Signal Processing: Image Communication 98 (2021) 116373

Contents lists available at ScienceDirect

Signal Processing: Image Communication

journal homepage: www.elsevier.com/locate/image

Fast pixel-matching for video object segmentation ☆

Siyue Yu [a], Jimin Xiao [a,*], Bingfeng Zhang [a], Eng Gee Lim [a], Yao Zhao [b]

[a] Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, China
[b] Beijing Jiaotong University, Beijing, China

ARTICLE INFO

Keywords:
Non-local pixel matching
Mask-propagation
Encoder–decoder

ABSTRACT

Video object segmentation, aiming to segment the foreground objects given the annotation of the first frame, has been attracting increasing attentions. Many state-of-the-art approaches have achieved great performance by relying on online model updating or mask-propagation techniques. However, most online models require high computational cost due to model fine-tuning during inference. Most mask-propagation based models are faster but with relatively low performance due to failure to adapt to object appearance variation. In this paper, we are aiming to design a new model to make a good balance between speed and performance. We propose a model, called NPMCA-net, which directly localizes foreground objects based on mask-propagation and non-local technique by matching pixels in reference and target frames. Since we bring in information of both first and previous frames, our network is robust to large object appearance variation, and can better adapt to occlusions. Extensive experiments show that our approach can achieve a new state-of-the-art performance with a fast speed at the same time (86.5% IoU on DAVIS-2016 and 72.2% IoU on DAVIS-2017, with speed of 0.11s per frame) under the same level comparison. Source code is available at https://github.com/siyueyu/NPMCA-net.

1. Introduction

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

To tackle these problems, some studies adopted online-training mechanism [1–4]. Given the ground-truth mask of the first frame in a test video, they used it to fine-tune the model to obtain the object appearance. In the following inference process, they used the predicted masks to further fine-tune their models. With fine-tuning, the models can adapt to object appearance change, though, the online learning process is time-consuming and inefficient.

Recently, boosted by the rapid development of mask-propagation based VOS models [5–7], a better balance between speed and accuracy is reached. The core idea of these methods is to use the estimated mask of the previous frame to guide the model to make segmentation prediction for the current frame. For example, Perazzi et al. [4] proposed to use guidance of previous predicted mask as guidance for the network to learn mask prediction and it proposed a combination of offline and online training method to train the model. They firstly used static image datasets for offline training, and then used the first frame of a test video sequence to fine-tune the model. Oh et al. [5] proposed a Siamese encoder–decoder network with guidance of the previous mask to produce the target object probability map. Johnander et al. [6] offered an appearance module which utilized a class-conditional mixture of Gaussians to model the foreground object appearance for mask prediction. Sun et al. [8] considered both the mask of previous frame and the optical flow to predict target mask. These approaches are usually faster than online training based VOS methods, but they are less adaptive to object appearance variation.

Both online training and mask-propagation based VOS models have limitations, a balance between segmentation accuracy and running speed is crucial for VOS. Early mask-propagation based networks use current frame with previous estimated mask [4] or adding first frame with its provided mask as reference information [5] to directly predict the segmentation mask of current frame. Additionally, Sun et al. [8] used optical flow to build relationship between the previous and the current frames. Different from these methods, we design an attention-based pixel-matching module to find the pixels belonging to the target object in the current frame based on the feature similarity between the current frame and reference frames. In order to capture the object feature without the interference of background, we choose to mask it

☆ The work was supported by National Key Research and Development of China under 2018AAA0102100, National Natural Science Foundation of China under 61972323, U1936212, 61972323, U1936212 and Key Program Special Fund in XJTLU under KSFT-02, KSF-P-02.

* Corresponding author.
E-mail addresses: siyue.yu@xjtlu.edu.cn (S. Yu), jimin.xiao@xjtlu.edu.cn (J. Xiao), bingfeng.zhang@xjtlu.edu.cn (B. Zhang), enggee.lim@xjtlu.edu.cn (E.G. Lim), yzhao@bjtu.edu.cn (Y. Zhao).

https://doi.org/10.1016/j.image.2021.116373
Received 18 March 2020; Received in revised form 18 April 2021; Accepted 2 July 2021
Available online 22 July 2021

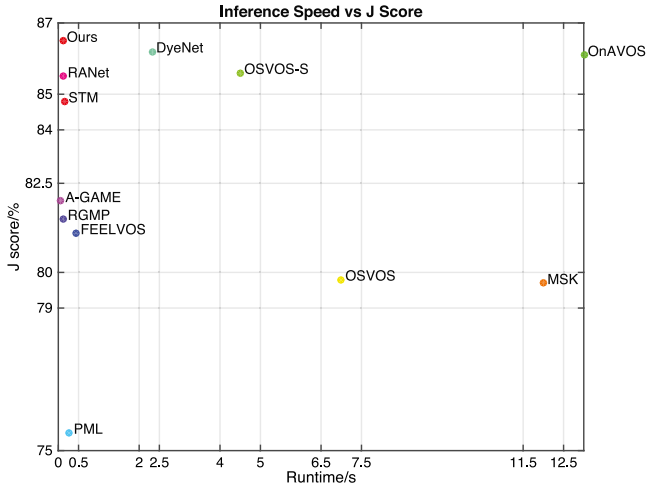


Fig. 1. The IoU score ($\mathcal{J}$) versus running time on each frame (s) for various VOS approaches on the DAVIS-2016 validation set. Our model can keep a good balance between performance and efficiency.

out and discard the background pixels. However, the target object is varying frame by frame, such process will cause large object appearance variation. Therefore, we choose to use both the first frame and the previous frame as references to provide object information for our pixel-matching module.

With the target object's appearance information, we need to determine the target object location, in terms of mask, in the current frame. We design our model based on mask-propagation to keep efficiency, where the non-local structure [9] is adopt to generate the object mask using the obtained target object's appearance information. Specifically, we design a video object segmentation model called Non-local Pixel-Matching network with Channel Attention (NPMCA-net), which includes a newly designed pixel-matching module and a channel attention module. The pixel-matching module is designed to match pixels between the target frame and the reference frames with given ground-truth mask or estimated mask. The channel attention module is used to augment the matched feature map to achieve better decoding. Extensive experiments have shown that our network can achieve a new state-of-the-art performance without loss of efficiency. To better display the accuracy and speed trade-off, we plot our IoU score versus speed in Fig. 1. Our NPMCA-net can achieve both high performance and high efficiency at the same time. Our main contribution is summarized as follows:

- We propose a video object segmentation model (NPMCA-net) that strikes a good balance between accuracy and running speed. The model does not rely on online fine-tuning technique, so as to lower the computational demands, yet it can adaptively catch the target object's appearance variation by using both image and predicted mask information in the previous frame.
- Our proposed non-local pixel-matching module can effectively predict the target object mask by aggregating multi-frame information. Moreover, the proposed model also provides high level interpretability by visualizing the obtained feature maps.
- Our model achieves new state-of-the-art performances on DAVIS-2016 (IoU: 86.5%) and DAVIS-2017 (IoU: 72.2%) datasets, using the same experimental setting.

2. Related works

Video object segmentation. Different from statistic image tasks [10–14], VOS only considers to segment the moving object without class reference or prediction. VOS research can be divided into two main categories, one is unsupervised methods and the other is semi-supervised methods. Unsupervised methods, such as [15–17], tried to segment the foreground objects without any given labels. Semi-supervised methods aim to segment the objects in a video with a given ground-truth mask in the first frame. For example, some approaches [1–4] used online fine-tuning to make the model robust to object appearance variation. And some studies [5–7] based on mask-propagation solely relied on offline training for this task, making the models more efficient. Some [18,19] took advantage of Mask R-CNN [20] to predict corresponding box of each object and then conduct segmentation. Additionally, Sun et al. [21] used reinforcement learning to choose better proposals for target object bounding box and then conduct segmentation. However, offline methods generally are less performing than online ones. In this paper, we will focus on mask-propagation based methods under case of semi-supervision and try to design a fast and high-performance model.

Embedding based network. Embedding based networks use an embedding vector to represent each pixel. They have been successfully employed in many vision tasks [22–24]. Many successful VOS approaches are also based on embedding. PML [25] employed an embedding vector to represent each pixel, and then embedding vectors in reference frame are matched with that of target frame using a triplet loss. VideoMatch [26] proposed a matching based algorithm for VOS, which learned to match extracted features to a provided template without memorizing the appearance of target objects. Besides, Ci et al. [27] attempted to predict foreground object by learning location-sensitive embedding. FEELVOS [28] proposed a semantic pixel-wise embedding with a global and local matching mechanism for this task, and Yoon et al. [29] utilized features from different depth layers by combinations of convolution, max pooling and Rectified Linear Units to distinguish the target area from the background. However, most embedding based networks need guidance information to tell which pixels belong to foreground and which ones belong to background. In this paper, we directly match the features with our proposed mechanism (directly compute the similarity of pixels) without the separation of positive or negative pools.

Cosegmentation. Some methods utilize cosegmentation to discover video object. VODC [30] was proposed to distinguish which frames contained the target object and then segmentation was conducted on these corresponding frames. They designed a spatio-temporal auto-context model to obtain superpixel label for each frame and then a multiple instance boosting algorithm with spatial reasoning was deployed to synchronously detect whether a frame contained the target object and predict the segmentation map. Besides, Wang et al. [31] proposed an energy optimization framework which combined intraframe saliency, interframe consistency and across-video similarity. They used saliency and spatio-temporal SIFT flow to detect initial pixels for common object. Then, the spatio-temporal SIFT was used to refine the coarse object regions generated by the prior step. Additionally, Li et al. [32] designed a robust ensemble clustering scheme to predict object-like proposals for unsupervised cosegmentation task. Once the proposals were generated, unary and pairwise energy potentials were minimized with the α-expansion to train the model. Although these methods have achieved satisfactory results, they are designed to detect the common object among different video sequences. In our task, we aim to track the same object marked in the first frame for the specific video sequence.

Channel attention networks. Channel attention modules have been ever-increasing popular in different computer vision tasks. A multi-channel attention selection mechanism was proposed in SelectionGAN [33] to refine the coarsely generated one on a target image. A residual channel attention network was designed in RCAN [34] to learn the inter-dependencies of features among channels for image super-resolution. SCA-CNN [35] leveraged channel attention to select semantic attributions of corresponding sentence context. Additionally, Qiu et al. [36] proposed to learn multiple attention maps to obtain hierarchical context information for object detection. All the above

methods show that the attention mechanism can help models learn better representations for corresponding targets. Therefore, we consider using the attention module to help our network learn better feature representation for the target object to be tracked and segmented.

Non-local networks. Non-local operation is mainly treated as a self-attention mechanism to compute the relationships of the pixels through a global view in the network. Wang et al. [9] proposed a non-local operation for capturing long-range dependencies in video classification and static image recognition. DANet [37] plugged non-local operation as position attention module and channel attention module into scene segmentation. In this paper, we introduce the non-local mechanism as a pixel-matching operation to match target pixels and reference pixels to realize the localization of target object in the target frame.

3. Method

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

3.1. Video object segmentation architecture

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

As illustrated in Fig. 2, we provide three different kinds data for the three encoders: the target frame encoder takes the current frame with the estimated labels of the previous frame as 4-channel input [5]; two parameter-shared reference frame encoders take the first frame and the previous frame as input, respectively. Note that when providing data for reference frame encoders, background pixels from the first frame and the previous frame are removed using groundtruth (first frame) or estimated mask (previous frame). Whist for the target frame encoder, background pixels are not masked-out since the masks for the current and previous frames are different. Then, the feature maps of reference and target frames are extracted by respective encoders. In this way, we can obtain the changing object appearance information and target frame features.

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

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

3.2. Non-local pixel matching with channel attention

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

Non-local pixel-matching module. The non-local pixel-matching module is one main module of our NPMCA-net, which is used to obtain object appearance of the target frame and localize the target object simultaneously by matching the feature maps of the reference frames and the target frame. Different from the matching process using convolution layers [29] or using metric learning to pull in similar embedding vectors and push away different embedding vectors [25,28], we directly compute similarities between pixels. The framework of NLPMM is illustrated in Fig. 4(a). The inputs of this module are the feature map of reference frame and the feature map of target frame (defined as $f_{ref} \in \mathbb{R}^{H\times W\times C}$ and $f_{tar} \in \mathbb{R}^{H\times W\times C}$, where H, W, C are the height, width, and channel number, respectively) extracted from respective encoders. In order to reduce memory and improve efficiency for our approach, once feature maps are fed into the module, a 3×3 convolution layer with padding is used to reduce the channel number of input feature maps from C to $C/4$, the new feature maps are with size $f_{ref} \in \mathbb{R}^{H\times W\times \frac{C}{4}}$ and $f_{tar} \in \mathbb{R}^{H\times W\times \frac{C}{4}}$, respectively. After that, the two reduced feature maps are reshaped to $f_{ref} \in \mathbb{R}^{N\times \frac{C}{4}}$ and $f_{tar} \in \mathbb{R}^{N\times \frac{C}{4}}$, where $N = H \times W$. The similarity between pixels in the two feature maps is computed:

$$S = f_{ref} f_{tar}^{T}, \tag{1}$$

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

$$S'(i, j) = \frac{exp(S(i, j))}{\sum_{i=1}^{N} exp(S(i, j))}. \tag{2}$$

With Eqs. (1) and (2), we can generate the relations between any two pixels in the target feature map and the reference feature map. The pixel pair with a large similarity value has high probability belonging to the same pixel of one foreground object. In this case, we can not only match the object appearance but also localize the object. Finally, the new matched feature map $f_{matched}$ is calculated by a matrix multiplication between the transpose of the reduced reference feature map f_{ref} and the non-local similarity map S',

$$f_{matched} = f_{ref}^{T} S'. \tag{3}$$

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

The coarse mask of the target frame can be obtained by the matrix multiplication between the reference feature map and the similarity map, namely, we can use Eq. (3) to obtain the pixels of foreground objects in the target frame. To more intuitively understand the matching and localization process, we show the process in Fig. 3. Fig. 3(a) shows how the similarity map is computed, and Fig. 3(b) displays how the matching process can also accomplish the localization. Therefore,

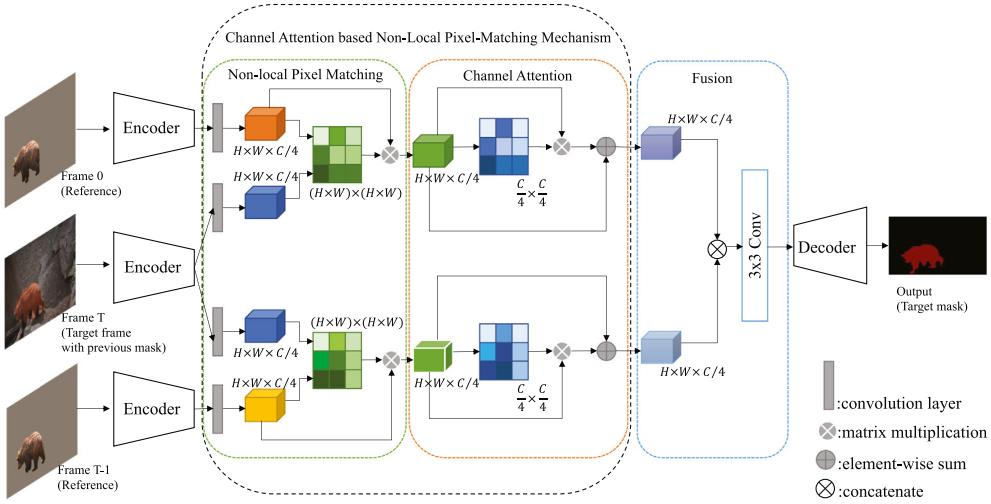
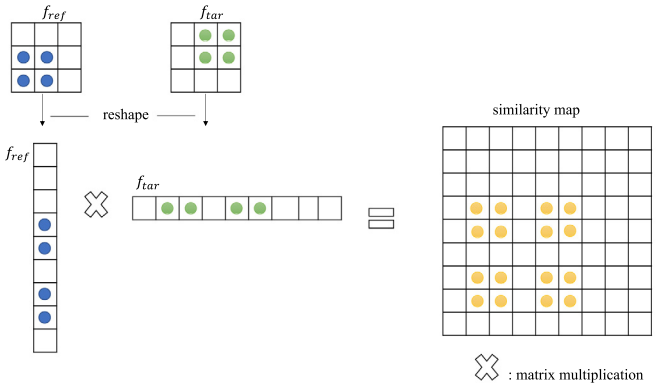
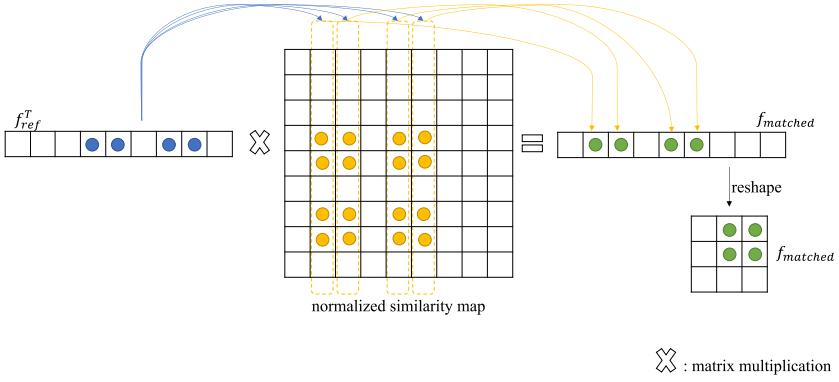


Fig. 2. The framework of our NPMCA-net. It consists of three encoders, where the encoders for the two reference frames are shared. NPMCA-net contains a non-local pixel-matching module, a channel attention module, a fusion module and a decoder.



(a) The Process of Similarity Computation

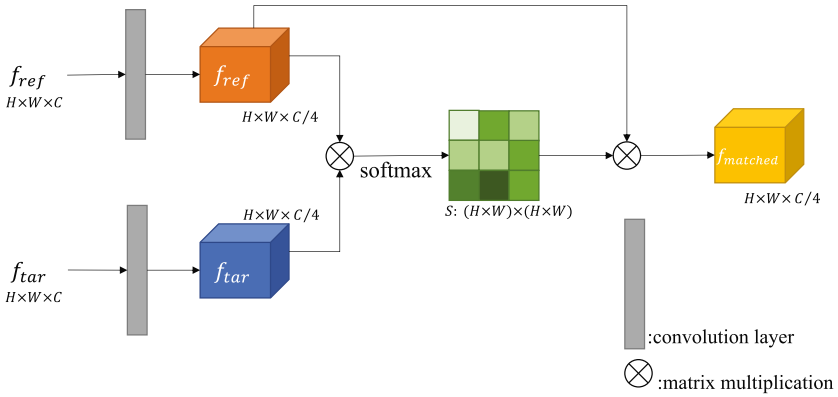


(b) The Process of Matching and Localization

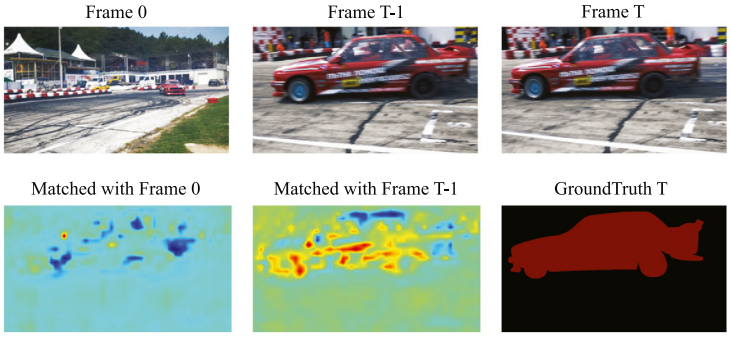
Fig. 3. (a) The process of similarity computation (Eq. (1)). The two reduced feature maps are reshaped into $f_{ref} \in \mathbb{R}^{N\times\frac{C}{4}}$ and $f_{tar} \in \mathbb{R}^{\frac{C}{4}\times N}$, and the similarity is computed by the matrix multiplication. (b) The process of target object matching and localization (Eq. (3)).

we can obtain foreground object appearance and its location at the same time. Besides, visualization of the output of our non-local pixel-matching module is shown in Fig. 4(b). It can be found that this matching module is able to localize the object and mask the target object appearance. The highlighted part (warm color) in the "matched with frame T-1" better demonstrates the matched pixels for the target object. When there is only frame 0 to be referred, it is difficult for the network to find out the pixels for the moving object in the case of large appearance variation.

Channel attention module. We adopt a channel attention module after the non-local pixel-matching module to strengthen the feature representation of foreground object in this task. The details of our channel attention module is illustrated in Fig. 5(a). The input for this module f_{in} is the output feature map of non-local pixel-matching module, *i.e.*, $f_{in} = f_{matched}$ and $f_{in} \in \mathbb{R}^{H\times W\times\frac{C}{4}}$. In order to compute the inter-dependencies between different channels, f_{in} is first reshaped into $f_{in} \in \mathbb{R}^{N\times\frac{C}{4}}$, where $N = H \times W$. Then the channel attention map



(a) Non-Local Pixel-Matching Module



(b) Visualization of Output Feature Map of NLPMM

Fig. 4. (a) Framework of non-local pixel-matching module (NLPMM). Our NLPMM has two inputs, including the reference feature map and the target feature map. The output is the matched feature map. (b) Visualization of output feature map from NLPMM. The matched feature map can coarsely acquire the foreground object appearance and its location.

$A \in \mathbb{R}^{\frac{C}{4}\times\frac{C}{4}}$ is computed by:

$$A = f_{in}^{T} f_{in}, \tag{4}$$

$$A'(i,j) = \frac{exp(A(i,j))}{\sum_{i=1}^{N} exp(A(i,j))}, \tag{5}$$

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

$$f_A = f_{in} A'. \tag{6}$$

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

$$f_{out} = \gamma f_A + f_{in}, \tag{7}$$

where $\gamma \geq 0$ is a learned parameter. We do not apply any convolution layer in the channel attention map. The channel attention map is in series with the non-local pixel-matching module to strengthen the representation of feature map instead of adopting a parallel mode in [37]. Some visualizations of the output feature map of the channel attention module are displayed in Fig. 5(b).

3.3. Two-stage training method

We take two-stage training for our network. Firstly, we pre-train our NPMCA-net through static images. Then, we use the video object segmentation datasets to fine-tune the model. We use IoU loss in [7,38] and Adam [39] optimizer with randomly cropped resolution of (256 × 432) patches for both pre-training and fine-tuning. All experiments are running on one NVIDIA GeForce 2080 Ti GPU.

Pre-training on static images. Pre-training on static images for video object segmentation is becoming popular recently since it can help the network adapt to different foreground object appearance. We follow several successful practice in [4,5,40] to pre-train our network by applying random affine transformation on static images. We use saliency datasets MSRA10K [41], ECSSD [42], segmentation datasets Pascal VOC dataset [43] and COCO [44]. In this case, the network can be adapted to different object appearance and categories, so as to avoid easy over-fitting. For pre-training, we set a fixed learning rate as $1e{-}5$.

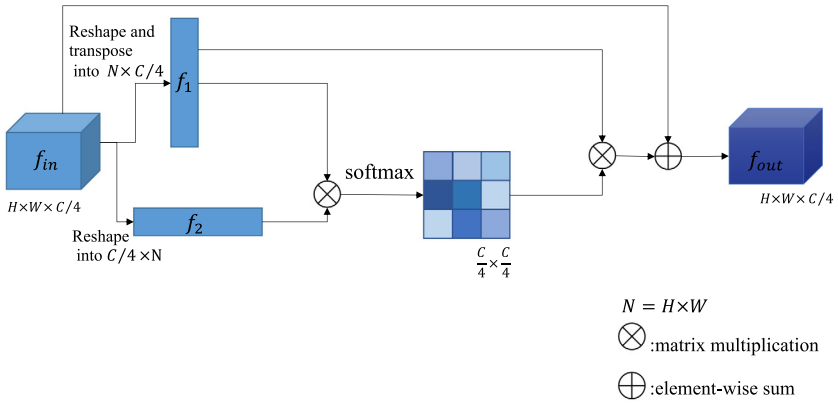
Fine-tuning on videos. Then, we fine-tune the pre-trained model on video object segmentation dataset. We only use DAVIS-17 [45] training set for fine-tuning. During training, we sample three frames in temporal order to obtain temporal information. In order to acquire big variation of object appearance for a long time, we randomly skip frames for sampling. The maximum random skip is 5 and the learning rate for fine-tuning is set as $1e{-}6$.

4. Experiment

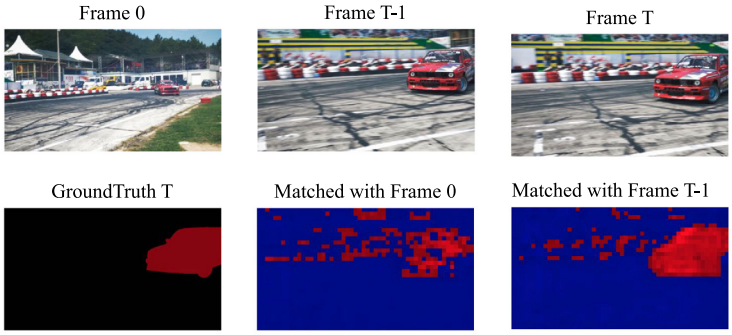
4.1. Inference

Our network is based on the assumption that the ground-truth mask of the first frame is given for semi-supervised video object segmentation. In other words, the first frame is set as the reference frame for all the rest frames. Therefore, to make our network efficient, we only compute the feature map of the first frame once for a test video clip. Following the architecture of our approach, we use previous frame with predicted segmentation mask as another reference frame. We also follow [5] to set three different scale sizes and compute their average as the final output.

Multi-object case. We use softmax aggregation [5] to softly combine multiple objects. Finally, the output probability map is computed



(a) Channel Attention Module



(b) Visualization of Output Feature Map of CM

Fig. 5. (a) Framework of channel attention module (CM). The input of CM is the output of NLPMM (matched feature map), and it outputs the strengthened feature map. (b) Visualization of Output feature map from CM. CM is able to strengthen the feature representation.

by:

$$P_{i,m} = \frac{p_{i,m}/(1-p_{i,m})}{\sum_{j=0}^{M} p_{i,j}/(1-p_{i,j})}, \tag{8}$$

where $p_{i,m}$ is the output probability of instance m at position i. $m = 0$ is for background and M is the total number of instances. We use Eq. (8) to compute the probability map of multi-objects and apply it to next frame inference.

4.2. Implementation

Encoder. We design three encoders based on ResNet-50 [46] for three inputs (two references and one target). Like [5], the target frame encoder takes 4-channel inputs and two reference frame encoders take 3-channel inputs. Instead of using res5 in [5], we take res4 as the final encoded feature map, whose channel number is 1024. This is because the feature map of res5 is with low resolution, making it inaccurate for small objects. On the other hand, three res5 encoders will cause large memory occupation.

Decoder. After the fusion layer, the fused feature map is finally fed into the decoder. Similar to [5], the decoder also takes the encoder stream through skip-connection as input to produce the mask. With the help of skip-connection, the high resolution feature can replenish the missing information. Finally, the feature map is gradually upsampled with a factor of two till it reaches the same size as input.

4.3. Experiment results

We evaluate our network on video object segmentation datasets, DAVIS-2017 [45], DAVIS-2016 [47] and SegTrack-v2 [48]. The evaluation metrics include mean intersection-over-union (IoU) of predicted mask and the ground-truth ($\mathcal{J}$), contour accuracy between contour points on predicted mask and the ground-truth ($\mathcal{F}$), and the average of the two metrics ($\mathcal{J}\&\mathcal{F}$).

DAVIS-2017. DAVIS-2017 is a multi-object dataset. There are 90 videos in total, 60 for training and 30 for validation. We evaluate our method on its validation set. The comparison results with recent state-of-the-art approaches are shown in Table 1. The results are listed from the lowest score of $\mathcal{J}$ to the highest score. The upper part is from approaches with online-learning or with optical flow. It can be found that our method achieves comparable scores with the best performing ones. Our score is slightly lower than PReMVOS [49], but PReMVOS needs longer running time than all other approaches because both online-learning and optical flow need expensive computational cost. We reach the best performance compared with all other methods without online-learning or optical flow. It can be demonstrated that our NLPMM can realize find out where the target object is in current frame. Further, we directly using masked-out object as the input for reference, making our model less sensitive to the influence of backgrounds while focusing on the object itself. By doing this, our method can capture enough object features. Besides, using the masked-out objects of the first frame and the previous frame as references provides enough information for handling appearance variation.

DAVIS-2016. DAVIS-2016 contains 50 videos (30 for training and 20 for validation) for single-object video object segmentation. We report comparison results of the validation set in Table 2. It can be found that our approach achieves better performance than the methods using pixel-matching or metric learning, such as PLM [29], PML [25], FEELVOS [28], and RGMP [5]. We also obtain higher score than other methods without online learning. For metric $\mathcal{J}$, our method is 1.7% higher than STM [40], whist for the contour accuracy, our method is 0.8% lower than STM [40], this might be caused by the adopted IoU loss. Moreover, our results are competitive with online-learning based methods. According to the running time listed in Table 2, our

Table 1
Evaluation on DAVIS-17 validation set. 'OL' denotes online-learning. 'OF' means using optical flow. Our NPMCA-net obtains a score of 3% higher than STM [40].

Method	OL	OF	$\mathcal{J}$ (%)	$\mathcal{F}$ (%)	$\mathcal{J}\&\mathcal{F}$ (%)	Time (s)
OSVOS [1]	✓		56.6	63.9	60.3	10
OnAVOS [3]	✓		61.6	69.1	65.4	13
OSVOS-S [2]	✓		64.7	71.3	68.0	4.5
AGSS-VOS [7]		✓	64.9	69.9	67.4	–
CINN [50]	✓		67.2	74.2	70.7	>120
PReMVOS [49]	✓	✓	**73.9**	**81.8**	**77.8**	–
VideoMatch [26]			56.5	68.2	62.4	0.35
MAARU [51]			61.3	65.3	63.3	0.13
RANet [52]			63.2	68.2	65.7	–
RGMP [5]			64.8	68.6	66.7	0.28
DIPNet [53]			65.3	71.6	68.5	–
A-GAME [6]			67.2	72.7	70.0	–
DMM-Net [18]			68.1	73.3	70.7	–
FEELVOS [28]			69.1	74.0	71.6	0.51
STM [40]			69.2	74.0	71.6	–
TVOS [54]			69.9	74.7	72.3	0.027
NPMCA-net (Ours)			**72.2**	**77.4**	**74.8**	0.25

Table 2
Evaluation on DAVIS-16 validation set. 'OL' denotes online-learning. 'OF' means using optical flow. Our NPMCA-net can even achieve a bit higher performance than methods with online-learning.

Method	OL	OF	$\mathcal{J}$ (%)	$\mathcal{F}$ (%)	$\mathcal{J}\&\mathcal{F}$ (%)	Time (s)
MSK [4]	✓	✓	79.7	75.4	77.6	12
OSVOS [1]	✓		79.8	80.6	80.2	7
MaskRNN [55]	✓	✓	80.7	80.9	80.8	–
CINN [50]	✓		83.4	85.0	84.2	>30
Lucid [56]	✓	✓	83.9	82.0	83.0	–
PReMVOS [49]	✓	✓	84.9	**88.6**	**86.8**	>30
OSVOS-S [2]	✓		85.6	86.4	86.0	4.5
OnAVOS [3]	✓		86.1	84.9	85.5	13
DyeNet [57]	✓		**86.2**	–	–	2.32
PLM [29]			70.0	62.0	66.0	0.3
PML [25]			75.5	79.3	77.4	0.28
VideoMatch [26]			81.0	–	–	0.32
FEELVOS [28]			81.1	82.2	81.7	0.45
RGMP [5]			81.5	82.0	81.8	0.13
A-GAME [6]			82.0	82.2	82.1	0.07
MAARU [51]			83.9	83.8	83.9	0.12
RANet [52]			85.5	85.4	85.5	0.13
DIPNet [53]			85.8	86.4	86.1	0.92
STM [40]			84.8	**88.1**	86.5	0.15
NPMCA-net (Ours)			**86.5**	87.3	**86.9**	0.11

approach can achieve a good balance between accuracy and efficiency. It demonstrates that our NLPMM is able to localize moving objects with masked-out object references. Additionally, pre-training with statistic images also helps network to adapt to different object classes. In this way, our approach does not rely on online training to learn the object information of current video.

SegTrack v2. We also evaluate our network on the SegTrack v2 [48] dataset. The results are shown in Table 3. It can be found that our network also achieve competitive performance on SegTrack v2 dataset under the same level comparison. Therefore, our network has competitive generalization ability. Our performance even defeat MSK [4] and MaskRNN [55], where online training is used. We set the same training dataset as DMM-net. It can be seen that our method can obtain comparable results with DMM-net. However, we obtain lower performance than DyeNet. This phenomenon may be caused by the fact that they use template matching, which predicts bounding box of the target object first then conduct segmentation. In this way, much background noise can be reduced. In the SegTrack v2 dataset, there are several videos with the background very similar to the target object. In such cases, template can better decrease the disturbance of background. However, for other datasets, such as, DAVIS17, DAVIS16, such conditions are not satisfied, the performance of DyeNet is lower than ours, as reported in Tables 1 and 2.

Table 3
Evaluation on SegTrack v2. The IoU performance for the baseline methods are from [5] and [18]. 'OL' denotes online-learning.

Method	OL	IoU (%)
OnAVOS [3]	✓	66.7
MSK [4]	✓	70.3
MaskRNN [55]	✓	72.1
CINN [50]	✓	77.1
Lucid [56]	✓	**77.6**
RGMP [5]		71.1
DIPNet [53]		73.8
DMM-Net [18]		76.7
DyeNet [57]		78.3
NPMCA-net (Ours)		76.1

Table 4
Training methods analysis on DAVIS-2017 validation set. The two-stage training method helps our NPMCA-net better adapt to different categories. With only DAVIS-2017 training set, the network is easy to get over-fitting.

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

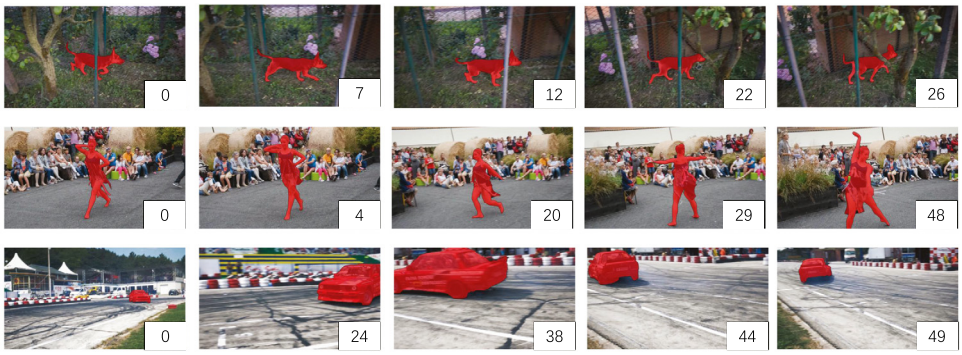
4.4. Qualitative results

Qualitative results on two DAVIS datasets are shown in Fig. 6. For each displayed video, we choose 5 frames with the cases of large object appearance variation or occlusion. It can be found that our model can handle different challenges. For example, our model performs well with large object appearance variation cases like in row 2 and 3 in Fig. 6(a) and row 1 in Fig. 6(b). Besides, our model can also segment each object when they are occluded by background as shown in row 1 in Fig. 6(a) and row 2, 3 in Fig. 6(b). The qualitative comparison between our model and other methods are shown in Fig. 6(c).

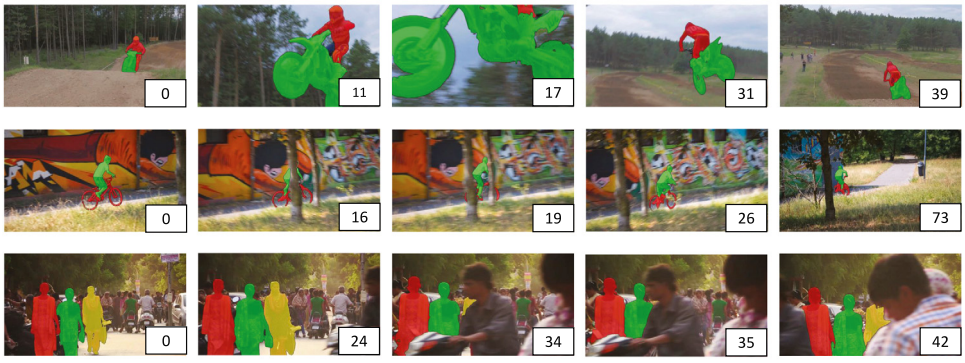
4.5. Ablation studies

Two-stage training method. We firstly conduct the ablation study for the two-stage training method, and the results are displayed in Table 4. It is surprising to find that the performance of pre-train-only case is much better than fine-tune-only case. Both the intersection-over-union score ($\mathcal{J}$) and the contour accuracy ($\mathcal{F}$) of pre-train-only are almost 25% larger than of fine-tuning-only. It proves that two-stage training is necessary. If we only train on DAVIS-2017, the categories are far less enough. It can also be found that our approach will perform better when more categories are used for training. The combination of pre-train and fine-tuning achieves the best performance, because pre-training help our model adapt to large categories and fine-tuning help our model to obtain temporal information and adapt to video sequence.

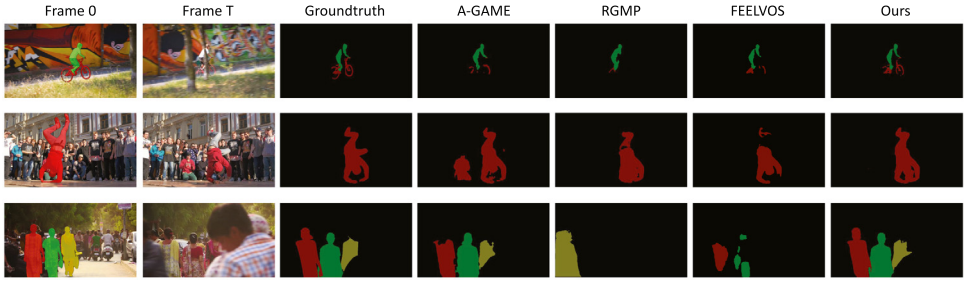
Different modules. We also conduct ablation experiments with some components disabled or removed, and the results are displayed in Table 5. We test three different combinations of the channel attention module and the use of the predicted mask from the previous frame. If we remove our channel attention module, the IoU score and the contour accuracy are 3.4% and 3.7% lower than the full combination, respectively. Therefore, we can conclude that the channel attention module can strengthen the feature representation to help our network better adapt to foreground pixels. On the other hand, if we take out the predicted mask from the previous frame, the IoU score and the contour accuracy are 5.3% and 4.8% lower than the full combination, respectively, which proves that the predicted mask from the previous frame can guide our network to segment the foreground object. Overall, the full NPMCA-net achieves the best performance. It demonstrates that the channel attention module and the use of the predicted mask for the previous frame benefit from each other.



(a) The visual results of our NPMCA-net on DAVIS-2016.



(b) The visual results of our NPMCA-net on DAVIS-2017.



(c) The visual comparison with other approaches on DAVIS-2017.

Fig. 6. We display the frames with large appearance variation or before and after occlusion and the comparison between ours and other approaches.

Table 5
Network module analysis on DAVIS-2017 validation set. 'CM' denotes to the channel attention module, and 'PM' denotes that the input of current frame with the predicted mask from the previous frame.

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

Table 6
Encoder settings analysis on DAVIS-2017 validation set. 'One encoder' denotes to using same encoder for all the inputs 'Two encoders' denotes to the setting of parameter-shared only for the reference frames.

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

Encoder setting. Finally, we conduct the ablation study on the setting of encoders with only training with DAVIS-2017 dataset. We conduct the experiment to show the necessity of the parameter-shared encoder for the two references and different encoder for the target frame. The results is shown in Table 6. 'One encoder' denotes to use same encoder for the three inputs and 'Two encoders' denotes to parameter-shared setting. It can be found that with only one encoder, the result is almost 5% lower than the two-encoder setting. VOS aims to segment the target object from the first frame to the end. To capture consistent reference object feature information, we set parameter-shared encoder for the first frame and previous frame (where background is masked out). Parameter-shared can map the input reference features into the same representation space, thereby the two reference frames' information can be equally treated. Additionally, parameter-shared can reduce parameters for training. If we use just one encoder for the first, the previous and the target frames, the network will be confused, because the encoder for the current frame needs to encode both image and previous predicted mask information, where the background is not masked out. However, for the first and the previous frames, the background is masked out, and we only use the foreground pixels of the frames.

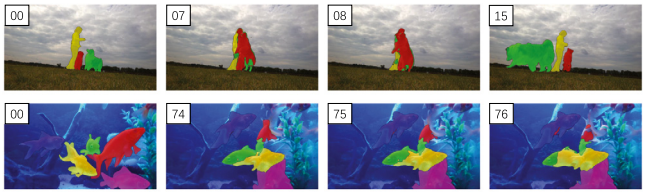


Fig. 7. Limited cases of our network.

4.6. Limitations

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

5. Conclusion

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

CRediT authorship contribution statement

Siyue Yu: Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Investigation. **Jimin Xiao:** Supervision, Writing – review & editing, Funding acquisition. **Bingfeng Zhang:** Writing – review & editing. **Eng Gee Lim:** Supervision, Project administration. **Yao Zhao:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixe, D. Cremers, L. Van Gool, One-shot video object segmentation, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[2] K.-K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, L. Van Gool, Video object segmentation without temporal information, IEEE Trans. Pattern Anal. Mach. Intell. 41 (6) (2018) 1515–1530.

[3] P. Voigtlaender, B. Leibe, Online adaptation of convolutional neural networks for video object segmentation, 2017, arxiv preprint arXiv:1706.09364.

[4] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, A. Sorkine-Hornung, Learning video object segmentation from static images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2663–2672.

[5] S. Wug Oh, J.-Y. Lee, K. Sunkavalli, S. Joo Kim, Fast video object segmentation by reference-guided mask propagation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7376–7385.

[6] J. Johnander, M. Danelljan, E. Brissman, F.S. Khan, M. Felsberg, A generative appearance model for end-to-end video object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 8953–8962.

[7] H. Lin, X. Qi, J. Jia, AGSS-VOS: Attention guided single-shot video object segmentation, in: ICCV, 2019.

[8] J. Sun, D. Yu, Y. Li, C. Wang, Mask propagation network for video object segmentation, 2018, arxiv preprint arXiv:1810.10289.

[9] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[10] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 834–848.

[11] C. Liu, L.-C. Chen, F. Schroff, H. Adam, W. Hua, A.L. Yuille, L. Fei-Fei, Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[12] B. Zhang, J. Xiao, Y. Wei, M. Sun, K. Huang, Reliability does matter: An end-to-end weakly supervised semantic segmentation approach, 2019, arxiv preprint arXiv:1911.08039.

[13] B. Zhang, J. Xiao, T. Qin, Self-guided and cross-guided learning for few-shot segmentation, 2021, arxiv preprint arXiv:2103.16129.

[14] S. Yu, B. Zhang, J. Xiao, E.G. Lim, Structure-consistent weakly supervised salient object detection with local saliency coherence, 2020, arxiv preprint arXiv:2012.04404.

[15] W. Wang, H. Song, S. Zhao, J. Shen, S. Zhao, S.C.H. Hoi, H. Ling, Learning unsupervised video object segmentation through visual attention, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[16] S. Li, B. Seybold, A. Vorobyov, A. Fathi, Q. Huang, C.-C. Jay Kuo, Instance embedding transfer to unsupervised video object segmentation, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[17] S. Li, B. Seybold, A. Vorobyov, X. Lei, C.-C. Jay Kuo, Unsupervised video object segmentation with motion-based bilateral networks, in: The European Conference on Computer Vision (ECCV), 2018.

[18] X. Zeng, R. Liao, L. Gu, Y. Xiong, S. Fidler, R. Urtasun, DMM-Net: Differentiable mask-matching network for video object segmentation, in: The IEEE International Conference on Computer Vision (ICCV), 2019.

[19] M. Sun, J. Xiao, E.G. Lim, B. Zhang, Y. Zhao, Fast template matching and update for video object tracking and segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10791–10799.

[20] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.

[21] M. Sun, J. Xiao, E.G. Lim, Y. Xie, J. Feng, Adaptive ROI generation for video object segmentation using reinforcement learning, Pattern Recognit. 106 (2020) 107465.

[22] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H.O. Song, S. Guadarrama, K.P. Murphy, Semantic instance segmentation via deep metric learning, 2017, arxiv preprint arXiv:1703.10277.

[23] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 815–823.

[24] K. Sohn, Improved deep metric learning with multi-class n-pair loss objective, in: Advances in Neural Information Processing Systems, 2016, pp. 1857–1865.

[25] Y. Chen, J. Pont-Tuset, A. Montes, L. Van Gool, Blazingly fast video object segmentation with pixel-wise metric learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1189–1198.

[26] Y.-T. Hu, J.-B. Huang, A.G. Schwing, Videomatch: Matching based video object segmentation, in: The European Conference on Computer Vision (ECCV), 2018.

[27] H. Ci, C. Wang, Y. Wang, Video object segmentation by learning location-sensitive embeddings, in: The European Conference on Computer Vision (ECCV), 2018.

[28] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, L.-C. Chen, Feelvos: Fast end-to-end embedding learning for video object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9481–9490.

[29] J. Shin Yoon, F. Rameau, J. Kim, S. Lee, S. Shin, I. So Kweon, Pixel-level matching for video object segmentation using convolutional neural networks, in: The IEEE International Conference on Computer Vision (ICCV), 2017.

[30] L. Wang, G. Hua, R. Sukthankar, J. Xue, N. Zheng, Video object discovery and co-segmentation with extremely weak supervision, in: European Conference on Computer Vision, Springer, 2014, pp. 640–655.

[31] W. Wang, J. Shen, X. Li, F. Porikli, Robust video object cosegmentation, IEEE Trans. Image Process. 24 (10) (2015) 3137–3148.

[32] H. Li, F. Meng, Q. Wu, B. Luo, Unsupervised multiclass region cosegmentation via ensemble clustering and energy minimization, IEEE Trans. Circuits Syst. Video Technol. 24 (5) (2013) 789–801.

[33] H. Tang, D. Xu, N. Sebe, Y. Wang, J.J. Corso, Y. Yan, Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2417–2426.

[34] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. Fu, Image super-resolution using very deep residual channel attention networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 286–301.

[35] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, T.-S. Chua, Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5659–5667.

[36] H. Qiu, H. Li, Q. Wu, F. Meng, L. Xu, K.N. Ngan, H. Shi, Hierarchical context features embedding for object detection, IEEE Trans. Multimed. 22 (12) (2020) 3039–3050.

[37] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.

[38] Z. Li, Q. Chen, V. Koltun, Interactive image segmentation with latent diversity, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 577–585.

[39] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arxiv preprint arXiv:1412.6980.

[40] S.W. Oh, J.-Y. Lee, N. Xu, S.J. Kim, Video object segmentation using space-time memory networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 9226–9235.

[41] M.-M. Cheng, N.J. Mitra, X. Huang, P.H. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2014) 569–582.

[42] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1155–1162.

[43] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[44] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: Common objects in context, in: European Conference on Computer Vision, Springer, 2014, pp. 740–755.

[45] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, L. Van Gool, The 2017 davis challenge on video object segmentation, 2017, arxiv preprint arXiv:1704.00675.

[46] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[47] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, A benchmark dataset and evaluation methodology for video object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 724–732.

[48] F. Li, T. Kim, A. Humayun, D. Tsai, J.M. Rehg, Video segmentation by tracking many figure-ground segments, in: ICCV, 2013.

[49] J. Luiten, P. Voigtlaender, B. Leibe, PReMVOS: Proposal-generation, refinement and merging for video object segmentation, in: Asian Conference on Computer Vision, Springer, 2018, pp. 565–580.

[50] L. Bao, B. Wu, W. Liu, CNN in MRF: Video object segmentation via inference in a CNN-based higher-order spatio-temporal MRF, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[51] L. Fu, Y. Zhao, X. Sun, J. Huang, D. Wang, Y. Ding, Video object segmentation based on motion-aware ROI prediction and adaptive reference updating, Expert Syst. Appl. 167 (2021) 114153.

[52] Z. Wang, J. Xu, L. Liu, F. Zhu, L. Shao, Ranet: Ranking attention network for fast video object segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 3978–3987.

[53] P. Hu, J. Liu, G. Wang, V. Ablavsky, K. Saenko, S. Sclaroff, DIPNet: Dynamic identity propagation network for video object segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2020, pp. 1904–1913.

[54] Y. Zhang, Z. Wu, H. Peng, S. Lin, A transductive approach for video object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6949–6958.

[55] Y.-T. Hu, J.-B. Huang, A. Schwing, Maskrnn: Instance level video object segmentation, in: Advances in Neural Information Processing Systems, 2017, pp. 325–334.

[56] A. Khoreva, R. Benenson, E. Ilg, T. Brox, B. Schiele, Lucid data dreaming for object tracking, in: The DAVIS Challenge on Video Object Segmentation, 2017.

[57] X. Li, C. Change Loy, Video object segmentation with joint re-identification and attention-aware mask propagation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 90–105.