Salient Object Detection and Segmentation in Video Surveillance

Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy by

Siyue Yu

August 2022

Abstract

Video surveillance outputs different portrait information of scenes such as crime investigation, security system, automatic driving system, and environmental monitoring. Recently, deep learning based video surveillance is also an essential topic in computer vision. The specific tasks include object tracking, video object segmentation, salient object detection, and video salient object detection. Thus, this thesis studies salient object detection and segmentation in video surveillance, mainly on video object segmentation and salient object detection.

In video object segmentation, we study the case of given the first frame's mask and try to design a network that can adapt to different object appearance variations. Therefore, this thesis proposes a framework based on the non-local attention mechanism to localize and segment the target object in the current frame, referring to both the first frame with its given mask and the previous frame with its predicted mask. Our approach can achieve 86.5% IoU on DAVIS-2016 and 72.2% IoU on DAVIS-2017, with a speed of 0.11s per frame.

Then for salient object detection, this thesis focuses on scribble annotations. However, scribbles fail to contain enough integral appearance information. To solve this problem. A local saliency coherence loss is proposed to assist partial cross-entropy loss and thereby help the network learn more complete object information. Further, A self-consist mechanism is designed to help the network not sensitive to different input scales. Our method can achieve comparable results compared with fully supervised methods. Our method achieves a new state-of-the-art performance on six benchmarks (e.g. for the ECSSD dataset: $F_\beta = 0.8995$, $E_\xi = 0.9079$ and $MAE = 0.0489$).

Lastly, co-salient object detection is also studied. Recent methods explore both intra- and inter-image consistency through an attention mechanism. We find that existing attention mechanisms can only focus on limited related pixels. Thus, we propose a new framework with a self-contrastive loss to mine more related pixels to obtain comprehensive features. Our method obtains 0.598 for maximum F-measure for COCA.

In this way, the tasks in this thesis are well handled and our methods can serve as new baselines for future works.

Key Words: Video Object Segmentation, Salient Object Detection, Co-salient Object Detection, Pixel Matching, Attention Mechanism, Feature Mining

Acknowledgements

Looking back on the short four years, my research and study cannot succeed without the help and support of supervisors, colleagues, friends, and family. First and foremost, I would like to express my sincere appreciation to my principal supervisors Professor Eng Gee Lim and Professor Jimin Xiao. Without their support, I would not get chance for further study on my research. I cannot achieve today's work without their patience, stimulation, and immense knowledge. Their conversations and meetings inspired me to conduct research and form a comprehensive understanding of my tasks. Their encouragement gave me faith to keep going, and their guidance helped me in the time of my research and the writing of papers, including this thesis.

Then, I would like to thank my IPAP advisors, Professor Qiufeng Wang, and Professor Jeremy S. Smith, for their guidance and feedback throughout the four years. I am also grateful for my examiners, Professor Shuai Li and Professor Yong Yue. Their rigorous comments and patience helped me broaden my mind and understand my research from multiple perspectives.

Additionally, I would like to express my sincere thanks to Professor Shaofeng Lu. Without his suggestions and recommendation, I would not have had the chance to touch research and apply for a doctoral program.

Besides them, I extend my gratitude to all my friends and labmates in both EE408 and EE410: Dr. Yanchun Xie, Dr. Dingyuan Zheng, Dr. Bingfeng Zhang, Dr. Mingjie Sun, Dr. Samer Jammal, Miss Hui Li, Mr. Xinqiao Zhao, Mr. Xiaoyang Wang. Mr. Junkun Peng, and Mr. Jian Wang. I really treasure the days being with them, discussing with them, chatting with them, and dining together. Without their precious support, it would not be possible to conduct this research.

My sincere thanks also go to my family, especially my parents, for their meticulous care and being my shelter when I was in bad moods. Without their love and unconditional support, none would be possible. I also thank my best friend, Miss Siyun Tan, who always encouraged me not to be afraid, spent time listening to my complaints, and was happy for me when I harvested.

Lastly, I would like to thank myself for not giving up. I appreciate all the difficulties and joys during the past four years.

Contents

List of Figures

List of Tables

List of Acronyms

CNN Convolutional Neural Network

CoSOD Co-salient Object Detection

DFE Democratic Feature Enhancement Module

DPG Democratic Prototype Generation Module

DRB Democratic Response Block

IoU Intersection over Union

SCL Self-contrastive Learning Module

SOD Salient Object Detection

SSB Seed Selection Block

VOS Video Object Segmentation

Chapter 1

Introduction

1.1 Research Motivation

Video surveillance is closely linked with people's life. They are usually used for different purposes. One important role of video surveillance is to provide security containing protection against theft, burglaries and other crimes. This kind of video surveillance can provide evidence and help find the criminals. Additionally, video surveillance can be used for traffic monitoring. In this way, the traffic flow can be improved and accidents can be monitored. Further, video surveillance can also be used for private monitoring for baby care or pet care, etc. As shopping online becomes more and more popular, young people tend to apply video surveillance to monitor the delivery to protect against stealing their goods or other crimes. Besides that, automatic drive is gaining attention nowadays. Video surveillance is one of the important technologies to recognize road condition, parking condition or other demands for video understanding. Therefore, video surveillance is an important security application and needs to adapt for different demands.

In recent years, with the development of deep neural network, video surveillance is also a major topic in computer vision tasks. The major task about video surveillance in computer vision is video understanding, including object detection, object tracking, object segmentation, action recognition, video retrieval, etc. As illustrated in Fig. 1.1,among these tasks, video object segmentation (VOS) contributes a lot. It distinguishes each pixel into foreground and background to provide a pixel-level observation of target object. In this case, it can assist video object tracking, action recognition and video retrieval. However, VOS faces many crucial problems from the dynamic target object, such as appearance variation, and occlusion. Moreover, when the background is noisy, the models tend to fail to distinguish foreground and background. In this case, it is difficult for VOS models to capture

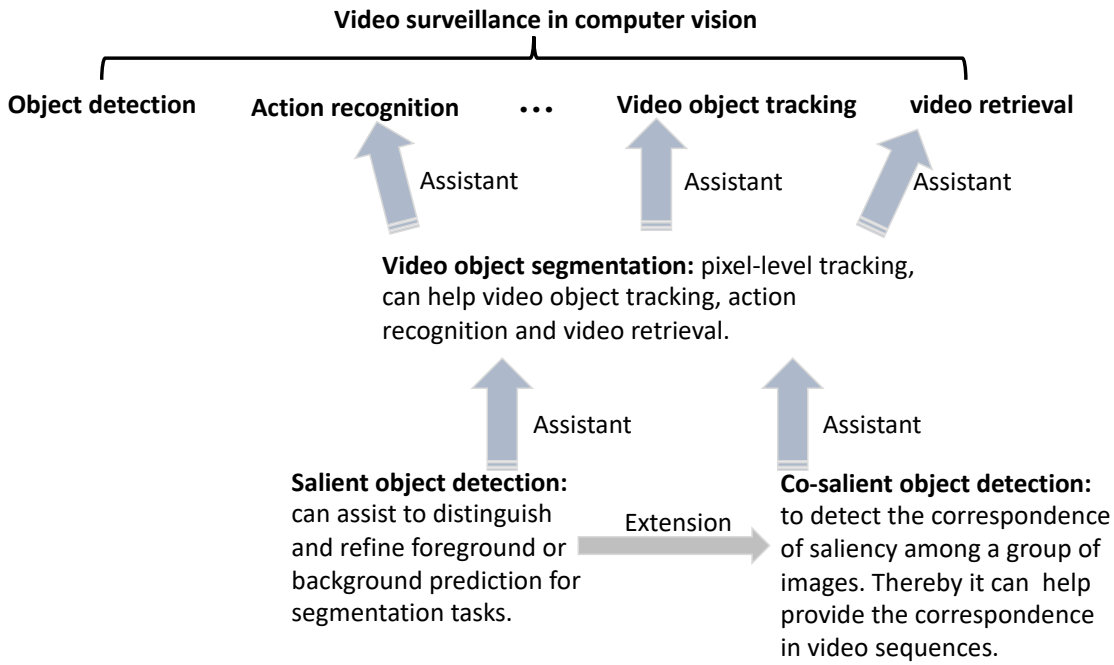


Figure 1.1: Relationships of the tasks in this thesis.

clear and complete target object. Usually, the target objects in VOS are salient objects and they are contrast to their surroundings. Then, one possible solution is to involve salient object detection (SOD) to utilize saliency information as an indicator [112, 5, 77, 113]. Saliency information can help filter background and complement foreground information for cases without good foreground reference masks. Additionally, saliency maps have been used in many image processing tasks, such as semantic segmentation [102, 53], lesion segmentation [3], person search [49] and image retrieval [107]. Additionally, to help understand the correspondence in VOS, co-salient object detection (CoSOD), which extends SOD segment the common objects among a group of relative images, is studied. CoSOD can help provide a global cue which is the appearance consistency to model the target foreground object. For example, some works [11, 27] treat the video frames as a group of images containing relative objects and adopt CoSOD to mine co-existed foreground objects. Nevertheless, CoSOD methods should be further designed to mine more comprehensive features if applied to VOS for better performance, especially when there are noisy frames [131]. Thus, this thesis also conduct experiments on CoSOD to design a method for better appearance consistency.

As shown in Fig. 1.2 (a), video object segmentation (VOS) aims to segment all the pixels belonging to the target object. Specifically, given the target object in the first frame, VOS needs to localize the target object in the following frames and segment all the pixels of the object in each frame. It usually suffers from occlusion, object appearance variation, blur, scale variation and other challenges [126]. Additionally, VOS is computation complexity as it needs to tackle video sequences, especially when the sequence is long. Therefore, this thesis focuses on how to obtain great performance and keep high efficiency at the same time.

Then, salient object detection targets at finding the object that can attract people in a image and is illustrated in Fig. 1.2 (b). Although recent approaches have achieved great performance, they need large annotations, even introducing extra datasets like edge detection dataset for smooth boundary. Moreover, sparse labelling becomes increasingly popular in recent years to save label time. However, it is difficult for sparse labelling to reveal the structure and shape of target objects. Thus, how to design a model can learn to predict smooth and integral objects without help of extra dataset or post-process is important for salient object detection.

Finally, co-salient object detection is an extension of salient object detection. Co-salient object detection needs to detect and segment the common object through a group of image, which can be seen as formulating human attention from multiple perspectives as shown in Fig. 1.2 (c). The most key challenging in CoSOD is how to establish intra-image and inter-image consistency to find the common objects due to the lack of classification information. Although existed deep learning

based methods have obtained outstanding performances, they rely on extra information to learn discriminative co-salient features, such as extra salient object detection training-set or classification information. Therefore, we consider thoroughly exploring the intrinsic characteristics of co-salient objects and background to link both intra-image and inter-image consitency.

1.2 Research Questions

In this thesis, plenty of research questions about conducted tasks are dealt with. Different research questions are discussed in different chapters as following:

In chapter 3, fast pixel matching video object segmentation task is studied. Video object segmentation, aiming to segment the foreground objects given the annotation of the first frame, has been attracting increasing attentions. Many state-of-the-art approaches have achieved great performance by relying on online model updating or mask-propagation techniques. However, most online models require high computational cost due to model fine-tuning during inference. Most mask-propagation based models are faster but with relatively low performance due to failure to adapt to object appearance variation. In this case, the main research question is how to design a framework to obtain higher performance with fast speed.

In chapter 4, structure consistent weakly supervised salient object detection is analyzed. Sparse labels have been gaining much attention in recent years, especially scribble annotations. However, the performance gap between scribble supervised and fully supervised salient object detection methods is huge, because that the scribbles cannot provide appearance and integral boundary information. To solve this problem, most previous works adopt complex training methods like introducing extra dataset for boundary prediction and post-process for better predictions. Therefore, the main research in this task is how to predict accurate saliency maps with only scribble annotations and how to eliminate post-process or multi-stage training for simplification.

In chapter 5, comprehensive feature mining for co-salient object detection is further explored. Co-salient object detection, with the target of detecting co-existed salient objects among a group of images, is gaining popularity. Recent works use the attention mechanism to link the appearance consistency through different images. However, their methods lead to incomplete even incorrect responses for target objects. They also need extra information like extra salient object detection and classification to help learn co-saliency. Thus, the main research question here is how to enlarge the responses to mine more comprehensive co-salient features for better predictions without the help of extra information.

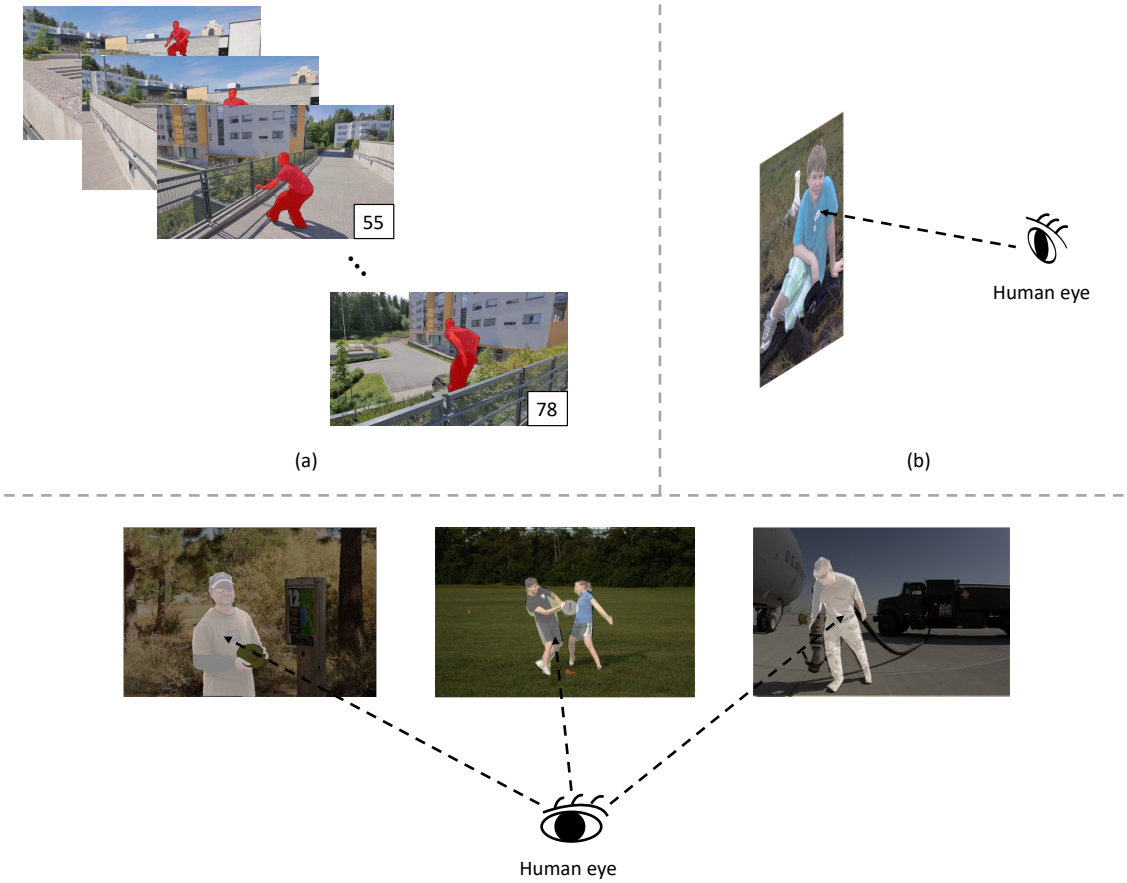


Figure 1.2: The main tasks in this thesis: (a) video object segmentation; (b) salient object detection, (c) co-salient object detection.

1.3 Main Contributions

The main contributions of this thesis are the analysis of video object segmentation, salient object detection and co-salient object detection. All this three tasks are binary segmentation. How to predict target maps without classification information is discussed and new frameworks are designed for each task to make the best of given information and realize the improvement of performances.

Fast Pixel Matching for Video Object Segmentation. We are aiming to design a new model to make a good balance between speed and performance. We propose a novel NPMCA-net, which directly localizes foreground objects based on mask-propagation and non-local attention mechanism to match pixels in reference and current frames. Since we bring in information of both first and previous frames, our network is robust to large object appearance variation, and can better adapt to occlusions. Extensive experiments show that our approach can achieve a new state-of-the-art performance with a fast speed at the same time (86.5% IoU on DAVIS-2016 and 72.2% IoU on DAVIS-2017, with speed of 0.11s per frame) under the same level comparison.

Structure Consistent Weakly Supervised Salient Object Detection. We propose a one-round end-to-end training approach for weakly supervised salient object detection via scribble annotations without pre/post-processing operations or extra supervision data. Since scribble labels fail to offer detailed salient regions, we propose a local coherence loss to propagate the labels to unlabeled regions based on image features and pixel distance, so as to predict integral salient regions with complete object structures. We design a saliency structure consistency loss as self-consistent mechanism to ensure consistent saliency maps are predicted with different scales of the same image as input, which could be viewed as a regularization technique to enhance the model generalization ability. Additionally, we design an aggregation module (AGGM) to better integrate high-level features, low-level features and global context information for the decoder to aggregate various information. Extensive experiments show that our method achieves a new state-of-the-art performance on six benchmarks (e.g. for the ECSSD dataset: $F_\beta = 0.8995$, $E_\xi = 0.9079$ and $MAE = 0.0489$), with an average gain of 4.60% for F-measure, 2.05% for E-measure and 1.88% for MAE over the previous best methods on this task.

Comprehensive Feature Mining for Co-salient Object Detection. We aim to mine comprehensive co-salient features with democracy and reduce background interference without introducing any extra information. To achieve this, we design a democratic prototype generation module to generate democratic response maps, covering sufficient co-salient regions and thereby involving more shared attributes of co-salient objects. Then a comprehensive prototype based

Chapter	Task	Input	Output	Level of Supervision
3	Video object segmentation	Video frames	Object segmentation map	Fully supervised
4	Salient object detection	Single image	Saliency map	Weakly supervised
5	Co-salient object detection	Group of images	Co-saliency map	Fully supervised

Figure 1.3: Tasks studied in this thesis.

on the response maps can be generated as a guide for final prediction. To suppress the noisy background information in the prototype, we propose a self-contrastive learning module, where both positive and negative pairs are formed without relying on additional classification information. Besides, we also design a democratic feature enhancement module to further strengthen the co-salient features by readjusting attention values. Extensive experiments show that our model obtains better performance than previous state-of-the-art methods, especially on challenging real-world cases (, for CoCA, we obtain a gain of 2.0% for MAE, 5.4% for maximum F-measure, 2.3% for maximum E-measure, and 3.7% for S-measure) under the same settings.

1.4 Thesis Outline

The rest of this thesis is arranged as follows. Chapter **??** provides a brief literature review of each task conducted in this thesis. The basic settings of chapter 3 ∼ chapter 5 are listed in Fig. 1.3. Chapter 3 introduces the task of video object segmentation. It deals with video frames under fully supervision and use attention mechanism to match pixels from reference frames and current frame, so as to localize the target object in current frame. Next, Chapter 4 describes the task of salient object detection. It tackles single image and trains the network through scribble annotations in an end-to-end learning style. After that, chapter 5 explores co-salient object detection. This task needs to detect the common objects from multiple images under the fully supervision. It studies how to mine comprehensive features of common object and predict clean and complete co-salient object masks. All the three tasks belong to the binary segmentation. The outputs are binary masks. '1' for foregrounds and '0' for backgrounds. Then, chapter 6 gives a gathering conclusion of above

tasks and recommends some future researches.

Chapter 2

Literature Review

In this chapter, a literature review is introduced about the tasks conducted in this thesis, mainly about video object segmentation, salient object detection and co-salient object detection.

2.1 Video Object Segmentation

Video object segmentation (VOS) is one of the fundamental and challenging tasks in video understanding. It aims to track the target object and classify each pixel of the frame into foreground or background. It can be further applied into other video understanding tasks like video object tracking, action recognition and video retrieval. Thus, it plays important role in autonomous driving , auto-mated surveillance or other video-related scenes.

Video object segmentation only cares about the target object. This task does not concern the specific categories of the target object. According to the number of target objects, VOS can be classified into two sub-tasks. One is single-object video object segmentation as shown in Fig. 2.1 (a). It only tracks and segments one object in a video sequence. Another one is multi-object segmentation like in Fig. 2.1 (b). It needs to track and segment multiple target objects and distinguish each instance, such as the 'Fish' sample in Fig. 2.1 (b). In such case, it can be also defined as instance video object segmentation. Additionally, VOS can be fundamentally divided into two categories based on how many human interference are involved during inference: 1) semi-automatic video object segmentation; 2) automatic video object segmentation.

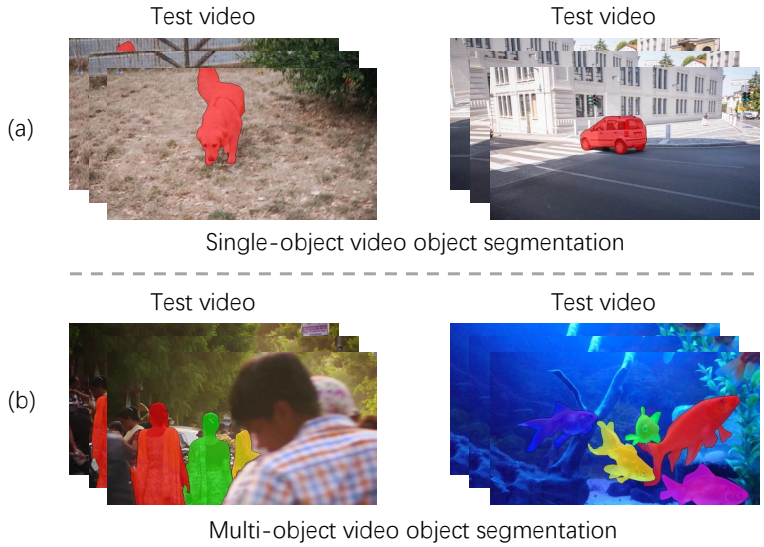


Figure 2.1: Two sub-tasks of video object segmentation: (a) single-object video object segmentation; (b) multi-object segmentation.

2.1.1 Semi-automatic Video Object Segmentation

Semi-automatic video object segmentation, also called semi-supervised video object segmentation or one-shot video object segmentation [115]. In this task, there are hints for target object. The location of the target object is given and the designed model needs to track and segment the given target for the remainder frames. There are typically four kinds of hints as shown in Fig. 2.2. The most common one is the mask of the first frame. all the pixels belonging to the target object are labelled like in Fig. 2.2 (a). In such case, it is also known as pixel-wise tracking or mask propagation [115]. In this setting, the methods usually use the first frame or previous frames' predictions as reference to guide the target segmentation on current frame. Additionally, bounding box is a kind of fast annotation to indicate the target object. There is a rectangular to label the target object like in Fig. 2.2 (b). Although bounding box saves labelling cost, it is difficulty to provide detail information of target object, such as appearance, shape or boundary. With the increase attraction on multi-modal, language hints also introduced into video object segmentation. In such case, there is at least one sentence to describe the target object as shown in Fig. 2.2 (c). Then, the cross models between image feature and language feature are fused to generate the target feature and predict the corresponding masks. The involved language information can help localize and detect the whole regions of target object. Besides, it can help save pixel-level labelling cost.

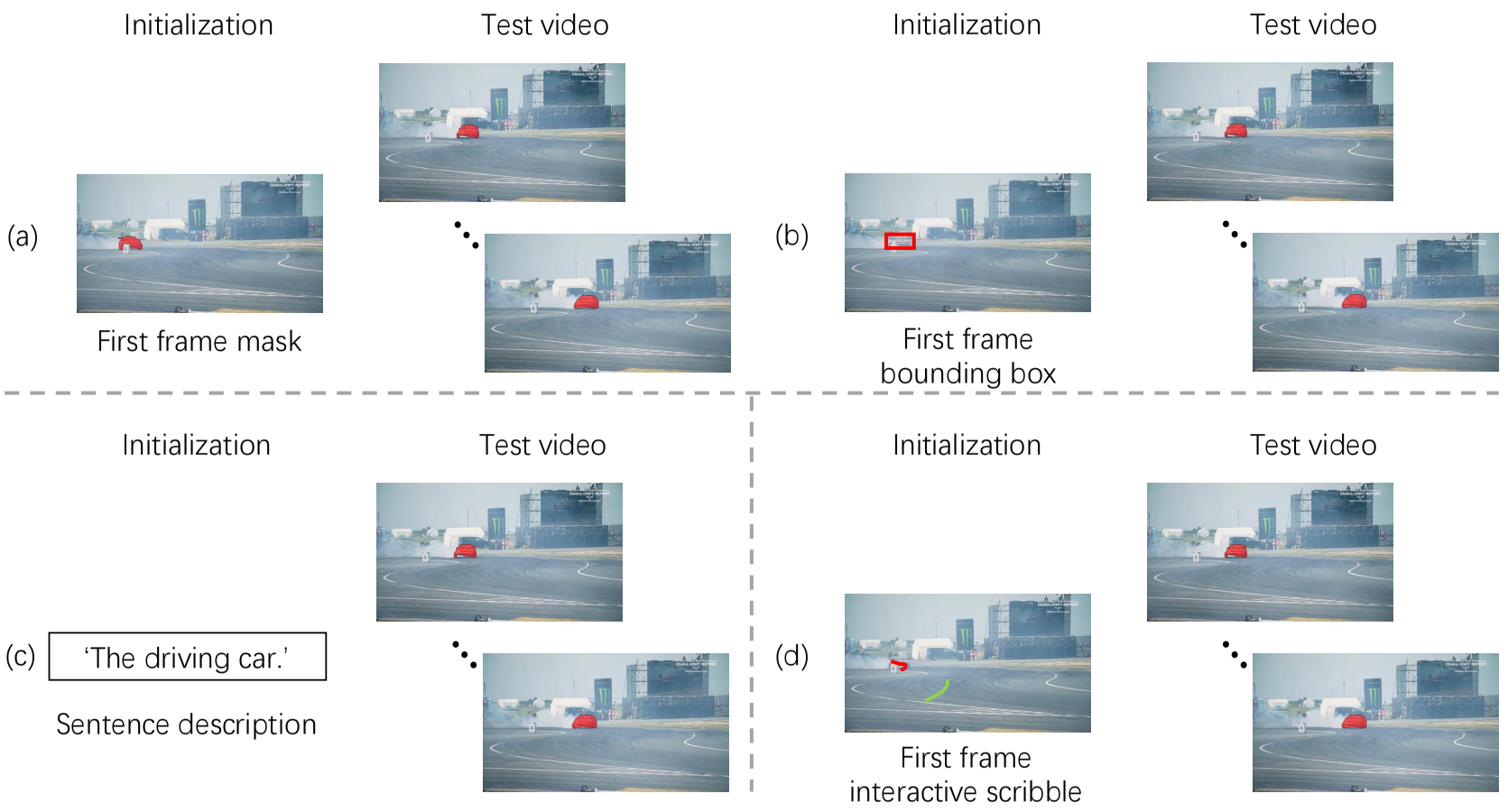


Figure 2.2: Different settings of semi-automatic video object segmentation: (a) given the mask of the first frame to indicator the target object; (b) given the bounding box to localize the target object; (c) given a sentence to describe the target. (d) given interactive scribble to indicate the target object and background.

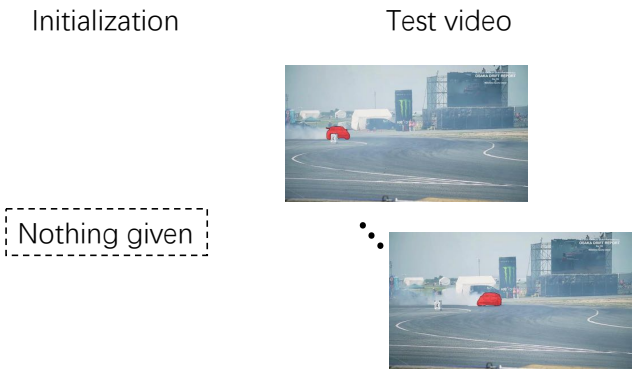


Figure 2.3: Automatic video object segmentation. There is no target object information for initialization. The model needs to detect the object first.

However, it is also hard for caption to reflect the detail appearance of the object target. Methods in this sub-task also needs to learn how to acquire the smooth boundary and complete appearance prediction. Further, in industrial, cooperating with user guidance throughout the video sequence is gaining interest, which is known as interactive video object segmentation. In Fig. 2.2 (d), we shown the example of this setting. Users can draw different scribbles on the target object and background in a certain frame to guide the network to track and segment the labelled target object. Although interactive video object segmentation can allow users to specify the segmentation constraints, it requires the algorithms to response to these constraints quickly for good use experience.

2.1.2 Automatic Video Object Segmentation

Automatic video object segmentation is also called unsupervised video object segmentation or zero-shot video object segmentation [115]. In this kind of setting, there is no any initialization as shown in Fig. 2.3. The algorithm needs to detect the target object according the consistency across frames. In this setting, VOS approaches need to search the target object through cross-frame consistency first, then segment the target object in all the frames. Moreover, there is nothing can help provide appearance cues. It is difficult to get smooth and complete object masks. Some methods [54] propose to choose the best reference frame instead of directly use the first frame as the reference frame since not all the first frames can provide the complete and transparent appearance of a target object. Additionally, AGS [114] verifies the consistency of visual attention behavior among human observers and they introduce dynamic fixation data to train a initial video attention module to detect the target object first.

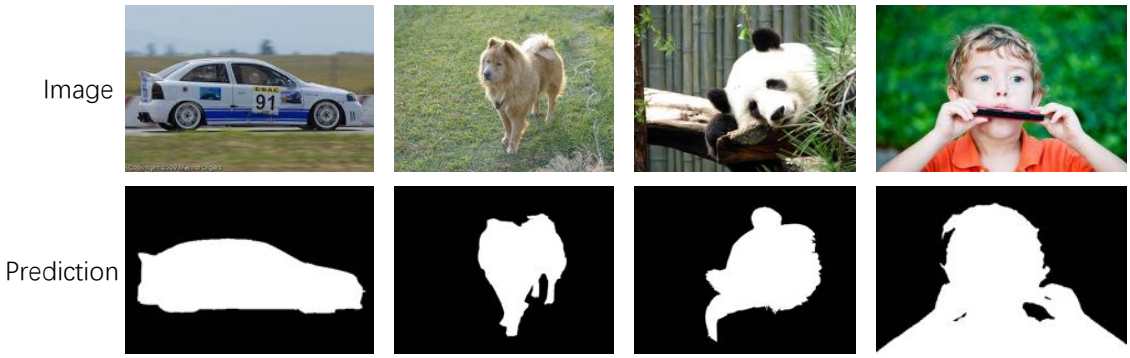


Figure 2.4: The input images and outputs of salient object detection.

To sum up, main challenges of video object segmentation is how to obtain excellent segmentation masks for the dynamic objects, containing appearance variation, occlusion, scale variation, re-appearing after disappearing for some frames, blur, and long-range position change in succession frames. Further, VOS is computing complexity. Therefore, strong and efficient models are desired to be proposed. In this thesis, we dedicate to the semi-automatic video object detection with the given mask of the first frame. We find that previous methods use online fine-tuning to help their methods learn appearance variations. However, online fine-tuning slows the inference. Mask-propagation based methods are more efficient, but their performances are lower. Therefore, we try to design a framework with both good performance and high efficiency.

2.2 Salient Object Detection

Salient object detection can be a upstream task for video understanding as it targets at modelling how human attention system works. It can help provide the target hint for video object segmentation and other binary segmentation tasks. This technique can further be applied in the automatic driving, crime tracking or abnormal detection in video surveillance.

Salient object detection(SOD) originates from eye fixation prediction [8] to find out the most salient object in the picture that can deeply attract human eyes [45, 111]. For one image, there always exists at least one pixel that can attract human eyes at the first glance. Understanding and modeling such ability, which is defined as visual attention or visual saliency, are fundamental and popular subjects in psychology, neurobiology, cognitive science, and computer vision [111]. In the first instance, eye fixation prediction (FP) [100] is first proposed to mimic this process and learn human attention mechanism. FP comes from cognitive and psychology communities and it aims

at detecting the fixation positions when people observing a single image [111].

In contrast, SOD task needs to detect the salient object and synchronously segment all the pixels belonging to the salient object. Thus, in SOD, given one image, the task needs to predict the corresponding saliency map as shown in Fig 2.4. The first research of SOD is from the model proposed by Itti, et al [40]. They propose an initial model to implement computational frameworks and psychological theories of bottom-up attention by considering center-surround mechanisms for scene understanding. Note that, bottom-up means using low-level and image-based outliers and conspicuities [7]. Then, fixation is used to verify the saliency hypothesis and as evaluation tool [85, 9]. After that, saliency detection is defined as a binary segmentation task in [70, 1]. Recently, due to the explosion of convolutional neural networks, hand-crafted features can be eliminated and center bias knowledge can be alleviated [8]. Therefore, SOD can be divided into two categories: 1) traditional methods and, 2) deep learning based methods. Traditional methods make use of low-level features and certain heuristics to detect the salient objects, like image contrast [15], background prior [120] and generic objectness [4]. With the rapidly development of convolution neural networks, SOD transfers from traditional methods into deep learning based methods. Deep learning based methods can improve performance with high efficiency.

2.2.1 Traditional Methods

Traditional methods for SOD often consider intrinsic characteristics to locate the salient pixels. Then, a series of contrast based saliency detection methods have been proposed. Cheng and et al [15] also follow the bottom up data driven saliency detection. They find that saliency depends more on its contrast to the nearby regions and less on distant regions, and thereby propose a global contrast based method to separate the target salient object from its surroundings [15]. Additionally, Ma and Zhang [76] propose a fuzzy growing method for saliency detection based on local contrast analysis. Besides, Perazzi and et al [87] design high-imensional Gaussian filters to measure pixel-accurate saliency map that can uniformly cover the target object and separate foreground and background smoothly. In addition to image contrast, Some works use center-surround contrast to localize salient regions. In [51], Kullback-Leibler divergence between distributions of features like intensity color and orientation is deployed to compute center-surround contrast. Besides, a cost-sensitive max-margin classification is designed to model center-surround contrast in [61]. They treat center patch as a positive sample and surrounding patches as negative samples. Then, a trained cost-sensitive support vector matching (SVM) is applied to separate center patch and surrounding patches to determine the saliency of the center patch. Besides that, distinctiveness of

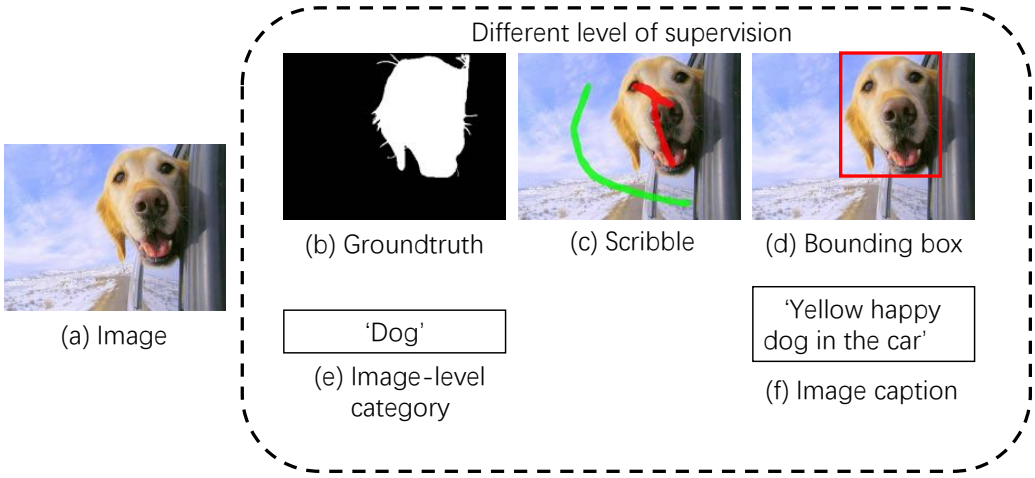


Figure 2.5: Different level of supervision in salient object detection. (a) Input image; (b) Pixel-level annotation, where each pixel has its label; (c) scribble annotation, where there is the red line represents the salient object and the green line represents the background; (d) Bounding box, where there is a rectangular, inside is salient object and outside is background; (e) Image-level category label, the category of the salient object is labelled; (f) Image caption, where there is a sentence to describe the salient object.

complementary cues like texture and structure are introduced to further help detect the salient objects. In [79], they argue that the salient object should satisfy that their local neighborhood is distinctive in both color and pattern. Thus, they propose to use principal component analysis (PCA) to reflect the inner structure of data to represent the distinctness of regions. Additionally, a salient object can also be implied by saptial distribution prior, defined as the more widely spread the color is distributed in the image, the less possible it is for this color to be contained in the salient object. A center prior means that salient object is tend to appear in the center of the image. On the opposite, backgroundness prior assumes that background region is composed of a narrow border of the image. In this case, traditional methods of salient object detection devote to make the best of contrast between features of salient object and background. It is based on the definition of salient object and how human attention is revealed on the salient object.

2.2.2 Deep Learning Based Methods

Deep learning based salient object detection approaches usually train convolution neural networks (CNN) to learn the salient characteristics and predict the corresponding binary saliency maps

of input images. Specifically, given an input image set $I = \{x_n, y_n\}_{n=1}^{N}$, where x_n is the input image, y_n is the corresponding groundtruth for supervision, and N is the total number of training images. Then, the goal of learning is to find the model can minimize the prediction error between predictions and groundtruth. Due to the rapid development of CNN, different network architectures have been designed for SOD, which can be classified into four categories: MLP-based, FCN-based, hybrid network-based, and capcule-based [111].

MLP-based methods. They use multi-layer perceptron (MLP) as clasifier to predict saliency maps. The inputs are deep CNN features of superpixels/pathes [144, 52, 33] orobject proposals [58, 108]. MLP-methods are time-consuming and fail to explore saptial information. Therefore, FCN-based methods are adopted for the efficiency.

FCN-based methods. They use FCN architecture [74] to extract image features and lead to a end-to-end spatial saliency representation learning. [111]. The single feed-forward process helps save both training and inference time with high efficiency.

Hybrid network-based methods. Then, the hybrid network-based methods combine the MLP-based and FCN-based subnets. These methods target at edge preserving. However, the combining of both pixel-level and region-level tends to increase computational complexity, though the performance can be improved.

Capsule-based methods. Finally, the capsule-based methods are based on the fresh familiy of neural networks, called Capsule [34]. Capsules consist of a group of neurons which use vectors instead of scalar values in CNN as bridges to link inputs and outputs. In this case, the features can be comprehensively modeled.

In addition to network architectures, SOD methods can also be classified into either fully-supervised or weakly-/unsupervised methods.

Fully Supervised Methods. Fully-supervised methods provide the pixel-level labelling is like (b) in Fig. 2.5. Each pixel will be labelled as '1' for salient object and '0' for background. This kind of annotations is time-consuming and expensive. Moreover, networks trained with pixel-level annotations are easy to overfit and they tend to predict terrible saliency maps when it comes to real-life images [111].

Weakly-/Unsupervised Methods. In order to save laborious manual labelling, some weak supervision levels are explored, including image-level category labels, image captions, scribbles, bounding boxes, and predictions from traditional methods as a kind of unsupervision. For image-level category labels, the category of the salient object is labelled as different classifications, as demonstrated in Fig. 2.5 (e). Image-level labels can provide the coarse location of the salient object using class activate map [57]. Then, the generated rough pixel-level probability map is refined by

iteratively training [57]. Image-level supervised methods usually consider using CRF to further fine-tune the predictions for better performance. To help models learn more information about both the target salient object itself and the surroundings around it, image captions are introduced since they can provide comprehensive descriptions of target object, as shown in Fig. 2.5 (f). For example, MSW [130] designs a caption generation network to make the network search the entire object to predict captions. In this way, the network can capture entire salient object compared with image-level labels. However, the captions also describe the background, such as 'car' in Fig. 2.5 (f). In this case, the network may also pay attention to the irrelevant background pixels and lead to inaccurate predictions. Except for the language-level labels, scribbles shown in Fig. 2.5 (c) are proposed because it is convenient and fast. It takes only $1 \sim 2$ seconds to label one image [134]. Thus, scribbles are gaining attraction in recent years. However, scribbles cannot provide object boundary information as the annotations do not entirely cover the object. WSSA [134] adopts an extra edge detection task to auxiliary the network predicting smooth boundaries. Further, they design a scribble boosting scheme to rectify the predictions iteratively. Finally, the refined predictions are treated as pseudo labels to train their network to learn high-quality saliency maps. Additionally, bounding boxes like (d) in Fig. 2.5 are another widely used in weak supervision. Bounding boxes can offer correct localization of target objects but lack shape and boundary information of target objects. In [32], the authors first use Grabcut to generate pseudo labels with bounding box supervision. Then, the proposed saliency network is trained through pseudo labels. Although bounding boxes can filter significant complex backgrounds, it is difficult for them to supply detailed target features. Thereby, [72] combines bounding box supervision with pseudo labels predicted by unsupervised methods or traditional methods, where there are no grountruth. They refine the pseudo labels in the light of bounding boxes. Then for unsupervised methods, there is no any groundtruth. Nothing is given except for the input images for training. In this case, pseudo labels are adopted. Pseudo labels refer to the predictions from traditional methods. The coarse maps can help the network learn basic salient knowledge. Nevertheless, if only dependent on the given rough labels, it is difficult for the network to learn enough salient information as there are many errors. It is easy to over-fit to the wrong salient regions. Therefore, how to generate clean pseudo labels is crucial in the level of unsupervision. For example, SVF [133] fuses multiple weak by fast saliency models to obtain more substantial pseudo supervision. Further, they gradually infer the difficulty of each training image to reflect the reliability of pseudo labels. They establish a curriculum learning to generate confident pseudo labels for final training gradually.

In this thesis, we focus on scribble supervision. Scribbles are easy to be labelled compared to pixel-level annotations. However, it is difficult for scribbles to provide integral supervision

for object appearance, although it can indicate both the coarse salient regions and background regions. Recent scribble supervised methods rely on extra boundary supervision and complex training procedures to learn integral object information. Therefore, we try to design an end-to-end learning procedure without using extra information and eliminate post-process.

2.3 Co-Salient Object Detection

Co-salient object detection (CoSOD) is the extension of salient object detection. It can link different images and aim to detect the objects belonging to the same category or the same object from a group of relative images. CoSOD can be regarded as adding correspondence into SOD [17]. It devotes to linking the objects from the same class in different images. Therefore, CoSOD can help provide appearance consistency from different frames in video surveillance.

Co-salient object detection contains two main tasks. One is within-image co-salient object detection, and another is co-salient object detection among a group of images. For within-image co-salient object detection, which is shown in 2.6 (a), it devotes to detecting the salient objects of the same class in a single image and synchronously segmenting the target objects. Within-image co-salient object detection can help detect multiple instances of the same category in an image and thereby more accurate and reliable features can be learnt for object recognition and detection [127]. Therefore, In [127], they first generate multiple object proposals for target objects. Then, they design an optimization algorithm to select proposals of common objects. They consider that common objects should share similar appearance features and low spatial overlap for their selection. Finally, the selected proposals are fused by their clustering-based algorithm and low-rank based algorithm.

In addition to within-image co-salient object detection, co-salient object detection among multiple images is another relative task. As illustrated in Fig. 2.6 (b), this task focuses on detecting and segmenting the co-existed object from different images. In this thesis, We only pay attention to co-salient object detection across multiple images and following CoSOD mainly means this case. Co-saliency can indicate the common visual attention among a group of images. To detect co-saliency is a computational process. The desired algorithm needs to infer both the intra-image consistency and inter-image consistency without the category information. Note that intra-image consistency defines the consistency of pixels of the target object in each image. On the other hand, the inter-image consistency is used to link the feature consistency across different images of the same group. These two informative cues are the basic characteristics that co-saliency should satisfy. Although CoSOD aims to detect common objects from a certain class, category information

Figure 2.6: Different co-salient object detection tasks. (a) Within-image co-salient object detection, where common salient objects are detected and segmented in the same single image; (b) Co-salient object detection among a group of images, where co-existed salient objects from the same category need to be detected and segmented among a group of images, usually over 2 images.

is not provided. This is one of the difficulties of the CoSOD task. Existing CoSOD approaches mainly focus on three key problems: 1) how to extract excellent feature representation for image , 2) how to acquire valuable characteristics for co-salient objects, and 3) how to design frameworks to formulate co-saliency with high efficiency [131].

In the initial co-salient object detection methods, low-level features, like color histograms, Gabor filter, or SIFT descriptor, are introduced, because these features are considered to share a certain consistency [131]. In [59], color and texture are deployed to represent region aspects of local appearance. Color features are from the RGB, Lab, and YCbCr color spaces, and texture features are from histograms of patchwords. Then, a co-multilayer graph is designed, where the node-pair distance is computed as similarity to reflect consistency. Besides, in [73], They propose a hierarchical segmentation based model for co-saliency detection. They use regional histograms, generated by quantizing each color channel in the Lab color space, to measure regional similarities between region pairs. They also use regional contrasts within each image to evaluate intra-saliency. In addition to low-level features, high-level features can embed more semantic information. Specifically, high-level features here mean deep layers of a CNN. Therefore, high-level features are not limited by appearance variation, shape variation, scale, or luminance of the common objects. Thus, high-level features can provide more deep concept-level characteristics of target objects. Moreover, they can help the network learn both semantic inter-image consistency and intra-image consistency. Additionally, high-level features can help distinguish different objects

from the view of semantic information. However, there is no detailed information in high-level features. It is difficult to determine a specific instance based on semantic information. For example, if the co-salient object is the same person but around a crowd of people, high-level features can only provide the information of human but cannot distinguish different people. In this case, low-level features can be adopted to separate different people. Thereby, both low-level and high-level features are useful in co-salient object detection. They can not only individually handle CoSOD, but also cooperate with each other.

In recent years, deep learning based co-salient object detection models have achieved great performance. Most of them try to mine hidden patterns and learn discriminative feature representation. Some works establish graphs to model the relationship among pixels from a group of images [138, 44, 42, 37, 119, 43], then the co-salient objects can be mined with consistent features. Some works adopt extra salient object detection to mine salient objects first and then conduct CoSOD [140, 46, 139]. Besides, SAEF [101] proposes to use saliency proposals generated by unsupervised deep learning based models first and then conduct CoSOD according to those proposals. Other works [25, 143, 136] try to formulate shared attributes among input images to reflect the co-salient pixels and use classification information as a supplement of semantic information. In CoEGNet [23], edge detection is used for better structure prediction. More information on CoSOD can be found in surveys[24, 131, 17].

In CoSOD, the main challenges is how to link both intra- and inter-image consistency. Recent methods adopt attention mechanism to localize the common objects. However, their methods can only response limited pixels. They are highly dependent on extra training-set and classification information to learn more discriminative co-salient features. Therefore, in this thesis, we thoroughly explore the intrinsic characteristics of co-salient objects and background to realize CoSOD without using the SOD dataset or extra classification information.

Chapter 3

Fast Pixel Matching for Video Object Segmentation

Video object segmentation, which helps predict pixel-level foreground and background masks, is a fundamental task in video understanding. It can assist other tasks by providing observation of the target object, like how the target changes in sequences in video surveillance. In this chapter, a new framework for mask-propagation based semi-automatic video object segmentation is studied, where the first frame's pixel-level mask is given as an indicator of the target object. The task needs to detect and segment the target object in the following video sequence. We deploy an attention mechanism in a spatial direction to realize pixel matching between the reference frame and current frame to localize the target object in the current frame. Then a channel attention mechanism is deployed to enhance the matched feature further. Our method can obtain both outstanding performance and high efficiency.

3.1 Motivation

Video object segmentation (VOS) has been attracting increasing attention in recent years due to its significance in video understanding. The aim of this task is to track the target object from the first frame to the end of the video sequence and segment all the pixels belonging to the tracked target object, which faces problems of object occlusion and appearance variance.

To tackle these problems, some studies adopted online-training mechanism [10, 78, 105, 86]. Given the ground-truth mask of the first frame in a test video, they used it to fine-tune the model to obtain the object appearance. In the following inference process, they used the predicted masks

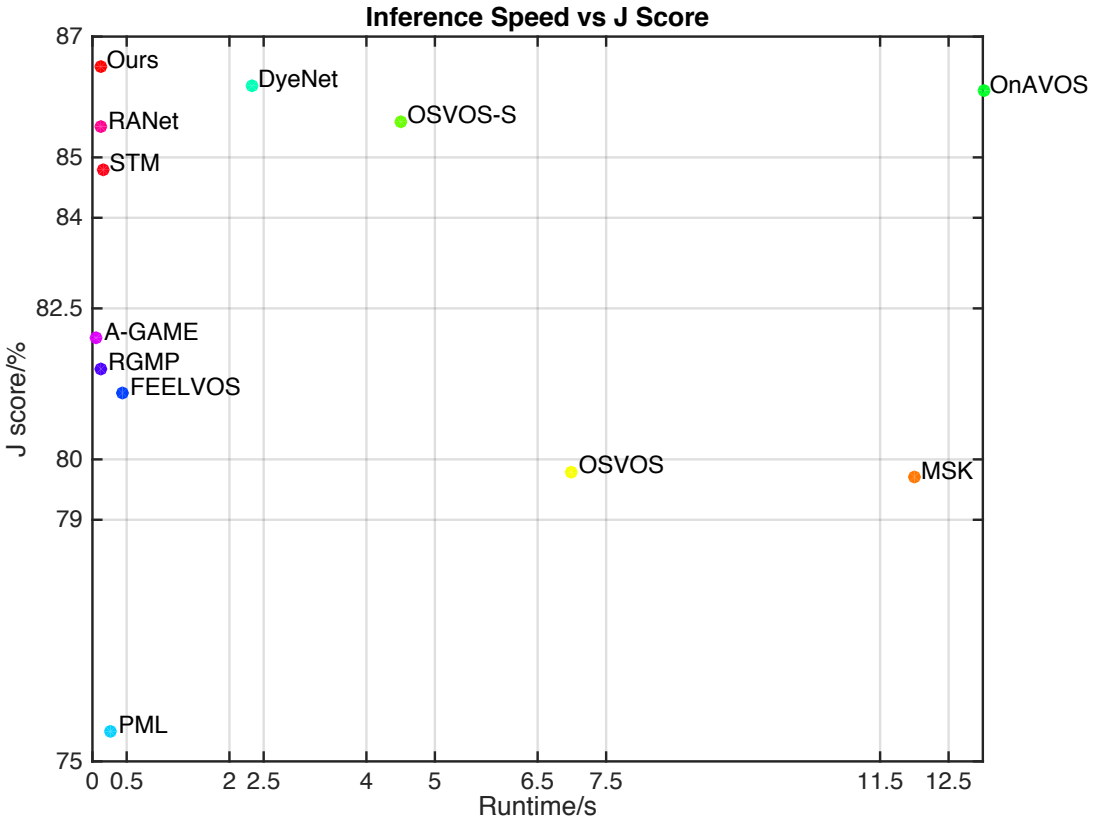


Figure 3.1: The IoU score ($\mathcal{J}$) versus running time on each frame (s) for various VOS approaches on the DAVIS-2016 validation set. Our model can keep a good balance between performance and efficiency.

to further fine-tune their models. With fine-tuning, the models can adapt to object appearance change, though, the online learning process is time-consuming and inefficient.

Recently, boosted by the rapid development of mask-propagation based VOS models [123, 47, 65], a better balance between speed and accuracy is reached. The core idea of these methods is to use the estimated mask of the previous frame to guide the model to make segmentation prediction for the current frame. For example, Perazzi et al. [86] proposed to use guidance of previous predicted mask as guidance for the network to learn mask prediction and it proposed a combination of offline and online training method to train the model. They firstly used static image datasets for offline training, and then used the first frame of a test video sequence to fine-tune the model. Oh et al. [123] proposed a Siamese encoder-decoder network with guidance of the previous

mask to produce the target object probability map. Johnander et al. [47] offered an appearance module which utilized a class-conditional mixture of Gaussians to model the foreground object appearance for mask prediction. Sun et al. [97] considered both the mask of previous frame and the optical flow to predict target mask. These approaches are usually faster than online training based VOS methods, but they are less adaptive to object appearance variation.

Both online training and mask-propagation based VOS models have limitations, a balance between segmentation accuracy and running speed is crucial for VOS. Early mask-propagation based networks use current frame with previous estimated mask [86] or adding first frame with its provided mask as reference information [123] to directly predict the segmentation mask of current frame. Additionally, Sun et al. [97] used optical flow to build relationship between the previous and the current frames. Different from these methods, we design an attention-based pixel-matching module to find the pixels belonging to the target object in the current frame based on the feature similarity between the current frame and reference frames. In order to capture the object feature without the interference of background, we choose to mask it out and discard the background pixels. However, the target object is varying frame by frame, such process will cause large object appearance variation. Therefore, we choose to use both the first frame and the previous frame as references to provide object information for our pixel-matching module.

With the target object's appearance information, we need to determine the target object location, in terms of mask, in the current frame. We design our model based on mask-propagation to keep efficiency, where the non-local structure [116] is adopt to generate the object mask using the obtained target object's appearance information. Specifically, we design a video object segmentation model called Non-local Pixel-Matching network with Channel Attention (NPMCA-net), which includes a newly designed pixel-matching module and a channel attention module. The pixel-matching module is designed to match pixels between the target frame and the reference frames with given ground-truth mask or estimated mask. The channel attention module is used to augment the matched feature map to achieve better decoding. Extensive experiments have shown that our network can achieve a new state-of-the-art performance without loss of efficiency. To better display the accuracy and speed trade-off, we plot our IoU score versus speed in Fig. 3.1. Our NPMCA-net can achieve both high performance and high efficiency at the same time. Our main contribution is summarized as follows:

- We propose a video object segmentation model (NPMCA-net) that strikes a good balance between accuracy and running speed. The model does not rely on online fine-tuning technique, so as to lower the computational demands, yet it can adaptively catch the target

object's appearance variation by using both image and predicted mask information in the previous frame.

- Our proposed non-local pixel-matching module can effectively predict the target object mask by aggregating multi-frame information. Moreover, the proposed model also provides high level interpretability by visualizing the obtained feature maps.
- Our model achieves new state-of-the-art performances on DAVIS-2016 (IoU: 86.5%) and DAVIS-2017 (IoU: 72.2%) datasets, using the same experimental setting.

3.2 Method

Our motivation is to make VOS model adaptive to object appearance variation and occlusion, and keep a high efficiency at the same time. Therefore, we design a new mechanism by matching the pixels in target frame and reference frames (first and previous frames) to acquire the predicted mask for the target frame.

3.2.1 Video Object Segmentation Architecture

Given a video with annotated mask for the first frame, we need to segment the rest frames according to the given mask. In VOS, object appearance is often changing frame by frame for the video object segmentation task. Thus, it is not sufficient if we only care about the object appearance in the first frame, especially when large object appearance variation occurs in the middle of the video.

As illustrated in Fig. 3.2, we provide three different kinds data for the three encoders: the target frame encoder takes the current frame with the estimated labels of the previous frame as 4-channel input [123]; two parameter-shared reference frame encoders take the first frame and the previous frame as input, respectively. Note that when providing data for reference frame encoders, background pixels from the first frame and the previous frame are removed using groundtruth (first frame) or estimated mask (previous frame). Whist for the target frame encoder, background pixels are not masked-out since the masks for the current and previous frames are different. Then, the feature maps of reference and target frames are extracted by respective encoders. In this way, we can obtain the changing object appearance information and target frame features.

Following that, the feature maps are input into our non-local pixel-matching module. The target feature map is matched with the feature maps from two references using our newly designed

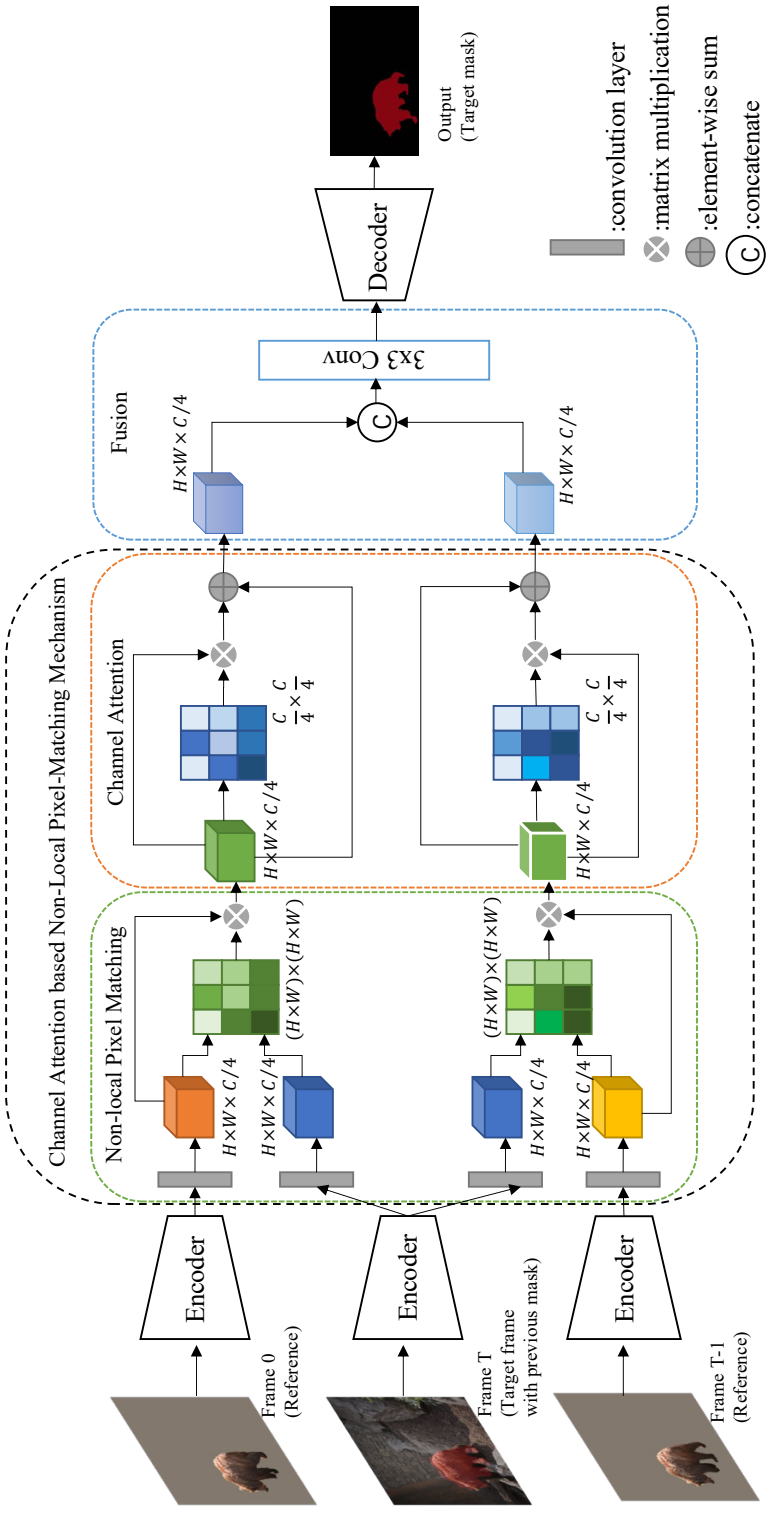


Figure 3.2: The framework of our NPMCA-net. It consists of three encoders, where the encoders for the two reference frames are shared. NPMCA-net contains a non-local pixel-matching module, a channel attention module, a fusion module and a decoder.

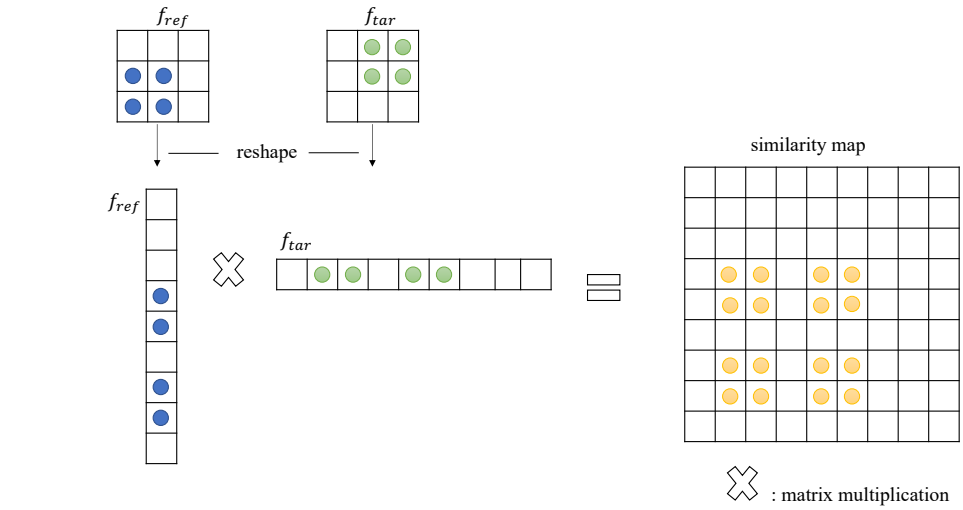
non-local pixel-matching module to localize the target objects. In this process, the target feature is matched with two references one by one, individually. Therefore, there are two output feature maps: one is the matched feature map of the target frame with the first frame, and the other one is the matched feature map of target with previous frame. With the help of the previous frame, our network can adapt to object appearance variation, since the gap between the current and previous frames are smaller than that between the current and first frame. On the other hand, if we only consider the previous frame, for the occlusion case, the model will lose the initial object appearance for frames after the occlusion.

After that, the channel attention module is applied to strengthen features by allocating different weights for each feature channel. Once the features are matched and enhanced, the obtained two feature maps are concatenated, where a 3×3 convolution layer is used to fuse the two feature maps. Finally, the fused feature map is decoded by the decoder to predict and output the target object masks. Our method can be viewed as an encoder-decoder process, which can directly obtain the segmentation mask of current frame without any post-processing.

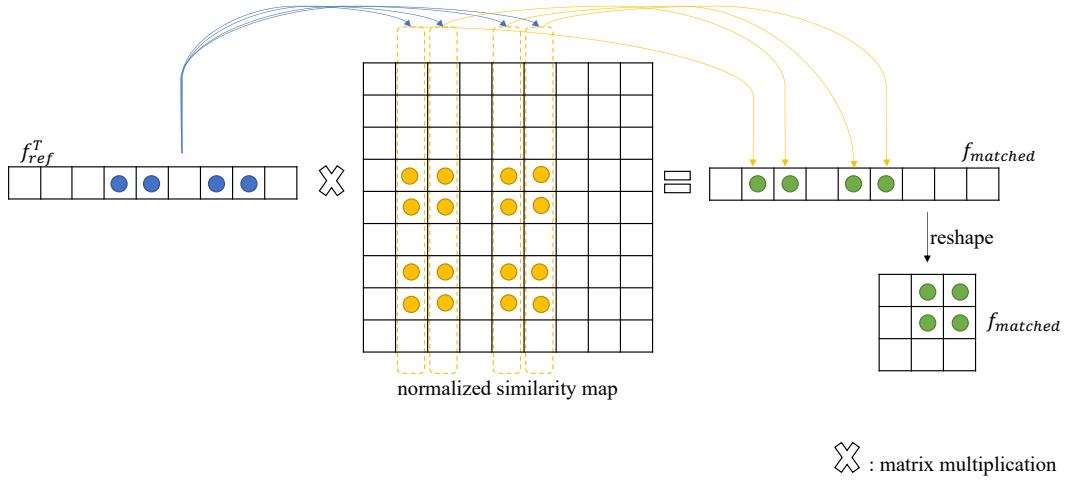
3.2.2 Non-Local Pixel Matching with Channel Attention

Our NPMCA-net contains two parts, including a non-local pixel-matching module (NLPMM) and a channel attention module (CM). The CM is in series with the NLPMM. The NLPMM is a non-local structure which can match pixels over the whole feature map. And CM conducts self-attention through the channel dimension instead of the spatial dimension to strengthen the feature representation. With the combination of these two modules, our network can obtain feature representations of the foreground objects for the target frame. The details are discussed as follows.

Non-Local Pixel-Matching Module. The non-local pixel-matching module is one main module of our NPMCA-net, which is used to obtain object appearance of the target frame and localize the target object simultaneously by matching the feature maps of the reference frames and the target frame. Different from the matching process using convolution layers [94] or using metric learning to pull in similar embedding vectors and push away different embedding vectors [104, 12], we directly compute similarities between pixels. The framework of NLPMM is illustrated in Fig.3.4(a). The inputs of this module are the feature map of reference frame and the feature map of target frame (defined as $f_{ref} \in \mathbb{R}^{H \times W \times C}$ and $f_{tar} \in \mathbb{R}^{H \times W \times C}$, where H, W, C are the height, width, and channel number, respectively) extracted from respective encoders. In order to reduce memory and improve efficiency for our approach, once feature maps are fed into the module, a 3×3

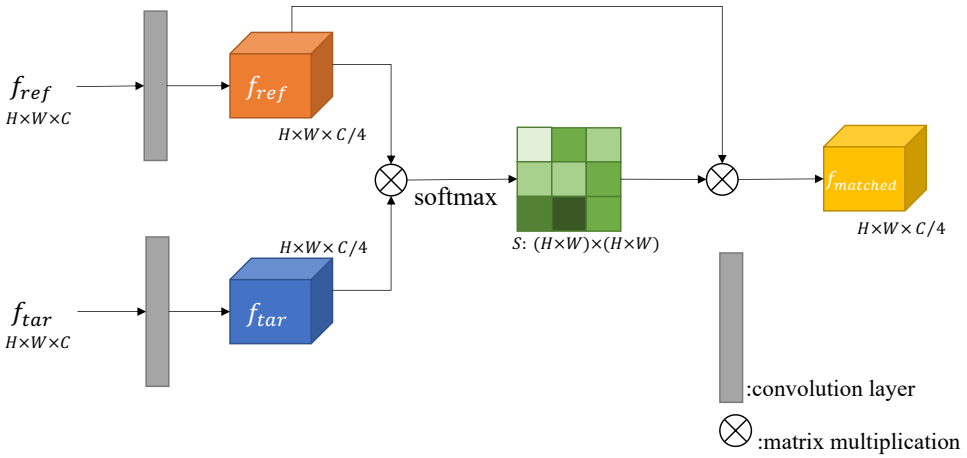


(a) The Process of Similarity Computation

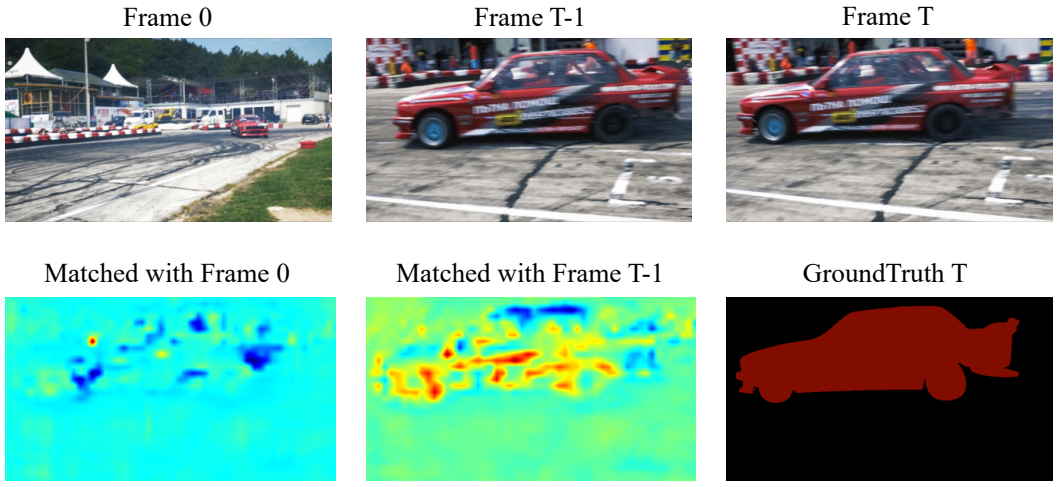


(b) The Process of Matching and Localization

Figure 3.3: (a) The process of similarity computation (Eq.(3.1)). The two reduced feature maps are reshaped into $f_{ref} \in \mathbb{R}^{N \times \frac{C}{4}}$ and $f_{tar} \in \mathbb{R}^{\frac{C}{4} \times N}$, and the similarity is computed by the matrix multiplication. (b) The process of target object matching and localization (Eq.(3.3)).



(a) Non-Local Pixel-Matching Module



(b) Visualization of Output Feature Map of NLPMM

Figure 3.4: (a) Framework of non-local pixel-matching module (NLPMM). Our NLPMM has two inputs,including the reference feature map and the target feature map. The output is the matched feature map. (b) Visualization of output feature map from NLPMM. The matched feature map can coarsely acquire the foreground object appearance and its location.

convolution layer with padding is used to reduce the channel number of input feature maps from C to $C/4$, the new feature maps are with size $f_{ref} \in \mathbb{R}^{H\times W\times \frac{C}{4}}$ and $f_{tar} \in \mathbb{R}^{H\times W\times \frac{C}{4}}$, respectively. After that, the two reduced feature maps are reshaped to $f_{ref} \in \mathbb{R}^{N\times \frac{C}{4}}$ and $f_{tar} \in \mathbb{R}^{N\times \frac{C}{4}}$, where $N = H \times W$. The similarity between pixels in the two feature maps is computed:

$$S = f_{ref} f_{tar}^{T}, \tag{3.1}$$

with $S(i, j)$ measuring the similarity between i^{th} position on reference feature map and j^{th} position on target feature map. The similarity of each pixel is calculated in a non-local way, where all positions of the two feature maps are included. Meanwhile, it computes the relation between two spatial pixels from two temporal frames because the inputs are from a temporal sequence. Therefore, it is a space-temporal similarity calculation. After that, instead of directly using the calculated result, we apply softmax to normalize the non-local similarity map S, and obtain S' ($S' \in \mathbb{R}^{N\times N}$, $N = H \times W$), with its element value $S'(i, j)$ being

$$S'(i, j) = \frac{exp(S(i, j))}{\sum_{i=1}^{N} exp(S(i, j))}. \tag{3.2}$$

With Eq.(3.1) and Eq.(3.2), we can generate the relations between any two pixels in the target feature map and the reference feature map. The pixel pair with a large similarity value has high probability belonging to the same pixel of one foreground object. In this case, we can not only match the object appearance but also localize the object. Finally, the new matched feature map $f_{matched}$ is calculated by a matrix multiplication between the transpose of the reduced reference feature map f_{ref} and the non-local similarity map S' ,

$$f_{matched} = f_{ref}^{T} S'. \tag{3.3}$$

Finally, the matched feature map is reshaped back to $f_{matched} \in \mathbb{R}^{H\times W\times \frac{C}{4}}$.

The coarse mask of the target frame can be obtained by the matrix multiplication between the reference feature map and the similarity map, namely, we can use Eq.(3.3) to obtain the pixels of foreground objects in the target frame. To more intuitively understand the matching and localization process, we show the process in Fig.3.3. Fig.3.3(a) shows how the similarity map is computed, and Fig.3.3(b) displays how the matching process can also accomplish the localization. Therefore, we can obtain foreground object appearance and its location at the same time. Besides, visualization of the output of our non-local pixel-matching module is shown in

Fig.3.4(b). It can be found that this matching module is able to localize the object and mask the target object appearance. The highlighted part (warm color) in the "matched with frame T-1" better demonstrates the matched pixels for the target object. When there is only frame 0 to be referred, it is difficult for the network to find out the pixels for the moving object in the case of large appearance variation.

Channel Attention Module. We adopt a channel attention module after the non-local pixel-matching module to strengthen the feature representation of foreground object in this task. The details of our channel attention module is illustrated in Fig.3.5(a). The input for this module f_{in} is the output feature map of non-local pixel-matching module, *i.e.*, $f_{in} = f_{matched}$ and $f_{in} \in \mathbb{R}^{H \times W \times \frac{C}{4}}$. In order to compute the inter-dependencies between different channels, f_{in} is first reshaped into $f_{in} \in \mathbb{R}^{N \times \frac{C}{4}}$, where $N = H \times W$. Then the channel attention map $A \in \mathbb{R}^{\frac{C}{4} \times \frac{C}{4}}$ is computed by:

$$A = f_{in}^T f_{in}, \tag{3.4}$$

$$A'(i,j) = \frac{exp(A(i,j))}{\sum_{i=1}^{N} exp(A(i,j))}, \tag{3.5}$$

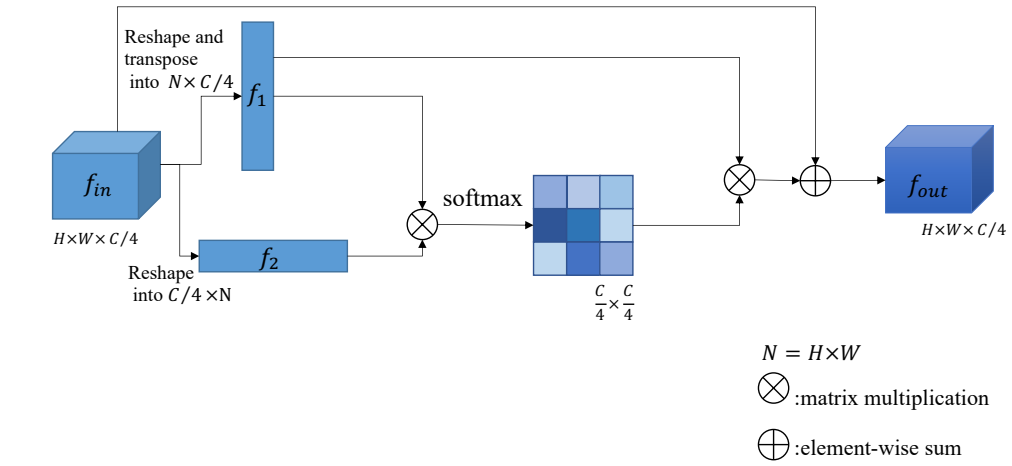
where $A(i,j)$ measures the relationship between i^{th} channel and j^{th} channel of f_{in}. Then matrix multiplication is applied to get the strengthened feature map. Mathematically, the strengthened feature is:

$$f_A = f_{in} A'. \tag{3.6}$$

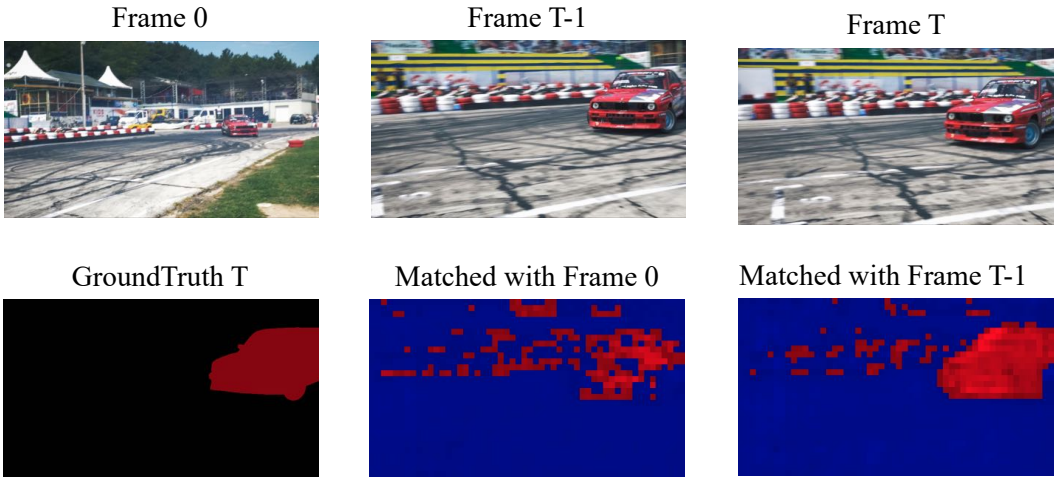
Then the strengthened feature map f_A is reshaped back into the size of input feature map, *i.e.*, $f_A \in \mathbb{R}^{H \times W \times \frac{C}{4}}$. The final output of channel attention module is the weighted sum of the strengthened feature map and the module input feature map f_{in}:

$$f_{out} = \gamma f_A + f_{in}, \tag{3.7}$$

where $\gamma \geq 0$ is a learned parameter. We do not apply any convolution layer in the channel attention map. The channel attention map is in series with the non-local pixel-matching module to strengthen the representation of feature map instead of adopting a parallel mode in [28]. This module can help further lay stress on the channels which are more related to the semantic of the target object. In this case, the features can be further strengthened. Some visualizations of the output feature map of the channel attention module are displayed in Fig.3.5(b).



(a) Channel Attention Module



(b) Visualization of Output Feature Map of CM

Figure 3.5: (a) Framework of channel attention module (CM). The input of CM is the output of NLPMM (matched feature map), and it outputs the strengthened feature map. (b) Visualization of Output feature map from CM. CM is able to strengthen the feature representation.

3.2.3 Two-stage Training Method

We take two-stage training for our network. Firstly, we pre-train our NPMCA-net through static images. Then, we use the video object segmentation datasets to fine-tune the model. We use IoU loss in [65, 64] and Adam [50] optimizer with randomly cropped resolution of (256×432) patches for both pre-training and fine-tuning. All experiments are running on one NVIDIA GeForce 2080 Ti GPU.

Pre-training on static images. Pre-training on static images for video object segmentation is becoming popular recently since it can help the network adapt to different foreground object appearance. We follow several successful practice in [123, 86, 82] to pre-train our network by applying random affine transformation on static images. Specifically, we apply different random affine transformation on both the foreground from one dataset and the background from another dataset to simulate a video sequence. We use saliency datasets MSRA10K [15], ECSSD [124], segmentation datasets Pascal VOC dataset [20] and COCO [67]. In this case, the network can be adapted to different object appearance and categories, so as to avoid easy over-fitting. For pre-training, we set a fixed learning rate as $1e$-5.

Fine-tuning on videos. Then, we fine-tune the pre-trained model on video object segmentation dataset. We only use DAVIS-17 [90] training set for fine-tuning. During training, we sample three frames in temporal order to obtain temporal information. In order to acquire big variation of object appearance for a long time, we randomly skip frames for sampling. The maximum random skip is 5 and the learning rate for fine-tuning is set as $1e$-6.

3.2.4 Inference

Our network is based on the assumption that the ground-truth mask of the first frame is given for semi-supervised video object segmentation. In other words, the first frame is set as the reference frame for all the rest frames. Therefore, to make our network efficient, we only compute the feature map of the first frame once for a test video clip. Following the architecture of our approach, we use previous frame with predicted segmentation mask as another reference frame. We also follow [123] to set three different scale sizes and compute their average as the final output.

Multi-object case. We use softmax aggregation [123] to softly combine multiple objects. Finally, the output probability map is computed by:

$$P_{i,m} = \frac{p_{i,m}/(1-p_{i,m})}{\sum_{j=0}^{M} p_{i,j}/(1-p_{i,j})}, \tag{3.8}$$

where $p_{i,m}$ is the output probability of instance m at position i. $m = 0$ is for background and M is the total number of instances. We use Eq.(3.8) to compute the probability map of multi-objects and apply it to next frame inference.

3.3 Experiment

3.3.1 Implementation Details

Encoder. We design three encoders based on ResNet-50 [31] for three inputs (two references and one target). Like [123], the target frame encoder takes 4-channel inputs and two reference frame encoders take 3-channel inputs. Instead of using res5 in [123], we take res4 as the final encoded feature map, whose channel number is 1024. This is because the feature map of res5 is with low resolution, making it inaccurate for small objects. On the other hand, three res5 encoders will cause large memory occupation.

Decoder. After the fusion layer, the fused feature map is finally fed into the decoder. Similar to [123], the decoder also takes the encoder stream through skip-connection as input to produce the mask. With the help of skip-connection, the high resolution feature can replenish the missing information. Finally, the feature map is gradually upsampled with a factor of two till it reaches the same size as input.

3.3.2 Experiment Results

We evaluate our network on video object segmentation datasets, DAVIS-2017 [90], DAVIS-2016 [88] and SegTrack-v2 [56]. The evaluation metrics include mean intersection-over-union (IoU) of predicted mask and the ground-truth ($\mathcal{J}$), contour accuracy between contour points on predicted mask and the ground-truth ($\mathcal{F}$), and the average of the two metrics ($\mathcal{J}\&\mathcal{F}$).

DAVIS-2017. DAVIS-2017 is a multi-object dataset. There are 90 videos in total, 60 for training and 30 for validation. We evaluate our method on its validation set. The comparison results with recent state-of-the-art approaches are shown in Table 3.1. The results are listed from the lowest score of $\mathcal{J}$ to the highest score. The upper part is from approaches with online-learning or with optical flow. It can be found that our method achieves comparable scores with the best performing ones. Our score is slightly lower than PReMVOS [75], but PReMVOS needs longer running time than all other approaches because both online-learning and optical flow need expensive computational cost. We reach the best performance compared with all other methods without online-learning or optical flow. It can be demonstrated that our NLPMM can realize find out where

the target object is in current frame. Further, we directly using masked-out object as the input for reference, making our model less sensitive to the influence of backgrounds while focusing on the object itself. By doing this, our method can capture enough object features. Besides, using the masked-out objects of the first frame and the previous frame as references provides enough information for handling appearance variation.

DAVIS-2016. DAVIS-2016 contains 50 videos (30 for training and 20 for validation) for single-object video object segmentation. We report comparison results of the validation set in Table 3.2. It can be found that our approach achieves better performance than the methods using pixel-matching or metric learning, such as PLM [94], PML [12], FEELVOS [104], and RGMP [123]. We also obtain higher score than other methods without online learning. For metric $\mathcal{J}$, our method is 1.7% higher than STM [82], whist for the contour accuracy, our method is 0.8% lower than STM [82], this might be caused by the adopted IoU loss. Moreover, our results are competitive with online-learning based methods. According to the running time listed in Table 3.2, our approach can achieve a good balance between accuracy and efficiency. It demonstrates that our NLPMM is able to localize moving objects with masked-out object references. Additionally, pre-training with statistic images also helps network to adapt to different object classes. In this way, our approach does not rely on online training to learn the object information of current video.

SegTrack v2. We also evaluate our network on the SegTrack v2 [56] dataset. The results are shown in Table 3.3. It can be found that our network also achieve competitive performance on SegTrack v2 dataset under the same level comparison. Therefore, our network has competitive generalization ability. Our performance even defeat MSK [86] and MaskRNN [38], where online training is used. We set the same training dataset as DMM-net. it can be seen that our method can obtain comparable results with DMM-net. However, we obtain lower performance than DyeNet. This phenomenon may be caused by the fact that they use template matching, which predicts bounding box of the target object first then conduct segmentation. In this way, much background noise can be reduced. In the SegTrack v2 dataset, there are several videos with the background very similar to the target object. In such cases, template can better decrease the disturbance of background. However, for other datasets, such as, DAVIS17, DAVIS16, such conditions are not satisfied, the performance of DyeNet is lower than ours, as reported in Table 3.1 and Table 3.2.

3.3.3 Qualitative Results

Qualitative results on two DAVIS datasets are shown in Fig. 3.6 and Fig. 3.7. For each displayed video, we choose 5 frames with the cases of large object appearance variation or occlusion. It can

Table 3.1: Evaluation on DAVIS-17 validation set. 'OL' denotes online-learning. 'OF' means using optical flow. Our NPMCA-net obtains a score of 3% higher than STM [82].

Method	OL	OF	$\mathcal{J}$ (%)	$\mathcal{F}$ (%)	$\mathcal{J}\&\mathcal{F}$ (%)	Time (s)
OSVOS [10]	✓		56.6	63.9	60.3	10
OnAVOS [105]	✓		61.6	69.1	65.4	13
OSVOS-S [78]	✓		64.7	71.3	68.0	4.5
AGSS-VOS [65]		✓	64.9	69.9	67.4	-
CINN [6]	✓		67.2	74.2	70.7	>120
PReMVOS[75]	✓	✓	**73.9**	**81.8**	**77.8**	-
VideoMatch[39]			56.5	68.2	62.4	0.35
MAARU [29]			61.3	65.3	63.3	0.13
RANet [118]			63.2	68.2	65.7	-
RGMP [123]			64.8	68.6	66.7	0.28
DIPNet [36]			65.3	71.6	68.5	-
A-GAME [47]			67.2	72.7	70.0	-
DMM-Net [129]			68.1	73.3	70.7	-
FEELVOS [104]			69.1	74.0	71.6	0.51
STM [82]			69.2	74.0	71.6	-
TVOS [142]			69.9	74.7	72.3	0.027
NPMCA-net (Ours)			**72.2**	**77.4**	**74.8**	0.25

Table 3.2: Evaluation on DAVIS-16 validation set. 'OL' denotes online-learning. 'OF' means using optical flow. Our NPMCA-net can even achieve a bit higher performance than methods with online-learning.

Method	OL	OF	$\mathcal{J}$ (%)	$\mathcal{F}$ (%)	$\mathcal{J}\&\mathcal{F}$ (%)	Time (s)
MSK [86]	✓	✓	79.7	75.4	77.6	12
OSVOS [10]	✓		79.8	80.6	80.2	7
MaskRNN [38]	✓	✓	80.7	80.9	80.8	-
CINN [6]	✓		83.4	85.0	84.2	>30
Lucid [48]	✓	✓	83.9	82.0	83.0	-
PReMVOS [75]	✓	✓	84.9	**88.6**	**86.8**	>30
OSVOS-S [78]	✓		85.6	86.4	86.0	4.5
OnAVOS [105]	✓		86.1	84.9	85.5	13
DyeNet [62]	✓		**86.2**	-	-	2.32
PLM [94]			70.0	62.0	66.0	0.3
PML [12]			75.5	79.3	77.4	0.28
VideoMatch[39]			81.0	-	-	0.32
FEELVOS [104]			81.1	82.2	81.7	0.45
RGMP [123]			81.5	82.0	81.8	0.13
A-GAME [47]			82.0	82.2	82.1	0.07
MAARU [29]			83.9	83.8	83.9	0.12
RANet [118]			85.5	85.4	85.5	0.13
DIPNet [36]			85.8	86.4	86.1	0.92
STM [82]			84.8	**88.1**	86.5	0.15
NPMCA-net (Ours)			**86.5**	87.3	**86.9**	0.11

Table 3.3: Evaluation on SegTrack v2. The IoU peformance for the baseline methods are from [123] and [129]. 'OL' denotes online-learning.

Method	OL	IoU (%)
OnAVOS [105]	✓	66.7
MSK [86]	✓	70.3
MaskRNN [38]	✓	72.1
CINN [6]	✓	77.1
Lucid [48]	✓	**77.6**
RGMP [123]		71.1
DIPNet [36]		73.8
DMM-Net [129]		76.7
DyeNet [62]		78.3
NPMCA-net (Ours)		76.1

Table 3.4: Training methods analysis on DAVIS-2017 validation set. The two-stage training method helps our NPMCA-net better adapt to different categories. With only DAVIS-2017 training set, the network is easy to get over-fitting.

Training Method	$\mathcal{J}(\%)$	$\mathcal{F}(\%)$
Pre-train only	65.7	71.3
Fine-tuning only	41.0	43.9
Full Training	**72.2**	**77.4**

be found that our model can handle different challenges. For example, our model performs well with large object appearance variation cases like in row 2 and 3 in Fig. 3.6 and row 1 in Fig. 3.7. Besides, our model can also segment each object when they are occluded by background as shown in row 1 in Fig. 3.6 and row 2, 3 in Fig. 3.7. The qualitative comparison between our model and other methods are shown in Fig. 3.8.

3.3.4 Ablation Studies

Two-stage training method.

We firstly conduct the ablation study for the two-stage training method, and the results are displayed in Table 3.4. It is surprising to find that the performance of pre-train-only case is much better than fine-tune-only case. Both the intersection-over-union score ($\mathcal{J}$) and the contour accuracy ($\mathcal{F}$) of pre-train-only are almost 25% larger than of fine-tuning-only. It proves that two-stage training is necessary. If we only train on DAVIS-2017, the categories are far less enough. It can also be found that our approach will perform better when more categories are used for training. The combination of pre-train and fine-tuning achieves the best performance, because pre-training help our model adapt to large categories and fine-tuning help our model to obtain temporal information and adapt to video sequence.

Different Modules.

We also conduct ablation experiments with some components disabled or removed, and the results are displayed in Table 3.5. We test three different combinations of the channel attention module

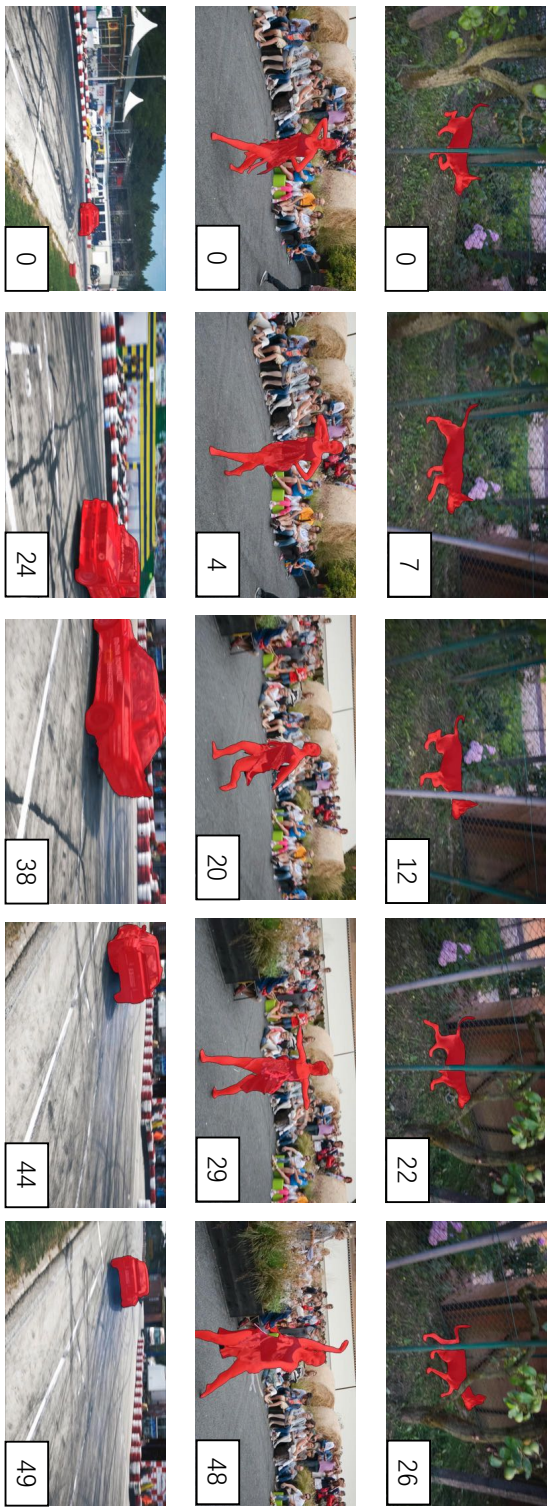


Figure 3.6: The visual results of our NPMCA-net on DAVIS-2016.

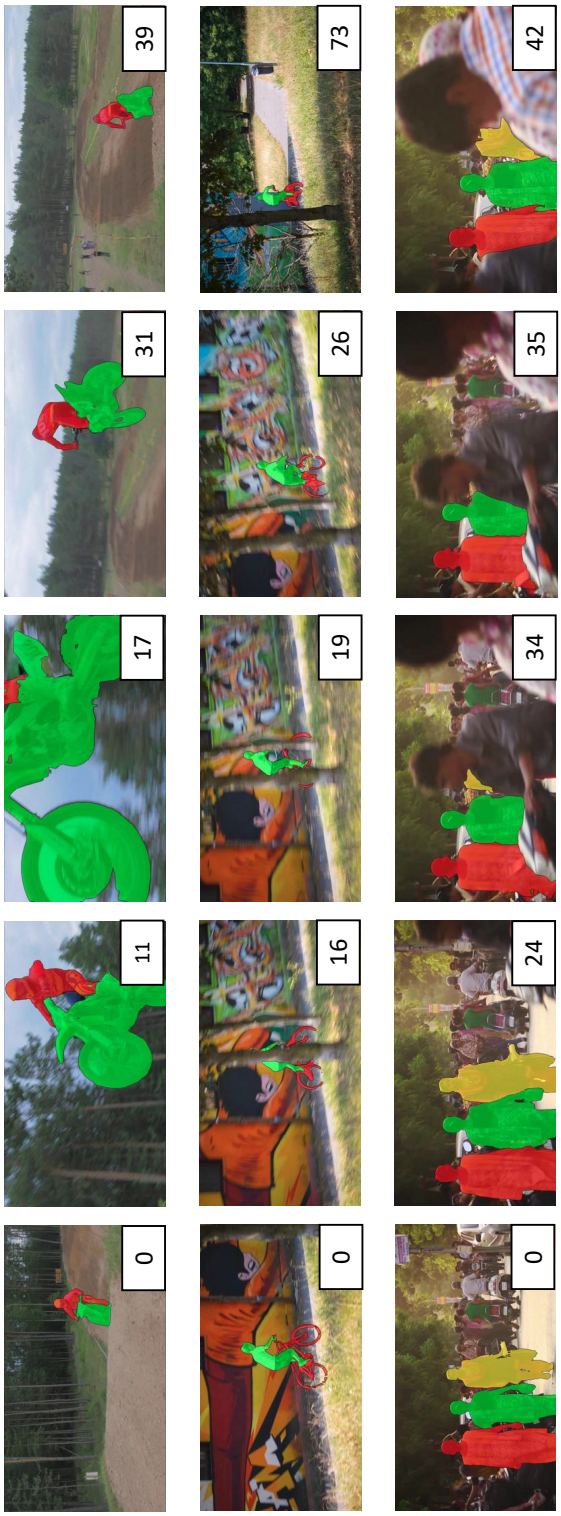


Figure 3.7: The visual results of our NPMCA-net on DAVIS-2017.

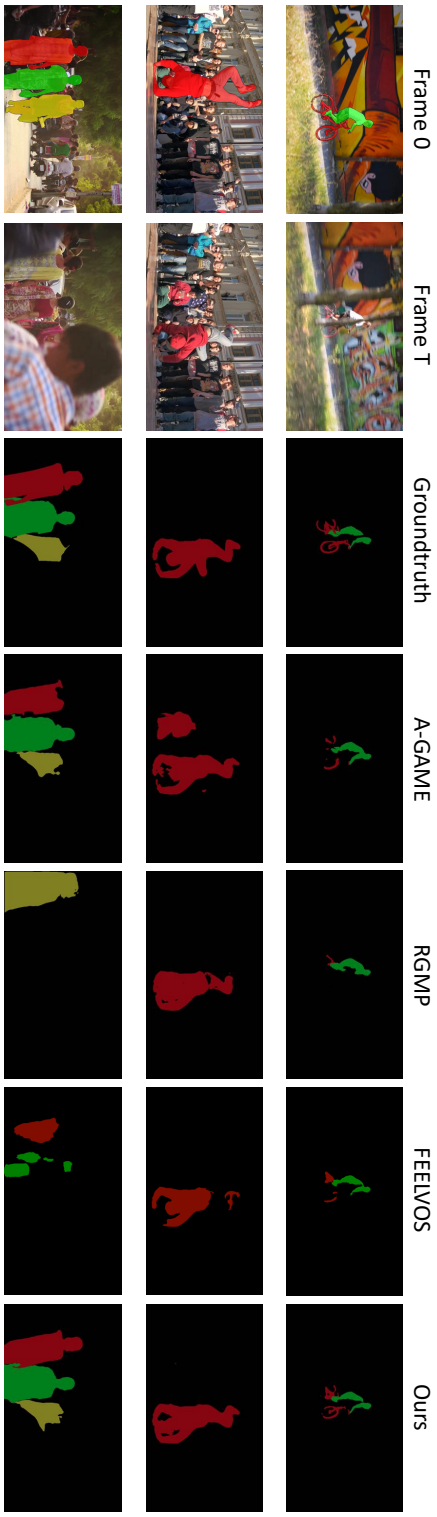


Figure 3.8: The visual comparison with other approaches on DAVIS-2017.

Table 3.5: Network module analysis on DAVIS-2017 validation set. 'CM' denotes to the channel attention module, and 'PM' denotes that the input of current frame with the predicted mask from the previous frame.

	CM	PM	$\mathcal{J}(\%)$	$\mathcal{F}(\%)$
1		✓	68.8	73.7
2	✓		66.9	72.6
3	✓	✓	**72.2**	**77.4**

and the use of the predicted mask from the previous frame. If we remove our channel attention module, the IoU score and the contour accuracy are 3.4% and 3.7% lower than the full combination, respectively. Therefore, we can conclude that the channel attention module can strengthen the feature representation to help our network better adapt to foreground pixels. On the other hand, if we take out the predicted mask from the previous frame, the IoU score and the contour accuracy are 5.3% and 4.8% lower than the full combination, respectively, which proves that the predicted mask from the previous frame can guide our network to segment the foreground object. Overall, the full NPMCA-net achieves the best performance. It demonstrates that the channel attention module and the use of the predicted mask for the previous frame benefit from each other.

Encoder Setting.

Finally, we conduct the ablation study on the setting of encoders with only training with DAVIS-2017 dataset. we conduct the experiment to show the necessity of the parameter-shared encoder for the two references and different encoder for the target frame. The results is shown in Table 3.6. 'One encoder' denotes to use same encoder for the three inputs and 'Two encoders' denotes to parameter-shared setting. It can be found that with only one encoder, the result is almost 5% lower than the two-encoder setting. VOS aims to segment the target object from the first frame to the end. To capture consistent reference object feature information, we set parameter-shared encoder for the first frame and previous frame (where background is masked out). Parameter-shared can map the input reference features into the same representation space, thereby the two reference frames' information can be equally treated. Additionally, parameter-shared can reduce parameters for training. If we use just one encoder for the first, the previous and the target frames, the network

Table 3.6: Encoder settings analysis on DAVIS-2017 validation set. 'One encoder' denotes to using same encoder for all the inputs 'Two encoders' denotes to the setting of parameter-shared only for the reference frames.

Encoder Setting	$\mathcal{J}(\%)$	$\mathcal{F}(\%)$
One encoder	34.7	38.6
Two encoders	41.0	43.9

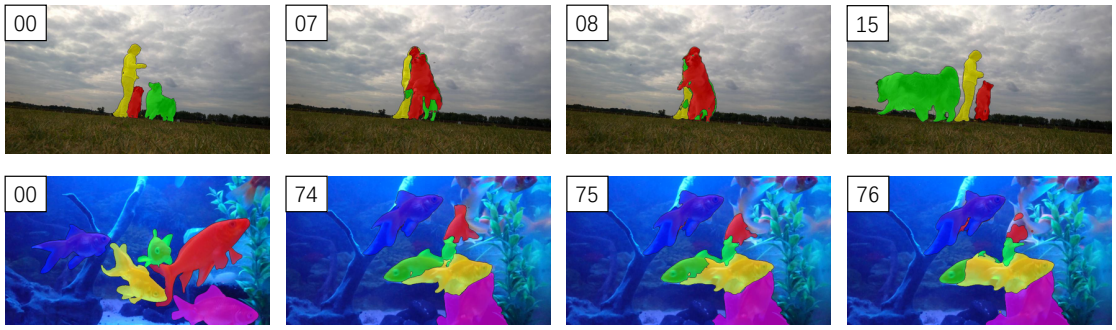


Figure 3.9: Limited Cases of Our Network

will be confused, because the encoder for the current frame needs to encode both image and previous predicted mask information, where the background is not masked out. However, for the first and the previous frames, the background is masked out, and we only use the foreground pixels of the frames.

3.3.5 Limitation Discussion

Some failure cases from our model are shown in Fig. 3.9. When foreground objects are overlapped, our model tends to produce incorrect segmentation for those occluded objects, especially when the overlapped objects are with the same category. Nevertheless, if the foreground objects are well separated afterwards, our model can adjust to the correct tracking and segmentation status due to the use of the first frame information, like in row 1 of Fig.7. This example shows that our method can catch back to the target object after occlusion. However, when there is occlusion for multi-objects, especially when the targets are in the same category, our method will be confused and lose the target (like in the second row of Fig. 7). To overcome this limitation, we consider that

we can generate some prototypes to represent each object and push away their feature distances to make the network be sensitive to different object in the future.

3.4 Conclusions

In this chapter, we have proposed a new video object segmentation network NPMCA-net, which combines a non-local pixel-matching module and a channel attention module in series connection. Our network achieves the state-of-the-art performance on both DAVIS-2017 and DAVIS-2016 validation set. Additionally, our NPMCA-net has a good generalization ability. Moreover, our network does not need any post-processing, so as to keep a good balance between accuracy and efficiency. In the future,we consider that we can generate some prototypes to represent each object and push away their feature distances to make the network be sensitive to different object.

Chapter 4

Structure Consistent Weakly Supervised Salient Object Detection

Salient object detection can reveal human attention when looking at a picture. This task can help offer foreground information to filter noise from the background. The saliency prediction can give a target hint for video object segmentation when there are no reference target masks. Therefore, in this chapter, the task of salient object detection is studied. Although excellent performance has been achieved recently, it needs significant pixel-level annotations. To save annotation cost, we focus on weakly supervised salient object detection which is supervised by scribbles. The intrinsic characteristic of an object, which is the consistency of RGB information and position information, is considered to help the network learn integral salient object structures. Our method can complement the limited information provided by scribbles.

4.1 Motivation

Salient object detection (SOD) aims to detect the most attractive regions in an image according to the human perception. It can be further applied in different computer vision tasks, such as image-sentence matching [41], image segmentation [96] and image retrieval [107]. In the last decade, deep learning based salient object detection algorithms [13, 68, 91] have become popular due to their superior performance. These methods usually design different modules to help their networks learn better feature representations for saliency prediction. However, they are highly dependent on pixel-wise saliency labels, which are time-consuming and costly with manual annotations.

In recent years, sparse labeling methods have attracted much attention. Many weakly su-

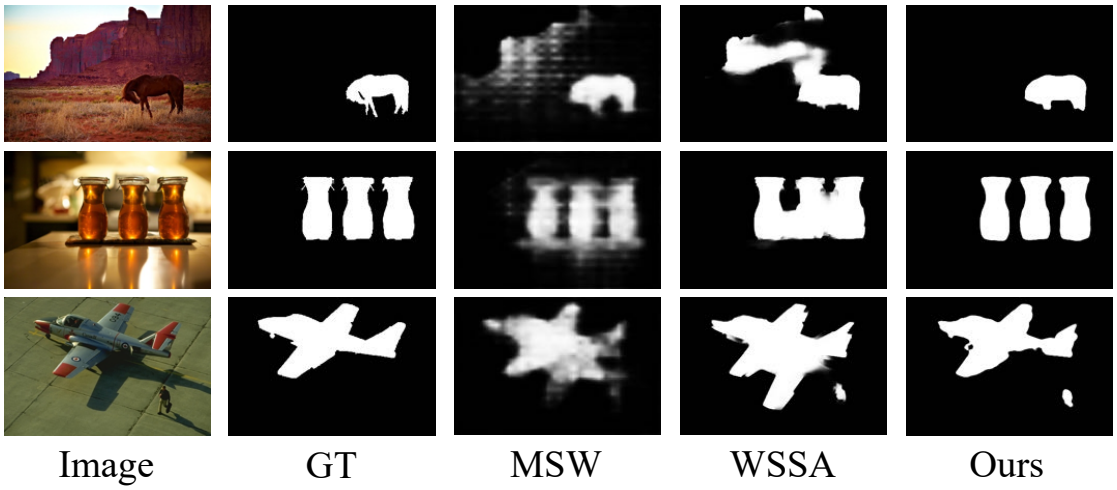


Figure 4.1: Our predicted saliency maps are compared with that of other weakly supervised methods. From left to right: Input image; Ground-truth; MSW [130]; WSSA [134]; Ours.

pervised salient object detection methods have been proposed to improve label efficiency while maintaining model performance. Image level labels are utilized in some methods [109, 57] to learn salient object detection. However, these works usually use image-level tags for saliency localization and then further train their models with predicted saliency maps through multiple-stage learning. Besides, some other works [80, 133] train their models with noisy pseudo labels from handcrafted methods and/or predicted maps by other weakly supervised SOD models, where pre-processing steps are used to clean noisy labels. All above mentioned works need complex training steps to obtain final saliency maps.

Additionally, scribble annotations are proposed recently due to their flexibility to label winding objects and low-cost compared to annotating per-pixel saliency masks. However, scribble annotations cannot cover the whole object region or directly provide object structure. Therefore, edge detection is used in the framework [134] to obtain object boundaries, and the SOD model is trained with predicted edge maps from other trained edge detection models. However, this step introduces extra data information into the SOD training process to recover integral object structure. The training process of [134] is also complex, as they design a scribble boosting scheme to iteratively train their model using initial saliency predictions to obtain higher quality saliency maps.

In this thesis, we aim to tackle the aforementioned issues in existing weakly supervised SOD methods. Specifically, we aim to design a high performance SOD method with scribble annotations

via one-stage end-to-end training, where no pre/post-processing steps nor extra supervision (e.g., edge maps) will be used. To mitigate the issue of poor boundary localization caused by scribble annotations and partial cross-entropy loss [134], we design a local saliency coherence loss to provide supervision for unlabeled points, based on the idea that points with similar features and/or close positions should have similar saliency values. By doing this, we take advantage of intrinsic properties of an image instead of extra edge or other assisting information to help our model learn better object structure and predict integral salient regions.

Besides, we find that weakly supervised SOD models fail to predict consistent saliency maps with different scales of the same image as input. To handle this problem, we propose a saliency structure consistency loss, which could be viewed as a regularization technique to enhance the model generalization ability.

Additionally, global context information can infer the relationship of different salient regions and help network predict better results [13]. High-level features can provide better semantic information and low-level features can capture rich spatial information [35]. In the decoder layers, we design an aggregation module called AGGM to integrate all information for better feature representations using the attention mechanism [93].

With our specially designed loss functions and network structures, our model can predict saliency maps close to human perception. Some obtained saliency maps are illustrated in Fig. 4.1. Our method predicts smoother and integral saliency objects even for the challenging cases with background disturbance, object shadow, and multiple objects. In general, our main contributions can be summarized as:

- A local saliency coherence loss is proposed for scribble supervised saliency object detection, which helps our network to learn integral salient object structures without any extra assisting data or complex training process.
- A self-consistent mechanism is introduced to ensure that consistent saliency masks will be predicted with different scales of the same image as input. It is an effective regularization to enhance the model generalization ability.
- An aggregation module named AGGM is designed in the encoder-decoder framework for weakly supervised SOD, which effectively aggregates global context information as well as high-level and low-level features.
- Comprehensive experiments show that our approach achieves a new state-of-the-art performance compared with other scribble supervised SOD algorithms on six widely-used

benchmarks, with an average gain of 4.60% for F-measure, 2.05% for E-measure and 1.88% for MAE over the previous best performing method.

4.2 Methodology

4.2.1 Overview

Firstly, the training dataset is defined as $U = \{x_n, y_n\}_{n=1}^{N}$, where x_n is the input image, y_n is the corresponding label, and N is the total number of training images. Note that, in our task, the label y_n is annotated as scribble.

The whole network and learning framework are shown in Fig. 4.2. The network contains an encoder and a decoder, and the designed aggregation module AGGM is applied in each layer of the decoder. In this way, the three kinds of information can be better propagated into next layers. Sigmoid function is applied on the output of the decoder to normalize the output saliency values to $[0, 1]$. Thus, our network receives the input image and outputs the corresponding saliency map directly. During training, the proposed local saliency coherence loss and saliency structure consistency loss are applied with the partial cross entropy loss as the dominant loss to supervise the final predicted saliency map. Meanwhile, to facilitate training, we enforce an auxiliary loss, which includes the local saliency coherence loss and the partial cross entropy loss on each sub-stage to supervise the intermediate low-resolution saliency maps. Note that the intermediate low-resolution saliency maps are upsampled to the corresponding scale of x_n to supervise the intermediate low-resolution saliency map. In the following sections, we will discuss the AGGM and each loss in details.

4.2.2 Aggregation Module

Our network follows an encoder-decoder framework. The encoder layers learn different features of the salient regions and further propagate them to the decoder. Since some detailed features might be diluted, each decoding layer uses output of preceding layer, the features of corresponding encoding layer and the global context information, which is the final output of the encoder, as inputs to predict salient regions. However, we argue that each input should be allocated to different weights for each decoding layer as the influence of each input are different for different images. To learn the importance of each input feature by self-learning, our aggregation module is designed as in Fig. 4.3. 3×3 convolution layers and global average pooling layers are applied to learn the importance of each input feature. Once the weights are obtained, normalization is applied, which

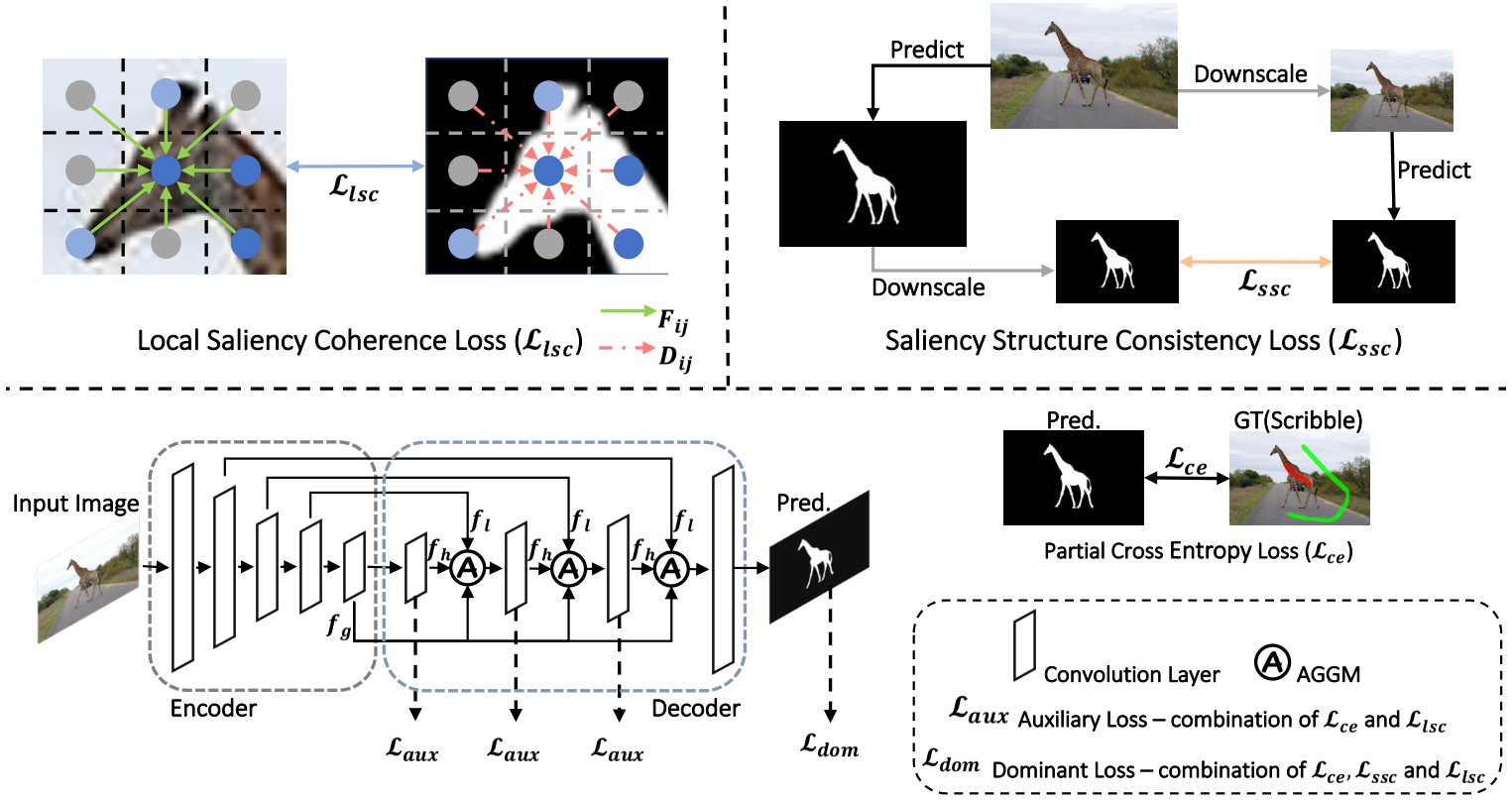


Figure 4.2: The framework of our network and learning procedure. Specifically, f_l, f_h, f_g denote to the low-level, high-level features and global context information, respectively. The AGGM is applied in the decoder to integrate multi-level features. The proposed local saliency coherence loss and saliency structure consistency loss are applied with partial cross entropy loss to optimize the network as a dominant loss. To facilitate optimization, our local saliency coherence loss is applied with partial cross entropy loss as auxiliary losses to further supervise intermediate low-resolution saliency maps.

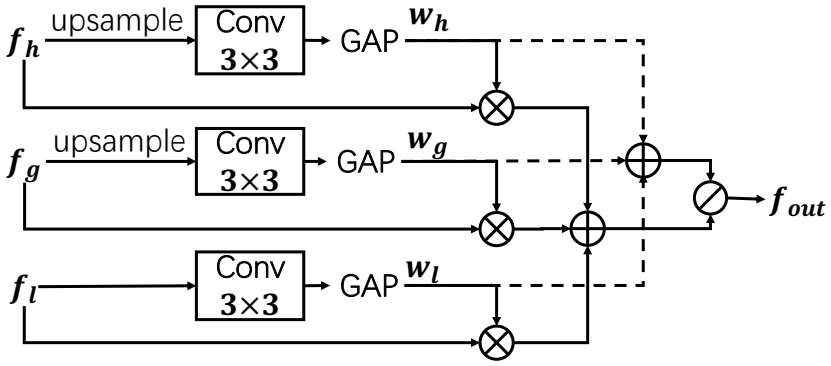


Figure 4.3: Framework of AGGM, where 'GAP' denotes to global average pooling, '×' is multiplication, '+' is addition and '/' is division.

can be written as:

$$f_{out} = \frac{w_h f_h + w_g f_g + w_l f_l}{w_h + w_g + w_l}, \tag{4.1}$$

where f_h is the high-level feature, f_g is the global context information, and f_l is the low-level feature. Then w_h, w_g and w_l are the obtained corresponding weights. Note that, the size of f_h and f_g are first resized into the same size of f_l. With Eq. (4.1), the network can aggregate multi-level features and with our AGGM, the importance of each feature can be learned by self-learning.

4.2.3 Local Saliency Coherence Loss

For scribble annotations, there are a great number of unlabeled pixels. With only the given scribble labels, it is hard to learn rich information of salient regions. In addition, there is no category information in the SOD task, making it more difficult to learn object structures. Therefore, the network needs other supervision to get better saliency maps with clear object boundaries. In this case, we design a local saliency coherence loss to help network predict smooth saliency maps with the scribble annotations. We consider that for pixel i and pixel j of the same input image, if they are with similar features or close positions, they tend to have similar saliency scores. On the other hand, if two points do not share similar features or they are distant from each other, they are more likely to have different saliency scores. We first define the saliency difference between two different pixels i and j as follows:

$$D(i, j) = |S_i - S_j|, \tag{4.2}$$

where S_i and S_j are the predicted saliency scores of pixels i and j, respectively. We use $L1$ distance to directly compute the discrepancy of the saliency scores.

Instead of computing the similarities between any two pixels in an image, which introduces too much background noise and takes too much GPU memory, we compute the discrepancy of a reference point with its adjacent points in a $k \times k$ kernel size. However, if we directly compute the loss using Eq. (4.2), the network fails to distinguish the salient region with background, especially for the pixels close to the boundaries. For pixels around object boundaries, their saliency scores are not always similar with their adjacent pixels. Therefore, we set a similarity energy between two pixels i and j, which is defined based on Gaussian kernel bandwidth filter [81], to draw close saliency scores for pixels with similar features and/or with small distance. Then, the final local saliency coherence loss is designed as:

$$\mathcal{L}_{lsc} = \sum_{i} \sum_{j \in K_i} F(i,j) D(i,j), \tag{4.3}$$

where K_i is the region covered by a $k \times k$ kernel around pixel i, and $F(i,j)$ denotes to the following filter:

$$F(i,j) = \frac{1}{w} \exp\left(-\frac{\|P(i) - P(j)\|^2}{2\sigma_P^2} - \frac{\|I(i) - I(j)\|^2}{2\sigma_I^2}\right), \tag{4.4}$$

where $1/w$ is the normalized weight, $P(\cdot)$ and $I(\cdot)$ are the position and RGB color of a pixel, respectively. σ_P and σ_I are hyper parameters for the scale of Gaussian kernels. $\|\cdot\|^2$ is an $L2$ operation.

The local saliency coherence loss $\mathcal{L}_{lsc}$ enforces similar pixels in the kernel to share consistent saliency scores, which further propagates labeled points to the whole image during training. With the partial cross entropy loss to supervise labeled points, the network can acquire enlarged salient regions with limited labels without any extra information.

4.2.4 Self-Consistent Mechanism

For a good salient object detection model, saliency maps predicted with different scales of the same image should be consistent. We define a salient object detection function as $f_\theta(\cdot)$ with parameter θ, and a transformation as $T(\cdot)$. Then, for an ideal $f_\theta(x)$, it should satisfy this equation:

$$f_\theta(T(x)) = T(f_\theta(x)). \tag{4.5}$$

However, we find that it is difficult for weakly supervised SOD networks to predict consistent

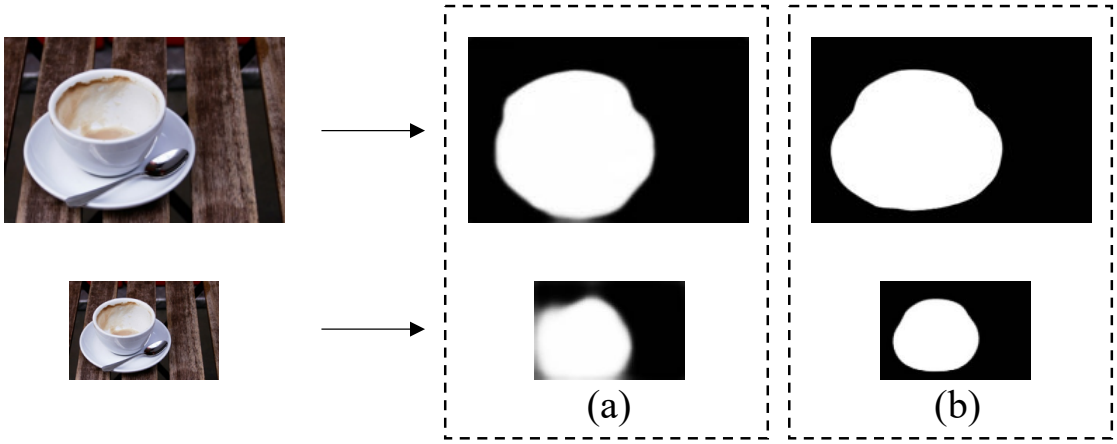


Figure 4.4: Comparison of predicted saliency maps for an input image with different scales: (a) without self-consistent mechanism; (b) with self-consistent mechanism.

saliency maps with different input scales, as shown in Fig. 4.4(a). Therefore, by considering Eq. (4.5) as a regularization, we design a structure consistency loss on predicted saliency maps from different input scales, which is defined as follows:

$$\begin{aligned}\mathcal{L}_{ssc} = &\frac{1}{M}\sum_{u,v}\alpha\frac{1-SSIM(S^{\Downarrow}_{u,v}, S^{\downarrow}_{u,v})}{2} \\ &+ (1-\alpha)|S^{\Downarrow}_{u,v} - S^{\downarrow}_{u,v}|,\end{aligned} \tag{4.6}$$

where $S^{\downarrow}$ is down-scaled predicted saliency map of a normal input image, $S^{\Downarrow}$ is the predicted saliency map of the same image with down-scaled size, and M is the number of pixels. SSIM denotes to the single scale SSIM [117, 30] and $\alpha = 0.85$ [30]. With Eq. (4.6), the network can learn more information on object structure and enhance generalization ability for different input scales. As shown in Fig. 4.4(b), with our self-consistent mechanism, the network can adapt to different scales and predict saliency maps with better object structure.

4.2.5 Objective Function

As shown in Fig. 4.2, our final loss function is the combination of a dominant loss and auxiliary losses following GCPANet [13]. Specifically, a 3×3 convolution layer is conducted to squeeze the channel to 1 at each stage of the decoder to compute the saliency scores. Then, our proposed

losses are used to cooperate with the partial cross entropy loss, which can be written as:

$$\mathcal{L}_{ce} = \sum_{i \in \tilde{\mathcal{Y}}} -y_i log\hat{y}_i - (1 - y_i) log(1 - \hat{y}_i), \tag{4.7}$$

where y denotes for ground-truth, $\hat{y}$ is the predicted values and $\tilde{\mathcal{Y}}$ is the set of labeled pixels via scribble annotations.

The auxiliary loss $\mathcal{L}_{aux}$ and dominant loss $\mathcal{L}_{dom}$ can be written as:

$$\mathcal{L}^q_{aux} = \mathcal{L}_{ce} + \beta\mathcal{L}_{lsc} \quad q \in \{1, 2, 3\} \ , \tag{4.8}$$

$$\mathcal{L}_{dom} = \mathcal{L}_{ce} + \mathcal{L}_{ssc} + \beta\mathcal{L}_{lsc}, \tag{4.9}$$

where the hyper-parameter β shares the same value in Eq. (4.8) and Eq. (4.9), and q in Eq. (4.8) stands for the index of a decoder layer.

Finally, the overall objective function of our network is:

$$\mathcal{L}_{total} = \mathcal{L}_{dom} + \sum_{q=1}^{3} \lambda_q \mathcal{L}^q_{aux}, \tag{4.10}$$

where λ_q is to balance the auxiliary loss of each stage, and we take the same value as in GCPANet [13].

4.3 Experiments

4.3.1 Implementation Details and Setup

Implementation Details. We use GCPANet [13] with backbone of ResNet-50 [31] pretrained on ImageNet [19] as baseline. The partial cross entropy loss (Eq. (4.7)) is computed for background and foreground individually. w, σ_P, and σ_I in Eq. (4.4) are set to 1, 6 and 0.1, respectively. β in Eq. (4.8) and Eq. (4.9) is set to 0.3. The model is optimized by SGD with batch size of 16, momentum of 0.9 and weight decay of 5×10^{-4}. Additionally, we use triangular warm-up and decay strategies with the maximum learning rate of 0.01 and the minimum learning rate of 1×10^{-5} to train the network with 40 epochs. During training, each image is resized to 320×320 with random horizontal flipping and random cropping. In the inference stage, input images are simply resized to 320×320 and then fed into the network to predict saliency maps without any post-processing. All experiments are

Table 4.1: Comparison with other state-of-the-art approaches on 3 benchmarks: ECSSD, DUT-OMRON, and PASCAL-S. ↑ means that larger is better and ↓ denotes that smaller is better. The best performance on each dataset is highlighted in boldface under different cases of supervision. 'Sup.' denotes for supervision information. 'F' means fully supervised. 'I' means image-level supervised. 'S' means scribble-level supervised. 'M' means multi-source supervised and 'Un' is for unsupervised. '†' means two-round training.

		ECSSD			DUT-OMRON			PASCAL-S		
Methods	Sup.	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$
DGRL [110]	F	0.9027	0.9371	0.043	0.7264	0.8446	0.0632	0.8289	0.8353	0.1150
PiCANet [69]	F	0.8864	0.9128	0.0464	0.7173	0.8407	0.0653	0.7979	0.8330	0.0750
PAGR[141]	F	0.8718	0.8869	0.0644	0.6754	0.7717	0.0709	0.7656	0.7545	0.1516
MLMSNet [121]	F	0.8856	0.9218	0.0479	0.7095	0.8306	0.0636	0.8129	0.8219	0.1193
CPD [122]	F	0.917	0.925	0.037	0.747	0.866	0.056	0.824	0.849	0.072
AFNet [26]	F	0.9008	0.9294	0.0450	0.7425	0.8456	0.0574	0.8241	0.8269	0.1155
PFAN [145]	F	0.8592	0.8636	0.0467	0.7009	0.7990	0.0615	0.7544	0.7464	0.1372
BASNet [91]	F	0.880	0.916	0.037	0.756	0.869	0.056	0.775	0.832	0.076
GCPANet [13]	F	0.9184	0.927	0.035	0.7479	0.839	0.056	0.8335	0.861	**0.061**
MINet [83]	F	**0.924**	**0.953**	**0.033**	**0.756**	**0.873**	**0.055**	**0.842**	**0.899**	0.064
SVF [133]	Un	0.7823	0.8354	0.0955	0.6120	0.7633	0.1076	0.7351	0.7459	0.1669
MNL [135]	Un	0.8098	0.8357	0.0902	0.5966	0.7124	0.1028	0.7476	0.7408	0.1576
ASMO [57]	I	0.7621	0.7921	0.0681	0.6408	0.7605	0.0999	0.6532	0.6474	0.2055
WSS [109]	I	0.7672	0.7693	0.1081	0.5895	0.7292	0.1102	0.6975	0.6904	0.1843
MSW [130]	M	0.7606	0.7876	0.0980	0.5970	0.7283	0.1087	0.6850	0.6932	0.1780
WSSA [134]	S	0.845	0.898	0.068	0.679	0.823	0.074	0.772	0.791	0.145
WSSA† [134]	S	0.8650	0.9077	0.0610	0.7015	0.8345	0.0684	0.7884	0.7975	0.1399
Ours	S	**0.8995**	**0.9079**	**0.0489**	**0.7580**	**0.8624**	**0.0602**	**0.8230**	**0.8465**	**0.0779**

Table 4.2: Comparison with other state-of-the-art approaches on 3 benchmarks: HKU-IS, THUR, and DUT-TEST. ↑ means that larger is better and ↓ denotes that smaller is better. The best performance on each dataset is highlighted in boldface under different cases of supervision. 'Sup.' denotes for supervision information. 'F' means fully supervised. 'I' means image-level supervised. 'S' means scribble-level supervised. 'M' means multi-source supervised and 'Un' is for unsupervised. '†' means two-round training.

		HKU-IS			THUR			DUTS-TEST		
Methods	Sup.	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$
DGRL [110]	F	0.8844	0.9388	0.0374	0.7271	0.8378	0.0774	0.7989	0.8873	0.0512
PiCANet [69]	F	0.8704	0.9355	0.0433	-	-	-	0.7589	0.8616	0.0506
PAGR[141]	F	0.8638	0.8979	0.0475	**0.7395**	**0.8417**	**0.0704**	0.7781	0.8422	0.0555
MLMSNet [121]	F	0.8780	0.9304	0.0387	0.7177	0.8288	0.0794	0.7917	0.8829	0.0490
CPD [122]	F	0.891	0.944	0.034	-	-	-	0.805	0.886	0.043
AFNet [26]	F	0.8877	0.9344	0.0358	0.7327	0.8398	0.0724	0.8123	0.8928	0.0457
PFAN [145]	F	0.8717	0.8982	0.0424	0.6833	0.8038	0.0939	0.7648	0.8301	0.0609
BASNet [91]	F	0.895	0.946	0.032	0.7366	0.8408	0.0734	0.791	0.884	0.048
GCPANet [13]	F	0.8984	0.920	0.031	-	-	-	0.8170	0.891	0.038
MINet [83]	F	**0.908**	**0.961**	**0.028**	-	-	-	**0.828**	**0.917**	**0.037**
SVF [133]	Un	0.7825	0.8549	0.0753	0.6269	0.7699	0.1071	0.6223	0.7629	0.1069
MNL [135]	Un	0.8196	0.8579	0.0650	0.6911	0.8073	0.0860	0.7249	0.8525	0.0749
ASMO [57]	I	0.7625	0.7995	0.0885	-	-	-	0.5687	0.6900	0.1156
WSS [109]	I	0.7734	0.8185	0.0787	0.6526	0.7747	0.0966	0.6330	0.8061	0.1000
MSW [130]	M	0.7337	0.7862	0.0843	-	-	-	0.6479	0.7419	0.0912
WSSA [134]	S	0.835	0.911	0.055	0.696	0.824	0.085	0.728	0.857	0.068
WSSA† [134]	S	0.8576	0.9232	0.0470	0.7181	0.8367	0.0772	0.7467	0.8649	0.0622
Ours	S	**0.8962**	**0.9376**	**0.0375**	**0.7545**	**0.8430**	**0.0693**	**0.8226**	**0.8904**	**0.0487**

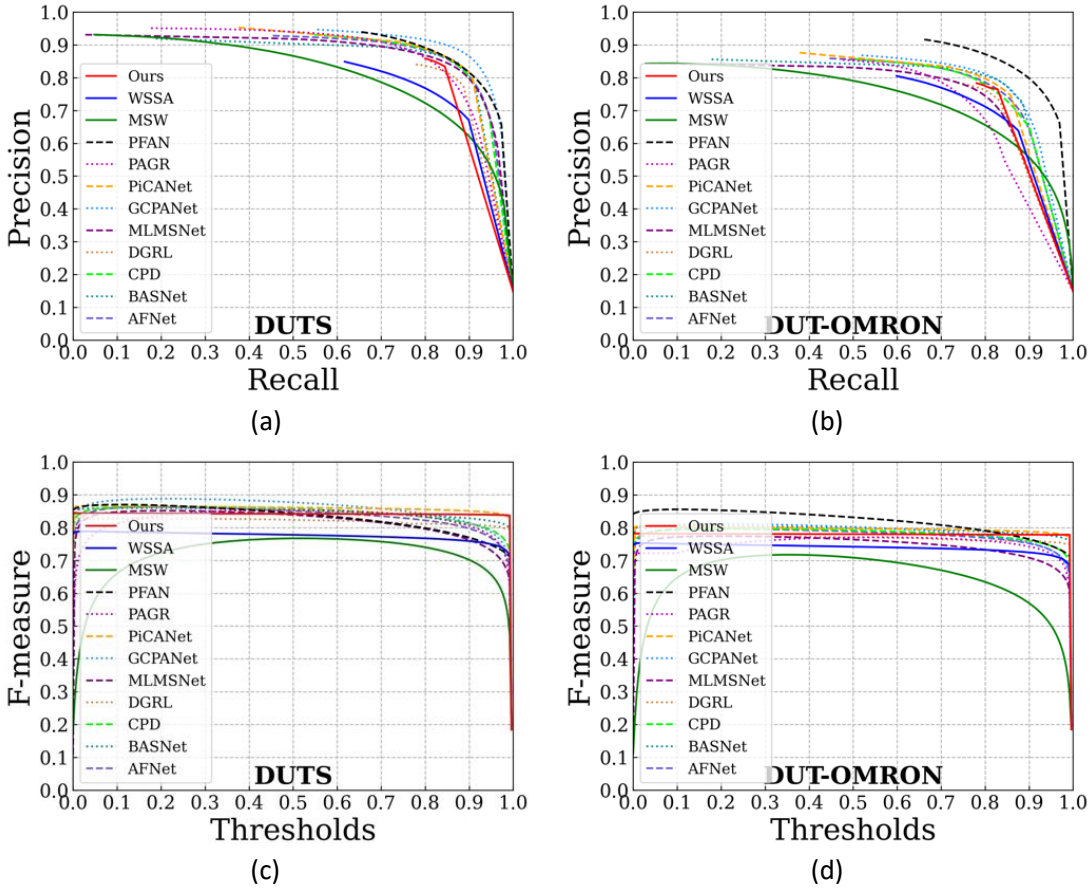


Figure 4.5: PR-curves and F-measure curves. (a) and (b) are precision curves for DUT-TEST and DUT-OMORON; (c) and (d) are F-measure curves for DUT-TEST and DUT-OMOTON.

run on NVIDIA GeForce RTX 2080 Ti. The total time for training is around 8 hours and inference speed is around 40 fps.

Datasets. We train our network on scribble annotated dataset S-DUTS [134] and evaluate our model on six widely-used salient object detection benchmarks: (1) ECSSD [124]; (2) DUT-OMRON [125]; (3) PASCAL-S [63]; (4) HKU-IS [58]; (5) THUR [14]; (6) DUTS-TEST [109].

Baseline Methods and Evaluation Metrics. Our model is compared with 6 state-of-the-art weakly supervised or unsupervised SOD methods and 10 fully supervised SOD methods as baselines. We take three widely-used evaluation metrics for fair comparison: mean F-measure (F_β), mean E-measure (E_ξ) [22], and Mean Absolute Error (MAE) [18]. We also list PR curves and F-measure curves in Fig. 4.5.

4.3.2 Comparison with State-of-the-arts

Quantitative Comparison. In Table 4.1 and Table 4.2, we compare our approach with other state-of-the-art approaches. Our method achieves a new state-of-the-art performance among weakly supervised or unsupervised approaches under all the evaluation metrics. Our one-round training method obtains an average gain of 4.60% for F_β, 2.05% for E_ξ, and 1.88% for MAE, compared with the previous best two-round training method WSSA [134]. Besides, our approach is comparable or even superior to some fully supervised methods, like PiCANet [69], PAGR[141] and MLMSNet [121]. The PR curves and F-measure curves shown in Fig. 4.5 can reflect the generalization.

Qualitative Evaluation. We demonstrate some samples of our predicted saliency maps from the ECSSD dataset [124] in Fig. 4.6. It shows that our predicted saliency maps are more complete and precise compared with previous state-of-the-arts (MSW and WSSA). Moreover, our approach is more general to different object classes and more robust to the disturbance of foreground-background (see rows 3 & 4 in Fig. 4.6). In some cases, our approach even performs better than fully supervised methods, such as CPD, BASNet and GCPANet (see rows 3 & 5 in Fig. 4.6).

4.3.3 Ablation Study

We conduct different ablation studies to analyze the proposed method, including the loss functions and our aggregation module. The experiments are evaluated on the DUTS-TEST dataset [109]. We conduct the experiments by combining different parts of our method. As shown in Table 5.2, our method obtains the best performance using all the components, which illustrates that all the loss functions and the AGGM are necessary to realize the one-step training. We use GCPANet [13]

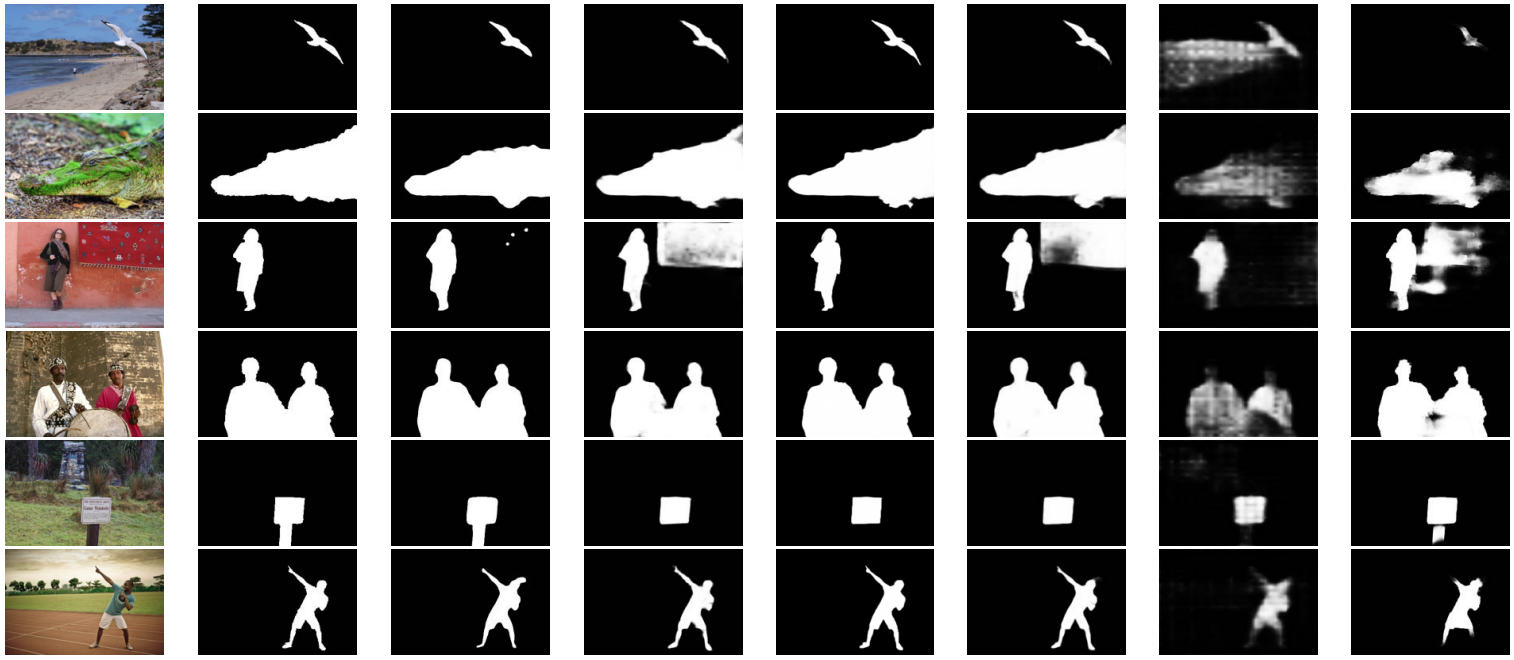


Figure 4.6: Qualitative comparisons of saliency maps predicted by our method and other state-of-the-art methods. Obviously, the maps predicted by ours are closer to the ground-truth compared with other weakly supervised approaches (MSW [130] & WSSA [134]), and some of our results even cover more details than that of fully supervised approaches (CPD [122], BASNet [91], GCPANet [13]) as in row 5.

Table 4.3: Ablation study for our losses and AGGM on DUTS-TEST dataset. 'Base.' denotes for baseline and 'A.' denotes for AGGM. Our overall method obtains the best results.

	Base.	A.	$\mathcal{L}_{ssc}$	$\mathcal{L}_{lsc}$	$F_\beta \uparrow$	$E_\xi \uparrow$	$MAE \downarrow$
1	✓				0.707	0.843	0.064
2	✓	✓			0.706	0.845	0.064
3	✓	✓	✓		0.758	0.873	0.059
4	✓	✓	✓	✓	**0.823**	**0.890**	**0.049**

Table 4.4: Ablation study for our proposed AGGM on DUT-OMRON and DUTS-TEST datasets. It can be seen that our AGGM is compatible to our loss functions.

	DUT-OMRON			DUTS-TEST		
	$F_\beta\uparrow$	$E_\xi\uparrow$	$M.\downarrow$	$F_\beta\uparrow$	$E_\xi\uparrow$	$MAE\downarrow$
w/o	0.730	0.845	0.069	0.800	0.877	0.053
w	**0.758**	**0.862**	**0.060**	**0.823**	**0.890**	**0.049**

as our baseline. If the network is directly trained with partial cross entropy loss, the results are relatively low, as listed in 1 of Table 5.2. This phenomenon shows that the partial cross entropy loss is insufficient for sparse labels. Moreover, comparing our final results with the baseline, we obtain gains of 11.61% for F_β, 4.73% for E_ξ and 1.54% for MAE, respectively.

Impact of AGGM. In Table 4.4, we evaluate the influence of our aggregation module AGGM when all the loss functions are enabled. It is interesting to see that using AGGM can obtain an average gain of 2.56% for F_β, 1.52% for E_ξ and 0.63% for MAE on DUT-OMRON and DUTS-TEST. However, when training without our proposed losses, as listed in 1 and 2 of Table 5.2, our AGGM contributes little compared with the baseline mode. This phenomenon shows that our AGGM is complementary with the proposed loss functions for sparse labels.

Impact of Saliency Structure Consistency Loss. We conduct this ablation study by adding saliency structure consistency loss ($\mathcal{L}_{ssc}$) to the baseline (AGGM is enabled). The results are shown

Table 4.5: Ablation study for SSIM in the saliency structure consistency loss on DUT-OMRON and DUTS-TEST. It can be observed that the SSIM in the saliency structrue consistency loss is can help learn better structure information.

	DUT-OMRON			DUTS-TEST		
	$F_\beta\uparrow$	$E_\xi\uparrow$	$M.\downarrow$	$F_\beta\uparrow$	$E_\xi\uparrow$	$MAE\downarrow$
w/o	0.670	0.828	0.077	0.735	0.863	0.063
w	**0.708**	**0.841**	**0.072**	**0.758**	**0.873**	**0.059**

in 3 of Table 5.2. Compared with only using partial cross entropy loss, using $\mathcal{L}_{ssc}$ achieves the improvement of 5.29% for F_β, 2.81% for E_ξ and 0.47% for *MAE*. Therefore, our $\mathcal{L}_{ssc}$ can regularize partial cross entropy loss and enhance the model generalization ability. Further, we evaluate the impact of SSIM in Eq. (4.6) which is shown in Table 4.5. We train the network using $\mathcal{L}_{ssc}$ with and without SSIM separately. The scores of the three evaluation metrics with SSIM are all higher than those without SSIM, which indicates that Eq. (4.6) needs SSIM to make better prediction.

Impact of Local Saliency Coherence Loss. We list the evaluation of the local saliency coherence loss ($\mathcal{L}_{lsc}$) in Table 5.2. With $\mathcal{L}_{lsc}$, the network can obtain the best results. Specifically, using $\mathcal{L}_{lsc}$ improves F_β from 0.7584 to 0.8226, E_ξ from 0.8732 to 0.8904 and *MAE* from 0.0589 to 0.0487. It is the new state-of-the-art performance as reported in Table 4.1 and Table 4.2. Since there is no extra supervision information like edges, such performance demonstrates that our $\mathcal{L}_{lsc}$ can learn integral salient object structures.

4.3.4 Limitation Discussion

We list some failures of our method in Fig. 4.7. It is difficult for our method to detect the salient object with noisy background, especially when the background is similar to the foreground. This phenomenon may be caused that although our method introduces RGB information to help learn the more integral appearance of the target object, sometimes the background and foreground may share similar RGB information. Our method may be misled to wrong predictions when there is noisy background. To overcome this problem, the extracted features can be involved in our local saliency coherence loss to help provide more details to guide the predictions because the extracted features have more semantic information than RGB features.

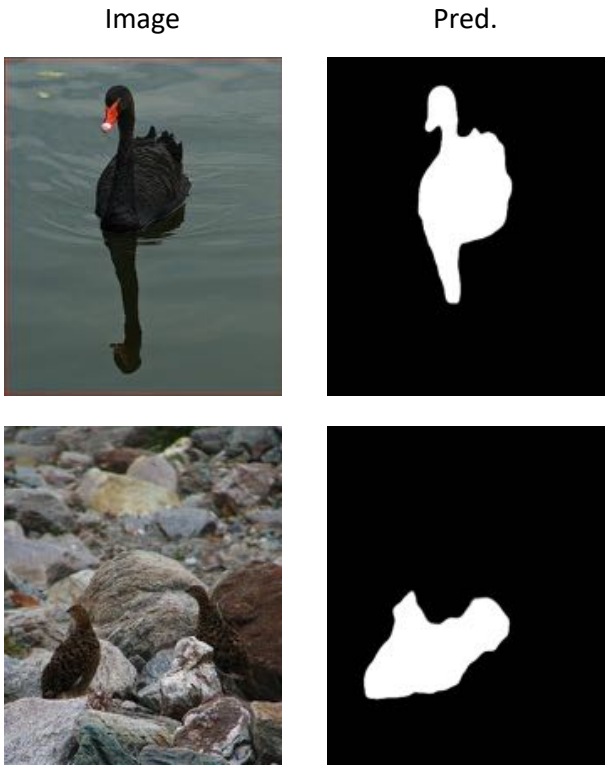


Figure 4.7: Limited Cases of Our Network. 'Pred.' means predictions.

4.4 Conclusions

In this chapter, we have explored one-round training for salient object detection via scribble annotations. We propose a local saliency coherence loss to supervise unlabeled points. Besides, we deploy a self-consistent mechanism via saliency structure consistency loss to improve the network generalization ability. Moreover, we have designed an aggregation module to better integrate multiple levels of features, so as to predict better saliency maps for weakly supervised SOD. Experiments show that our approach outperforms previous state-of-the-arts under different evaluation metrics on 6 datasets with a significant margin. Furthermore, our proposed loss functions utilize intrinsic properties of input images to supervise unlabeled points, such that no extra supervision is introduced.

Chapter 5

Comprehensive Feature Mining for Co-salient Object Detection

In this chapter, the task of co-salient object detection is studied. We find that the attention mechanism can also be applied in this task to search for the pixel of each image, with the highest probability belonging to the co-salient objects. Then, the detected pixel can be regarded as an indicator to mine similar pixels for the co-salient object. In such a case, comprehensive co-salient features can be mined for our framework.

5.1 Motivation

Co-salient object detection (CoSOD) aims to detect the common salient objects among a group of input images. Unlike salient object detection (SOD), which is to detect the most attractive objects by mimicking human eyes [13, 55, 128, 98, 84, 60, 89, 71], CoSOD focuses on detecting salient yet co-existed objects among all the input images. In this case, CoSOD faces two main challenges: 1) reduce the interference of noisy background in complex scenes; 2) mine integral co-salient objects with large appearance variations. Some works introduce extra SOD dataset to provide saliency guidance [140, 139] or predict saliency maps [46] in order to mask out the co-salient objects. However, these approaches highly depend on the extra dataset, leading to supererogatory human effort to provide annotations.

Recent approaches [46, 139, 25, 136] try to use attention mechanism [103] to strengthen co-salient features or build feature consistency to formulate the shared attributes of co-salient objects for integral predictions. However, there are two main drawbacks when directly applying attention

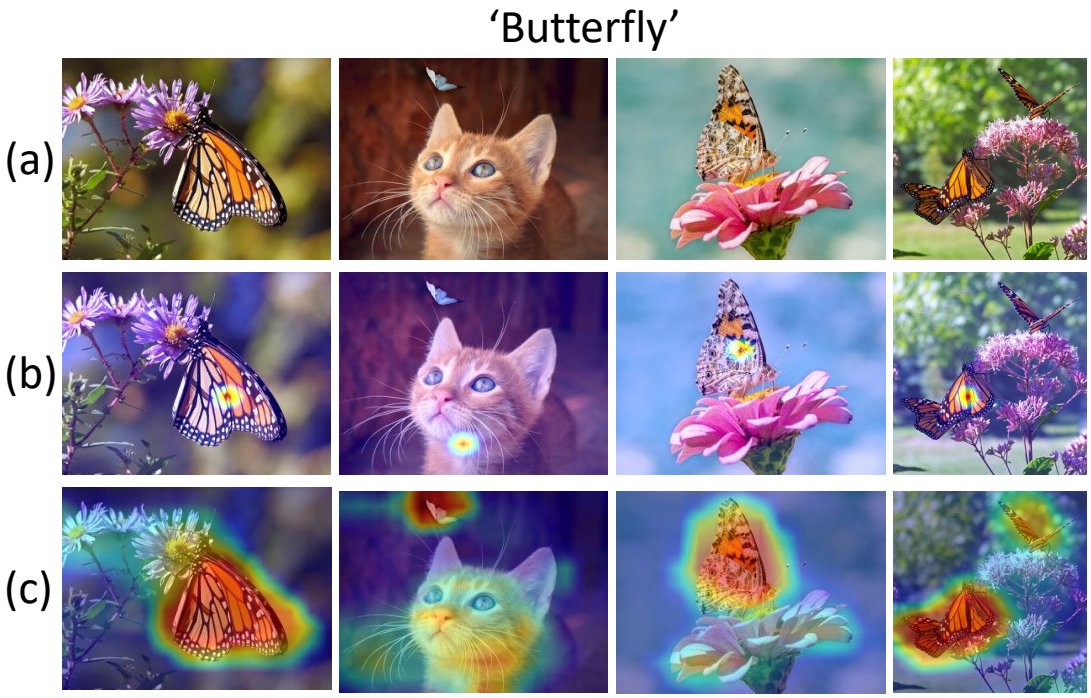


Figure 5.1: Visualization of response maps. (a) Inputs; (b) Response maps generated by the previous approach [25]; (c) Ours. It can be seen that ours can cover more co-salient objects.

mechanism for this task. On the one hand, the response maps reflecting the shared attributes, obtained in the attention mechanism, can only cover limited pixels belonging to co-salient objects, as shown in Fig. 5.1.(b). In this case, it is difficult for the model to learn comprehensive shared attributes of co-salient objects. On the other hand, for complex scenes, the attention mechanism tends to focus on the wrong object regions, as shown in the second picture of Fig. 5.1.(b). Some methods such as GCoNet [25] propose a kind of group collaborative learning by collecting artificial negative group pairs. However, their pairs are grouped based on the auxiliary classification information, which requires major effort to group dissimilar negative category pairs as there is no clear definition of natural discrete object categories in real world [99].

To solve aforementioned issues, we design a novel **D**emocratic **C**o-salient-**F**eature-**M**ining framework (**DCFM**). Our DCFM can directly mine more comprehensive features and suppress the noisy background effectively without using extra SOD dataset or classification information. Specifically, in order to mine sufficient co-salient information, we first design a democratic prototype generation module (DPG), where democratic response maps are generated to capture more shared attributes. As shown in Fig. 5.1.(c), our response maps cover more regions of co-salient objects. Then, a prototype with comprehensive co-salient information can be generated according to the democratic response maps, which can further guide the model to predict the co-salient objects.

Next, in order to suppress noisy background information in our prototype and avoid introducing extra classification information, we propose a simple self-contrastive learning module (SCL) to form positive and negative pairs to filter noise. We argue that the prototype generated from original images should be consistent with that generated when the image background regions are erased, and should be different from that generated when the co-salient objects are erased. Thus, a self-contrastive loss among these prototypes is designed to suppress the influence of noisy background and help the model learn more discriminative features of co-salient objects.

Finally, to further strengthen the detected co-salient features from the above modules, we design a democratic feature enhancement module (DFE) based on the attention mechanism [103]. As mentioned before, the attention mechanism tends to focus on a limited number of correlated features, which fails to provide comprehensive information. Therefore, we readjust the attention values to generate a democratic attention map aggregating more correlated pixels for feature enhancement.

Generally, our main contributions can be summarized as:

- A democratic prototype generation module (DPG) is designed to build response maps covering sufficient co-salient regions, so as to generate a prototype containing comprehensive

shared attributes as guidance for co-saliency prediction.

- A self-contrastive learning module (SCL) is proposed to help our model reduce the influence of noisy background without relying on additional classification information, where both positive and negative samples are generated from the image itself.
- A democratic feature enhancement module (DFE) is designed to further strengthen the co-salient features by adjusting attention values to involve more related pixels.
- Extensive experiments show that our method performs better than state-of-the-art methods, especially on challenging real-world cases, such as the CoCA dataset, we obtain a gain of 2.0% for MAE, 5.4% for maximum F-measure, 2.3% for maximum E-measure, and 3.7% for S-measure under the same settings.

5.2 Methodology

5.2.1 Overview

The CoSOD dataset includes groups of images with labels. Each group is represented as $G = \{I, Y\}$, where $I = \{x_n\}_{n=1}^{N}$, $Y = \{y_n\}_{n=1}^{N}$, x_n is the input image, y_n is the corresponding label, N is the total number of images in group G, and all images contain related objects. The labels are unavailable during inference. The model needs to detect the co-existed salient objects in each image of the same group. In this work, we aim to design a model that can detect the co-salient objects by thoroughly exploring the shared attributes to mine comprehensive co-salient features, and suppress noisy background through self-contrastive learning without using classification information or extra SOD dataset.

The framework of our method and the learning procedure are demonstrated in Fig. 5.2. There are five main modules in our network, including a feature extractor, a democratic prototype generation module (DPG), a self-contrastive learning module (SCL), a democratic feature enhancement module (DFE), and a decoder. Note that the SCL is only applied for training and will be removed during inference. The overall process can be summarized as:

1. Firstly, the feature extractor encodes a group of relative images (N images) as initial features, which are then proceeded by the DPG to generate a comprehensive co-salient prototype.
2. Meanwhile, to avoid mining noisy information from the background in the prototype, our SCL is deployed for auxiliary training.

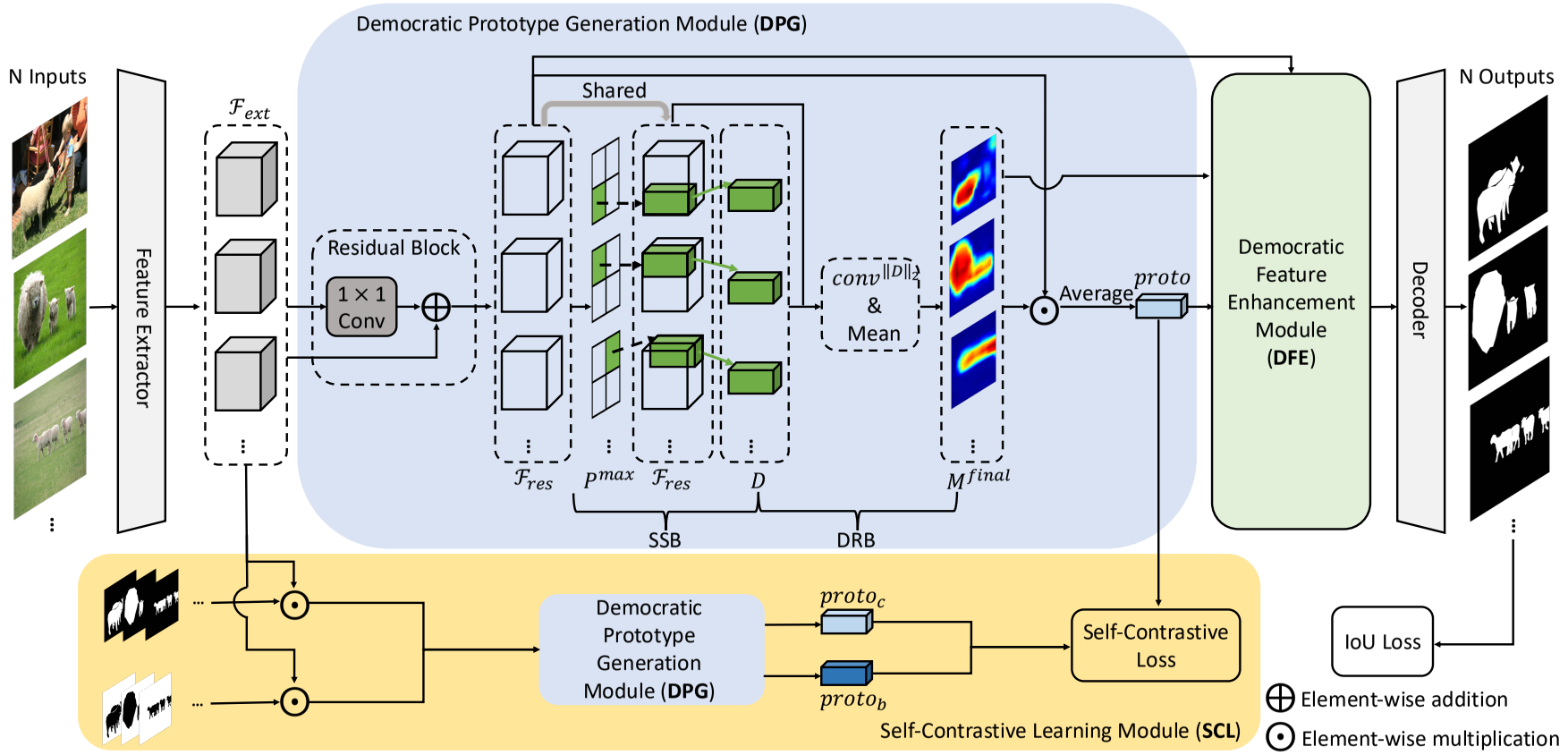


Figure 5.2: The framework of our network and the learning procedure. Specifically, the network contains five main parts, including a feature extractor, a democratic prototype generation module (DPG), a self-contrastive learning module (SCL), a democratic feature enhancement module (DFE), and a decoder. Note that the SCL is only used during training.

3. Then, the prototype is fused into the visual features, and the fused features are transmitted into the DFE to strengthen the features further.

4. Finally, the strengthened features are input into the decoder to predict the corresponding co-saliency maps.

In the following sections, the details about the democratic prototype generation module, the self-contrastive learning module, and the democratic feature enhancement module will be discussed, respectively.

5.2.2 Democratic Prototype Generation Module

Our democratic prototype generation module (DPG) mainly contains three parts in series, which are the residual block, the seed selection block (SSB), and the democratic response block (DRB).

After passing the feature extractor, we obtain the initial features $\mathcal{F}_{ext} \in \mathbb{R}^{N\times C\times H\times W}$ (C, H, W are the channel number, height, and width), which are processed by the residual block first to generate strengthened residual features $\mathcal{F}_{res}$:

$$\mathcal{F}_{res} = \mathcal{F}_{ext} + conv^{1\times1}(\mathcal{F}_{ext}), \tag{5.1}$$

where $conv^{1\times1}$ represents for the 1×1 convolution layer and $\mathcal{F}_{res} \in \mathbb{R}^{N\times C\times H\times W}$.

Then, the generated features $\mathcal{F}_{res}$ are passed into the SSB to select the most discriminative seeds for the co-salient objects in each input image. Next, the selected seeds are correlated with the residual feature maps to produce the response maps by the DRB. Finally, the response maps are multiplied with the residual features and averaged to generate the prototype, containing comprehensive co-salient feature information and guiding following prediction.

Seed Selection Block (SSB). The SSB is demonstrated in Fig. 5.3. This block is deployed to detect each image's most representative pixel as seed for response map generation. First, the residual features $\mathcal{F}_{res}$ are input to our SSB. Then, the attention mechanism is employed, in which two 1×1 convolution layers are deployed to obtain two feature maps, namely $K \in \mathbb{R}^{N\times C\times H\times W}$ and $Q \in \mathbb{R}^{N\times C\times H\times W}$. After reshaping both K and Q to shape $\mathbb{R}^{NHW\times C}$, the feature similarity map (S) of each pixel is computed as

$$S = KQ^{\top}, \tag{5.2}$$

where $S \in \mathbb{R}^{NHW\times NHW}$, $\top$ means transpose, and each row of S represents similarities between one pixel and all pixels of the N inputs.

Then, we first reshape S into $S \in \mathbb{R}^{NHW \times N \times HW}$ and choose its maximum similarity value in each image, to get N maximum similarity values for each pixel. This process is calculated by

$$S^{N\text{-}max} = \max_{i=1 \cdots HW} S[:, :, i], \tag{5.3}$$

where $S^{N\text{-}max} \in \mathbb{R}^{NHW \times N}$. Afterwards, the average of the N maximum similarity values is treated as the co-salient probability of each pixel,

$$P = \frac{1}{N} \sum_{n=1}^{N} S^{N\text{-}max}[:, n], \tag{5.4}$$

where $P \in \mathbb{R}^{NHW}$.

Then, the probability map is reshaped back to $P \in \mathbb{R}^{N \times H \times W}$. We can locate the pixel with the highest probability of being the co-salient object in each image by

$$P^{max} = \max_{\substack{h=1, \cdots H \\ w=1, \cdots W}} P[:, h, w], \tag{5.5}$$

$$index = ind(P^{max}), \tag{5.6}$$

where $ind(\cdot)$ means taking out the index of P^{max}.

Finally, we take out the feature vectors from the $\mathcal{F}_{res}$ according to the index in Eq. 5.6 as the final seeds by

$$D = \mathcal{F}_{res}(index). \tag{5.7}$$

Note that each image will provide one seed vector, and there are totally N seeds. These seeds can represent the essential characteristics of the co-salient objects in each input image and be used for localization.

Democratic Response Block (DRB). The DRB is demonstrated in Fig. 5.3. If we directly use the seeds D as the prototype, it fails to aggregate comprehensive characteristics of the co-salient objects. This is because it is difficult for limited seeds to express the integral co-existed objects, especially when there are large appearance variations among the group. Thus, we try to involve more pixels of the co-salient objects to generate a comprehensive prototype by considering the correlation between each pixel and the seeds D from SSB.

Specifically, we first use L2 normalization in channel dimension to obtain the normalized residual features $\|\mathcal{F}_{res}\|_2$ and the normalized seeds $\|D\|_2$. Then, $\|D\|_2$ are treated as the kernel to

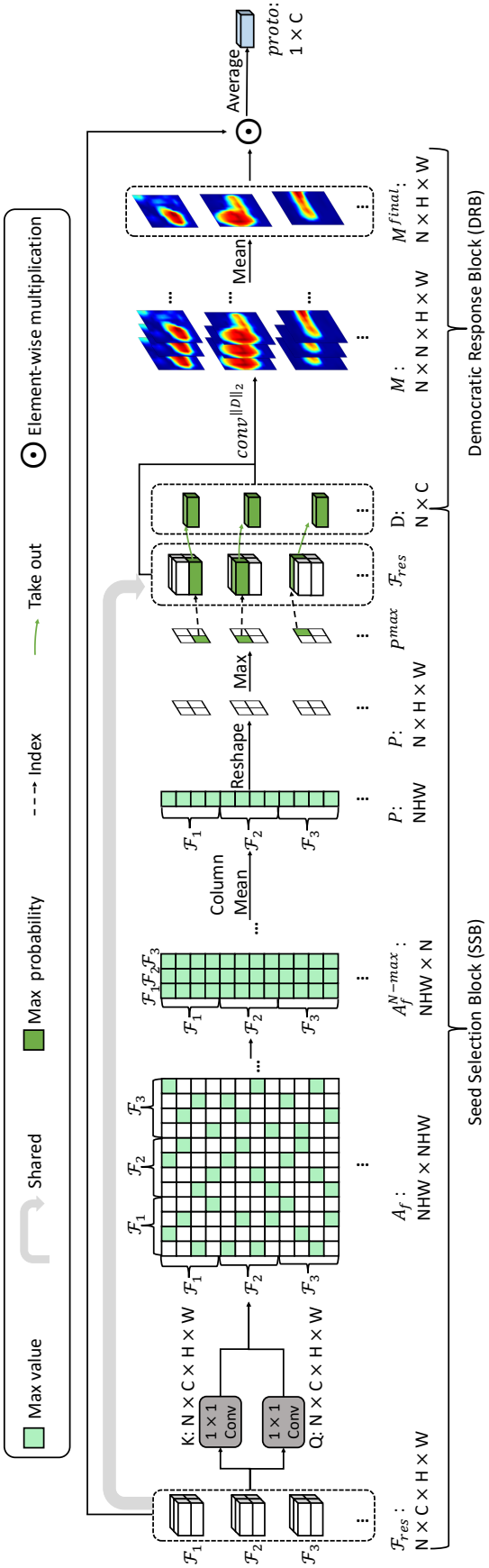


Figure 5.3: The framework of the seed selection block (SSB) and democratic response block (DRB). The inputs are the residual features. Then, the co-salient seeds are selected first from the residual features by SSB. After that, the response maps are produced using the selected seeds and the residual features through DRB. The final response maps and the input residual features are fused to generate the prototype.

conduct convolution on the $\|\mathcal{F}_{res}\|_2$:

$$M = conv^{\|D\|_2}(\|\mathcal{F}_{res}\|_2), \tag{5.8}$$

where M means response maps, $conv^{\|D\|_2}$ is the convolution with $\|D\|_2$ as kernel. Since D has N seed vectors, the size of response maps become $\mathbb{R}^{N\times N\times H\times W}$ after Eq. 5.8, with channel dimension being the number of response maps for each input.

The final democratic response map of each image is computed as the mean value of the N response maps:

$$M^{final} = \frac{1}{N}\sum_{n=1}^{N} M[:, n, :, :], \tag{5.9}$$

where $M^{final} \in \mathbb{R}^{N\times H\times W}$. In this way, more pixels have chance to contribute to the response maps.

Finally, the prototype ($proto \in \mathbb{R}^{1\times C}$) is generated by

$$proto = avg(M^{final} \odot \mathcal{F}_{res}), \tag{5.10}$$

where the M^{final} is broadcast to the same size as $\mathcal{F}_{res}$, $\odot$ denotes element-wise multiplication, and $avg(\cdot)$ means averaging feature vector of all the pixels from all inputs.

5.2.3 Self-Contrastive Learning Module

To further help the DPG to suppress the noise of background, and learn discriminative features without depending on classification information, a self-contrastive learning module (SCL) is designed as shown in Fig. 5.2. Our motivation is that the prototype generated by the original inputs (*proto*) should be consistent with co-salient prototype generated by inputs where background is erased ($proto_c$), but different from the background prototype generated by inputs where the co-salient objects are erased ($proto_b$). Note that the inputs here are the initial extracted features $\mathcal{F}_{ext}$ from the feature extractor. The co-salient prototype and background prototype can be generated as

$$proto_c = \phi_{\text{DPG}}(\mathcal{F}_{ext} \odot Y^{\downarrow}), \tag{5.11}$$

$$proto_b = \phi_{\text{DPG}}(\mathcal{F}_{ext} \odot (1 - Y^{\downarrow})), \tag{5.12}$$

where ϕ_{DPG} is short for the process of DPG, '$\downarrow$' means downscaling the groundtruth Y to the same size as $\mathcal{F}_{ext}$ then broadcasting to the same channel number.

Then, $proto$ and $proto_c$ are treated as a positive pair, while $proto$ and $proto_b$ are treated as a negative pair. A self-contrastive loss is designed to pull together the positive pair and push away the negative pair. First, we define the cosine-style similarity between the prototypes by

$$cos(p_1, p_2) = (1 + \frac{p_1 \cdot p_2}{|p_1|\,|p_2|}) \times 0.5. \tag{5.13}$$

After that, the self-contrastive loss is defined as

$$cos_c = cos(proto, proto_c), \tag{5.14}$$

$$cos_b = cos(proto, proto_b), \tag{5.15}$$

$$\mathcal{L}_{sc} = -log(cos_c + \epsilon) - log(1 - cos_b + \epsilon), \tag{5.16}$$

where ϵ is a small constant value ensuring non-zero values for $log(\cdot)$ and set as 1×10^{-5}. Our SCL is only applied during training as an auxiliary loss to help the DPG learn more discriminative co-salient features. This part is not used during inference.

5.2.4 Democratic Feature Enhancement Module

We design a democratic feature enhancement module (DFE) to further strengthen the fused co-salient features from DPG for final prediction. Our DFE is based on the attention mechanism [103]. We observe that conventional attention [103] tends to focus on a limited number of related pixels. Thus, we argue that democracy also matters in this case, and more pixels should be involved in enhancing the fused features. Thus, we try to amplify small positive attention values to involve more pixels for feature enhancement. Negative attention values are not considered here as they usually represent irrelevance. First, we generate the fused features using the guidance of both response maps and the prototype derived from Eq. 5.9 and Eq. 5.10 in the DPG as

$$\mathcal{F}_{fused} = \mathcal{F}_{res} \odot M^{final} + \mathcal{F}_{res} \odot proto, \tag{5.17}$$

where both the M^{final} and $proto$ are broadcast into the same size as $\mathcal{F}_{res}$. Therefore, the fused features by Eq. 5.17 contain both specific attributes and shared attributes.

The fused features of each input image are enhanced with our DFE individually and independently. As shown in Fig. 5.4, the corresponding $\mathcal{F}_{fused}$ is input to a 1×1 convolution layer followed by a ReLU activation to obtain $\mathcal{F}_{conv} \in \mathbb{R}^{C \times H \times W}$ first. After that, key, query and value convolutions

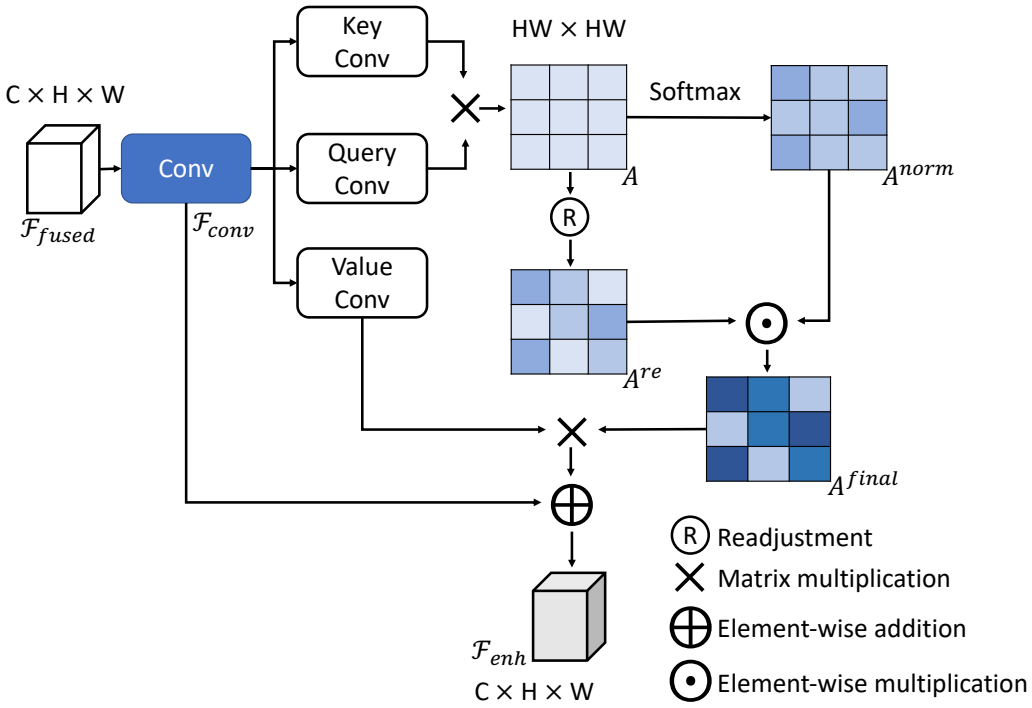


Figure 5.4: Flow chat of democratic feature enhancement module.

are applied and then reshaped to generate $\mathcal{F}_k \in \mathbb{R}^{HW\times C}$, $\mathcal{F}_q \in \mathbb{R}^{HW\times C}$ and $\mathcal{F}_v \in \mathbb{R}^{HW\times C}$. Then, the initial attention map (A) can be computed by

$$A = \mathcal{F}_k \mathcal{F}_q^\top, \tag{5.18}$$

where $A \in \mathbb{R}^{HW\times HW}$ and $\top$ means transpose.

Next, a softmax is applied to A to obtain the normalized attention map (A^{norm}). Moreover, the initial attention map A is sorted in a descending order to generate the sorting index matrix (Z). As we adopt the descending order, the small attention values are assigned with large sorting index. Then, we apply the following formula to amplify the small positive attention values,

$$A^{re}_{i,j} = \begin{cases} (Z_{i,j}+1)^{\alpha}, \text{if } A_{i,j} > 0 \\ \\ 1, \text{else} \end{cases}, \tag{5.19}$$

where A^{re} denotes the weights for readjusting the attention, α is a coefficient for determining the degree of amplification, i and j are the spatial index. Then, the final attention map is computed by

$$A^{final} = A^{norm} \odot A^{re}, \tag{5.20}$$

and the final enhanced features can be computed by

$$\mathcal{F}_{enh} = \mathcal{F}_{conv} + A^{final}\mathcal{F}_v, \tag{5.21}$$

where the result of $A^{final}\mathcal{F}_v$ is first reshaped back into the same size as $\mathcal{F}_{conv}$. Finally, the augmented features $\mathcal{F}_{enh}$ are transmitted into the decoder to predict the corresponding co-saliency maps.

5.2.5 Objective Function

The objective function for training is a combination of IoU loss [92, 140] and our self-contrastive loss in Eq. 5.16. The IoU loss can be illustrated as

$$\mathcal{L}_{iou} = 1 - \frac{1}{N}\sum \frac{\hat{Y} \cap Y}{\hat{Y} \cup Y}, \tag{5.22}$$

where $\hat{Y}$ denotes for predictions and Y denotes for the groundtruth. Then, the final objective function is

$$\mathcal{L}_{tot} = \mathcal{L}_{iou} + \lambda \mathcal{L}_{sc}, \tag{5.23}$$

where λ is to balance IoU loss and self-contrastive loss.

5.3 Experiment

5.3.1 Implementation Details

We use Feature Pyramid Network (FPN) [66] with VGG-16 [95] as our backbone. The hyper-parameter α in Eq. 5.19 is 3 and λ in Eq. 5.23 is 0.1. Additionally, we use Adam [50] as our optimizer to train our model for 200 epochs. The learning rate is set as 1×10^{-5} for feature extractor and 1×10^{-4} for other parts. The weight decay is set as 1×10^{-4}. In each training episode, we randomly choose one group (16 samples) of relative images. For inference, all samples in each group are input at one time. The inputs are resized into 224×224 for both training and inference. The total training time is around 3 hours and the inference time is around 84.4 fps. All experiments are run on one NVIDIA GeForce RTX 2080 Ti.

5.3.2 Dataset and Evaluation Metrics

Dataset. We use COCO-SEG [106], a subset of COCO dataset [67], which contains 9,213 images from 65 groups for training. We evaluate our method on three popular CoSOD benchmarks: CoCA [143], Cosal2015 [132] and CoSOD3k [24]. CoCA and CoSOD3k are proposed for challenging real-world co-saliency evaluation, containing multiple co-salient objects in some images, large appearance and scale variations, and complex background clutters. Cosal2015 is a widely used large dataset for the evaluation.

Evaluation Metrics. The evaluation metrics include mean absolute error (*MAE*) [16], maximum F-measure (F_β^{max}) [2], maximum E-measure (E_ϕ^{max}) [22] and S-measure (S_α) [21]. Specifically, the value of *MAE* is the smaller, the better. While others are the larger, the better.

5.3.3 Complexity Analysis with State-of-the-art Methods

The computational complexity of Eq.(2) and Eq.(18) in our paper is $O((NHW)^2)$ and $O((HW)^2)$ respectively. The increment of FLOPs is small since the input size is small. We list the complexity comparisons in Table 5.4, '†' means without DFE. Ours can achieve an impressive performance

Table 5.1: Comparisons with other state-of-the-art approaches on 3 benchmarks. ↑ means that larger is better and ↓ denotes that smaller is better. 'SOD' denotes training with extra SOD dataset.

Methods	SOD	CoCA				CoSOD3k				Cosal2015			
		$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$
ICNet [46] (NeurIPS20)	✓	0.148	0.506	0.698	0.651	0.097	0.744	0.832	0.780	0.058	0.855	0.900	0.856
CADC [139] (ICCV21)	✓	0.132	0.548	0.744	0.681	0.096	0.759	0.840	0.801	0.064	0.862	0.906	0.866
CoADNet [140] (NeurIPS20)	✓	-	-	-	-	0.070	0.825	-	0.837	0.064	0.875	-	0.861
CSMG [137] (CVPR19)		0.124	0.503	0.734	0.632	0.157	0.645	0.723	0.711	0.130	0.777	0.818	0.774
GCAGC [138] (CVPR20)		0.111	0.523	0.754	0.669	0.100	0.740	0.816	0.785	0.085	0.813	0.866	0.817
CoEGNet [23] (TPAMI21)		0.106	0.493	0.717	0.612	0.092	0.736	0.825	0.762	0.077	0.832	0.882	0.836
GICD [143] (ECCV20)		0.126	0.513	0.715	0.658	0.079	0.770	0.848	0.797	0.071	0.844	0.887	0.844
DeepACG [136] (CVPR21)		0.102	0.552	0.771	0.688	0.089	0.756	0.838	0.792	**0.064**	0.842	0.892	**0.854**
GCoNet [25] (CVPR21)		0.105	0.544	0.760	0.673	0.071	0.777	0.860	0.802	0.068	0.847	0.887	0.845
DCFM (Ours)		**0.085**	**0.598**	**0.783**	**0.710**	**0.067**	**0.805**	**0.874**	**0.810**	0.067	**0.856**	**0.892**	0.838

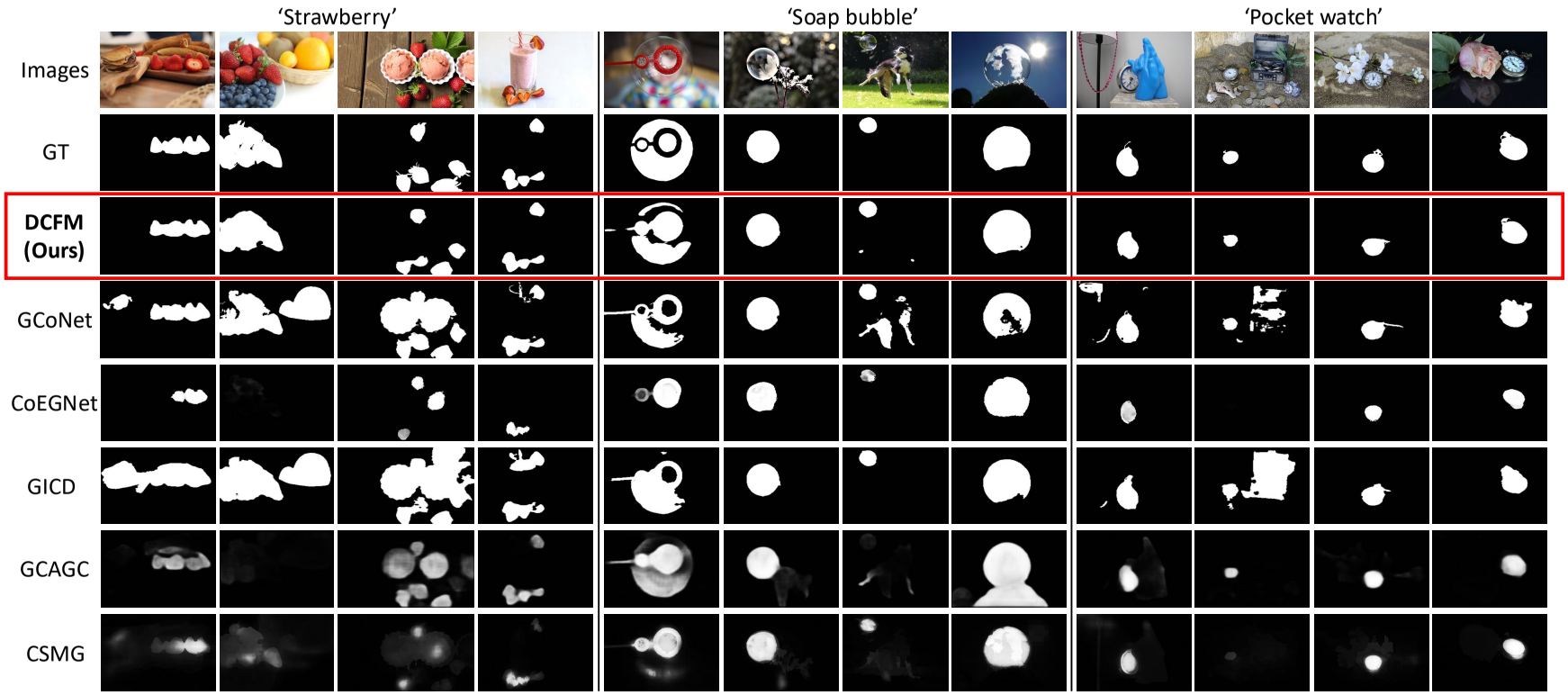


Figure 5.5: The qualitative comparisons with other state-of-the-art methods. It is evident that our method can predict smoother co-saliency maps with less noise of background, compared with other state-of-the-art methods.

Figure 5.6: More visualizations of our predictions and comparisons with previous state-of-the-art approaches. It can be found that our model can better differentiate the co-salient objects and background in complex scenes.

Table 5.2: Ablation study for our proposed modules. 'Base.' denotes baseline. Our overall method obtains the best results.

	Base.	DPG	SCL	DFE	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$
1	✓				0.129	0.521	0.735	0.655
2	✓	✓			0.097	0.575	0.763	0.696
3	✓	✓	✓		0.087	0.592	0.775	0.701
4	✓	✓	✓	✓	**0.085**	**0.598**	**0.783**	**0.710**

Table 5.3: Ablation study for different parts in DPG. 'RB' means the residual block. The overall process obtains the best performance.

	RB	SSB	DRB	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$
1				0.129	0.521	0.735	0.655
2	✓			0.124	0.527	0.745	0.659
3	✓	✓		0.126	0.527	0.739	0.657
4	✓	✓	✓	**0.097**	**0.575**	**0.763**	**0.696**

Table 5.4: Complexity comparisons. 'param.' denotes the number of parameters. We set 5 inputs to compute FLOPs.

method	FLOPs (G)	param. (M)	runtime (fps)	$F_\beta^{max} \uparrow$
CADC [139]$_{\text{ICCV21}}$	457.9	392.8	18.0	0.548
GICD [143]$_{\text{ECCV20}}$	467.6	278.0	40.8	0.513
GCoNet [25]$_{\text{CVPR21}}$	311.5	142.0	116.2	0.544
DCFM†(ours)	313.0	140.5	101.9	0.592
DCFM (ours)	316.6	142.3	84.4	**0.598**

with fewer FLOPs and parameters compared with CADC [139] and GICD [143]. Besides, ours can obtain a better performance with limited increment of FLOPs and parameters compared with GCoNet [25], especially for DCFM†. Overall, our method has an impressive performance with comparable runtime.

5.3.4 Comparison with State-of-The-Art

Compared Methods. We mainly compare with previous state-of-the-art methods trained on common single CoSOD training dataset for fair comparison, including CSMG [137], GCAGC [138], CoEGNet [23], GICD [143], GCoNet [25], and DeepACG [136]. We also list several methods trained on both CoSOD dataset and SOD dataset, such as CADC [139], ICNet [46] and CoADNet [140].

Quantitative Comparison. In Table 5.1, we list the performance comparisons between ours and previous state-of-the-art methods. It can be seen that our method reaches a new state-of-the-art performance compared with other approaches under the same settings. Specifically, for the two challenging real-world datasets CoCA and CoSOD3k, e.g., for CoCA, we obtain a gain of 2.0% for MAE, 5.4% for maximum F-measure, 2.3% for maximum E-measure, and 3.7% for S-measure compared with GCoNet [25]. Moreover, our method can even outperform those trained with extra SOD dataset on these two datasets, such as ICNet [46] and CADC [139]. For Cosal2015, our method obtains comparable results with DeepACG [136] and GCoNet [25]. This phenomenon may be caused by the fact that both DeepACG [136] and GCoNet [25] use extra classification information to provide structure information, while our method does not rely on any extra information.

Qualitative Comparison. We also report some qualitative comparisons with state-of-the-art methods in Fig. 5.5. The groups are from CoCA dataset. It can be found that our model can predict more integral and less noisy co-saliency maps compared with others. Specifically, when there are multiple co-salient objects in one image, like the group 'Strawberry', our model can detect all the target objects, compared with CoEGNet [23] and CSMG [137]. In the group 'Soap bubble', ours are sensitive to appearance variations compared with others. When the background noise level is high, such as the group 'Pocket watch', our predictions contain less noise, compared with GCoNet [25] and GICD [143]. We list more qualitative comparisons with previous sate-of-the-art methods in Fig. 5.6. It is evident that our predictions are closer to the ground truth. When the background contains misleading objects, such as the humans in the group 'Binoculars', our model can suppress the noisy information and focus on the targets, compared with GCoNet [25] and GICD [143]. Additionally, when there are complex background clutters, like images in the groups 'Pillow' and 'Tablet', compared with all other methods, ours are robust to this challenging setting.

5.3.5 Ablation Study

We conduct the ablation study of our method on the CoCA dataset by adding one module each time and treating the network with all our modules removed as the baseline. The results are shown in Table 5.2. It can be found that each proposed module contributes a lot. With our DPG, the performance can increase 3.2% for MAE, 5.4% for maximum F-measure, 2.8% for maximum E-measure, and 4.1% for S-measure. Our SCL enables further improvement by 1.0% for MAE, 1.7% for maximum F-measure, 1.2% for maximum E-measure, and 0.5% for S-measure, respectively. Besides, our model with DFE reach 0.085 for MAE, 0.598 for maximum F-measure, 0.783 for maximum E-measure, and 0.710 for S-measure. The new state-of-the-art performance is obtained when all the modules are included.

Impact of Democratic Prototype Generation Module. The evaluation of each block of our DPG is listed in Table 5.3. The experiment is conducted by adding one block at a time. Compared with the baseline (row 1), each part of DPG devotes to the final results. Specifically, if we only use RB and SSB, where we take the mean of seeds as the prototype, the results are even lower than the case without SSB, comparing row 2 and 3. On the other hand, with DRB, comparing row 3 and 4, the results will be increased by 2.9% for MAE, 4.8% for maximum F-measure, 2.4% for maximum E-measure, and 3.9% for S-measure. This phenomenon can verify that democracy does matter. More co-salient pixels should be enrolled for the comprehensive prototype.

Impact of Self-Contrastive Learning Module. We also evaluate two main parts in our

Table 5.5: Ablation study for different parts in Eq. 5.16 of SCL. 'cos_c' denotes the case with only positive pair for the loss and 'cos_b' denotes that with only negative pair. DFE is not used.

	cos_c	cos_b	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$
1			0.097	0.575	0.763	0.696
2	✓		0.093	0.574	0.764	0.695
3		✓	0.095	0.583	0.773	0.697
4	✓	✓	**0.087**	**0.592**	**0.775**	**0.701**

Table 5.6: Ablation study for readjustment in DFE. 'w/o DFE' denotes not using DFE, 'w/o RA' denotes using DFE without readjustment and 'w/ RA' denotes using DFE with readjustment.

		$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\xi^{max} \uparrow$	$S_\alpha \uparrow$
1	w/o DFE	0.087	0.592	0.775	0.701
2	w/o RA	0.100	0.567	0.769	0.691
3	w/ RA	**0.085**	**0.598**	**0.783**	**0.710**

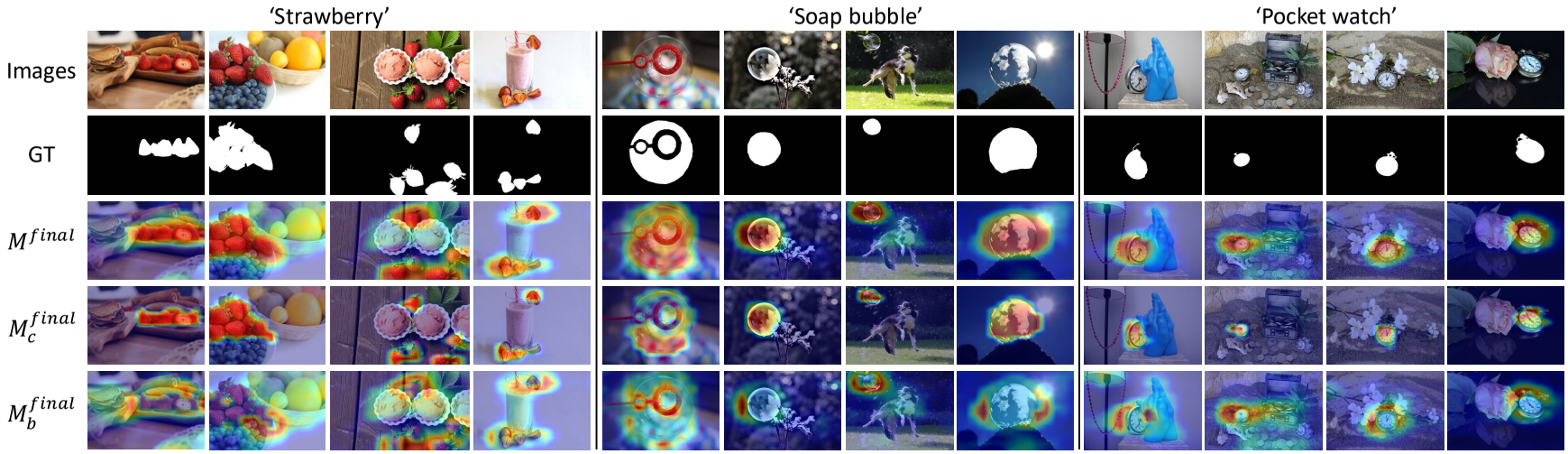


Figure 5.7: Visualization of the response maps in different cases. The visualizations can verify our assumption of the self-contrastive learning module as M^{final} is consistent with M_c^{final} but different from M_b^{final}.

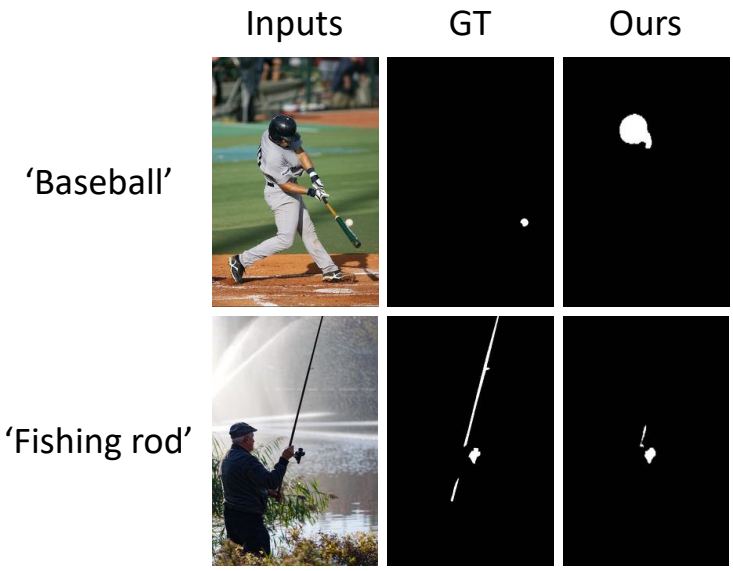


Figure 5.8: Visualizations of some failed cases.

self-contrastive loss as listed in Table 5.5. We conduct this experiment by removing one part each time. It can be seen that with only positive pair cos_c, by comparing row 1 and 2, we can get comparable results. With only negative pair cos_b, the performance is clearly improved, by comparing row 1 and 3. This phenomenon proves that the negative pair is important for removing background noise. Nevertheless, by comparing row 3 and 4, the contrastive learning with both positive and negative pairs promotes balanced training for higher results. More analysis can be found in our supplementary material. Further, we display some response maps in different cases on the CoCA dataset [143] in Fig. 5.7. Note that this dataset is used for evaluation. M^{final} denotes the normal response maps generated by original inputs, M_c^{final} denotes the co-salient response maps generated by inputs where the background regions are erased, and M_b^{final} denotes the background response maps generated by inputs where the co-salient objects are erased. Then, $proto$, $proto_c$ and $proto_b$ can be derived based on the corresponding response maps. As shown in Fig. 5.7, it can be found that the M^{final} can focus on most regions of the target co-salient objects. Moreover, comparing M_c^{final} and M_b^{final}, the M_c^{final} can highlight all the related co-salient objects. In contrast, the M_b^{final} are sensitive to the surroundings of the co-salient objects. In this case, our assumption of SCL, where $proto$ and $proto_c$ are pulled together while $proto$ and $proto_b$ are pushed away, can be verified. With our SCL, the model can learn to differentiate co-salient features and background features. Thus, the noise information can be suppressed.

Table 5.7: Influence of alpha in Eq.(19) in our thesis.

α	0.1	1	2	3	4
$F_\beta^{max} \uparrow$	0.578	0.592	0.593	**0.598**	0.587

Impact of Democratic Feature Enhancement Module. We also experiment on the readjustment of attention values in Table 5.6. When the readjustment is removed but using conventional attention in our DFE, the performance is even worse than the case without our DFE, as shown in row 1 and 2. Thus, democracy does matter in this module as well. Conventional attention mechanism focusing on limited pixels cannot provide sufficient information for the decoder while more related pixels should be involved. Additionally, we add the ablation study of alpha in Table 5.7. The performance smoothly increases with larger alpha. However, performance decreases when alpha is too big (α=4). When α>4, the model even fails to be trained. This is because in this case, the weight of small positive attention values will be much bigger. Thus, the attention mechanism will be confused and tend to focus on those small values but neglect original high values.

5.4 Limitation Discussion

We also report some failure cases in Fig. 5.8. As shown in the figure, it is difficult for our model to predict small objects precisely. This may be caused by the fact that the inputs are resized into the size of 224×224. Then, with the feature extractor, the size of the output features is 14×14. In this case, it may cause information lost for small objects. Thus, it is difficult for our model to capture the corresponding features. Therefore, how to enhance model robustness for small objects is a direction for our future work.

5.5 Conclusions

In this chapter, we have proposed a new method for CoSOD without using the SOD dataset and classification information. We design a democratic prototype generation module (DPG) to build democratic response maps first so as to generate a comprehensive prototype as guidance for further prediction. Moreover, to help suppress noisy background information in the prototype, we design a self-contrastive learning module (SCL), where both positive and negative pairs are

generated from the image itself without relying on classification information. Besides, we also design a democratic feature enhancement module (DFE) to strengthen co-salient features from DPG for final prediction. Both our DPG and DFE show that democracy does matter. More related pixels should be involved for mining comprehensive features for CoSOD.

Chapter 6

Conclusions

6.1 Summary

Video object segmentation and salient object detection are crucial tasks in video surveillance. Video object segmentation can help video understanding, and salient object detection helps mimic human attention. Thus, this thesis devotes to learning about these tasks. We propose a framework for fast pixel matching between reference and current frames in video object segmentation. We use the first frame with the given mask and the previous frame with the estimated mask as the references as guidance for target object localization and segmentation. After obtaining the matched features, a channel attention mechanism is adopted for further feature enhancement. Further experiments show that our approach can achieve a new state-of-the-art performance with a fast speed at the same time (86.5% IoU on DAVIS-2016 and 72.2% IoU on DAVIS-2017, with speed of 0.11s per frame) under the same level comparison.

We propose a local saliency coherence loss for salient object detection under scribble supervision. The loss is based on the assumption that points with similar features and/or close positions should have similar saliency values. In contrast, points with dissimilar feature and/or distinct positions should have different saliency values. In this case, more integral object features can be learnt to complement the scribble information. Further, different input scale is considered for consistent structure predictions. Experiments show that our method achieves a new state-of-the-art performance on six benchmarks (e.g. for the ECSSD dataset: $F_\beta = 0.8995$, $E_\xi = 0.9079$ and $MAE = 0.0489$), with an average gain of 4.60% for F-measure, 2.05% for E-measure and 1.88% for MAE over the previous best method on this task.

Last but not least, for co-salient object detection, we propose a novel framework called DCFM

to mine comprehensive co-salient features without the help of extra dataset or classification information. The democratic prototype generation module in our method can involve more related pixels to generate a comprehensive prototype to guide the following layers. Further, a self-contrastive module is applied to help learn the co-salient information and background information during training. Finally, a democratic feature enhancement module is designed to strengthen co-salient features. Extensive experiments show that our model obtains better performance than previous state-of-the-art methods, especially on challenging real-world cases (, for CoCA, we obtain a gain of 2.0% for MAE, 5.4% for maximum F-measure, 2.3% for maximum E-measure, and 3.7% for S-measure) under the same settings.

6.2 Future Works

There are still many challenges for future work. For video object segmentation, the critical problems faced are appearance variation, occlusion, and disappearing. Moreover, these problems become more serious when the video sequence is long. One possible way is to consider more reference frames. However, not every frame can offer a complete and smooth reference. Then, we consider to add an estimation branch to determine whether the reference frame is qualified. Further, bounding box or unsupervised cases can be studied to save labeling cost.

Then, for weakly supervised salient object detection, the model still faces problems of error estimation. Our local saliency coherence loss considers each pixel equally. In such way, some hard pixels are easy to be neglected. Thus, we consider to add a confidence map to reveal the hard and easy samples to help the network pay more attention to hard samples. Additionally, our self-consist mechanism only cares about scale variation. Other augmentations can be taken into consideration. Moreover, the smoother boundaries of the salient object should be predicted. Thus, we also consider to add a boundary branch to detect integral and precise contours.

Besides, in co-salient object detection, the computation complexity is high. To overcome this problem, we will try to use global average pooling as the co-salient semantic guidance to save the computation of the spatial attention mechanism. Besides, the over-fitting of training-set is a crucial challenge when there is a big domain gap between the test-set and training-set. Existed methods usually use an extra training-set to help the network learn more categories. However, such choice will add the cost of annotations. Thus, we plan to design a pixel-level contrastive loss to reveal the contrast between co-salient and background. Further, low-level features can also be used to distinguish detailed features.

Finally, salient object detection can help provide target object hints for video object segmenta-

tion. Thus, we consider to use saliency detection to help detect the target object first for automatic video object segmentation. However, the pseudo mask of the first frame may be not clear or integral if only referred to the first frame. In this case, co-salient object detection can be adopt to link several frames to help obtain target consistency through different frames for more complete target object cues. Then, with a good pseudo mask of the first frame, the automatic video object segmentation can be treated as semi-automatic video object segmentation for better predictions.

Publications

Conference:

1. Democracy Does Matter: Comprehensive Feature Mining for Co-salient Object Detection. Siyue Yu, Jimin Xiao, Bingfeng Zhang, Eng Gee Lim, *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2022.*

2. Structure Consistent Weakly Supervised Salient Object Detection with Local Saliency Coherence. Siyue Yu, Bingfeng Zhang, Jimin Xiao, Eng Gee Lim, *In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2021.*

Journal:

1. Fast pixel-matching for video object segmentation. Siyue Yu, Jimin Xiao, Bingfeng Zhang, Eng Gee Lim, Yao Zhao, *In Signal Processing: Image Communication. 98: 116373, 2021.*

2. Weight-Guided Class Complementing for Long-tailed Image Recognition. Xinqiao Zhao, Jimin Xiao, **Siyue Yu**, Hui Li, Bingfeng Zhang, *Submitted to Pattern Recognition (PR)*

Bibliography

[1] Radhakrishna Achanta, Francisco Estrada, Patricia Wils, and Sabine Süsstrunk. Salient region detection and segmentation. In *Int. Conf. Comput. Vis.*, pages 66–75. Springer, 2008.

[2] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009.

[3] Euijoon Ahn, Jinman Kim, Lei Bi, Ashnil Kumar, Changyang Li, Michael Fulham, and David Dagan Feng. Saliency-based lesion segmentation via background detection in dermoscopic images. *IEEE Journal of Biomedical and Health Informatics*, 21(6):1685–1693, 2017.

[4] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. What is an object? In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 73–80. IEEE, 2010.

[5] Dan Banica, Alexandru Agape, Adrian Ion, and Cristian Sminchisescu. Video object segmentation by salient segment chain composition.

[6] Linchao Bao, Baoyuan Wu, and Wei Liu. Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. In *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2018.

[7] Ali Borji. Boosting bottom-up and top-down visual features for saliency estimation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 438–445. IEEE, 2012.

[8] Ali Borji. What is a salient object? a dataset and a baseline model for salient object detection. *IEEE Transactions on Image Processing*, 24(2):742–756, 2014.

[9] Neil Bruce and John Tsotsos. Saliency based on information maximization. *Adv. Neural Inform. Process. Syst.*, 18, 2005.

[10] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixe, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, July 2017.

[11] Xiaochun Cao, Feng Wang, Bao Zhang, Huazhu Fu, and Chao Li. Unsupervised pixel-level video foreground object segmentation via shortest path algorithm. *Neurocomputing*, 172:235–243, 2016.

[12] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1189–1198, 2018.

[13] Zuyao Chen, Qianqian Xu, Runmin Cong, and Qingming Huang. Global context-aware progressive aggregation network for salient object detection. In *AAAI*, 2020.

[14] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, and Shi-Min Hu. Salientshape: Group saliency in image collections. *The Visual Computer*, 30(4):443–453, 2014.

[15] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):569–582, 2014.

[16] Ming-Ming Cheng, Jonathan Warrell, Wen-Yan Lin, Shuai Zheng, Vibhav Vineet, and Nigel Crook. Efficient salient region detection with soft image abstraction. In *Int. Conf. Comput. Vis.*, 2013.

[17] Runmin Cong, Jianjun Lei, Huazhu Fu, Ming-Ming Cheng, Weisi Lin, and Qingming Huang. Review of visual saliency detection with comprehensive information. *IEEE Trans. Circuit Syst. Video Technol.*, 29(10):2941–2959, 2018.

[18] Runmin Cong, Jianjun Lei, Huazhu Fu, Weisi Lin, Qingming Huang, Xiaochun Cao, and Chunping Hou. An iterative co-saliency framework for rgbd images. *IEEE Transactions on Cybernetics*, 49(1):233–246, 2017.

[19] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009.

[20] 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.

[21] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[22] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *IJCAI*, 2018.

[23] Deng-Ping Fan, Tengpeng Li, Zheng Lin, Ge-Peng Ji, Dingwen Zhang, Ming-Ming Cheng, Huazhu Fu, and Jianbing Shen. Re-thinking co-salient object detection. *arXiv preprint*, 2020.

[24] Deng-Ping Fan, Zheng Lin, Ge-Peng Ji, Dingwen Zhang, Huazhu Fu, and Ming-Ming Cheng. Taking a deeper look at co-salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[25] Qi Fan, Deng-Ping Fan, Huazhu Fu, Chi-Keung Tang, Ling Shao, and Yu-Wing Tai. Group collaborative learning for co-salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

[26] Mengyang Feng, Huchuan Lu, and Errui Ding. Attentive feedback network for boundary-aware salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[27] Huazhu Fu, Dong Xu, Bao Zhang, Stephen Lin, and Rabab Kreidieh Ward. Object-based multiple foreground video co-segmentation via multi-state selection graph. *IEEE Transactions on Image Processing*, 24(11):3415–3424, 2015.

[28] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[29] Lihua Fu, Yu Zhao, Xiaowei Sun, Jialiang Huang, Dan Wang, and Yu Ding. Video object segmentation based on motion-aware roi prediction and adaptive reference updating. *Expert Systems with Applications*, 167:114153, 2021.

[30] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016.

[32] Mengqi He, Jing Zhang, and Wenxin Yu. Salient object detection via bounding-box supervision. *arXiv preprint arXiv:2205.05245*, 2022.

[33] Shengfeng He, Rynson WH Lau, Wenxi Liu, Zhe Huang, and Qingxiong Yang. Supercnn: A superpixelwise convolutional neural network for salient object detection. *International Journal of Computer Vision*, 115(3):330–344, 2015.

[34] Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, pages 44–51. Springer, 2011.

[35] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[36] Ping Hu, Jun Liu, Gang Wang, Vitaly Ablavsky, Kate Saenko, and Stan Sclaroff. Dipnet: Dynamic identity propagation network for video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1904–1913, 2020.

[37] Rongyao Hu, Zhenyun Deng, and Xiaofeng Zhu. Multi-scale graph fusion for co-saliency detection. In *AAAI*, 2021.

[38] Yuan-Ting Hu, Jia-Bin Huang, and Alexander Schwing. Maskrnn: Instance level video object segmentation. In *Adv. Neural Inform. Process. Syst.*, pages 325–334, 2017.

[39] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G. Schwing. Videomatch: Matching based video object segmentation. In *Eur. Conf. Comput. Vis.*, September 2018.

[40] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998.

[41] Zhong Ji, Haoran Wang, Jungong Han, and Yanwei Pang. Saliency-guided attention network for image-sentence matching. In *Int. Conf. Comput. Vis.*, 2019.

[42] Bo Jiang, Xingyue Jiang, Jin Tang, and Bin Luo. Co-saliency detection via a general optimization model and adaptive graph learning. *IEEE Trans. Multimedia*, 23:3193–3202, 2021.

[43] Bo Jiang, Xingyue Jiang, Jin Tang, Bin Luo, and Shilei Huang. Multiple graph convolutional networks for co-saliency detection. In *Int. Conf. Multimedia and Expo*, 2019.

[44] Bo Jiang, Xingyue Jiang, Ajian Zhou, Jin Tang, and Bin Luo. A unified multiple graph learning and convolutional network model for co-saliency estimation. In *ACM Int. Conf. Multimedia*, 2019.

[45] Huaizu Jiang, Jingdong Wang, Zejian Yuan, Yang Wu, Nanning Zheng, and Shipeng Li. Salient object detection: A discriminative regional feature integration approach. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2083–2090, 2013.

[46] Wen-Da Jin, Jun Xu, Ming-Ming Cheng, Yi Zhang, and Wei Guo. Icnet: intra-saliency correlation network for co-saliency detection. In *Adv. Neural Inform. Process. Syst.*, 2020.

[47] Joakim Johnander, Martin Danelljan, Emil Brissman, Fahad Shahbaz Khan, and Michael Felsberg. A generative appearance model for end-to-end video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8953–8962, 2019.

[48] Anna Khoreva, Rodrigo Benenson, Eddy Ilg, Thomas Brox, and Bernt Schiele. Lucid data dreaming for object tracking. In *The DAVIS Challenge on Video Object Segmentation*, 2017.

[49] Hanjae Kim, Sunghun Joung, Ig-Jae Kim, and Kwanghoon Sohn. Prototype-guided saliency feature learning for person search. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4865–4874, 2021.

[50] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[51] Dominik A Klein and Simone Frintrop. Center-surround divergence of feature statistics for salient object detection. In *Int. Conf. Comput. Vis.*, pages 2214–2219. IEEE, 2011.

[52] Gayoung Lee, Yu-Wing Tai, and Junmo Kim. Deep saliency with encoded low level distance map and high level features. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 660–668, 2016.

[53] Seungho Lee, Minhyun Lee, Jongwuk Lee, and Hyunjung Shim. Railroad is not a train: Saliency as pseudo-pixel supervision for weakly supervised semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5495–5505, 2021.

[54] Youngjo Lee, Hongje Seong, and Euntai Kim. Iteratively selecting an easy reference frame makes unsupervised video object segmentation easier. *arXiv preprint arXiv:2112.12402*, 2021.

[55] Aixuan Li, Jing Zhang, Yunqiu Lv, Bowen Liu, Tong Zhang, and Yuchao Dai. Uncertainty-aware joint salient object and camouflaged object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

[56] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M. Rehg. Video segmentation by tracking many figure-ground segments. In *Int. Conf. Comput. Vis.*, 2013.

[57] Guanbin Li, Yuan Xie, and Liang Lin. Weakly supervised salient object detection using image labels. In *AAAI*, 2018.

[58] Guanbin Li and Yizhou Yu. Visual saliency based on multiscale deep features. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5455–5463, 2015.

[59] Hongliang Li and King Ngi Ngan. A co-saliency model of image pairs. *IEEE Transactions on Image Processing*, 20(12):3365–3375, 2011.

[60] Jia Li, Jinming Su, Changqun Xia, Mingcan Ma, and Yonghong Tian. Salient object detection with purificatory mechanism and structural similarity loss. *IEEE Trans. Image Process.*, 30:6855–6868, 2021.

[61] Xi Li, Yao Li, Chunhua Shen, Anthony Dick, and Anton Van Den Hengel. Contextual hypergraph modeling for salient object detection. In *Int. Conf. Comput. Vis.*, pages 3328–3335, 2013.

[62] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *Eur. Conf. Comput. Vis.*, pages 90–105, 2018.

[63] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014.

[64] Zhuwen Li, Qifeng Chen, and Vladlen Koltun. Interactive image segmentation with latent diversity. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 577–585, 2018.

[65] Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Agss-vos: Attention guided single-shot video object segmentation. In *Int. Conf. Comput. Vis.*, 2019.

[66] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[67] 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 *Eur. Conf. Comput. Vis.*, pages 740–755. Springer, 2014.

[68] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan L. Yuille, and Li Fei-Fei. Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2019.

[69] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[70] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2):353–367, 2010.

[71] Yun Liu, Xin-Yu Zhang, Jia-Wang Bian, Le Zhang, and Ming-Ming Cheng. Samnet: stereoscopically attentive multi-scale network for lightweight salient object detection. *IEEE Trans. Image Process.*, 30:3804–3814, 2021.

[72] Yuxuan Liu, Pengjie Wang, Ying Cao, Zijian Liang, and Rynson WH Lau. Weakly-supervised salient object detection with saliency bounding boxes. *IEEE Transactions on Image Processing*, 30:4423–4435, 2021.

[73] Zhi Liu, Wenbin Zou, Lina Li, Liquan Shen, and Olivier Le Meur. Co-saliency detection based on hierarchical segmentation. *IEEE Signal Processing Letters*, 21(1):88–92, 2013.

[74] 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.

[75] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *Asian Conference on Computer Vision*, pages 565–580. Springer, 2018.

[76] Yu-Fei Ma and Hong-Jiang Zhang. Contrast-based image attention analysis by using fuzzy growing. In *ACM Int. Conf. Multimedia*, pages 374–381, 2003.

[77] Dwarikanath Mahapatra, Syed Omer Gilani, and Mukesh Kumar Saini. Coherency based spatio-temporal saliency detection for video object segmentation. *IEEE Journal of Selected Topics in Signal Processing*, 8(3):454–462, 2014.

[78] K-K Maninis, Sergi Caelles, Yuhua Chen, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. Video object segmentation without temporal information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(6):1515–1530, 2018.

[79] Ran Margolin, Ayellet Tal, and Lihi Zelnik-Manor. What makes a patch distinct? In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1139–1146, 2013.

[80] Tam Nguyen, Maximilian Dax, Chaithanya Kumar Mummadi, Nhung Ngo, Thi Hoai Phuong Nguyen, Zhongyu Lou, and Thomas Brox. Deepusps: Deep robust unsupervised saliency prediction via self-supervision. In *Adv. Neural Inform. Process. Syst.* 2019.

[81] Anton Obukhov, Stamatios Georgoulis, Dengxin Dai, and Luc Van Gool. Gated crf loss for weakly supervised semantic image segmentation. *arXiv preprint arXiv:1906.04651*, 2019.

[82] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Int. Conf. Comput. Vis.*, pages 9226–9235, 2019.

[83] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu. Multi-scale interactive network for salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[84] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu. Multi-scale interactive network for salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[85] Derrick Parkhurst, Klinton Law, and Ernst Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision Research*, 42(1):107–123, 2002.

[86] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2663–2672, 2017.

[87] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 733–740, 2012.

[88] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 724–732, 2016.

[89] Yongri Piao, Jian Wang, Miao Zhang, and Huchuan Lu. Mfnet: multi-filter directive network for weakly supervised salient object detection. In *Int. Conf. Comput. Vis.*, 2021.

[90] 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*, 2017.

[91] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[92] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: boundary-aware salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[93] Baifeng Shi, Qi Dai, Yadong Mu, and Jingdong Wang. Weakly-supervised action localization by generative attention modeling. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[94] Jae Shin Yoon, Francois Rameau, Junsik Kim, Seokju Lee, Seunghak Shin, and In So Kweon. Pixel-level matching for video object segmentation using convolutional neural networks. In *Int. Conf. Comput. Vis.*, Oct 2017.

[95] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Int. Conf. Learn. Represent.*, 2014.

[96] Fengdong Sun and Wenhui Li. Saliency guided deep network for weakly-supervised image segmentation. *Pattern Recognition*, 120:62–68, 2019.

[97] Jia Sun, Dongdong Yu, Yinghong Li, and Changhu Wang. Mask propagation network for video object segmentation. *arXiv preprint arXiv:1810.10289*, 2018.

[98] Lv Tang, Bo Li, Yijie Zhong, Shouhong Ding, and Mofei Song. Disentangled high quality salient object detection. In *Int. Conf. Comput. Vis.*, 2021.

[99] Janine Thoma, Danda Pani Paudel, and Luc V Gool. Soft contrastive learning for visual localization. In *Adv. Neural Inform. Process. Syst.*, 2020.

[100] Anne M Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, 1980.

[101] Chung-Chi Tsai, Kuang-Jui Hsu, Yen-Yu Lin, Xiaoning Qian, and Yung-Yu Chuang. Deep co-saliency detection via stacked autoencoder-enabled fusion and self-trained cnns. *IEEE Trans. Multimedia*, 22(4):1016–1031, 2019.

[102] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *Int. Conf. Comput. Vis.*, pages 10052–10062, 2021.

[103] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inform. Process. Syst.*, 2017.

[104] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9481–9490, 2019.

[105] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for video object segmentation. *arXiv preprint arXiv:1706.09364*, 2017.

[106] Chong Wang, Zheng-Jun Zha, Dong Liu, and Hongtao Xie. Robust deep co-saliency detection with group semantic. In *AAAI*, 2019.

[107] Haoxiang Wang, Zhihui Li, Yang Li, Brij B Gupta, and Chang Choi. Visual saliency guided complex image retrieval. *Pattern Recognition*, 130:64–72, 2020.

[108] Lijun Wang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Deep networks for saliency detection via local estimation and global search. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3183–3192, 2015.

[109] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[110] Tiantian Wang, Lihe Zhang, Shuo Wang, Huchuan Lu, Gang Yang, Xiang Ruan, and Ali Borji. Detect globally, refine locally: A novel approach to saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[111] Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, Haibin Ling, and Ruigang Yang. Salient object detection in the deep learning era: An in-depth survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[112] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3395–3402, 2015.

[113] Wenguan Wang, Jianbing Shen, Ruigang Yang, and Fatih Porikli. Saliency-aware video object segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(1):20–33, 2017.

[114] Wenguan Wang, Hongmei Song, Shuyang Zhao, Jianbing Shen, Sanyuan Zhao, Steven CH Hoi, and Haibin Ling. Learning unsupervised video object segmentation through visual attention. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3064–3074, 2019.

[115] Wenguan Wang, Tianfei Zhou, Fatih Porikli, David Crandall, and Luc Van Gool. A survey on deep learning technique for video segmentation. *arXiv preprint arXiv:2107.01153*, 2021.

[116] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2018.

[117] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[118] Ziqin Wang, Jun Xu, Li Liu, Fan Zhu, and Ling Shao. Ranet: Ranking attention network for fast video object segmentation. In *Int. Conf. Comput. Vis.*, pages 3978–3987, 2019.

[119] Lina Wei, Shanshan Zhao, Omar El Farouk Bourahla, Xi Li, Fei Wu, and Yueting Zhuang. Deep group-wise fully convolutional network for co-saliency detection with graph propagation. *IEEE Trans. Image Process.*, 28(10):5052–5063, 2019.

[120] Yichen Wei, Fang Wen, Wangjiang Zhu, and Jian Sun. Geodesic saliency using background priors. In *Eur. Conf. Comput. Vis.*, pages 29–42. Springer, 2012.

[121] Runmin Wu, Mengyang Feng, Wenlong Guan, Dong Wang, Huchuan Lu, and Errui Ding. A mutual learning method for salient object detection with intertwined multi-supervision. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[122] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[123] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7376–7385, 2018.

[124] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1155–1162, 2013.

[125] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013.

[126] Rui Yao, Guosheng Lin, Shixiong Xia, Jiaqi Zhao, and Yong Zhou. Video object segmentation and tracking: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(4):1–47, 2020.

[127] Hongkai Yu, Kang Zheng, Jianwu Fang, Hao Guo, Wei Feng, and Song Wang. Co-saliency detection within a single image. In *AAAI*.

[128] Siyue Yu, Bingfeng Zhang, Jimin Xiao, and Eng Gee Lim. Structure-consistent weakly supervised salient object detection with local saliency coherence. In *AAAI*. AAAI Palo Alto, CA, USA, 2021.

[129] Xiaohui Zeng, Renjie Liao, Li Gu, Yuwen Xiong, Sanja Fidler, and Raquel Urtasun. Dmm-net: Differentiable mask-matching network for video object segmentation. In *Int. Conf. Comput. Vis.*, October 2019.

[130] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, Lihe Zhang, Mingyang Qian, and Yizhou Yu. Multi-source weak supervision for saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6074–6083, 2019.

[131] Dingwen Zhang, Huazhu Fu, Junwei Han, Ali Borji, and Xuelong Li. A review of co-saliency detection algorithms: fundamentals, applications, and challenges. *ACM Trans.Intell. Syst. and Tech.*, 9(4):1–31, 2018.

[132] Dingwen Zhang, Junwei Han, Chao Li, Jingdong Wang, and Xuelong Li. Detection of co-salient objects by looking deep and wide. *Int. J. Comput. Vis.*, 120(2):215–232, 2016.

[133] Dingwen Zhang, Junwei Han, and Yu Zhang. Supervision by fusion: Towards unsupervised learning of deep salient object detector. In *Int. Conf. Comput. Vis.*, 2017.

[134] Jing Zhang, Xin Yu, Aixuan Li, Peipei Song, Bowen Liu, and Yuchao Dai. Weakly-supervised salient object detection via scribble annotations. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[135] Jing Zhang, Tong Zhang, Yuchao Dai, Mehrtash Harandi, and Richard Hartley. Deep unsupervised saliency detection: A multiple noisy labeling perspective. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[136] Kaihua Zhang, Mingliang Dong, Bo Liu, Xiao-Tong Yuan, and Qingshan Liu. Deepacg: co-saliency detection via semantic-aware contrast gromov-wasserstein distance. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

[137] Kaihua Zhang, Tengpeng Li, Bo Liu, and Qingshan Liu. Co-saliency detection via mask-guided fully convolutional networks with multi-scale label smoothing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[138] Kaihua Zhang, Tengpeng Li, Shiwen Shen, Bo Liu, Jin Chen, and Qingshan Liu. Adaptive graph convolutional network with attention graph clustering for co-saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[139] Ni Zhang, Junwei Han, Nian Liu, and Ling Shao. Summarize and search: learning consensus-aware dynamic convolution for co-saliency detection. In *Int. Conf. Comput. Vis.*, 2021.

[140] Qijian Zhang, Runmin Cong, Junhui Hou, Chongyi Li, and Yao Zhao. Coadnet: collaborative aggregation-and-distribution networks for co-salient object detection. In *Adv. Neural Inform. Process. Syst.*, 2020.

[141] Xiaoning Zhang, Tiantian Wang, Jinqing Qi, Huchuan Lu, and Gang Wang. Progressive attention guided recurrent network for salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[142] Yizhuo Zhang, Zhirong Wu, Houwen Peng, and Stephen Lin. A transductive approach for video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6949–6958, 2020.

[143] Zhao Zhang, Wenda Jin, Jun Xu, and Ming-Ming Cheng. Gradient-induced co-saliency detection. In *Eur. Conf. Comput. Vis.*, 2020.

[144] Rui Zhao, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Saliency detection by multi-context deep learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1265–1274, 2015.

[145] Ting Zhao and Xiangqian Wu. Pyramid feature attention network for saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.