box. Adopting the framework outlined by [309], we implement an uncertainty estimation network tailored to assess the localization uncertainty

derived from these regressed box offsets (l, r, t, b), defined as Gaussian distributions:

$$l \sim \mathcal{N}(\mu_l, \sigma_l^2), \tag{6.3}$$

$$r \sim \mathcal{N}(\mu_r, \sigma_r^2), \tag{6.4}$$

$$t \sim \mathcal{N}(\mu_t, \sigma_t^2), \tag{6.5}$$

$$b \sim \mathcal{N}(\mu_b, \sigma_b^2). \tag{6.6}$$

Here, μ and σ^2 signify the mean and variance of the offsets, respectively. To determine these parameters accurately, we deploy a neural network featuring a dual-head architecture. One head predicts the mean values while the other calculates the logarithm of the variance as *uncertainty*, naming Var_{box}, ensuring that the variance σ_{box} is always positive:

$$\mu_{\text{box}} = \text{MLP}_{\mu}(\mathbf{f}_{\text{en}}), \tag{6.7}$$

$$\sigma_{\text{box}}^2 = \log(1 + \exp(\text{MLP}_{\sigma}(\mathbf{f}_{\text{en}}))). \tag{6.8}$$

Further, [309] introduce a Negative Power Log-Likelihood Loss, which is reformulated to achieve an uncertainty loss. This uncertainty loss compels the network to output a higher uncertainty value when the coordinate predictions from the regression branch are off-target:

$$L_{box} = - \sum_{c \in \{l,r,t,b\}} IoU \cdot \log P_{\Theta} \left(C \mid \mu_c, \sigma_c^2 \right). \tag{6.9}$$

This equation emphasizes the integration of the Intersection over Union (IoU) as a scaling factor, where IoU measures the overlap between the predicted and actual bounding boxes, enhancing the training focus on precision. The probability density function P_{Θ}, parameterized by network parameters Θ, plays a crucial role in adjusting the model's certainty regarding the predicted localizations, thus pushing the boundaries of accuracy in object detection in highly dynamic and unpredictable scenes.

6.2.3 Uncertainty-aware Interaction Refinement

Secondly, to refine the interaction, we model the uncertainty of the interaction classification via the prediction variance. Typically, models tend to make less accurate predictions in

complex interaction areas. By modeling uncertainty, we are able to quantitatively calculate this uncertainty. Specifically, if there is a significant difference between predictions made with and without dropout, the variance will be high. This reflects the model's uncertainty in predicting interactions. Following previous works [310], we add structured noise to the interaction feature representation via dropout. We denote $\hat{\boldsymbol{y}}^i = f\left(\boldsymbol{x}; \hat{\boldsymbol{\theta}}_i\right)$ and $\boldsymbol{y} = f(\boldsymbol{x}; \boldsymbol{\theta})$ as the interaction representation with/without dropout. Following recent works in the field of uncertainty estimation [176], we predict the interaction variance *uncertainty* Var_{inter} as the KL divergence between the two representation:

$$D_{kl} = \mathbb{E}\left[\boldsymbol{y} \log\left(\frac{\boldsymbol{y}}{\hat{\boldsymbol{y}}^i}\right)\right]. \tag{6.10}$$

Following [176, 311], we regularize the interaction variance by minizing the prediction bias, thus enabling the learning from inaccurate interaction. The objective could be formulated as:

$$L_{inter} = \mathbb{E}\left[\exp\left\{-D_{kl}\right\} L_{ce} + D_{kl}\right]. \tag{6.11}$$

The final loss to be minimized consists of four parts:

$$L_{\text{total}} = L_{loc}^{h} + L_{loc}^{o} + \lambda_o L_{box} + \lambda_a L_{inter}. \tag{6.12}$$

Here, λ_1 and λ_2 are the weights of two uncertainty losses, L_{loc}^{h} and L_{loc}^{o} are computed by box regression loss. In this situation, during the optimization process, the variance of both bounding boxes and the interation will be minimized.

6.3 Experiment

6.3.1 Experimental Setup

Network Architecture. To ensure a fair comparison with existing works [69, 70, 72], we adopt ResNet-50 as our backbone, followed by a six layer transformer encoder as our visual feature extrator. Both the Localization and Interaction Decoder consist of four Transformer decoder layers.

Training. During training, all of the transformer layer weights are initialized with Xavier init [312]. UAHOI is optimized by AdamW [252] and we set the initial learning rate of both encoder and decoder to 10^{-4} and weight decay to 10^{-4}. The weight coeffieicnts λ_o and λ_a are set to 1 and 1. To fairly compare with existing methods, the Backbone, Image Encoder and both Localization and Interaction Decoder are pretrained in MS-COCO and frozen during training. All the augmentation are the same as those in DETR [42]. All experiments are conducted on 8 A40 GPUs with a batch size of 16.

6.3.2 Results Comparison

We comprehensively compare our UAHOI with the recently leading approaches in two representative human-object interaction datasets, HICO-DET [11] and V-COCO [12]. Additionally, we provide some qualitative results in Figure 6.3.

6.3.2.1 Results for HICO-DET

Dataset. We first assess UAHOI on human-object interaction datasets HICO-DET [11]. HICO-DET has $47,776$ images, with $38,118$ for training and $9,658$ designated for testing. There are 600 HOI categories (in total) over 117 interactions and 80 object categories. The interactions are further split into 138 Rare and 462 Non-Rare catorigies. We calculate the mAP scores using two different setups folowing: (i) the Default Setup, where we compute the mAP across all test images; and (ii) the Known Object Setup, where we calculate the Average Precision (AP) for each object separately, only within the subset of images that contain the specified object.

Evaluation Metric. Following the standard evaluation protocoles [47, 70], we adopt mean Average Precision (mAP) as metric.

Comparison with State-of-the-Art Methods We first conduct experiments on HICO-DET [11] with ResNet-50 as the backbone to verify the effectiveness of our proposed methods, and report result in Table 6.1 and Table 6.2. Compared to most transformer-based single-decoder works HOITrans [72] and QPIC [71], Our UAHOI achieves better performance, which validates the effectiveness to adopt multi-decoders for detecting more accurate Human Object Interaction pair. Comapred to RLIP [79] which introduce language as additional cues for more accurate HOI detection, our method attains improvements from 32.84 mAP to 34.19 mAP for full evaluation under default setting. Even when comparing

to the state-of-the-art method GEN-VLKT [77], our UAHOI reaches 34.19/31.54/35.27 mAP on the full/rare/non-rare evaluation under the default setting (The best results are highlighted in bold). Especially, UAHOI significantly promotes mAP from 29.25 to 31.54 for rare evaluation under default setting. These results substantiate our motivation to refine the human/object localization and interaction recognition via uncertainty estimation. For the known objects setting, it could be seen that UAHOI achieves 37.44/34.18/38.65 mAP on the full/rare/non-rare evaluation.

6.3.2.2 Results for V-COCO

Dataset. We next assess UAHOI on a smaller dataset V-COCO [12] which is originates from COCO [292]. V-COCO has $2,533/2,867/4,946$ images for training, validation, and testing respectively. It consists of 80 objects identical to those in HICO-DET and 29 action categories in total.

Evaluation Metric. We also adopt Average Precision (AP) to report performance and compute it under two scenarios to address the challenge of objects missing due to occlusion. We denote these two scenarios with the subscripts S^{S1}_{role} and S^{S2}_{role}. In scenario S^{S1}_{role}, when an object is occluded, we predict empty object boxes to consider the detected pair as a match with the corresponding ground truth. In scenario S^{S2}_{role}, object boxes are automatically considered matched in cases of occlusion, without the need to predict empty boxes.

Comparison with State-of-the-Art Methods We next access UAHOI on V-COCO [12] dataset. Table 6.2 illustrates the results with both Scenario1 and Scenario2. It could be seen that UAHOI outperforms all existing methods without extra knowledge. We outperforms state-of-the-art method GEN-VLKT [77] by a large marin of 3.2 mAP under Scenario2. In addition, Compared to the methods with extra language knowledge, UAHOI is still competitive.

6.3.2.3 Qualitative Results

Additional qualitative results are presented in this section. As shown in Fig 6.3, we leavrage red lines to denote the detected HOI pairs and blue/green boxes to represent human/object. It could be seen that for common interactions such as sitting, riding, lying, reading, etc., UAHOI demonstrates robust performance. Furthermore, when tackling more challenging

TABLE 6.1: Comparison of detection performance on the HICO-DET [11] test set, using ResNet50 backbone. The best performance is emphasized in bold.

		Default Setup			Known Objects Setup		
Method	**Backbone**	Full	Rare	Non-rare	Full	Rare	Non-rare
HO-RCNN [11]	CaffeNet	7.81	5.37	8.54	10.41	8.94	10.85
InteractNet [67]	ResNet-50-FPN	9.94	7.16	10.77	-	-	-
GPNN [47]	ResNet-101	13.11	9.34	14.23	-	-	-
iCAN [313]	ResNet-50	14.84	10.45	16.15	16.26	11.33	17.73
TIN [52]	ResNet-50	17.03	13.42	18.11	19.17	15.51	20.26
Gupta et al [48]	ResNet-152	17.18	12.17	18.68	-	-	-
VSGNet [53]	ResNet-152	19.80	16.05	20.91	-	-	-
DJ-RN [59]	ResNet-50	21.34	18.53	22.18	23.69	20.64	24.60
PPDM [61]	Hourglass-104	21.94	13.97	24.32	24.81	17.09	27.12
VCL [56]	ResNet-50	23.63	17.21	25.55	25.98	19.12	28.03
ATL [314]	ResNet-50	23.81	17.43	27.42	27.38	22.09	28.96
DRG [55]	ResNet-50-FPN	24.53	19.47	26.04	27.98	23.11	29.43
IDN [57]	ResNet-50	24.58	20.33	25.86	27.89	23.64	29.16
HOTR [70]	ResNet-50	25.10	17.34	27.42	-	-	-
FCL [315]	ResNet-50	25.27	20.57	26.67	27.71	22.34	28.93
HOI-Trans [72]	ResNet-101	26.61	19.15	28.84	29.13	20.98	31.57
AS-Net [69]	ResNet-50	28.87	24.25	30.25	31.74	27.07	33.14
SCG [66]	ResNet-50-FPN	29.26	24.61	30.65	32.87	27.89	34.35
QPIC [71]	ResNet-101	29.90	23.92	31.69	32.38	26.06	34.27
MSTR [74]	ResNet-50	31.17	25.31	32.92	34.02	28.82	35.57
CDN [73]	ResNet-101	32.07	27.19	33.53	34.79	29.48	36.38
UPT [75]	ResNet-101-DC5	32.62	28.62	33.81	36.08	31.41	37.47
RLIP [79]	ResNet-50	32.84	26.85	34.63	-	-	-
GEN-VLKT [77]	ResNet-50	33.75	29.25	35.10	36.78	32.75	37.99
UAHOI	ResNet-50	**34.19**	**31.54**	**35.27**	**37.44**	**34.18**	**38.65**

scenes with an increased number of humans, ranging from 2 to 5 such as (h), (i), (j), UAHOI continues to perform effectively.

TABLE 6.2: Comparison of detection performance on the V-COCO [12] test set, using ResNet50 backbone. The best performance is emphasized in bold.

Method	**Backbone**	AP^{S1}_{role}	AP^{S2}_{role}
InteractNet [67]	ResNet-50-FPN	40.0	-
GPNN [47]	ResNet-101	44.0	-
iCAN [313]	ResNet-50	45.3	52.4
TIN [52]	ResNet-50	47.8	54.2
VSGNet [53]	ResNet-152	51.8	57.0
VCL [56]	ResNet-50	48.3	-
DRG [55]	ResNet-50-FPN	51.0	-
IDN [57]	ResNet-50	53.3	60.3
HOTR [70]	ResNet-50	55.2	64.4
FCL [315]	ResNet-50	52.4	-
HOI-Trans [72]	ResNet-101	52.9	-
AS-Net [69]	ResNet-50	53.9	-
SCG [66]	ResNet-50-FPN	54.2	60.9
QPIC [71]	ResNet-101	58.8	61.0
MSTR [74]	ResNet-50	62.0	65.2
CDN [73]	ResNet-101	**63.9**	65.9
UPT [75]	ResNet-101-DC5	61.3	**67.1**
RLIP [79]	ResNet-50	61.9	64.2
GEN-VLKT [77]	ResNet-50	62.4	64.5
UAHOI	ResNet-50	62.6	66.7

6.3.2.4 Diagnostic Experiment

We evaluate the contribution of each component present in our framework. Specifically, we evaluate UAHOI on the task of HICO-DET [11], with Res-Net 50 backbone.

Major Components Analysis. In this section, we conduct experiments for evaluating major components in our UAHOI: Uncertainty-aware Instance Localization and Interaction Refinement. As shown in Table 6.3, our base model, utilizing a traditional handcrafted threshold, achieved 31.75 mAP. We then modeled the uncertainty of the bounding box and incorporated the proposed localization refinement module, which improved our results from 31.75 mAP to 32.52 mAP. Furthermore, to enhance the model accuracy in predicting complex interactions and prevent the model from discarding uncertain interaction categories

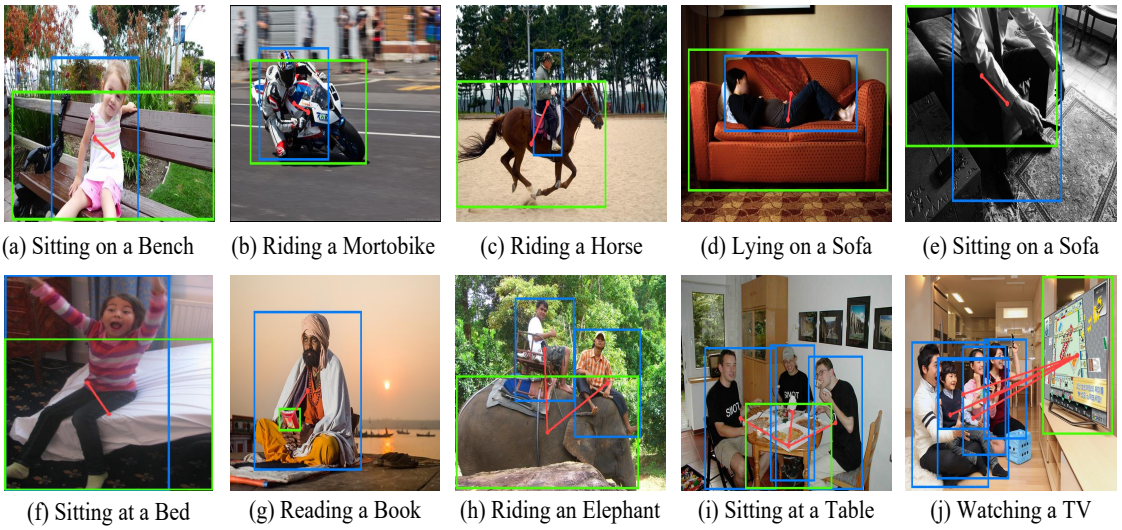

FIGURE 6.3: Visualization results of our UAHOI.

TABLE 6.3: Major component analysis on the HICO-DET test set. The best results are averaged across three runs.

Strategy	Full	Rare	Non-Rare
fixed threshold	31.75	29.42	32.50
+ localization refine	32.52	30.48	33.63
+ interaction refine	33.65	31.27	34.88
+ both	34.19 ± 0.09	31.54 ± 0.17	35.27 ± 0.15

due to a fixed threshold, we applied specific regularization to the prediction variance of the interactions. Using such a dynamic threshold, we optimized interaction uncertainty and achieved higher performance, reaching 33.65 mAP. Finally, by simultaneously employing both localization and interaction uncertainty modules, we elevated the results from 31.75 mAP to 33.65 mAP, achieving optimal performance. This validates the effectiveness of our approach in modeling uncertainty on two levels and integrating uncertainty regularization into our optimization objectives.

Effect of various uncertainty. Neural networks are theoretically capable of providing estimates of both confidence, known as aleatoric uncertainty, and model-based uncertainty, referred to as epistemic uncertainty. Existing literatures [176, 316] employs various strategies to model uncertainty, and we compared our method with two other approaches. The first approach involves an extra classifier. This method adds another classifier to the

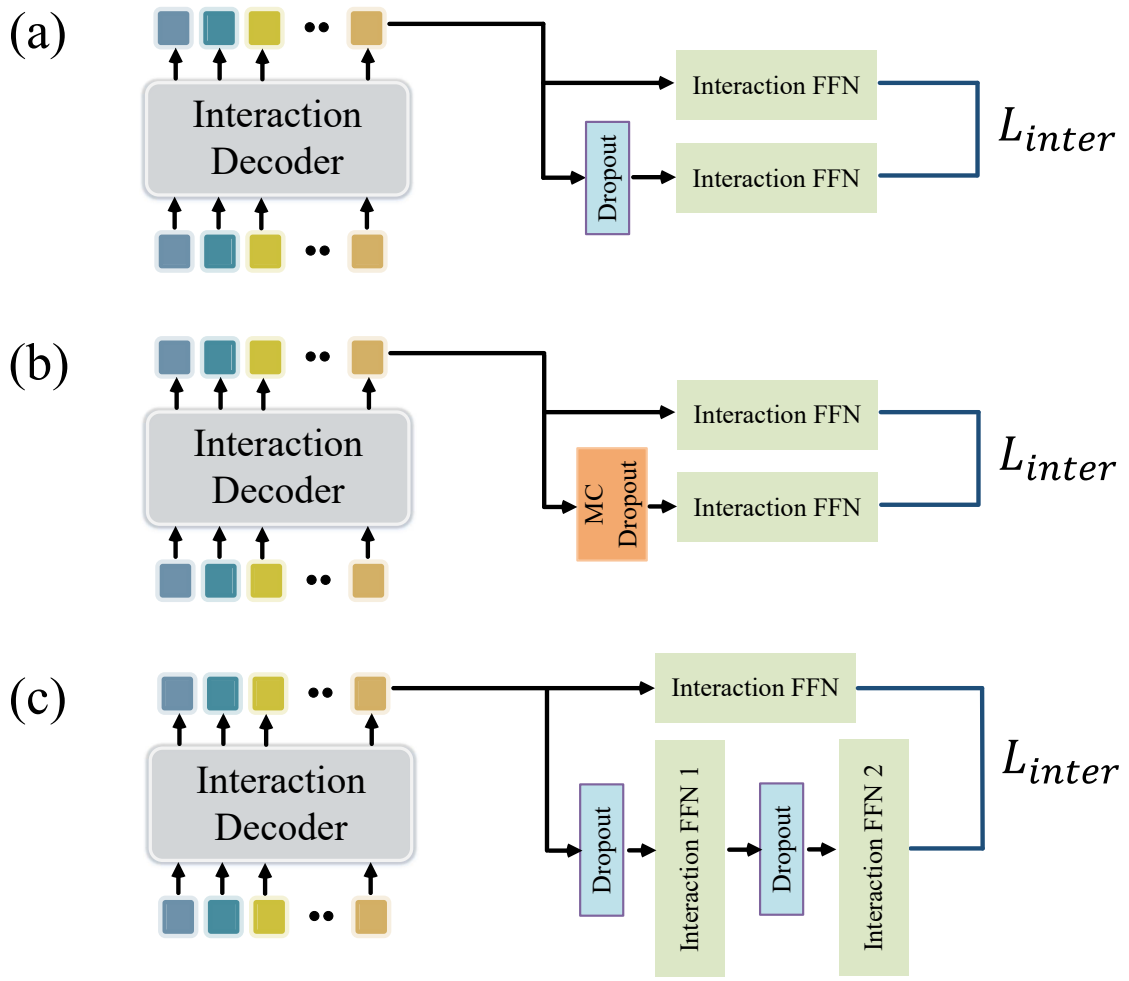

FIGURE 6.4: For a more comprehensive validation of the effects of different uncertainty estimation methods, we further designed our interaction prediction tests by incorporating MC dropout (b), as well as utilizing architectures of Fully Feed-Forward Networks (FFN) with varying depths (c). By adding dropout at different layers, we achieved varying degrees of prediction variance. The results, comparisons, and further analyses are presented in Table 6.4.

existing network architecture to predict interactions, commonly referred to as the auxiliary classification head, while the original classifier is termed the primary classification head. Both classifiers provide predictions, and due to the inherent uncertainty of the model, the outputs from both classifiers are often more uncertain in challenging scenarios. Through regularization of both classifiers' predictions, refinement of interaction predictions is achieved. As shown in the table, this method reached an accuracy of 33.86 mAP. Subsequently, we

TABLE 6.4: The mAP of different uncertainty modeling strategy on the HICO-DET test set.

Strategy	Full	Rare	Non-Rare
base	32.52	30.48	33.63
+ Additional Classifier	33.86	30.79	34.82
+ MC Dropout 0.5	33.15	33.65	34.79
+ MC Dropout 0.7	33.22	33.71	34.70
+ MC Dropout 0.9	32.97	33.51	34.35
+ Dropout	**34.19**	**31.54**	**35.27**

TABLE 6.5: The mAP of different dropout depth on the HICO-DET test set.

Strategy	Full	Rare	Non-Rare
base	32.52	30.48	33.63
with one FFN layer	34.19	31.54	35.27
with two FFN layers	**34.27**	**31.60**	**35.43**

compared this with Monte Carlo Dropout (MC-Dropout). MC-Dropout [316–318] is used as another means to estimate epistemic uncertainty. We implemente different MC-Dropout rates of 0.5, 0.7, and 0.9 instead of the standard dropout. The results indicate that MC-Dropout also enhances performance and is not sensitive to the dropout rate.

Effect of dropout depth. According to the experiments presented in the Table 6.4, our model is not sensitive to changes in the dropout rate. Therefore, in this section, we perform dropout sampling on models with varying depths of Feed-Forward Networks (FFNs), with results shown in the Table 6.5. Our base model uses a single layer of FFN without employing dropout, achieving a result of 32.52 mAP. After implementing dropout and conducting uncertainty refinement, the score increased to 34.19 mAP. Using two layers of FFN and applying dropout to each layer further enhanced the performance to 34.27 mAP.

Parameter sensitivity analysis on loss weights. We conduct sensitivity analysis on the parameters of loss weights to evaluate the sensitivity of UAHOI on HICO-DET test set. As shown in Table 6.6, we select loss weights λ_o and $\lambda_a \in \{0.01, 0.1, 0.5, 1.0, 2.0\}$. When

TABLE 6.6: Parameter sensitivity analysis on the weight of localization uncertainty loss on HICO-DET test set.

λ_o	Full	Rare	Non-Rare	λ_a	Full	Rare	Non-Rare
0.01	32.18	29.45	32.89	0.01	33.65	30.45	34.89
0.1	33.29	30.60	34.06	0.1	33.77	30.86	34.11
0.5	34.09	31.49	35.19	0.5	33.86	31.04	34.55
1.0	**34.19**	**31.54**	**35.27**	1	**34.19**	**31.54**	**35.27**
2.0	32.25	30.22	34.01	2.0	31.95	29.77	33.68

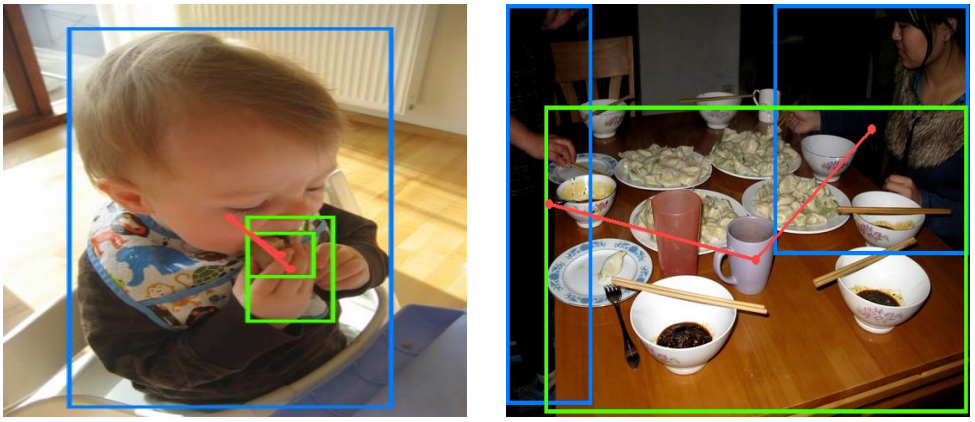

FIGURE 6.5: Visualization results of two failure cases.

λ_o and λ_a change significantly, the model exhibits a bit sensitive to the assigned weights. However, when the changes in the coefficients are relatively small, the model is insensitive to the weights, and our method has achieved competitive results under various weights.

6.4 Limitations

Our architecture consists of a shared transformer encoder along with two separate decoders, making it computationally expensive, which could be detrimental in practical applications. To mitigate this impact, we employ a pre-trained backbone and only fine-tune our network during training to minimize the consumption of computational resources. Additionally, we

show failure cases in Fig 6.5. Due to the presence of complex or overlapping objects, the model is unable to accurately identify all from the visual context.

6.5 Conclusion

In this chapter, we have delved into the application of uncertainty estimation within Human-Object Interaction (HOI) detection, exploring its integration in two key aspects: interaction and detection. For interaction, we utilized dropout not only as a regularization but also as a means for estimating the variance in interaction predictions. This dual-purpose use of dropout allows our model to assess and adapt to the reliability of its interaction classifications dynamically. For detection, we treated the bounding box coordinates as Gaussian-distributed random variables, which enables our system to quantify the uncertainty of object localizations and integrate this information into the learning process, thus enhancing prediction confidence and accuracy. Our approach is designed to be orthogonal to existing methods, allowing it to be seamlessly integrated with other techniques, thereby augmenting their effectiveness with robust uncertainty modeling capabilities. This integration capability provides a flexible framework that can be adopted to enhance current HOI detection systems without requiring extensive modifications to existing architectures.

Chapter 7

Conclusion and Future Works

7.1 Summary of Contributions

This thesis navigates the generalization problem of DNNs from two aspects: ❶ generalization from one domain to another and ❷ generalization from one task to another.

For ❶, a novel PiPa framework is proposed in Chapter 3, which encourages models to mine the inherent domain-invariant contextual feature. Since PiPa does not introduce extra parameters or annotations, it can be combined with other existing methods to further facilitate the intra-domain knowledge learning. Additionally, a depth-aware multi-task learning framework is introduced in Chapter 4 that leverages depth guidance to enhance data augmentation and contextual learning. This framework refines cross-domain mixing by simulating real-world layouts with depth distributions of objects and introduces a cross-task encoder that optimizes multi-task features and focuses on discriminative depth features to aid contextual learning.

For ❷, GvSeg, the generalist video segmentation solution is presented in Chapter 5 that accommodates task-oriented properties into model learning. GvSeg conducts a holistic investigation on segment targets by disentangling them into three essential constitutes: appearance, shape, and position. By adjusting the involvement of these key elements in query initialization and object association, GvSeg realize customizable prioritization of *instance discrimination* or *semantic understanding* to address different tasks. GvSeg achieves customizable prioritization of instance discrimination or semantic understanding to address different tasks. This tailored approach allows GvSeg to consistently achieve

top-leading results in several video segmentation tasks, demonstrating strong generalization ability. Finally, Chapter 6 explores comprehensive visual understanding by focusing on a high-level semantic task: HOI Detection. Specifically, it delves into the application of uncertainty estimation within Human-Object Interaction (HOI) detection, integrating it in two key aspects: interaction and detection. This integration capability provides a flexible framework that can be adopted to enhance current HOI detection systems without requiring extensive modifications to existing architectures.

7.2 Future Works

Understanding visual scenes is a primary goal of computer vision. Future research will continue to enhance comprehensive visual understanding by expanding and refining the methods proposed in this thesis. Firstly, further study of the PiPa framework on relevant tasks, such as domain-adaptive video segmentation and open-set adaptation, is planned. Additionally, introducing knowledge from more modalities under the multi-task learning framework, such as LiDAR and 3D point clouds, has the potential to further enhance contextual learning for scene understanding.

Moreover, the proposed GvSeg framework, which addresses generalist video segmentation for EVS, VIS, VSS, and VPS, provides insight into designing a universal model capable of addressing a broader spectrum of vision-related tasks. The disentanglement of task-specific properties of moving objects can benefit various video tasks, such as Video Object Detection (VOD) and Multi-Object Tracking and Segmentation (MOTS).

From a social impacts perspective, it is important to note that the methods proposed in this thesis may face potential operational challenges in practical applications. To proactively address any adverse effects on individuals and society, a robust security protocol could be established in the future to ensure the safety and well-being of users and the broader community in case of any unforeseen issues.

In addition to the aforementioned perspectives, several unresolved challenges and potential research avenues merit further investigation. While the proposed methods demonstrate strong performance across multiple domains and tasks, existing solutions for comprehensive visual understanding still face limitations in handling extreme domain shifts, rare event detection, and long-term temporal reasoning. Addressing these challenges will require more robust and adaptable learning frameworks.

Moreover, there is an emerging trend in the computer vision community towards large-scale pre-training and the development of foundational models that can generalize across diverse visual tasks. Integrating insights from this trend into future work could further enhance the generalization ability and scalability of the proposed methods. Specifically, pre-training on massive, diverse video datasets and leveraging cross-modal signals, such as language and audio, could enable the construction of more resilient and versatile vision models.

Exploring these directions may significantly advance the goal of building comprehensive, flexible, and socially responsible visual understanding systems.

Bibliography

[1] Jiaxu Miao, Xiaohan Wang, Yu Wu, Wei Li, Xu Zhang, Yunchao Wei, and Yi Yang. Large-scale video panoptic segmentation in the wild: A benchmark. In *CVPR*, 2022.

[2] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, 2019.

[3] Jiaxu Miao, Yunchao Wei, Yu Wu, Chen Liang, Guangrui Li, and Yi Yang. Vspw: A large-scale dataset for video scene parsing in the wild. In *CVPR*, 2021.

[4] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018.

[5] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation. In *ECCV*, 2022.

[6] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[7] Mark Weber, Jun Xie, Maxwell Collins, Yukun Zhu, Paul Voigtlaender, Hartwig Adam, Bradley Green, Andreas Geiger, Bastian Leibe, Daniel Cremers, et al. Step: Segmenting and tracking every pixel. In *NeurIPS*, 2021.

[8] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip HS Torr, and Song Bai. Occluded video instance segmentation: A benchmark. *IJCV*, 130(8):2022–2039, 2022.

[9] Ali Athar, Jonathon Luiten, Paul Voigtlaender, Tarasha Khurana, Achal Dave, Bastian Leibe, and Deva Ramanan. Burst: A benchmark for unifying object recognition, segmentation and tracking in video. In *WACV*, 2023.

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

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

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

[13] Mu Chen, Minghan Chen, and Yi Yang. Uahoi: Uncertainty-aware robust interaction learning for hoi detection. *Computer Vision and Image Understanding*, page 104091, 2024.

[14] Mu Chen, Zhedong Zheng, and Yi Yang. Transferring to real-world layouts: A depth-aware framework for scene adaptation. In *ACM Multimedia*, 2024.

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

[16] Mu Chen, Zhedong Zheng, Yi Yang, and Tat-Seng Chua. Pipa: Pixel-and patch-wise self-supervised learning for domain adaptative semantic segmentation. In *ACM MM*, 2023.

[17] Mu Chen, Zhedong Zheng, and Yi Yang. Pipa++: Towards unification of domain adaptive semantic segmentation via self-supervised learning. *arXiv preprint arXiv:2407.17101*, 2024.

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

[19] Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *ECCV*, 2018.

[20] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *CVPR*, 2021.

[21] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2017.

[22] 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.

[23] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016.

[24] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016.

[25] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish

Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *CVPR*, 2018.

[26] Zhenchao Jin, Tao Gong, Dongdong Yu, Qi Chu, Jian Wang, Changhu Wang, and Jie Shao. Mining contextual information beyond image for semantic segmentation. In *ICCV*, 2021.

[27] Changqian Yu, Jingbo Wang, Changxin Gao, Gang Yu, Chunhua Shen, and Nong Sang. Context prior for scene segmentation. In *CVPR*, 2020.

[28] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *ECCV*, 2020.

[29] Wenchao Gu, Shuang Bai, and Lingxing Kong. A review on 2d instance segmentation based on deep neural networks. *Image and Vision Computing*, 2022.

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

[31] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *ECCV*, 2020.

[32] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. SOLOv2: Dynamic and fast instance segmentation. In *NeurIPS*, 2020.

[33] Justin Lazarow, Kwonjoon Lee, Kunyu Shi, and Zhuowen Tu. Learning instance occlusion for panoptic segmentation. In *CVPR*, 2020.

[34] Haochen Wang, Ruotian Luo, Michael Maire, and Greg Shakhnarovich. Pixel consensus voting for panoptic segmentation. In *CVPR*, 2020.

[35] Qizhu Li, Xiaojuan Qi, and Philip HS Torr. Unifying training and inference for panoptic segmentation. In *CVPR*, 2020.

[36] Huanyu Liu, Chao Peng, Changqian Yu, Jingbo Wang, Xu Liu, Gang Yu, and Wei Jiang. An end-to-end network for panoptic segmentation. In *CVPR*, 2019.

[37] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *CVPR*, 2019.

[38] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. In *NeurIPS*, 2021.

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

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

[41] Jitesh Jain, Jiachen Li, Mang Tik Chiu, Ali Hassani, Nikita Orlov, and Humphrey

Shi. Oneformer: One transformer to rule universal image segmentation. In *CVPR*, 2023.

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

[43] Bowen Cheng, Anwesa Choudhuri, Ishan Misra, Alexander Kirillov, Rohit Girdhar, and Alexander G Schwing. Mask2former for video instance segmentation. *arXiv preprint arXiv:2112.10764*, 2021.

[44] James Chenhao Liang, Tianfei Zhou, Dongfang Liu, and Wenguan Wang. Clustseg: Clustering for universal segmentation. In *ICML*, 2023.

[45] Hefeng Wang, Jiale Cao, Rao Muhammad Anwer, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. Dformer: Diffusion-guided transformer for universal image segmentation. *arXiv preprint arXiv:2306.03437*, 2023.

[46] Wenguan Wang, Yi Yang, and Yunhe Pan. Visual knowledge in the big model era: Retrospect and prospect. *arXiv preprint arXiv:2404.04308*, 2024.

[47] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018.

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

[49] 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.

[50] Bingjie Xu, Yongkang Wong, Junnan Li, Qi Zhao, and Mohan S Kankanhalli. Learning to detect human-object interactions with knowledge. In *CVPR*, 2019.

[51] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *ICCV*, 2019.

[52] 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.

[53] Oytun Ulutan, ASM Iftekhar, and Bangalore S Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. In *CVPR*, 2020.

[54] Hai Wang, Wei-shi Zheng, and Ling Yingbiao. Contextual heterogeneous graph network for human-object interaction detection. In *ECCV*, 2020.

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

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

[57] Yong-Lu Li, Xinpeng Liu, Xiaoqian Wu, Yizhuo Li, and Cewu Lu. Hoi analysis: Integrating and decomposing human-object interaction. In *NeurIPS*, 2020.

[58] 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.

[59] 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.

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

[61] 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.

[62] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *CVPR*, 2020.

[63] 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.

[64] Yang Liu, Qingchao Chen, and Andrew Zisserman. Amplifying key cues for human-object-interaction detection. In *ECCV*, 2020.

[65] Tianfei Zhou, Wenguan Wang, Siyuan Qi, Haibin Ling, and Jianbing Shen. Cascaded human-object interaction recognition. In *CVPR*, 2020.

[66] Frederic Z Zhang, Dylan Campbell, and Stephen Gould. Spatially conditioned graphs for detecting human-object interactions. In *ICCV*, 2021.

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

[68] 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.

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

[70] 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.

[71] Masato Tamura, Hiroki Ohashi, and Tomoaki Yoshinaga. Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information. In *CVPR*, 2021.

[72] 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.

[73] 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.

[74] 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.

[75] 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.

[76] 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.

[77] 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.

[78] 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.

[79] Hangjie Yuan, Jianwen Jiang, Samuel Albanie, Tao Feng, Ziyuan Huang, Dong Ni, and Mingqian Tang. Rlip: Relational language-image pre-training for human-object interaction detection. In *NeurIPS*, 2022.

[80] Xubin Zhong, Changxing Ding, Zijian Li, and Shaoli Huang. Towards hard-positive query mining for detr-based human-object interaction detection. In *ECCV*, 2022.

[81] Hangjie Yuan, Mang Wang, Dong Ni, and Liangpeng Xu. Detecting human-object interactions with object-guided cross-modal calibrated semantics. In *AAAI*, 2022.

[82] Yichao Cao, Qingfei Tang, Xiu Su, Chen Song, Shan You, Xiaobo Lu, and Chang Xu. Detecting any human-object interaction relationship: universal hoi detector with spatial prompt learning on foundation models. In *NIPS*, 2023.

[83] Liulei Li, Jianan Wei, Wenguan Wang, and Yi Yang. Neural-logic human-object interaction detection. In *NeurIPS*, 2023.

[84] Lin Li, Jun Xiao, Hanrong Shi, Hanwang Zhang, Yi Yang, Wei Liu, and Long Chen.

Nicest: Noisy label correction and training for robust scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[85] Lin Li, Jun Xiao, Guikun Chen, Jian Shao, Yueting Zhuang, and Long Chen. Zero-shot visual relation detection via composite visual cues from large language models. In *NIPS*, 2023.

[86] Lin Li, Guikun Chen, Jun Xiao, Yi Yang, Chunping Wang, and Long Chen. Compositional feature augmentation for unbiased scene graph generation. In *ICCV*, 2023.

[87] Ali Athar, Alexander Hermans, Jonathon Luiten, Deva Ramanan, and Bastian Leibe. Tarvis: A unified approach for target-based video segmentation. In *CVPR*, 2023.

[88] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *IEEE TPAMI*, 44:4701–4712, 2021.

[89] 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.

[90] Jiaxu Miao, Yunchao Wei, and Yi Yang. Memory aggregation networks for efficient interactive video object segmentation. In *CVPR*, 2020.

[91] Ruizheng Wu, Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Memory selection network for video propagation. In *ECCV*, 2020.

[92] 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.

[93] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *ICCV*, 2019.

[94] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019.

[95] Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao, and Fatih Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In *CVPR*, 2019.

[96] Wenguan Wang, Jianbing Shen, Fatih Porikli, and Ruigang Yang. Semi-supervised video object segmentation with super-trajectories. *IEEE TPAMI*, 41(4):985–998, 2018.

[97] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *CVPR*, 2015.

[98] Hongje Seong, Seoung Wug Oh, Joon-Young Lee, Seongwon Lee, Suhyeon Lee, and Euntai Kim. Hierarchical memory matching network for video object segmentation. In *CVPR*, 2021.

[99] Yunyao Mao, Ning Wang, Wengang Zhou, and Houqiang Li. Joint inductive and transductive learning for video object segmentation. In *ICCV*, 2021.

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

[101] Ho Kei Cheng and Alexander G. Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV*, 2022.

[102] 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.

[103] Kwanyong Park, Sanghyun Woo, Seoung Wug Oh, In So Kweon, and Joon-Young Lee. Per-clip video object segmentation. In *CVPR*, 2022.

[104] Ye Yu, Jialin Yuan, Gaurav Mittal, Li Fuxin, and Mei Chen. Batman: Bilateral attention transformer in motion-appearance neighboring space for video object segmentation. In *ECCV*, 2022.

[105] 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.

[106] 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.

[107] Miran Heo, Sukjun Hwang, Jeongseok Hyun, Hanjung Kim, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. A generalized framework for video instance segmentation. In *CVPR*, 2023.

[108] Jiale Cao, Rao Muhammad Anwer, Hisham Cholakkal, Fahad Shahbaz Khan, Yanwei Pang, and Ling Shao. Sipmask: Spatial information preservation for fast image and video instance segmentation. In *ECCV*, 2020.

[109] Dongfang Liu, Yiming Cui, Wenbo Tan, and Yingjie Chen. Sg-net: Spatial granularity network for one-stage video instance segmentation. In *CVPR*, 2021.

[110] Shusheng Yang, Yuxin Fang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Crossover learning for fast online video instance segmentation. In *ICCV*, 2021.

[111] Su Ho Han, Sukjun Hwang, Seoung Wug Oh, Yeonchool Park, Hyunwoo Kim, Min-Jung Kim, and Seon Joo Kim. Visolo: Grid-based space-time aggregation for efficient online video instance segmentation. In *CVPR*, 2022.

[112] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *ICCV*, 2021.

[113] Feng Zhu, Zongxin Yang, Xin Yu, Yi Yang, and Yunchao Wei. Instance as identity: A generic online paradigm for video instance segmentation. In *ECCV*, 2022.

[114] Minghan Li, Shuai Li, Lida Li, and Lei Zhang. Spatial feature calibration and temporal fusion for effective one-stage video instance segmentation. In *CVPR*, 2021.

[115] Lei Ke, Xia Li, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Prototypical cross-attention networks for multiple object tracking and segmentation. In *NeurIPS*, 2021.

[116] Huaijia Lin, Ruizheng Wu, Shu Liu, Jiangbo Lu, and Jiaya Jia. Video instance segmentation with a propose-reduce paradigm. In *ICCV*, 2021.

[117] De-An Huang, Zhiding Yu, and Anima Anandkumar. Minvis: A minimal video instance segmentation framework without video-based training. In *NeurIPS*, 2022.

[118] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In defense of online models for video instance segmentation. In *ECCV*, 2022.

[119] Fei He, Haoyang Zhang, Naiyu Gao, Jian Jia, Yanhu Shan, Xin Zhao, and Kaiqi Huang. Inspro: Propagating instance query and proposal for online video instance segmentation. In *NeurIPS*, 2022.

[120] Rajat Koner, Tanveer Hannan, Suprosanna Shit, Sahand Sharifzadeh, Matthias Schubert, Thomas Seidl, and Volker Tresp. Instanceformer: An online video instance segmentation framework. In *AAAI*, 2023.

[121] Qihao Liu, Junfeng Wu, Yi Jiang, Xiang Bai, Alan L Yuille, and Song Bai. Instmove: Instance motion for object-centric video segmentation. In *CVPR*, 2023.

[122] Minghan Li, Shuai Li, Wangmeng Xiang, and Lei Zhang. Mdqe: Mining discriminative query embeddings to segment occluded instances on challenging videos. In *CVPR*, 2023.

[123] Ali Athar, Sabarinath Mahadevan, Aljosa Osep, Laura Leal-Taixé, and Bastian Leibe. Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *ECCV*, 2020.

[124] Jialian Wu, Sudhir Yarram, Hui Liang, Tian Lan, Junsong Yuan, Jayan Eledath, and Gerard Medioni. Efficient video instance segmentation via tracklet query and proposal. In *CVPR*, 2022.

[125] Shusheng Yang, Xinggang Wang, Yu Li, Yuxin Fang, Jiemin Fang, Wenyu Liu, Xun Zhao, and Ying Shan. Temporally efficient vision transformer for video instance segmentation. In *CVPR*, 2022.

[126] Dahun Kim, Jun Xie, Huiyu Wang, Siyuan Qiao, Qihang Yu, Hong-Seok Kim, Hartwig Adam, In So Kweon, and Liang-Chieh Chen. Tubeformer-deeplab: Video mask transformer. In *CVPR*, 2022.

[127] Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *CVPR*, 2020.

[128] Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. In *NeurIPS*, 2021.

[129] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *CVPR*, 2021.

[130] Junfeng Wu, Yi Jiang, Song Bai, Wenqing Zhang, and Xiang Bai. Seqformer: Sequential transformer for video instance segmentation. In *ECCV*, 2022.

[131] Miran Heo, Sukjun Hwang, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. Vita: Video instance segmentation via object token association. In *NeurIPS*, 2022.

[132] 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.

[133] Tianfei Zhou, Wenguan Wang, Ender Konukoglu, and Luc Van Gool. Rethinking semantic segmentation: A prototype view. In *CVPR*, 2022.

[134] Liulei Li, Wenguan Wang, and Yi Yang. Logicseg: Parsing visual semantics with neural logic learning and reasoning. In *ICCV*, 2023.

[135] Liulei Li, Tianfei Zhou, Wenguan Wang, Jianwu Li, and Yi Yang. Deep hierarchical semantic segmentation. In *CVPR*, 2022.

[136] Liulei Li, Wenguan Wang, Tianfei Zhou, Ruijie Quan, and Yi Yang. Semantic hierarchy-aware segmentation. *IEEE TPAMI*, 2023.

[137] Mu Chen, Zhedong Zheng, and Yi Yang. Transferring to real-world layouts: A depth-aware framework for scene adaptation. *arXiv preprint arXiv:2311.12682*, 2023.

[138] Tianfei Zhou and Wenguan Wang. Cross-image pixel contrasting for semantic segmentation. *IEEE TPAMI*, 2024.

[139] Yu-Syuan Xu, Tsu-Jui Fu, Hsuan-Kung Yang, and Chun-Yi Lee. Dynamic video segmentation network. In *CVPR*, 2018.

[140] Behrooz Mahasseni, Sinisa Todorovic, and Alan Fern. Budget-aware deep semantic video segmentation. In *CVPR*, 2017.

[141] David Nilsson and Cristian Sminchisescu. Semantic video segmentation by gated recurrent flow propagation. In *CVPR*, 2018.

[142] Samvit Jain, Xin Wang, and Joseph E Gonzalez. Accel: A corrective fusion network for efficient semantic segmentation on video. In *CVPR*, 2019.

[143] Yifan Liu, Chunhua Shen, Changqian Yu, and Jingdong Wang. Efficient semantic video segmentation with per-frame inference. In *ECCV*, 2020.

[144] Tianfei Zhou, Fatih Porikli, David J Crandall, Luc Van Gool, and Wenguan Wang. A survey on deep learning technique for video segmentation. *IEEE TPAMI*, 45(6):7099–7122, 2022.

[145] Matthieu Paul, Martin Danelljan, Luc Van Gool, and Radu Timofte. Local memory attention for fast video semantic segmentation. In *IROS*, 2021.

[146] Wei Ji, Jingjing Li, Cheng Bian, Zongwei Zhou, Jiaying Zhao, Alan L Yuille, and Li Cheng. Multispectral video semantic segmentation: A benchmark dataset and baseline. In *CVPR*, 2023.

[147] Guolei Sun, Yun Liu, Henghui Ding, Thomas Probst, and Luc Van Gool. Coarse-to-fine feature mining for video semantic segmentation. In *CVPR*, 2022.

[148] Jiangtong Li, Wentao Wang, Junjie Chen, Li Niu, Jianlou Si, Chen Qian, and Liqing Zhang. Video semantic segmentation via sparse temporal transformer. In *ACM MM*, 2021.

[149] Guolei Sun, Yun Liu, Hao Tang, Ajad Chhatkuli, Le Zhang, and Luc Van Gool. Mining relations among cross-frame affinities for video semantic segmentation. In *ECCV*, 2022.

[150] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *CVPR*, 2020.

[151] Sanghyun Woo, Dahun Kim, Joon-Young Lee, and In So Kweon. Learning to associate every segment for video panoptic segmentation. In *CVPR*, 2021.

[152] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *CVPR*, 2021.

[153] Lars Kreuzberg, Idil Esen Zulfikar, Sabarinath Mahadevan, Francis Engelmann, and Bastian Leibe. 4d-stop: Panoptic segmentation of 4d lidar using spatio-temporal object proposal generation and aggregation. In *ECCV*, 2022.

[154] Yi Zhou, Hui Zhang, Hana Lee, Shuyang Sun, Pingjun Li, Yangguang Zhu, ByungIn

Yoo, Xiaojuan Qi, and Jae-Joon Han. Slot-vps: Object-centric representation learning for video panoptic segmentation. In *CVPR*, 2022.

[155] Haobo Yuan, Xiangtai Li, Yibo Yang, Guangliang Cheng, Jing Zhang, Yunhai Tong, Lefei Zhang, and Dacheng Tao. Polyphonicformer: unified query learning for depth-aware video panoptic segmentation. In *ECCV*, 2022.

[156] Junwen He, Yifan Wang, Lijun Wang, Huchuan Lu, Bin Luo, Jun-Yan He, Jin-Peng Lan, Yifeng Geng, and Xuansong Xie. Towards deeply unified depth-aware panoptic segmentation with bi-directional guidance learning. In *ICCV*, 2023.

[157] Inkyu Shin, Dahun Kim, Qihang Yu, Jun Xie, Hong-Seok Kim, Bradley Green, In So Kweon, Kuk-Jin Yoon, and Liang-Chieh Chen. Video-kmax: A simple unified approach for online and near-online video panoptic segmentation. *arXiv preprint arXiv:2304.04694*, 2023.

[158] Xiangtai Li, Wenwei Zhang, Jiangmiao Pang, Kai Chen, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Video k-net: A simple, strong, and unified baseline for video segmentation. In *CVPR*, 2022.

[159] Anwesa Choudhuri, Girish Chowdhary, and Alexander G Schwing. Context-aware relative object queries to unify video instance and panoptic segmentation. In *CVPR*, 2023.

[160] Xiangtai Li, Wenwei Zhang, Jiangmiao Pang, Kai Chen, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Tube-link: A flexible cross tube baseline for universal video segmentation. In *ICCV*, 2023.

[161] Tao Zhang, Xingye Tian, Yu Wu, Shunping Ji, Xuebo Wang, Yuan Zhang, and Pengfei Wan. Dvis: Decoupled video instance segmentation framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1282–1291, 2023.

[162] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *CVPR*, 2018.

[163] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019.

[164] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *CVPR*, 2019.

[165] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain

adaptation. In *ICML*, 2018.

[166] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen. Reusing the task-specific classifier as a discriminator: Discriminator-free adversarial domain adaptation. In *CVPR*, 2022.

[167] Zuxuan Wu, Xin Wang, Joseph E Gonzalez, Tom Goldstein, and Larry S Davis. Ace: Adapting to changing environments for semantic segmentation. In *ICCV*, 2019.

[168] Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. In *ICCV*, 2019.

[169] Yuyang Zhao, Zhun Zhong, Zhiming Luo, Gim Hee Lee, and Nicu Sebe. Source-free open compound domain adaptation in semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(10):7019–7032, 2022.

[170] Xinyu Zhang, Dongdong Li, Zhigang Wang, Jian Wang, Errui Ding, Javen Qinfeng Shi, Zhaoxiang Zhang, and Jingdong Wang. Implicit sample extension for unsupervised person re-identification. In *CVPR*, 2022.

[171] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018.

[172] Hao Feng, Minghao Chen, Jinming Hu, Dong Shen, Haifeng Liu, and Deng Cai. Complementary pseudo labels for unsupervised domain adaptation on person re-identification. *IEEE Transactions on Image Processing*, 30:2898–2907, 2021.

[173] Qing Lian, Fengmao Lv, Lixin Duan, and Boqing Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *ICCV*, 2019.

[174] Zhedong Zheng and Yi Yang. Unsupervised scene adaptation with memory regularization in vivo. In *IJCAI*, 2020.

[175] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *ICCV*, 2019.

[176] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *International Journal of Computer Vision*, 129(4):1106–1120, 2021.

[177] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *CVPR*, 2021.

[178] Qiming Zhang, Jing Zhang, Wei Liu, and Dacheng Tao. Category anchor-guided unsupervised domain adaptation for semantic segmentation. In *NeurIPS*, 2019.

[179] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *WACV Workshop*, 2021.

[180] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *CVPR*, 2020.

[181] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *CVPR*, 2019.

[182] Haoran Wang, Tong Shen, Wei Zhang, Ling-Yu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *ECCV*, 2020.

[183] Lin Chen, Zhixiang Wei, Xin Jin, Huaian Chen, Miao Zheng, Kai Chen, and Yi Jin. Deliberated domain bridging for domain adaptive semantic segmentation. In *NeurIPS*, 2022.

[184] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. Sepico: Semantic-guided pixel contrast for domain adaptive semantic segmentation. *arXiv:2204.08808*, 2022.

[185] Yahao Liu, Jinhong Deng, Jiale Tao, Tong Chu, Lixin Duan, and Wen Li. Undoing the damage of label shift for cross-domain semantic segmentation. In *CVPR*, 2022.

[186] Minghao Chen, Hongyang Xue, and Deng Cai. Domain adaptation for semantic segmentation with maximum squares loss. In *ICCV*, 2019.

[187] Shaohua Guo, Qianyu Zhou, Ye Zhou, Qiqi Gu, Junshu Tang, Zhengyang Feng, and Lizhuang Ma. Label-free regional consistency for image-to-image translation. In *ICME*, 2021.

[188] Jinyu Yang, Weizhi An, Sheng Wang, Xinliang Zhu, Chaochao Yan, and Junzhou Huang. Label-driven reconstruction for domain adaptation in semantic segmentation. In *ECCV*, 2020.

[189] Nikita Araslanov and Stefan Roth. Self-supervised augmentation consistency for adapting semantic segmentation. In *CVPR*, 2021.

[190] Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *ICCV*, 2019.

[191] Luke Melas-Kyriazi and Arjun K Manrai. Pixmatch: Unsupervised domain adaptation via pixelwise consistency training. In *CVPR*, 2021.

[192] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Jiangmiao Pang, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Context-aware mixup for domain adaptive

semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[193] Behnam Gholami, Pritish Sahu, Ognjen Rudovic, Konstantinos Bousmalis, and Vladimir Pavlovic. Unsupervised multi-target domain adaptation: An information theoretic approach. *IEEE Transactions on Image Processing*, 29:3993–4002, 2020.

[194] Antoine Saporta, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Multi-target adversarial frameworks for domain adaptation in semantic segmentation. In *ICCV*, 2021.

[195] Seunghun Lee, Wonhyeok Choi, Changjae Kim, Minwoo Choi, and Sunghoon Im. Adas: A direct adaptation strategy for multi-target domain adaptive semantic segmentation. In *CVPR*, 2022.

[196] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

[197] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[198] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020.

[199] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

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

[201] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *CVPR*, 2019.

[202] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, 2017.

[203] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021.

[204] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *ICCV*, 2021.

[205] 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.

[206] Weizhe Liu, David Ferstl, Samuel Schulter, Lukas Zebedin, Pascal Fua, and Christian Leistner. Domain adaptation for semantic segmentation via patch-wise contrastive

learning. *arXiv:2104.11056*, 2021.

[207] Zhengkai Jiang, Yuxi Li, Ceyuan Yang, Peng Gao, Yabiao Wang, Ying Tai, and Chengjie Wang. Prototypical contrast adaptation for domain adaptive segmentation. In *ECCV*, 2022.

[208] Jiaxing Huang, Dayan Guan, Aoran Xiao, Shijian Lu, and Ling Shao. Category contrast for unsupervised domain adaptation in visual tasks. In *CVPR*, 2022.

[209] Binhui Xie, Mingjia Li, and Shuang Li. Spcl: A new framework for domain adaptive semantic segmentation via semantic prototype-based contrastive learning. *arXiv:2111.12358*, 2021.

[210] Inigo Alonso, Alberto Sabater, David Ferstl, Luis Montesano, and Ana C Murillo. Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *ICCV*, 2021.

[211] Qianyu Zhou, Chuyun Zhuang, Ran Yi, Xuequan Lu, and Lizhuang Ma. Domain adaptive semantic segmentation via regional contrastive consistency regularization. In *ICME*, 2022.

[212] Xin Lai, Zhuotao Tian, Li Jiang, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. Semi-supervised semantic segmentation with directional context-aware consistency. In *CVPR*, 2021.

[213] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Zequn Jie, Xiang Li, and Jian Yang. Joint task-recursive learning for semantic segmentation and depth estimation. In *ECCV*, 2018.

[214] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *CVPR*, 2018.

[215] Menelaos Kanakis, David Bruggemann, Suman Saha, Stamatios Georgoulis, Anton Obukhov, and Luc Van Gool. Reparameterizing convolutions for incremental multi-task learning without task interference. In *ECCV*, 2020.

[216] Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *ECCV*, 2020.

[217] Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *ICML*, 2020.

[218] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *CVPR*, 2019.

[219] Yaxiong Wang, Yunchao Wei, Xueming Qian, Li Zhu, and Yi Yang. Ainet: Association implantation for superpixel segmentation. In *ICCV*, 2021.

[220] Kuan-Hui Lee, German Ros, Jie Li, and Adrien Gaidon. Spigan: Privileged adversarial learning from simulation. *arXiv:1810.03756*, 2018.

[221] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *ICCV*, 2019.

[222] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *CVPR*, pages 1841–1850, 2019.

[223] Suman Saha, Anton Obukhov, Danda Pani Paudel, Menelaos Kanakis, Yuhua Chen, Stamatios Georgoulis, and Luc Van Gool. Learning to relate depth and semantics for unsupervised domain adaptation. In *CVPR*, 2021.

[224] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. Domain adaptive semantic segmentation with self-supervised depth estimation. In *ICCV*, 2021.

[225] Quanliang Wu and Huajun Liu. Unsupervised domain adaptation for semantic segmentation using depth distribution. In *NeurIPS*, 2022.

[226] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[227] 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 transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[228] 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.

[229] 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.

[230] Liang Han, Pichao Wang, Zhaozheng Yin, Fan Wang, and Hao Li. Exploiting better feature aggregation for video object detection. In *ACM Multimedia*, 2020.

[231] Shuyu Yang, Yinan Zhou, Yaxiong Wang, Yujiao Wu, Li Zhu, and Zhedong Zheng. Towards unified text-based person retrieval: A large-scale multi-attribute and language search benchmark. *arXiv:2306.02898*, 2023.

[232] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogerio Feris, Jinjun Xiong, Wen-mei Hwu, Thomas S Huang, and Honghui Shi. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. In *CVPR*, 2020.

[233] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016.

[234] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016.

[235] Tingyu Wang, Zhedong Zheng, Yaoqi Sun, Tat-Seng Chua, Yi Yang, and Chenggang Yan. Multiple-environment self-adaptive network for aerial-view geo-localization. *arXiv preprint arXiv:2204.08381*, 2022.

[236] Jianwu Fang, Fan Wang, Peining Shen, Zhedong Zheng, Jianru Xue, and Tat-seng Chua. Behavioral intention prediction in driving scenes: A survey. *arXiv:2211.00385*, 2022.

[237] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *CVPR*, 2020.

[238] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Significance-aware information bottleneck for domain adaptive semantic segmentation. In *ICCV*, 2019.

[239] Haoshuo Huang, Qixing Huang, and Philipp Krahenbuhl. Domain transfer through deep activation matching. In *ECCV*, 2018.

[240] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *ICCV*, 2019.

[241] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*, 2020.

[242] Nikos Komodakis and Spyros Gidaris. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018.

[243] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[244] Chao Sun, Zhedong Zheng, Xiaohan Wang, Mingliang Xu, and Yi Yang. Self-supervised point cloud representation learning via separating mixed shapes. *IEEE Transactions on Multimedia*, 2022.

[245] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020.

[246] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. 2017.

[247] Zhedong Zheng and Yi Yang. Adaptive boosting for domain adaptation: Toward robust predictions in scene segmentation. *IEEE Transactions on Image Processing*, 31:5371–5382, 2022.

[248] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022.

[249] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv:1710.09412*, 2017.

[250] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *NeurIPS*, 2021.

[251] 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.

[252] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[253] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *WACV*, 2021.

[254] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[255] Yuhu Shan, Chee Meng Chew, and Wen Feng Lu. Semantic-aware short path adversarial training for cross-domain semantic segmentation. *Neurocomputing*, 380:125–132, 2020.

[256] Wei Zhou, Yukang Wang, Jiajia Chu, Jiehua Yang, Xiang Bai, and Yongchao Xu. Affinity space adaptation for semantic segmentation across domains. *IEEE Transactions on Image Processing*, 30:2549–2561, 2020.

[257] Jihan Yang, Ruijia Xu, Ruiyu Li, Xiaojuan Qi, Xiaoyong Shen, Guanbin Li, and Liang Lin. An adversarial perturbation oriented domain adaptation approach for semantic segmentation. In *AAAI*, 2020.

[258] Yahao Liu, Jinhong Deng, Xinchen Gao, Wen Li, and Lixin Duan. Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation. In *ICCV*, 2021.

[259] Midhun Vayyat, Jaswin Kasi, Anuraag Bhattacharya, Shuaib Ahmed, and Rahul Tallamraju. Cluda: Contrastive learning in unsupervised domain adaptation for semantic segmentation. *arXiv:2208.14227*, 2022.

[260] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. Content-consistent matching for domain adaptive semantic segmentation. In *ECCV*, 2020.

[261] Sicheng Zhao, Bo Li, Xiangyu Yue, Yang Gu, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source domain adaptation for semantic segmentation. *NeurIPS*, 2019.

[262] Jianzhong He, Xu Jia, Shuaijun Chen, and Jianzhuang Liu. Multi-source domain adaptation with collaborative learning for semantic segmentation. In *CVPR*, 2021.

[263] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. Mic: Masked image consistency for context-enhanced domain adaptation. In *CVPR*, 2023.

[264] David Brüggemann, Christos Sakaridis, Prune Truong, and Luc Van Gool. Refign: Align and refine for adaptation of semantic segmentation to adverse conditions. In *WACV*, 2023.

[265] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[266] Xinyi Wu, Zhenyao Wu, Hao Guo, Lili Ju, and Song Wang. Dannet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In *CVPR*, 2021.

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

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

[269] 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.

[270] Harsh Rangwani, Sumukh K Aithal, Mayank Mishra, Arihant Jain, and

Venkatesh Babu Radhakrishnan. A closer look at smoothness in domain adversarial training. In *ICML*, 2022.

[271] Ruihuang Li, Shuai Li, Chenhang He, Yabin Zhang, Xu Jia, and Lei Zhang. Class-balanced pixel-level self-labeling for domain adaptive semantic segmentation. In *CVPR*, 2022.

[272] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. MIC: Masked image consistency for context-enhanced domain adaptation. In *CVPR*, 2023.

[273] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, 2019.

[274] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016.

[275] Adriano Cardace, Luca De Luigi, Pierluigi Zama Ramirez, Samuele Salti, and Luigi Di Stefano. Plugging self-supervised monocular depth into unsupervised domain adaptation for semantic segmentation. In *CVPR*, 2022.

[276] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *ECCV*, 2020.

[277] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[278] Ping Hu, Fabian Caba, Oliver Wang, Zhe Lin, Stan Sclaroff, and Federico Perazzi. Temporally distributed networks for fast video semantic segmentation. In *CVPR*, 2020.

[279] Chen Liang, Wenguan Wang, Tianfei Zhou, Jiaxu Miao, Yawei Luo, and Yi Yang. Local-global context aware transformer for language-guided video segmentation. *IEEE TPAMI*, 45(8):10055–10069, 2023.

[280] Tianrui Hui, Si Liu, Zihan Ding, Shaofei Huang, Guanbin Li, Wenguan Wang, Luoqi Liu, and Jizhong Han. Language-aware spatial-temporal collaboration for referring video segmentation. *IEEE TPAMI*, 45(7):8646–8659, 2023.

[281] 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.

[282] Wenguan Wang, Jianbing Shen, Xuelong Li, and Fatih Porikli. Robust video object cosegmentation. *IEEE TIP*, 24(10):3137–3148, 2015.

[283] Wenguan Wang, Jianbing Shen, Jianwen Xie, and Fatih Porikli. Super-trajectory for video segmentation. In *ICCV*, 2017.

[284] Zheyun Qin, Xiankai Lu, Xiushan Nie, Dongfang Liu, Yilong Yin, and Wenguan Wang. Coarse-to-fine video instance segmentation with factorized conditional appearance

flows. *IEEE/CAA Journal of Automatica Sinica*, 10(5):1192–1208, 2023.

[285] Edward H Adelson. On seeing stuff: the perception of materials by humans and machines. In *Human vision and electronic imaging VI*, 2001.

[286] Jack M Loomis, John W Philbeck, and Pavel Zahorik. Dissociation between location and shape in visual space. *Journal of Experimental Psychology: Human Perception and Performance*, 28(5):1202, 2002.

[287] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE TPAMI*, 24(4):509–522, 2002.

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

[289] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, Ping Luo, and Tong Lu. Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In *CVPR*, 2022.

[290] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, 2020.

[291] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[292] 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.

[293] Bin Yan, Yi Jiang, Jiannan Wu, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Universal instance perception as object discovery and retrieval. In *CVPR*, 2023.

[294] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Alexander Schwing, and Joon-Young Lee. Tracking anything with decoupled video segmentation. In *ICCV*, 2023.

[295] Junlong Li, Bingyao Yu, Yongming Rao, Jie Zhou, and Jiwen Lu. Tcovis: Temporally consistent online video instance segmentation. In *ICCV*, 2023.

[296] Kaining Ying, Qing Zhong, Weian Mao, Zhenhua Wang, Hao Chen, Lin Yuanbo Wu, Yifan Liu, Chengxiang Fan, Yunzhi Zhuge, and Chunhua Shen. Ctvis: Consistent training for online video instance segmentation. In *ICCV*, 2023.

[297] Arne Hoffhues and Jonathon Luiten. Trackeval. 2020.

[298] Bin Yan, Yi Jiang, Jiannan Wu, Dong Wang, Ping Luo, Zehuan Yuan, and Huchuan Lu. Universal instance perception as object discovery and retrieval. In *CVPR*, 2023.

[299] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *IJCV*, 129:548–578, 2021.

[300] Zhifan Ni, Esteve Valls Mascaró, Hyemin Ahn, and Dongheui Lee. Human–object interaction prediction in videos through gaze following. *Computer Vision and Image Understanding*, 233:103741, 2023.

[301] Rosario Leonardi, Francesco Ragusa, Antonino Furnari, and Giovanni Maria Farinella. Exploiting multimodal synthetic data for egocentric human-object interaction detection in an industrial scenario. *Computer Vision and Image Understanding*, 242:103984, 2024.

[302] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton Van Den Hengel. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40, 2017.

[303] Qi Rao, Xin Yu, Guang Li, and Linchao Zhu. Cmgnet: Collaborative multi-modal graph network for video captioning. *Computer Vision and Image Understanding*, 238:103864, 2024.

[304] Fudong Nian, Teng Li, Yan Wang, Xinyu Wu, Bingbing Ni, and Changsheng Xu. Learning explicit video attributes from mid-level representation for video captioning. *Computer Vision and Image Understanding*, 163:126–138, 2017.

[305] Utku Ozbulak, Baptist Vandersmissen, Azarakhsh Jalalvand, Ivo Couckuyt, Arnout Van Messem, and Wesley De Neve. Investigating the significance of adversarial attacks and their relation to interpretability for radar-based human activity recognition systems. *Computer Vision and Image Understanding*, 202:103111, 2021.

[306] Liyue Shen, Serena Yeung, Judy Hoffman, Greg Mori, and Li Fei-Fei. Scaling human-object interaction recognition through zero-shot learning. 2018.

[307] Chen Liang, Wenguan Wang, Jiaxu Miao, and Yi Yang. Logic-induced diagnostic reasoning for semi-supervised semantic segmentation. In *ICCV*, 2023.

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

[309] Youngwan Lee, Joong-won Hwang, Hyung-Il Kim, Kimin Yun, Yongjin Kwon, Yuseok Bae, and Sung Ju Hwang. Localization uncertainty estimation for anchor-free object detection. In *ECCV*, 2022.

[310] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[311] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *NeurIPS*, 2017.

[312] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

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

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

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

[316] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. 2016.

[317] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NeurIPS*, 2017.

[318] Kamil Ciosek, Vincent Fortuin, Ryota Tomioka, Katja Hofmann, and Richard Turner. Conservative uncertainty estimation by fitting prior networks. In *ICLR*, 2019.