symmetry

MDPI

Article

Symmetry Encoder-Decoder Network with Attention Mechanism for Fast Video Object Segmentation

Mingyue Guo [1], Dejun Zhang [1,2,*], Jun Sun [1] and Yiqi Wu [3]

[1] College of Information and Engineering, Sichuan Agricultural University, Yaan 625014, China
[2] School of Geography and Information Engineering, China University of Geosciences, Wuhan 430074, China
[3] College of Computer Science, China University of Geosciences, Wuhan 430074, China
[*] Correspondence: zhangdejun@cug.edu.cn

Received: 22 July 2019; Accepted: 1 August 2019; Published: 4 August 2019

Abstract: Semi-supervised video object segmentation (VOS) has obtained significant progress in recent years. The general purpose of VOS methods is to segment objects in video sequences provided with a single annotation in the first frame. However, many of the recent successful methods heavily fine-tune the object mask in the first frame, which decreases their efficiency. In this work, to address this issue, we propose a symmetry encoder-decoder network with the attention mechanism for video object segmentation (SAVOS) requiring only one forward pass to segment the target object in a video. Specifically, the encoder generates a low-resolution mask with smoothed boundaries, while the decoder further refines the details of the segmentation mask and integrates lower level features progressively. Besides, to obtain accurate segmentation results, we sequentially apply the attention module on multi-scale feature maps for refinement. We conduct several experiments on three challenging datasets (i.e., DAVIS 2016, DAVIS 2017, and SegTrack v2) to show that SAVOS achieves competitive performance against the state-of-the-art.

Keywords: video object segmentation; convolutional neural network; attention mechanism; semi-supervised; encoder-decoder

1. Introduction

In recent years, convolutional neural networks (CNN) have been successfully used in many areas of computer vision. Especially, CNN greatly promote the development of video object segmentation. Video object segmentation (VOS) is aimed at automatically segmenting the target object in video sequences. VOS has become a hot topic in recent years, which is a crucial step for many video analysis tasks, such as video summarization [1], video editing [2], and scene understanding [3]. Existing VOS approaches can be classified into two settings based on the degrees of human involvement, namely unsupervised and semi-supervised. Unsupervised methods [4–7] mainly segment the target object from the background without any annotations, e.g., the initial object mask. On the contrary, semi-supervised methods [8–12] include an initial object mask as critical visual cues of the target. However, unsupervised methods cannot handle multiple object segmentation as they are not competent to identify a specific instance.

In this work, we tackle the task of semi-supervised video object segmentation. While a wide range of learning architectures is available [8,11,13–17], the paradigm splits the video object segmentation into two main steps: train a fully-convolutional network (FCN) [18] to segment the target object firstly; next, the general network is fine-tuned based on the first frame of the video, and hundreds of iterations are performed to adapt the model to a particular video sequence.

Regardless of the high accuracies achieved by the above approaches [11,13,19], the fine-tuning process is time consuming, which makes it not adaptable for real-time applications. To fill this gap, we perform video object segmentation with a comparable accuracy level to the state-of-the-art, while our method could immediately segment video frames once obtained. Towards this goal, we present a novel approach making use of both the previous mask and the reference frame. SAVOS propagates the previous mask to the current frame and utilizes the reference frame to specify the target in the current frame.

We propose a novel network for semi-supervised video object segmentation, whose intuition is shown in Figure 1. SAVOS aims at segmenting target objects once having obtained a video sequence, requiring only one forward pass. From experimental results, SAVOS is proven to be effective and favorable against state-of-the-art methods.

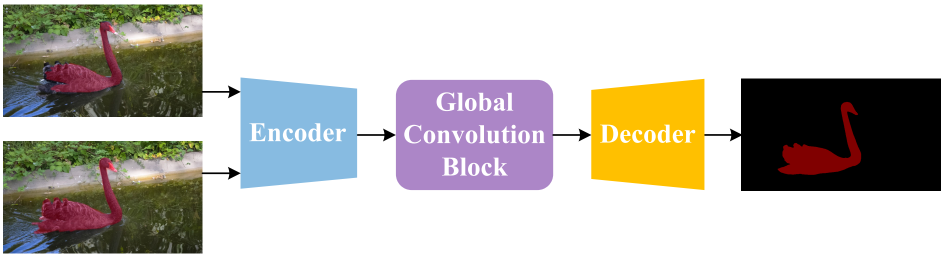


Figure 1. The sketch of symmetry encoder-decoder network with the attention mechanism for video object segmentation (SAVOS). Our model consists of three parts: encoder, global convolution block, and decoder. The model could be adapted to segment an object of an arbitrary size in a video sequence instantly.

As shown in Figure 1, our model generates the segmentation mask using a single pass of a fully-convolutional network. Specifically, ResNet-50 [20] is adopted as the backbone of the encoder. Inputs to our model contain two RGB images, each with a mask map. We designed an attention block to focus on the target object to be segmented. Consequently, SAVOS works robustly without any online learning or post-processing, resulting in great efficiency at test time. The whole pipeline is differentiable and learns in an end-to-end manner using the standard stochastic gradient descent. The experimental results showed that SAVOS outperformed previous approaches without additional fine-tuning.

The contributions of our model are summarized as follows:

- We introduce SAVOS, which requires only one forward pass through the symmetry encoder-decoder network to generate all parameters that are needed to adapt to the specific object instance.
- We design an attention module providing guidance to focus on the target object in the current frame, which helps to improve accuracy.
- Extensive experiments are conducted on three datasets, namely DAVIS 2016, DAVIS 2017, and SegTrack v2, to demonstrate that SAVOS achieves favorable performance compared to the state-of-the-art.

The remainder of this paper is organized as follows. Section 2 briefly describes the existing approaches correlated with unsupervised video object segmentation, semi-supervised video object segmentation, and the attention mechanism, which motivate this work. Section 3 illustrates the general pipeline with its shortness compared to most state-of-the-art VOS methods and further introduces the motivation of our method. Next, we present in detail the design of SAVOS with a thorough analysis of every component in Sections 4 and 5. Section 6 presents the performance of SAVOS on three public datasets with comprehensive evaluation protocols and comparisons with state-of-the-art methods. Finally, the conclusion and discussion of our future work are contained in Section 7.

2. Related Work

Due to the increasing need to process large amounts of video data automatically, many researchers have been devoted to the area of video object segmentation. Video object segmentation methods are mainly divided into two groups: unsupervised methods and semi-supervised methods. Unsupervised VOS methods do not require any manual annotations, while semi-supervised methods heavily rely on annotations for objects in the first frame. We first briefly review entirely unsupervised and entirely semi-supervised methods for completeness.

2.1. Unsupervised Video Object Segmentation

The purpose of unsupervised methods is to segment the foreground object without any annotations of the object. Several types of techniques have been presented to generate object segmentation via saliency [5,14,21], optical flow [22,23], or superpixels [24–26]. Papazoglou et al. [7] proposed a fast and unsupervised VOS method, which simply aggregates the pixels in the video to generate proposals by combining two motion boundaries extracted from optical flow. Tomakov et al. [27] presented a fully-convolutional network to learn the motion pattern in videos to segment video objects, which designed an encoder-decoder architecture to learn the rough representation of optical flow field characteristics, and then iteratively refined it to produce high-resolution motion labels. For the sake of higher-level information such as objectness, using object proposals to track object segments and produce consistent regions was adopted in [28,29]. However, these methods typically require a large amount of computational load to generate region proposals and associate thousands of segments, so these methods are only applicable to offline applications. In this work, we adopted semi-supervised tactics, which always obtain higher accuracy and efficiency than unsupervised tactics, to handle video object segmentation.

2.2. Semi-Supervised Video Object Segmentation

Semi-supervised methods are aimed at segmenting a specific object or multiple objects with the given annotated frame. Importantly, given the annotation frame, the model can obtain a good appearance initialization that unsupervised VOS methods lack. Numerous semi-supervised algorithms have been developed in the literature, including non-CNN methods and CNN-based methods. Traditional non-CNN methods mainly rely on graphical models [14,15] or object proposals [30]. In [31], a patch-based probability graph model was provided for semi-supervised VOS using a time tree structure to link patches in adjacent frames to infer pixel labels in a video accurately. Wen et al. [32] combined segmentation and multi-part tracking into a unified energy target to handle the problem of VOS, which can be solved by a random sample consensus style (RANSAC-style) approach.

Recently, the CNN-based method could achieve better accuracy than non-CNN methods. Caelles et al. [13] utilized an FCN to learn the appearance of an object and then segment the rest of the videos in parallel. Follow-up works extended CNN-based methods with diverse techniques, such as semantic instance segmentation [33,34] and online adaptation [12]. Different from all the previous approaches, our method works without any techniques like online learning or post-processing, which are time consuming and computationally expensive. Some algorithms with offline and online learning benefit from both strategies. Offline learning provides a refined mask from the estimation of the previous frame, while online learning captures the appearance information of the specific instance. Cheng et al. [8] presented a network that is end-to-end trainable, simultaneously predicting optical flow and the pixel-level object segmentation in the video. The offline pre-training learns the whole and then fine-tunes the specific object online. In [35], a method using a recurrent neural network (RNN) was proposed to fuse the results of the binary segmentation network and the bounding box of each target instance in each video frame, which can take advantage of the temporal structure of long-term video data and reject outliers.

2.3. Attention Mechanism

Recently, one of the most promising research trends has been the incorporation of attention mechanisms into deep learning frameworks, such as natural language processing [36–38], and computer vision [39–42]. In the area of segmentation, semantic segmentation and panoptic segmentation [43–46] use the attention mechanism to guide the feed-forward network for segmenting more accurately. Especially, the attention mechanism in video object segmentation helps to focus on target objects and overlook confusing background [41,47,48]. As for the attention mechanism itself, there exist different variants: hierarchical attention [49], self-attention [50], and coattention [51]. In [52], they applied channel attention to recognition tasks. In this work, we utilize a coarse-to-refine process to apply sequentially the attention module on multi-scale feature maps to focus on the target.

3. Motivation

3.1. Baselines

To date, most of the semi-supervised VOS methods have adopted the general pipeline to boost performance (see Figure 2). The pipeline contains two components: two-stage processing (offline training and online fine-tuning) and post-processing, which could significantly improve accuracy.

- *Two-stage paradigm:* A large number of CNN-based semi-supervised VOS methods adopt the two-stage paradigm (see Figure 2): firstly, a base CNN is trained to segment the target object; second, the trained network is fine-tuned based on the first frame of the test video to adapt to the object appearance, leading to the performance boost. Perazzi et al. [11] proposed a method combining offline and online learning strategies. The offline training phase feeds the coarsened previous frame mask into the trained network to predict the object mask in the current frame. Then, it further improves video segmentation quality by online fine-tuning. Caelles et al. [13] firstly trained a base CNN to segment the foreground object from the background and then used online fine-tuning to adapt to the specific object. Comparing with the aforementioned two-stage strategies, Voigtlaender et al. [12] added one more pre-training step on the PASCAL dataset in the first stage and further fine-tuned the model by online adaptation in the second stage.
- *Post-processing:* To promote accuracy, post-processing is also adopted in many VOS approaches. In [13], boundary snapping was used to snap the object mask to accurate contours, which resulted in more accurate results. Maninis et al. [34] employed two conditional classifiers for post-processing to better model different distributions, one for predicting instance foreground pixels and the other for predicting background pixels. Post-processing means such as conditional random fields (CRF) and optical flow have been proven to be helpful to further refine segmentation masks, achieving additional gains in many methods.

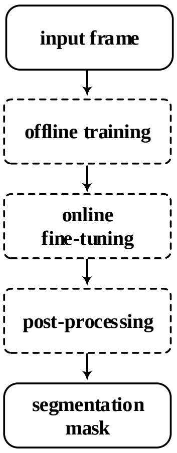


Figure 2. The general pipeline of the baseline methods to handle VOS.

3.2. Challenges and Solutions

Although both the two-stage paradigm and post-processing could lead to a boost in performance, they are time consuming and computationally expensive. For example, the algorithm in [11] needed

12 s to process a video frame, Caelles et al. [13] processed a video frame in 9 s, which are far from the time demands of real-time applications. Handling VOS problems with the two-stage paradigm and post-processing could result in a large number of parameters of models and require GPUs with large memory to train models.

The purpose of this work is to improve efficiency while ensuring the accuracy of segmentation. To speed up the process of segmenting video objects, the proposed method abandons the online fine-tuning stage. Instead, we propose a one-phase method, which only demands one forward pass through the symmetry encoder-decoder network to process each frame. Furthermore, we design an attention module to improve accuracy. The attention module emphasizes the weight of the specific object, assisting the network to learn knowledge of the target object, which will be described in the next section.

4. Network Architecture

The presented model is a symmetry encoder-decoder structure that is capable of handling four inputs and generating a segmentation mask. Figure 3 shows the architecture of SAVOS. An encoder with two symmetry branches, a global convolution block, and a decoder comprise SAVOS. The network was designed to be fully convolutional, being able to deal with arbitrary input image size and produce a sharp segmentation mask. Given a reference frame and its corresponding ground-truth mask, SAVOS aims to segment automatically the foreground object through the entire video. The key idea of SAVOS is to take advantage of both the annotated reference frame and the previous mask estimation to predict the object mask in the current frame in a deep network. The network matches the appearance of the reference frame and the current frame to detect the target object. Meanwhile, the previous mask is tracked by referencing the previous object mask in the current video frame.

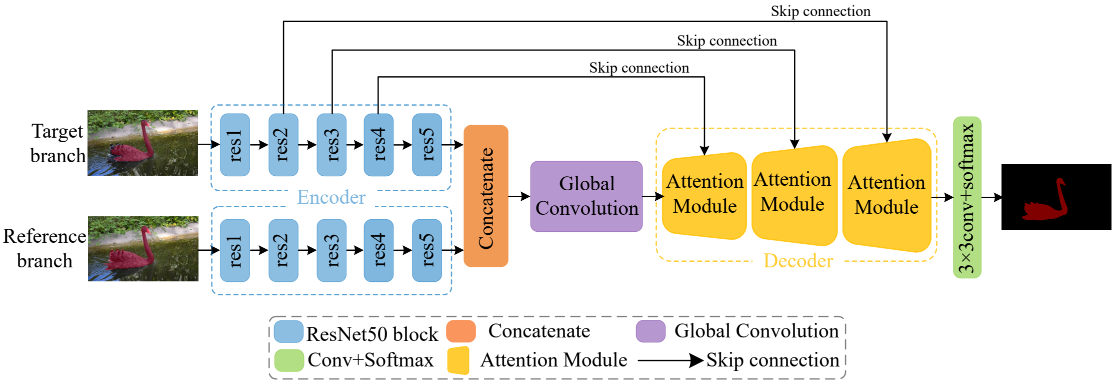


Figure 3. Overview of the network architecture of SAVOS. The whole pipeline includes an encoder consisting of the target branch and the reference branch, a global convolution block, and a decoder constituted by three attention modules. Skip connections are used to integrate low-level features with high-level features.

4.1. Symmetry Encoder

We adopted the concept of "symmetry", which emphasizes the networks that are not only the architecture of the subnetworks are identical, but the weights have to be shared among them. In this work, the encoder was designed to have two branches, which share the same architecture and the weights. In Figure 3, the encoder contains two symmetry branches: a reference branch and a target branch, between which the filter weights are shared. Inputs to the reference branch consisted of the first frame of the video (as the reference frame) and its corresponding ground-truth mask. Meanwhile, a current frame and a guidance mask corresponding to the previous frame served as inputs to the target branch. The video frame and the mask were concatenated along the channel axis and then input into the symmetry encoder. The symmetry encoder maps its two branch data into the same feature space.

ResNet50 [20] was used as the encoder backbone and modified to take a four-channel tensor as the input. We initialized the network weights from the ImageNet [53] pre-trained model and gave the newly-added filters random initialization.

4.2. Global Convolution Block

The outputs of the encoder were concatenated to be the input of the global convolution block, which was able to localize the target object accurately by matching the global feature of the reference and target branches. We used the global convolution network module in [54]. This module combines $1 \times k + k \times 1$ and $k \times 1 + 1 \times k$ convolution layers to enlarge the receptive field. Note that batch normalization [55] was removed in this work.

4.3. Decoder

The output of the global convolution block and features in the target encoder branch via skip-connections served as inputs to the decoder to produce a segmentation mask. To fuse features of different scales efficiently, we used the attention module to be the building block of our decoder. Three attention modules formulated the decoder, which was followed by two layers: a convolution layer and a softmax layer. The finally generated target mask size was 1/4 of the input image size.

The attention module was designed to assist the network to concentrate on regions of interest and learn useful features to segment the foreground object; see Figure 4. Currently, the attention mechanism could be applied to help refine intermediate feature maps in the area of segmentation. Inspired by [56], we incorporated the convolutional block attention module to obtain attention maps, which were then multiplied by the feature map for further refinement. Given a feature map, channel attention and spatial attention computed the complementary attention, the former focusing on what the target object was and the latter focusing on where the target object was. We exploited finding an efficient location to put the attention module in the network to make the best use of both channel and spatial attention with a simple design.

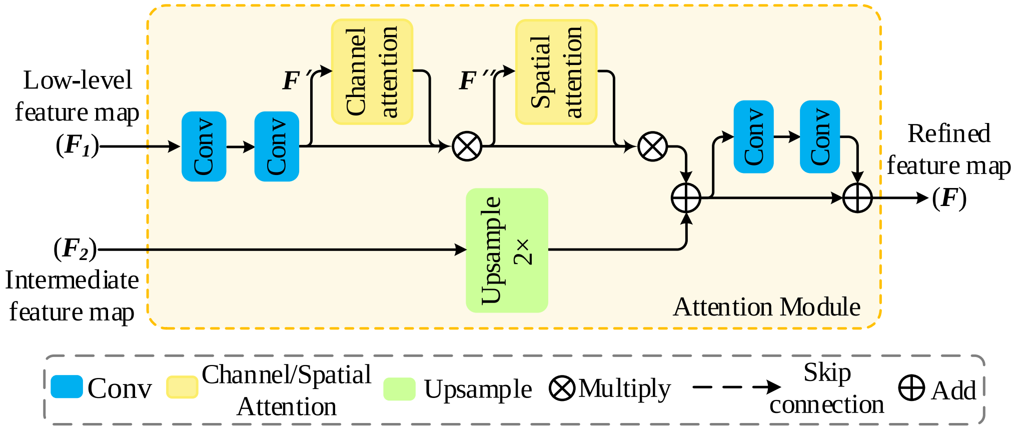


Figure 4. Illustration of the attention module. The upper branch contains the channel attention and spatial attention, which help adaptively refine the feature map. The attention module has two inputs: low-level feature map F_1 and intermediate feature map F_2, which can be obtained via skip connection and flow from the global convolution block. F' is the input of channel attention, F'' the input of spatial attention, and F the output of the attention module.

Given a low-level feature map $F_1 \in \mathbb{R}^{C_1 \times H_1 \times W_1}$ and an intermediate feature map $F_2 \in \mathbb{R}^{C_2 \times H_2 \times W_2}$, the attention module inferred an attention map $F \in \mathbb{R}^{C \times H \times W}$, which is shown in Figure 4.

The attention module exploited the inter-channel correlation of features to produce a channel attention map. The input to the channel attention F' is computed as:

$$F' = f^{3\times3}(f^{3\times3}(F_1)) \tag{1}$$

where $f^{3\times3}$ denotes the convolution operation with the filter size of 3×3.

Spatial information of a feature map was firstly aggregated by using both average pooling and max-pooling, forming two different spatial context descriptors: F^c_{avg} and F^c_{max}, which represent features obtained from the average pooling and max-pooling operations. respectively. The channel attention was computed as:

$$\begin{aligned} M_c(F') &= \sigma(MLP(AvgPool(F')) + MLP(MaxPool(F'))) \\ &= \sigma(W_1(W_0(F^c_{avg})) + W_1(W_0(F^c_{max}))) \end{aligned} \tag{2}$$

where σ represents the sigmoid function, r represents the reduction ratio, $W_0 \in \mathbb{R}^{C_1/r \times C_1}$, $W_1 \in \mathbb{R}^{C_1 \times C_1/r}$. W_0 and W_1 are shared for both inputs, and W_0 follows the ReLU activation function.

Then, the input of the spatial attention F'' is calculated as:

$$F'' = M_c(F') \otimes F' \tag{3}$$

We utilized the inter-spatial correlation of features to generate a spatial attention map. Average-pooling and max-pooling operations were first applied along the channel axis and then concatenated to produce an efficient feature descriptor.

Channel information of a feature map was aggregated by utilizing two pooling operations to form two maps: $F^s_{max} \in \mathbb{R}^{1\times H_1 \times W_1}$ and $F^s_{avg} \in \mathbb{R}^{1\times H_1 \times W_1}$. The spatial attention was computed as:

$$\begin{aligned} M_S(F'') &= \sigma(f^{7\times7}([AvgPool(F''); MaxPool(F'')])) \\ &= \sigma(f^{7\times7}([F^s_{avg}; F^s_{max}])) \end{aligned} \tag{4}$$

where σ represents the sigmoid function and $f^{7\times7}$ indicates a convolution operation with the filter size of 7×7.

5. Inference

In this work, we tackled the problem of semi-supervised VOS, which provides the ground-truth mask of the first frame. The first frame was set as the reference to estimate the masks of the remaining frames sequentially.

Single object: The probability map of the previous frame was used as the guidance mask for the current frame without binarization. To adapt to the appearance of the target object, SAVOS uses a forward pass when testing a video sequence. To promote the robustness of the target object to scale-change, we processed video frames at different scales (e.g., 0.5, 0.75, and 1) for averaging the results.

Multiple objects: To tackle multiple objects, we used the above model of a single object, but handled multiple objects at the inference time. One traditional technique is to handle each object independently and assign the label depending on the largest output probability. We used another approach to handle the scenario by exploiting the disjoint constraint of objects. This approach improved the accuracy compared to the traditional approach by setting non-maximum instance probabilities to zeros at each estimation.

However, it was still far from optimal as some useful information was lost. To this end, we utilized the softmax aggregation proposed in [57,58], which combines multiple instance probabilities softly, while constraining them to be positive and sum to one:

$$p_{i,m} = \sigma(\text{logit}(\hat{p}_{i,m})) \quad = \frac{\hat{p}_{i,m}/(1-\hat{p}_{i,m})}{\sum_{j=0}^{K} \hat{p}_{i,j}/(1-\hat{p}_{i,j})} \tag{5}$$

where σ represents the softmax function and logit represents the logit function. The output probability $\hat{p}_{i,m}$, where i indicates the pixel location and m indicates the instance, $i, m = 0$ denotes the background, and K indexes the number of instances. We used Equation (5) to aggregate the network outputs of instances at each step and pass the network outputs to the next frame.

6. Experiments

We experimentally present the performances of SAVOS on overlap similarity, running speed, and contour accuracy tasks. In addition, several details were analyzed through ablation experiments about the effectiveness of "lucid dream" data augmentation and the attention module. Section 6.1 describes the experimental settings, data augmentation, and evaluation measures. Section 6.2 reports the results and discussions on the DAVIS 2016, DAVIS 2017, and SegTrack v2 datasets. Section 6.3 provides ablative studies to analyze the efficacy of "lucid dream" data augmentation and the attention module, while Section 6.4 further verifies the generalization ability of our model with add-on studies.

6.1. Implementation Details

6.1.1. Datasets

For the evaluation, we conducted experiments on three complicated datasets: DAVIS 2016 dataset [59], DAVIS 2017 dataset [60], and SegTrack v2 dataset [61]. These three datasets provide the pixel-level ground-truth mask. More specifically, DAVIS 16 and SegTrack v2 datasets provides the binary (foreground-background) ground-truth, while DAVIS 17 provides the instance-level segmentation ground-truth. Challenges such as fast motion, occlusion, and appearance change are included in all these datasets. Hence, these datasets serve as a good platform for evaluating different video object segmentation techniques.

6.1.2. Data Augmentation

Although the DAVIS 2017 training set contains 60 videos, it was not sufficient to train our network from scratch with the pre-trained encoder backbone model. To tackle this issue, we applied the "lucid dream" strategy [10], which uses the given first frame and its annotation mask to generate training data, including five steps: illumination changing, foreground-background splitting, object motion simulating, camera view changing, and foreground-background merging. Notably, in contrast to [10], we did not produce the optical flow since our approach required no optical flow for video segmentation.

6.1.3. Implementation

We trained SAVOS using 60 video sequences in the DAVIS 2017 training set [60] and tested on the DAVIS 17 validation set. Although our model was trained on the DAVIS 2017 dataset, we found it worked well on other datasets. Therefore, we evaluated on the DAVIS 2016 and SegTrack v2 [61] validation sets using the model trained on the DAVIS 17 training set. PyTorch was used to implement the algorithm. The entire network was trained end-to-end using the stochastic gradient descent optimizer with momentum set to 0.5. The initial learning rate was set to 10^{-5} and gradually decreased over time. We conducted all training and testing on a single NVIDIA GeForce 1080 Ti GPU.

6.1.4. Evaluation Measure

The Jaccard index ($\mathcal{J}$), also known as intersection over union (IoU), is a common evaluation metric for evaluating the segmentation quality. The mean of the IoU is calculated across all frames in a sequence, so it is also referred to as mIoU.

Contour accuracy ($\mathcal{F}$) [59] is computed through a bipartite matching calculation between contour pixels of the predicted segmentation mask and contour pixels of the ground-truth segmentation mask. We computed the contour accuracy via the F1 score.

Given a segmentation mask S and the corresponding ground-truth segmentation mask S^*, the Jaccard index ($\mathcal{J}$) is calculated as:

$$\mathcal{J} = \frac{S \cap S^*}{SUS^*} \tag{6}$$

The contour accuracy $\mathcal{F}$ is computed by the F-measure of the contour-based precision P_c and recall R_c between the contour pixels of the estimated segmentation mask S and the ground-truth segmentation mask S^*, defined as:

$$\mathcal{F} = \frac{2P_cR_c}{P_c + R_c} \tag{7}$$

6.2. Comparing to the State-of-the-Art

6.2.1. DAVIS 2016 Dataset

The DAVIS 2016 dataset [56] comprises 50 sequences, 3455 annotated frames with a binary pixel-level foreground/background mask. Due to the computational complexity being a major bottleneck in video processing, sequences in the dataset were short in temporal extent ranging from two to four seconds, but included all primary challenges that are typically found in longer video sequences, such as background clutter, fast-motion, edge ambiguity, camera-shake, and out-of-view. We tested SAVOS on the 480p resolution set.

In Table 1, we compare SAVOS with state-of-the-art semi-supervised methods, i.e., OSVOS [13], OnAVOS [12], MSK [11], VPN [62], OSMN [63], and FAVOS [64], on the DAVIS 2016 validation set.

Table 1. Quantitative comparison on the DAVIS 2016 validation set. Online learning (OL) and post-processing (PP) are highlighted. We classify semi-supervised methods according to whether online learning is used. †: a variant without online learning. The rightmost column shows the approximate runtimes of the corresponding algorithms (seconds per frame).

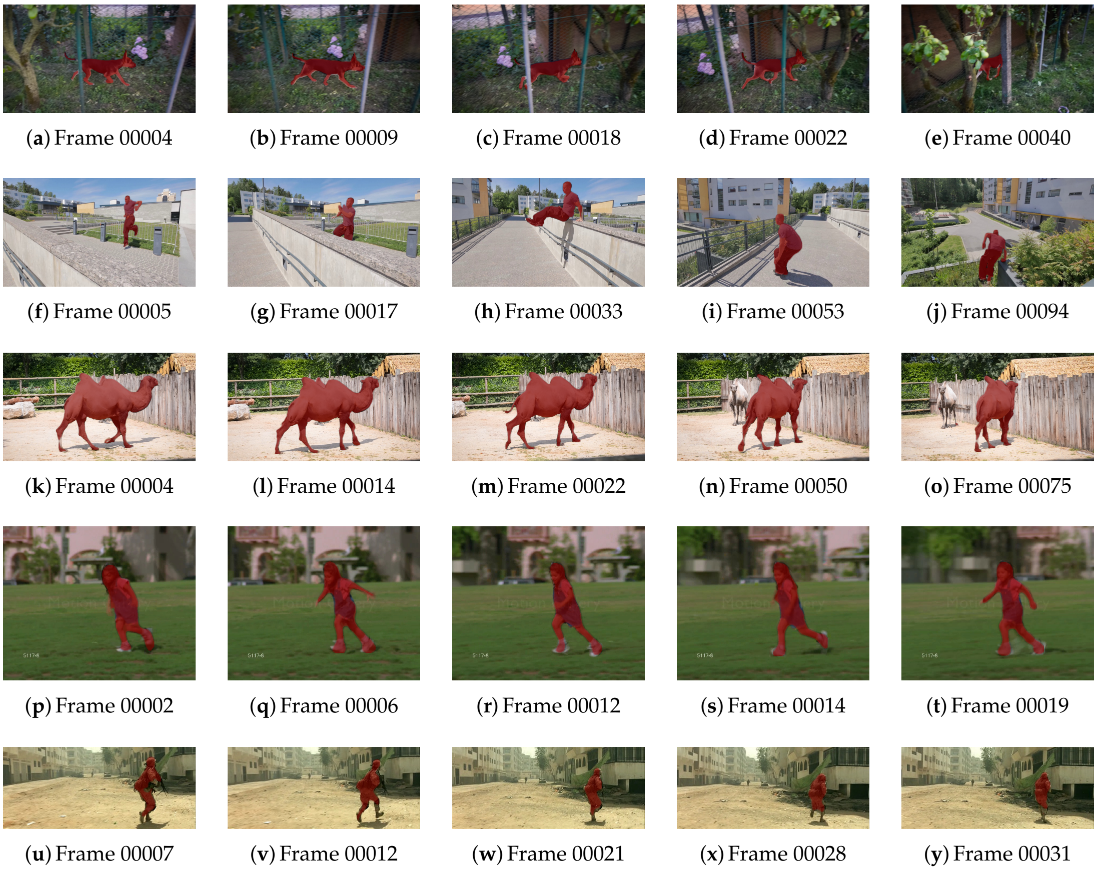
Method	OL	PP	$\mathcal{J}$ Mean	$\mathcal{F}$ Mean	Time
MSK [11]	✓	✓	79.7	75.4	12 s
OSVOS [13]	✓	✓	79.8	80.6	9 s
$OSVOS^S$ [34]	✓	✓	85.6	86.4	4.5 s
OnAVOS [12]	✓	✓	86.1	84.9	13 s
VPN [62]			70.2	65.5	0.63 s
BVS [15]			60	58.8	0.37 s
OFL [14]			68.0	63.4	120 s
OnAVOS †			72.7	-	-
Ours			80.3	79.5	0.51 s

As shown in Table 1, SAVOS obtained comparable result compared to the existing semi-supervised methods. In contrast to the methods like OSVOS [13] and MSK [11], which apply both online learning (OL) and post-processing (PP), our method outperformed them by 0.63% (80.3 vs. 79.8) and 0.75% (80.3 vs. 79.7), respectively. Moreover, with OL, the methods [12,34] need to fine-tune a general-purpose network on the first frame of each test video, which is computationally expensive and inconvenient. With PP, for example, MSK [11] uses dense CRF [24] and optical flow, and OSVOS [13] applies boundary snapping to refine the output, requiring heavy consumption of time and calculation resource. SAVOS is much more efficient and does not require online learning or post-processing in both the training and testing phase.

Compared to the methods without OL, our approach achieved significant improvement. In terms of the $\mathcal{J}$ mean, SAVOS outperformed VPN [62], BVS [15], and OnAVOS† [12] (without online learning) by 10.1, 20.3, and 7.6, respectively. Seen from Figure 5, the proposed method was capable of dealing with situations such as occlusion and confusing background in the DAVIS 2016 dataset. Meanwhile, the proposed method could also handle the challenge of fast motion well in the SegTrack v2 dataset.

In Table 1, the rightmost column displays the runtimes of OSVOS [13] and OFL [14], which were cited from [11,12], respectively. Compared with other state-of-the-art methods such as OSVOS (9 s/f) and OnAVOS (13 s/f), SAVOS ran at 0.51 s per frame on average on DAVIS 2016 dataset, which showed much faster speed. SAVOS showed outstanding efficiency against previous methods due to fast inference without additional fine-tuning, i.e., online fine-tuning and post-processing.

Among methods without online fine-tuning, SAVOS greatly outperformed almost all other methods. Compared to techniques with online learning, our method achieved comparable accuracy under the condition of no further online fine-tuning and post-processing. Figure 5 shows some qualitative visual results.



(**a**) Frame 00004 (**b**) Frame 00009 (**c**) Frame 00018 (**d**) Frame 00022 (**e**) Frame 00040

(**f**) Frame 00005 (**g**) Frame 00017 (**h**) Frame 00033 (**i**) Frame 00053 (**j**) Frame 00094

(**k**) Frame 00004 (**l**) Frame 00014 (**m**) Frame 00022 (**n**) Frame 00050 (**o**) Frame 00075

(**p**) Frame 00002 (**q**) Frame 00006 (**r**) Frame 00012 (**s**) Frame 00014 (**t**) Frame 00019

(**u**) Frame 00007 (**v**) Frame 00012 (**w**) Frame 00021 (**x**) Frame 00028 (**y**) Frame 00031

Figure 5. The qualitative segmentation results of our method on the DAVIS 2016 (first three rows: libby, parkour, camel) and SegTrack v2 (fourth row: girl) datasets. The pixel-level output is indicated by the red mask. The results show that our method is able to segment the object under several challenges, such as occlusions, camera view change, and confusing background. The last row shows unsatisfactory results on SegTrack v2.

6.2.2. DAVIS 2017

We evaluated the proposed method on the DAVIS 2017 validation set [60], which consisted of 30 video sequences with various challenging cases including multiple objects with similar appearance, heavy occlusion, large appearance variation, clutter background, etc. The mean of region similarity $\mathcal{J}$ and the mean of contour accuracy $\mathcal{F}$ were used to evaluate the performance in Table 2.

Table 2. Comparisons of SAVOS and the other four state-of-the-art algorithms on the DAVIS 17 validation set. Segmentation results show that SAVOS achieves comparable performance.

Method	**$\mathcal{J}$ Mean**	**$\mathcal{F}$ Mean**
OSVOS [13]	52.1	-
OnAVOS [12]	61	66.1
FAVOS [64]	45.1	55.4
RGMP [58]	64.8	68.6
OSMN [63]	52.5	57.1
Ours	62.1	63.5

SAVOS performed favorably against most of the semi-supervised methods, e.g., OSVOS [13], OnAVOS [12], FAVOS [64], and OSMN [63], with a 62.1 mean Jaccard index $\mathcal{J}$ and a 63.5 mean contour accuracy $\mathcal{F}$. Even compared to methods with fine-tuning (i.e., OSVOS, OnAVOS), SAVOS still achieved better performance by a non-negligible margin while being faster.

6.2.3. SegTrack v2

SegTrack-v2 [61] contains 14 video sequences with 24 objects and 947 frames providing pixel-level annotations. Under the condition of segmenting multiple objects, each target object segmentation is treated as an independent problem with provided instance-level annotations.

We evaluated SAVOS on SegTrack v2 [61] using the exactly same model and parameters as in the DAVIS experiments to estimate object masks. In Figure 6, our method shows competitive performance with state-of-the-art methods that apply fine-tuning. Note that our network trained on the training set of the DAVIS 2017 dataset did not see the SegTrack v2 dataset. Some qualitative results could be seen from Figure 5. Hence, this experiment demonstrated the generalization performance of SAVOS.

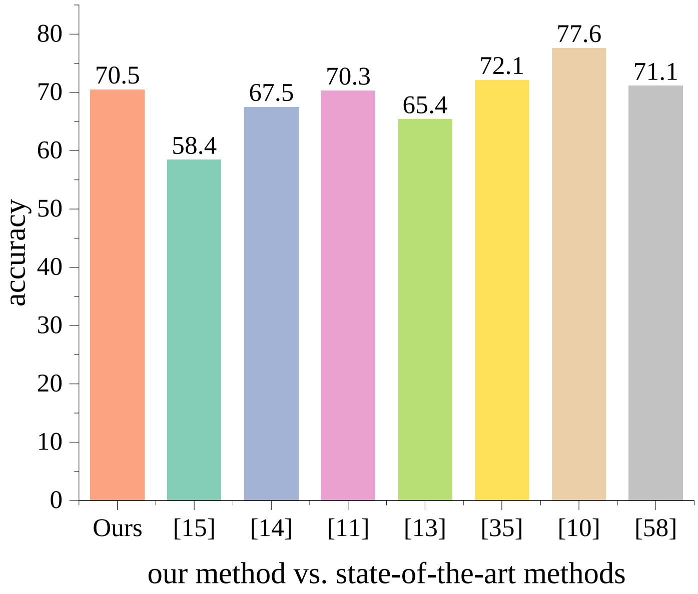


Figure 6. Segmentation results of SAVOS on the SegTrack v2 dataset. Compared to related state-of-the-arts, SAVOS provides favorable results.

6.3. Ablation Study

To understand the proposed method comprehensively, we conducted several ablation experiments. Specifically, we constructed two variants and evaluated them on the DAVIS 2016 validation set to validate the efficacy of different components in SAVOS, as shown in Table 3. In order to make a fair comparison, we set the same parameters except for the specific declaration.

Table 3. Effectiveness of various components in SAVOS. All variants are evaluated on the DAVIS 2016 dataset. Results below show that both "lucid dream" augmentation and the attention module contribute to the performance of SAVOS.

Method	**Component**		$\mathcal{J}$ **Mean**	$\mathcal{F}$ B
	Lucid Dream	**Attention Module**		
Basic version			78.3	77.5
Variant 1	✓		79.2	77.8
Variant 2	✓	✓	80.3	79.5

6.3.1. Lucid Dream Augmentation

To demonstrate the effect of the "lucid dream" augmentation, we removed it from our network. As shown in Table 3, we found that the region similarity was reduced from 79.2 to 78.3. This decline demonstrated that the "lucid dream" data augmentation was useful to improve the performance.

6.3.2. Attention Module

For validation on the efficacy of the attention module, we constructed an algorithm by further removing the attention block (see the third column in Table 3). In this way, the object region was not specifically concentrated by the network. The fourth and fifth rows in Table 3 demonstrate that the attention module was critical to the performance boost. The main reason was that the attention module was gradually applied on multi-scale features maps, enforcing the network to focus on the object region to generate more accurate results.

6.4. Add-On Study

In this section, additional components are added to SAVOS to investigate how these components further improve the performance. We conducted this add-on study on the DAVIS 2016 validation set, and the results can be seen in Table 4.

Table 4. Add-on study on DAVIS 2016. Models with additional components, i.e., online-learning and CRF, are compared with each other. In the last row, corresponding additional time per frame is provided.

	Our	**+OL**	**+CRF**
$\mathcal{J}$ Mean	80.3	81.0	80.8
$\mathcal{F}$ Mean	79.5	79.8	79.3
time	0.51 s	+1.81 s	+2.71 s

6.4.1. Online Learning

We fine-tuned SAVOS on the first frame of a test video, like previous online learning methods, to assist the network to learn the appearance of the target object. We used "lucid dream" data augmentation to generate inputs to the reference branch and the target branch from a single image by applying sequential transformations. The improvement ($\mathcal{J}$: 80.3 to 81.0) was relatively small compared with previous methods. This result implies that our method could almost achieve the same performance of online fine-tuning methods without the considerable computational overhead of online learning.

6.4.2. CRF Refinement

We used the dense CRF [24], which helps to better localize the object contours, as the post-processing to refine segmentation mask. We applied the same method in [58] to find the hyper-parameter for dense CRF.

From Table 4, we observe that the dense CRF benefited the refining mask boundaries to be aligned with targets and increased the overall overlapping area ($\mathcal{J}$:80.3 to 80.8), but hurt the fine details ($\mathcal{F}$:79.5 to 79.3), where $\mathcal{F}$ was very sensitive. The backbone architecture in our network could better handle fine details than previous backbones (e.g., AlexNet, LeNet), not to mention the attention module used in the decoder, which was able to recover fine details without additional post-processing.

7. Conclusions

In this work, we presented a symmetry encoder-decoder architecture with the attention module for video object segmentation (SAVOS). Without online learning and post-processing, our network achieved favorable performance, requiring one pass forward, which made it much faster than comparable methods. In addition, to obtain accurate segmentation results, a coarse-to-fine process was applied on multi-scale feature maps to refine the prediction. Extensive experimental results on three challenging datasets, i.e., DAVIS 2016, DAVIS 2017, and SegTrack v2, demonstrated that the proposed method achieved competitive performance.

There are several future directions for our method to improve. In our experiments, we found that using the same model as in the DAVIS experiments on the SegTrack v2 dataset sometimes could get unsatisfactory masks on specific categories, like soldier in Figure 5. It is supposed that edge ambiguity may cause such unsatisfactory segmentation results. Hence, we could further improve the generalization performance of SAVOS in different datasets. Next, optical flow techniques could be incorporated into SAVOS to exploit temporal coherence between adjacent frames for further performance boost.

Author Contributions: M.G. conceived of and designed the algorithm and the experiments. M.G. and D.Z. analyzed the data. M.G. wrote the manuscript. D.Z. supervised the research. Y.W. and J.S. provided suggestions for the proposed method and its evaluation and assisted in the preparation of the manuscript. All authors approved the final manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grants 61702350 and 61802355.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ji, Z.; Xiong, K.; Pang, Y.; Li, X. Video Summarization with Attention-Based Encoder-Decoder Networks. *IEEE Trans. Circuits Syst. Video Technol.* **2019**. [CrossRef]
2. Bakkay, M.C.; Pizenberg, M.; Carlier, A. Protocols and software for simplified educational video capture and editing. *J. Comput. Educ.* **2019**, *6*, 257–276. [CrossRef]
3. Pham, Q.H.; Nguyen, T.; Hua, B.S.; Roig, G.; Yeung, S.K. JSIS3D: Joint Semantic-Instance Segmentation of 3D Point Clouds with Multi-Task Pointwise Networks and Multi-Value Conditional Random Fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
4. Faktor, A.; Irani, M. Video Segmentation by Non-Local Consensus voting. In Proceedings of the British Machine Vision Conference, Nottingham, UK, 1–5 September 2014.
5. Jain, S.D.; Xiong, B.; Grauman, K. Fusionseg: Learning to Combine Motion and Appearance for Fully Automatic Segmention of Generic Objects in Videos. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; Volume 1.

6. Keuper, M.; Andres, B.; Brox, T. Motion Trajectory Segmentation via Minimum Cost Multicuts. In Proceedings of the the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
7. Papazoglou, A.; Ferrari, V. Fast Object Segmentation in Unconstrained Video. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Sydney, NSW, Australia, 1–8 December 2013.
8. Cheng, J.; Tsai, Y.H.; Wang, S.; Yang, M.H. SegFlow: Joint Learning for Video Object Segmentation and Optical Flow. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 686–695.
9. Jang, W.; Kim, C. Online Video Object Segmentation via Convolutional Trident Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7474–7483.
10. Khoreva, A.; Benenson, R.; Ilg, E.; Brox, T.; Schiele, B. Lucid Data Dreaming for Video Object Segmentation. *Int. J. Comput. Vis.* **2019**, *127*, 1175–1197. [CrossRef]
11. Perazzi, F.; Khoreva, A.; Benenson, R.; Schiele, B.; Sorkine-Hornung, A. Learning Video Object Segmentation from Static Images. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 3491–3500.
12. Voigtlaender, P.; Leibe, B. Online Adaptation of Convolutional Neural Networks for Video Object Segmentation. *arXiv* **2017**, arXiv:1706.09364.
13. Caelles, S.; Maninis, K.K.; Pont-Tuset, J.; Leal-Taixé, L.; Cremers, D.; Van Gool, L. One-Shot Video Object Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
14. Tsai, Y.H.; Yang, M.H.; Black, M.J. Video Segmentation via Object Flow. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 3899–3908.
15. Märki, N.; Perazzi, F.; Wang, O.; Sorkine-Hornung, A. Bilateral Space Video Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 743–751.
16. Yoon, J.S.; Rameau, F.; Kim, J.; Lee, S.; Shin, S.; Kweon, I.S. Pixel-Level Matching for Video Object Segmentation Using Convolutional Neural Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2186–2195.
17. Zhang, D.; Luo, M.; He, F. Reconstructed Similarity for Faster GANs-Based Word Translation to Mitigate Hubness. *Neurocomputing* **2019**. [CrossRef]
18. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
19. Hu, P.; Wang, G.; Kong, X.; Kuen, J.; Tan, Y.P. Motion-Guided Cascaded Refinement Network for Video Object Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1400–1409.
20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
21. Ji, S.; Xu, W.; Yang, M.W.; Yu, K. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *35*, 221–231. [CrossRef] [PubMed]
22. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected Crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef] [PubMed]
23. Strand, R.; Ciesielski, K.; Malmberg, F.; Saha, P.K. The minimum barrier distance. *Comput. Vis. Image Underst.* **2013**, *117*, 429–437. [CrossRef]
24. Krähenbühl, P.; Koltun, V. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In *Advances in Neural Information Processing Systems 24*; Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2011; pp. 109–117.

25. Dosovitskiy, A.; Fischer, P.; Ilg, E.; Häusser, P.; Hazirbas, C.; Golkov, V.; Smagt, P.; Cremers, D.; Brox, T. FlowNet: Learning Optical Flow with Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 2758–2766.
26. Vijayanarasimhan, S.; Grauman, K. Active Frame Selection for Label Propagation in Videos. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012.
27. Tokmakov, P.; Alahari, K.; Schmid, C. Learning Motion Patterns in Videos. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 531–539.
28. Fragkiadaki, K.; Arbeláez, P.A.; Felsen, P.; Malik, J. Learning to segment moving objects in videos. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4083–4090.
29. Yu, G.; Yuan, J. Fast action proposals for human action detection and search. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1302–1311.
30. Perazzi, F.; Wang, O.; Gross, M.; Sorkine-Hornung, A. Fully Connected Object Proposals for Video Segmentation. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 3227–3234.
31. Badrinarayanan, V.; Budvytis, I.; Cipolla, R.; Member, S. Semi-Supervised Video Segmentation Using Tree Structured Graphical Models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2751–2764 [CrossRef] [PubMed]
32. Wen, L.; Du, D.; Lei, Z.; Li, S.Z.; Yang, M.H. JOTS: Joint Online Tracking and Segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 2226–2234.
33. Caelles, S.; Chen, Y.; Pont-Tuset, J.; Gool, L.V. Semantically-Guided Video Object Segmentation. *arXiv* **2017**, arXiv:1704.01926..
34. Maninis, K.K.; Caelles, S.; Chen, Y.; Pont-Tuset, J.; Leal-Taixé, L.; Cremers, D.; Gool, L.V. Video Object Segmentation without Temporal Information. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 1515–1530. [CrossRef] [PubMed]
35. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
36. Kumar, A.; Irsoy, O.; Ondruska, P.; Iyyer, M.; Bradbury, J.; Gulrajani, I.; Zhong, V.; Paulus, R.; Socher, R. Ask Me Anything: Dynamic Memory Networks for Natural Language Processing. In Proceedings of the 33rd International Conference on Machine Learning (PMLR), New York, NY, USA, 19–24 June 2016; pp. 1378–1387.
37. Choi, H.; Cho, K.; Bengio, Y. Fine-grained attention mechanism for neural machine translation. *Neurocomputing* **2018**, *284*, 171–176. [CrossRef]
38. Zhang, B.; Xiong, D.; Su, J. Neural Machine Translation with Deep Attention. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**. [CrossRef] [PubMed]
39. Guo, H.; Zheng, K.; Fan, X.; Yu, H.; Wang, S. Visual Attention Consistency Under Image Transforms for Multi-Label Image Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
40. Dai, T.; Cai, J.; Zhang, Y.; Xia, S.T.; Zhang, L. Second-Order Attention Network for Single Image Super-Resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
41. Wang, L.; Wang, Y.; Liang, Z.; Lin, Z.; Yang, J.; An, W.; Guo, Y. Learning Parallax Attention for Stereo Image Super-Resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
42. Si, C.; Chen, W.; Wang, W.; Wang, L.; Tan, T. An Attention Enhanced Graph Convolutional LSTM Network for Skeleton-Based Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
43. Zhang, T.; Lin, G.; Cai, J.; Shen, T.; Shen, C.; Kot, A.C. Decoupled Spatial Neural Attention for Weakly Supervised Semantic Segmentation. *IEEE Trans. Multimed.* **2019**. [CrossRef]

44. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
45. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. Learning a Discriminative Feature Network for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
46. Li, Y.; Chen, X.; Zhu, Z.; Xie, L.; Huang, G.; Du, D.; Wang, X. Attention-Guided Unified Network for Panoptic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
47. Li, X.; Change Loy, C. Video Object Segmentation with Joint Re-identification and Attention-Aware Mask Propagation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
48. Lu, X.; Wang, W.; Ma, C.; Shen, J.; Shao, L.; Porikli, F. See More, Know More: Unsupervised Video Object Segmentation With Co-Attention Siamese Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
49. Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; Hovy, E. Hierarchical Attention Networks for Document Classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 1480–1489.
50. Cheng, J.; Dong, L.; Lapata, M. Long Short-Term Memory-Networks for Machine Reading. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 551–561.
51. Xiong, C.; Zhong, V.; Socher, R. Dynamic Coattention Networks for Question Answering. *arXiv* **2017**, arXiv:1611.01604.
52. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
53. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Li, F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
54. Peng, C.; Zhang, X.; Yu, G.; Luo, G.; Sun, J. Large Kernel Matters—Improve Semantic Segmentation by Global Convolutional Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1743–1751.
55. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning (PMLR), Lille, France, 6–11 July 2015; p. 9.
56. Xu, K.; Wen, L.; Li, G.; Bo, L.; Huang, Q. Spatiotemporal CNN for Video Object Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
57. Arribas, J.I.; Cid-Sueiro, J.; Adali, T.; Figueiras-Vidal, A.R. Neural architectures for parametric estimation of a posteriori probabilities by constrained conditional density functions. In Proceedings of the Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No. 98TH8468), Madison, WI, USA, 25–25 August 1999; pp. 263–272.
58. Oh, S.W.; Lee, J.Y.; Sunkavalli, K.; Kim, S.J. Fast Video Object Segmentation by Reference-Guided Mask Propagation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7376–7385.
59. Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Gool, L.V.; Gross, M.; Sorkine-Hornung, A. A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 724–732.
60. Pont-Tuset, J.; Perazzi, F.; Caelles, S.; Arbeláez, P.; Sorkine-Hornung, A.; Van Gool, L. The 2017 DAVIS Challenge on Video Object Segmentation. *arXiv* **2017**, arXiv:1704.00675.
61. Li, F.; Kim, T.; Humayun, A.; Tsai, D.; Rehg, J.M. Video Segmentation by Tracking Many Figure-Ground Segments. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2192–2199.

62. Jampani, V.; Gadde, R.; Gehler, P.V. Video Propagation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
63. Yang, L.; Wang, Y.; Xiong, X.; Yang, J.; Katsaggelos, A.K. Efficient Video Object Segmentation via Network Modulation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6499–6507.
64. Cheng, J.; Tsai, Y.H.; Hung, W.C.; Wang, S.; Yang, M.H. Fast and Accurate Online Video Object Segmentation via Tracking Parts. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7415–7424.