

Weakly Supervised Video Salient Object Detection

Wangbo Zhao¹ Jing Zhang^{2,3} Long Li¹ Nick Barnes² Nian Liu⁴ Junwei Han¹✉*

¹ The Brain and Artificial Intelligence Laboratory, Northwestern Polytechnical University

² Australian National University ³ CSIRO, Australia

⁴ Inception Institute of Artificial Intelligence

{wangbo.zhao96, zjnwpu, longli.nwpu, liunian228, junweihan2010}@gmail.com,
nick.barnes@anu.edu.au

Abstract

Significant performance improvement has been achieved for fully-supervised video salient object detection with the pixel-wise labeled training datasets, which are time-consuming and expensive to obtain. To relieve the burden of data annotation, we present the first weakly supervised video salient object detection model based on relabeled “fixation guided scribble annotations”. Specifically, an “Appearance-motion fusion module” and bidirectional ConvLSTM based framework are proposed to achieve effective multi-modal learning and long-term temporal context modeling based on our new weak annotations. Further, we design a novel foreground-background similarity loss to further explore the labeling similarity across frames. A weak annotation boosting strategy is also introduced to boost our model performance with a new pseudo-label generation technique. Extensive experimental results on six benchmark video saliency detection datasets illustrate the effectiveness of our solution¹.

1. Introduction

Video salient object detection (VSOD) models are designed to segment salient objects in both the spatial domain and the temporal domain. Existing VSOD methods focus on two different solutions: 1) encoding temporal information using a recurrent network [30, 10, 44], e.g. LSTM; and 2) encoding geometric information using the optical flow constraint [18, 29]. Although considerable performance improvements have been achieved, we argue that the huge burden of pixel-wise labeling makes VSOD a much more expensive task than the RGB image-based saliency detection task [14, 23, 22, 48, 43].

The standard pipeline to train a deep video saliency

*Corresponding author: Junwei Han (junweihan2010@gmail.com)

¹Our code and data is publicly available at: <https://github.com/wangbo-zhao/WSVOD>.

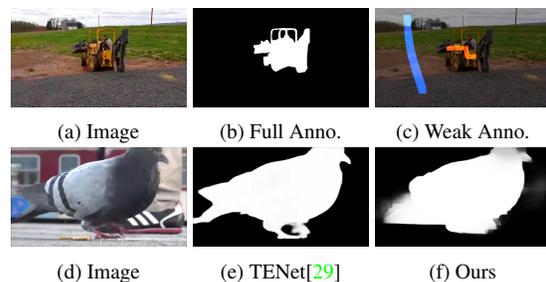


Figure 1. Training with our weak annotation (c), we achieve competitive performance (f) compared with TENet[29] (e).

detection model involves two main steps. Firstly, the network is pre-trained on an existing static RGB image-based saliency detection training dataset, e.g. DUTS [35] or MSRA10K [6]. Then, it is fine-tuned on video saliency detection datasets, e.g. DAVSOD [10] and DAVIS [28]. The main reason for using this strategy is that video saliency datasets usually have limited scene diversity. Although the largest DAVSOD dataset [10] has more than 10K frames for training, the large redundancy across the frames of each clip makes it still insufficient to effectively train deep video saliency models. Specifically, DAVSOD has a total of 107 clips for training and validation, which only indicates around 107 diverse scenes. Hence, directly training with a VSOD dataset may lead to poor model generalization ability, as the model may overfit on the highly redundant data.

To obtain an effective video saliency detection model, existing fully supervised VSOD methods [18, 29, 10] rely on both RGB image saliency datasets and VSOD training datasets. The problem behind the above pipeline is the huge requirement for pixel-wise labeling, which is time-consuming and expensive to obtain. For example, RGB image saliency training datasets have more than 10K labeled samples [35, 6]. Further, as shown in Tab. 1, widely used VSOD training datasets (DAVSOD and DAVIS) con-

Table 1. Details of existing video sod datasets. Dataset: name of the dataset, Size: number of frames, Annotated size: labeled frames(per pixel), Training: Frames used for training, /: this dataset is not split.

Dataset	Released Year	Size	Annotated size	Training
DAVSOD[10]	2019	23,938	23,938	12,670
VOS[19]	2018	116,103	7,467	5,927
DAVIS[28]	2016	3,455	3,455	2,079
ViSal[39]	2015	963	193	/
FBMS[26]	2014	13,860	720	353
SegV2[16]	2013	1,065	1,065	/

tain more than 14K pixel-wise labeled frames. Both of them required large burden to perform data annotations.

To relieve the burden of pixel-wise labeling, one can resort the weakly supervised learning technique [47, 35] to learn saliency from image scribble or image-level labels. In this paper, considering the efficiency of scribble annotation, we aim to learn a weakly supervised video saliency detection network via scribble. However, the main problem is that the per-image labeled scribble has no temporal information. To incorporate temporal information into our weak annotation, we adopt the fixation annotation in existing VSOD training datasets as guidance, and propose fixation guided scribble annotation as shown in Fig. 1 (c). Specifically, we first define the regions that have the peak response of fixation as foreground and those without fixation as background. Then we label both foreground scribble and background scribble following [47].

Based on the fixation guided scribble annotation, we design an appearance-motion fusion module to fuse both appearance information from the RGB image and motion information from optical flow as shown in Fig. 2. Furthermore, a bidirectional LSTM [30] based temporal information enhanced module is presented to further obtain long-term temporal information. Note that, we use scribble annotation from S-DUTS [35] to pre-train our video saliency detection network as the conventional way. Build upon both scribble annotation from the RGB image saliency dataset and video saliency dataset, our weakly supervised video saliency detection network leads to a very cheap configuration compared with existing deep video saliency detection models. Moreover, considering the cross-frame redundancy of the video saliency dataset, we introduce the foreground-background similarity loss to fully explore our weak annotation. We also introduce a weak annotation boosting strategy by leveraging our scribble annotation and the saliency map generated from the off-the-shelf fully-supervised SOD model. Benefiting from these, our model can achieve comparable results with state-of-the-art fully-supervised methods. *e.g.* Fig. 1 (f) and (e).

Our main contributions are: 1) We introduce the first weakly supervised video salient object detection network based on our fixation guided scribble annotation; 2) We propose an appearance-motion fusion module and a tempo-

ral information enhance module to effectively fuse appearance and motion features; 3) We present the foreground-background similarity loss to explore our weak annotation in adjacent frames; 4) We combine saliency maps generated from an off-the-shelf saliency model and our scribble annotations to further boost model performance.

2. Related Work

Fully supervised video salient object detection: As the mainstream of video salient object detection, the fully supervised video saliency detection models mainly focus on exploring both spatial and temporal information of the training dataset. Wang *et al.* [40] models the short-term spatial-temporal information by taking two adjacent frames as input. To model the longer spatio-temporal information, [30, 17] adopt ConvLSTM to capture richer spatial and temporal features simultaneously. Some methods also model the human attention mechanism to select interesting regions in different frames, *e.g.* self-attention [12], spatial attention supervised by human eye fixation data [10, 37]. As objects with motion in a video are usually salient, li *et al.* [18] use optical flow as guidance to find the salient regions. Ren *et al.* [29] combine the spatial and temporal information and present a semi-curriculum learning strategy to reduce the learning ambiguities.

While these methods show their successes on VSOD, they heavily rely on the large densely annotated training datasets. Annotating a high-quality dataset is expensive and time-consuming. Different from them, depending on only weak annotations, our method greatly relief the labeling burden, which is both cheaper and more accessible.

Weakly/semi/un-supervised video salient object detection: There are many traditional unsupervised VSOD methods *e.g.* [38, 39, 19], most of which exploit handcrafted features, which makes them unsatisfactory in the real-world application. When it comes to the learning-based methods, although the problem of depending on laborious and costly pixel-wise dense annotation is obvious and serious, few methods make an effort to alleviate it. To the best of our knowledge, there is no previous method to solve VSOD with totally weakly labeled data. There are only several methods that try to use less annotated data. Yan *et al.* [44] addresses VSOD in a semi-supervised manner by using pseudo labels, where the optical flow map serves as guidance for pseudo label generation with the sparsely annotated frames. Tang *et al.* [33] uses limited manually labeled data and pseudo labels generated from existing saliency models to train their model. Recently weakly-supervised finetuning during testing is also explored. Li *et al.* [21] proposes to generate pseudo labels to weakly retrain pre-trained saliency models during testing. But high-quality labeled data is still inevitable to obtain the pseudo labels.

Different from previous methods, which rely on all or

part of the fully annotated dataset, our model is end-to-end trainable without using any densely annotated labels.

Video object segmentation: There are two types of video object segmentation (VOS) models, including Zero-shot VOS [37, 30, 49] and One-shot VOS [11, 3, 42]. During testing, the former aims at segmenting primary objects in a video without any help, while the first annotated frame is given in the later. Since Zero-shot VOS is more similar to VSOD, we only discuss the related literature in this paper. Among them, Song *et al.* [30] solve VOS and VSOD at the same time. Wang *et al.* [36] builds a fully connected graph to mine rich and high-order relations between video frames. Zhou *et al.* [49] proposes a motion attention block to leverage motion information to reinforce spatio-temporal object representation. [25] introduces an encoder-decoder network consisting entirely of 3D convolutions.

Like VSOD, most of the video segmentation models are fully supervised, and the weakly-supervised or semi-supervised counterparts is still under-explored. Until recently, Lu *et al.* [24] introduces the intrinsic properties of VOS at multiple granularities to learn from weak supervision. However, the undesirable performance and slow inference speed makes it desirable to further explore this task.

3. Our Method

3.1. Overview

As a weakly supervised video saliency detection framework, we first relabel existing video saliency detection datasets DAVSOD [10] and DAVIS [28] with scribble labels. Due to a lack of temporal information in the per-image scribble annotation, we introduce fixation guided scribble annotation as shown in Fig. 3. Our training dataset is then defined as $T = \{X, F, Y\}$, where X is the RGB image, F is the optical flow map predicted from [31], Y is our fixation guided scribble annotation. We first design a *Saliency feature extraction* f_α to extract features $f_\alpha(X)$ and $f_\alpha(F)$ from RGB images and flow respectively, where α is the network parameter set. Then, we present the *Appearance-Motion Fusion Module* (AMFM) $g_\beta(f_\alpha(X), f_\alpha(F))$ to effectively learn from both appearance information (*e.g.* the RGB image branch) and motion information (*e.g.* the flow branch). Further, we introduce a *Temporal Information Enhanced Module* (TIEM) $s_\gamma(g_\beta)$ by using ConvLSTM to model the long-term temporal information between frames. For each frame, we fuse features from TIEM in different levels in a top-down manner to get the final output. To further explore the temporal information from our fixation guided scribble annotation, we present a *Foreground-background similarity loss* as a frame-wise constraint. Moreover, we present a *Saliency boosting strategy* to further improve the performance of our method. An overview of our network is shown in Fig. 2.

3.2. Fixation guided scribble annotation

The largest video saliency detection dataset, *i.e.* DAVSOD [10], is annotated in two steps: 1) an eye tracker is used to record fixation points, the output of which is then Gaussian blurred to obtain a dense fixation map; and 2) annotators segment the whole scope of the salient foreground based on the peak response region². As indicated in [10], the extra fixation annotation introduces useful temporal information to the video saliency dataset. Conventionally, DAVSOD is combined with the DAVIS [28] dataset to train fully supervised VSOD models. Originally, DAVIS had no fixation annotation, however, it was added by Wang *et al.* [37]. As a weak video saliency detection network, we intend to use the fixation data as guidance to obtain temporal information, and we then replace the pixel-wise clean annotation with scribble for weakly supervised learning. Given every frame in our video saliency training dataset, as shown in Fig. 3 (a), and the corresponding fixation map as shown in Fig. 3 (b), we annotate the foreground scribble in the objects with peak response regions, and background scribble in other region as shown in Fig. 3 (d). In this case, the generated scribble annotation encodes temporal information, which is different from [47] where the scribble is totally image-based, with no temporal information.

3.3. Saliency feature extraction

As shown in Fig. 2, the saliency feature extraction module is used to extract the appearance saliency feature $f_\alpha(X)$ from the RGB image X and motion saliency feature $f_\alpha(F)$ from the optical flow map F . We build our architecture upon ResNet-50 [13] and remove the down-sampling operations in stage four³ to keep the spatial information. Apart from this, we replace the convolutional layers in the last layer with dilated convolutions [46] with a dilated rate of 2. An ASPP [4] module is added after stage four to extract multi-scale spatial features, which includes one 1×1 convolutional layer, and three 3×3 dilated convolutional layers with dilation rate of 6, 12, and 18, and a global average pooling operation. With our saliency feature extraction module, we obtain the appearance feature $f_\alpha(X) = \{f_r^1, f_r^2, f_r^3, f_r^4\}$ and motion feature $f_\alpha(F) = \{f_m^1, f_m^2, f_m^3, f_m^4\}$ respectively, where f_r^k and f_m^k are the appearance feature and motion feature of the k -th stage of the network, and k indexes network stages. More details can be found in Section 4. We also add an extra edge detection branch to recover the structure information of the final output, and details of which can be found in [47].

²The peak response region is the region with the densest fixation points.

³We define a group of convolutional layers of the same spatial size as belonging to the same stage.

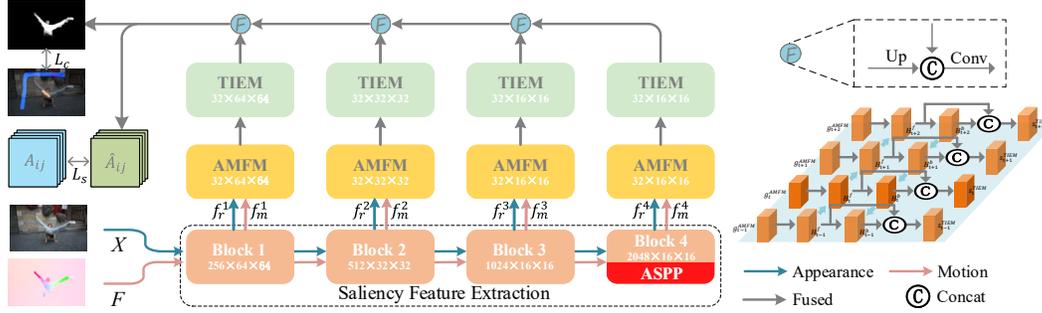


Figure 2. Overview of the proposed model. For simplicity, we do not show the edge detection branch borrowed from [47] here. Details about TIEM can be found on the right. There is no upsample operation in the first "F". "Up": "Upsample operation"; "C": concatenation operation; "Conv": 3×3 convolutional layer.

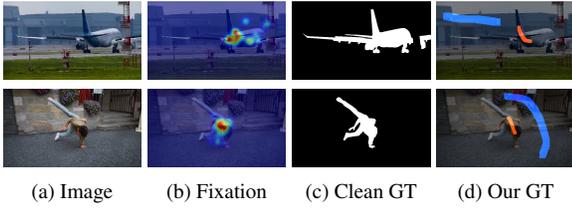


Figure 3. The fixation guided scribble annotation, where we obtain the sequence of scribble labels(d) with the temporal information from fixation annotation(b).

3.4. Appearance-motion fusion module

The appearance-motion fusion module aims to effectively fuse the appearance feature $f_\alpha(X)$ and motion feature $f_\alpha(F)$. As shown in Fig. 4, the inputs of the AMFM are the appearance feature f_r^k and the motion feature f_m^k of size $C \times W \times H$. We use two convolutional layers with a ReLU activation function to reduce the number of channels of f_r^k and f_m^k to $C = 32$ respectively. Then the concatenation operation and a 1×1 convolutional layer is adopted to obtain the fused feature g_{rm}^k of size $C \times W \times H$, which contains the appearance and motion information. We use g_{rm} instead of g_{rm}^k in the following for simplicity.

There exists three sub-modules in our AMFM, namely the gate module (GM), the channel attention module (CAM) and the spatial attention module (SAM). The gate module is designed to control the importance of appearance features and motion features, and the two attention modules are used to select the discriminative channels and locations. In GM, two different gates can be generated from g_{rm} , namely the appearance gate $G_r(g_{rm})$ and motion gate $G_m(g_{rm})$. This module is designed to control the importance of f_r and f_m , which is formulated as:

$$\mathbf{G} = \text{GAP}(\sigma(\text{Conv}(g_{rm}; \beta))), \quad (1)$$

where $\mathbf{G} = [G_r, G_m]$, and G_r, G_m are two scalars in the range $[0, 1]$. $\text{Conv}(g_{rm}; \beta)$ is a 1×1 convolutional layer,

which reduces the channels of feature g_{rm} from C to 2. $\text{GAP}(\ast)$ is the global average pooling layer in the spatial dimension, and β is the network parameter set. $\sigma(\ast)$ is the sigmoid function.

The gate module produces two different scalars, representing the importance of appearance information and motion information. However, it can not emphasize important channels and spatial locations in appearance and motion features. Based on this, we propose our two attention modules, namely CAM and SAM, as:

$$\mathbf{CA} = \text{Softmax}(\text{FC}(\text{MaxPooling}(g_{rm}); \beta)), \quad (2)$$

$$\mathbf{SA} = \sigma(\text{Conv}(g_{rm}; \beta)), \quad (3)$$

where $\mathbf{CA} = [c_r, c_m]$ are the two channel attention maps of size $C \times 1 \times 1$ for appearance and motion. $\text{MaxPooling}(\ast)$ is in the spatial dimensions. FC is a fully connected layer with $2C$ output channels. The Softmax function is implemented in every C channels along the channel dimension. $\mathbf{SA} = [s_r, s_m]$, and s_r, s_m are two spatial attention maps of size $1 \times W \times H$. Subsequently, the obtained gates $[G_r, G_m]$, channel attention tensors $[c_r, c_m]$, spatial attention tensors $[s_r, s_m]$ can be multiplied with f_r and f_m respectively to achieve both importance reweighting (the gate module) and attention reweighting (the attention modules). However, such a simple multiplication approach may lose some useful information. Inspired by [18, 41], we use the gated feature in a residual form as:

$$g_r = (G_r \otimes f_r)(1 + s_r \otimes c_r), \quad (4)$$

$$g_m = (G_m \otimes f_m)(1 + s_m \otimes c_m), \quad (5)$$

where \otimes denotes element-wise multiplication with broadcast. Finally, the output will be added to get the fused feature $g^{AMFM} = g_r + g_m$.

3.5. Temporal information enhanced module

Although the appearance-motion fusion module can effectively fuse appearance information from the RGB image

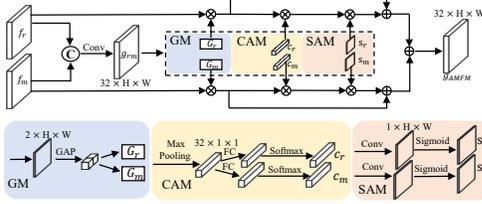


Figure 4. Appearance-motion fusion module. GM, CAM and SAM are the gate module, channel and spatial attention module respectively. Their outputs are G , CA and SA , respectively.

and motion information from the flow, we still observe undesirable prediction when we use it alone. We argue that this arises due to two reasons: 1) the optical flow map can only provide temporal information between two adjacent frames, where no long-term temporal information exists; 2) as the flow feature is fused with the appearance feature in AMFM, some lower quality of flow may introduce extra noise to the network, leading to deteriorated predictions.

In order to solve this problem, we model long-term temporal information in our ‘‘Temporal information enhanced module’’ (TIEM) by adopting the bidirectional ConvLSTM [30] to further constrain the cross-frames spatial and temporal information. Unlike previous methods [10, 44], which only add the temporal model in the highest level, we add a TIEM after each AMFM to promote information flow in each feature level between frames.

With the bidirectional ConvLSTM [30], we obtain hidden states H_t^f and H_t^b from both the forward and backward ConvLSTM units, which can be formulated as:

$$H_t^f = \text{ConvLSTM}(H_{t-1}^f, g_t^{\text{AMFM}}; \gamma), \quad (6)$$

$$H_t^b = \text{ConvLSTM}(H_{t+1}^b, H_t^f; \gamma), \quad (7)$$

$$s_t^{\text{TIEM}} = \text{Conv}(\text{Cat}(H_t^f, H_t^b); \gamma), \quad (8)$$

where g_t^{AMFM} and s_t^{TIEM} represent the features from the AMFM and TIEM respectively.

3.6. Foreground-background similarity loss

Different from [47] which can learn saliency from independent static images, our model learns video saliency with fixation-guided scribbles, where annotation of adjacent frames are related. The large redundancy in adjacent frames makes it possible to re-use scribble annotations of other frames to supervise the current frame. Further, we observe that it is difficult for the network to determine the category of each pixel without per-pixel annotation. Motivated by [45], we propose our ‘‘Foreground-background similarity loss’’ to take advantage of limited weakly annotated labels and model the relationship of all points in adjacent frames. We argue that the similarity of features of the same category (both salient or both background) should be

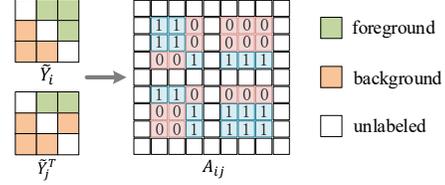


Figure 5. The illustration of how to get the ground truth of the similarity map $\hat{A}_{i,j}$. Note that, we do not define the similarity between unlabeled points and other points in $A_{i,j}$.

larger than that of different categories. Based on this, we first calculate the similarity of two feature maps. To be specific, for the feature map of the i th frame f_i and j th frame f_j , we first use a 1×1 convolutional layer to map them into an embedding space. Subsequently, we reshape them into $C \times WH$. Then we conduct matrix multiplication followed the sigmoid activation function σ to get the similarity map \hat{A} of $HW \times HW$ size. It can be formulated as:

$$\hat{A}_{i,j} = \sigma(\text{Conv}(f_i)^T \text{Conv}(f_j)) \quad (9)$$

where $\text{Conv}(\ast)$ is a 1×1 convolutional layer. $\hat{A}_{i,j}$ represents the obtained similarity map between i th frame and j th frame. Then we need to construct a ground truth map to supervise $\hat{A}_{i,j}$. Given the weakly annotated label of i th frame Y_i , we first downsample it into the same size as the feature map f_i , so we obtain a smaller label \tilde{Y}_i . We encode the foreground part in \tilde{Y}_i into $[1, 0]$ and the background part into $[0, 1]$, leading to a tensor \tilde{Y}_i of size $2 \times H \times W$. Then, we reshape it into $2 \times HW$. We do the same operations to the j -th frame and obtain \tilde{Y}_j . Then, we conduct the matrix multiplication again and obtain $A_{i,j} = \tilde{Y}_i \tilde{Y}_j^T$ of size $HW \times HW$. We visualize this process in Fig. 5. Note that all operations above are done on labeled points, which means that we do not define the similarity between unlabeled points and other points. We use J to represent the set of points in $A_{i,j}$. Then we can adapt partial cross-entropy loss [32] to supervise the similarity map:

$$L_s^{i,j} = - \sum_{u,v \in J} (A_{u,v} \log \hat{A}_{u,v} + (1 - A_{u,v}) \log(1 - \hat{A}_{u,v})). \quad (10)$$

For each iteration we have T frames, we can calculate the similarity loss for the current frame with other frames and itself. So the total similarity loss can be formulated as:

$$L_s = \sum_{i=1}^T \sum_{j=i}^T L_s^{i,j}. \quad (11)$$

3.7. Loss Function

As shown in Fig. 2, we employ both partial cross-entropy loss L_c and the proposed foreground-background similarity loss L_s to train our model. Apart from this, the gated

structure-aware loss L_g and edge loss l_e proposed in [47] are also used. Note that, we do not show L_g and l_e in Fig. 2 for simplicity. Both L_s , L_g and l_e are the loss for learning from scribble labels of the static image and L_s is the loss for learning from a series of frames. Following the conventional video saliency detection pipeline, we pretrain with an RGB saliency dataset. Differently, we use the scribble annotation based dataset, namely S-DUTS [47]. Then, we finetune the network with our fixation guided scribble annotation. To pretrain the network, we define the loss as:

$$L_{pretrain} = \beta_1 \cdot L_c + \beta_3 \cdot L_g + \beta_4 \cdot L_e. \quad (12)$$

Then we finetune the network with loss function:

$$L_{fine} = \beta_1 \cdot L_c + \beta_2 \cdot L_s + \beta_3 \cdot L_g + \beta_4 \cdot L_e. \quad (13)$$

Empirically, we set $\beta_1 = \beta_2 = \beta_4 = 1$ and $\beta_3 = 0.3$.

3.8. Saliency boosting strategy

Our model based on the fixation guided scribble annotation leads to competitive performance as shown in Table 2 “Ours”. Furthermore, we notice that some SOD methods, *e.g.* [48], can also achieve reasonable results on VSOD datasets. Inspired by [33], we propose a saliency consistency based pseudo label boosting technique guided by the SOD model to further refine our annotation.

Specifically, we adopt EGNNet [48] to generate the saliency maps for the RGB images and optical flow of our video saliency training dataset, which are defined as p_{rgb} and p_m respectively. Note that choosing other off-the-shelf SOD methods is also reasonable. As done in [21], we choose the intersection of p_{rgb} and p_m as the fused saliency map $p = p_{rgb} \odot p_m$, which captures the consistent salient regions of p_{rgb} and p_m . Our basic assumption is that p contains all the foreground scribble, and covers no background scribble. With this, we define the quality score of p as:

$$score = \frac{\|T(p) \odot s_{fore}\|_0}{\|s_{fore}\|_0} \cdot (1 - \frac{\|T(p) \odot s_{back}\|_0}{\|s_{back}\|_0}) \quad (14)$$

where $T(*)$ binarizes the saliency map with threshold 0.5. $\|*\|_0$ is the L0 norm. s_{fore} and s_{back} are the foreground and background scribble respectively as shown in Fig. 3 (d).

The first part of the quality score aims to evaluate the coverage of foreground scribble over p , while the latter encourages no background scribble to overlap p . In this way, the higher quality score indicates a better saliency map of p . We then choose saliency maps with quality score larger than a pre-defined threshold, *e.g.* $Tr = 0.98$. For each sequence, we can then obtain a set of high-quality pseudo saliency maps P . If the number of pseudo saliency maps in P is larger than 10% of the number of frames in current sequence, we replace the scribble annotation with the generated high quality pseudo label. Otherwise, we keep

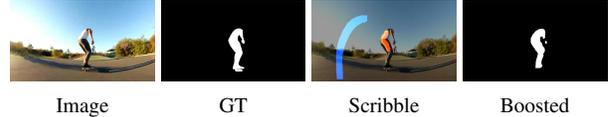


Figure 6. Our boosting strategy can refine the partial annotation.

our original fixation guided scribble annotation for the entire sequence. We define the new weak annotation set as our first stage pseudo label set D_{b1} .

For sequences with high quality pseudo labels, we train them individually with the corresponding annotation in D_{b1} . After K^4 iterations of training, we perform inference using the trained model to obtain the second stage pseudo label set D_{b2} . Note that the model to train each sequence is introduced in Section 4.2 as the ablation study “B”. Then, we treat D_{b2} as our boosted annotation, and train our whole model with D_{b2} . During training, if a frame has a generated pseudo label, we directly use it as supervision. Otherwise, we use our scribble annotation to supervise. In Fig. 6 we show the boosted annotation “Boosted”, which clearly show the effectiveness of our boosting strategy.

4. Experimental Results

Dataset: Similar to the conventional VSOD learning pipeline, our model is pre-trained on the scribble based image saliency dataset S-DUTS [47] and then fine-tuned on our fixation guided scribble annotations, namely the DAVIS-S and DAVSOD-S datasets. We evaluate the proposed method on six public datasets: VOS[19], DAVIS[28], DAVSOD[10], FBMS[26], SegV2[16] and ViSal[39]. The details of those datasets are shown in Table 1.

Implementation details: As shown in Fig 2, our network takes image and flow as input. We first adopt an off-the-shelf optical flow estimation method [31] to compute the flow map from the previous frame to the current frame. For the S-DUTS dataset, we just take the output from [31] as the flow map of the static image by inputting the two same images into it. We take ResNet-50 [13] pretrained on ImageNet [8] as the backbone. Note that during pretraining, there is no TIEM in our network. We resize all the frames to same spatial size of 256×256 before we feed them to the network. Random flipping and cropping are also added to avoid over-fitting. The optimization algorithm is Adam [15] and the learning rate is $1e-4$. We pre-train and fine-tune for 30 epochs and 20 epochs respectively. The batch size is set to one, and the length of frames per batch is set to four due to computation resource limitations. The whole pre-training and fine-tuning takes about four hours and nine hours respectively on a PC with an NVIDIA GeForce RTX 2080Ti GPU. During test, our average processing time for one frame of a sequence is 0.035s.

⁴ $K = 8 \times$ the size of the current sequence.

Table 2. Benchmarking results. Bold numbers represent the best performance. \uparrow & \downarrow denote larger and smaller is better, respectively. Ours* means our method with the proposed boosting strategy. We use red and blue to indicate the two best scores.

Metric	Fully Sup. Models												Weakly/unsup. Models						
	EGNet [48]	SCRN [43]	PoolNet [22]	SCOM [5]	MBNM [20]	FCNS [40]	PDB [30]	FGRN [17]	MGA [18]	RCRNet [44]	SSAV [10]	PSCA [12]	TENet [29]	SSOD [47]	GF [39]	SAG [38]	Ours	Ours*	
VOS	$S_\alpha \uparrow$	0.793	0.825	0.773	0.712	0.742	0.760	0.818	0.715	0.791	0.873	0.786	0.828	0.845	0.682	0.615	0.619	0.750	0.765
	$F_\beta \uparrow$	0.698	0.749	0.709	0.690	0.670	0.675	0.742	0.669	0.734	0.833	0.704	0.747	0.781	0.648	0.506	0.482	0.666	0.702
	$\mathcal{M} \downarrow$	0.082	0.067	0.082	0.162	0.099	0.099	0.078	0.097	0.075	0.051	0.091	0.065	0.052	0.106	0.162	0.172	0.091	0.089
DAVIS	$S_\alpha \uparrow$	0.829	0.879	0.854	0.832	0.887	0.794	0.882	0.838	0.910	0.886	0.892	0.902	0.905	0.795	0.688	0.676	0.828	0.846
	$F_\beta \uparrow$	0.768	0.847	0.815	0.783	0.861	0.708	0.855	0.783	0.892	0.848	0.860	0.880	0.881	0.734	0.569	0.515	0.779	0.793
	$\mathcal{M} \downarrow$	0.057	0.029	0.038	0.048	0.031	0.061	0.028	0.043	0.023	0.027	0.028	0.022	0.017	0.044	0.100	0.103	0.037	0.038
DAVSOD	$S_\alpha \uparrow$	0.719	0.745	0.702	0.599	0.637	0.657	0.698	0.693	0.741	0.741	0.755	0.741	0.779	0.672	0.553	0.565	0.705	0.694
	$F_\beta \uparrow$	0.604	0.652	0.592	0.464	0.520	0.521	0.572	0.573	0.643	0.654	0.659	0.656	0.697	0.556	0.334	0.370	0.605	0.593
	$\mathcal{M} \downarrow$	0.101	0.085	0.089	0.220	0.159	0.129	0.116	0.098	0.083	0.087	0.084	0.086	0.070	0.101	0.167	0.184	0.103	0.115
FBMS	$S_\alpha \uparrow$	0.878	0.876	0.839	0.794	0.857	0.794	0.851	0.809	0.908	0.872	0.879	0.868	0.916	0.747	0.651	0.659	0.778	0.803
	$F_\beta \uparrow$	0.848	0.861	0.830	0.797	0.816	0.759	0.821	0.767	0.903	0.859	0.865	0.837	0.915	0.727	0.571	0.564	0.786	0.792
	$\mathcal{M} \downarrow$	0.044	0.039	0.060	0.079	0.047	0.091	0.064	0.088	0.027	0.053	0.040	0.040	0.024	0.083	0.160	0.161	0.072	0.073
SegV2	$S_\alpha \uparrow$	0.845	0.817	0.782	0.815	0.809	*	0.864	*	0.880	0.843	0.849	0.866	0.868	0.733	0.699	0.719	0.804	0.819
	$F_\beta \uparrow$	0.774	0.760	0.704	0.764	0.716	*	0.808	*	0.829	0.782	0.797	0.811	0.810	0.664	0.592	0.634	0.738	0.762
	$\mathcal{M} \downarrow$	0.024	0.025	0.025	0.030	0.026	*	0.024	*	0.027	0.035	0.023	0.024	0.025	0.039	0.091	0.081	0.033	0.033
ViSal	$S_\alpha \uparrow$	0.946	0.948	0.902	0.762	0.898	0.881	0.907	0.861	0.940	0.922	0.942	0.946	0.949	0.853	0.757	0.749	0.857	0.883
	$F_\beta \uparrow$	0.941	0.946	0.891	0.831	0.883	0.852	0.888	0.848	0.936	0.907	0.938	0.941	0.949	0.831	0.683	0.688	0.831	0.875
	$\mathcal{M} \downarrow$	0.015	0.017	0.025	0.122	0.020	0.048	0.032	0.045	0.017	0.026	0.021	0.017	0.012	0.038	0.107	0.105	0.041	0.035

Competing methods: We compare our method with 16 state-of-the-art image/video saliency methods as shown in Table 2. Since SSOD [47] is the only scribble based saliency model, we finetune it with our scribble DAVSOD dataset for a fair comparison.

Evaluation metrics: We use three criteria to evaluate the performance of our method and competing methods, including Mean Absolute Error (MAE), F-measure [1] (F_β), and the structure measure S-measure [9] (S_α).

4.1. Comparison with the state-of-the-art

Quantitative Comparison: In Table 2, we show the results of our method and the competing methods. We can observe that our method can outperform all other weakly supervised or unsupervised method on six datasets. Comparing with the only scribble base method SSOD [47], although it has been finetuned on DAVIS-S and DAVOSD-S, we still can surpasses it by a large margin. That is mainly because our method can take advantage of the motion and temporal information between frames. Moreover, our method can also achieve competitive performance with some fully-supervised methods, *e.g.*, FCNS [40], FGRN [17], SCOM [5] and MBNM [20].

Qualitative Comparison: We select four representative frames from four sequences in the testset of DAVSOD in Fig. 7. We compare our method with the five best fully-supervised methods MGA [18], RCRNet [44], SSAV [10], PSCA [12], and TENet [29] and two weakly/unsupervised methods SSOD [47] and GF [39]. More qualitative comparison can be found in the supplementary materials. Benefiting from motion and temporal information, our method can locate salient objects more accurately than other weakly/unsupervised methods. Our method also shows comparable performance with fully-supervised methods on sequences with complex scenes (row 1), quick motion (row 2), multiple objects (row 3) and slow motion (row4).

Table 3. Performance of our ablation study related experiments.

Method	DAVSOD			FBMS		
	$S_\alpha \uparrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
B	0.670	0.543	0.116	0.749	0.707	0.085
B(G)	0.578	0.424	0.222	0.631	0.568	0.196
B-Fc	0.669	0.538	0.123	0.769	0.743	0.081
B-Fa	0.678	0.556	0.119	0.763	0.735	0.084
B-Fo	0.682	0.562	0.112	0.775	0.763	0.08
B-Fo-Th	0.681	0.556	0.121	0.780	0.769	0.076
B-Fo-Ta	0.694	0.585	0.108	0.781	0.781	0.074
B-Fo-Ta-L	0.705	0.605	0.103	0.778	0.786	0.072

4.2. Ablation study

We thoroughly analyze the proposed framework and provide extra experiments as shown in Table 3.

Scribble annotation based baseline: We employ the proposed DAVSOD-S and DAVIS-S to finetune the base model, which has been pretrained on S-DUTS. The base model is constructed by removing all TIEM and replacing all AMFM with convolutional layers. It only takes RGB images as input. The performance is marked as ‘‘B’’. We also conduct an experiment by leveraging GraphCut [2] to generate masks given scribble annotations and directly adopting them to train ‘‘B’’. This model is denoted as ‘‘B(G)’’. The result in Table. 3 shows inferior performance of ‘‘B(G)’’. The main reason is graph-based algorithms can not generate accurate masks from simple scribble annotations, which further explains superior performance of the proposed solution.

Different appearance-motion fusion strategy: In order to add the motion information, we introduce the optical flow map into the base model and add the AMFM module into ‘‘B’’. This model is named ‘‘B-Fo’’. To demonstrate the effectiveness of the proposed appearance-motion fusion strategy, we also compare it with two simple fusion strategies: element-wise addition and concatenation. They are denoted as ‘‘B-Fa’’ and ‘‘B-Fc’’ respectively. As shown in Tab. 3, our method surpasses ‘‘B-Fa’’ by 0.4% on S_α and 0.6% on F_β on DAVSOD. The improvement on FBMS is much larger, compared with ‘‘B-Fc’’ and ‘‘B-Fa’’ the F_β is increased by

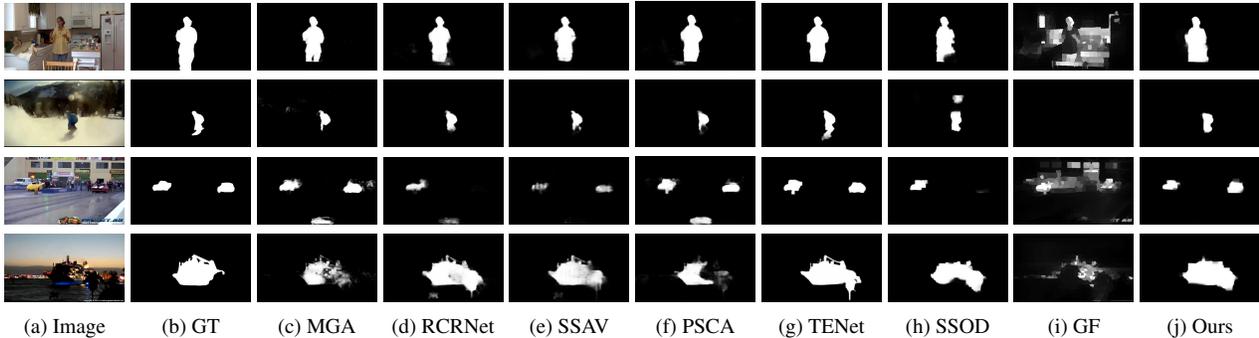


Figure 7. Qualitative comparison with state-of-the-art video salient object detection methods.

2% and 2.8%, respectively.

Temporal information enhanced module: We add TIEM to “B-Fo” to explore the effectiveness of long-term temporal information. Specifically, we propose two different solutions. Firstly, we only use the temporal model at the highest level of the network (Block 4 in Fig. 2), which leads to “B-Fo-Th”. Secondly, we introduce TIEM to every level of our network as in Fig. 2, which is “B-Fo-Ta”. Experiments show that “B-Fo-Ta” works better than “B-Fo-Th”. Especially on DAVSOD, we observe significant performance improvement of “B-Fo-Ta”, which achieves \mathcal{M} of 0.108, far better than “B-Fo-Th” with \mathcal{M} of 0.121.

Foreground-background similarity loss: We add the foreground-background similarity loss to “B-Fo-Ta” to cooperate with binary cross-entropy loss. The performance is indicated as “B-Fo-T-L”. Compared with “B-Fo-Ta”, since the proposed loss can provide extra frame-wise supervision, we obtain better performance with less training ambiguity.

Boosting strategy: We perform the boosting strategy to our method in Tab. 2 “Ours”, and achieve “Ours*”. We observe that this strategy can generally improve model performance, which clearly shows the effectiveness of the proposed boosting strategy. Further, we notice decreased performance of “Ours*” compared with “Ours” on DAVSOD dataset. We then analyse the generated pseudo labels from the boosting strategy, and find that there exist some low quality pseudo labels, which mainly come from the inconsistent performance of EGNNet [48] on our training dataset. This inspires us to explore further on boosting strategy. Designing a strategy to avoid the accumulated error due to boosting by taking both labels before and after the boosting strategy into consideration is a potential solution.

5. Zero-shot Video Object Segmentation

Similar to video salient object detection, the zero-shot video object segmentation aims to segment the primary object in a video sequence, which is usually the salient object. To evaluate generalization ability of the proposed method, we test on the validation set of DAVIS, which is

Table 4. Performance of video object segmentation on DAVIS.

Metric	Fully Sup. Models			Weakly/Un sup. Models				
	PDB [30]	AGNN [36]	MATNet [49]	MM [27]	TSN [7]	COSE [34]	MuG [24]	Ours
$\mathcal{J} \uparrow$	77.2	80.7	82.4	48.9	31.2	52.8	61.2	63.8
$\mathcal{F} \uparrow$	74.5	79.1	80.7	39.1	32.2	49.3	56.1	52.4

widely used for zero-shot video object segmentation evaluation. We compare our method with three fully-supervised methods (PDB [30], AGNN [36], MATNet [49]) and four weakly/unsupervised methods (MM [27], TSN [7], COSE [34], MuG [24]) on mean Jaccard index \mathcal{J} and mean contour accuracy \mathcal{F} . As shown in Table 4, our method outperforms other weakly/unsupervised video segmentation methods on \mathcal{J} , with slightly decreased \mathcal{F} compared with Mug[24]. Note that Mug [24] is trained with more than 1.5 million frames, which is 100 times larger than our training dataset. Furthermore, its inference time is 0.6 seconds per frame, and ours is 0.011 seconds per frame. Both the decreased amount of training data and the efficient inference time indicate that the proposed weakly-supervised strategy has the potential to be applied to video object segmentation.

6. Conclusion

We propose a novel weakly-supervised VSOD network trained on the proposed fixation guided scribble datasets, namely DAVIS-S and DAVSOD-S. We introduce multi-modality learning and a temporal constraint to effectively model spatio-temporal information. Furthermore, we propose a novel similarity loss and fully explore the limited weakly annotations. A saliency boosting strategy is also introduced to leverage off-the-shelf SOD methods. Extensive experiments on VSOD and VOS illustrate that our method is effective and general. Moreover, we are the first to use a weakly-supervised setting and achieve comparable results, which we hope may be inspiring for future work.

Acknowledgements

This research was supported in part by the National Science Foundation of China under Grant U1801265 and CSIRO’s Machine Learning and Artificial Intelligence Future Science Platform (MLAI FSP).

References

- [1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1597–1604, 2009. [7](#)
- [2] Yuri Y Boykov and M-P Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Int. Conf. Comput. Vis.*, volume 1, pages 105–112, 2001. [7](#)
- [3] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 221–230, 2017. [3](#)
- [4] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. [3](#)
- [5] Y. Chen, W. Zou, Y. Tang, X. Li, C. Xu, and N. Komodakis. Scom: Spatiotemporal constrained optimization for salient object detection. *IEEE T. Image Process.*, 27(7):3345–3357, 2018. [7](#)
- [6] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE T. Pattern Anal. Mach. Intell.*, 37(3):569–582, 2015. [1](#)
- [7] Ioana Croitoru, Simion-Vlad Bogolin, and Marius Leordeanu. Unsupervised learning from video to detect foreground objects in single images. In *Int. Conf. Comput. Vis.*, pages 4335–4343, 2017. [8](#)
- [8] 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.*, pages 248–255, 2009. [6](#)
- [9] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *Int. Conf. Comput. Vis.*, pages 4548–4557, 2017. [7](#)
- [10] Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. Shifting more attention to video salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8554–8564, 2019. [1](#), [2](#), [3](#), [5](#), [6](#), [7](#)
- [11] Matthias Grundmann, Vivek Kwatra, Mei Han, and Irfan Essa. Efficient hierarchical graph-based video segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2141–2148, 2010. [3](#)
- [12] Yuchao Gu, Lijuan Wang, Ziqin Wang, Yun Liu, Ming-Ming Cheng, and Shao-Ping Lu. Pyramid constrained self-attention network for fast video salient object detection. In *AAAI Conf. Art. Intell.*, pages 10869–10876, 2020. [2](#), [7](#)
- [13] 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. [3](#), [6](#)
- [14] 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.*, pages 3203–3212, 2017. [1](#)
- [15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. [6](#)
- [16] 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.*, pages 2192–2199, 2013. [2](#), [6](#)
- [17] Guanbin Li, Yuan Xie, Tianhao Wei, Keze Wang, and Liang Lin. Flow guided recurrent neural encoder for video salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3243–3252, 2018. [2](#), [7](#)
- [18] Haofeng Li, Guanqi Chen, Guanbin Li, and Yizhou Yu. Motion guided attention for video salient object detection. In *Int. Conf. Comput. Vis.*, pages 7274–7283, 2019. [1](#), [2](#), [4](#), [7](#)
- [19] Jia Li, Changqun Xia, and Xiaowu Chen. A benchmark dataset and saliency-guided stacked autoencoders for video-based salient object detection. *IEEE T. Image Process.*, 27(1):349–364, 2017. [2](#), [6](#)
- [20] Siyang Li, Bryan Seybold, Alexey Vorobyov, Xuejing Lei, and C.-C. Jay Kuo. Unsupervised video object segmentation with motion-based bilateral networks. In *Eur. Conf. Comput. Vis.*, September 2018. [7](#)
- [21] Yunxiao Li, Shuai Li, Chenglizhao Chen, Aimin Hao, and Hong Qin. A plug-and-play scheme to adapt image saliency deep model for video data. *IEEE T. Circuit Syst. Video Technol.*, 2020. [2](#), [6](#)
- [22] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3917–3926, 2019. [1](#), [7](#)
- [23] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Pixel-wise contextual attention learning for accurate saliency detection. *IEEE T. Image Process.*, 2020. [1](#)
- [24] Xiankai Lu, Wenguan Wang, Jianbing Shen, Yu-Wing Tai, David J Crandall, and Steven CH Hoi. Learning video object segmentation from unlabeled videos. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8960–8970, 2020. [3](#), [8](#)
- [25] Sabarinath Mahadevan, Ali Athar, Aljoša Ošep, Sebastian Hennen, Laura Leal-Taixé, and Bastian Leibe. Making a case for 3d convolutions for object segmentation in videos. *arXiv preprint arXiv:2008.11516*, 2020. [3](#)
- [26] Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE T. Pattern Anal. Mach. Intell.*, 36(6):1187–1200, 2013. [2](#), [6](#)
- [27] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2701–2710, 2017. [8](#)
- [28] 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. [1](#), [2](#), [3](#), [6](#)
- [29] Sucheng Ren, Chu Han, Xin Yang, Guoqiang Han, and Shengfeng He. Tenet: Triple excitation network for video salient object detection. *arXiv preprint arXiv:2007.09943*, 2020. [1](#), [2](#), [7](#)

- [30] Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *Eur. Conf. Comput. Vis.*, pages 715–731, 2018. [1](#), [2](#), [3](#), [5](#), [7](#), [8](#)
- [31] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8934–8943, 2018. [3](#), [6](#)
- [32] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. Normalized cut loss for weakly-supervised cnn segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1818–1827, 2018. [5](#)
- [33] Yi Tang, Wenbin Zou, Zhi Jin, Yuhuan Chen, Yang Hua, and Xia Li. Weakly supervised salient object detection with spatiotemporal cascade neural networks. *IEEE T. Circuit Syst. Video Technol.*, 29(7):1973–1984, 2018. [2](#), [6](#)
- [34] Yi-Hsuan Tsai, Guangyu Zhong, and Ming-Hsuan Yang. Semantic co-segmentation in videos. In *Eur. Conf. Comput. Vis.*, pages 760–775, 2016. [8](#)
- [35] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan. Learning to detect salient objects with image-level supervision. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3796–3805, 2017. [1](#), [2](#)
- [36] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *Int. Conf. Comput. Vis.*, pages 9236–9245, 2019. [3](#), [8](#)
- [37] Wenguan Wang, Jianbing Shen, Xiankai Lu, Steven CH Hoi, and Haibin Ling. Paying attention to video object pattern understanding. *IEEE T. Pattern Anal. Mach. Intell.*, 2020. [2](#), [3](#)
- [38] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3395–3402, 2015. [2](#), [7](#)
- [39] Wenguan Wang, Jianbing Shen, and Ling Shao. Consistent video saliency using local gradient flow optimization and global refinement. *IEEE T. Image Process.*, 24(11):4185–4196, 2015. [2](#), [6](#), [7](#)
- [40] Wenguan Wang, Jianbing Shen, and Ling Shao. Video salient object detection via fully convolutional networks. *IEEE T. Image Process.*, 27(1):38–49, 2017. [2](#), [7](#)
- [41] 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. [4](#)
- [42] 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. [3](#)
- [43] Zhe Wu, Li Su, and Qingming Huang. Stacked cross refinement network for edge-aware salient object detection. In *Int. Conf. Comput. Vis.*, pages 7264–7273, 2019. [1](#), [7](#)
- [44] Pengxiang Yan, Guanbin Li, Yuan Xie, Zhen Li, Chuan Wang, Tianshui Chen, and Liang Lin. Semi-supervised video salient object detection using pseudo-labels. In *Int. Conf. Comput. Vis.*, pages 7284–7293, 2019. [1](#), [2](#), [5](#), [7](#)
- [45] Changqian Yu, Jingbo Wang, Changxin Gao, Gang Yu, Chunhua Shen, and Nong Sang. Context prior for scene segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12416–12425, 2020. [5](#)
- [46] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. [3](#)
- [47] 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.*, pages 12546–12555, 2020. [2](#), [3](#), [4](#), [5](#), [6](#), [7](#)
- [48] Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnnet: Edge guidance network for salient object detection. In *Int. Conf. Comput. Vis.*, pages 8779–8788, 2019. [1](#), [6](#), [7](#), [8](#)
- [49] Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *AAAI Conf. Art. Intell.*, volume 2, page 3, 2020. [3](#), [8](#)