# AN ABSTRACT OF THE DISSERTATION OF

Jialin Yuan for the degree of Doctor of Philosophy in Computer Science presented on October 18, 2023.

Title: Visual Object Discovery and Understanding

Abstract approved: _____

Fuxin Li

Learning to recognize objects is a fundamental and essential step in human perception and understanding of the world. Accordingly, research of object discovery across diverse modalities plays a pivotal role in the context of computer vision. This field not only contributes significantly to enhancing our understanding of visual information but also offers a plethora of potential applications, like augmented reality, e-commerce, and robotics, particularly in industrial manipulation scenarios.

We first address the task of discovering objects from still images regardless of any predefined categories. We introduce a novel variational relaxation approach tailored to the task. By framing it as an optimization problem for piecewise-constant segmentation, this technique enables direct training of a fully convolutional network (FCN) for predicting object labels on each pixel. Applying our approach to the *instance segmentation* task achieved results almost as good as mask R-CNN without depending on a two-stage framework. Note that the training of the network does not depend on the category label, enabling our approach to discover objects unbounded by predefined categories.

Next, we extend our exploration to video sequences, focusing on the task of *unsupervised video object segmentation*. Here, we aim to discover and track objects within videos. Noticing that single-frame object proposals often fail to obtain a good proposal due to motion blur, occlusion, and other reasons, our approach involves refining key frame

proposals using a Multi-proposal graph constructed from proposals initially generated in nearby frames and then propagated to the key frame. We then compute the maximal cliques within this graph, which contains proposals that represent the same object. Pixel-level voting is performed within each clique to generate the key frame proposals that could be better than any of the single-frame proposals. Then a semi-supervised VOS algorithm subsequently tracks these key frame proposals across the entire video, showcasing the potential for precise and robust object tracking in dynamic visual environments.

We further explore into the domain of Vision-Language, where we seek to identify objects associated with a specific textual context. In this multifaceted context, we tackle the intricate challenge of *content moderation (CM)*, which assesses multimodal user-generated content to detect material that is illegal, harmful, or insulting. We present a novel CM model to address the asymmetric in semantics between vision and language. Our model features an innovative asymmetric fusion architecture that not only fuses the common knowledge in both modalities but also leverages the unique information present in each modality. Additionally, we introduce a novel cross-modality contrastive loss to capture knowledge that arises exclusively in multimodal context, which is crucial for addressing harmful intent that may emerge at the intersection of these modalities.

# Visual Object Discovery and Understanding

by
Jialin Yuan

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented October 18, 2023
Commencement June 2024

Doctor of Philosophy dissertation of Jialin Yuan presented on October 18, 2023.

APPROVED:

---

Major Professor, representing Computer Science

---

Head of the School of Electrical Engineering and Computer Science

---

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

---

Jialin Yuan, Author

# ACKNOWLEDGEMENTS

First and foremost I am grateful to my advisor, Dr. Fuxin Li, for the support, encouragement, and mentorship through the whole PhD program. Fuxin has been an extraordinary advisor and an academic role model to me. He always encourages me to practice engineering skills and presentation skills. Meanwhile, he emphasizes the importance of thinking deeper and going beyond architecture engineering in research. I am also thankful to each member of my committee: Prof. Sinisa Todorovic, Prof. Raviv Raich, Prof. Prasad Tadepalli, and Prof. Geoff Hollinger. I really appreciate their responsiveness in all communications and their time and effort during the exams and meetings. I extend a special and sincere thanks to Prof. Chao Chen, who gave me lots of guidance with his expertise in my first research work and valuable research advice.

Next, I would like to thank Microsoft Corporation Inc. and Uber Technologies, Inc., for providing me the great internship experiences during my PhD. I learned a lot from these internships including technical knowledge, communications, and teamwork. I am very grateful to my mentors and co-workers during the internships: Jingchen Liu, Yuh-jie Chen, Ye Yu, Gaurav Mittal, and Mei Chen. They kept me grounded in the task of solving real problems and were willing to assist me with any problem at any time. All of these have impacted every aspect of my research. Ye Yu and Mei Chen, in particular, have provided consistent, invaluable feedback and guidance. I cannot possibly extend the work to the Vision-Language field without their help.

Additionally, my heartfelt thanks go out to Prof. Steven H. Strauss, Prof. Yuan Jiang, Prof. Yanming Di, Prof. Daniel Curry, Prof. Tucker Hermans, Dr. Michael Nagle, Dr. Cathleen Ma, and Yixuan Huang. They not only patiently listened to numerous reports but also provided valuable guidance and insights beyond computer science during our collaborations. I am also grateful to all my exceptional colleagues, as well as all my course instructors, from whom I gained extensive knowledge through our discussions and interactions. I would like to thank all staff members in the EECS department and the OIS office of Oregon State University for their unwavering support and assistance. Your dedication was greatly appreciated throughout my academic journey.

Completing a PhD is an arduous journey, and a supportive team can truly make a world

of difference. While it's impossible to name everyone, I want to express my deep appreciation for my lab mates (in alphabetical order) Alrik, Ali, Chanho, Daman, Hung, Jay, Kaibo, Lawrence, Michael, Neale, Nicholas, Nihar, Saeed, Tim, Wenxuan, Xianfang, Xingyi, Xinyao, Zehuan, Zheng, Zhongang, and Ziwen. They patiently listened to countless presentations and practice talks, providing invaluable feedback and advice. I extend special thanks to Zheng and Lawrence Neal, who played pivotal roles during my initial onboarding at Oregon State University and helped set up the environment for my first research work. I'm also grateful to Alex, Chanho, Daman, Hung, Xinyao, Nicholas, Nihar, and Zheng, with whom I had the opportunity to collaborate closely, learning numerous valuable skills while working on joint projects.

Finally, I would like to give sincere thanks to my grandmother, parents, sister, brother, and all the family members. My family is always with me with love, care, and support that have been a constant source of strength throughout my journey. I am deeply thankful to my best friends, Yiding and Kewen, for their companionship over the past decade, particularly during my first year in the United States. I extend my appreciation to Qian, Muqi, Jessica, Vivi, and Brianne, for their companionship and the shared experiences of life.

# TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

# LIST OF FIGURES

# LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

# LIST OF TABLES

LIST OF TABLES (Continued)

# LIST OF ALGORITHMS

# Chapter 1: Introduction

*Infants and toddlers derive implicit theories to explain the actions of* ***objects*** *and the behavior of people; these theories form the foundation for causal learning and more sophisticated understanding of the physical and social worlds. [Johnson, 2010]*

## 1.1 Motivation

Observing and analyzing the physical objects and artifacts that make up our environment is an intuitive way for humans to learn about the physical world, and it is a process that begins at an early age. When humans encounter a new scene, they rapidly divide it into distinct regions, each representing a separate object, within milliseconds. Afterward, with another less than milliseconds, they focus their attention on objects that pique their interest to ascertain their identity and relationships. A computer vision system that can recognize objects in digital images or videos from the real world, acting as an artificial offset of human perception, makes it possible for businesses to solve customer needs without too many human interactions in areas such as robotics, medical imaging, autonomous vehicles, and security and surveillance.

Before the advent of deep learning, object recognition, which aims at identifying and classifying objects or patterns within an image or a visual scene, has already been a primary task and has been applied in lots of areas, such as face recognition [Jain and Li, 2011], traffic sign recognition [Stallkamp et al., 2012], handwriting recognition [Lorigo and Govindaraju, 2006], etc. It relies on traditional computer vision techniques and handcrafted features to localize or segment objects of interest in an image. However, these traditional approaches were effective to some extent because the handcrafted features had limitations in handling variations in object appearance, occlusion, and large-scale datasets. The breakthrough of deep learning since the mid-2000s brought significant improvements in task accuracy and ignited the research interest in tackling more complex challenges, such as instance segmentation (Fig. 1.1), which involves the precise

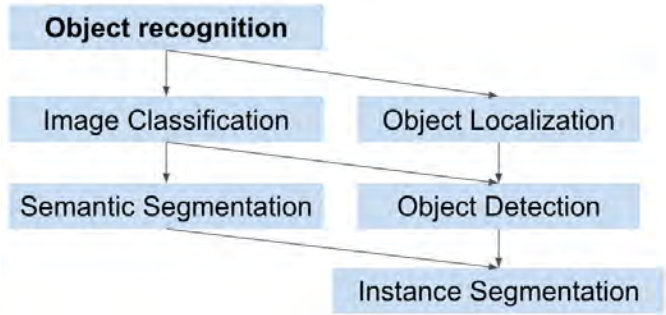



Figure 1.1: A suite of challenging computer vision tasks related to object recognition has been discussed. Prior to the deep learning era, object recognition primarily encompassed image classification and object detection [Russakovsky et al., 2015]. Recently, with the success of deep learning, more and more research interests focused on more challenging tasks, including instance segmentation.

categorization and segmentation of objects.

Motivated by the important role of objects in the cognition system, this thesis focuses on the challenge of object discovery and comprehension within digital scenes, offering efficient deep-learning solutions. The initial step is to swiftly and accurately identify objects in a static image. Once this goal is achieved, we proceed to classify these objects to decipher their semantic significance. In scenarios where the image is part of a video sequence, we extend our capabilities to track objects across frames, addressing video object segmentation. Finally, we explore to uncover intricate relationships between objects and facilitate the analysis of the semantics of the scene itself, such as identifying whether an image has been created with the intent of provoking specific groups.

The first part of this work aims to discover objects in static images, irrespective of any pre-defined categories. In this research area, many prior studies [Chen et al., 2020b, Joon Oh et al., 2017, Li et al., 2014] have dealt with discovering the salient objects in an image in the form of binary segmentation, where salient objects are typically the large central objects. In contrast, our objective is to discover all objects present in the image and separate them, regardless of their saliency. Our focus is on improving both speed and segmentation quality under the successful fully convolutional network (FCN) architecture [Long et al., 2015]. To achieve this, we introduce an innovative variational strategy that

re-frames the task as an optimization problem for piecewise-constant segmentation. This approach enables direct training of an FCN to predict object labels on each pixel. Once objects are identified, we proceed to extract each object from the image to predict its category. As a result, we obtain both object and category labels at the pixel level, effectively addressing the instance segmentation task.

Although deep learning methods have substantially improved segmentation accuracy, there remains a limitation in the model's capabilities. Segmentation results remain highly dependent on image quality, posing challenges in segmenting objects with blurred boundaries, significant occlusions, or out-of-focus, which are common in videos. However, considering the difference between detecting objects in videos from detecting them in static images, the temporal dimension, poses a chance to find clues from the video to identify objects. We observe that in the temporal neighbors of such problematic frames, where objects are observed at different timestamps, object boundaries can be clearer and occlusions are reduced. As a result, better segments can be obtained in these neighboring frames. We propose a novel methodology to improve the precision of object proposals on sampled key frames, thereby achieving superior video object segmentation throughout a sequence.

As mentioned above, humans possess the ability to discern relationships between objects in a scene and extract the semantics of the scene itself. It's important to note that human perceives the world through many channels, such as visual information gathered by the eyes or auditory information received through the ears. Humans can naturally align and fuse the information collected from multiple channels and grasp the essential concepts for a deeper understanding of the world. Inspired by the way humans perceive scenes, we explore the domain of Vision-Language (VL), which entails integrating rich inputs from the linguistic domain to reveal objects associated with the textual context, ultimately leading to a holistic scene understanding. Within this multifaceted context, we tackle the intricate challenge posed by the *content moderation* task, characterized by its semantic disparities between two domains. We introduce an innovative asymmetric fusion architecture designed to not only combine shared knowledge from both modalities but also harness their unique information. Additionally, we present a novel cross-modality contrastive loss with the purpose of capturing knowledge that uniquely emerges in multi-modal contexts. This component is critical as some harmful intent may

only be conveyed through the intersection of both modalities.

## 1.2 Research Goals and Contributions

### 1.2.1 Research Goals

The main research goals of this thesis include:

- Develop an instance segmentation algorithm designed to predict instance labels for objects within an image, that provides a detailed understanding of object boundaries and distinctions.
- Develop an algorithm aimed at enhancing segmentation quality, specifically targeting problematic frames within a video. This algorithm will be applied to various video-related tasks, including unsupervised video object segmentation (UVOS) and video instance segmentation (VIS), with the goal of improving overall segmentation performance.
- Develop a VL algorithm designed for the task of content moderation (CM), which is capable of assessing the harmfulness of content by fusing information extracted from both detected objects in an image and the accompanying sentence.

### 1.2.2 Contributions

We summarize the key contributions (comprehensive discussion in the individual chapters) as follows:

- Propose an instance segmentation algorithm that relaxes instance segmentation into a variational problem with a novel variational objective that includes a permutation-invariant component. It leads to an end-to-end training framework with an FCN directly predicting continuous instance labels on the image (Section 3).
- Propose a method to refine key frame proposals using temporal information. We reason about key frame proposals through a graph built with the object probability masks initially generated from nearby frames and then propagated to the key frame. This graph computes maximal cliques, each representing a candidate object.

Allowing multiple proposals within the clique to vote for the key frame proposals results in improved key frame proposals, potentially surpassing the quality of the single-frame proposals (Section 4).

- Present a novel modular framework that can integrate with any instance segmentation and semi-supervised VOS algorithm to address the UVSO task and VIS task (Section 4).

- Present a novel fusion transformer architecture to fuse different modalities asymmetrically, which exists in the CM task. It is designed to enhance the unique knowledge in each modality while effectively fusing the information from the asymmetric semantic levels. Based on it, we design a novel contrastive loss to squeeze out the distinct semantic that only exists in multimodality, which is critical as some harmful intent may only be conveyed through the intersection of both modalities (Section 5).

## 1.3 Thesis overview

This thesis is organized into several chapters through our efforts in achieving the research goal. Each of the main ideas corresponds to a published or submitted paper. In some sections, we extend our published results to connect related topics and show additional applications of our work.

We begin with describing prior work most relevant to this thesis in **Chapter 2**, including a survey of the literature covering the topic areas: FCN, Transformer, Vision-Language, and Maximal Clique.

In **Chapter 3**, we introduce our first work targeting an instance segmentation task. We propose *Deep Variational Instance Segmentation* (DVIS), that employs an FCN to predict continuous instance labels on the image, all within an end-to-end system. To allow the training with permutation-invariant ground truth in instance segmentation, we propose a variational relaxation of the task as minimizing an optimization function for a piecewise-constant segmentation problem. It extends the classical Mumford-Shah variational segmentation algorithm to be able to handle the permutation-invariant ground truth. Experiments on PASCAL VOC 2012 and the MSCOCO 2017 dataset show that

the DVIS efficiently tackles the instance segmentation task.

In **Chapter 4**, we explore the enhancement of object segmentation accuracy in scenarios where a single frame suffers from poor quality. Our solution *Maximal Cliques on Multi-frame Proposal Graph* (MCMPG), is a novel, lightweight framework that can seamlessly integrate an instance segmentation algorithm and a VOS algorithm, thereby enhancing the performance within UVOS/VIS tasks. MCMPG leverages maximal cliques within a graph of object proposals initially generated from adjacent frames and then propagated to key frame. By reasoning over multiple similar proposals within a maximal clique, we achieve significant improvements in object proposals. These enhancements, in turn, lead to improved overall sequence segmentation performance when coupled with a semi-supervised VOS algorithm that tracks these key frame proposals throughout the entire video. We conduct comprehensive testing on DAVIS-UVOS 2017 and YouTube-VIS 2019 datasets, employing a variety of instance segmentation methods and VOS methods. The results consistently demonstrate that our approach consistently outperforms the baseline $w/o$ MCMPG and competing methods.

In **Chapter 5**, we delve into the realm of achieving a holistic understanding of a scene by incorporating linguistic information to focus on the relevant objects. We present a cutting-edge CM model, *Asymmetric Mixed-Modal Moderation* (AM3), which addresses both multimodal and unimodal CM tasks. Unlike conventional VL models that seek to establish a unified understanding of the vision and language modalities, AM3 is strategically designed for asymmetric fusion. It features a novel asymmetric fusion architecture and introduces a pioneering cross-modality contrastive loss, both of which serve to enhance the unique information within each modality while effectively integrating information from asymmetric levels. Furthermore, we leverage unimodal image/text datasets in the pretraining phase to incorporate domain-specific knowledge and improve model performance. This approach relaxes the constraint of requiring both modalities to be present, allowing us to harness domain-specific unimodal data for training. Our model's effectiveness is substantiated through experiments conducted across 7 multimodal and unimodal CM benchmarks

Finally, we conclude in chapter 6 with insights for possible future research directions and applications based on the research presented in this thesis.

# Chapter 2: Related Work

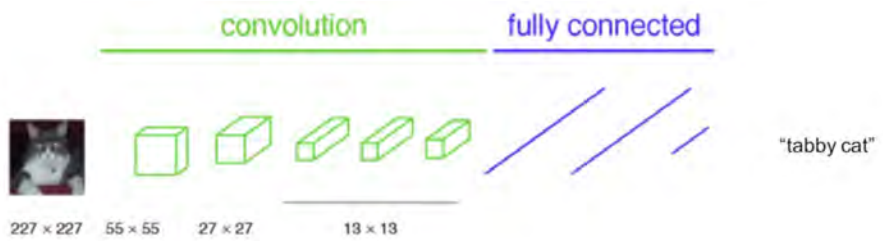
In the chapter, we introduce fields most relevant to this thesis.

## 2.1 Fully Convolutional Networks

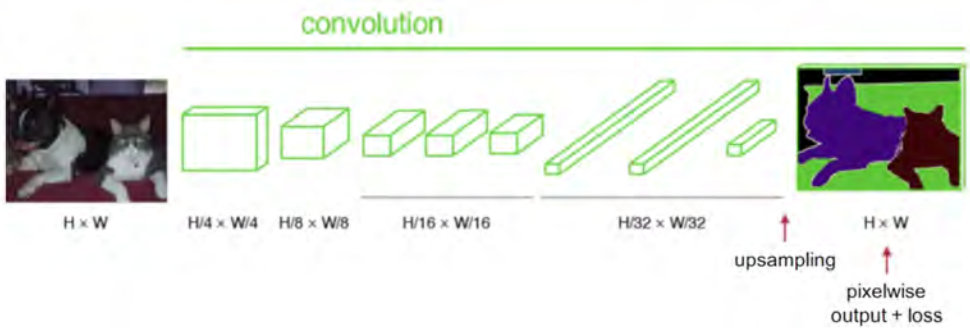
Fully Convolutional Networks (FCNs)[Chen et al., 2016, Long et al., 2015, Noh et al., 2015, Ronneberger et al., 2015] are an extension of traditional Convolutional Neural Networks (CNNs). In traditional CNNs, which consist of both convolution layers and fully connected (FC) layers, the primary application is tasks such as image classification. Fig. 2.1(a) illustrates this process: an input image is downsized and processed through the convolution layers and FC layers, ultimately outputting a single predicted label for the input image.

In contrast, FCNs replace all FC layers with convolution layers, this modification empowers FCNs to efficiently analyze entire images and perform pixel-wise classification or segmentation. Fig. 2.1(b) demonstrates this difference: the output is no longer a single label but rather a label map, typically with a smaller size than the input image due to pooling. FCNs utilize upsampling technologies to generate a pixel-wise output. To enhance output quality, various methods have been introduced. DeconvNet[Noh et al., 2015] incorporates deconvolution and unpooling layers to improve downsized features. Deeplab [Chen et al., 2016] introduces "atrous convolution" for denser feature extraction while preserving the size of the receptive field. U-Net [Ronneberger et al., 2015] extends the FCN architecture with an encoder and decoder, where the decoder upsamples the feature map followed by 2x2 convolution (*up-convolution*) that halves the number of feature channels a concatenation with the cropped feature map from the *down-convolution* path. FCNs are known for their ability to process images of arbitrary size and maintain spatial details, making them well-suited for tasks demanding precise spatial localization. They played a pivotal role in numerous computer vision applications, including semantic segmentation, instance segmentation, and even tasks that require real-time processing,

like autonomous driving and medical image analysis. In this thesis, Sections 3 and 4 build upon the FCN architecture to develop novel methods.




(a)




(b)

Figure 2.1: Visualization[1]of (a) Image classification using a CNN. (b) Semantic segmentation using an FCN.

## 2.2 Vision-Language Models: Leveraging Transformers

The transformer architecture, introduced in 2017 by Vawani et al. [Vaswani et al., 2017], has become the foundation for a wide range of state-of-the-art models in natural language processing (NLP), such as GPT [Dale, 2021, van Dis et al., 2023], BERT [Devlin et al., 2018, Tenney et al., 2019], and T5 [Roberts et al., 2020]. At its core, the transformer employs a novel self-attention mechanism, which assesses the significance of various elements within a sequence during processing. This design allows transformers to effectively capture dependencies and relationships among elements, regardless of

[1]The figures are sourced from https://towardsdatascience.com/review-fcn-semantic-segmentation-eb8c9b50d2d1

their positions within the sequence. This inherent parallelism sets transformers apart from traditional recurrent neural networks (RNNs) and convolutional neural networks (CNNs), providing them with a notable advantage. Moreover, the transformative impact of the transformer extends beyond NLP, finding applications in various domains, including Vision-Language (VL) tasks.

VL tasks represent a pioneering approach to artificial intelligence. It leverages both image and text inputs to bridge that gap between human perception and machine comprehension. In recent years, there has been a surge in the popularity of VL models designed to tackle a wide range of VL tasks. These tasks include applications like visual captioning, visual question answering, image-text retrieval, content recommendation, content generation, and more. Building on the remarkable success of transformer models in NLP, researchers naturally ventured into applying them to VL tasks. Wherein BERT has been widely adopted, leading to an explosion of Bert-based multimodal architectures [Alayrac et al., 2022, Chen et al., 2020a, Huang et al., 2020, Kim et al., 2021, Su et al., 2019]. These models are designed to process and generate information from both textual and visual inputs, enabling them to comprehend and describe the scene in a more human-like manner.

## 2.3 Maximal Cliques

A maximal clique in graph theory is a fundamental concept that plays a crucial role in understanding the connections between nodes or vertices within a graph. In essence, a maximal clique is a subset of vertices within a graph where every pair of vertices is connected by an edge, and this subset cannot be expanded further by adding an adjacent vertex without violating the clique's property.

The utility of maximal clique algorithms extends across diverse domains, including image segmentation. For instance, [Felzenszwalb and Huttenlocher, 2004] and [Ma and Latecki, 2012] have introduced graph-based segmentation methods that employ maximal cliques to identify connected components within the graph. [Achanta et al., 2012] and [Uijlings et al., 2013] leverage maximal clique-based region grouping to efficiently generate superpixels and object proposals, showcasing the versatility and effectiveness of this concept in image analysis.

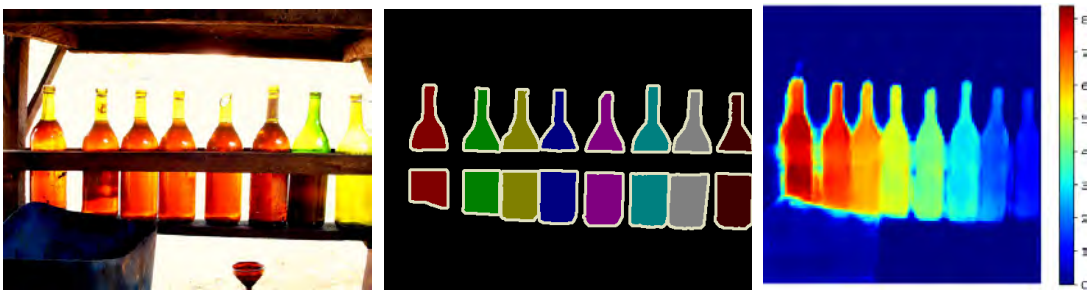
# Chapter 3: Instance Segmentation

In this chapter, we introduce a novel approach for object discovery in static images ignoring the category information. This step is pivotal in achieving a comprehensive scene understanding. After effectively segmenting each individual object, we utilize ROI Align[He et al., 2017] to extract each object and include an image classification method to predict the category of these objects. As a result, we tackle the instance segmentation task, which involves segmenting an image into distinct regions by identifying different object instances, even when they share the same semantic label.

## 3.1 Introduction

Instance segmentation is developed from semantic segmentation, which aims to classify each pixel in an image into one of several predefined object categories such as *car* or *person*. Witnessed rapid development in semantic segmentation [Chen et al., 2016, Jaderberg et al., 2015, Long et al., 2015, Noh et al., 2015], i.e., instance segmentation [Everingham et al., 2010, Hariharan et al., 2011, Lin et al., 2014] takes the task a step further by not only classifying pixels but also identifying individual object instances of the same class. It is more challenging, because (1) different instances may have similar appearances if they belong to the same category; (2) the number of instances is often unknown during prediction; and (3) labels of the instances are *permutation-invariant*, i.e., randomly permuting instance labels in the training set ground truth should not change the learning outcome (Fig. 3.1).

For such permutation-invariant instance labels, one cannot directly train the model using conventional objectives such as the cross-entropy (CE) loss. One popular strategy is to combine detection and segmentation into a two-stage approach. One network generates object proposals, while another one classifies and refines each proposal [Arnab and Torr, 2017, Chen et al., 2018a, Dai et al., 2016a, Hariharan et al., 2014, He et al., 2017, Li et al., 2016, Liu et al., 2018, Romera-Paredes and Torr, 2016, Uhrig et al., 2018]. To ensure all



(a) Input Image (b) GT (c) Real-valued Predicted Labels

Figure 3.1: (a): An example from PASCAL VOC [Everingham et al., 2010] with 8 bottles. (b) Ground truth. Labels of the bottles can be either 1 to 8 or 8 to 1. (c) Our approach solves a variational relaxation of the problem and predicts real-valued labels on the image (best in color)

instances are segmented, these methods often need to generate a significant amount of proposals ($1,000 - 3,000$ per image), and many are based on a sliding window approach that is similar to a complete search on a low-resolution image with anchor boxes. These proposals are verified with an object classifier and a smaller but still significant amount ($200 - 2,000$) is sent to the second stage for classification and refinement. To improve the efficiency, some recent works remove the anchor boxes by directly dividing the output image into a regular grid cell and segmenting the object that is centered in each cell [Chen et al., 2019, Wang et al., 2019b, 2020c, Xie et al., 2020]. However, they still require a significant amount of proposals. Another alternative solution is the search-free approach, which does not explicitly generate object proposals. Most methods learn to predict surrogates for instance labels for each pixel, and then use heuristic post-processing procedures to segment each instance [Bai and Urtasun, 2017, Kirillov et al., 2016, Liu et al., 2017, Uhrig et al., 2016, Zhang et al., 2015, 2016].

We note that the goal of instance segmentation is to generate piecewise constant predictions on each pixel that match with a given ground truth. This resonates with the classic and elegant variational principle introduced to computer vision almost three decades ago. Such variational methods, originated from the Mumford-Shah model [Mumford and Shah, 1989], parse an image into meaningful sub-regions by finding a piecewise smooth approximation. These approaches were traditionally limited to simple problems

such as image restoration and active contours, mainly because of the difficulties at that time in estimating nonlinear functions from an image. However, they could be inherently appealing in a deep network setting, since these variational objectives work with real-valued inputs and outputs. e.g., the Mumford-Shah functional, that are naturally differentiable.

We believe such variational approaches could be very powerful when combined with deep learning since they enable us to solve deep learning problems that are difficult for conventional objective functions such as cross-entropy. On the other hand, parametrizing variational approaches with a deep network enables them to model complex functions originating from an image. It also allows them to generalize to testing images. In this paper, we propose *deep variational instance segmentation* (DVIS), which is a fully convolutional neural network (FCN) that directly predicts instance labels – a piecewise-constant function, with each constant sub-region corresponding to a different instance. A novel variational objective is proposed to accommodate the permutation-invariant nature of the ground truth in instance segmentation, which leads to end-to-end training of the network.

With this proposed approach, we are directly gazing at instances from a top-down FCN viewpoint without the need to generate bounding box proposals using search protocols. Our approach outperforms the other search-free instance segmentation methods on the PASCAL VOC dataset [Everingham et al., 2010, Hariharan et al., 2011] and it is the first search-free method tested on the MS-COCO dataset [Lin et al., 2014], obtaining a performance close to these search-based methods, but with significantly faster speed.

## 3.2 Related Work

Instance segmentation identifies every single instance at the pixel level. We group the approaches tackling the task as search-based and search-free methods. Most search-based approaches are anchor-based, they break the task into two cascaded sub-tasks: the first one generates region proposals with carefully designed anchor boxes, e.g., with a region proposal network (RPN) [Ren et al., 2015]. Another network classifies and refines each proposal. This architecture solves the counting problem by adopting non-

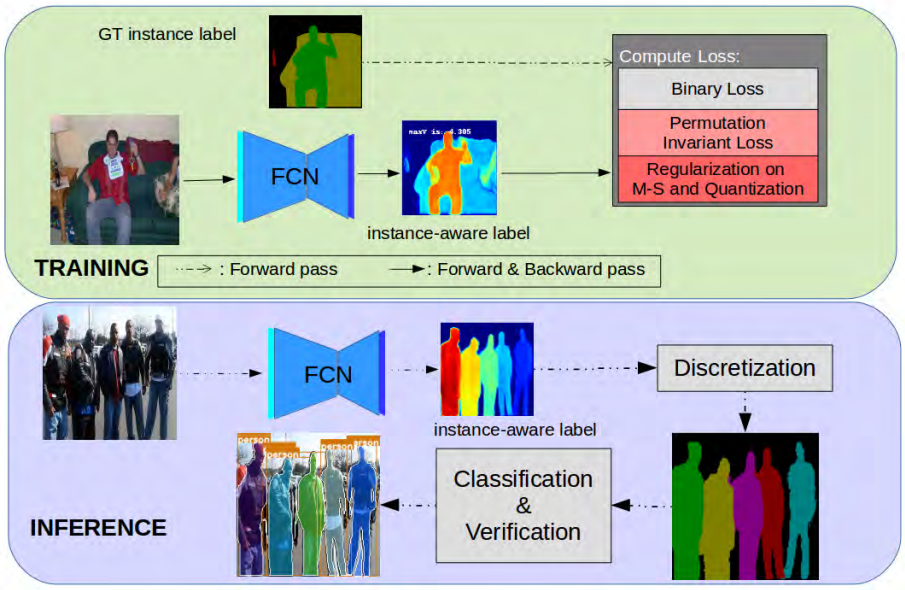



Figure 3.2: The proposed deep variational instance segmentation (DVIS): An FCN is trained to directly output real-valued instance labels, using a novel variational framework we proposed that combines a binary loss function, a permutation-invariant loss function, and regularization terms. During inference, we discretize the predicted instance map into several instances. After classification and verification, we output the final segmentation with both semantic and instance labels (best viewed in color)

maximum suppression (NMS) [Dai et al., 2016b, He et al., 2017, Huang et al., 2019, Liu et al., 2016, Redmon and Farhadi, 2018, Ren et al., 2015] or determinant point processes (DPP) [Azadi et al., 2017, Lee et al., 2016] to remove overlapping detections. Besides RPN, [Uijlings et al., 2013] uses selective search to generate proposals, [Pont-Tuset et al., 2017a] uses a network to generate region proposals in the form of a binary mask. However, such a search-base process is inherently slow, as many different proposals with various sizes and aspect ratios need to be generated and scored, which might be unacceptable in realistic application scenarios where engineers are striving to obtain real-time performance. [Chen et al., 2018a, Liu et al., 2018, Uhrig et al., 2018] integrate instance-related features into the second stage in the anchor-based architecture. The global context information encoded in these features can help refine the final segmentation. Recently, [Bolya et al., 2019a,b] proposed to use a network to learn mask prototypes

from the input image and combine these prototypes to generate the final mask for each detected instance. But they still search with anchor boxes of different scales and shapes hence generating significantly more proposals than ours. To reduce the redundancy of the anchor boxes, [Chen et al., 2019, Wang et al., 2019b, 2020c, Xie et al., 2020] directly predict instance mask centered on each pixel in the output image. Instances that might share the same centers are predicted at different scales from the FPN network [Lin et al., 2017].

We focus our literature review more on search-free methods that are directly relevant to our work. Some search-free approaches focus on exploring instance-aware and learning them using an FCN. [Bai and Urtasun, 2017, Ren and Zemel, 2016, Romera-Paredes and Torr, 2016] predict the energy of the watershed transform, [Uhrig et al., 2016] predicts the direction on each pixel to the object center, [Kirillov et al., 2016] predicts instance-level boundary score, and [Liu et al., 2017] attempts to locate instance segment breakpoints to separate each instance. However, these approaches do not directly generate an instance prediction and hence need to resort to a significant amount of heuristic post-processing such as template matching [Uhrig et al., 2016], MultiCut[Kirillov et al., 2016] or recurrent neural network[Ren and Zemel, 2016, Romera-Paredes and Torr, 2016].

[Fathi et al., 2017, Kong and Fowlkes, 2018] are search-free approaches based on the metric learning idea. [Kong and Fowlkes, 2018] learns to map pixels to a multi-dimensional embedding space using pairwise associative loss. [Fathi et al., 2017] formulates it using metric learning. The network is trained to enforce pixels from the same instance to be close to each other while pixels from different instances are far away in the learned feature space. These approaches have not employed binary terms as in ours. Hence, in the embedding space generated by these methods, the background (stuff categories such as water, grass, etc.) is no different than "yet another instance" and the separation between foreground and background is usually weak, hence these methods require more post-processing and depend on semantic segmentation to distinguish background and foreground, our foreground/background binary term directly suppresses output on the background pixels and outputs a cleaner instance map.

## 3.3 DVIS

### 3.3.1 The Mumford-Shah Model

The Mumford-Shah model is an energy-based model introduced in 1989 [Mumford and Shah, 1989] for image segmentation. It relaxes the task to a continuous energy minimization problem that computes the optimal piecewise-smooth approximation of a given image. Let $I$ denote an observed image on a bounded domain $\Omega \subset \mathcal{R}^2$ to be segmented. We define $\hat{I}$ as an approximation of $I$ and $C \subset \Omega$, the set of edges delineating the boundaries of different objects. the Mumford-Shah functional is:

$$F(\hat{I}, C) = \int_{\Omega} (\hat{I}(x,y) - I(x,y))^2 dxdy + \mu \int_{\Omega \backslash C} |\nabla \hat{I}|^2 dxdy + \nu |C|, \tag{3.1}$$

where $\mu, \nu$ are non-negative parameters, $\Omega \backslash C$ is the set of non-edge pixels, $|C|$ is the number of pixels in $C$. Minimizing the above functional essentially seeks to optimize for a piecewise smooth function (ideally constant inside each segment) which may be non-smooth on the edges/boundaries. The first term drives $\hat{I}$ to be close to $I$. The second term imposes smoothness prior inside each segment $\Omega \backslash C$ and protects from under-segmentation. The last term encourages shorter object contours to avoid over-segmentation. By adjusting the parameters $\mu, \nu$, it can optimally segment the given image.

The Mumford-Shah functional was well-regarded as a solid variational model that has been analyzed aplenty [Chan et al., 2006, Grady and Alvino, 2008, Pock et al., 2009, Strekalovskiy and Cremers, 2014, Vese and Chan, 2002, Xu et al., 2011]. It appropriately regularizes on the length of object boundaries while capable of modeling multiple objects within the same image. However, because the first term is usually only enforcing the approximation to be close to the input image function, it was traditionally only utilized in superpixel segmentation and active contours [Morar et al., 2012, Vese and Chan, 2002].

**From unsupervised to supervised setting.** We note the similarity between the unsupervised Mumford-Shah model and the supervised instance segmentation problem. Both optimize for a piecewise-constant function, where each piece corresponds to one

object instance and the number of pieces in the image is unknown. Both enforce constancy within each piece and a short boundary length would also be an ideal prior for instance segmentation, albeit to our knowledge we have never previously seen an approach that incorporates that. The second term in the MS-model is a common pairwise term that enforces piecewise-constancy, similar to those used in metric-learning-based instance segmentation methods [Fathi et al., 2017, Kong and Fowlkes, 2018]. Previous work [Strekalovskiy and Cremers, 2014, Xu et al., 2011] have shown that the second and third terms can be combined as a robust loss on the pairwise term (see Sec. 3.3.3 for more details).

The main difficulty of extending this variational approach to solve the instance segmentation problem lies in utilizing the matching potential $\int(\hat{I}(x,y) - I(x,y))^2 dxdy$, where a simple MSE or CE loss would not suffice for instance segmentation because of the permutation-invariance of ground truth labels. However, there is one ground truth label that remains the same throughout the whole dataset: the background label. Thus, a new variational formulation is needed. In the next subsection, we propose a novel variational formulation that solves the instance segmentation problem.

### 3.3.2 Deep Variational Instance Segmentation

As discussed above, we relax the supervised instance segmentation to a continuous energy minimization problem. We first note that the ground truth label $GT$ in instance segmentation usually has two distinct aspects: 1) when the label of a pixel is 0, then the pixel is background; 2) when the label of a pixel is larger than 0, then the label is *permutation-invariant*, i.e. one can switch labels of different objects (e.g. between object 3 and 5) without affecting their actual meaning. Hence, when defining a variational functional for instance segmentation, both of these components need to be considered.

We define a variational functional for instance segmentation as:

$$F(f,C) = \underbrace{\int_\Omega \mathcal{L}_b\left(f(x,y), \mathbb{I}_{[GT(x,y)=0]}\right) dxdy}_{\text{Binary Loss}} + \underbrace{\mu \int_\Omega \|\nabla f\|^2 dxdy + \nu|C|}_{\text{Regularization}} + \underbrace{\int_\Omega |f - Round(f)| dxdy}_{\text{Quantization}}$$
$$+ \underbrace{\int_\Omega \int_\Omega \mathcal{L}_{pi}\left(|f(x_1,y_1) - f(x_2,y_2)|, \mathbb{I}_{[GT(x_1,y_1)\neq GT(x_2,y_2)]}\right) dx_1 dy_1 dx_2 dy_2}_{\text{Permutation Invariant Loss}} \tag{3.2}$$

where $f$ denotes the continuous-valued label map predicted by our network, an FCN with parameters $\omega$. $Round(\cdot)$ is the operation rounding to the nearest integer. $\mathcal{L}_b$ compares the instance label with the binarized ground truth label that indicates object/background and $\mathcal{L}_{pi}$ denotes the *permutation-invariant* loss function which compares the difference between two-pixel labels $|f(x_1,y_1) - f(x_2,y_2)|$ with $\mathbb{I}_{[GT(x_1,y_1)\neq GT(x_2,y_2)]}$, which indicates whether the ground truth labels at these pixels are different. Using $\mathcal{L}_{pi}$, the exact values of the ground truth labels no longer play a role in the loss. The smoothness and minimal edge length terms are the same as in Mumford-Shah. We incorporate an additional quantization term, which drives the output label value to be closer to integers.

Training on this variational functional enables us to learn $f$ from a training set with instance-level ground truth and generalize it onto unseen testing images. This improves over traditional variational segmentation which does not have learning capabilities. Note that in our permutation-invariant loss $\mathcal{L}_{pi}$, we would in principle integrate over *all* pixel pairs within the image that are not boundaries, instead of only in a small neighborhood as in the traditional conditional random field (CRF) approaches. This is because instance segmentation is an inherently non-local problem: due to occlusion the same instance can be separated into several pieces in 2D that are possibly very far away from each other, hence, only local consistency is not enough. Empirically we have also found that if we only enforce local consistency, we may have small, smooth changes in the predicted instance labels $f$ that could add up to a significant amount and lead to changing instance labels within the same instance.

In practice, we discretize $\mathcal{L}_b$ on all the pixels, and discretize the integral $\mathcal{L}_{pi}$ on sampled pixel pairs. Either stratified sampling or random sampling of pixel pairs can be used. In stratified sampling, we sample all the immediate neighbors in the 4-neighborhood of a

pixel and reduce the sampling density for further away pixel pairs. In random sampling, we randomly select pixel pairs across the whole image for computing the integral on $\mathcal{L}_{pi}$. We have found that on smaller resolutions, stratified sampling is efficient whereas when resolutions are very large, random sampling is more efficient.

Also note that there is a significant difference between variational approaches such as ours and CRF approaches, although both employ matching (unary) and regularization (pairwise) terms. In CRFs, the labels come from a discrete set, while in variational approaches the labels are relaxed to be continuous themselves. It is difficult for a CNN to simulate the full CRF inference process and one would have to resort to a recurrent network [Zheng et al., 2015], increasing the complexity of the model. On the other hand, our variational formulation eq.(3.2) would only require an FCN to simultaneously handle images with an undetermined amount of objects, since it predicts labels as continuous real-valued numbers. since it predicts labels as continuous real-valued numbers.

### 3.3.3 Loss Functions

As a variational approach, our output $f$ values are continuous. Hence, loss functions would be more similar to regression loss functions. Here we mostly utilize variants of the robust Huber loss function $L_h(v, \theta) = \frac{v^2}{2\theta}$ if $v < \theta$ and $v - \frac{\theta}{2}$ otherwise. We set $\theta = 0.1$ throughout the work.

**Binary Loss:** Our first $\mathcal{L}_b$ seeks to separate a labeled instance from "stuff" classes such as road, water, sky, etc. which would not have individual instances in them and are usually labeled as background in instance segmentation tasks. Thus, $\mathcal{L}_b$ drives segmentation to be non-positive in background pixels and sufficiently positive in foreground pixels. Let $GT(x, y) = 0$ on the background pixels and $GT(x, y) > 0$ on the foreground pixels, the binary loss is computed as:

$$\mathcal{L}_b(f(x,y), GT(x,y)) = \begin{cases} L_h(ReLU(f(x,y))) & \text{if } GT(x,y) = 0 \\ L_h(ReLU(m_1 - f(x,y))) & \text{if } GT(x,y) > 0 \end{cases} \tag{3.3}$$

where $ReLU(x) = \max(x, 0)$ is the commonly used ReLU activation function, and $m_1$ is a parameter of the loss function to separate foreground from background. With this loss, on

foreground pixels, when $f(x, y) \geq m_1$, the loss will be 0, this accommodates foreground objects taking different $f(x, y)$ values. On background pixels, once $f(x, y) \leq 0$, the loss will be 0. In experiments, we set $m_1 = 2$. We formulate the term as regression with the robust Huber loss, instead of as binary classification with the CE loss. This is because the regression loss can obtain exactly 0 when the label value $\geq m_1$ in the foreground and $\leq 0$ in the background, whereas the CE loss tends to push to positive/negative infinity.

**Permutation Invariant Loss:** We use $\mathcal{L}_{pi}$ to enforce similarity between ground truth instance labels and predicted instance labels, taking into account that the ground truth labels are permutation-invariant. Let $p_1$ and $p_2$ be two pixels from a neighborhood and their ground truth as $GT_{p_1}, GT_{p_2}$, respectively, the relative loss is computed by:

$$f_d = |ReLU(f(x_1, y_1)) - ReLU(f(x_2, y_2))| \tag{3.4}$$

$$\mathcal{L}_{pi}\left(f_d, GT(x_1, y_1), GT(x_2, y_2))\right) = \begin{cases} L_h(f_d), & \text{if } GT(x_1, y_1) = GT(x_2, y_2) \\ L_h(m_2 - f_d), & \text{if } GT(x_1, y_1) \neq GT(x_2, y_2) \end{cases} \tag{3.5}$$

where $m_2$ is a parameter used to adjust the margin between predicted labels from different instances. We set $m_2 = 1$ in practice. Hence, there is no loss if the difference between predicted labels on two pixels is more than 1, which indicates that the two pixels belong to different instances. On the other hand, if the two pixels belong to the same instance, the loss is 0 only when their predicted labels are the same.

**Regularization**: Mumford-Shah regularization is helpful for obtaining sharper boundaries. We have noticed that without such regularization the predicted label map tends to change more smoothly at object boundaries, creating intermediate values that do not belong to any object which makes post-processing more difficult. There has been a significant amount of work on optimizing the Mumford-Shah term. We follow [Strekalovskiy and Cremers, 2014] to discretize Mumford-Shah as a robust loss function:

$$\mathcal{L}_{MS}(f(x, y)) = \min(\mu \|\nabla f(x, y)\|^2, \nu) \tag{3.6}$$

which is equivalent to the original Mumford-Shah formulation. [Strekalovskiy and Cremers, 2014] then solves the formulation using a primal-dual algorithm, but in our case,

we do not need to exactly solve the optimization problem since optimization is never exact with a deep network. Hence we just use a simple quasi-convex robust loss function as in the Cauchy loss:

$$\mathcal{L}'_{MS}(f(x,y)) = \log\left((f(x,y) - f(x,y+1))^2 + (f(x,y) - f(x+1,y))^2 + 1\right) \tag{3.7}$$

Note one way to approach proper Mumford-Shah regularization is to anneal the loss gradually towards a Welsch loss function as in [Barron, 2019], which we did not do because the difference is very minor.

Finally, the quantization term minimizes the distance between the output label and its nearest integer. The gradient of this term is back-propagated from the first $f$. Since the operation $round(\cdot)$ is piecewise-constant, its gradient is 0). This term helps to create a sufficient margin between different label values, making post-processing easier.

In summary, we relax a supervised instance segmentation to a deep variational minimization problem. With our formulation, the proposed variational problem can be tackled by training an FCN to optimize these loss functions and output the real-valued approximation of instance segmentation labels. Through directly optimizing on instance segmentation, our proposed approach has the advantage of generating different labels for different objects while having the capability of capturing multiple scattered parts, e.g. of an occlude sofa as a single object (Fig. 3.2).

## 3.4 Experiments

### 3.4.1 Implementation Details

**FCN for Instance Segmentation:** An encoder-decoder FCN network is adapted to solve instance segmentation with our variational loss. We employ *ResNet-50* and *ResNet-101* with output stride 8 as our base network and its output is then upsampled by 2 using a decoder network similar to the upsampling branch in FPN[Lin et al., 2017] to generate higher resolution output. The last layer of the FCN network outputs the real-valued label map as one output channel, which is then used to compute our variational loss eq. (3.2) and backpropagation. We remove negative label outputs by adding a *ReLU*

*activation* on the FCN output. Note we did not employ multiple output heads as in FPN.

**Training:** We scale the input image to 513×513 for PASCAL and with the minimum edge equal to 700 for COCO (preserving the height-to-width ratio). The window size for computing relative loss is set to 128 throughout all experiments. We initialize the backbone network with the pre-trained weights for the semantic segmentation task on PASCAL and the pre-trained weights for the object detection task on COCO.

**Permutation-Invariant Loss:** Given an input image in size $H \times W$ and the FCN with a downsampling factor $d$, the output size would be $\frac{H}{d} \times \frac{W}{d} \times 1$. The number of pixel pairs is a huge number $\frac{HW}{d^2} \times \frac{HW}{d^2}$. In our model, with the binary loss to separate background and foreground, it suffices to only consider the pixel pairs located on instances, which reduces the number of pixel pairs that need to be computed. Then we utilize the stratified sampling to sample pairs to compute the permutation-invariant loss. Given a pixel $(x, y)$ and the window size $w$, we sampled all pixels inside the center area with distance $c(c < r)$ and we selected the rest pixels with a dilation rate of 'r', similar to dilated convolutions [Chen et al., 2016]. The base setting we use is $w = 129, c = 8, r = 8$.

**Discretization to instance segmentation:** After we obtain the real-valued instance labels, we apply the mean-shift segmentation algorithm on it with different bandwidths, 0.9 and 0.4 to discretize it to two different label maps. Because $m_2$ is fixed to 1, the bandwidth of 0.9 works well to separate objects the network believe is different. And when the network does not learn to separate the instances well enough, bandwidth 0.4 helps to segment the objects. these two bandwidth proves to be enough to generate all instance segments, which are then verified in the next module.

**Classification and Verification:** We utilize a classification network to verify the segments. It first takes CNN features from the bounding box of each predicted instance from the FCN with ROIAlign [He et al., 2017], and concatenates it with the predicted binary mask for the instance. We then run a small convolutional network with 7 layers that will classify each predicted instance into the pre-defined semantic categories. Besides, we have an IoU head [Huang et al., 2019] that attempts to predict the Intersection-Over-Union between the predicted instance with the ground truth instance that best matches

it, using a Huber regression loss. Finally, we reject false positive instances by thresholding the weighted sum of predicted confidences on the semantic classification and the predicted IOU. Note that we are only verifying on average $5 - 15$ segments per image, which is significantly less than previous approaches (Table 3.6), hence the overhead of this stage is very small (Table 3.5). Hence, this classification step does not impact our speed advantage over search-based methods.

### 3.4.2 Datasets

We evaluate the proposed approach for instance segmentation on the challenging PASCAL VOC dataset [[Everingham et al., 2010]] on the *val* split and the SBD split [[Hariharan et al., 2011]], as well as the COCO dataset [[Lin et al., 2014]].

**PASCAL VOC 2012** consists of 20 object classes and one background class. It has been the benchmark challenge for segmentation over the years. The original dataset contains 1,464, 1,449, and 1,456 images for training, validation, and testing. It is augmented by extra annotation from [[Hariharan et al., 2011]], resulting in 10,582 training images. The metric we use to evaluate PASCAL is average precision (AP) with pixel intersection-over-union (IoU) thresholds at 0.5, 0.6, 0.7, 0.8, and 0.9 averaged across the 20 object classes. As there is no ground truth on the testing set, we use the *val* set to test.

**PASCAL SBD** is a different split on the PASCAL VOC dataset. In order to compare with [[Bolya et al., 2019a, Li et al., 2016]], we train a separate model on SBD's training set and evaluate its 5,732 validation images.

**COCO** is a very challenging dataset for instance segmentation and object detection. It has 115,000 images and 5,000 images for training and validation, respectively. 20,000 images are used as *test-dev* from the split of 2017. There are 80 instance classes for instance segmentation and object detection challenges. There are more objects in each image than in PASCAL VOC. We train our model on the *train 2017* subset and run prediction on *val 2017* and *test-dev 2017* subsets respectively. We adopt the public *cocoapi* to report the performance metrics $AP$, $AP_{50}$, $AP_{75}$, $AP_S$, $AP_M$, and $AP_L$.

### 3.4.3 Comparison to the state-of-the-art

**Results on PASCAL VOC and SBD** are shown in Table 3.1 and Table 3.2 respectively. Our approach significantly outperforms search-free approaches SGN and Embedding [Kong and Fowlkes, 2018, Liu et al., 2017] on all mAP thresholds. The latter two are state-of-the-art metric learning approaches. Besides, on the SBD dataset, we also outperformed well-regarded anchor-based approaches DIN and FCIS [Arnab and Torr, 2017, Li et al., 2016] significantly (Table 3.2). The recent YOLACT [Bolya et al., 2019a] achieved slightly better results than ours on mAP at 50% IoU, however, our approach is significantly better than it at 70% IoU, which requires more precise segmentation of each object. We note that 50% IoU is a quite low standard for segmentation since there can still be a significant amount of segmentation errors at this threshold. Our better performance at a higher threshold shows that our variational approach is capable of segmenting objects more precisely, especially on objects of non-rectangular shapes. Some proposal-free approaches such as DWT take each connected component as an instance, hence they do not work well for many PASCAL VOC objects which are separated into several parts with occlusions. We significantly outperformed SGN which is known to be superior to DWT. Qualitative results are shown in section 3.5.

Table 3.1: $AP^r$ result on the PASCAL VOC 2012 *val.* set.

| Method | backbone | architecture | $mAP^r$ | | | | | $AP^r_{avg}$ |
|---|---|---|---|---|---|---|---|---|
| | | | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | |
| DIN[Arnab and Torr, 2017] | PSPNet(Resnet-101) | anchor-based | 61.7 | 55.5 | 48.6 | 39.5 | 25.1 | 46.1 |
| SGN[Liu et al., 2017] | PSPNet(Resnet-101) | search-free | 61.4 | 55.9 | 49.9 | 42.1 | 26.9 | 47.2 |
| DML[Fathi et al., 2017] | DeepLab-v2(Resnet-101) | | 62.1 | 53.3 | 41.5 | - | - | - |
| Embedding[Kong and Fowlkes, 2018] | DeepLab-v3(Resnet-101) | | 64.5 | - | - | - | - | - |
| DVIS | Resnet-50-FCN | | 68.4 | 63.3 | 58.1 | 49.1 | 33.7 | 54.5 |
| DVIS | DeepLab-v3(Xception 65) | | **70.3** | **68.0** | **60.2** | **50.6** | **33.7** | **56.6** |

Table 3.2: $AP^r$ result on the PASCAL SBD *val.* set.

| Method | backbone | architecture | $mAP^r$ | | | | | $AP^r_{avg}$ |
|---|---|---|---|---|---|---|---|---|
| | | | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | |
| DIN [Arnab and Torr, 2017] | PSPNet(Resnet-101) | anchor-based | 62.0 | - | 44.8 | - | - | - |
| FCIS[Li et al., 2016] | Resnet-101-C5 | | 65.7 | - | 52.1 | - | - | - |
| YOLACT[Bolya et al., 2019a] | Resnet-50-FPN | search-based | **72.3** | | 56.2 | | | |
| DVIS | Resnet-50-FCN | search-free | 70.0 | 67.0 | 61.0 | 49.1 | 27.8 | 55.0 |
| DVIS | DeepLab-v3(Xception 65) | | *70.5* | **68.5** | **62.9** | **55.2** | **34.5** | **58.3** |

**Results on COCO** are shown in Table 3.3 and Table 3.4. One can see that with a search-free algorithm, we obtain performances very close to the two-stage mask R-CNN,

trailing mainly on small objects, where a complete search over all pixels would understandably help. We outperform the state-of-the-art anchor-based approach YOLACT on AP with multiple settings on both the *val-2017* and *test-dev 2017* datasets. YOLACT-700 results are only available on *test-dev* hence we compare with YOLACT-550 on *val*. The authors have a more recent improvement, YOLACT++ where they used deformable convolutions which is orthogonal to our contributions and could be applied in our case to further improve performance. Moreover, in Table 3.4, speed analysis on a V100 GPU (all post-processing included) is shown in the column *FPS*. Our method runs faster than all other baselines under the same backbone. The ResNet50 DVIS runs at 38.0 $fps$ and has $AP = 32.6\%$. Qualitative results are shown in section 3.5.

Table 3.3: $AP^r$ result on COCO's *val 2017* set

| Method | backbone | architecture | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ | FPS |
|---|---|---|---|---|---|---|---|---|---|
| PANet[Liu et al., 2018] | Resnet-101-FPN | anchor-based | **37.6** | 59.1 | 40.6 | 20.3 | 41.3 | 53.8 | - |
| Mask R-CNN[Chen et al., 2019] | Resnet-101-FPN | | 36.5 | 58.1 | 39.1 | 18.4 | 40.2 | 50.4 | 11.1 |
| YOLACT-550[Bolya et al., 2019a] | Resnet-50-FPN | | 30.0 | - | - | - | - | - | 44.9 |
| SOLO-800[Wang et al., 2019b] | Resnet-50-FPN | search-based | 36.0 | 57.5 | 38.0 | - | - | - | 12.1 |
| SOLO-800[Wang et al., 2019b] | Resnet-101-FPN | | - | - | - | - | - | - | 10.4 |
| PolarMask-800[Xie et al., 2020] | Resnet-101-FPN | | 29.1 | 49.5 | 29.7 | - | - | - | 12.3 |
| DVIS-700 | Resnet-50-FCN | search-free | 32.6 | 53.4 | 35.0 | 13.1 | 34.8 | 48.1 | 38.0 |
| DVIS-700 | Resnet-101-FCN | | 35.7 | **58.0** | **37.5** | **14.7** | **38.6** | **50.6** | 30.4 |

Table 3.4: $AP^r$ result on COCO's *test-dev 2017* set

| Method | backbone | architecture | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ | FPS |
|---|---|---|---|---|---|---|---|---|---|
| PANet[Liu et al., 2018] | Resnet-50-FPN | anchor-based | 36.6 | 58.0 | 39.3 | 16.3 | 38.1 | 53.1 | - |
| FCIS[Li et al., 2016] | Resnet-101-C5 | | 29.5 | 51.5 | 30.2 | 8.0 | 31.0 | 49.7 | 9.5 |
| Mask R-CNN[He et al., 2017] | Resnet-101-FPN | | 35.7 | 58.0 | 37.8 | 15.5 | 38.1 | 52.4 | 13.5 |
| YOLACT-700[Bolya et al., 2019a] | Resnet-101-FPN | | 31.2 | 50.6 | 32.8 | **12.1** | 33.3 | 47.1 | 28.7 |
| SOLO-800[Wang et al., 2019b] | Resnet-50-FPN | search-based | 36.8 | 58.6 | 39.0 | 15.9 | 39.5 | 52.1 | 12.1 |
| SOLO-800[Wang et al., 2019b] | Resnet-101-FPN | | 37.8 | 59.5 | 40.4 | 16.4 | 40.6 | 54.2 | 10.4 |
| PolarMask-800[Xie et al., 2020] | Resnet-101-FPN | | 32.1 | 53.7 | 33.1 | 14.7 | 33.8 | 45.3 | 12.3 |
| DVIS-700 | Resnet-50-FCN | search-free | 30.3 | 48.6 | 33.0 | 11.0 | 33.2 | 46.1 | 38.0 |
| DVIS-700 | Resnet-101-FCN | | **32.9** | **52.6** | **34.6** | **12.5** | **36.7** | **48.1** | 30.4 |

### 3.4.4 Ablation Study

**Inference cost.** We report the total number of float point operations (FLOPs) needed to compute instance segmentation with our approach compared with the state-of-the-art on the COCO *val2017* set. Table 3.5 shows that our model requires significantly less

computation than YOLACT[Bolya et al., 2019a], the state-of-the-art in inference speed, due to the fact that we have much fewer segments to work on (see also next paragraph and Table 3.6). We also present breakdowns of DVIS timings, where it can be seen that the majority of our computation is within the FCN network itself. Besides the network, the mean shift grouping and the classification module together only require about an extra 2% in terms of FLOPs.

Table 3.5: Number of FLOPs on the COCO *val 2017* set

| Method | backbone | image size | |
|---|---|---|---|
| | | 550 | 700 |
| YOLACT[Bolya et al., 2019a] | Resnet-50-FPN | 61.59 G | 98.89 G |
| YOLACT[Bolya et al., 2019a] | Resnet-101-FPN | 86.05 G | 137.70 G |
| DVIS | Resnet-50-FCN | 38.49 G | 60.94 G |
| DVIS | Resnet-101-FCN | 66.24 G | 106.35 G |
| Breakdown for Postprocessing time on DVIS (ResNet-101) | | | |
| Mean Shift Grouping | - | 94.79 M | 124.42 M |
| Classification Module | Resnet-101-FCN | 1.54 G | 2.44 G |

Table 3.6: Number of candidates inputted to post-processing

| Method | No. |
|---|---|
| FCIS[Li et al., 2016] | 2,000 |
| PANet[Liu et al., 2018] | 1,000 |
| Mask R-CNN[He et al., 2017] | 1,000 |
| YOLACT[Bolya et al., 2019a] | 200 |
| SOLO[Wang et al., 2019b] | 500 |
| PolarMask[Xie et al., 2020] | 3000 |
| DVIS@ PASCAL VOC | 4.15 |
| DVIS@ COCO | 14.83 |

**Number of Candidates in Post-Processing.** We compare the average number of candidates from our discretization process with previous one or two-stage instance segmentation algorithms in Table 3.6. All the search-based (even anchor-free) algorithms [Li et al., 2016, Liu et al., 2018, Xie et al., 2020] send over 200 proposals to their second stage. SOLO [Wang et al., 2019b] selects top-500 and YOLACT [Bolya et al., 2019a] selects top-200 proposals for post-processing. Meanwhile, we only average about $5 - 15$ segments per image sent to the classification module, further illustrating that our search-free FCN network has already precisely located the instances, thanks to the variational framework.

**How many labels can DVIS predict?** We investigate an interesting question, which is how many distinct objects can our framework predict. With multiple objects in the scene, the network has to be able to "see" all the objects, in order to assign them different values. Fig. 3.3 shows the number of candidate segments inputted to post-processing on the PASCAL VOC and MS-COCO dataset, which showed that our number of candidates is usually slightly higher than the number of objects. This showed that DVIS could detect enough objects for each image, and also did not generate an overabundance of candidate

segments.

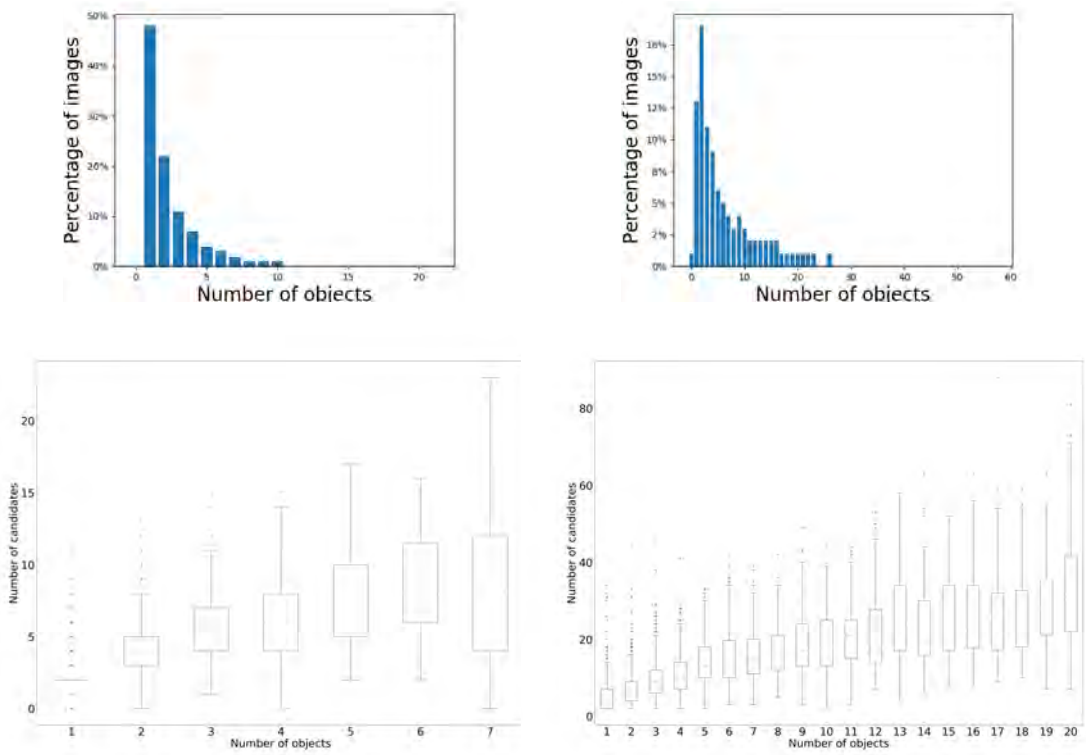



Figure 3.3: Number of Objects DVIS predicted vs. number of objects in the image on Pascal VOC(the left column) and COCO (the right column). The figures are (from top to bottom): histogram of the number of ground truth objects in the dataset and the number of discretized instances over the number of GT objects. Note that by using 2 sets of thresholds we are capable of detecting more objects than the maximal prediction value. And the number of candidate segments is only slightly more than the number of objects in the images

**Window size for computing relative loss** We show an ablation study to verify that it is indeed necessary for the permutation-invariant loss to compare pixel labels with a large spatial displacement. The ablation study is done on the PASCAL VOC dataset. We compared results where we limit the permutation-invariant loss to pixel pairs that are close by, with ranges of 8, 16, 32, 64, and 128 pixels tested respectively. Table 3.7 shows that a large window size significantly improves our performance.

Table 3.7: $AP^r$ result on PASCAL VOC val. set for different window size taken for the permutation-invariant loss

| Method | $mAP^r$ | | | | | $AP^r_{avg}$ |
|---|---|---|---|---|---|---|
| | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | |
| range 8 | 63.98 | 57.74 | 50.54 | 36.48 | 14.23 | 44.59 |
| range 16 | 63.38 | 57.55 | 49.72 | 37.49 | 14.09 | 44.45 |
| range 32 | 65.4 | 59.7 | 51.4 | 39.8 | 15.7 | 46.4 |
| range 64 | 68.21 | 62.82 | 56.73 | 49.34 | 33.5 | 54.1 |
| range 128 | **70.3** | **68.0** | **60.2** | **50.6** | **33.7** | **56.6** |

**Regularization and Quantization** Since the Mumford-Shah regularization term and the quantization term mostly work on improving the boundaries, their impact on the interior of the object is relatively small. Unfortunately, the commonly used IoU metric is almost exclusively focused on the interior and ignores small differences in the boundaries. Hence to illustrate the use of the MS-regularization, we compute the F1-measure, a semantic contour-based score from [Csurka et al., 2013], to depict the effect of the Mumford-Shah regularization.

$$
\begin{aligned}
P_i^c &= \frac{1}{C} \sum_{c=1\sim C} \frac{1}{M} \sum_{k=1\sim M} [d(z_{i,k}, GT_i^c) < \theta] \\
R_i^c &= \frac{1}{C} \sum_{c=1\sim C} \frac{1}{M} \sum_{k=1\sim M} [d(z_{i,k}, GT_i^c) \geq \theta] \\
F_1 &= \frac{1}{N} \sum_{i=1\sim N} \frac{2 \cdot P_i^c \cdot R_i^c}{R_i^c + P_i^c}
\end{aligned}
$$

Where $i, c, m$ indicates the $m$-$th$ object in image $i$ with class $c$. $\theta$ is the distance error tolerance. The $[\cdot]$ is the Iversons bracket notation. $M$ is the number of objects with class $c$ in image $i$. $C$ is the total number of supported categories. $N$ is the number of images.

From Table 3.8, the model trained with $\mathcal{L}_{MS}$ is 2% better than the model w/o $\mathcal{L}_{MS}$ at 1 distance error tolerance, which shows it improves significantly performance near the boundary. The model trained with adding quantization has equivalent performance to the model without it and it has a higher score with larger distance error tolerance

since this term can increase the margin between different instances and the detected instances are better shaped. Fig. 3.4 shows some visual examples, the predicted instance map is smoother, both inside the instances and in the background. Besides, instance boundaries are sharper with $\mathcal{L}_{MS}$. Different instances are better separated from each other by adding quantization.

Table 3.8: semantic contour F1-score on PASCAL VOC *val*.

| $\theta$ | 1 | 5 | 10 |
|---|---|---|---|
| w/o $\mathcal{L}_{MS}$ | 21.6 | 59.1 | 69.6 |
| w/ $\mathcal{L}_{MS}$ | 23.5 | 59.6 | 69.9 |
| w/ quantization and $\mathcal{L}_{MS}$ | 23.3 | 60.2 | 71.7 |

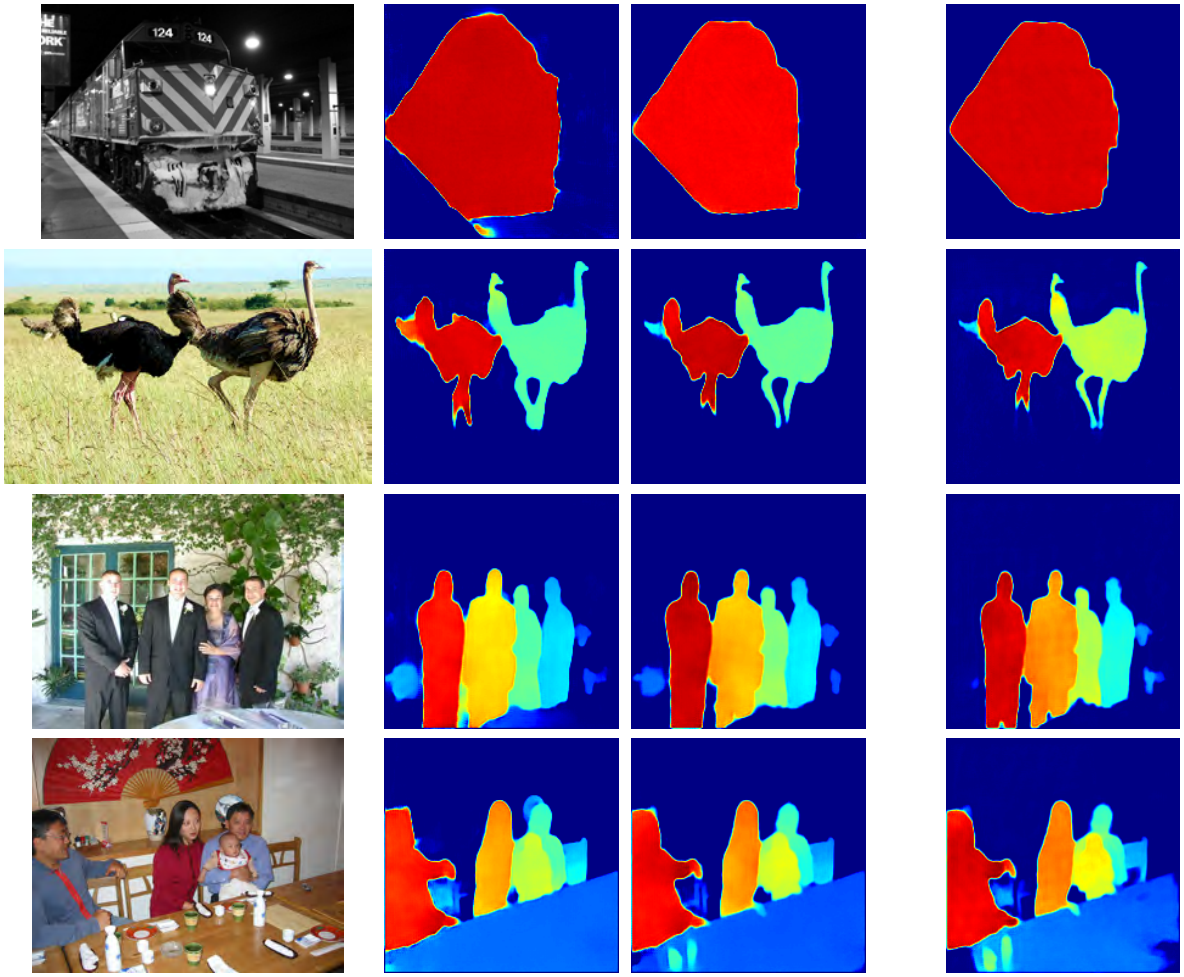
**Influence of the IoU head** We run an ablation study to identify how the classification confidence $S_{cls}$ and the predicted IoU $S_{iou}$ affect the results. The weighted sum is computed as $\alpha * S_{iou} + (1-\alpha) * S_{cls}$ with $\alpha = [0, 1]$. Fig. 3.5 shows that it achieves better mAP at 70% $\sim$ 90% IoU as $\alpha$ increases, which means the predicted IoU can detect more objects in higher quality.

**Predict instance map on unseen categories** Because our DVIS method learns to segment instances directly from instance-level ground truth, it can recognize 'objectness' for unseen categories by relating them to seen ones. We test it by running the model trained on PASCAL VOC *train set* on images containing unseen categories from the DAVIS challenge [[Pont-Tuset et al., 2017c]]. Examples are shown in Fig. 3.6, which shows DVIS can recognize 'objectness' and segment the instances.

## 3.5 Quanlitative Results

We show some qualitative results on the PASCAL VOC dataset in Fig. 3.7 and the MS-COCO dataset in Fig. 3.8 and Fig. 3.9. We also show some failure cases in Fig. 3.10. In those failure cases, our method fails to predict a good instance map when the scene becomes too crowded.

Note that part of the reason the algorithm is failing on those crowded scenes may be because of the way COCO is labeled. As can be seen in Fig. 3.10, among all the persons



RGB image | without $\mathcal{L}_{MS}$ | with $\mathcal{L}_{MS}$ | with quantization and $\mathcal{L}_{MS}$

Figure 3.4: This figure shows the predicted instance map from the model trained w/o or w/ the Mumford-Shah regularization, where the previous one is smoother inside the instances and the background and there is less noise along instances' boundaries

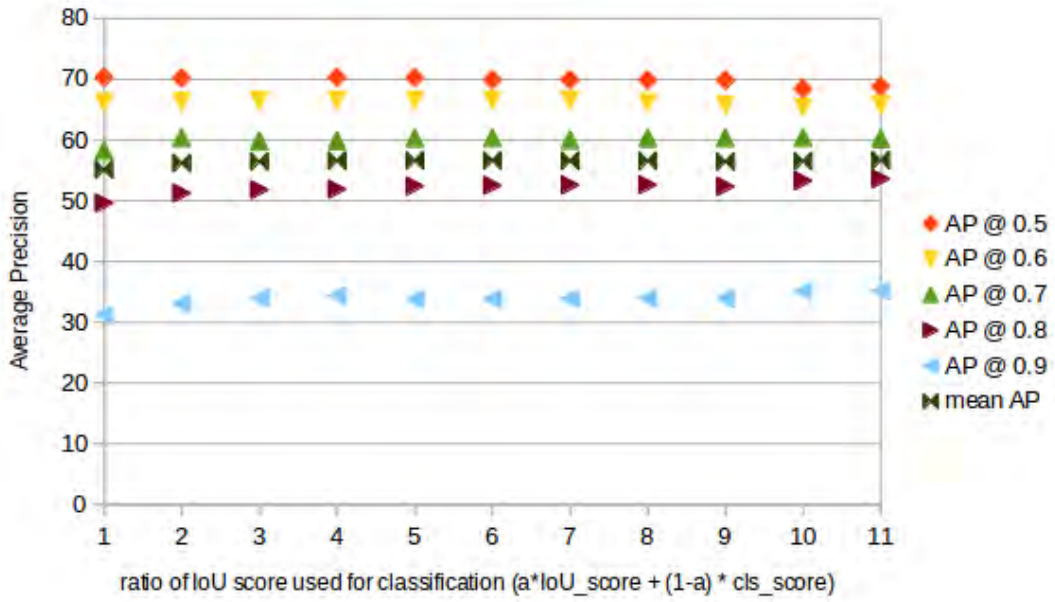
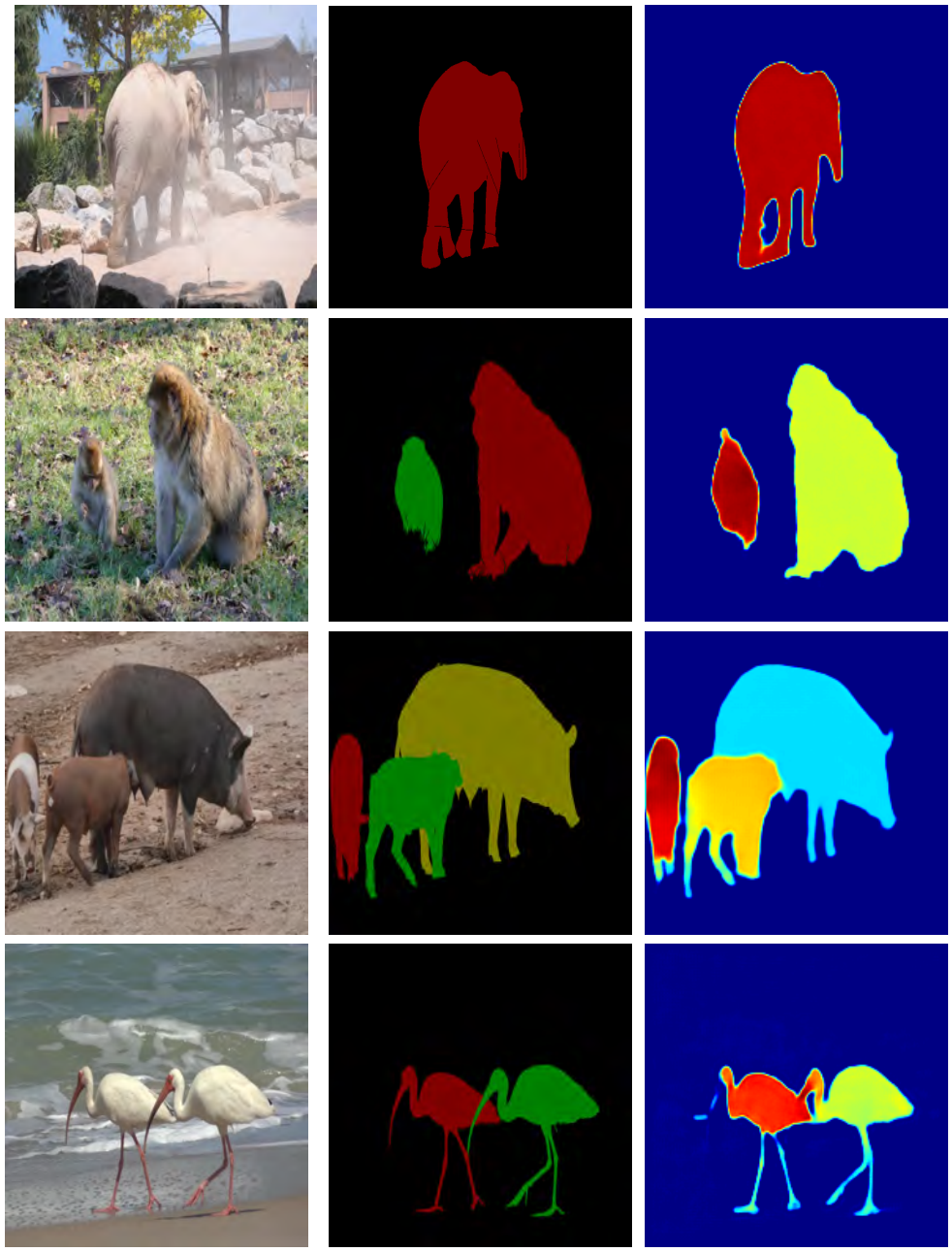



Figure 3.5: Ablation study on how the IoU score affects the instance segmentation on PASCAL VOC *val*.

in the scene, only some are labeled as persons while some are not. We hypothesize this confuses our algorithm more than the anchor-based algorithms, since our permutation-invariant loss looks globally at all pixel pairs, whereas anchor box-based methods only analyze locally within each box. It would be interesting if we ran the algorithm on a dataset where instances are more consistently labeled.

## 3.6 Conclusion

In this Chapter, we proposed deep variational instance segmentation (DVIS), which relaxes instance segmentation into a variational problem with a novel variational objective that includes a permutation-invariant component. Such a variational objective leads to an end-to-end training framework with an FCN directly predicting real-valued instance labels on the image. During inference time, we discretize the predicted continuous labels and utilize a small CNN to categorize them into semantic categories, as well as reject false positives. Experiments have shown that the proposed approach improves



RGB image GT predicted instance map

Figure 3.6: Predicted instance map on unseen categories from DAVIS challenge [[Pont-Tuset et al., 2017c]].

over the state-of-the-art in search-free instance segmentation approaches, especially on higher overlap thresholds, while being much faster. Such performance shows that our model is effective and efficient in capturing the global shape information in objects and segmenting objects with higher precision.

DVIS showed a distinct philosophical difference from most search-based algorithms in that it inherently processes the entire image with a single global glance. Most search-based algorithms look carefully at each local region to locate small objects, whereas DVIS directly gazes at the entire image and extracts objects in one shot. Hence, DVIS might be missing out on some small objects, as our COCO results have shown. However, we argue that there are plenty of applications e.g. in robotics where segmenting the prominent objects quickly and accurately is of the utmost importance, rather than an exhaustive list of small and far-away objects. In those scenarios, the fast global approach of DVIS would make more sense since it deals with a significantly smaller amount of object candidates. In the future, we will further explore variants of the top-down instance segmentation paradigm from DVIS to improve its performance on small objects.

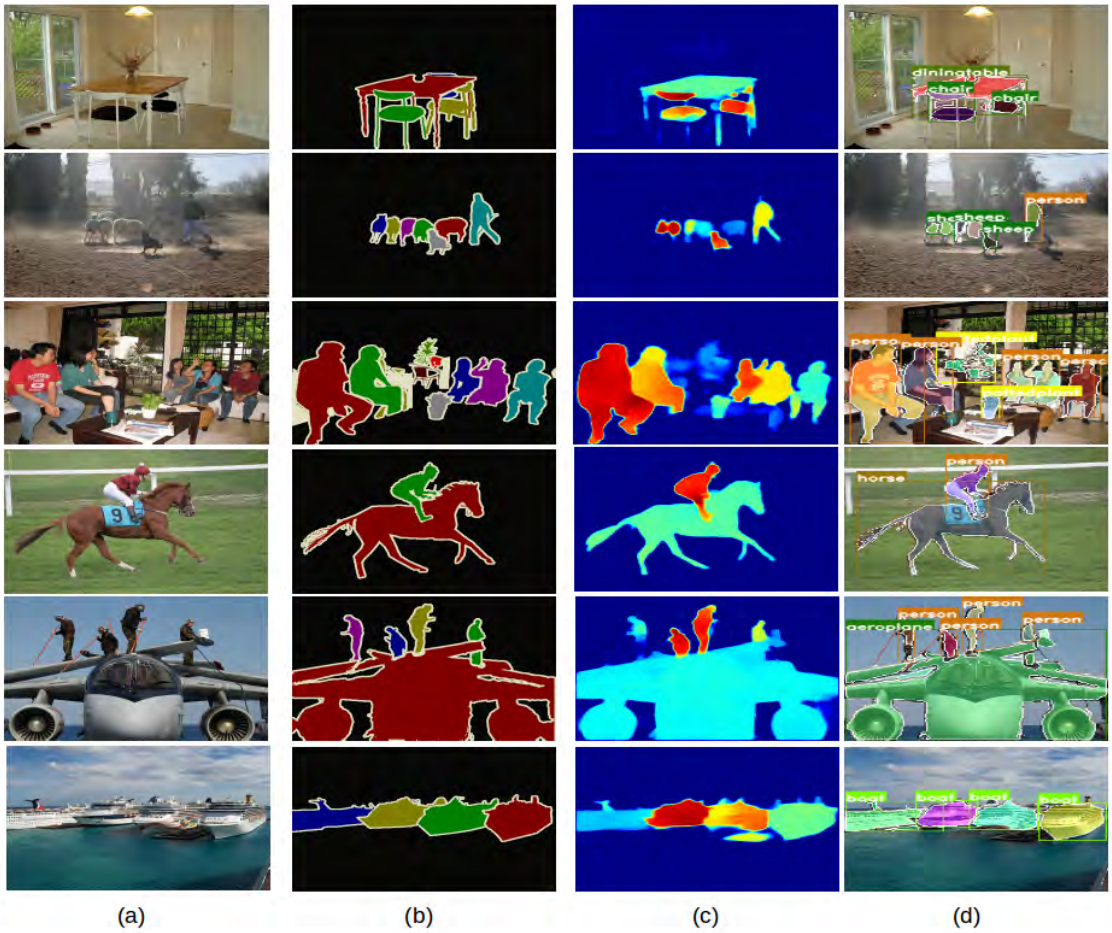
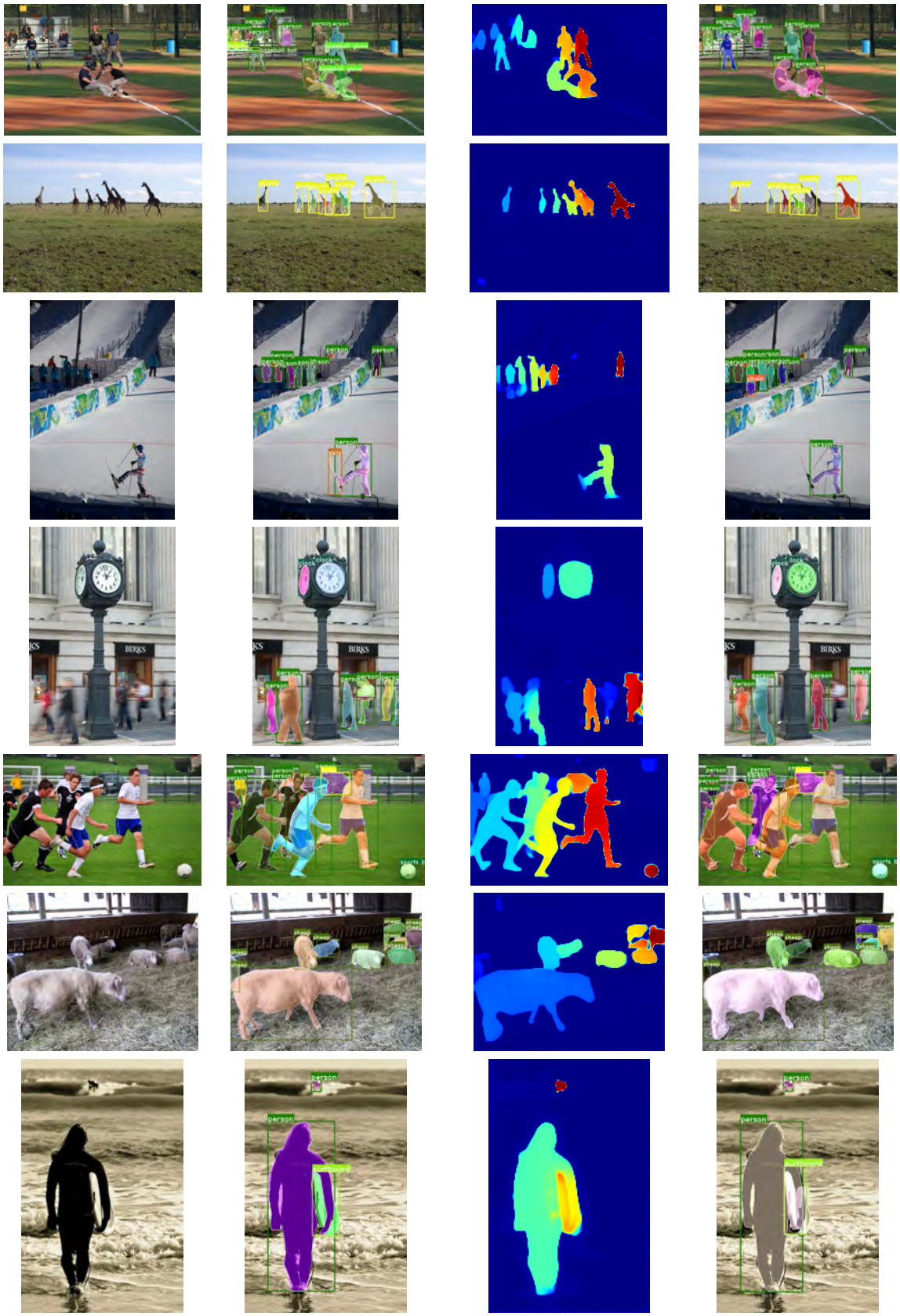



Figure 3.7: Examples from Pascal VOC 2012 *val* subset. From left to right: Image, Ground Truth, Predicted Instance Map, Final Instance Segmentation from DVIS(best viewed in color)

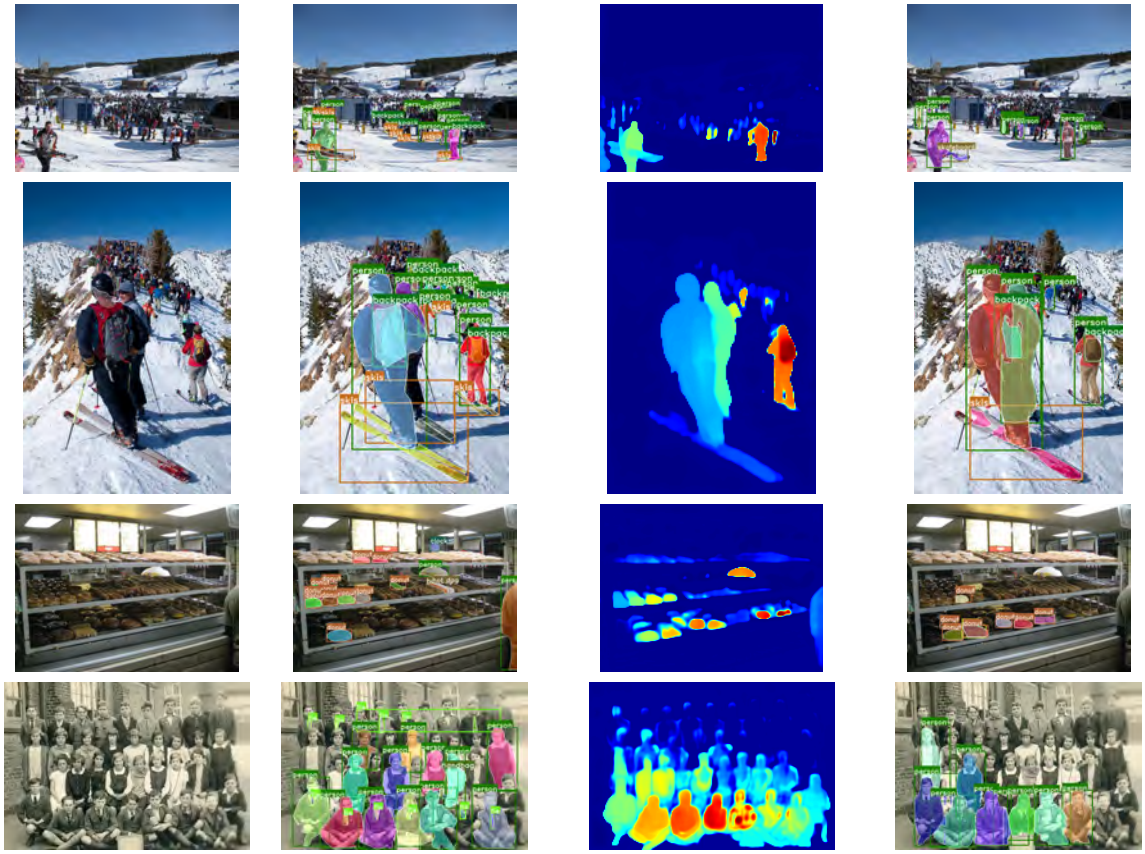
RGB image GT Predicted Instance Map final Seg.

Figure 3.8: This figure shows qualitative results on COCO *val2017* set, part(1)



RGB image GT Predicted Instance Map final Seg.

Figure 3.9: This figure shows qualitative results on COCO *val2017* set, part (2)



RGB image GT Predicted Instance Map final Seg.

Figure 3.10: Examples of inaccurate predicted instance maps with crowded objects on the COCO *val2017* set

# Chapter 4: Unsupervised Video Object Segmentation

In our first work, we developed a model capable of segmenting objects within images. However, the accuracy of segmentation is contingent upon the quality of the image, making it particularly challenging to accurately delineate objects with blurred boundaries, significant occlusions, or those that were out of focus. These issues are especially prevalent in videos.

In this chapter, we delve into our second work, focused on tackling the Unsupervised Video Object Segmentation (UVOS) task. UVOS involves the discovery of objects within individual frames and the consistent assignment of coherent object IDs to these objects throughout the entire sequence. To address these challenges, we introduce a framework designed to enhance segmentation quality by leveraging information from nearby frames. By incorporating a Video Object Segmentation (VOS) algorithm to track the refined object segmentation across the entire video, we provide a solution for the UVOS task.

## 4.1 Introduction

For robots to operate safely and reliably in dynamic environments or 'in-the-wild', they must be able to discover novel unseen objects with no supervision from continuous video streams. Robots can be pre-trained to understand what general objects may look like, but once deployed in the field, it would be very difficult to supply them with additional annotations and they are left on their own to recognize objects from novel categories. Hence, the capability of unsupervised object discovery from new videos, which more commonly is called Unsupervised Video Object Segmentation (UVOS)[Caelles et al., 2019], is an important research problem.

In the related problem of semi-supervised Video Object Segmentation (VOS), the first frame annotation is provided to the algorithm, which tracks and segments each object throughout the rest of the video. Most recent works typically utilize space-time trans-

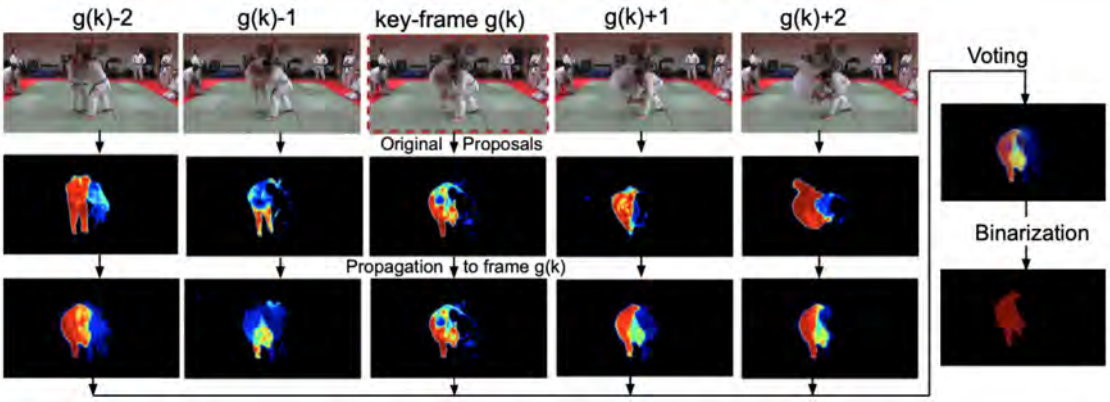



Figure 4.1: Illustration of the proposed MCMPG for object proposal refinement using a key frame clip with size 5 ( *"judo"* from DAVIS 2017 *val* set). **On the left side:** the first row is the RGB frames within the local window (the key frame $g(k)$ is the central frame highlighted in the red border). The second row is the segmentation of the object (the left person in *"judo"*) on each frame. The third row shows the proposals propagated to the key frame. **On the right side:** the first image is the voting of the object proposals inside a maximal clique on the *MP-Graph*, which is created with all propagated object proposals initially generated on the key frame clip. The second image is the final binarized object mask we obtained.

formers like STM [Oh et al., 2019] which properly match the visual features in a new frame with previous frames using a deformable attention model. This helps the systems track objects across significant motion, deformation, and occlusion.

Hence, a simple and natural idea to address the UVOS problem in prior work is to identify object proposals on a few key frames and then utilize a semi-supervised VOS algorithm to track them [Luiten et al., 2020]. Usually, instance segmentation algorithms such as Mask-RCNN [He et al., 2017] are utilized to identify object proposals in those key frames. However, for VOS to work well, the starting frame usually needs to be annotated with high precision, because wrongly annotated regions in this frame, serving as ground truth, could lead to significant drift in subsequent frames. Similarly, a missing part from the annotation might be missed forever because the tracker thinks it belongs to the background instead of the object. Hence, achieving high segmentation accuracy at those key frames is essential for better UVOS performance.

However, single-frame instance segmentation is often noisy and does not always provide the required precision for tracking. In this work, we aim to solve the novel task of **improving the segmentation quality on key frames**. The idea is to take into account object proposals from nearby frames and use them to jointly reason about the segmentation in the key frame, which allows segmentations from different frames to cancel out the noise in each other. Through experiments, we show that refining key frame proposals *before long-term tracking* leads to significantly better UVOS performance.

Our approach builds a **M**ulti-frame **P**roposal **G**raph (MP-Graph) using object proposals initially generated in a local window around each key frame, and then locates maximal cliques in this graph from which the final segmentation on the key frame is generated. Each clique in the graph consists of multiple segments that correspond to the same object, hence jointly reasoning among all of them may generate more precise segmentations. Fig. 4.1 shows an example, where the segmentation of the object is poor on the key frame, meanwhile, none of the segmentations from the five frames are perfect. However, their joint voting produces a segmentation very close to the ground truth.

Once better key frame segmentations are obtained, one can use any VOS algorithm to propagate them to the entire video and use a sequence non-maximum suppression (NMS) approach to filter out redundant objects. Our approach is lightweight and fast and thus adds little computational overhead to the VOS algorithms used for tracking key frame proposals. We also show that our improved key frame segmentations benefit a similar problem of Video Instance Segmentation (VIS), where it is required to classify the tracked object instances to a known set of categories, extending image instance segmentation to the video domain.

We validate our approach through extensive experiments providing quantitative and qualitative analysis on both tasks. Experiments on the DAVIS-UVOS and Youtube-VIS benchmark show that the better key frame segmentations from our approach lead to state-of-the-art performance. Notably, our approach outperforms the state-of-the-art [Lin et al., 2021] in UVOS that jointly trains the proposal generation model and the STM model. Not needing joint training is a significant advantage of our model – this makes it future-proof because it can be then plugged in seamlessly to any future VOS models that achieve better performance without a cumbersome re-training process.

To summarize, our main contributions are,

- We propose to solve the novel task of improving key frame segmentations using nearby frames.
- We propose **M**aximal **C**liques on **M**ulti-Frame **P**roposal **G**raph (MCMPG), which utilizes maximal cliques over a graph of object proposals from the local window. Reasoning over the multiple similar proposals within these maximal cliques yields better object proposals. MCMPG is *modular*, *lightweight*, and *fast*, enabling it to be plugged into any VOS/VIS algorithms that track object proposals to improve their performance on the UVOS/VIS tasks.
- UVOS with improved key frame segmentations from MCMPG outperforms all SOTA methods on the DAVIS-UVOS validation and test-dev set. MCMPG also significantly improves the performance of the Video Instance Segmentation Task on the Youtube-VIS 2019 validation set.

## 4.2 Related Work

**Image Instance Segmentation.** As introduced in Section 3, the task is to produce pixel-level predictions for each object instance in a frame. Top-down [He et al., 2017, Huang et al., 2019, Liu et al., 2018] approaches such as Mask-RCNN [He et al., 2017] and its follow-ups adopt the 'detect-then-segment' paradigm. These two-stage approaches are accurate but relatively slow due to the exhaustive search process.

To overcome these drawbacks, bottom-up methods [De Brabandere et al., 2017, Liu et al., 2017, Newell et al., 2016] view the problem as 'label-then-cluster' where the model learns an affinity function to group pixel embeddings belonging to the same object instance. Single-stage algorithms [Wang et al., 2020b,c, Xie et al., 2020, Yuan et al., 2020] simplify computational-heavy post-processing, and in particular, SOLO [Wang et al., 2020b] and SOLOv2 [Wang et al., 2020c] segment the object instances by locations without using bounding boxes or metric learning. DETR [Carion et al., 2020] inspires End-to-End transformer-based models [Cheng et al., 2021a, 2022, Dong et al., 2021, Thawakar et al., 2022b, Wang et al., 2021a]. The most recent Mask2Former [Cheng et al., 2022] uses masked attention to achieve state-of-the-art performance in the instance segmentation

task.

**Semi-Supervised Video Object Segmentation.** Video Object Segmentation (VOS) can be applied to acquire pixel-level segmentations of primary objects in the scene given unconstrained videos. Depending on the level of supervision, they can be categorized as semi-supervised (one-shot), interactive, and unsupervised (zero-shot). Early work [Caelles et al., 2017, Cheng et al., 2017, Perazzi et al., 2017] fine-tuned a pretrained network at test-time using multiple data augmentations on the mask of each object from the first frame. They are usually very slow due to the excessive test-time fine-tuning. Their performance under occlusion and appearance changes is also limited due to the overfitting to the appearance of the first frame. Later approaches improved speed and accuracy through metric learning [Chen et al., 2018b, Voigtlaender et al., 2019], guided propagation [Oh et al., 2018, 2019, Yang et al., 2018] and transformer-type networks [Cheng et al., 2021b, Li et al., 2020b, Nguyen and Li, 2021, Oh et al., 2019, Seong et al., 2020, Wu et al., 2020].

**Unsupervised Video Object Segmentation.** Early work utilizes motion patterns such as clustering object motion trajectories [Brox and Malik, 2010, Fragkiadaki et al., 2012, Xie et al., 2019] or CNN-based spatio-temporal grouping [Dave et al., 2019, Xie et al., 2019]. Some combine appearance with optical flow for enhanced features [Cheng et al., 2017, Lu et al., 2019, Zhou et al., 2020], or use optical flow alone [Tokmakov et al., 2017]. A common drawback to these methods lies in their inability to be generalized to videos that have static objects, large motion blur, or cluttered backgrounds.

For multi-object VOS, learning appearance models of all the object proposals have been previously explored [Li et al., 2013, Wu et al., 2015]. Currently, the 'track-by-detect' [Garg and Goel, 2021, Luiten et al., 2020, Ventura et al., 2019, Wang et al., 2019a] paradigm is popular where an object discovery framework generates object proposals via Mask-RCNN [He et al., 2017] and then these objects are tracked consistently through a video sequence. UnOVOST [Luiten et al., 2020] pruned tracklets from proposals into long-term tracks via visual similarity. Most recently, [Zhou et al., 2021a] proposed a novel instance segmentation, tracking, and re-identification network. In AGNN [Wang et al., 2019a], mask proposals over a video sequence were aggregated via graph neural networks. Similar works optimizing cliques for VOS have also been proposed [Koh et al.,

2018, Ma and Latecki, 2012]. Our work uses cliques for refining the key frame segment instead of VOS, hence very different from the approaches above.

**Video Instance Segmentation.** The VIS task was proposed in MaskTrack R-CNN [Yang et al., 2019] which adds a tracking head to Mask RCNN and an external memory to store and associate features of object instances across multiple frames. This tracking paradigm is extended in [Bertasius and Torresani, 2020, Cao et al., 2020]. STEM-seg [Athar et al., 2020] models video clips as 3D space-time volumes to predict masks by clustering learned embeddings. An application of graph neural networks is seen in VisSTG [Wang et al., 2021b]. Transformer-based techniques have become increasingly successful [Hwang et al., 2021, Thawakar et al., 2022a, Wang et al., 2021c, Wu et al., 2022b] applying cross-attention to process video clips. Mask2Former is extended to VIS [Cheng et al., 2022] by directly making predictions on the entire video sequence. Online VIS methods also exist, but they usually have lower accuracy due to not observing the entire sequence [Han et al., 2022, Wu et al., 2022a]. Propose-Reduce [Lin et al., 2021] introduces an alternative paradigm of "segment-then-propagate" to benefit from the progress made in VOS tasks. As object mask propagation is sensitive to its segmentation on the reference frame, this method generates instance proposals on multiple key frames and then reduces redundant sequences of the same instances using non-maximum suppression. Note that our work is different from [Lin et al., 2021] in that our focus is to present a modular approach that improves key frame proposals (before long-term tracking) without requiring any joint training.

## 4.3 MCMPG

MCMPG aims to generate the key frame proposals with higher quality by creating a multi-frame proposal graph and finding its maximal cliques. Afterward, any semi-supervised VOS algorithm can be used to track each instance proposal to the beginning and end of the sequence. Sequence NMS can then be used to remove duplicate segments. The architecture of using MCMPG to perform UVOS is shown in Fig. 4.2.

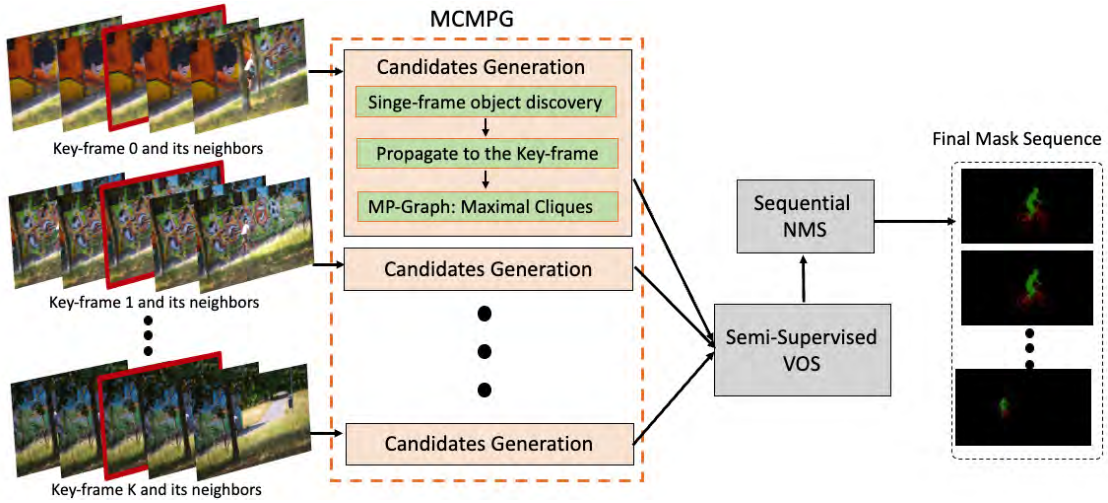



Figure 4.2: Illustration of using the proposed MCMPG algorithm for the UVOS task. The proposed MCMPG is used to refine the proposals on the key frame from its *key frame clip*, which includes the key frame and its neighboring frames. Afterwards, an off-the-shelf semi-supervised VOS algorithm can track these proposals bidirectionally through the whole sequence to obtain the final unsupervised video object segmentation. MCMPG includes 3 steps: 1) Discover objects on each frame in the key frame clip; 2) Propagate the proposals to the key frame; 3) Create the *MP-Graph* from the propagated proposals and locate its maximal cliques. By combining proposals within these maximal cliques, we can obtain refined key frame segmentations and subsequently improve the performance of the UVOS task. (Best viewed in color)

### 4.3.1 Problem Definition

Given a set of RGB frames $\mathcal{I} = \{I_t\}_{t=0}^{T-1}$ where $I_t \in \mathcal{R}^{3\times h\times w}$ and $T$ is the total number of frames, the goal is to produce a sequence of consistent segmentation masks $\mathcal{S} = \{M_t\}_{t=0}^{T-1}$ for each of the $m$ objects in the video. where $M_t \in \mathcal{R}^{m\times h\times w}$ represents the masks for all of the objects.

### 4.3.2 Proposal Refinement on Key Frames

As we argued in the introduction, the quality of the generated key frame proposals is crucial for successful unsupervised video object segmentation. However, as Fig. 4.1 shows, even state-of-the-art instance segmentation approaches can generate bad segments

in some frames due to motion blur, occlusion, and object poses that are very different from the training set. In this section, we introduce our main contribution, MCMPG , that improves key frame proposals by joint reasoning from multiple frames.

**Key Frame Selection.** For a $T$-frame video, $K$ key frames $\{I_{g(0),\cdots,I_{g(K-1)}}\}$ can be selected with fixed intervals: $g(k) = k\max(\lfloor T/K \rfloor), 1), k = 0, \cdots, K-1$.

In contrast to [Lin et al., 2021] that uses the segments in the key frames directly as key frame proposals to track, we select a key frame clip to generate the key frame proposals. Here, the key frame clip on a key frame $I_{g(k)}$ is $I^C_{g(k)} = \{I_i | i = g(k) - \frac{H-1}{2}, \cdots, g(k) + \frac{H-1}{2}\}$. It contains the key frame itself and its $H-1$ neighbors from a local window centered around the key frame.

**Proposal Generation and Propagation.** With the key frame clip $I^C_{g(k)}$, we first discover objects in each frame individually. Then we propagate all the object segments $S^I_i$ to the key frame $g(k)$ by using the same semi-supervised VOS algorithm that we used in a later stage (in Fig. 4.2). The set of propagated proposals is denoted as $S^{g(k)} = \{S^{g(k)}_i | i = g(k) - \frac{H-1}{2}, \cdots, g(k) + \frac{H-1}{2}\}$, where $S^{g(k)}_i \in R^{L_i \times h \times w}$ is the probability masks of $L_i$ proposals that are segmented in frame $i$ and then propagated to the key frame $g(k)$. This set can be used to create the multi-frame proposal graph which is introduced below.

***MP-Graph* (Multi-frame Proposal Graph).** On the proposal set $S^{g(k)}$ from one key frame clip, we create an undirected graph where each propagated proposal in $S^{g(k)}$ is one vertex in the graph, and an edge is created between a pair of vertices if their Intersection-over-Union (IoU), Eq.(4.1)) is larger than $t_0$.

$$IoU(S^{g(k)}_{i,o_1}, S^{g(k)}_{j,o_2}) = \frac{S^{g(k)}_{i,o_1} \cap S^{g(k)}_{j,o_2}}{S^{g(k)}_{i,o_1} \cup S^{g(k)}_{j,o_2}} \tag{4.1}$$

where $S^{g(k)}_{i,o_1}$, $S^{g(k)}_{j,o_2}$ are two propagated proposals from the temporal frame $i$ and $j$ to the key frame $g(k)$ so that their IoU is measured in the same key frame.

We name the graph as the Multi-frame Proposal Graph given that its nodes are propagated proposals from different temporal frames and its edges are created based on the

**Algorithm 1** Key frame Proposal Generation and Refinement

**Input :** key frame and its neighbours $\{I_i | i = g(k) - \frac{H}{2}, \cdots, g(k) + \frac{H}{2}\}$
**Output:** Instance Proposals S in the key frame
**for** $i \leftarrow g(k) - \frac{H}{2}$ **to** $g(k) + \frac{H}{2}$ **do**
  $S_i^I \leftarrow InstanceSegmentation(I_i)$ $S_i^{g(k)} \leftarrow Propagate(S_i, \{I_i \cdots I_{g(k)}\})$
$G = MP\text{-}Graph(S^{g(k)})$ $Cliques = G.maximalCliques()$
**for** $C \in Cliques$ **do**
  $S^c \leftarrow combine(C, S^{(g(k))})$ `// Eq.4.2`
$S_k \leftarrow \cup S^c$
**return** $S_k$

spatial IoU computed in the same time frame $g(k)$. After the *MP-Graph* is created, we adopt the *maximal clique algorithm* [Bron and Kerbosch, 1973] to generate the final key frame instance proposals to be tracked (See Fig. 4.3).

**Key Frame Proposals.** In an undirected graph, a *clique* is a complete sub-graph in which every two vertices are adjacent. A *maximal clique* is a clique that *cannot be extended* by including *any more adjacent* vertex. The largest maximal clique is called a *maximum clique*. Accordingly, in the *MP-Graph*, a *maximal clique* is a subset of propagated proposals which all significantly overlap each other.

Given a maximal clique $C$ that contains $n$ propagated proposals $\{O_i | i = 0, \cdots, n-1\}, n \leq H$, its corresponding key frame object proposal $S^C$ is computed as:

$$S^C = (\frac{1}{n} \sum_{i=0,\cdots n-1} O_i) \geq t_1 \tag{4.2}$$

where $t_1$ is a threshold that can be set to a small value (0.2 in the experiments) without introducing noise in the segmentation. Notably, the algorithm can retrieve the object proposals even when the segmentation on the key frame is poor. In the example presented in Fig.4.1, the segment of the person on the left is noisy on frame $g(k)$ due to serious motion blur, exhibiting low confidence in detecting the person's left hand and high confidence in detecting the left leg. Conversely, the propagated proposals from $g(k) + 1$ and $g(k) + 2$ exhibit high confidence in detecting the left hand and low confidence in detecting the left leg. In contrast, the propagated proposal from $g(k) - 2$ mistakenly

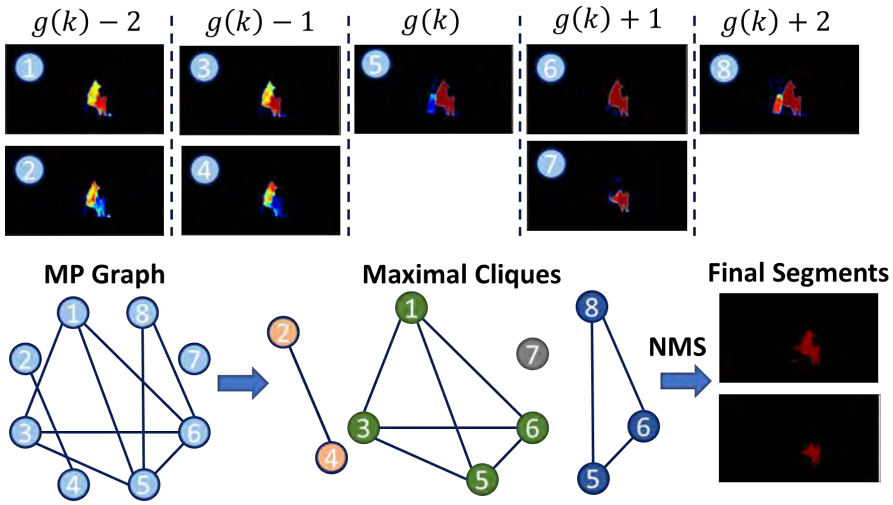



Figure 4.3: An example of Multi-frame Proposal Graph. Propagated segments are connected based on their IoU on the key frame, then one segment is generated from each clique (Best viewed in color)

includes a portion of another person. The maximal clique, consisting of the proposals propagated from $g(k)-2, g(k), g(k)+1$, and $g(k)+2$, effectively captures information on all parts of the person and accurately segments the individual without errors involving another person.

### 4.3.3 Using MCMPG in the UVOS task

Once we have MCMPG-refined proposals on $K$ key frames, we can use any VOS algorithm to track the proposals through the video, and use sequence NMS to remove duplicates. An example architecture to use MCMPG for the UVOS task is shown in Fig. 4.2.

**Sequence propagation.** After the key frame proposals are obtained, any off-the-shelf semi-supervised VOS method can be used to track all the objects from the key frame bidirectionally through the video sequence. In this way, we can always plug in state-of-the-art VOS algorithms for better performance. The propagation results from each key frame $\{S_k | k = 0, \cdots, K-1\}$ are then concatenated together as $\hat{S} \in \mathcal{R}^{T \times N \times h \times w}$, assuming we segment $N$ objects in total after the tracking stage.

**Sequence Score.** Define $S_i^t$ and $C_i^t$ as the $i^{\text{th}}$ object proposal and its objectness score at time $t$, respectively. Define a tracked object sequence $\hat{S}_o = \{\hat{S}_o^t, \hat{S}_o^1, ..., \hat{S}_o^{T-1}\}$ where $\hat{S}_o^t$ represents the tracked mask at time $t$. The score of $\hat{S}_o$ is computed by:

$$Score_o = \frac{1}{T} \sum_{t=0,\cdots,T-1} \max_i (IoU(\hat{S}_o^t, S_i^t) C_i^t) \tag{4.3}$$

where we use the objectness of per-frame proposals $S_i^t$ and its IoU with $\hat{S}_o^t$ to obtain the score of $\hat{S}_o^t$. This helps us to generalize to SOLOv2 which does not allow recomputing objectness on a new mask $\hat{S}_o^t$ that comes from tracking. The sequence score for each object sequence is used in Sequence NMS for removing duplicate object sequences that are detected in different key frames. It is also used to rank object sequences in the case where the algorithm is allowed to output only a fixed number of detections. Our sequence NMS follows [Lin et al., 2021] which removes overlapping tracks by running the traditional NMS algorithm with the tracking scores and the sequence IoU.

## 4.4 Experiments

### 4.4.1 Implementation Details

In the experiments, key frame proposals with areas smaller than 10 pixels are discarded. And except ablations, the number of key frames $K$ is set to 2 on DAVIS and 8 on Youtube-VIS, the size of key frame clip $H = 3$, threshold $t_0 = 0.5, t_1 = 0.2$.

### 4.4.2 Object proposal quality with MCMPG.

The DAVIS 2017 [Pont-Tuset et al., 2017b] benchmark is used for video multi-object segmentation with high-quality masks for salient objects. It consists of 60 sequences used for training and 30 for validation. We provide a comparison of the object proposals quality with and without MCMPG on the key frames evaluated against the ground truth objects on the DAVIS 2017 *val* set. MCMPG improves the key frame proposals by 3.7% in terms of *mIoU* as shown in Table 4.1. This significant improvement in the quality of the key frame proposal is the main driver of the performance in downstream tasks.

Table 4.1: Quality of the key frame proposals on DAVIS 2017 *val*.

| | *w/ MP-Graph* | *w/o MP-Graph* |
|---|---|---|
| mIoU (%) | 79.2 | 75.5 |

### 4.4.3 Unsupervised Video Object Segmentation

**Dataset.** Besides DAVIS 2017, DAVIS 2019 [Caelles et al., 2019] is an extension of DAVIS 2017 for the UVOS task. It has the same training and validation set as DAVIS 2017 and 30 new sequences in its *test-dev* set. To demonstrate that our proposed *MP-Graph* is network-agnostic and can work with a wide range of object discovery models, we show experimental results with both SOLOv2 [Wang et al., 2020c] and Mask-RCNN instance segmentations from [Lin et al., 2021]. SOLOv2 model is initialized with COCO pre-trained weights. We then finetune its kernel branch and feature branch for 10 epochs, then the FPN for 5 epochs, and finally the ResNet-101 backbone blocks from the last block to the first block for 5 epochs per block. Similarly, for the tracker, we show experimental results with both STM [Oh et al., 2019] and STCN[Cheng et al., 2021b].

**Metric.** We follow the standard evaluation settings [Perazzi et al., 2016]: the performance is reported in terms of region similarity $\mathcal{J}$ , boundary accuracy $\mathcal{F}$, and the overall metric $\mathcal{J}\&\mathcal{F}$. The evaluation scores on the *test-dev* set are obtained from the evaluation server of the DAVIS 2019 challenge.

**Results on DAVIS 2017 *val*.** In Table 4.2, we compare our approach with state-of-the-art unsupervised video multi-object segmentation methods on the DAVIS 2017 dataset. The common baselines from published works are included: AGS [Wang et al., 2020a], MATNet [Zhou et al., 2020], AGNN [Wang et al., 2019a], Stem-Seg [Athar et al., 2020], UnOVOST [Luiten et al., 2020], Target-Aware [Zhou et al., 2021a], and Propose-Reduce [Lin et al., 2021]. As shown in Table 4.2, on DAVIS 2017 *val*, our approach achieves the highest overall results across most metrics. Prior methods such as UnOVOST and MATNet are computationally expensive and also need to compute optical flow for motion estimation. Our work requires only RGB frames as input and outperforms the previous best method Propose-Reduce [Lin et al., 2021] by 1.7% in terms of $\mathcal{J}$ & $\mathcal{F}$-Mean when using the same segmentation method and the same ResNeXt-101 backbone. With the more complex ResNeXt-101 backbone, we outperform [Lin et al., 2021] by 7.8%.

Note that by utilizing frames next to the key frames, our approach may be thought of as utilizing more frames than [Lin et al., 2021]. However, the ablation study in [Lin et al., 2021] shows that more key frames do not further help their performance on this dataset, which shows the importance of MCMPG in terms of combining and refining the proposals.

In Table 4.2, we also adopt the Mask-RCNN module from [Lin et al., 2021] to compute the key frame proposals. This approach without the *MP-Graph* achieves a score 1.7% higher than [Lin et al., 2021] since we adopt a separate STM model to perform tracking. Adding *MP-Graph* improves another 1.7% over this baseline, which shows the effectiveness of MCMPG even with the same object proposal algorithm as [Lin et al., 2021]. MCMPG can be plugged into any tracking algorithm. In order to show this, we report the performance of MCMPG with STCN [Cheng et al., 2021b] as well.

**Results on DAVIS 2019 *test-dev*.** We evaluate the proposed approach MCMPG on DAVIS 2019 *test-dev* set shown in Table 4.3. Our approach achieves the state-of-the-art result in terms of $\mathcal{J}$ & $\mathcal{F}$-Mean at 61.2. Compared with the previous state-of-the-art Target-Aware [Zhou et al., 2021a], our approach improves significantly on the boundary F-metric, which shows that our proposals cover object boundaries significantly better. Here we do not test the ResNeXt-101 backbone for a fair comparison with prior work, which also does not use this more complex backbone.

Table 4.2: Quantitative video multi-object segmentation results on DAVIS 2017 *val*.

| Methods | Instance Seg. | backbone | $\mathcal{J}$ & $\mathcal{F}$ Mean | $\mathcal{J}$ -Mean | $\mathcal{J}$ - Recall | $\mathcal{J}$ - Decay | $\mathcal{F}$ -Mean | $\mathcal{F}$ - Recall | $\mathcal{F}$ - Decay |
|---|---|---|---|---|---|---|---|---|---|
| AGS [Wang et al., 2020a] | - | ResNet-101 | 57.5 | 55.5 | 61.6 | 7.0 | 59.5 | 62.8 | 9.0 |
| MATNet [Zhou et al., 2020] | - | ResNet-101 | 58.6 | 56.7 | 65.2 | -3.6 | 60.4 | 68.2 | 1.8 |
| AGNN [Wang et al., 2019a] | - | ResNet-101 | 61.1 | 58.9 | 65.7 | 11.7 | 63.2 | 67.1 | 1.2 |
| STEm-Seg [Athar et al., 2020] | - | ResNet-101 | 64.7 | 61.5 | 70.4 | **-4.0** | 67.8 | 75.5 | 1.2 |
| UnOVOST [Luiten et al., 2020] | - | ResNet-101 | 67.9 | 66.4 | 76.4 | -0.2 | 69.3 | 76.9 | **0.0** |
| Target-Aware [Zhou et al., 2021a] | - | ResNet-101 | 65.0 | 63.7 | 71.9 | 6.9 | 66.2 | 73.1 | 9.4 |
| Propose-Reduce [Lin et al., 2021] | Mask-RCNN | ResNet-101 | 68.3 | 65.0 | - | - | 71.6 | - | - |
| Propose-Reduce [Lin et al., 2021] | Mask-RCNN | ResNeXt-101 | 70.6 | 67.2 | - | - | 73.9 | - | - |
| MCMPG + STM (*w/o MP-Graph*) | Mask-RCNN | ResNeXt-101 | 70.0 | 67.1 | 73.0 | -1.1 | 72.3 | 80.0 | 0.9 |
| MCMPG + STM (*w/ MP-Graph*) | Mask-RCNN | ResNeXt-101 | 71.7 | 68.9 | 74.6 | -4.9 | 75.8 | 83.2 | -2.1 |
| MCMPG + STCN (*w/o MP-Graph*) | Mask-RCNN | ResNet-101 | 73.6 | 70.2 | 77.5 | -2.3 | 77.1 | 83.4 | 0.2 |
| MCMPG + STCN (*w/ MP-Graph*) | Mask-RCNN | ResNet-101 | 76.8 | 73.8 | 81.9 | -1.2 | 79.2 | 85.5 | 1.9 |
| MCMPG + STM (*w/o MP-Graph*) | SOLOv2 | ResNet-101 | 71.2 | 68.2 | 76.5 | -2.2 | 74.0 | 81.2 | 0.9 |
| MCMPG + STM (*w/ MP-Graph*) | SOLOv2 | ResNet-101 | 72.5 | 69.0 | 77.3 | -3 | 76.1 | 83.3 | 5.3 |
| MCMPG + STM (*w/o MP-Graph*) | SOLOv2 | ResNeXt-101 | 72.7 | 69.9 | 76.6 | -3.7 | 75.5 | 82.7 | -1.1 |
| MCMPG + STM (*w/ MP-Graph*) | SOLOv2 | ResNeXt-101 | **78.4** | **75.4** | **83.9** | 0.05 | **81.4** | **88.9** | 0.04 |

Table 4.3: Quantitative video multi-object segmentation results on DAVIS 2019 *test-dev*.

| Methods | Backbone | $\mathcal{J}$ & $\mathcal{F}$ Mean | $\mathcal{J}$ -Mean | $\mathcal{J}$ - Recall | $\mathcal{J}$ - Decay | $\mathcal{F}$ -Mean | $\mathcal{F}$ - Recall | $\mathcal{F}$ - Decay |
|---|---|---|---|---|---|---|---|---|
| PDB [Song et al., 2018] | ResNet-50 | 40.4 | 37.7 | 42.6 | 4.0 | 43.0 | 44.6 | 3.7 |
| AGS [Wang et al., 2020a] | ResNet-101 | 45.6 | 42.1 | 48.5 | 2.6 | 49.0 | 51.5 | 2.6 |
| UnOVOST [Luiten et al., 2020] | ResNet-101 | 58.0 | 54.0 | 62.9 | 3.5 | 62.0 | 66.6 | 6.6 |
| Target-Aware [Zhou et al., 2021a] | ResNet-101 | 59.8 | 56.0 | **65.1** | 7.8 | 63.7 | 68.4 | 11.0 |
| MCMPG + STM (*w/MP-Graph*) | ResNet-101 | **61.2** | **56.1** | 63.5 | **-0.2** | **66.4** | **71.9** | **-0.5** |

### 4.4.4 Video Instance Segmentation

**Video Instance Segmentation (VIS).** Different from UVOS which segments salient object instances, VIS aims at discovering and segmenting all object instances of predefined object categories from videos. It requires predictions for both object segmentation and object categories. Usual VIS approaches contain a category classification head to predict the category score.

We adapt MCMPG to the VIS domain by adopting the Mask-RCNN module from [Lin et al., 2021] to generate object segments and the category scores on each frame as different settings. Meanwhile, STM [Oh et al., 2019] is used to propagate object segments generated on frames in a key frame clip to the key frame and to track key frame proposals bidirectionally throughout the videos. We also test our approach to the task by utilizing the latest transformer-based method, Mask2Former-VIS from [Cheng et al., 2022], as the single-frame object discovery network.

**Dataset.** YouTube-VIS 2019 [Yang et al., 2019] is a large-scale dataset for VIS with objects in multiple categories. It contains $2,283$ high-resolution YouTube videos for training and 302 for validation, covering 4,883 unique object instances out of 40 categories. We use this dataset to examine the performance of our model in more challenging scenarios.

**Metrics.** YouTube-VIS adopts the standard evaluation metrics in image instance segmentation, average precision (AP), and average recall (AR), to evaluate performance. It follows COCO evaluation [Lin et al., 2014] to compute AP by averaging it over multiple intersection-over-union (IoU) thresholds from 50% to 95% at step 5%.

**Results on YouTube-VIS 2019 *val*.** We compare our approach with state-of-the-art video VIS approaches on the YouTube-VIS 2019 benchmark. As shown in Table 4.4, our

approach achieves consistent improvements with all different backbones and instance segmentation methods. Specifically, adding the *MP-Graph* achieves at least 1.0% higher over the baseline without the *MP-Graph*, which shows the effectiveness of the MCMPG. Note that the results on the YouTube-VIS are significantly affected by the accuracy of the object categorization, which is orthogonal to our contribution to improving the key frame segmentation. Hence, our AP@50 is not necessarily the best, since this metric is mainly affected by classification accuracy, but our higher AP and higher AP@75 indicate better segmentation quality our approach achieves. Compared with SeqFormer [Wu et al., 2022b], our AP is higher, but AP@50 and AP@75 are both slightly lower. This shows that we very likely have achieved significantly better performance in the AP regimes even higher than 75% IoU, greatly indicating the strong segmentation quality our approach provides.

Table 4.4: Results on YouTube-VIS 2019 *val*.

| Methods | Instance Seg. | backbone | AP | AP@50 | AP@75 | AR@1 | AR@10 |
|---|---|---|---|---|---|---|---|
| SipMask [Cao et al., 2020] | - | ResNet-50 | 33.7 | 54.1 | 35.8 | 35.4 | 40.1 |
| STEm-Seg [Athar et al., 2020] | - | ResNet-101 | 34.6 | 55.8 | 37.9 | 34.4 | 41.6 |
| Target-Aware [Zhou et al., 2021a] | - | ResNet-101 | 37.1 | 57.1 | 40.9 | 34.8 | 43.2 |
| Propose-Reduce [Lin et al., 2021] | - | ResNet-101 | 43.8 | 65.5 | 47.4 | 43.0 | 53.2 |
| Propose-Reduce [Lin et al., 2021] | - | ResNeXt-101 | 47.6 | **71.6** | 51.8 | 46.3 | 56.0 |
| MCMPG (*w/o MP-Graph*) | Mask-RCNN | ResNet-101 | 43.4 | 64.4 | 48.9 | 45.0 | 57.1 |
| MCMPG (*w MP-Graph*) | Mask-RCNN | ResNet-101 | 44.6 | 64.2 | 49.5 | 46.4 | 58.5 |
| MCMPG (*w/o MP-Graph*) | Mask-RCNN | ResNeXt-101 | 47.4 | 70.6 | 52.3 | 47.5 | 60.0 |
| MCMPG (*w MP-Graph*) | Mask-RCNN | ResNeXt-101 | **48.4** | 70.4 | **52.7** | **48.6** | **60.1** |
| Transformer-based methods | | | | | | | |
| SeqFormer[Wu et al., 2022b] | - | ResNet-101 | 49.0 | 71.1 | **55.7** | **46.8** | **56.9** |
| Mask2Former[Cheng et al., 2022] | Mask2Former-VIS | ResNet-101 | 49.2 | **72.8** | 54.2 | - | - |
| MCMPG(*w/o MP-Graph*) | Mask2Former-VIS | ResNet-101 | 48.5 | 65.7 | 53.0 | 43.7 | 54.0 |
| MCMPG(*w MP-Graph*) | Mask2Former-VIS | ResNet-101 | **50.5** | 70.3 | 55.0 | 45.0 | 55.8 |

### 4.4.5 Ablation studies

We conduct ablation studies on the DAVIS 2017 *val* to validate the effectiveness of the proposed MCMPG in different settings and discuss the challenge in the task.

**Key frame Count and Key frame Clip Size.** In Table 4.5, we illustrate the effectiveness of MCMPG on combining and refining the proposals using different numbers of key frames and keyframe clip size on DAVIS 2017 *val*. With *MP-Graph* ($H = 3/5/7$), better segments that are obtained from merging proposals in the same clique lead to

performance improvement at every setting of key frames, which validates that the proposed approach provides significant performance improvement as discussed in the paper. Also, we observe that the performance is robust to different numbers of key frames and satisfactory with just 2 key frames, not requiring an excessive amount of key frames to obtain good performance on this dataset.

Table 4.5: Ablation study on DAVIS 2017 *val* on the influence of the number of key frames $K$ and the size of key frame clip $H$ in terms of $\mathcal{J}$ & $\mathcal{F}$-Mean.

| $K$ \ $H$ | 1 | 3 | 5 | 7 |
|---|---|---|---|---|
| 1 - baseline | 71.3 | 73.8 | 74.7 | 74.2 |
| 2 | **72.7** | **78.4** | **77.6** | **77.0** |
| 3 | 72.1 | 78.2 | 77.5 | 76.9 |
| 4 | 72.4 | 77.9 | 77.2 | 76.3 |
| 5 | 71.9 | 76.5 | 76.3 | 74.2 |

**Objects from unseen categories and different sizes.** Instance segmentation algorithms (usually pre-trained on COCO) can detect objects in unseen categories if they resemble the shapes or textures of seen categories. In order to show this, we manually located 18 objects from non-COCO categories and 48 objects from COCO categories in DAVIS *val*. Table 4.6 shows that MCMPG performs well on objects of unseen categories. It also includes a breakdown of our results based on object sizes. Objects smaller than 4100 pixels (1% of the DAVIS image resolution 480p) are considered small. Our performance is indeed lower on small objects which are harder to track and refine.

Table 4.6: Ablation Study on DAVIS 2017 *val* for unseen / seen categories and instances in different sizes

| type of objects | No. of objects | $\mathcal{J}$ & $\mathcal{F}$-Mean | $\mathcal{J}$-Mean | $\mathcal{F}$-Mean |
|---|---|---|---|---|
| unseen categories | 18 | 77.5 | 75.0 | 79.9 |
| seen categories | 48 | 78.7 | 75.5 | 82.0 |
| small objects | 11 | 63.8 | 61.4 | 66.2 |
| large objects | 55 | 81.3 | 78.2 | 84.4 |
| all | 66 | 78.4 | 75.4 | 81.4 |

**Different strategies of proposal generation from *MP-Graph*.** One simple baseline is to use the connected components from *MP-Graph* as the proposals. We test it on the DAVIS UVOS task shown in Table 4.7, and it results in a $\mathcal{J}$ & $\mathcal{F}$-Mean of 72.0% on the *val*

set, significantly worse than our 78.4% and even worse than the result without MCMPG (72.7%). We also compare the proposal quality in terms of *mIoU*, which drops 0.7% from the object segmentation on the key frame itself. This is because connected components can easily introduce extra noise from the neighboring frames' segmentation.

We also attempted to construct a 3-layer Graph Convolutional Network (GCN) to learn how to generate refined proposals from *MP-Graph*. The input graph to the network has the same edges as *MP-Graph*, with each node representing one pixel instead of a whole object mask as in *MP-Graph*. Each node has two features: "mask score on the pixel" and "objectness score". The network performs binary classification. GCN is applied to all pixels individually and updates each pixel with the pixels in the same location from their adjacent objects in *MP-Graph*. Finally, the NMS algorithm is used to eliminate duplicate proposals after all pixels are updated. It should be noted that the network is relatively small, with only 354 parameters. As demonstrated in Table 4.7, the mIoU on proposals is 2.6% better, and the $\mathcal{J}$ & $\mathcal{F}$-Mean is 2.4% better than without MP-Graph.

Table 4.7: Different strategies to obtain the key frame proposals on DAVIS 2017 *val*.

| Methods | mIoU (%) | $\mathcal{J}$ & $\mathcal{F}$ Mean |
|---|---|---|
| MCMPG *w/ MP-Graph* | **79.2** | **78.4** |
| *w/o MP-Graph* | 75.5 | 72.7 |
| connected-component *w/ MP-Graph* | 74.8 | 72.0 |
| GCN *w/ MP-Graph* | 77.1 | 75.1 |

### 4.4.6 Run-Time Analysis.

We report the run-time of each module in MCMPG in Fig. 4.4. The results are generated by using an NVIDIA Tesla V100 GPU. It shows that the process of generating proposals and improving them with the MP-Graph is very lightweight and takes minimal time to run. The most time-consuming part of the system is the semi-supervised VOS module. A limitation of using MCMPG in the UVOS/VIS tasks is that the VOS algorithm needs to track objects starting from multiple key frames.

In the STM algorithm, the encoder takes the object mask as *input*. Hence, with each new object mask, the backbone has to be run again, which is quite sub-optimal, especially for our approach which requires running tracking on a significantly larger amount of

proposals than the regular semi-supervised VOS task for which STM was designed.

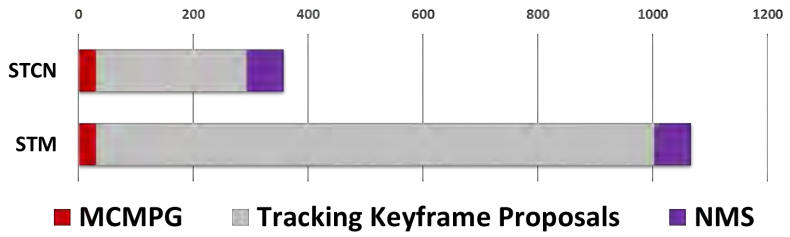



Figure 4.4: Run-time (in seconds) of MCMPG on DAVIS-UVOS 2017 *val* with 2 key frames. Note that MCMPG is fast (only **8.7**% or **2.9**% of the running time for STCN and STM VOS models, respectively) while improving the final tracking performance significantly. The bottleneck of the speed comes from an external tracking algorithm such as STM which requires re-running the backbone network for each proposal. Alternatively, one could use a newer tracking algorithm such as STCN where all the proposals can share the same backbone features, which would make the system much faster.

STCN [Cheng et al., 2021b] proposed to replace the memory encoder in STM with a lightweight encoder that does not require the mask as input. This improved both the speed and performance on the VOS task. For us, it implied that we will only need to run the encoder once. Thus, the speed of our system will be significantly faster without compromising performance if the system utilizes an STCN-type encoder, which we believe will be standard in the future. In Fig. 4.4, running UVOS on MCMPG proposals with STCN turns out to be at least 2.2× faster than STM.

### 4.4.7 Online UVOS with MCMPG

Here, we explore the online UVOS with MCMPG considering its value for tasks that require real-time understanding and tracking of objects within video streams, including robots.

In Fig.4.5, we illustrate the online setting with MCMPG for the task of UVOS. In the initial frame, we utilize MCMPG to segment objects. Then, we designate frames as key frames with a specified interval of *step*. For non-key frames, we adopt a VOS model, where each object has its own memory, to propagate objects forward. On key frames, we first generate object proposals using MCMPG and then merge them into the tracking

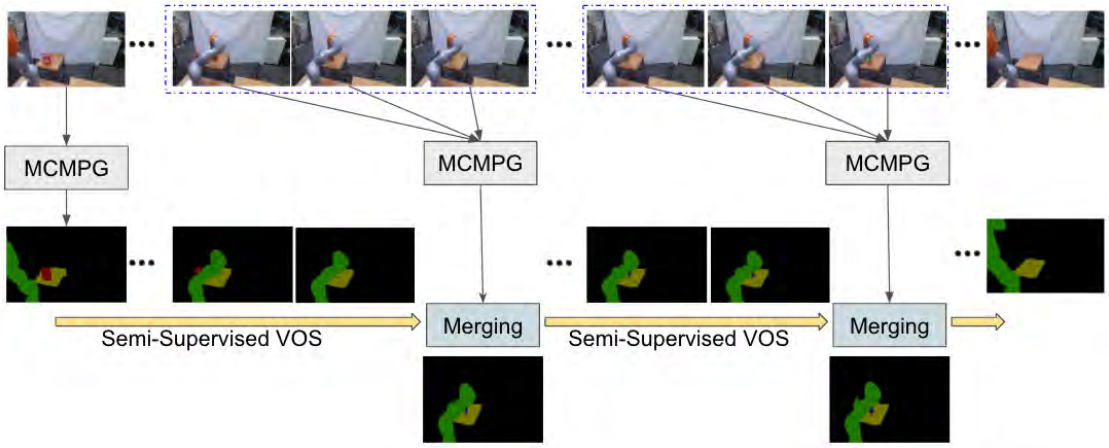


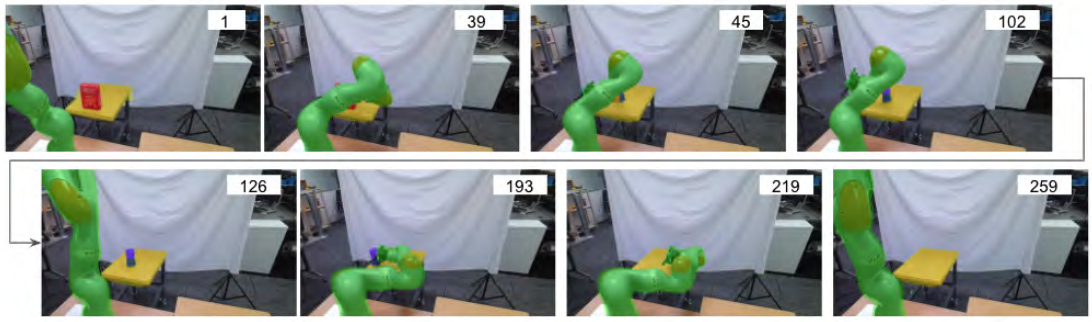

Figure 4.5: Online MCMPG : We segment objects in the initial frame using MCMPG and then track these objects until the next key frame. On key frames, we merge image-level segmentation from MCMPG with tracking results from VOS to incorporate new objects.

result. Consistent IDs are assigned to object segments matching previously tracked objects, while new IDs are assigned to those that have no previous matches.

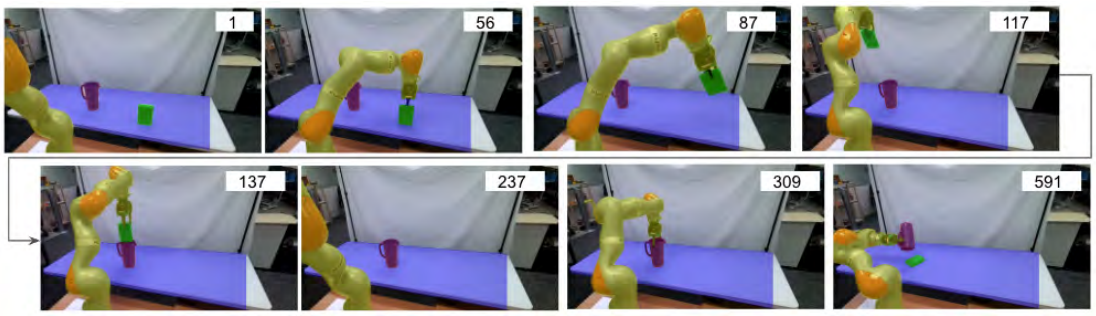
**Merging.** We adopt a greedy solution to merge the MCMPG segmentation $S^t$ with tracked objects $\hat{S}_o^t$ at time $t$. We first compute IoU between each pair of objects, one from $S^t$ and one from $\hat{S}_o^t$. Then we check the MCMPG segmentation in the decreasing order based on their objectness score. If a proposal $S_i^t$ from $S^t$ has an IoU larger than 0.5 with an object in $\hat{S}_o$, it is matched and we use $S_i^t$ as the output mask for the matched object considering the objects from near future is more accurate in most scenarios. If the IoU is less than 0.5, we add $S_i^t$ as a new object.

**Object Memory.** We introduce two variables: $N_o$ and $N_f$, to manage the object memory's maximum size, to specify the maximum number of objects to track and the maximum number of temporal frames to record. When there is a need to add a new object and the object memory is already full, we delete the inactive object that disappeared earliest from memory. Additionally, when key frames are encountered or when a fixed frame interval is reached, the object memory is updated. If the memory size exceeds the maximum temporal size, we delete the memory associated with the earliest frame.


Figure 4.6: Labeled objects in the training set.




(a)




(b)

Figure 4.7: Results of applying online MCMPG to robot manipulation tasks.

We apply the online setting to a robot application [Huang et al., 2023] focused on reason-

ing about object permanence. For segmentation, we utilize the SOLOv2 model [Wang et al., 2020c], and for temporal propagation, we employ the STM model [Oh et al., 2019]. To refine the SOLOv2 model, we annotate a total of 175 images for the object and environment segments. The environment segments encompass elements such as the robot arm, table, and shelf. The annotated objects are visually presented in Fig.4.6. We augment these annotations with data from the YCB-Video dataset [Xiang et al., 2017], resulting in a training dataset comprising 1174 images and 7105 object masks. To prevent potential dataset imbalances, we downsample the YCB-Video dataset. The segmentation model then underwent finetuning with a learning rate of 1e−6 over 100 epochs. Two illustrative examples from this use case are presented in Fig.4.7.

## 4.5 Quanlitative Results

We show the qualitative segmentation results of our approach on DAVIS-UVOS on **DAVIS 2017 *val.*** in Fig. 4.8, The segmentation results are overlayed on the input RGB sequence where different colors are used to indicate different object instances. Some qualitative segmentation results of our approach on DAVIS-UVOS 2019 *test-dev* are shown in Fig.4.9. Some qualitative segmentation results of our approach on YouTube-VIS 2019 *val* are shown in Fig.4.10.

## 4.6 Conclusion

In this chapter, we studied the task of refining key frame object proposals. We introduce a novel algorithm that aggregates object proposals in a local window, based on maximal cliques on a graph built from all proposals propagated to the key frame. The improved key frame proposals enable more robust and accurate propagation through a video sequence. Experiments demonstrate that the mask proposal refinement provides significant performance improvements over state-of-the-art methods in the DAVIS-UVOS and Youtube-VIS benchmarks across different backbones and instance segmentation algorithms. In the future, we would like to pursue applications of this algorithm in realistic object discovery tasks, such as in robotics and autonomous driving applications.

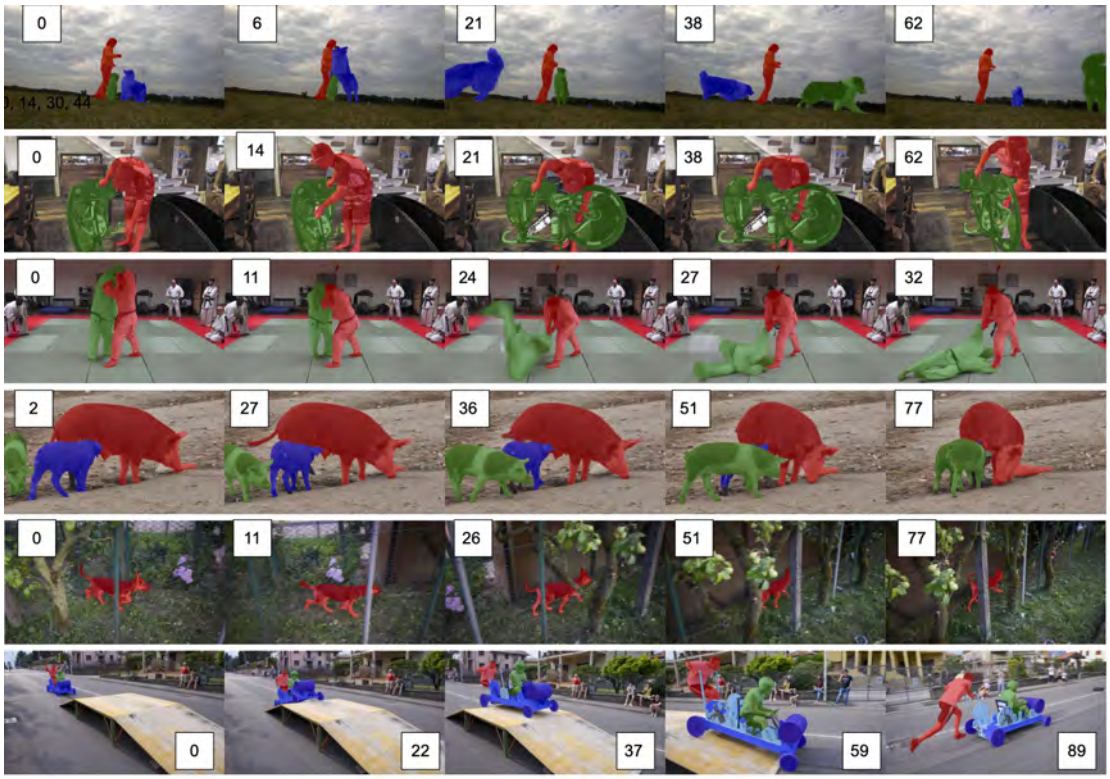



Figure 4.8: Qualitative results on three sequences from the DAVIS 2017 *val* set. We show frames that are sampled from challenging scenarios such as fast motion, background clutter, occlusions, and multi-object interaction

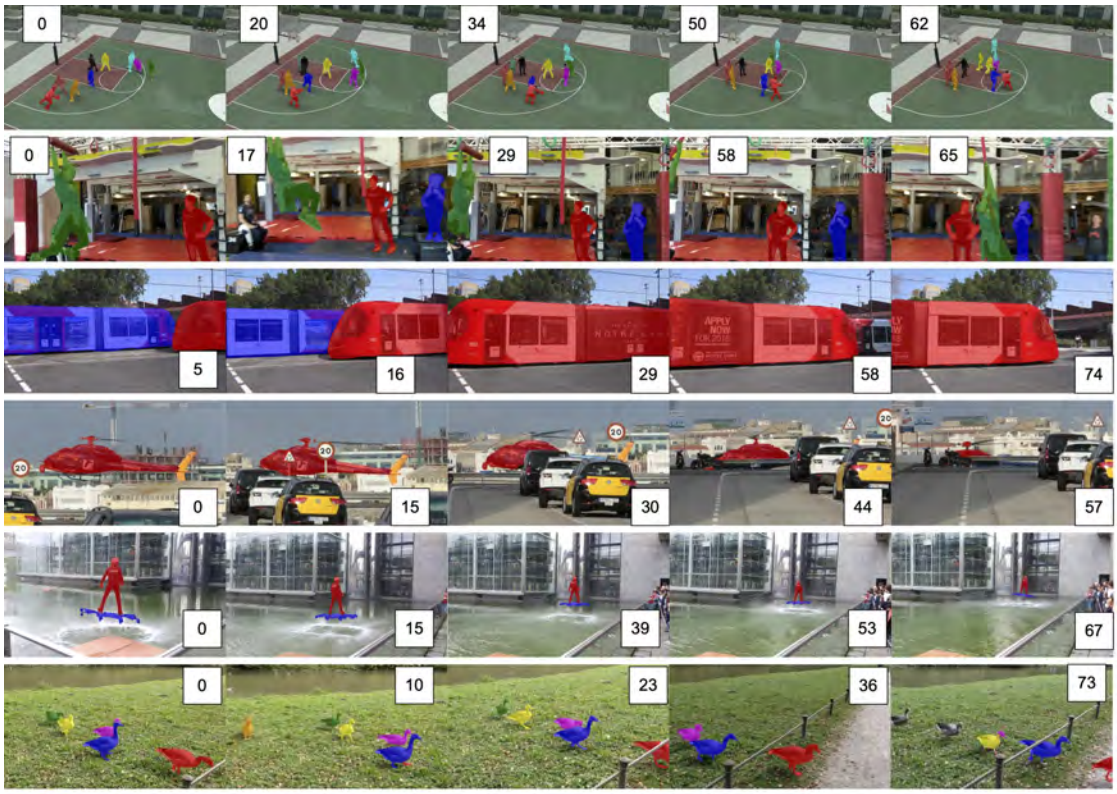



Figure 4.9: Qualitative results on sequences from DAVIS 2019 *test-dev*

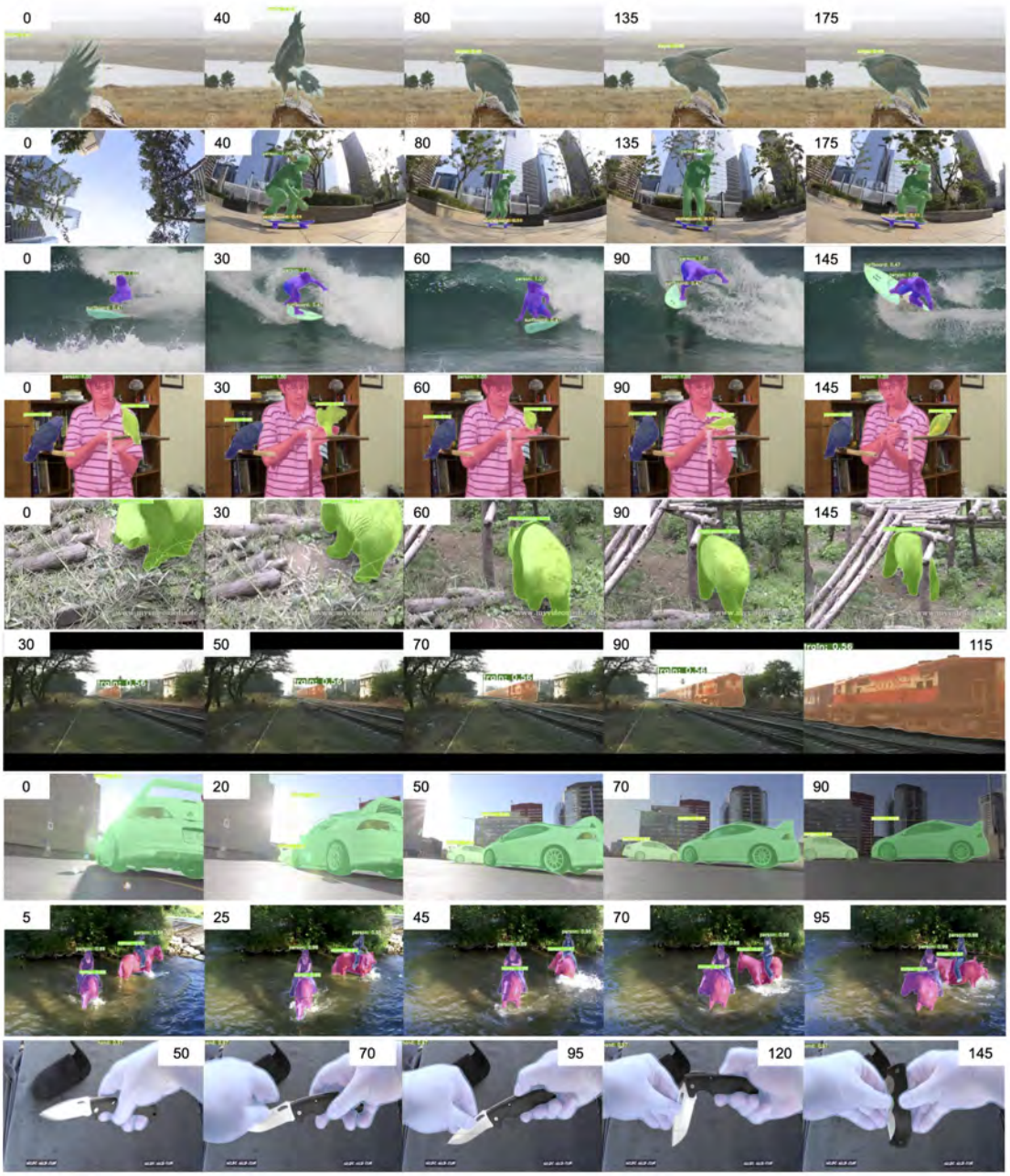



Figure 4.10: Qualitative results on sequences from YouTube-VIS 2019 *val*

# Chapter 5: Multimodal Understanding: Bridging Vision and Language

In this chapter, we take the concept of object discovery a step further by applying it to Vision-Language tasks, introducing new challenges related to identifying relationships between objects detected in images and leveraging objects that are contextually relevant to textual information. This extension leads us to the domain of content moderation (CM), where we propose an innovative model designed to efficiently fuse knowledge in an asymmetric manner.

## 5.1 Introduction

With the proliferation of multimodal social media and online gaming, user-generated content followed by recent AI-generated content (e.g., via DALL-E[Ramesh et al., 2022], GPT-3[Brown et al., 2020], ChatGPT[van Dis et al., 2023]) can spread across the internet at a faster rate than ever. While this enables free speech and facilitates information exchange, it comes with the risk of misuse for fake news [Nakamura et al., 2019, Vosoughi et al., 2018] and hate speech [Davidson et al., 2017, Schmidt and Wiegand, 2017].

Leaving harmful content on social platforms can lead to harmful consequences, but moderating the tremendous amount of user/AI-generated content on the platforms manually is infeasible due to the large scale and can be harmful to the mental health of human moderators. Therefore, automated content moderation (CM) systems are necessary. There has been extensive research on text-based content moderation [Vosoughi et al., 2018, Waseem and Hovy, 2016, Waseem et al., 2017]. Recently, there is a study on image-based pornographic content classification and sexual object detection tasks [Phan et al., 2022]. As social platforms allow the use of different modalities, unsafe multimodal content may evade detection by existing unimodal content moderation systems. Hence, multimodal harmful content detection benchmarks [Gomez et al., 2020, Kiela et al., 2020] have emerged followed by works [Das et al., 2020, Zhu, 2020] aiming to automatically

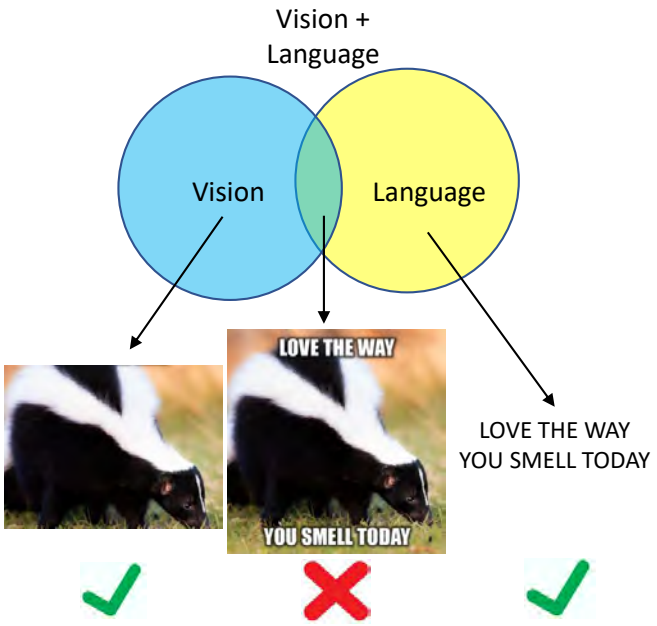



Figure 5.1: An example of a mean meme from Hateful Memes[Kiela et al., 2020] for illustrative purposes. The unimodal vision and language are both benign while the multimodal meme is sarcastic and mean. This is not an actual example of the CM dataset ⋆, which is hateful and would be distasteful to show here.

detect unsafe multimodal content, including child abuse material, violence, hate speech, sexual content, cyberbullying content, and disinformation [Banko et al., 2020].

One important form of multimodal content online is memes, which are a combination of image and short text. Understanding memes is a multimodal vision language (VL) task. As noted in previous studies [Gomez et al., 2020], offensive terms by themselves may not necessarily signify hate. It is the overall context that determines whether the intent is harmful or not. Fig. 5.1 shows an example of a mean meme, where the text by itself is just a compliment and the image also seems benign. However, when combining the two modalities the meme becomes sarcastic and mean. This example is for illustrative purpose only. For actual examples which are indeed hateful, please refer to supplementary. To combat the spread of harmful VL content such as hateful memes on social platforms, different VL datasets have been constructed: Facebook proposed a Hateful Memes Chal-

lenge and constructed a corresponding dataset [Kiela et al., 2020], which contains memes designed to evade detection by unimodal methods. MMHS150K [Gomez et al., 2020], a large-scale image-text pair dataset originated from Twitter postings, is proposed to benchmark hate speech detection in multimodal publications.

In this work, we approach multimodal (image + text) harmful content detection and propose a novel mixed-modal (a mix of multimodality and unimodality) CM model, Asymmetric Mixed-Modal Moderation (AM3). Image and text are intrinsically different in the information they convey: text is more structured and semantically at a higher level (usually describing the main components of an image while overlooking the subtle details, especially the background). On the other hand, image is unstructured: it is composed of pixels that can provide more low-level details of the context. For example, an image caption is likely to focus on the foreground or the objects of interest in the image. It may contain semantic details like the color or shape of the objects, but is unlikely to cover all the details, especially those in the background. We call this ***asymmetry in semantics*** of VL content. To address this asymmetry, we propose a novel fusion transformer architecture that attempts to maintain the unique knowledge in each modality while fusing the information from the asymmetric semantic levels. As shown in Fig. 5.1, the knowledge learned from the joint multimodality should contrast that from each unimodality due to this asymmetry in semantics. Sometimes this subtle missing part in unimodality is the determinant for content moderation decisions. We name the discrepancy in the information conveyed by multimodality and each unimodality ***asymmetry in modalities***. To tackle this challenge, we propose a novel contrastive loss between the representation learned from multimodality versus each unimodality. In order to learn domain-specific knowledge, we mix multimodal dataset with additional unimodal CM datasets in pretraining, similar to [Li et al., 2021b]. We call this ***asymmetry in data*** as either modality may be missing in the data, so that the conventional multimodality (each sample contains both modalities) setup becomes mixed-modality (mix of multimodality and unimodality, where each sample may contain both modalities or each unimodality). By including the unimodal CM dataset in pretraining, AM3 learns the domain-specific knowledge which helps the model adapt to the downstream tasks. Hence, the downstream CM task performance is improved.

We summarize the main contributions of work below,

- **Asymmetry in semantics**: We propose a novel fusion transformer architecture to fuse different modalities asymmetrically. It enhances the unique knowledge in each modality while effectively fusing the information from the asymmetric semantic levels.
- **Asymmetry in modalities**: We design a novel contrastive loss to squeeze out the distinct knowledge that only exists in multimodality, which is essential in multimodal content moderation.

## 5.2 Related Work

**Harmful Content Detection.** As social media platforms have grown, so have the challenges of content moderation. These challenges have pushed platforms toward automated content moderation as a necessary tool for detecting harmful content. Initially, most of the works are on text [Baly et al., 2018, Das et al., 2020]. In [Davidson et al., 2017], 25K Tweets are collected and annotated based on whether they contain hate speech keywords or have implicit hate. Logistic regression and SVM are tested to automatically detect hate speech. Besides web crawling data, a large scale machine generated dataset of toxic and benign text statements is provided in [Hartvigsen et al., 2022] using GPT-3 [Brown et al., 2020]. These labels are then validated by human annotators, and over 95% of the generated toxic labels are legitimately toxic. Over time, images and videos have gained more attention as visual contents are easier to consume and more popular to spread. A large scale dataset for pornographic visual content classification is given in [Phan et al., 2022]. In [Soldner et al., 2019], videos of conversations are collected as a benchmark for deception detection. Recently, multimodal harmful content detection has attracted more attention. Facebook proposed a Hateful Memes Challenge [Kiela et al., 2020], where each image is associated with a short text. The winner of the challenge [Zhu, 2020] outperforms the other competitors significantly by leveraging external labels such as race, age, and entity. Following the same practice, DisMultiHate [Lee et al., 2021] further improves the performance by disentangling target entities in multimodal memes. Hate-CLIPper[**?**] proposes a method of intermediate fusion to alleviate the ambiguity alignment between image and text representations. The importance of each modality in the Hateful Memes dataset and the robustness of SOTA multimodal classfication algorithms are investigated

in [Ma et al., 2022].

**Vision-Language Pretraining.** Recent years have witnessed rapid progress in vision-language pretraining (VLP) where vision and language modalities are jointly encoded using a fusion model. The success of BERT [Vaswani et al., 2017] inspired many follow-up multimodal fusion models, such as VL-bert[Su et al., 2019], VinVL[Zhang et al., 2021], SimVLM [Wang et al., 2021d], and OFA [Wang et al., 2022], where the text features are concatenated with vision features from image encoder and then fused by BERT or its variants. Besides the masked language modelling (MLM) loss used in BERT pretraining, various loss functions targeting multimodal feature fusion are used, e.g., image-text matching (ITM) loss, region-of-interest (RoI) classification loss. Most of these works learn the joint representation of vision and language through a symmetric feature encoding and fusion process. For example, VL-BERT[Su et al., 2019] constructs the multimodal inputs symmetrically where every multimodal feature map has the same components, i.e., text embedding, visual embedding, segment embedding, and positional embedding. Each text embedding is associated to the visual embedding of the entire image while each RoI visual embedding is associated with a dummy text embedding. This simple symmetric architecture enables the fusion of multiple modalities. However, each text token only contains a subset of the entire image. Linking the entire image embedding to it may introduce noise that decreases the performance. On the other hand, the dummy text embedding does not contain any meaningful information. VinVL[Zhang et al., 2021] simply concatenates text embeddings with the object label embeddings as well as RoI visual embeddings before feeding into the fusion transformer. It assumes that the text embeddings and the visual embeddings share the same (symmetric) level of knowledge and processes them equally.

Recent works on VL foundation models show that dual-encoder architectures can learn strong representation through contrastive objectives on large scale noisy image-text pairs [Pham et al., 2021, Radford et al., 2021, Yuan et al., 2021]. Florence [Yuan et al., 2021] developed a unified contrastive objective [Yang et al., 2022] in VLP that enables the model to be adapted for a wide range of vision and VL tasks. Flamingo [Alayrac et al., 2022] utilizes an 80B-parameter language model frozen in training and fused with a vision encoder. The huge capacity of Flamingo enables the state-of-the-art performance for few-shot learning.

Our method shares numerous ideas of the previous works mentioned above. However, we pivot to looking at the multimodal content moderation task from an asymmetric angle, both in architecture and data, and target mixed-modality (both multimodal and unimodal) downstream CM tasks. We exploit the discrepancy in vision, language, and multimodal VL pairs, to improve the model capability and training.

## 5.3 AM3

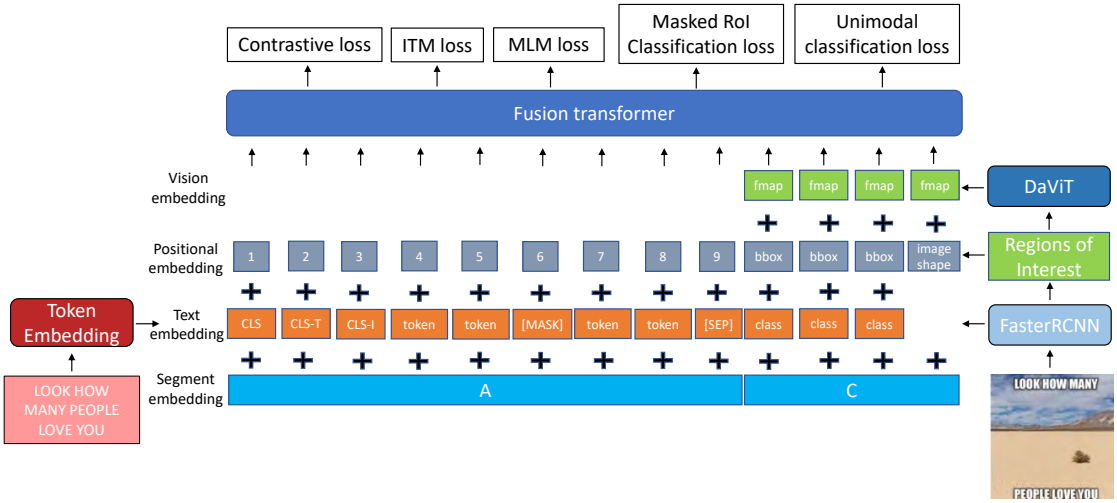



Figure 5.2: Architecture overview. It shows an example of the pretraining of AM3 with a $(T, I)$ input. For text inputs, we sum up text embeddings, positional embeddings, and segment embeddings. Visual inputs consist of text embeddings from detected objects' category labels, the feature map from the vision encoder, positional embeddings, and segment embeddings. The positional embeddings of visual inputs are computed based on object bounding boxes so that they are permutations invariant to object order.

In this section, we introduce a novel fusion transformer architecture pre-trained on both VL datasets and unimodality datasets. To tackle the asymmetry in the semantics of CM VL content, we construct vision and language embeddings differently to encourage the model to capture essential knowledge in each modality. Meanwhile, we follow [Zhang et al., 2021] to utilize the object labels from detection as anchors to bridge the language with the corresponding image RoI features. Due to the asymmetry in modality, there is unique knowledge that only exists in the intersection of both modalities. To drive the

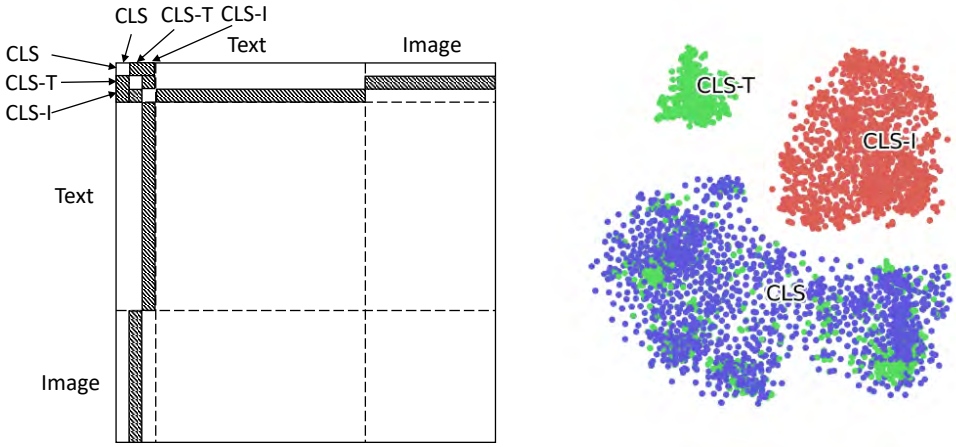
model to obtain an understanding of this, we introduce a novel contrastive loss, *Cross-modality* Contrastive Loss, as part of our pre-training tasks. We use an asymmetric mix of multimodal datasets as well as domain-specific unimodal datasets in pretraining, where a domain-specific classification loss is included to improve downstream task performance.

### 5.3.1 Model Architecture for Asymmetry in Semantics

Fig. 5.2 illustrates the overview architecture of AM3. The model takes mixed modality input: $(T, I)$, $(T)$, or $(I)$, where $T$ represents the text if it exists, and $I$ is the image if it exists. Unlike previous works that try to unify the feature encoding process from both vision and language modalities, we construct the text inputs and visual inputs to the fusion transformer asymmetrically. $T$ is first tokenized through a tokenizer and then fed to a token embedding layer whose outputs are added to positional embeddings and segment embeddings to generate the sequence of linguistic embeddings of text $\mathbf{w}$. The image $I$ is processed as follows: we first use an object detection model to detect objects existing in the image. We also include a bounding box for the entire image (so the bounding box becomes the shape of the image) without an object category associated. For each object, its category label will go through the same token embedding layer as the text input to obtain its text embedding. Its bounding boxes are transferred to the positional embeddings of the RoI through a linear layer. This makes the positional embeddings permutation invariant to the input order of the objects. The visual feature of each RoI is encoded through a feature extractor. We then sum up the text embeddings of object labels, positional embeddings from object bounding boxes, the features from the RoIs, and segment embeddings to obtain the sequence of visual embeddings $\mathbf{v}$. The concatenated pair of $(w, v)$ is fused through a fusion transformer.

### 5.3.2 Cross-modality Contrastive Loss for Asymmetry in Modalities

***Cross-modality* Contrastive Loss.** Due to the asymmetry in modalities, the capability of learning the unique knowledge only existing in the intersection of different modalities is critical to content moderation tasks, as demonstrated in Fig. 5.1. Therefore,




(a) (b)

Figure 5.3: (a): Modified attention mask for contrastive learning. Attention between image tokens (including CLS-I) and CLS-T tokens are masked out, and vice versa. (b): Visualization of the 3 CLS tokens from Hateful Memes after t-SNE[Gisbrecht et al., 2015] reduction.

we propose the *cross-modality* contrastive loss, as given in Equation(5.1):

$$L_{con} = \max(0, \cos(f_{VL}, f_V)) + \max(0, \cos(f_{VL}, f_L)) \tag{5.1}$$

where $\cos(\cdot)$ is the cosine similarity function. $f_{VL}$, $f_V$, and $f_L$ are the CLS output tokens from the fusion transformer for image + text, image only, and text only, respectively. As shown in Fig. 5.2, 3 CLS tokens are added to the fusion transformer input. The CLS token is designed to summarize the multimodal knowledge from all tokens while the CLS-I and CLS-T tokens only extract information for vision and language tokens, respectively. By summing up the similarity between $f_{VL}$ vs. $f_V$, and $f_{VL}$ vs. $f_L$ in the contrastive loss, we push the joint multimodal representation away from the unimodal representations, for the asymmetry in modality, forcing the model to learn the distinct semantic knowledge only in the intersection of both modalities. Fig. 5.3(a) illustrates how the attention mask is modified to compute the multimodal and unimodal representations: the attention between CLS and CLS-I/CLS-T as well as between CLS-I and CLS-T tokens are masked out to prevent information leak among different representations. The

attention between the CLS-T token and all image tokens are also masked out, and vice versa. In this way, as displayed in Fig. 5.3(b), the CLS output token summarizes the fused information learned from both modalities while CLS-I and CLS-T only contain unimodal information in vision and language, respectively.

**Binary classification on domain-related datasets.** To help the model effectively adapt to the new domain (CM in our case) when porting a generic model to a specific domain, we include a domain-specific classification loss ($L_{domain}$) in our pretraining objectives. We collect several content moderation related unimodality datasets discussed in Sec.5.4.1 into the pretraining corpus. When an input is from these datasets, its CLS output token is projected through a linear layer to predict if the input is harmful or not. We show that this domain-specific classification loss improves downstream performance on CM benchmarks. The domain-specific classification loss is:

$$L_{domain} = -E_{f_{VL}}[\log P(c_d|f_{VL})] \tag{5.2}$$

where $c_d$ is the domain category label and $f_{VL}$ is the fusion transformer output of the CLS token. In our experiment, we set $c_d = 1$ for harmful inputs while $c_d = 0$ for safe ones. For inputs from generic multimodal VL datasets, we set $c_d = -1$ so that they are ignored in the domain-specific classification task.

As shown in Fig. 5.2, there are 3 additional pretraining objectives for multimodal fusion: the Masked Language Modeling loss ($L_{mlm}$ on the text tokens similar to [Huang et al., 2020, Kim et al., 2021, Su et al., 2019, Zhang et al., 2021], the Image-Text Maching loss ($L_{itm}$) which is computed on the CLS token of joint modalities same as [Huang et al., 2020, Kim et al., 2021], and the Masked RoI classification loss ($L_{roi\text{-}cls}$) similar to [Su et al., 2019].

$$L_{mlm} = -E_{\mathbf{f}}[\log P(\mathbf{t_m}|\mathbf{f})] \tag{5.3}$$

$$L_{itm} = -E_{f_{VL}}[\log P(c|f_{VL})] \tag{5.4}$$

$$L_{roi-cls} = -E_{\mathbf{f}}[\log P(\mathbf{c_v}|\mathbf{f})] \tag{5.5}$$

Overall, our pretraining objective consists of terms as in Equation(5.6):

$$Loss = \alpha\ L_{con} + \beta\ L_{mlm} + \gamma\ L_{itm} + \lambda\ L_{roi\text{-}cls} + \omega\ L_{domain} \tag{5.6}$$

where $\alpha$, $\beta$, $\gamma$, $\lambda$, and $\omega$ are coefficients to balance the various objectives. We set $\lambda$ to 0.2 and all the other coefficients to 1 throughout the experiments.

## 5.4 Experiments

In this section, we first introduce the implementation details. We then discuss the results on downstream CM tasks. Finally, we show an ablation study on the proposed method.

### 5.4.1 Implementation Details

**Pretraining.** As shown in Fig. 5.2, following [Li et al., 2020a], we use pretrained FasterRCNN[Ren et al., 2015] for object detection, but other object detection models, like Yolo [Redmon et al., 2016], can be used as well. We use DaViT[Ding et al., 2022] as the vision encoder, which encodes the RoIs detected by FasterRCNN into vision embeddings. Both FasterRCNN and DaViT are frozen during training. We use $BERT_{base}$ $(Layers = 12, Hidden\ size = 768, Attention\ heads = 12)$ for text embedding and fusion transformer. The model is initialized with pretrained $BERT_{base}$ parameters and optimized using the AdamW optimizer with a base learning rate of $10^{-5}$ and weight decaying of $10^{-2}$. The learning rate was warmed up for 100 training steps and then decayed linearly to zero for the rest of the training. We use a probability of 0.15 in MLM and Masked RoI classification random masking and 0.5 in ITM random replacing. We assign segment tokens 'C' to all visual features. For captions, we set segment tokens to 'A', while for questions and answers, we use 'A' and 'B', respectively. We pretrain the model for $500K$ steps with a batch size of 6144 on 72 NVIDIA V100 GPUs.

**Pretraining corpus:** We construct our pretraining corpus based on three types of datasets: generic VL multimodal datasets, CM language datasets, and a CM vision dataset.

- **Generic VL multimodal datasets.** We build our corpus from image captioning and visual question-answer datasets, including COCO [Lin et al., 2014], Conceptual Captions (CC3M) [Sharma et al., 2018], SBU captions [Ordonez et al., 2011], Flickr30k [Young et al., 2014], CC12M [Changpinyo et al., 2021], Open-Images [Kuznetsova et al., 2020], GQA [Hudson and Manning, 2019], and VG-QAs datasets. Following [Zhang et al., 2021], machine-generated captions are used for the Open-Images dataset, while captions and question-answer segments are used as text inputs for the other datasets.

- **CM language datasets.** We use 4 language datasets in CM domain: ToxiGen [Hartvigsen et al., 2022], Jigsaw [Zaheri et al., 2020], HateXplain [Mathew et al., 2021], and ImplicitHate [ElSherief et al., 2021], where we preprocess data so each sample has a harmful or safe label. We use *train* sets in pretraining for ToxiGen, Jigsaw, and HateXplain to avoid data leakage. For text samples without images, we pad [PAD] to vision embeddings. A text classification head predicts the label using the domain-specific classification objective.

- **CM vision dataset.** We use LSPD (Large-Scale Pornographic Dataset) [Phan et al., 2022] image dataset for CM vision task. Similar to the CM language datasets, we use the *train* set with binary annotation for pretraining and pad [PAD] tokens to the text inputs for the fusion transformer. We use an image classification head to predict binary labels using the same classification objective.

**Downstream finetuning.** All the CM downstream tasks introduced in Sec.5.4.2 are formulated as classification tasks. The output token on CLS from the fusion transformer is fed into the classification head and trained with cross-entropy loss. Hyperparameters including batch size, learning rate, and training epochs are searched for each task. All classification heads are implemented with an MLP consisting of 2 linear layers and 1 ReLU layer.

**Downstream inference.** In each task(Sec. 5.4.2), we utilize the finetuning model and take the classification result from the CLS token as output, the model is named as **AM3**. On the downstream tasks, we conducted 5 experiments with random seeds, reporting their mean and standard variation across the multiple fine-tuning models. On CM VL tasks, we assessed the model's ability to learn cross-modal knowledge using the best

model with unimodal input as **AM3-text** and **AM3-image** respectively. Additionally, we combine the two unimodal results by taking the maximum classification probability as the predicted outcome and refer to it as **AM3-max**.

### 5.4.2 Downstream Datasets

To validate the effectiveness of AM3, we adapt the pre-trained model over the content moderation tasks in different modalities.

For CM VL tasks, we adopted Hateful Memes, MMHS150K, and Fakeddit datasets.

- **Hateful Memes [Kiela et al., 2020].** The Hateful Memes dataset consists of more than 10,000 memes, some of which are specially designed so that the text phrases and images are benign when considered separately, but hateful when combined. Therefore, the typical unimodal methods cannot yield good performance on them. To compare with prior works[Lee et al., 2021, Zhu, 2020], we use 2 different setups: (1) we finetune our model on the *train* set and evaluate on the *dev seen* set. (2) We finetune our model on the combination of *train* and *dev unseen* sets and evaluate on the *test unseen* set. The task uses the Area under Receiver Operating Characteristic curve (AUROC) and accuracy metrics.

- **MMHS150K [Gomez et al., 2020]** The MMHS150K dataset is based on Twitter data consisting of both images and text. We perform binary classification to decide whether a sample is hate or non-hate. We finetune the *train* and *val* sets and evaluate on the *test* set using F1-score, AUROC, and accuracy metrics.

- **Fakeddit[Nakamura et al., 2019].** The Fakeddit dataset is a large-scale multimodal fake news dataset that consists of over 1 million submissions from Reddit, a social news and discussion website where users can post submissions on various subreddits. 2-way, 3-way, and 6-way labels are provided for each sample. We follow the official dataset partition to only use multimodal samples. We focus on the 2-way classification and finetune our model on the *train* set. We compute accuracy on the *val* and *test* sets.

For CM text tasks, we use ToxiGen, HateXplain, and Jigsaw datasets.

- **ToxiGen [Hartvigsen et al., 2022].** ToxiGen is a machine-generated dataset using the massive pretrained language model GPT-3[Brown et al., 2020]. The dataset is designed to focus on creating hard-to-classify implicit abusive content in 13 minority groups. We use its *train* and *test* sets. The objective of the task is to predict if each sample is toxic or not and it is evaluated with AUROC.

- **HateXplain [Mathew et al., 2021].** The HateXplain dataset is constructed by collecting posts from Twitter and Gab for research on Explainable Hate Speech Detection. The task is evaluated using AUROC, accuracy, and F1-score.

- **Jigsaw [Sahoo et al., 2022].** The Jigsaw dataset is created using comments from Civil Comments for researchers to develop models to recognize toxicity and minimize this type of unintended bias with respect to mentions of identities, including gender, sexual orientation, and religious identity. We use the *train* and *test-public* splits for training and testing, respectively. AUROC is computed for evaluation.

We use LSPD for CM vision task.

- **LSPD [Phan et al., 2022].** LSPD is constructed for visual pornography classification with 5 categories: porn, hentai, drawing, sexy, and non-porn. We followed the porn/non-porn binary classification approach as [Phan et al., 2022], where the classes 'Hentai' and 'Porn' are grouped as 'porn', while all other classes were labeled 'non-porn' in the binary setting. To evaluate algorithms, accuracy, precision, and recall are measured.

Table 5.1: Comparisons to the state-of-the-art methods on Hateful Memes.

| | Dev seen | | Test unseen | |
|---|---|---|---|---|
| Method | AUROC | Accuracy | AUROC | Accuracy |
| ERNIE-VIL[Yu et al., 2021] | 78.7 | 69.0 | - | - |
| Uniter[Chen et al., 2020a] | 78.0 | 68.6 | 79.1 | 74.1 |
| VILLA[Gan et al., 2020] | 78.5 | 71.2 | 80.0 | 75.1 |
| VL-BERT[Su et al., 2019] | 78.8 | 71.4 | 79.5 | 74.5 |
| DisMultiHate[Lee et al., 2021] | 82.8 | 75.8 | - | - |
| AM3-text(Ours) | 59.1 | 65.2 | 62.2 | 64.1 |
| AM3-image(Ours) | 44.8 | 63.0 | 60.3 | 63.1 |
| AM3-max(Ours) | 56.7 | 64.3 | 64.0 | 64.5 |
| AM3(Ours) | **83.18(±0.19)** | **75.98(±0.67)** | **83.35 (± 0.23)** | **76.95(±0.36)** |

Table 5.2: Comparisons to the state-of-the-art methods on MMHS150K.

| Method | AUROC | Accuracy |
|---|---|---|
| TKM[Gomez et al., 2020] | 73.1 | 68.2 |
| SCM[Gomez et al., 2020] | 73.2 | 68.5 |
| FCM[Gomez et al., 2020] | 73.4 | 68.4 |
| AM3-text(Ours) | 72.7 | 68.3 |
| AM3-image(Ours) | 52.0 | 52.5 |
| AM3-max(Ours) | 72.2 | 67.7 |
| AM3(Ours) | **74.2 (±0.09)** | **68.57(±0.79)** |

Table 5.3: Comparisons to the state-of-the-art methods on Fakeddit.

| Method | Val acc. | Test acc. |
|---|---|---|
| BERT+ResNet50[Nakamura et al., 2019] | 89.3 | 89.1 |
| MVAE+[Li et al., 2021a] | - | 90.1 |
| MDID[Kirchknopf et al., 2021] | 90.8 | 91.0 |
| EMAF[Li et al., 2021a] | - | 92.3 |
| AM3-text(Ours) | 82.23 | 82.41 |
| AM3-image(Ours) | 76.2 | 75.9 |
| AM3-max(Ours) | 83.1 | 83.3 |
| AM3(Ours) | **93.04(±0.21)** | **93.2(± 0.11)** |

### 5.4.3 Result Analysis

**Performance comparison on VL tasks:** (1) Results on the Hateful Memes comparing to the state-of-the-art approaches are shown in Table 5.1. We compare to the challenge winner's solutions discussed in [Zhu, 2020]: ERNIE-Vil[Yu et al., 2021], UNITER[Chen et al., 2020a], VILLA[Gan et al., 2020], and VL-BERT[Su et al., 2019], where the results are reproduced in [Lee et al., 2021]. We also compare to the state-of-the-art solution, DisMultiHate[Lee et al., 2021]. Furthermore, comparing to the baselines: AM3-text, AM3-image, and AM3-max, the AM3 is at least 30% better in terms of AUROC score, which verifies the efficiency of cross-modality understanding in the finetuned model. We adopt the same data augmentation method proposed in [Zhu, 2020]: we use Google Vision Web Entity Detection[LLC, 2021] to generate entity tags of each image used as part of the text input. (2) The MMHS150K result is shown in Table 5.2. We compare to the Feature Concatenation Model (FCM), the Spatial Concatenation Model (SCM), and Texual Kernels Model (TKM) discussed in [Gomez et al., 2020], where they are all

CNN + RNN models. It is worth noting that the AM3-text achieves a 72.7 AUROC score, indicating that the dataset predominantly relies on its text modality. Merely considering the maximum classification probability to combine the image modality results leads to a decrease in performance. (3) The Fakeddit result is shown in Table 5.3. In [Nakamura et al., 2019], the authors of the Fakeddit dataset utilize BERT and ResNet50 to encode language and vision, respectively, and then use max-pooling to fuse the multimodal features. MAVE[Khattar et al., 2019] is enhanced with BERT in [Li et al., 2021a], which is denoted as MAVE+ in the table. EMAF[Li et al., 2021a] is set up with $BERT_{large\_uncased}$ $(Layers = 24, Hidden\ size = 1024, Attention\ heads = 16)$, which is computationally more expensive than our method. The AM3 outperforms unimodal results, AM3-text, AM3-image, and AM3-max,by at least 12.0% in terms of accuracy. On all three datasets, our proposed method achieves the best performance against previous state-of-the-art works. This demonstrates the efficacy of the proposed asymmetric mixed-modal approach. It effectively captures the distinct information that only appears in the intersection of modalities, which is critical in CM decision-making.

**Performance comparison on text tasks:** AM3 can also handle unimodal CM tasks. We first evaluate our approach on the content moderation text tasks. (1) Results on ToxiGen Classification are listed in Table 5.4, where we compare to HateBERT[Caselli et al., 2020] and ToxDectRoBERTa[Zhou et al., 2021b] on the top-k only version of the dataset. (2) Results on HateXplain dataset are shown in Table 5.5. Adaptive Length Reduction (AdapLeR)[Modarressi et al., 2022] is a method based on BERT while optimizing inference speed. BERT, BERT-HateXplain, BERT-MLM, BERT-RP, and BERT-MRP are different BERT variants discussed in [Kim et al., 2022]. (3) Results on Jigsaw are shown in Table 5.6, where we compare to the Toxiciology[jig] and Limerobot[jig], the top 2 solutions on the leaderboard. On all three CM text datasets, our approach outperforms all the state-of-the-art language models, suggesting the efficacy of the proposed method on mixed-modal (both multimodal and unimodal) downstream CM tasks.

**Performance comparison on vision task:** Results on the LSPD dataset are presented in Table 5.8, where we compare them to the outcomes of different methods discussed in [Phan et al., 2022]. Our approach outperforms previous state-of-the-art methods in terms of accuracy and obtained the highest recall score. Similar to the CM text datasets, this shows the capability of our mixed-modal method on downstream CM vision task,

Table 5.4: Comparisons to the state-of-the-art methods on ToxiGen.

| Method | AUROC |
|---|---|
| ToxDectRoBERTa[Hartvigsen et al., 2022] | 85.0 |
| HateBERT[Hartvigsen et al., 2022] | 88.0 |
| AM3(Ours) | **91.52 (± 0.16)** |

Table 5.5: Comparisons to the state-of-the-art methods on HateXplain.

| Method | AUROC | Accuracy | F1 |
|---|---|---|---|
| AdaptLeR[Modarressi et al., 2022] | - | 68.6 | - |
| BERT[Subramaniam et al., 2022] | 85.1 | 68.9 | 68.2 |
| BERT-HateXplain[Mathew et al., 2021] | 85.1 | 69.8 | 68.7 |
| BERT-MLM[Kim et al., 2022] | 85.4 | 70.0 | 67.5 |
| BERT-RP[Kim et al., 2022] | 85.3 | 70.7 | 69.3 |
| BERT-MRP[Kim et al., 2022] | 86.2 | 70.4 | 69.9 |
| AM3(Ours) | **88.25 (±0.25)** | **81.17(±0.45)** | **80.37(±0.42)** |

Table 5.6: Comparisons to the state-of-the-art methods on Jigsaw.

| Method | AUROC |
|---|---|
| Limerobot[jig] | 94.7 |
| Toxiciology[jig] | 94.7 |
| AM3(Ours) | **95.76(±0.27)** |

benefiting from a richer representation space with the mixed-modality pretraining.

Table 5.7: Ablation study of mixed-modality and *cross-modality* contrastive loss.

| | | | Hateful Memes | | MMHS150K | |
|---|---|---|---|---|---|---|
| Text dataset | Vision dataset | *Cross-modality* Contrastive Loss | AUROC | Accuracy | AUROC | Accuracy |
| no | no | no | 80.28(±0.18) | 74.28(±0.29) | 71.91 (± 0.02) | 67.44(±0.05) |
| no | no | yes | 81.18(±1.0) | 74.82(±0.51) | 72.76(±0.08) | 68.21(±0.05) |
| no | yes | no | 80.65(±0.2) | 73.82(±0.36) | 72.52(±0.05) | 68.03(±0.09) |
| no | yes | yes | 81.49(±1.25) | 74.98(±0.46) | 72.98(±0.13) | 68.18(±0.05) |
| yes | no | no | 82.39(±0.08) | 76.38(±0.02) | 72.79(±0.21) | 68.70(±0.09) |
| yes | no | yes | 82.65(±0.2) | 75.92(±0.17) | 73.36(±0.07) | 68.62(±0.09) |
| yes | yes | no | 82.44(±0.23) | 75.5(±0.31) | 72.85(±0.10) | 68.45(±0.07) |
| yes | yes | yes | **82.94(±0.15)** | **76.45(±0.36)** | **73.96(±0.08)** | **68.73(±0.05)** |

Table 5.8: Comparisons to state-of-the-art methods on LSPD for binary classification.

| Method | Accuray | Precision | Recall |
|---|---|---|---|
| Mask-RCNN | 86.70 | **98.33** | 88.00 |
| YOLOv4 | 92.59 | 97.03 | 87.86 |
| SSD | 85.32 | 94.11 | 85.64 |
| Cascaded Mask RCNN | 92.62 | 95.01 | 89.95 |
| CNN classifier | 87.22 | 84.86 | 90.59 |
| AM3(Ours) | **92.86(±0.03)** | 92.85 (±0.15) | **92.73(±0.14)** |

Table 5.9: Ablation study of fusion architecture design.

| | Hateful Memes | | MMHS150K | |
|---|---|---|---|---|
| Architecture | AUROC | Accuracy | AUROC | Accuracy |
| $arch_{VL-Bert}$ | 80.26(±0.38) | 75.72(±0.62) | 73.31(±0.15) | 68.54(±0.06) |
| $arch_{vinVL}$ | 80.53(±0.23) | 75.49(±0.17) | 73.39(±0.12) | 68.4(±0.07) |
| $arch_{bbox-position}$ | 80.74(±0.29) | 75.0(±0.09) | 73.43(±0.21) | 68.4(±0.11) |
| AM3(Ours) | **82.94(±0.15)** | **76.45(±0.36)** | **73.96(±0.08)** | **68.73(±0.05)** |

### 5.4.4 Ablation Studies

We selected Hateful Memes and MMHS150K for the ablation study of different design choices. To accelerate the analysis, all ablations are performed on a smaller pretraining corpus (Flickr30k, SBU, and COCO), and we pretrain our model for $50K$ iterations.

**Model Architecture.** To understand the effect of our proposed asymmetric fusion transformer, we create two fusion transformer variants following VL-Bert ($arch_{VL-Bert}$ [Su et al., 2019]) and vinVL ($arch_{vinVL}$ [Zhang et al., 2021]), two symmetric fusion designs. Specifically, $arch_{VL-Bert}$ constructs the multimodal embeddings symmetrically so that each text embedding adds to the visual feature of the entire image while each RoI visual embedding adds to a text embedding of a dummy token. $arch_{vinVL}$ creates multimodal embeddings for fusion transformer by simply concatenating the text embeddings from text input and object detection labels, along with visual embeddings. As shown in Table 5.9, our proposed asymmetric fusion architecture outperforms both symmetric designs, indicating the efficacy of our asymmetric fusion architecture in response to the *asymmetry in semantics*.

**Vision Position Embedding from Bounding Box.** To validate the effectiveness of using bounding boxes for positional embeddings, we created a variant using the counting index of the tokens for positional embeddings (used in [Su et al., 2019, Zhang et al., 2021]). As shown in Table 5.9, positional embeddings generated from bounding box captures the ordering information in image (permutation invariant to the input order). Therefore, it achieves a better performance.

***Cross-modality* Contrastive Loss.** As shown in Table 5.7, the average baseline scores on Hateful Memes and MMHS150K are 80.28% and 71.91%, respectively, measured in terms of AUROC. By adopting the *cross-modality* contrastive loss, the score is improved by +1.1% on Hateful Memes and +1.2% on MMHS150K. The significant improvements show that our approach to the *asymmetry in modalities* has a strong capability to capture distinct knowledge from the intersection of different modalities.

**Pretraining on Unimodal CM Datasets.** Table 5.7 shows the result $w/$ and $w/o$ the unimodal CM datasets in the pretraining corpus. Using the CM text datasets improves the task scores by +2.6% and +1.2% from baseline, respectively. Using the CM image datasets improves the score by +0.5% and 0.8%, respectively. This shows that introducing *asymmetry in data* into the pretraining stage, with the datasets relevant to the domain, is effective and can improve downstream tasks by a significant margin.

**Combination of *Cross-modality* Contrastive Loss and Unimodal CM Datasets.** As shown in Table 5.7, utilizing the CM text dataset and the CM vision dataset together leads to further improvement (+0.6% on Hateful Memes and +0.8% on MMHS150K) in comparison to the best score when using either CM text dataset or CM vision dataset. Adding *cross-modality* contrastive loss on top of the unimodal CM text and vision datasets further improve the performance: when enabling all of these components, we achieve the highest average AUROC score of 82.94% for Hateful Memes and 73.96% for MMHS150K. It indicates the efficacy of our proposed method.

## 5.5 Quanlitative Results

[CONTENT WARNING] This section includes visual examples of hateful content, which may be offensive to some readers.

### 5.5.1 Successful examples

We show 3 groups of successful examples in Fig. 5.4, 5.5, and 5.6.

- Fig.5.4 presents *examples that are not offensive in multimodality but are hateful on unimodality*.
- Fig.5.5 shows *the correctly detected offensive examples that are offensive in unimodality*.
- Fig.5.6 displays *the correctly detected offensive examples that are only offensive in multimodality*.

The examples indicate that accurately predicting hate or non-hate is more complex than just mapping the two modalities. For instance, the text in Fig.5.6(n) and Fig.5.4(e) conveys similar meanings, one is **hateful** and the other is **non-hate** when considering the objects in the images. Meanwhile, AM3 does not simply learn **Adolf (Hitler)** as **hateful** as shown in Fig.5.4(a). The image in Fig.5.4(h) and the Fig.4(a) in the main paper are similar and their text are both non-offensive, with the understanding of multimodalities in an asymmetric manner, AM3 successfully predict them as **non-hate** and **hateful**. The text in Fig.5.4(c), (d) and (e) seem to be **hateful**, with accurate detection and understanding of the objects in the image, the examples turn to be **non-hate**.

### 5.5.2 Failure examples

We present some *offensive* examples that AM3 failed to detect in Fig.5.7. In (a), (b), and (c), the false prediction is from missing the extended attributes of objects in the image, such as 'adoption,' 'poor,' and '9.11 attack'. In (d), detecting the gestures 'paper' and 'rock' is necessary for correct prediction. However, AM3 does not have detection of gestures, leading to a false prediction. In (e) and (f), understanding the relationship of the objects in the image is crucial to classify them as **hateful**. Improving the method to address these three problems may further enhance the task.

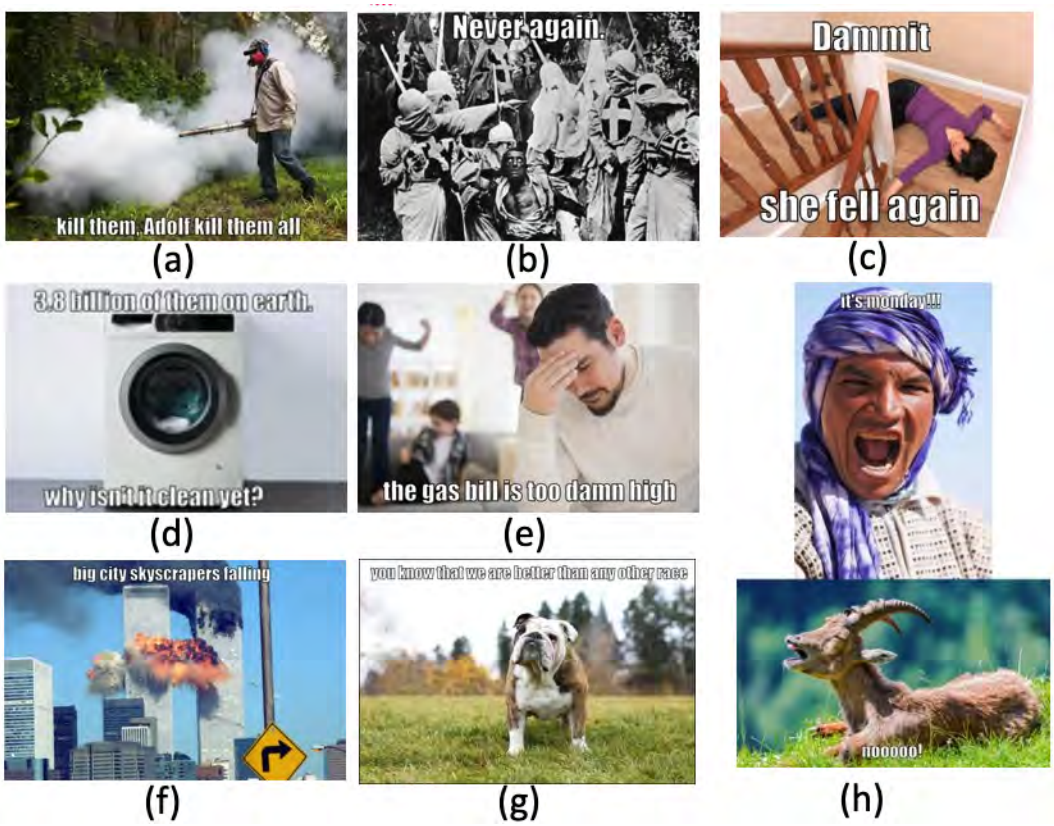



Figure 5.4: [CONTENT WARNING] non-hate examples on multi-modalities that AM3 correctly detected.

## 5.6 Conclusion

In this paper, we present a novel mixed-modal CM model, Asymmetric Mixed-Modal Moderation (AM3), for both multimodal and unimodal content moderation. We propose an asymmetric fusion architecture to fuse multimodal knowledge. Furthermore, we design a novel *cross-modality* contrastive loss to learn the distinct knowledge that can only be conveyed when combining both modalities, which is critical for multimodal CM tasks. Besides using multimodal VL datasets, we also include unimodal CM datasets in pretraining, which not only relaxes data constraints but also improves downstream task performance. With extensive experiments, we show AM3 achieves the new state-of-the art on various multimodal and unimodal CM benchmarks.

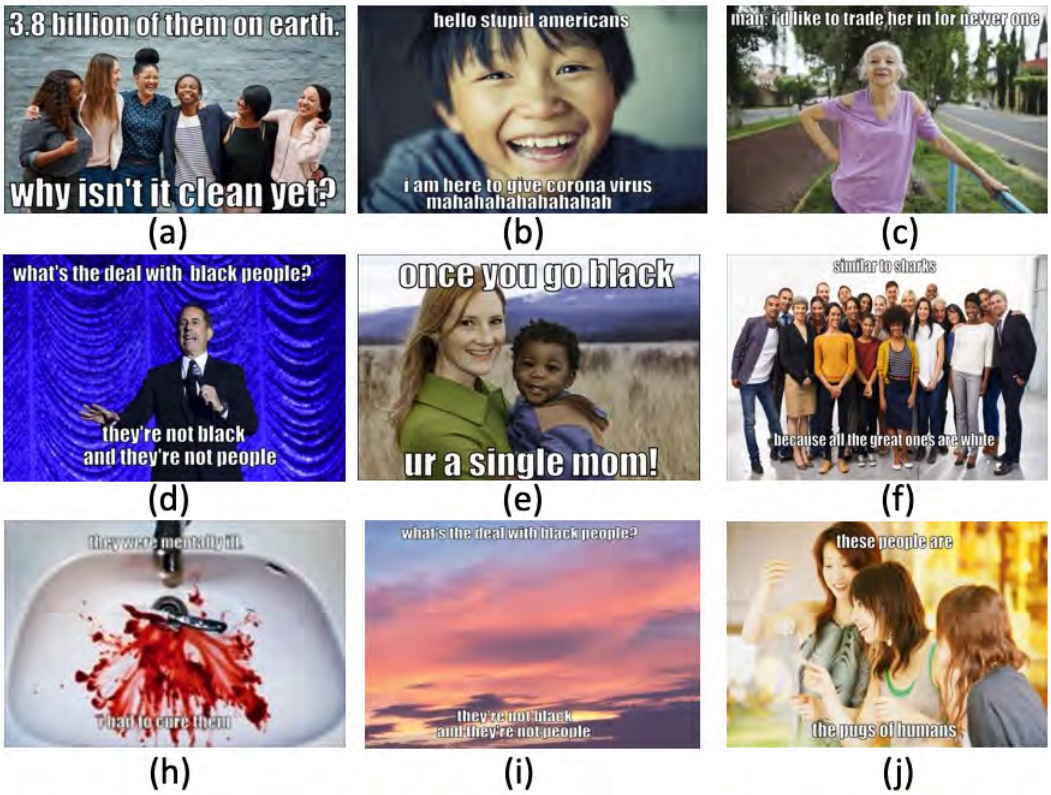



Figure 5.5: [CONTENT WARNING] Hateful examples on uni-modality that AM3 correctly detected.


Figure 5.6: [CONTENT WARNING] Hateful examples on multi-modalities that AM3 correctly detected.


Figure 5.7: [CONTENT WARNING] Hateful examples on multi-modalities that AM3 failed to detect.

# Chapter 6: Conclusion and Future Work

This chapter summarizes the proposed work and describes the methods to improve the algorithms and future research directions.

## 6.1 Conclusion

In this dissertation, we have thoroughly explored the realm of object discovery across various modalities. First, we introduced DVIS, a novel approach that offers a variational relaxation of instance segmentation. DVIS enables end-to-end training of an FCN for the direct prediction of continuous object labels. Our experiments on both PASCAL VOC and MS-COCO datasets showcased its robust performance and its ability to generate high-quality instance label masks for static images.

Next, we confronted the limitation imposed by image quality on static image segmentation tasks and sought to overcome this challenge. This led to the development of MCMPG, a method that leverages temporal information to enhance segmentation quality. Specifically, we employed object proposals initially generated from nearby frames and then propagated to the key frame to create an MP-Graph. Within this graph, we identified maximal cliques, resulting in improved object segments on frames with poor quality. MCMPG proves particularly effective in video segmentation tasks, including UVOS and VIS, especially when combined with a VOS algorithm to propagate detected objects throughout the entire video sequence. Our comprehensive experiments, conducted on the DAVIS-UVOS and YouTube-VIS datasets, yielded compelling evidence of MCMPG's effectiveness, consistently delivering improvements across various settings.

In our continued pursuit of object discovery, we took a step forward by extending it to Vision-Language tasks, thereby introducing a new set of challenges. These challenges revolved around recognizing relationships among objects identified in images and effectively utilizing them with contextual significance in textual information. To address this, we introduced AM3, an innovative asymmetric approach designed to capture unique

knowledge that exists only in cross-modality settings. Through extensive experiments on various challenges, AM3 surpassed prior works by a substantial margin, solidifying its position as a pioneering solution in the field.

## 6.2 Future work

Object discovery is an expansive research field within computer vision, and the work discussed here represents only a glimpse into its possibilities. We are committed to further enhancing the performance of the models introduced for the tasks discussed, as well as extending these models to address new and unexplored challenges.

For DVIS, we proposed a relaxation of instance segmentation, enabling the training of an FCN to directly predict instance labels. However, the model does have some limitations. Firstly, its performance depends on the average number of objects present in the training set images. Consequently, it may struggle to efficiently separate objects when the number of objects in an image exceeds its capacity. One potential solution to this problem is to shift from predicting continuous label values in one dimension to training the model to predict one-hot labels. These one-hot labels activate different channels, each corresponding to a distinct object, and piecewise constant optimization can then be applied to these one-hot predictions to avoid multiple segments on one object. Another challenge arises from the FCN architecture itself. It is difficult for pixels representing small objects to maintain their distinctive features after passing through numerous convolution and pooling layers. This issue leads to inaccurate segmentation of small objects within an image. To address this problem, we can explore technologies and techniques that have been developed to enhance the segmentation of small or thin objects, as demonstrated in recent works like [Ke et al., 2023].

For MCMPG, we leveraged object proposals generated in nearby frames to enhance the segmentation of the key frame. However, it's important to note that this technique may not always improve segmentation, especially when all the nearby frames are in of inferior quality. An alternative approach to harnessing temporal information is by using predicted optical flow on the key frame. This can be advantageous, particularly when historical frames with good quality are available. The predicted optical flow can help correctly segment different objects in various scenarios. The challenges are how to

efficiently maintain the optical flow with high-quality frames while avoiding noise from Low-quality frames. Robust techniques are required to effectively predict optical flow information.

Finally, for AM3, we introduced an innovative asymmetric fusion architecture to fuse multimodal knowledge and designed a novel cross-modality contrastive loss to learn to capture distinct knowledge that only exists in cross-modality. In the work, we demonstrated its effectiveness in various VL CM tasks as well as unimodal image/text CM tasks. However, multimodal content can encompass more modalities, such as video and audio. We believe that the AM3 model presented here can be adapted for CM tasks in these different modalities as well. Besides, we observed that emphasizing the asymmetric fusion of various modalities has a detrimental impact on the model's performance in general VL tasks, such as vision question answering and visual captioning. This presents an interesting avenue for further research to explore ways to overcome this challenge and improve model performance in these tasks.

# Publication

**Jialin Yuan**, Jay Patravali, Hung Nguyen, Chanho Kim, Li Fuxin. "Maximal Cliques on Multi-Frame Proposal Graph for Unsupervised Video Object Segmentation". In submission. arXiv preprint arXiv:2301.12352.

**Jialin Yuan**, Ye Yu, Gaurav Mittal, Matthew Hall, Sandra Sajeev, Mei Chen. "Rethinking Multimodal Content Moderation from an Asymmetric Angle with Mixed-modality". In submission. arXiv preprint arXiv:2305.10547.

**Jialin Yuan**, Chao Chen, Li Fuxin, "Deep Variational Instance Segmentation". Advances in neural information processing systems 33 (2020): 4811-4822.

**Jialin Yuan**, Damanpreet Kaur, Zheng.Zhou, Michael Nagle, et al. "Robust high-throughput phenotyping with deep segmentation enabled by a web-based annotator". Plant Phenomics 2022

Huang Yixuan, **Jialin Yuan**, Chanho Kim, Pupul Pradhan, Bryan Chen, Li Fuxin, and Tucker Hermans. "Out of Sight, Still in Mind: Reasoning and Planning about Unobserved Objects with Video Tracking Enabled Memory Models." arXiv preprint arXiv:2309.15278 (2023).

Nagle Michael F., **Jialin Yuan**, Damanpreet Kaur, Cathleen Ma, Ekaterina Peremyslova, Yuan Jiang, Bahiya Zahl et al. "GWAS identifies candidate genes controlling adventitious rooting in Populus trichocarpa." Horticulture Research 10, no. 8 (2023): uhad125.

Nagle Michael F., **Jialin Yuan**, Damanpreet Kaur, Cathleen Ma, Ekaterina Peremyslova, Yuan Jiang, Alexa Niño de Rivera et al. "GWAS identifies candidate regulators of in planta regeneration in Populus trichocarpa." bioRxiv (2022): 2022-06.

Ye Yu, **Jialin Yuan**, Gaurav Mittal, Li Fuxin, Mei Chen. "BATMAN : Bilateral Attention Transformer in Motion-Appearance Neighboring Space for Video Object Segmentation". ECCV 2022(Oral)

Shaban Amirreza, Alrik Firl, Ahmad Humayun, **Jialin Yuan**, Xinyao Wang, et al.,

"Multiple-Instance Video Segmentation with Sequence-Specific Object Proposals". CVPR Workshop. Vol. 1. 2017.

# Bibliography

https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification.

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022.

Anurag Arnab and Philip HS Torr. Pixelwise instance segmentation with a dynamically instantiated network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 441–450, 2017.

Ali Athar, Sabarinath Mahadevan, Aljosa Osep, Laura Leal-Taixé, and Bastian Leibe. Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *European Conference on Computer Vision*, pages 158–177. Springer, 2020.

Samaneh Azadi, Jiashi Feng, and Trevor Darrell. Learning detection with diverse proposals. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 7369–7377. IEEE, 2017.

Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2858–2866. IEEE, 2017.

Ramy Baly, Mitra Mohtarami, James Glass, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. Integrating stance detection and fact checking in a unified corpus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 21–27, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2004. URL https://aclanthology.org/N18-2004.

Michele Banko, Brendon MacKeen, and Laurie Ray. A unified taxonomy of harmful content. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages

125–137, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.alw-1.16. URL https://aclanthology.org/2020.alw-1.16.

Jonathan T Barron. A general and adaptive robust loss function. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4331–4339, 2019.

Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9739–9748, 2020.

Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: real-time instance segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9157–9166, 2019a.

Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact++: Better real-time instance segmentation. *arXiv preprint arXiv:1912.06218*, 2019b.

Coen Bron and Joep Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 1973.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. *Computer Vision–ECCV 2010*, pages 282–295, 2010.

S. Caelles, K.K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. One-shot video object segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

Sergi Caelles, Jordi Pont-Tuset, Federico Perazzi, Alberto Montes, Kevis-Kokitsi Maninis, and Luc Van Gool. The 2019 davis challenge on vos: Unsupervised multi-object segmentation. *arXiv:1905.00737*, 2019.

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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 1–18. Springer, 2020.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov,

and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. Hatebert: Retraining bert for abusive language detection in english. *arXiv preprint arXiv:2010.12472*, 2020.

Tony F Chan, Selim Esedoglu, and Mila Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM journal on applied mathematics*, 66(5):1632–1648, 2006.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021.

Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

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. *arXiv preprint arXiv:1606.00915*, 2016.

Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2018a.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020a.

Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1189–1198, 2018b.

Zixuan Chen, Huajun Zhou, Jianhuang Lai, Lingxiao Yang, and Xiaohua Xie. Contour-aware loss: Boundary-aware learning for salient object segmentation. *IEEE Transactions on Image Processing*, 30:431–443, 2020b.

Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not

all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021a.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022.

Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. *Advances in Neural Information Processing Systems*, 34, 2021b.

Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *Proceedings of the IEEE international conference on computer vision*, pages 686–695, 2017.

Gabriela Csurka, Diane Larlus, Florent Perronnin, and France Meylan. What is a good evaluation measure for semantic segmentation?. In *BMVC*, volume 27, page 2013, 2013.

Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3150–3158, 2016a.

Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016b.

Robert Dale. Gpt-3: What's it good for? *Natural Language Engineering*, 27(1):113–118, 2021.

Abhishek Das, Japsimar Singh Wahi, and Siyao Li. Detecting hate speech in multi-modal memes. *arXiv preprint arXiv:2012.14891*, 2020.

Achal Dave, Pavel Tokmakov, and Deva Ramanan. Towards segmenting anything that moves. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515, 2017.

Bert De Brabandere, Davy Neven, and Luc Van Gool. Semantic instance segmentation with a discriminative loss function. *arXiv preprint arXiv:1708.02551*, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Mingyu Ding, Bin Xiao, Noel Codella, Ping Luo, Jingdong Wang, and Lu Yuan. Davit: Dual attention vision transformers. *arXiv preprint arXiv:2204.03645*, 2022.

Bin Dong, Fangao Zeng, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Solq: Segmenting objects by learning queries. *Advances in Neural Information Processing Systems*, 34:21898–21909, 2021.

Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. Latent hatred: A benchmark for understanding implicit hate speech. *arXiv preprint arXiv:2109.05322*, 2021.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

Alireza Fathi, Zbigniew Wojna, Vivek Rathod, Peng Wang, Hyun Oh Song, Sergio Guadarrama, and Kevin P Murphy. Semantic instance segmentation via deep metric learning. *arXiv preprint arXiv:1703.10277*, 2017.

Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59:167–181, 2004.

Katerina Fragkiadaki, Geng Zhang, and Jianbo Shi. Video segmentation by tracing discontinuities in a trajectory embedding. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1846–1853. IEEE, 2012.

Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. *Advances in Neural Information Processing Systems*, 33:6616–6628, 2020.

Shubhika Garg and Vidit Goel. Mask selection and propagation for unsupervised video object segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1680–1690, 2021.

Andrej Gisbrecht, Alexander Schulz, and Barbara Hammer. Parametric nonlinear dimensionality reduction using kernel t-sne. *Neurocomputing*, 147:71–82, 2015.

Raul Gomez, Jaume Gibert, Lluis Gomez, and Dimosthenis Karatzas. Exploring hate speech detection in multimodal publications. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1470–1478, 2020.

Leo Grady and Christopher Alvino. Reformulating and optimizing the mumford-shah functional on a graph—a faster, lower energy solution. In *European Conference on Computer Vision*, pages 248–261. Springer, 2008.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2896–2905, 2022.

Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. IEEE, 2011.

Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. *arXiv preprint arXiv:2203.09509*, 2022.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. *arXiv preprint arXiv:1703.06870*, 2017.

Yixuan Huang, Jialin Yuan, Chanho Kim, Pupul Pradhan, Bryan Chen, Li Fuxin, and Tucker Hermans. Out of sight, still in mind: Reasoning and planning about unobserved objects with video tracking enabled memory models. *arXiv preprint arXiv:2309.15278*, 2023.

Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6409–6418, 2019.

Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. Pixel-bert: Aligning image pixels with text by deep multi-modal transformers. *arXiv preprint arXiv:2004.00849*, 2020.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual

reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. *Advances in Neural Information Processing Systems*, 34:13352–13363, 2021.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015.

Anil K Jain and Stan Z Li. *Handbook of face recognition*, volume 1. Springer, 2011.

Scott P Johnson. How infants learn about the visual world. *Cognitive science*, 34(7): 1158–1184, 2010.

Seong Joon Oh, Rodrigo Benenson, Anna Khoreva, Zeynep Akata, Mario Fritz, and Bernt Schiele. Exploiting saliency for object segmentation from image level labels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4410–4419, 2017.

Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. *arXiv preprint arXiv:2306.01567*, 2023.

Dhruv Khattar, Jaipal Singh Goud, Manish Gupta, and Vasudeva Varma. Mvae: Multimodal variational autoencoder for fake news detection. In *The world wide web conference*, pages 2915–2921, 2019.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *Advances in Neural Information Processing Systems*, 33:2611–2624, 2020.

Jiyun Kim, Byounghan Lee, and Kyung-Ah Sohn. Why is it hate speech? masked rationale prediction for explainable hate speech detection. *arXiv preprint arXiv:2211.00243*, 2022.

Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021.

Armin Kirchknopf, Djordje Slijepčević, and Matthias Zeppelzauer. Multimodal detection of information disorder from social media. In *2021 International Conference on Content-Based Multimedia Indexing (CBMI)*, pages 1–4. IEEE, 2021.

Alexander Kirillov, Evgeny Levinkov, Bjoern Andres, Bogdan Savchynskyy, and Carsten Rother. Instancecut: from edges to instances with multicut. *arXiv preprint arXiv:1611.08272*, 2016.

Yeong Jun Koh, Young-Yoon Lee, and Chang-Su Kim. Sequential clique optimization for video object segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 517–533, 2018.

Shu Kong and Charless C Fowlkes. Recurrent pixel embedding for instance grouping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9018–9028, 2018.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, 2020.

Donghoon Lee, Geonho Cha, Ming-Hsuan Yang, and Songhwai Oh. Individualness and determinantal point processes for pedestrian detection. In *European Conference on Computer Vision*, pages 330–346. Springer, 2016.

Roy Ka-Wei Lee, Rui Cao, Ziqing Fan, Jing Jiang, and Wen-Haw Chong. Disentangling hate in online memes. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 5138–5147, 2021.

Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M Rehg. Video segmentation by tracking many figure-ground segments. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2192–2199, 2013.

Peiguang Li, Xian Sun, Hongfeng Yu, Yu Tian, Fanglong Yao, and Guangluan Xu. Entity-oriented multi-modal alignment and fusion network for fake news detection. *IEEE Transactions on Multimedia*, 2021a.

Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. UNIMO: Towards unified-modal understanding and generation via cross-modal contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 2592–2607, Online, August 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.202. URL `https://aclanthology.org/2021.acl-long.202`.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-

training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020a.

Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. *arXiv preprint arXiv:1611.07709*, 2016.

Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 280–287, 2014.

Yu Li, Zhuoran Shen, and Ying Shan. Fast video object segmentation using the global context module. In *European Conference on Computer Vision*, pages 735–750. Springer, 2020b.

Huaijia Lin, Ruizheng Wu, Shu Liu, Jiangbo Lu, and Jiaya Jia. Video instance segmentation with a propose-reduce paradigm. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1739–1748, October 2021.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

Shu Liu, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. Sgn: Sequential grouping networks for instance segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017.

Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

Google LLC. Cloud vision api: Web entity detection. 2021. Accessed: September 20, 2021.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

Liana M Lorigo and Venugopal Govindaraju. Offline arabic handwriting recognition: a survey. *IEEE transactions on pattern analysis and machine intelligence*, 28(5):712–724, 2006.

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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Jonathon Luiten, Idil Esen Zulfikar, and Bastian Leibe. Unovost: Unsupervised offline video object segmentation and tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2000–2009, 2020.

Mengmeng Ma, Jian Ren, Long Zhao, Davide Testuggine, and Xi Peng. Are multimodal transformers robust to missing modality? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18177–18186, 2022.

Tianyang Ma and Longin Jan Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 670–677. IEEE, 2012.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14867–14875, 2021.

Ali Modarressi, Hosein Mohebbi, and Mohammad Taher Pilehvar. Adapler: Speeding up inference by adaptive length reduction. *arXiv preprint arXiv:2203.08991*, 2022.

Anca Morar, Florica Moldoveanu, and Eduard Gröller. Image segmentation based on active contours without edges. In *2012 IEEE 8th International Conference on Intelligent Computer Communication and Processing*, pages 213–220. IEEE, 2012.

David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 42(5):577–685, 1989.

Kai Nakamura, Sharon Levy, and William Yang Wang. r/fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection. *arXiv preprint arXiv:1911.03854*, 2019.

Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. *arXiv preprint arXiv:1611.05424*, 2016.

Hung Nguyen and Fuxin Li. Space time recurrent memory network. *arXiv preprint arXiv:2109.06474*, 2021.

Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *International Conference on Computer Vision (ICCV)*, 2015.

Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016.

Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2663–2672, 2017.

Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, et al. Combined scaling for open-vocabulary image classification. *arXiv preprint arXiv: 2111.10050*, 2021.

Dinh Duy Phan, Thanh Thien Nguyen, Quang Huy Nguyen, Hoang Loc Tran, Khac Ngoc Khoi Nguyen, and Duc Lung Vu. Lspd: A large-scale pornographic dataset for detection and classification. *International Journal of Intelligent Engineering and Systems*, 15(1), 2022.

Thomas Pock, Daniel Cremers, Horst Bischof, and Antonin Chambolle. An algorithm for minimizing the mumford-shah functional. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1133–1140. IEEE, 2009.

Jordi Pont-Tuset, Pablo Arbelaez, Jonathan T Barron, Ferran Marques, and Jitendra

Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. *IEEE transactions on pattern analysis and machine intelligence*, 39(1): 128–140, 2017a.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017b.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017c.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

Mengye Ren and Richard S Zemel. End-to-end instance segmentation and counting with recurrent attention. *arXiv preprint arXiv:1605.09410*, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*, 2020.

Bernardino Romera-Paredes and Philip Hilaire Sean Torr. Recurrent instance segmentation. In *European Conference on Computer Vision*, pages 312–329. Springer, 2016.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252, 2015.

Nihar Sahoo, Himanshu Gupta, and Pushpak Bhattacharyya. Detecting unintended social bias in toxic language datasets. *arXiv preprint arXiv:2210.11762*, 2022.

Anna Schmidt and Michael Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain, April 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-1101. URL `https://aclanthology.org/W17-1101`.

Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *European Conference on Computer Vision*, pages 629–645. Springer, 2020.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018.

Felix Soldner, Verónica Pérez-Rosas, and Rada Mihalcea. Box of lies: Multimodal deception detection in dialogues. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1768–1777, 2019.

Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 715–731, 2018.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012.

Evgeny Strekalovskiy and Daniel Cremers. Real-time minimization of the piecewise smooth mumford-shah functional. In *European conference on computer vision*, pages 127–141. Springer, 2014.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019.

Arvind Subramaniam, Aryan Mehra, and Sayani Kundu. Exploring hate speech detection with hatexplain and bert. *arXiv preprint arXiv:2208.04489*, 2022.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.

Omkar Thawakar, Sanath Narayan, Jiale Cao, Hisham Cholakkal, Rao Muhammad Anwer, Muhammad Haris Khan, Salman Khan, Michael Felsberg, and Fahad Shahbaz Khan. Video instance segmentation via multi-scale spatio-temporal split attention transformer. *Proc. European Conference on Computer Vision (ECCV)*, 2022a.

Omkar Thawakar, Sanath Narayan, Jiale Cao, Hisham Cholakkal, Rao Muhammad Anwer, Muhammad Haris Khan, Salman Khan, Michael Felsberg, and Fahad Shahbaz Khan. Video instance segmentation via multi-scale spatio-temporal split attention transformer. *arXiv preprint arXiv:2203.13253*, 2022b.

Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning motion patterns in videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3386–3394, 2017.

Jonas Uhrig, Marius Cordts, Uwe Franke, and Thomas Brox. Pixel-level encoding and depth layering for instance-level semantic labeling. In *German Conference on Pattern Recognition*, pages 14–25. Springer, 2016.

Jonas Uhrig, Eike Rehder, Björn Fröhlich, Uwe Franke, and Thomas Brox. Box2pix: Single-shot instance segmentation by assigning pixels to object boxes. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 292–299. IEEE, 2018.

Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104 (2):154–171, 2013.

Eva AM van Dis, Johan Bollen, Willem Zuidema, Robert van Rooij, and Claudi L Bockting. Chatgpt: five priorities for research. *Nature*, 614(7947):224–226, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Carles Ventura, Miriam Bellver, Andreu Girbau, Amaia Salvador, Ferran Marques, and Xavier Giro-i Nieto. Rvos: End-to-end recurrent network for video object segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Luminita A Vese and Tony F Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International journal of computer vision*, 50(3):271–293, 2002.

Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9481–9490, 2019.

Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *science*, 359(6380):1146–1151, 2018.

Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5463–5474, 2021a.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022.

Tao Wang, Ning Xu, Kean Chen, and Weiyao Lin. End-to-end video instance segmentation via spatial-temporal graph neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10797–10806, 2021b.

Wenguan Wang, Xiankai Lu, Jianbing Shen, David J. Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019a.

Wenguan Wang, Jianbing Shen, Xiankai Lu, Steven CH Hoi, and Haibin Ling. Paying attention to video object pattern understanding. *IEEE transactions on pattern analysis and machine intelligence*, 2020a.

Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. *arXiv preprint arXiv:1912.04488*, 2019b.

Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *European Conference on Computer Vision*, pages 649–665. Springer, 2020b.

Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic, faster and stronger. *arXiv preprint arXiv:2003.10152*, 2020c.

Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8741–8750, 2021c.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021d.

Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-2013. URL `https://aclanthology.org/N16-2013`.

Zeerak Waseem, Thomas Davidson, Dana Warmsley, and Ingmar Weber. Understanding abuse: A typology of abusive language detection subtasks. In *Proceedings of the First Workshop on Abusive Language Online*, pages 78–84, Vancouver, BC, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-3012. URL `https://aclanthology.org/W17-3012`.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 959–968, 2022a.

Junfeng Wu, Yi Jiang, Song Bai, Wenqing Zhang, and Xiang Bai. Seqformer: Sequential transformer for video instance segmentation. In *European Conference on Computer Vision*, pages 553–569. Springer, 2022b.

Ruizheng Wu, Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Memory selection network for video propagation. In *European Conference on Computer Vision*, pages 175–190. Springer, 2020.

Zhengyang Wu, Fuxin Li, Rahul Sukthankar, and James M Rehg. Robust video segment proposals with painless occlusion handling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4194–4203, 2015.

Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

Christopher Xie, Yu Xiang, Zaid Harchaoui, and Dieter Fox. Object discovery in videos

as foreground motion clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9994–10003, 2019.

Enze Xie, Peize Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12193–12202, 2020.

Li Xu, Cewu Lu, Yi Xu, and Jiaya Jia. Image smoothing via l 0 gradient minimization. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, pages 1–12, 2011.

Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, and Jianfeng Gao. Unified contrastive learning in image-text-label space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19163–19173, 2022.

Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K Katsaggelos. Efficient video object segmentation via network modulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6499–6507, 2018.

Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5188–5197, 2019.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3208–3216, 2021.

Jialin Yuan, Chao Chen, and Li Fuxin. Deep variational instance segmentation. *arXiv preprint arXiv:2007.11576*, 2020.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Sara Zaheri, Jeff Leath, and David Stroud. Toxic comment classification. *SMU Data Science Review*, 3(1):13, 2020.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2021.

Ziyu Zhang, Alexander G Schwing, Sanja Fidler, and Raquel Urtasun. Monocular object instance segmentation and depth ordering with cnns. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2614–2622, 2015.

Ziyu Zhang, Sanja Fidler, and Raquel Urtasun. Instance-level segmentation for autonomous driving with deep densely connected mrfs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 669–677, 2016.

Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015.

Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 13066–13073, 2020.

Tianfei Zhou, Jianwu Li, Xueyi Li, and Ling Shao. Target-aware object discovery and association for unsupervised video multi-object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6985–6994, June 2021a.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Yejin Choi, and Noah A Smith. The ineffectiveness of algorithmic debiasing for toxic language detection. In *EACL*, 2021b.

Ron Zhu. Enhance multimodal transformer with external label and in-domain pretrain: Hateful meme challenge winning solution. *arXiv preprint arXiv:2012.08290*, 2020.