See More, Know More: Unsupervised Video Object Segmentation with Co-Attention Siamese Networks

Xiankai Lu[1*], Wenguan Wang[1*], Chao Ma[2], Jianbing Shen[1†], Ling Shao[1], Fatih Porikli[3]

[1] Inception Institute of Artificial Intelligence, UAE

[2] MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, China

[3] Australian National University, Australia

carrierlxk@gmail.com wenguanwang.ai@gmail.com chaoma@sjtu.edu.cn

shenjianbingcg@gmail.com ling.shao@ieee.org fatih.porikli@anu.edu.au

https://github.com/carrierlxk/COSNet

Abstract

We introduce a novel network, called CO-attention Siamese Network (COSNet), to address the unsupervised video object segmentation task from a holistic view. We emphasize the importance of inherent correlation among video frames and incorporate a global co-attention mechanism to improve further the state-of-the-art deep learning based solutions that primarily focus on learning discriminative foreground representations over appearance and motion in short-term temporal segments. The co-attention layers in our network provide efficient and competent stages for capturing global correlations and scene context by jointly computing and appending co-attention responses into a joint feature space. We train COSNet with pairs of video frames, which naturally augments training data and allows increased learning capacity. During the segmentation stage, the co-attention model encodes useful information by processing multiple reference frames together, which is leveraged to infer the frequently reappearing and salient foreground objects better. We propose a unified and end-to-end trainable framework where different co-attention variants can be derived for mining the rich context within videos. Our extensive experiments over three large benchmarks manifest that COSNet outperforms the current alternatives by a large margin.

1. Introduction

Unsupervised video object segmentation (UVOS) aims to automatically separate primary foreground object(s) from their background in a video. Since UVOS does not require manual interaction, it has significant value in both academic

---

*The first two authors contribute equally to this work.
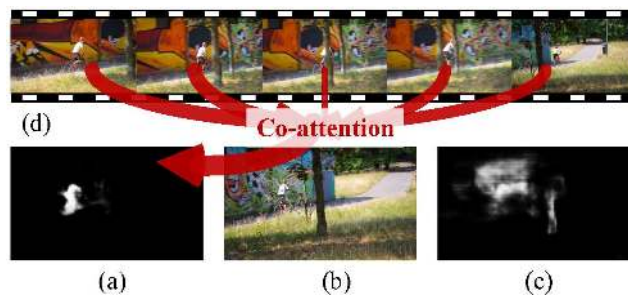
†Corresponding author: *Jianbing Shen.*

Figure 1. Illustration of our intuition. Given an input frame (b), our method leverages information from multiple reference frames (d) to better determine the foreground object (a), through a co-attention mechanism. (c) An inferior result without co-attention.

and applied fields, especially in this era of information-explosion. However, due to the lack of prior knowledge about the primary object(s), in addition to the typical challenges for semi-supervised video object segmentation (*e.g.*, object deformation, occlusion, and background clutters), UVOS suffers from another difficult problem, *i.e.*, how to correctly distinguish the primary objects from a complex and diverse background.

We argue that the primary objects in UVOS settings should be the most (**i**) distinguishable in an individual frame (locally salient), and (**ii**) frequently appearing throughout the video sequence (globally consistent). These two properties are essential for determining the primary objects. For instance, by only glimpsing a short video clip as illustrated in Fig. 1(b), it is hard to determine the primary objects. Instead, if we view the entire video (or a sufficiently long sequence) as in Fig. 1(d), the foreground can be easily discovered. Although primary objects tend to be highly correlated at a macro level (entire video), they often exhibit different appearances at a micro level (shorter video clips) due to articulated body motions, occlusions, out-of-view movements, camera movements, and environment varia-

tions. Clearly, micro level variations are the major sources of challenges in video segmentation. Thus, it is desirable to take advantage of the global consistency property and leverage the information from other frames.

By considering UVOS from a global perspective, we can help to locate primary objects and alleviate the local ambiguities. This notion also motivated the earlier heuristic models for UVOS [14], yet it is largely ignored by current deep learning based models.

Current deep learning based UVOS models typically focus on the intra-frame discrimination property of primary objects in appearance or motion, while ignoring the valuable global-occurrence consistency across multiple frames. These methods compute optical flows across a few consecutive frames [53, 24, 9, 32, 33], which is limited to a local receptive window in the temporal domain. Although recurrent neural networks (RNNs) [49] are introduced to memorize previous frames, this sequential processing strategy may fail to explicitly explore the rich relations between different frames, hence does not attain a global perspective.

With these insights, we reformulate the UVOS task as a co-attention procedure and propose a novel CO-attention Siamese Network (COSNet) to model UVOS from a global perspective. Specifically, during the training phase, COSNet takes a pair of frames from the same video as input and learns to capture their rich correlations. This is achieved by a differentiable, gated co-attention mechanism, which enables the network to attend more to the correlated, informative regions, and produce further discriminative foreground features. For a testing frame (Fig. 1(b)), COSNet is able to produce more accurate results (Fig. 1(a)) from a global view, *i.e.*, utilize the correlations between the testing frame and multiple reference frames. Fig. 1(c) shows the inferior result when considering only the information from the testing frame (Fig. 1(b)).

Another advantage of our COSNet is that it is remarkably efficient for augmenting training data, as it allows using a large number of arbitrary frame pairs within the same video. Additionally, as we explicitly model the relations between video frames, the proposed model does not need to compute optical flow, which is time-consuming and computationally expensive. Finally, the COSNet offers a unified, end-to-end trainable framework that efficiently mines rich contextual information within video sequences. We implement different co-attention mechanisms such as vanilla co-attention, symmetric co-attention, and channel-wise co-attention, which offers a more insightful glimpse into the task of UVOS. We quantitatively demonstrate that our co-attention mechanism is able to bring large improvement in performance, which confirms its effectiveness and the value of global information for UVOS. The proposed COSNet shows superior performance over the current state-of-the-art methods across three popular benchmarks: DAVIS16 [45],

FBMS [41] and Youtube-Objects [47].

2. Related Work

We start by providing an overview of representative work on video object segmentation (§2.1), followed by a brief overview of differentiable neural attention (§2.2).

2.1. Video Object Segmentation

According to its supervision type, video object segmentation can be broadly categorized into *unsupervised* (UVOS) and *semi-supervised* video object segmentation. In this paper, we focus on the UVOS task, which extracts primary object(s) without manual annotation.

Early **UVOS models** typically analyzed long-term motion information (trajectories) [4, 40, 17, 42, 28, 41], leveraged object proposals [31, 37, 70, 30, 18, 27] or utilized saliency information [60, 14, 55, 21], to infer the target. Later, inspired by the success of deep learning, several methods [16, 54, 43] began to approach UVOS using deep learning features. These were typically limited due to their lack of end-to-end learning ability [54] and use of heavyweight fully-connected network architectures [16, 43]. Recently, more research efforts have focused on the fully convolutional neural network based UVOS models. For example, Tokmakov *et al*. [52] proposed to separate independent object and camera motion using a learnable motion pattern network [52]. Li *et al*. learned an instance embedding network [32] from static images to better locate the object(s), and later they combined motion-based bilateral networks for identifying the background [33]. Two-stream fully convolution networks are also a popular choice [9, 24, 53, 32] to fuse motion and appearance information together for object inference. An alternative way to segment an object is through video salient object detection [49]. This method fine-tunes the pre-trained semantic segmentation network for extracting spatial saliency features, then trains ConvLSTM to capture temporal dynamics.

These deep UVOS models generally achieved promising results, which demonstrates well the advantages of applying neural networks to this task. However, they only consider the sequential nature of UVOS and short-term temporal information, lacking a global view and comprehensive use of the rich, inherent correlation information within videos.

For **SVOS methods**, the target object(s) is provided in the first frame and tracked automatically [60, 8, 5, 68, 2, 69, 64, 71] or interactively by users [1] in the subsequent frames. Numerous algorithms were proposed based on graphical models [54], object proposals [46], supertrajectories [61], *etc*. Recently, deep learning based methods achieved promising results. Some algorithms treated video object segmentation as a static segmentation task without using any temporal information [44], built a deep
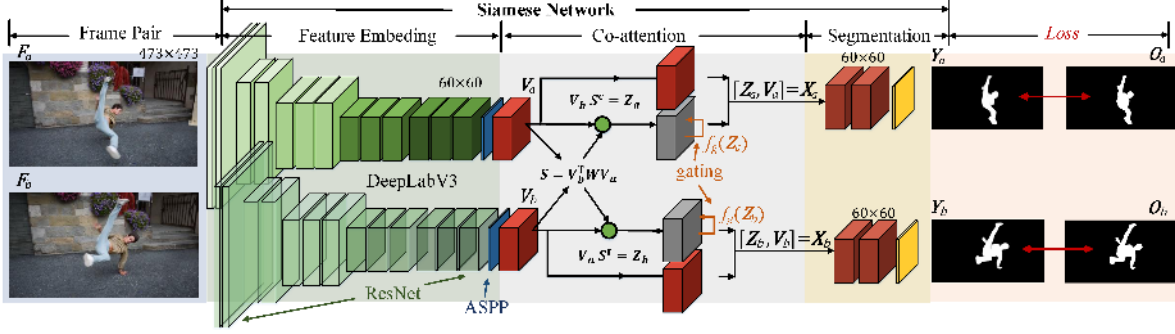
Figure 2. Overview of COSNet in the training phase. A pair of frames $\{\mathbf{F}_a, \mathbf{F}_b\}$ is fed into a feature embedding module to obtain the feature representations $\{\mathbf{V}_a, \mathbf{V}_b\}$. Then, the co-attention module computes the attention summaries that encode the correlations between $\mathbf{V}_a$ and $\mathbf{V}_b$. Finally, $\mathbf{Z}$ and $\mathbf{V}$ are concatenated and handed over to a segmentation module to produce segmentation predictions.

one-shot learning framework [5, 59], or used a mask-propagation network [25]. In addition, both object tracking [29, 8, 12, 36] and person re-identification [34, 66] have been fused into SVOS task to handle deformation and occlusion issues. Hu *et al.* [22] proposed a Siamese network based SVOS model. Compared with our COSNet, the differences are distinct, rather than their dissimilar supervision manners. First, since [22] was proposed based on image matching strategy, they used a Siamese network to propagate the first-frame annotation to the subsequent frames. Our method substantially differs in that we learn the Siamese network to capture rich and global correspondences within videos to further assist automatic primary object discovery and segmentation. Second, we provide the first approach that uses a co-attention scheme to facilitate correspondence learning for video object segmentation.

2.2. Attention Mechanisms in Neural Networks

Differentiable attentions, which are inspired by human perception [13, 58], have been widely studied in deep neural networks [26, 56, 38, 23, 57, 62, 15]. With end-to-end training, neural attention allows networks to selectively pay attention to a subset of inputs. For example, Chu *et al.* [11] exploited multi-context attention for human pose estimation. In [7], spatial and channel-wise attention were proposed to dynamically select an image part for captioning.

More recently, co-attention mechanisms have been studied in vision-and-language tasks, such as visual question answering [35, 65, 63, 39] and visual dialogue [63]. In these works, co-attention mechanisms were used to mine the underlying correlations between different modalities. For example, Lu *et al.* [35] created a model that jointly performs question-guided visual attention and image-guided question attention. In this way, the learned model can selectively focus on image regions and segments of documents. Our co-attention model is inspired by these works, but it is used to capture the coherence across different frames with a more elegant network architecture.

3. Proposed Algorithm

Our COSNet formulates UVOS as a co-attention procedure. A co-attention module learns to explicitly encode correlations between video frames. This enables COSNet to attend to the frequently coherent regions, thus further helping to discover the foreground object(s) and produce reasonable UVOS results. Specifically, during training, co-attention procedure can be decomposed into the correlation learning between any frame pairs from the same video (see Fig. 2). During testing, COSNet infers the primary target with a global view, *i.e.*, takes advantage of the co-attention information between the testing frame and multiple reference frames. We will elaborate the co-attention mechanisms in COSNet in §3.1, and detail the whole architecture of COSNet in §3.2. In §3.3, we will provide more implementation details.

3.1. Co-attention Mechanisms in COSNet

Vanilla co-attention. As shown in Fig. 2, given two video frames $\mathbf{F}_a$ and $\mathbf{F}_b$ from the same video, $\mathbf{V}_a \in \mathbb{R}^{W \times H \times C}$ and $\mathbf{V}_b \in \mathbb{R}^{W \times H \times C}$ denote the corresponding feature representations from a feature embedding network. $\mathbf{V}_a$ and $\mathbf{V}_b$ are 3D-tensors with the width W, height H and C channels. We leverage the co-attention mechanism [65, 35] to mine the correlations between $\mathbf{F}_a$ and $\mathbf{F}_b$ in their feature embedding space. More specifically, we first compute the affinity matrix $\mathbf{S}$ between $\mathbf{V}_a$ and $\mathbf{V}_b$:

$$\mathbf{S} = \mathbf{V}_b^\top \mathbf{W} \mathbf{V}_a \in \mathbb{R}^{(WH) \times (WH)}, \quad (1)$$

where $\mathbf{W} \in \mathbb{R}^{C \times C}$ is a weight matrix. Here $\mathbf{V}_a \in \mathbb{R}^{C \times (WH)}$ and $\mathbf{V}_b \in \mathbb{R}^{C \times (WH)}$ are flattened into matrix representations. Each column $\mathbf{V}_a^{(i)}$ in $\mathbf{V}_a$ represents the feature vector at position $i \in \{1, ..., WH\}$ with C dimensions. As a result, each entry of $\mathbf{S}$ reflects the similarity between each row of $\mathbf{V}_b^\top$ and each column of $\mathbf{V}_a$. Since the weight matrix $\mathbf{W}$ is a square matrix, the diagonalization of $\mathbf{W}$ can be represented as follows:

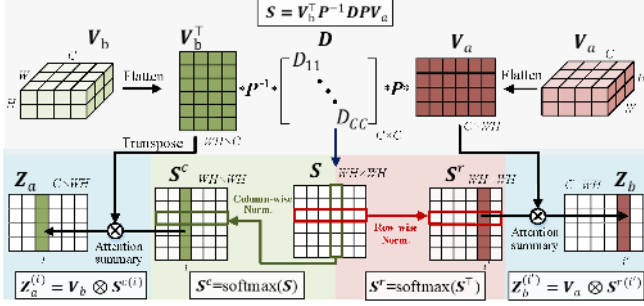$$\mathbf{W} = \mathbf{P}^{-1} \mathbf{D} \mathbf{P}, \quad (2)$$

Figure 3. Illustration of our co-attention operation.

where $\mathbf{P}$ is an invertible matrix and $\mathbf{D}$ is a diagonal matrix. Then, as shown in the gray area in Fig. 3, Eq. 1 can be re-written as:

$$\mathbf{S} = \mathbf{V}_b^{\top}\mathbf{P}^{-1}\mathbf{D}\mathbf{P}\mathbf{V}_a. \qquad (3)$$

Through the *vanilla co-attention* in Eq. 3, the feature representation of each frame first undergoes linear transformations, and then calculates the distance between any locations of themselves.

Symmetric co-attention. If we further constrain the weight matrix to be a symmetric matrix, the project matrix $\mathbf{P}$ becomes an orthogonal matrix: $\mathbf{P}^{\top}\mathbf{P} = I$, where I is a $C{\times}C$ identity matrix. A *symmetric co-attention* can be derived from Eq. 3:

$$\mathbf{S} = \mathbf{V}_b^{\top}\mathbf{P}^{\top}\mathbf{D}\mathbf{P}\mathbf{V}_a = (\mathbf{P}\mathbf{V}_b)^{\top}\mathbf{D}\mathbf{P}\mathbf{V}_a. \qquad (4)$$

Eq. 4 indicates that we project the feature embeddings $\mathbf{V}_a$ and $\mathbf{V}_b$ into an orthogonal common space and maintain their norm of $\mathbf{V}_a$ and $\mathbf{V}_b$. This property has proved valuable for eliminating the correlation between different channels (*i.e.*, C- dimension) [50] and improving the network's generalization ability [3, 48].

Channel-wise co-attention. Furthermore, the project matrix $\mathbf{P}$ can be simplified into an identity matrix I (*i.e.*, without space transformation), and then the weight matrix $\mathbf{W}$ becomes a diagonal matrix. In this case, $\mathbf{W}$ (*i.e.*, $\mathbf{D}$) can be further diagonalized into two diagonal matrices $\mathbf{D}_a$ and $\mathbf{D}_b$. Thus, Eq. 3 can be re-written as *channel-wise co-attention*:

$$\mathbf{S} = \mathbf{V}_b^{\top}I^{-1}\mathbf{D}I\mathbf{V}_a = \mathbf{V}_b^{\top}\mathbf{D}_a^{\top}\mathbf{D}_b\mathbf{V}_a = (\mathbf{D}_a\mathbf{V}_b)^{\top}\mathbf{D}_b\mathbf{V}_a. \quad (5)$$

This operation is equal to applying a channel-wise weight to $\mathbf{V}_a$ and $\mathbf{V}_b$ before computing the similarity. This helps to alleviate channel-wise redundancy, which shares a similar spirit to Squeeze-and-Excitation mechanism [7, 20]. During ablation studies (§4.2), we perform detailed experiments to assess the effect of the different co-attention mechanisms, *i.e.*, vanilla co-attention (Eq. 3), symmetric co-attention (Eq. 4) and channel-wise co-attention (Eq. 5).

After obtaining the similarity matrix $\mathbf{S}$, as shown in the green and red areas in Fig. 3, we normalize $\mathbf{S}$ row-wise and column-wise with a softmax function:

$$\mathbf{S}^{c} = \text{softmax}(\mathbf{S}), \quad \mathbf{S}^{r} = \text{softmax}(\mathbf{S}^{\top}), \qquad (6)$$

where softmax($\cdot$) normalizes each column of the input. In Eq. 6, the i-th column of $\mathbf{S}^c$ is a vector with length WH. This vector reflects the relevance of each feature $(1, ..., WH)$ in $\mathbf{V}_a$ to the i-th feature in $\mathbf{V}_b$. Next, the attention summaries for the feature embedding $\mathbf{V}_a$ w.r.t. $\mathbf{V}_b$ can be computed as (see the blue areas in Fig. 3):

$$\mathbf{Z}_a = \mathbf{V}_b\mathbf{S}^c = \left[\mathbf{Z}_a^{(1)}\ \mathbf{Z}_a^{(2)}\ ...\ \mathbf{Z}_a^{(i)}\ ...\ \mathbf{Z}_a^{(WH)}\right] \in \mathbb{R}^{C\times(WH)},$$
$$\mathbf{Z}_a^{(i)} = \mathbf{V}_b \otimes \mathbf{S}^{c(i)} = \sum_{j=1}^{WH}\mathbf{V}_b^{(j)} \cdot \mathbf{s}_{ij}^c \in \mathbb{R}^C, \qquad (7)$$

where $\mathbf{Z}_a^{(i)}$ denotes the i-th column of $\mathbf{Z}_a$, '$\otimes$' denotes the matrix times vector, $\mathbf{S}^{c(i)}$ is the i-th column of $\mathbf{S}^c$, $\mathbf{V}_b^{(j)}$ indicates the j-th column of $\mathbf{V}^{(j)}$ and $\mathbf{s}_{ij}^c$ is the j-th element in $\mathbf{S}^{c(i)}$. Similarly, for frame $\mathbf{F}_b$, we compute the corresponding co-attention enhanced feature as: $\mathbf{Z}_b = \mathbf{V}_a\mathbf{S}^r$.

Gated co-attention. Considering the underlying appearance variations between input pairs, occlusions, and background noise, it is better to weight the information from different input frames, instead of treating all the co-attention information equally. To this end, a self-gate mechanism is introduced to allocate a co-attention confidence to each attention summary. The gate is formulated as follows:

$$f_g(\mathbf{Z}_a) = \sigma(\mathbf{w}_f\mathbf{Z}_a + b_f) \in [0, 1]^{WH},$$
$$f_g(\mathbf{Z}_b) = \sigma(\mathbf{w}_f\mathbf{Z}_b + b_f) \in [0, 1]^{WH}, \qquad (8)$$

where σ is the logistic sigmoid activation function, and $\mathbf{w}_f$ and b_f are the convolution kernel and bias, respectively. The gate f_g determines how much information from the reference frame will be preserved and can be learned automatically. After calculating the gate confidences, the attention summaries are updated by:

$$\mathbf{Z}_a = \mathbf{Z}_a \star f_g(\mathbf{Z}_a), \quad \mathbf{Z}_b = \mathbf{Z}_b \star f_g(\mathbf{Z}_b), \qquad (9)$$

where '$\star$' denotes channel-wise Hadamard product. These operations lead to a *gated co-attention* framework.

Then we concatenate the final co-attention representation $\mathbf{Z}$ and the original feature $\mathbf{V}$ together:

$$\mathbf{X}_a = [\mathbf{Z}_a, \mathbf{V}_a] \in \mathbb{R}^{W\times H\times 2C}, \quad \mathbf{X}_b = [\mathbf{Z}_b, \mathbf{V}_b] \in \mathbb{R}^{W\times H\times 2C}, \quad (10)$$

where '$[\cdot]$' denotes the concatenation operation. Finally, the co-attention enhanced feature $\mathbf{X}$ can be fed into a segmentation network to produce a final result $\mathbf{Y} \in [0, 1]^{W\times H}$.

3.2. Full COSNet Architecture

Fig. 4 shows the training and testing pipelines of the proposed COSNet. Basically, COSNet is a Siamese network which consists of three cascaded parts: a DeepLabv3 [6] based feature embedding module, a co-attention module (detailed in §3.1) and a segmentation module.

Network architecture during training phase. In the training phase, the Siamese network based COSNet takes two
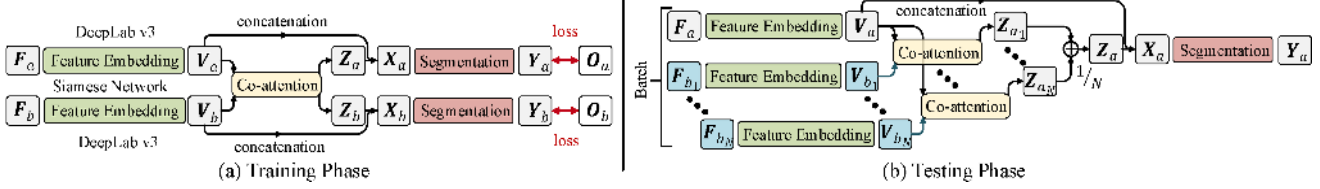
Figure 4. Schematic illustration of training pipeline (a) and testing pipeline (b) of COSNet.

streams as input, *i.e.*, a pair of the frame images $\{\mathbf{F}_a, \mathbf{F}_b\}$ which are randomly sampled from the same video. First, the feature embedding module is used to build their feature representations: $\{\mathbf{V}_a, \mathbf{V}_b\}$. Next, $\{\mathbf{V}_a, \mathbf{V}_b\}$ are refined by the co-attention module and the co-attention enhanced feature $\{\mathbf{X}_a, \mathbf{X}_b\}$ are computed through Eq. 10. Finally, the corresponding segmentation predictions $\{\mathbf{Y}_a, \mathbf{Y}_b\}$ are produced by the segmentation module which consists of multiple small kernel convolution layers. Detailed configurations of the three modules can be found in the next section.

As we discussed in §1, primary objects in videos have two essential properties: (i) intra-frame discriminability, and (ii) inter-frame consistency. To distinguish the foreground target(s) from the background (property (i)), we utilize data from existing salient object segmentation datasets [10, 67] to train our backbone feature embedding module. As primary salient object instances are annotated in each image of these datasets, the learned feature embedding can catch and discriminate the objects of most interest. Meanwhile, to ensure COSNet is able to capture the global inter-frame coherence of the primary video objects (property (ii)), we train the whole COSNet with video segmentation data, where the co-attention module plays a key role in capturing the correlations between video frames. Specifically, we take two randomly selected frames in a video sequence to build training pairs. It is worth mentioning that this operation naturally and effectively augments training data, compared to previous recurrent neural network based UVOS models that take only consecutive frames.

In this way, the COSNet is alternatively trained with static image data and dynamic video data. When using image data, we only train the feature embedding module, where an extra 1×1 convolution layer with *sigmoid* activation is added to generate intermediate segmentation sideoutput. The video data is used to train the whole COSNet, including the feature embedding module, the co-attention module as well as the segmentation module. We employ the weighted binary cross entropy loss to train the network:

$$\mathcal{L}_{\mathcal{C}}(\mathbf{Y},\mathbf{O}) = -\sum_{x} (1-\eta) o_x \log(y_x) + \eta(1-o_x)\log(1-y_x), \ (11)$$

where $\mathbf{O} \in \{0,1\}^{W \times H}$ denotes the binary ground-truth, y_x is the intermediate or final segment prediction $\mathbf{Y}$ at pixel x, and η is the foreground-background pixel number ratio.

In addition, for the symmetric co-attention in Eq. 4, we add an extra orthogonal regularization into the loss function

to maintain the symmetry of weight matrix $\mathbf{W}$:

$$\mathcal{L} = \mathcal{L}_{\mathcal{C}} + \lambda \left| \mathbf{W}\mathbf{W}^\top - I \right|, \quad (12)$$

where λ is the regularization parameter.

Network architecture during testing phase. Once the network is trained, we apply the COSNet to unseen videos. Intuitively, given a test video, we can feed each frame to be segmented, along with only one reference frame sampled from the same video, into the COSNet successively. Performing this operation frame-by-frame, we can obtain all the segmentation results. However, with such a simple strategy, the segmentation results still contain considerable noise, since the rich and global correlation information in the videos is not fully explored. Therefore, it is critical to include more references during the testing phase (see Fig. 4 (b)). One intuitive solution is to feed a set of N different reference frames (uniformly sampled from the same video) into the inference branches and average all predictions. A more favored way is that for the query frame $\mathbf{F}_a$, with the reference frame set $\{\mathbf{F}_{b_n}\}_{n=1}^N$ containing N reference frames, Eq. 9 is further reformulated by considering more attention summaries $\{\mathbf{Z}_{a_n}\}_{n=1}^N$:

$$\mathbf{Z}_a \leftarrow \frac{1}{N} \sum_{n=1}^N \mathbf{Z}_{a_n} \star f_g(\mathbf{Z}_{a_n}). \quad (13)$$

In this way, during the testing phase, the co-attention based feature $\mathbf{Z}_a$ is able to efficiently capture the foreground information from a global view by considering more reference frames. Then we feed $\mathbf{Z}_a$ into the segmentation module to generate the final output $\mathbf{Y}_a$. Following the widely used protocol [53, 52, 49], we apply CRF as a post-processing step. In §4.2, we will quantitatively demonstrate the performance improvement with the increasing number of reference frames.

3.3. Implementation Details

Detailed network architecture. The backbone network of our COSNet is DeepLabv3 [6], which consists of the first five convolution blocks from ResNet [19] and an atrous spatial pyramid pooling (ASPP) module [6]. For the vanilla co-attention module (Eq. 3), we implement the weight matrix $\mathbf{W}$ using a fully connected layer with 512×512 parameters. In addition, the channel-wise co-attention in Eq. 5 is built on a Squeeze-and-Excitation (SE)-like module [20]. Specifically, the channel weights generated through fully
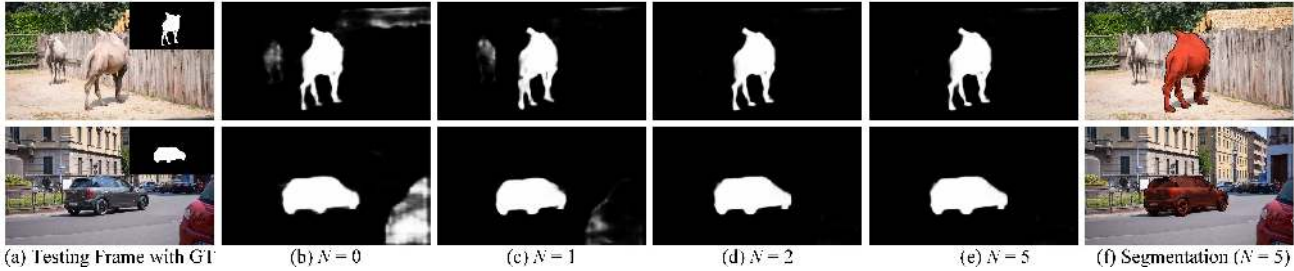
Figure 5. Performance improvement for an increasing number of reference frames (§4.2). (a) Testing frames with ground-truths overlaid. (b)-(e) Primary object predictions with considering different number of reference frames ($N = 0, 1, 2$, and 5). (f) Binary segments through applying CRF to (e). We can see that without co-attention, the COSNet degrades to a frame-by-frame segmentation model ((b): $N = 0$). Once co-attention is added ((c): $N = 1$), similar foreground distraction can be suppressed efficiently. Furthermore, more inference frames contribute to better segmentation performance ((c)-(e)).

connected layer with 512 nodes in one branch are applied to the feature embedding of the other branch [20]. Eq. 8 is implemented with 1×1 convolution layer with sigmoid activation function. The segmentation module consists of two 3×3 convolutional layers (with 256 filters and batch norm) and a 1×1 convolutional layer (with 1 filter and $sigmoid$ activation) for final segmentation prediction.

Training settings. The whole training procedure of our COSNet consists of two alternated steps. When using static data to fine-tune the DeepLabV3 based feature embedding module, we take advantage of image saliency datasets: MSRA10K [10] and DUT [67]. In this way, the pixels belong to the foreground target tend to close to each other. Meanwhile, we train the whole model with the training videos in DAVIS16 [45]. In this step, two randomly selected frames from the same sequence are fed into COSNet as training pairs. Given the input RGB frame images of size $473 \times 473 \times 3$, the size of the feature embeddings $\mathbf{V}_a$ and $\mathbf{V}_b$ are ($W = 60, H = 60, C = 512$). The entire network is trained using the SGD optimizer with an initial learning rate of 2.5×10^{-4}. During training, the batch size is set to 8 and the hyper-parameter λ in Eq. 12 is set to 10^{-4}. We implement the whole algorithm with Pytorch. All experiments and analyses are conducted on a Nvidia TITAN Xp GPU and an Intel (R) Xeon E5 CPU. The ove

4. Experiments

4.1. Experimental Setup

We conduct experiments on the three most famous UVOS datasets: DAVIS16 [45], FBMS [41] and Youtube-Objects [47] datasets.

DAVIS16 is a recent dataset which consists of 50 videos in total (30 videos for training and 20 for testing). Per-frame pixel-wise annotations are offered. For quantitative evaluation, following the standard evaluation protocol from [45], we adopt three metrics, namely region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$, and time stability $\mathcal{T}$.

FBMS is comprised of 59 video sequences. Different from

Network Variant	DAVIS		FBMS		Youtube-Objects	
	mean $\mathcal{J}$	$\Delta\mathcal{J}$	mean $\mathcal{J}$	$\Delta\mathcal{J}$	mean $\mathcal{J}$	$\Delta\mathcal{J}$
Co-attention Mechanism						
Vanilla co-attention (Eq. 3)	80.0	-0.5	75.2	-0.4	70.3	-0.2
Symmetric co-attention (Eq. 4)	80.5	-	75.6	-	70.5	-
Channel-wise co-attention (Eq. 5)	77.2	-3.3	72.7	-2.9	67.5	-3.0
w/o. Co-attention	71.3	-9.2	70.1	-5.5	62.9	-7.6
Fusion Strategy						
Attention summary fusion (Eq. 13)	80.5	-	75.6	-	70.5	-
Prediction segmentation fusion	79.5	-1.0	74.2	-1.4	69.9	-0.6
Frames Selection Strategy						
Global uniform sampling	80.53	-	75.61	-	70.54	-0.01
Global random sampling	80.52	-0.01	75.54	-0.02	70.55	-
Local consecutive sampling	80.26	-0.27	75.52	-0.09	70.43	-0.12

Table 1. Ablation study (§4.2) of COSNet on DAVIS16 [45], FBMS [41] and Youtube-Objects [47] datasets with different co-attention mechanisms, fusion strategies and sampling strategies.

Dataset	Number of reference frames (N)				
	0	1	2	5	7
DAVIS	71.3	77.6	79.7	80.5	80.5
FBMS	70.2	74.8	75.3	75.6	75.6
Youtube-Objects	62.9	67.7	70.5	70.5	70.5

Table 2. Comparisons with different numbers of reference frames during the testing stage on DAVIS16 [45], FBMS [41] and Youtube-Objects [47] datasets (§4.2). The mean $\mathcal{J}$ is adopted.

the DAVIS dataset, the ground-truth of FBMS is sparsely labeled (only 720 frames are annotated). Following the common setting [53, 52, 30, 32, 33, 49, 9], we validate the proposed method on the testing split which consists of 30 sequences. The region similarity $\mathcal{J}$ is used for evaluation.

Youtube-Objects contains 126 video sequences which belong to 10 objects categories with more than 20,000 frames in total. We use the region similarity $\mathcal{J}$ to measure the segmentation performance.

4.2. Diagnostic Experiments

In this section, we focus on exploration studies to assess the important setups and components of COSNet. The experiments were performed on the test sets of DAVIS16 [45] and FBMS [41] as well as the whole Youtube-Objects [47]. The evaluation criterion is mean region similarity ($\mathcal{J}$).

Comparison of different co-attention mechanisms. We

	Method	TRC [17]	CVOS [51]	KEY [31]	MSG [40]	NLC [14]	CUT [9]	FST [42]	SFL [28]	LMP [52]	FSEG [24]	LVO [53]	ARP [30]	PDB [49]	**COSNet**
	Mean	47.3	48.2	49.8	53.3	55.1	55.2	55.8	67.4	70.0	70.7	75.9	76.2	77.2	**80.5**
$\mathcal{J}$	Recall	49.3	54.0	59.1	61.6	55.8	57.5	64.9	81.4	85.0	83.0	89.1	91.1	90.1	**94.0**
	Decay	8.3	10.5	14.1	2.4	12.6	2.2	**0.0**	6.2	1.3	1.5	**0.0**	7.0	0.9	**0.0**
	Mean	44.1	44.7	42.7	50.8	52.3	55.2	51.1	66.7	65.9	65.3	72.1	70.6	74.5	**79.4**
$\mathcal{F}$	Recall	43.6	52.6	37.5	60.0	61.0	51.9	51.6	77.1	79.2	73.8	83.4	83.5	84.4	**90.4**
	Decay	12.9	11.7	10.6	5.1	11.4	3.4	2.9	5.1	2.5	1.8	1.3	7.9	**-0.2**	0.0
$\mathcal{T}$	Mean	39.1	**25.0**	26.9	30.2	42.5	27.7	36.6	28.2	57.2	32.8	26.5	39.3	29.1	31.9

Table 3. Quantitative results on the test set of DAVIS16 [45][1] (see §4.3), using the region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and time stability $\mathcal{T}$. We also report the recall and the decay performance over time for both $\mathcal{J}$ and $\mathcal{F}$. The best scores are marked in **bold**.

first study the effect of different co-attention mechanisms in COSNet, *i.e.*, vanilla co-attention (Eq. 3), symmetric co-attention (Eq. 4) and channel-wise co-attention (Eq. 5). In Table 1, both the fully connected method and the symmetric method achieve better performance than the channel attention mechanism. This proves the importance of space transformation in co-attention. Furthermore, compared with vanilla co-attention, we find symmetric co-attention performs slightly better. We attribute this to the orthogonal constraint which reduces feature redundancy while preserving the norm of the features unchanged.

Effect of co-attention mechanism. When excluding the co-attention module and only using the base feature embedding network (DeepLabv3), we observe a significant performance drop (mean $\mathcal{J}$: 80.5→71.3 in DAVIS), clearly showing the effectiveness of our strategy, which leverages co-attention mechanism to model UVOS from a global view.

Attention summary fusion *vs* prediction fusion. In Eq. 13, we fuse the information from other reference frames by averaging the corresponding co-attention summaries. To verify its effectiveness, we implement another alternative baseline *Prediction Fusion*: $\mathbf{Y}_a = \frac{1}{N}\sum_{n=1}^{N}\mathbf{Y}_{a_n}$, *i.e.*, directly average the predictions by considering different reference frames. The results in Table 1 demonstrate the superiority of fusion in the feature embedding space.

Comparison of different frame selection strategies. To investigate frame selection strategy during the testing phase on the final prediction, we further conduct a series of experiments using different sampling methods. Specifically, we adopt global random sampling, global uniform sampling as well as local consecutive sampling. From Table 1, it can be observed that both global-level sampling strategy achieve approximate performance but better than local sampling method. Meanwhile, local sampling-based results are still superior to the results obtained from the backbone network. Overall comparisons further prove the importance of incorporating global context.

Influence of the number of reference frames. It is also of interest to assess the influence of the number of reference frames N on the final performance. Table 2 shows the results for this. When N is equal to 0, this means that there is no co-attention for segmentation. We observe a large performance improvement when N changes from 0 to 1, which

Method	NLC [14]	FST [42]	FSEG [24]	MSTP [21]	ARP [30]
Mean $\mathcal{J}$	44.5	55.5	68.4	60.8	59.8
Method	IET [32]	OBN [33]	PDB [49]	SFL [9]	**COSNet**
Mean $\mathcal{J}$	71.9	73.9	74.0	56.0	**75.6**

Table 4. Quantitative performance on the test sequences of FBMS [41] (§4.3) using region similarity (mean $\mathcal{J}$).

proves the importance of co-attention. Furthermore, when N changes from 2 to 5, the quantitative results show increased performance. When we further increase N, the final performance does not change obviously. We set the value of N to 5 in the evaluation experiments.

Fig. 5 further visualizes the qualitative segmentation result for an increasing number of inference frames. When $N = 0$, the feature embedding module has learned to discriminate the foreground target from the background. However, when a similar object distractor appears (*e.g.*, the small camel in the first row, or the red car in the second row), the feature embedding module fails to capture the primary target, since no ground-truth is given. In this case, the proposed co-attention mechanism can refer to long-range frames and capture the primary object, thus effectively suppressing the similar target distraction.

4.3. Quantitative and Qualitative Results

Evaluation on DAVIS16 [45]. Table 3 shows the overall results, with all the top performance methods taken from the DAVIS 2016 benchmark[1] [45]. COSNet outperforms all the reported methods across most metrics. Compared with the second best method, PDB [49], our COSNet achieves gains of 2.6% and 4.9% on $\mathcal{J}$ Mean and $\mathcal{F}$ Mean, respectively.

In Table 3, several other deep learning based state-of-the-art UVOS methods [9, 52, 24, 53, 33] leverage both appearance as well as extra motion information to improve the performance. Different from these methods, the proposed COSNet only utilizes appearance information but achieves superior performance. We attribute our performance improvement to the consideration of more temporal information through the co-attention mechanism. Compared with these methods using optical flow to catch successive temporal information, the advantage of exploiting the temporal

---

[1]https://davischallenge.org/davis2016/soa_compare.html

Figure 6. Qualitative results on three datasets (§4.3). From top to bottom: *dance-twirl* from the DAVIS16 dataset [45], *horses05* from the FBMS dataset [41], and *bird0014* from the Youtube-Objects dataset [47].

Method	FST [42]	COSEG [55]	ARP [30]	LVO [53]	PDB [49]	FSEG [24]	SFL [9]	**COSNet**
Airplane (6)	70.9	69.3	73.6	86.2	78.0	81.7	65.6	81.1
Bird (6)	70.6	76.0	56.1	81.0	80.0	63.8	65.4	75.7
Boat (15)	42.5	53.5	57.8	68.5	58.9	72.3	59.9	71.3
Car (7)	65.2	70.4	33.9	69.3	76.5	74.9	64.0	77.6
Cat (16)	52.1	66.8	30.5	58.8	63.0	68.4	58.9	66.5
Cow (20)	44.5	49.0	41.8	68.5	64.1	68.0	51.1	69.8
Dog (27)	65.3	47.5	36.8	61.7	70.1	69.4	54.1	76.8
Horse (14)	53.5	55.7	44.3	53.9	67.6	60.4	64.8	67.4
Motorbike (10)	44.2	39.5	48.9	60.8	58.3	62.7	52.6	67.7
Train (5)	29.6	53.4	39.2	66.3	35.2	62.2	34.0	46.8
Mean $\mathcal{J}$	53.8	58.1	46.2	67.5	65.4	68.4	57.0	**70.5**

Table 5. Quantitative performance of each category on Youtube-Objects [47] (§4.3) with the region similarity (mean $\mathcal{J}$). We show the average performance for each of the 10 categories from the dataset and the final row shows an average over all the videos.

correlation from a global view is clear when dealing with similar target distractions.

Evaluation on FBMS [41]. We also perform experiments on the FBMS dataset for completeness. Table 4 shows that our COSNet performs better (75.6% in mean $\mathcal{J}$) than state-of-the-art methods [14, 42, 24, 21, 30, 32, 33, 49, 9]. In most competing methods, except for the RGB input, additional optical flow information is utilized to estimate the segmentation mask. Considering lots of foreground objects in FBMS share similar appearance with the background but have different motion patterns, optical flow information clearly benefits the prediction. By contrast, our COSNet only takes advantage of the original RGB information and achieves better performance.

Evaluation on Youtube-Objects [47]. Table 5 illustrates the results of all compared methods for different categories. Our approach outperforms all compared methods [42, 55, 30, 53, 49, 24, 9] by a large margin. FSEG performs second best under the mean $\mathcal{J}$ metric. It is worth noting that the Youtube-Objects dataset shares categories with the training samples in FSEG, which contributes to the enhanced performance [24]. In addition, all the categories in Youtube-Objects can be divided into two types: grid objects (*e.g.*, *Airplane*, *Train*) and non-grid objects (*e.g.*, *Bird*,

Cat). Despite the objects in the latter class often undergoing shape deformation and quick appearance variation, the COSNet can capture long-term dependency and handle these scenarios better than all compared methods.

Qualitative Results. Fig. 6 shows the qualitative results across three datasets. DAVIS16 [45] contains many challenging videos with fast motion, deformation and multiple instances of the same category. We can see that the proposed COSNet can track the primary region or target tightly by leveraging a co-attention scheme to consider global temporal information. The co-attention mechanism helps the proposed COSNet to segment out primary objects from the cluttered background. The effectiveness can also be seen in the *bird0014* sequence of the Youtue-Objects dataset. In addition, we observe that some videos contain multiple moving targets (*e.g.*, *horses05*) in the FBMS dataset, and the proposed COSNet can deal with such scenarios well.

5. Conclusion

By regarding UVOS as a temporal coherence capturing task, we proposed a novel model, COSNet, to estimate the primary target(s). Through an alternated network training strategy with saliency image and video pairs, the proposed network learns to discriminate primary objects from the background in each frame and capture the temporal correlation across frames. The proposed method achieved superior performance on three representative video segmentation datasets. Extensive experimental results proved that our method can effectively suppress similar target distraction despite no annotation being given during the segmentation. The COSNet is a general framework for handling sequential data learning, and can be readily extended to other video analysis tasks, such as video saliency detection and optical flow estimation.

References

[1] Xue Bai, Jue Wang, David Simons, and Guillermo Sapiro. Video SnapCut: robust video object cutout using localized classifiers. *TOG*, 28(3):70, 2009. 2

[2] Linchao Bao, Baoyuan Wu, and Wei Liu. CNN in MRF: Video object segmentation via inference in a CNN-based higher-order spatio-temporal MRF. In *CVPR*, 2018. 2

[3] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. In *ICLR*, 2017. 4

[4] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 2

[5] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *CVPR*, 2017. 2, 3

[6] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017. 4, 5

[7] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. SCA-CNN: spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, 2017. 3, 4

[8] Jingchun Cheng, Yi-Hsuan Tsai, Wei-Chih Hung, Shengjin Wang, and Ming-Hsuan Yang. Fast and accurate online video object segmentation via tracking parts. In *CVPR*, 2018. 2, 3

[9] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *ICCV*, 2017. 2, 6, 7, 8

[10] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE TPAMI*, 37(3):569–582, 2015. 5, 6

[11] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L. Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. In *CVPR*, 2017. 3

[12] Hai Ci, Chunyu Wang, and Yizhou Wang. Video object segmentation by learning location-sensitive embeddings. In *ECCV*, 2018. 3

[13] Misha Denil, Loris Bazzani, Hugo Larochelle, and Nando de Freitas. Learning where to attend with deep architectures for image tracking. *Neural Computation*, 24(8):2151–2184, 2012. 3

[14] Alon Faktor and Michal Irani. Video segmentation by non-local consensus voting. In *BMVC*, 2014. 2, 7, 8

[15] Hao-Shu Fang, Jinkun Cao, Yu-Wing Tai, and Cewu Lu. Pairwise body-part attention for recognizing human-object interactions. In *ECCV*, 2018. 3

[16] Katerina Fragkiadaki, Pablo Arbelaez, Panna Felsen, and Jitendra Malik. Learning to segment moving objects in videos. In *CVPR*, 2015. 2

[17] Katerina Fragkiadaki, Geng Zhang, and Jianbo Shi. Video segmentation by tracing discontinuities in a trajectory embedding. In *CVPR*, 2012. 2, 7

[18] Huazhu Fu, Dong Xu, Bao Zhang, and Stephen Lin. Object-based multiple foreground video co-segmentation. In *CVPR*, 2014. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[20] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 4, 5, 6

[21] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G. Schwing. Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation. In *ECCV*, 2018. 2, 7, 8

[22] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G. Schwing. Videomatch: Matching based video object segmentation. In *ECCV*, 2018. 3

[23] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *NIPS*, 2015. 3

[24] Suyog Dutt Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In *CVPR*, 2017. 2, 7, 8

[25] Varun Jampani, Raghudeep Gadde, and Peter V. Gehler. Video propagation networks. In *CVPR*, 2017. 3

[26] Saumya Jetley, Nicholas A Lord, Namhoon Lee, and Philip HS Torr. Learn to pay attention. In *ICLR*, 2018. 3

[27] Yeong Jun Koh, Young-Yoon Lee, and Chang-Su Kim. Sequential clique optimization for video object segmentation. In *ECCV*, 2018. 2

[28] Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *ICCV*, 2015. 2, 7

[29] A. Khoreva, R. Benenson, E. Ilg, T. Brox, and B. Schiele. Lucid data dreaming for object tracking. *CVPR Workshops*, 2017. 3

[30] Yeong Jun Koh and Chang-Su Kim. Primary object segmentation in videos based on region augmentation and reduction. In *CVPR*, 2017. 2, 6, 7, 8

[31] Yong Jae Lee, Jaechul Kim, and Kristen Grauman. Key-segments for video object segmentation. In *ICCV*, 2011. 2, 7

[32] Siyang Li, Bryan Seybold, Alexey Vorobyov, Alireza Fathi, Qin Huang, and C.-C. Jay Kuo. Instance embedding transfer to unsupervised video object segmentation. In *CVPR*, 2018. 2, 6, 7, 8

[33] Siyang Li, Bryan Seybold, Alexey Vorobyov, Xuejing Lei, and C.-C. Jay Kuo. Unsupervised video object segmentation with motion-based bilateral networks. In *ECCV*, 2018. 2, 6, 7, 8

[34] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *ECCV*, 2018. 3

[35] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NIPS*, 2016. 3

[36] Xiankai Lu, Chao Ma, Bingbing Ni, Xiaokang Yang, Ian Reid, and Ming-Hsuan Yang. Deep regression tracking with shrinkage loss. In *ECCV*, 2018. 3

[37] Tianyang Ma and Longin Jan Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *CVPR*, 2012. 2

[38] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *NIPS*, 2014. 3

[39] Duy-Kien Nguyen and Takayuki Okatani. Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. In *CVPR*, 2018. 3

[40] Peter Ochs and Thomas Brox. Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions. In *ICCV*, 2011. 2, 7

[41] Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE TPAMI*, 36(6):1187–1200, 2014. 2, 6, 7, 8

[42] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *ICCV*, 2013. 2, 7, 8

[43] Deepak Pathak, Ross Girshick, Piotr Dollar, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017. 2

[44] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, 2017. 2

[45] 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 *CVPR*, 2016. 2, 6, 7, 8

[46] Federico Perazzi, Oliver Wang, Markus H. Gross, and Alexander Sorkine-Hornung. Fully connected object proposals for video segmentation. In *ICCV*, 2015. 2

[47] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 2, 6, 8

[48] Pau Rodríguez, Jordi Gonzalez, Guillem Cucurull, Josep M Gonfaus, and Xavier Roca. Regularizing cnns with locally constrained decorrelations. In *ICLR*, 2017. 4

[49] Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *ECCV*, 2018. 2, 5, 6, 7, 8

[50] Yifan Sun, Liang Zheng, Weijian Deng, and Shengjin Wang. Svdnet for pedestrian retrieval. In *ICCV*, 2017. 4

[51] Brian Taylor, Vasiliy Karasev, and Stefano Soatto. Causal video object segmentation from persistence of occlusions. In *CVPR*, 2015. 7

[52] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning motion patterns in videos. In *CVPR*, 2017. 2, 5, 6, 7

[53] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning video object segmentation with visual memory. In *ICCV*, 2017. 2, 5, 6, 7, 8

[54] Yi-Hsuan Tsai, Ming-Hsuan Yang, and Michael J. Black. Video segmentation via object flow. In *CVPR*, 2016. 2

[55] Yi-Hsuan Tsai, Guangyu Zhong, and Ming-Hsuan Yang. Semantic co-segmentation in videos. In *ECCV*, 2016. 2, 8

[56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 3

[57] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *CVPR*, 2017. 3

[58] Wenguan Wang and Jianbing Shen. Deep visual attention prediction. *IEEE TIP*, 27(5):2368–2378, 2018. 3

[59] Wenguan Wang, Jianbing Shen, Xuelong Li, and Fatih Porikli. Robust video object cosegmentation. *IEEE TIP*, 24(10):3137–3148, 2015. 3

[60] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *CVPR*, 2015. 2

[61] Wenguan Wang, Jianbing Shen, Jianwen Xie, and Porikli Fatih. Super-trajectory for video segmentation. In *ICCV*, 2017. 2

[62] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 3

[63] Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton van den Hengel. Are you talking to me? Reasoned visual dialog generation through adversarial learning. In *CVPR*, 2018. 3

[64] Huaxin Xiao, Jiashi Feng, Guosheng Lin, Yu Liu, and Maojun Zhang. Monet: Deep motion exploitation for video object segmentation. In *CVPR*, 2018. 2

[65] Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. In *ICLR*, 2017. 3

[66] Yichao Yan, Bingbing Ni, Zhichao Song, Chao Ma, Yan Yan, and Xiaokang Yang. Person re-identification via recurrent feature aggregation. In *ECCV*, 2016. 3

[67] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, 2013. 5, 6

[68] Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K Katsaggelos. Efficient video object segmentation via network modulation. In *CVPR*, 2018. 2

[69] Jae Shin Yoon, François Rameau, Jun-Sik Kim, Seokju Lee, Seunghak Shin, and In So Kweon. Pixel-level matching for video object segmentation using convolutional neural networks. In *ICCV*, 2017. 2

[70] Dong Zhang, Omar Javed, and Mubarak Shah. Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In *CVPR*, 2013. 2

[71] Wangmeng Zuo, Xiaohe Wu, Liang Lin, Lei Zhang, and Ming-Hsuan Yang. Learning support correlation filters for visual tracking. *IEEE transactions on pattern analysis and machine intelligence*, 2018. 2