Learning Memory Propagation And Matching For Semi-Supervised Video Object Segmentation

Jiale Wang
Beijing Jiaotong University
Hongli Xu (hlxu@bjtu.edu.cn)
Beijing Jiaotong University https://orcid.org/0000-0002-3304-2499
Kexuan Fan
Beijing Jiaotong University
Hui Yin
Beijing Jiaotong University
Longfei Xia
Beijing Jiaotong University

Research Article

Keywords: Video object segmentation, Memory propagation, Multi-object matching, Spatio-temporal retrieval, High frequency refine

Posted Date: January 24th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1218266/v1

Learning memory propagation and matching for Semi-supervised video object segmentation

Jiale Wang[1], Hongli Xu[1*], Kexuan Fan[2], Hui Yin[2] and Longfei Xia[2]

[1*]Key Laboratory of Beijing for Railway Engineering, Beijing Jiaotong University, Haidian District, Beijing, 100044, China.

[2*]Beijing key lab of traffic data analysis and mining, Beijing Jiaotong University, Haidian District, Beijing, 100044, China.

*Corresponding author(s). E-mail(s): hlxu@bjtu.edu.cn;
Contributing authors: 20120416@bjtu.edu.cn; 21125160@bjtu.edu.cn; hyin@bjtu.edu.cn; xialongfei@bjtu.edu.cn;

Abstract

This paper studies the task of semi-supervised video object segmentation (VOS). Multiple works have shown the outstanding performance of the memory retrieval method based on matching, which performs temporal and spatial pixel-level matching, but does not pay attention to the temporal relationship of the frames. To this end, we propose a memory propagation and matching (MPM) method, combining the propagation-based method and matching-based method simultaneously, to reduce some wrong matching and maintain the consistency between adjacent frames and make the model more robust to occlusion and object disappearance and reproduction. Inspired by the remarkable effect of recurrent neural network (RNN) based methods in video tasks, we proposed memory propagation (MP) module which uses Convolution Gate Recurrent Unit (ConvGRU) for memory propagation, and the memory refinement is carried out when the target frame is segmented. At the same time, MPM matches the target frame with the first frame and the previous adjacent frame. The multi-object matching (MOM) module calculates the probability matrix of each pixel belonging to each object, so that the MPM model can effectively distinguish different objects. Experiments show that the MPM model has achieved $\mathcal{J}\&\mathcal{F}$ 82.8% on DAVIS 2017 Validation dataset and $\mathcal{J}\&\mathcal{F}$ 80.1% on YouTube-VOS dataset.

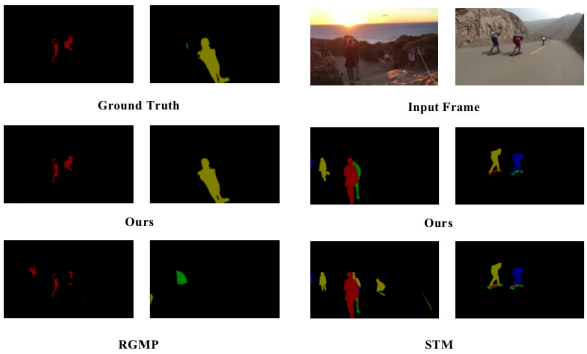
Keywords: Video object segmentation; Memory propagation; Multi-object matching; Spatio-temporal retrieval; High frequency refine

1 Introduction

Video object segmentation is a basic task in the field of computer vision, which is widely used in video editing, video synthesis and automatic driving. This paper focuses on semi-supervised video object segmentation, in which, the subsequent video frames are segmented according to the given mask of the first frame of the sequence. There are many challenges in video object segmentation, such as occlusion, object disappearance and reproduction, large-scale changes and morphological changes of objects. And there may be one or more objects in a video sequence, when there are multiple objects in the video frame, it is easy to confuse the segmentation results of multiple objects.

In recent years, in the task of semi-supervised video object segmentation, there are propagation-based methods(Perazzi et al, 2017; Lin et al, 2019; Yang et al, 2018; Oh et al, 2018) and detection-based methods(Chen et al, 2018b; Seong et al, 2020; Voigtlaender et al, 2019; Caelles et al, 2017; Oh et al, 2019; Voigtlaender and Leibe, 2017; Hu et al, 2018b; Yang et al, 2020). Propagation-based methods propagate the segmentation mask of reference frames to subsequent frames. However, as shown in Fig. 1a, on the left there is an occluded object, and on the right there is a object that disappears and reproduces in the sequence. It can be seen from the figure that the propagation-based method does not perform well in the above two cases. In detection-based methods, there are some methods(Caelles et al, 2017; Voigtlaender and Leibe, 2017) instead of using temporal information, learn an appearance model to detect and segment the object at the pixel level in each frame. During the inference, using the trained model to fine tune the first frame mask, which is effective but is very time-consuming. Among them, the matching-based methods(Son et al, 2015; Chen et al, 2018b; Hu et al, 2018b) generate the template of the object of interest according to the object annotation of the first frame, then match the template with the target frame and guide the segmentation by calculating the similarity matrix. Due to the lack of inter-frame consistency, the false matching in the background and object confusion often occur as show in Fig. 1b. In view of the above problems, we propose a method combining propagation and matching, integrating the two methods to complete the task of semi-supervised video object segmentation.

In the field of unsupervised video object segmentation, LVO(Tokmakov et al, 2017) uses bidirectional ConvGRU(Ballas et al, 2015) to memorize appearance features and optical flow features at the same time. PDB-ConvLSTM(Song et al, 2018) uses bidirectional Convolution Long-Short Term Memory (ConvLSTM) to learn spatio-temporal features. The above two methods have achieved remarkable results in the field of unsupervised VOS, but in semi-supervised VOS, the using of RNN methods does not give full play to their advantages. The existing STM series methods(Hu et al, 2021; Oh et al, 2019; Wang et al, 2021; Xie et al, 2021) that achieve the best results in the field of semi-supervised VOS use the spatio-temporal attention mechanism to perform pixel by pixel global feature matching in the target frame and historical frames. Compared with this method, the memory method based on RNN can memorize features frame by frame so as to track the spatio-temporal consistency of moving objects to a certain extent. Inspired by the above methods, this paper proposes a Memory Propagation (MP) module, ConvGRU is used to remember the features of the historical frames, and the memory features are guided to propagate to the target frame according to the similarity between the previous adjacent frame and the target frame. Experiments show that the MP module proposed in this paper can fully propagate the time clues and guide the segmentation of the target frame.



(a) Bad case in propagation-based methods.

(b) Bad case in matching-based methods.

Fig. 1 Bad case in propagation-based and matching-based methods. The propagation-based method in (a) has poor performance in occlusion and object reproduction. The matching-based method in (b) is prone to similar object in background and confusion of multiple objects.

And considering the severe challenges in VOS task, occlusion and object disappearance and reproduction, we proposes a Multi-object Matching (MOM) module, which firstly matches the object frame with the first frame and the previous adjacent frame, and then uses the feature of the first frame to guide the intra-matching of the target frame based on the accurate object information in the first frame mask. Intra-matching takes the probability into account that each pixel belongs to multiple objects to solve the problem of object confusion as much as possible.

In addition, we also propose a High Frequency Refine (HFR) module to pay attention to the edge information of the object, to make the segmented object edge more complete.

The contributions of this paper are summarized as follows:

- We propose a Memory Propagation module using temporal continuity information to guide the segmentation of the target frame.
- We propose a Multi-object Matching module which combines inter-matching and intra-matching searching target pixel to solve the problems of occlusion and object reproduction.
- We propose a High Frequency Refine module to make the model pay more attention to the edge information to produce higher quality segmentation results.
- Experiments show that the proposed method achieves very competitive results on DAVIS 2017 and YouTube-VOS datasets.

2 Related Works

2.1 Region proposal methods

Inspired by the tasks of image object detection and video object tracking, some methods(Luiten et al, 2019; He et al, 2020; Li and Loy, 2018; Huang et al, 2020) in VOS use the method of proposing candidate regions for segmentation. Some of them(He et al, 2020; Huang et al, 2020) are two-stage training, and some(Li and Loy, 2018) are end-to-end training. DTTM-TAN(Huang et al, 2020) use 3D convolution to extract the features of consecutive frames, and do spatio-temporal aggregation with the features of the target frame. Then generate multiple proposals on the aggregated features, and then match them with the templates in the dynamically updated template bank. PReMVOS(Luiten et al, 2019) uses MASK RCNN(He et al, 2020) to generate coarse mask suggestions, and conducts refinement and re-identification to achieve a high performance. DyeNet(Li and Loy, 2018) uses RPN(Ren et al, 2017) to extract the proposal, and uses RE-ID module to connect the proposal with cyclic mask propagation. In FTMU(Yang and Chan, 2018), reinforcement learning is used to decide which matching-based method to perform on the proposal, matching based on IOU or matching based on appearance. Another role of reinforcement learning is to decide whether to update the template used for matching. The method based on region proposal heavily depends on the pre-trained detector, and multiple thresholds are usually set in pipeline. The model is too complex, and mostly end to end training is not possible.

2.2 Propagation-based methods

The method based on propagation propagates the information of the historical frame to the target frame to assist in the segmentation of the target frame. DIPNet(Hu et al, 2020) decomposes the VOS task into dynamic propagation stage and spatial segmentation stage at each time step. In the dynamic propagation stage, a new object is used to represent the reference information from the adaptive propagation object, which enhances the robustness of video over time. DyeNet(Li and Loy, 2018) integrates template matching into the re-recognition network and integrates FlowNet(Ilg et al, 2017), mask propagation using optical flow information and bidirectional RNN makes its training complex. DTMNet(Zhang et al, 2020) stored the short-term and long-term video sequence information before the target frame as time memory for the purpose of modeling temporal information. The propagation based method can track the temporal continuity of the object well when the object changes smoothly, but it is also prone to false propagation and lack of robustness to the occlusion problem.

2.3 Matching-based methods

The matching based method performs pixel by pixel matching between the reference frames and the target frame. SSM(Zhu et al, 2021) method not only captures the pixel level similarity relationship between the reference frame and the target frame, but also reveals the separable structure of the specified object in the target frame. CFBI(Yang et al, 2020) encodes embedding features from the foreground and background, the matching between reference frames and target frame from pixel level and instance level is conducted, so that CFBI is robust to various object scales. FEELVOS(Chen et al, 2018a) proposed global and local matching according to the distance value. Method based on STM series(Hu et al, 2021; Oh et al, 2019; Wang et al, 2021; Xie et al, 2021) is a memory retrieval method, which is a spatio-temporal matching mechanism between the target frame and many historical frames. This method has achieved very remarkable results in the field of semi-supervised VOS. The matching based method is fast and can well deal

with the problems of occlusion and object disappearance and reproduction, but it is also easy to cause wrong matching due to the lack of temporal continuity information.

2.4 Propagation in similar fields

In some fields similar to VOS task, the use of memory propagation methods is different, but they have achieved quite good results in their fields, which also plays an enlightening role in this paper. In the field of video tracking, RFL(Yang and Chan, 2017) modifies the RNN method to ConvLSTM, which is used to generate a filter for specific objects. MemTrack(Yang and Chan, 2018) proposes a dynamic memory network in the tracking process to make the template adapt to the appearance change of the object. EVS(Paul et al, 2020) propagates optical flow, features and masks frame by frame. LERNet(Wu et al, 2020) mine rich features in the key frame, and calculate the overall attention with the subsequent non-key frame to spread the consistency information across frames in real time.

3 Methods

This section describes the specific design of the MPM model. Firstly, the overall network structure and network processing pipeline are introduced in Section 3.1. Then, the sub-network designs of Memory Propagation module and Multi-object Matching module are introduced in Sections 3.2 and 3.3 respectively. The sub-network design of High Frequency Refine module is described in Section 3.4.

3.1 Overview

The overall network structure is shown in Fig. 2. In this section, we change the target frame into another name, that is query frame. Firstly, the reference frame and query frame are sent to the Encoder (ResNet50(He et al, 2016) is regarded as our backbone, that is, Encoder) for feature extraction, we use the layer 4 output $res4$ for subsequent matching and propagation, $res4 \in \mathbb{R}^{H\times W\times C}$ and both H and W are $\frac{1}{16}$ of the input video frame size.

Then 3×3 convolution is used on encoder to generate feature embedding, $Embedding \in \mathbb{R}^{H\times W\times C/2}$. Then, starting from the first frame, every $Embedding$ is sent into ConvGRUCell. The initial hidden tensor of ConvGRUCell is zero tensor. Then we use ConvGRU to remember $Embedding$ frame by frame until the previous adjacent frame of the query frame. The structure of ConvGRUCell is similar to the LVO(Tokmakov et al, 2017) model. Convolutions in ConvGRUCell are all 5×5 in size. Then, the output of the final hidden layer, the $Embeddings$ of the previous adjacent frame and query frame are sent to MP module for memory propagation.

In addition, the first frame, the previous adjacent frame and the query frame also use 3×3 convolution to generate a $Value$ as the input of MOM module. Then, the three $Values$ are sent to the MOM module for matching.

We also propose the HFM module to make the network pay more attention to the object edge. The input of this module is the $Value$ of the query frame and the result of 1×1 convolution conducts on the encoder output of the query frame.

Finally, the outputs of the three modules are sent to the decoder for generating the final segmentation mask, we also use some skip connections to merge low-level features. The same decoder structure as the STM(Oh et al, 2019) model are used in our MPM model. The convolution weights of the reference frames and query frame are not shared.

3.2 Memory propogation module

The structure diagram of the MP module is shown in Fig. 3. The basic idea of the MP module is to use the similarity between the previous adjacent frame and the query frame to guide memory propagation. We define E_P as the $Embedding$ of the previous adjacent frame, E_Q as the $Embedding$ of the query frame, and h as the output hidden feature of the ConvGRU. Each pixel feature in E_P and E_Q can be regarded as a $C/2$ dimensional feature vector whose size is $H \times W$. Accordingly, similarity matrix is calculated between E_P and E_Q. The similarity matrix is calculated by using the corresponding $C/2$ dimensional feature vector to calculate the cosine similarity, then performs exponential operation $\mathrm{e}\,(x)$ to make it positive, and then it's divided by the maximum value Max to normalize it between 0 and 1. The reason we use the Max operation instead of using the Sigmoid function is that we believes that the relative similarity within the frame is more representative. Then, memory propagation is guided according to the similarity matrix. The calculation

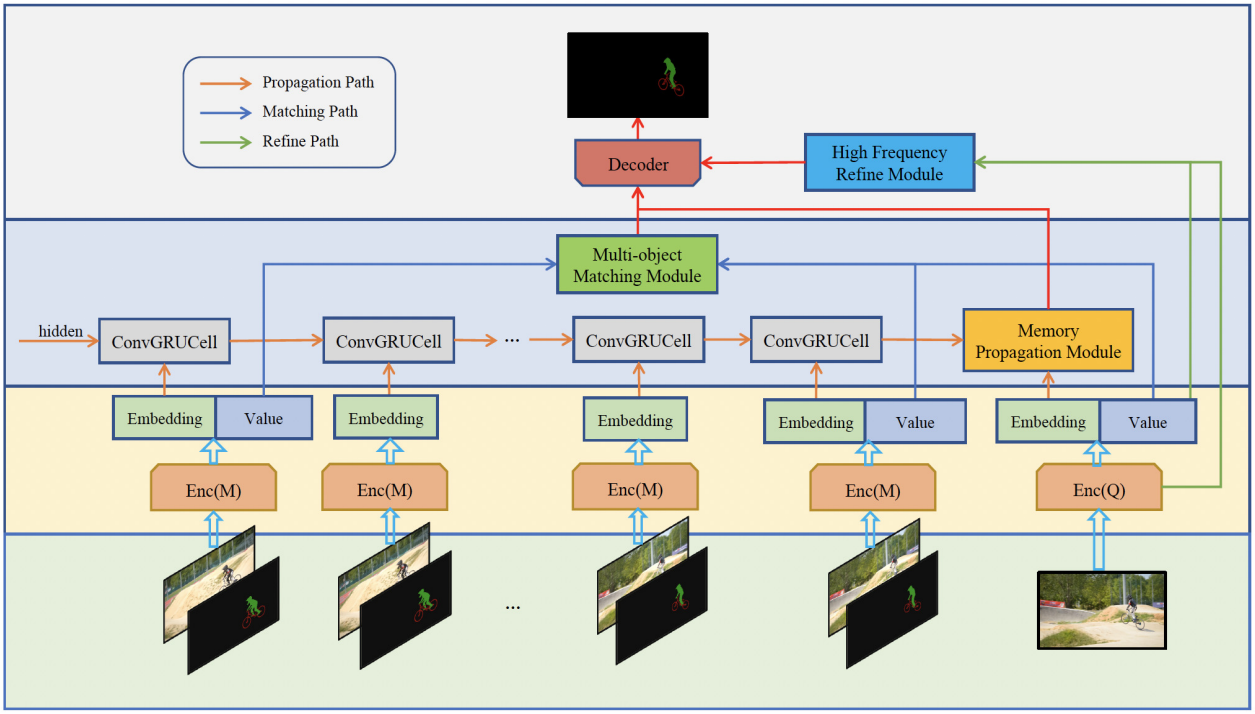


Fig. 2 Architecture overview of MPM. Embedding of past frames is memorized through ConvGRU. Memory Propagation (MP) module guides memory propagation through the similarity between the previous adjacent frame and the query frame. Multi-object Matching (MOM) module conducts spatio-temporal retrieval and intra-frame matching guided by the first frame. High Frequency Refine (HFR) module stands out the high frequency information of the query frame.

of similarity matrix and feature propagation process are defined as Eq. (1) and Eq. (2).

$$S_p(i) = \frac{\exp\left(\text{cosine}(E_P(i), E_Q(i))\right)}{\text{Max}\left(\exp\left(\text{cosine}\left(E_P(i), E_Q(i)\right)\right)\right)} \quad (1)$$

$$F^{MP} = \text{Conv}\left(\text{Concat}\left(S_p * h, (1 - S_p) * E_Q\right)\right) \quad (2)$$

Where Conv stands for convolution operation and Concat stands for concatenation operation, i is pixel index, S_p denotes the similarity matrix between the query frame and the previous adjacent frame, $S_p \in \mathbb{R}^{H \times W}$. F^{MP} is the memory propagation feature output by MP module, $F^{MP} \in \mathbb{R}^{H \times W \times C/2}$. The query frame is more similar to the previous adjacent frame, the weight of memory propagation is greater, The more dissimilar, the weight of memory update is greater. Then, the number of channels is changed to $C/2$ through 1×1 convolution, and then multi-scale feature propagation is carried out through ASPP(Chen et al, 2018a) module, and the output of the ASPP which has $H \times W \times C/2$ dimension is used as the output of this module.

3.3 Multi-object matching module

The MOM module proposed in this paper is divided into two parts. The first part is the Spatio-temporal Retrieval (STR), and the second part is the calculation of Multi-object Matching Probability (MOMP). We will discuss them in 3.3.1 and 3.3.2

3.3.1 Spatio-temporal retrieval

Inspired by STM(Oh et al, 2019), we designed the STR module as shown on the right side of Fig. 4, but we only used the first frame and the previous adjacent frame as the memory. We define V_F as the $Value$ of the first frame, V_P as the $Value$ of the previous adjacent frame and V_Q as the $Value$ of the query frame. Firstly, V_Q and the concatenated V_F and V_P are matched to get the spatial-temporal similarity, and then the Softmax operation is performed in the memory dimension. Then the memory retrieval operation is carried out according to the similarity matrix. Finally, the features of memory retrieval are output. The whole process is defined

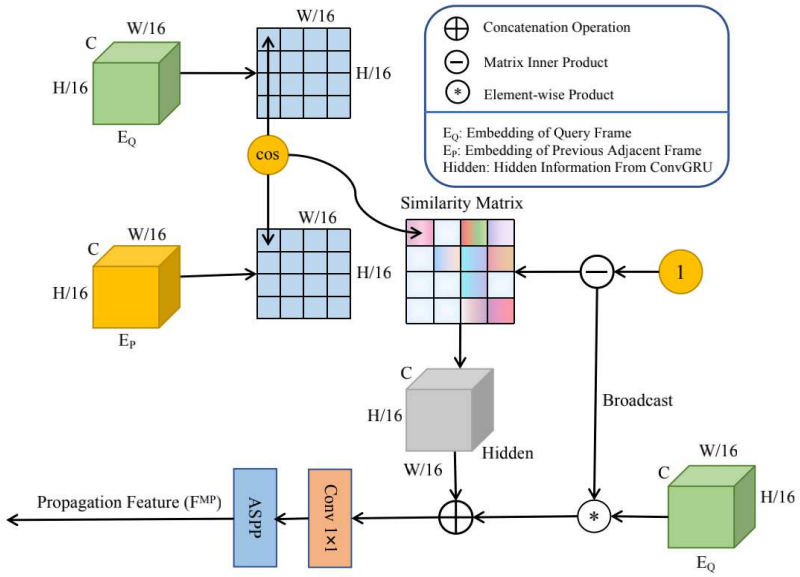


Fig. 3 The structure of Memory Propagation (MP) module.

as Eq. (3) and Eq. (4).

$$S_T(i,j) = \frac{\exp\left(V_M(i) \odot V_Q(j)^{\mathrm{T}}\right)}{\sum_i \exp\left(V_M(i) \odot V_Q(j)^{\mathrm{T}}\right)} \tag{3}$$

$$F^{STR}(j) = \sum_i S_T(i,j) \odot V_M(i) \tag{4}$$

Where i and j are pixel indices, V_M denotes the memory feature (results after concatenation of V_F and V_P), $V_M \in \mathbb{R}^{T\times H\times W\times C/2}$, $T = 2$, $\odot$ denotes matrix inner production, and S_T is the similarity matrix between the query frame and the memory, $S_T \in \mathbb{R}^{THW\times HW}$. F^{STR} is the retrieval feature output by STR module, $F^{STR} \in \mathbb{R}^{H\times W\times C/2}$.

3.3.2 Multi-object matching probability

The MOMP proposed in this paper firstly guide the intra-frame matching according to the inter-frame matching between the first frame and the query frame, and finally calculate the probability that each pixel in the *Value* of the query frame belongs to multiple objects according to the intra-frame matching. The reason why the first frame is selected for inter-frame matching is that the mask of the first frame contains accurate object information. The whole process of Multi-object Matching Probability is shown on the left side of Fig. 4. The process of inter-frame matching is as Eq. (5).

$$S_M(j) = \sum_{fi} V_F(f_i) \odot V_Q(j)^T \tag{5}$$

Where fi and j are the indices, $S_M(j)$ is the sum of the similarity between a pixel of the query frame and all pixels in the first frame, $S_M \in \mathbb{R}^{H\times W}$. According to the ground truth mask of the first frame, filter out the foreground features of the first frame, then calculating the similarity matrix and obtain the summation result.

Then, k query frame feature vectors with the greatest similarity to the foreground pixels in the first frame are selected from V_Q according to S_M, as Eq. (6) shows.

$$V_Q^{topk} = V_Q\left[\mathrm{topk}\left(S_M\right)\right] \tag{6}$$

V_Q^{topk} is the selected K pixel features most similar to the foreground pixels of the first frame, and the size of V_Q^{topk} is $K \times C$.

The process of intra-frame matching is as Eq. (7) and Eq. (8).

$$Prob = \mathrm{Softmax}\left(\frac{\sum_k \left(V_Q^{topk} \odot V_Q\right)}{k}\right) \tag{7}$$

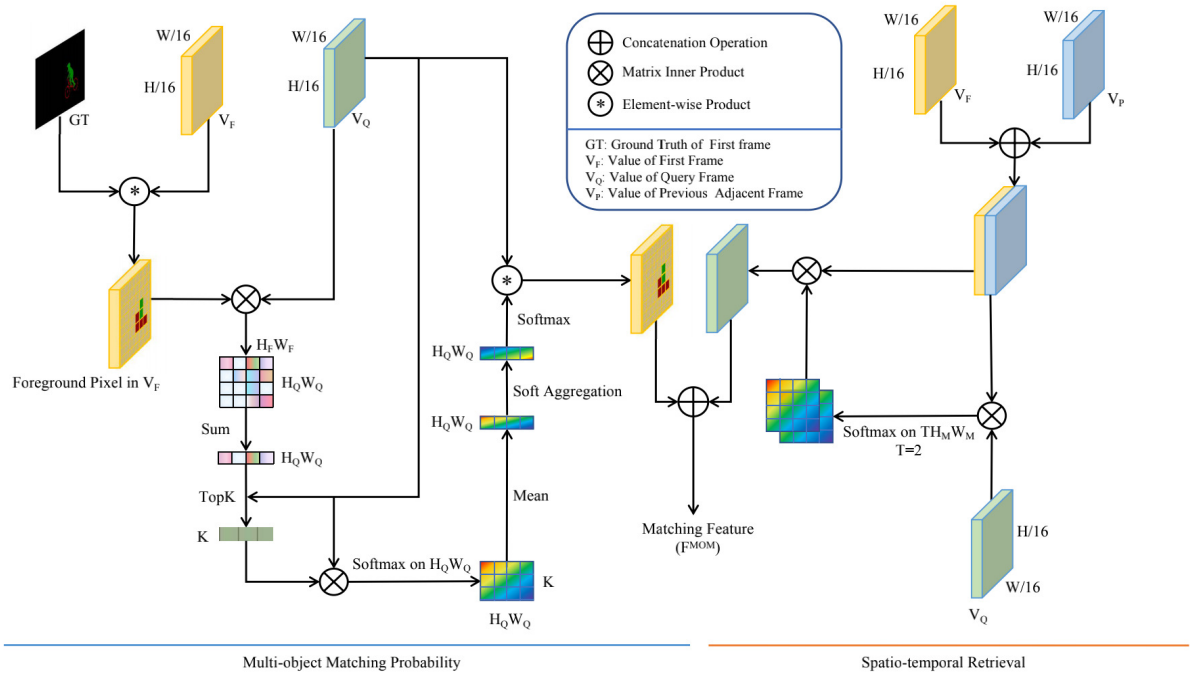


Fig. 4 The structure of Multi-object Matching (MOM) module.

$$F^{MOMP} = Prob * V_Q \tag{8}$$

Similar to inter-frame matching, let k key vectors V_Q^{topk} and V_Q for intra-frame matching. The obtained similarity matrix $Prob$ is averaged at the dimension of K. Then, multiple objects operate in this way, and the resulting $Prob$ matrix is passed through soft aggregation(Oh et al, 2019) calculates the probability of background. Then, the probability of each pixel belonging to each category (including background) is calculated by Softmax. Finally, the probability of multiple objects is taken out as the output (the background probability only helps to calculate the probability that each pixel belongs to multiple objects, and will not be used later). At last, the multi-object probability $Prob$ and the $Value$ of the query frame V_Q are multiplied as a spatial attention operation, and then output this feature F^{MOMP}, $F^{MOMP} \in \mathbb{R}^{H\times W\times C/2}$.

3.4 High fequency refine module

The HFR module is shown in Fig. 5. Inspired by the paper(Xu et al, 2020), this module pays attention to the high frequency information of the query frame to improve the segmentation quality. We apply a 3×3 convolution and a 1×1 convolution respectively on the encoder output of query frame, where the result of 3×3 convolution is the *Embedding* of the query frame shown in Fig. 1.

The processing of this module is as Eq. (9).

$$F^{HFR} = \text{Sigmoid}\left(E_Q^{1\times1} - E_Q^{3\times3}\right) * E_Q^{1\times1} \tag{9}$$

We subtract the result of 3×3 convolution from the result of 1×1 convolution, and then use the Sigmoid function to highlight the edge information of the query frame. Then the result obtained is multiplied by the 1×1 convoluted feature map to obtain the feature F^{HFR} after high frequency refine, $F^{HFR} \in \mathbb{R}^{H\times W\times C/2}$.

4 Experiments

In this section, firstly, the dataset and evaluation metrics are introduced in Section 4.1, the implementation details of the experiment are introduced in Section 4.2, and the ablation study is described in Section 4.3 to illustrate the contribution of different modules proposed in MPM. In section 4.4, we report the evaluation results on benchmarks.

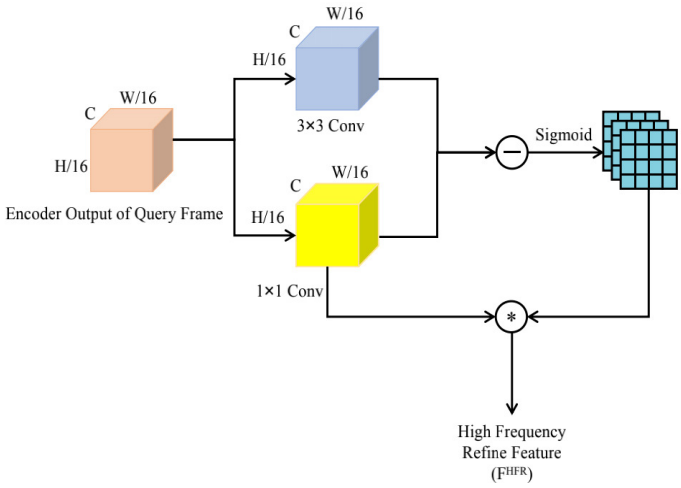


Fig. 5 The structure of High Frequency Refine (HFR) module.

4.1 Datasets and evaluation metrics

DAVIS 2016 and 2017. The DAVIS 2016(Perazzi et al, 2016) dataset contains 50 single object videos, including 3455 annotation frames. Considering its limited size and versatility, it was soon added to the DAVIS 2017(Pont-Tuset et al, 2017) dataset, including 150 sequences and 10459 annotation frames, one of which contains one or more objects. According to the standard of DAVIS dataset, this paper uses the average $\mathcal{J}$ index, the average boundary $\mathcal{F}$ score, and the average $\mathcal{J}\&\mathcal{F}$ to evaluate the accuracy of segmentation. $\mathcal{J}$ score calculates the average Intersection over Union (IoU) between the prediction and the ground truth mask. $\mathcal{F}$ score calculates an average boundary similarity between the boundary of the prediction and the ground truth mask, and $\mathcal{J}\&\mathcal{F}$ is the average of $\mathcal{J}$ and $\mathcal{F}$. In addition, frame rate per second (FPS) is used to measure the segmentation speed.

YouTube-VOS. YouTube-VOS(Xu et al, 2018) is a large video object segmentation dataset, including 4453 videos with multiple object annotations. Its validation set has 474 sequences covering 91 object classes, 26 of which are not seen in the training set. On YouTube-VOS, this paper reports the total score of accuracy evaluation of $\mathcal{J}\&\mathcal{F}$, which is the index average of object classes that have be seen and have not be seen in the training set.

4.2 Training and inference

In this section, we explain training and inference. Multi-object segmentation method is described in 4.2.1, in 4.2.2, we introduce the training strategy.

4.2.1 Multi-object segmentation method

According to the disjoint constraint of multiple objects, that is, each pixel can only belong to one object, this paper establishes a MOM module. This paper adopts the widely used Softmax classifier for classification. In fact, the multi-object processing of the network in this paper is also treated as a single object, but in the MOM module, this paper uses Softmax to calculate the probability that each pixel belongs to multiple objects, and soft aggregation is used to merge the segmentation results of multiple objects similar to Oh et al (2019). When calculating the training loss, we calculates the cross entropy loss of multiple classification, instead of calculating the loss of two categories like Zhu et al (2021); Oh et al (2018); Li et al (2020) etc.

In the VOS task, the number of objects in each video sequence is not uniform. This paper uses the batch advantage of the existing deep learning framework PyTorch to solve this problem. All training is implemented on 1 NVIDIA GeForce RTX 2080 Ti GPU. The test is carried out on 1 NVIDIA GeForce RTX 2080 Ti GPU.

4.2.2 Training strategy

Pre-training on static images. Following SwiftNet(Wang et al, 2021), we firstly pre-train the the network MPM on the 4-frame pseudo video sequence generated on the MS-COCO(Lin et al, 2014) dataset. In the pre-training stage, the input image size is set to 384×384. This paper uses Adam optimizer, and the learning rate starts from 5e-5. The learning rate is adjusted by polynomial scheduling. During training phase, all batch normalization layers in the backbone are fixed at their ImageNet pre-training values. In this paper, the batch size is 4, which is realized on 1 GPU by manual accumulation. The way to generate pseudo video sequences from image dataset is to randomly extract the foreground object from a static image, and then paste it onto a randomly sampled background image to form a new image. In this paper, affine transformations such as rotation, resize, clipping and translation are applied to the foreground and background to generate deformation and occlusion respectively. MPM performs 250K iterations with pseudo video sequences. After pre-training, $\mathcal{J}\&\mathcal{F}$ achieved 75.2% on DAVIS 2017 Validation dataset, which proves the effectiveness of the pre-training.

Fine-tuning on real video sequences. After the pre-training, we conduct 450K iterations on DAVIS 2017 and Youtube-VOS. In each iteration, we randomly samples 4 frames continuously (random skipped frames are less than or equal to 4 frames), and estimates the segmentation mask frame by frame. From the second frame, the segmentation mask of the previous frame (MPM's output) is sent to the network for the segmentation of subsequent frames. At the beginning of training, the maximum number of randomly skipped frames is 4 frames on DAVIS dataset. Due to fast movement, the maximum number of skipped frames on YouTube-VOS dataset is 2 frames. Then, every 20K iterations, the number of random skips is reduced by 1 until the number of random skips is reduced to 0. The reason why this paper adopts the method of decreasing random skip number for training is that the large skipped number in the first 20k iterations is conductive to the training of MOM module. The subsequent random skip number is kept at 0 to ensure the continuity of 4 sampling frames, which is conducive to the training of MP module.

4.3 Ablation study

In this section, we show the results of the ablation study. The ablation study is divided into two parts. The first part is the ablation study about hyperparameter K in the calculation of the Multi-object Matching Probability. The second part is to verify the effectiveness of the sub-module in the whole MPM model. All ablation studies were validated on the DAVIS 2017 Validation dataset.

Table 1 Ablation Study for K in Multi-object Matching Probability on DAVIS 2017 Validation dataset.

K	$\mathcal{J}\&\mathcal{F}\uparrow$(%)	$\mathcal{J}\uparrow$(%)	$\mathcal{F}\uparrow$(%)
8	81.4	78.7	84.0
16	81.9	79.0	84.8
32	**82.8**	**80.3**	**85.2**
48	82.4	79.7	85.1

Hyperparameter. Table 1 shows the verification results of hyperparameter K. When K is 8, K is 16 and K is 48, the $\mathcal{J}\&\mathcal{F}$ index of MPM achieved 81.4%, 81.9% and 82.4% respectively. When K is 32, the $\mathcal{J}\&\mathcal{F}$ index of MPM reached the optimum and achieved 82.8%. Due to memory constraints, we did not conduct ablation study with K greater than 48. We take $K = 32$ as our final MPM model.

Network Sub-module. Table 2 shows the ablation study results of the sub-module in MPM. When the Memory Propagation (MP) module is removed, $\mathcal{J}\&\mathcal{F}$ drops to 80.7% (-2.1%). When the Multi-object Matching (MOM) module is removed, $\mathcal{J}\&\mathcal{F}$ degrades to 77.2%, which decreased by 5.6% compared with the overall network structure(MPM). When the High Frequency Refine (HFR) module is removed, $\mathcal{J}\&\mathcal{F}$ drops to 82.6% (-0.2%). Our MOM module consists of two parts: Multi-object Matching Probability (MOMP) and Spatio-temporal Retrieval (STR). In order to verify the effectiveness of MOMP, we also conducted ablation study on this. When the MOMP is removed, $\mathcal{J}\&\mathcal{F}$ achieved 82.5% (-0.3%). Since STR provides parameter update for MOMJ during training, we did not do the experiment of removing STR alone.

Figure 6 shows some visualization results of removing each module. It can be seen from the camel sequence and loading sequence that removing MP and HFR modules will lead to a large degree of

Table 2 Ablation Study of the network sub-module on DAVIS 2017 Validation dataset.

MP[a]	MOM-MOMP[b]	MOM-STR[c]	HFM[d]	$\mathcal{J}\&\mathcal{F}\uparrow(\%)$	$\mathcal{J}\uparrow(\%)$	$\mathcal{F}\uparrow(\%)$
	✓	✓	✓	80.7	78.2	83.2
✓		✓	✓	82.5	79.6	**85.3**
✓			✓	77.2	74.2	80.3
✓	✓	✓		82.6	79.9	**85.3**
✓	✓	✓	✓	**82.8**	**80.3**	85.2

[a] MP: Memory Propagation module.
[b] MOM-MOMP: Multi-object Matching Probability of Multi-object Matching module.
[c] MOM-STR: Spatio-temporal Retrieval of Multi-object Matching module.
[d] HFM: High Frequency Refine module.

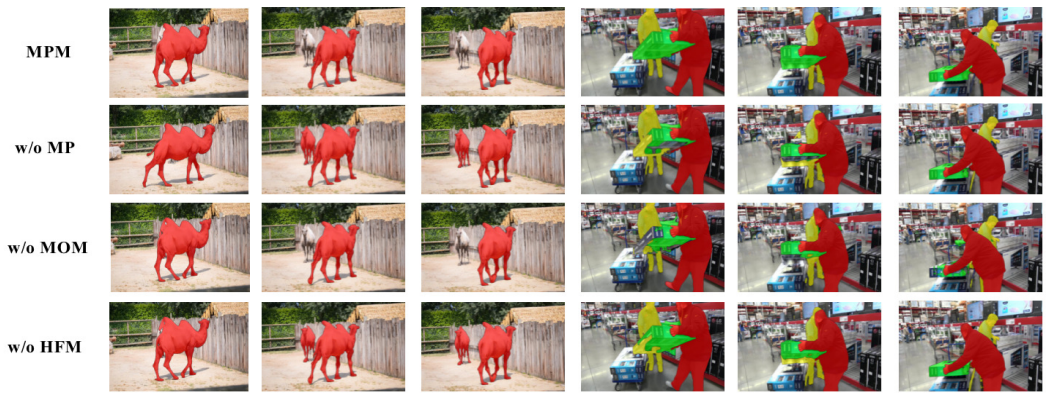


Fig. 6 Visual comparison results of ablation study without network sub-module. The first sequence name is camel, the second sequence name is loading, both sequences are from DAVIS 2017 Validation dataset.

wrong matching. It can also be seen from the loading sequence that removing the MOM module easily leads to incomplete segmentation results.

This ablation study verifies that the Multi-object Matching module makes a great contribution to our MPM model, and other modules also contribute to the final segmentation results.

4.4 Evaluations on benchmarks

DAVIS 2017. As shown in Table 3, our method MPM achieves 82.8% on the DAVIS 2017 Validation dataset, which is same to the previous state-of-the-art model GraphMem(Lu et al, 2020). Compared with other recent methods, MPM also achieves the most advanced performance. STM(Oh et al, 2019) is a memory retrieval method, which needs to sample historical frames to construct memory, resulting in larger memory occupation and slower segmentation time as the segmentation progresses. The segmentation time and memory occupation of our method do not increase with time. In this case, we still exceed STM by 1% on the $\mathcal{J}\&\mathcal{F}$ index. AFB URR(Liang et al, 2020) aiming at the shortcomings of STM, an adaptive feature bank is proposed to dynamically absorb new features and discard obsolete features. This method is not trained with YouTube-VOS sequences, and the $\mathcal{J}\&\mathcal{F}$ index reaches 74.6%. The training result of our MPM model without YouTube-VOS data is 78.0%, which also outperforms AFB URR by 3.4%. GC(Li et al, 2020) also improves the deficiencies of STM. A fixed size feature representation is proposed to replace the using of many previous frames in STM. Although its segmentation speed is fast, the $\mathcal{J}\&\mathcal{F}$ index does not exceed our training results without YouTube-VOS for training (71.4%vs.78.0%). We also reported the segmentation result on DAVIS 2017 Test-dev dataset, and the $\mathcal{J}\&\mathcal{F}$ index also reaches the best result of 75.2%, significantly exceeding STM (+3%). In short, the quantitative results prove that our

Table 3 The quantitative evaluation on DAVIS 2017 dataset.

Method	OL[a]	$\mathcal{J}\&\mathcal{F}\uparrow(\%)$	$\mathcal{J}\uparrow(\%)$	$\mathcal{F}\uparrow(\%)$
Validation Set				
OnAVOS(Voigtlaender and Leibe, 2017)	✓	67.9	64.5	71.3
OSVOS(Caelles et al, 2017)	✓	60.3	56.6	63.9
OSVOS-S(Maninis et al, 2019)	✓	68.0	64.7	71.3
PReMVOS(Luiten et al, 2019)	✓	77.8	73.9	81.8
OSMN(Yang et al, 2018)	×	54.8	52.5	57.1
FAVOS(Cheng et al, 2018)	×	58.2	54.6	61.8
VideoMatch(Hu et al, 2018b)	×	62.4	56.5	68.2
RGMP(Oh et al, 2018)	×	63.2	64.8	68.6
RANet(Wang et al, 2019)	×	65.7	63.2	68.2
A-GAME(Johnander et al, 2019)	×	70.0	67.2	72.7
DMM(Zeng et al, 2019)	×	70.7	68.1	73.3
FEELVOS(Voigtlaender et al, 2019)(+YV)[b]	×	71.6	69.1	74.0
STM(Oh et al, 2019)(+YV)	×	81.8	79.2	84.3
GC(Li et al, 2020)	×	71.4	69.3	73.5
AFB URR(Liang et al, 2020)	×	74.6	73.0	76.1
GraphMem(Lu et al, 2020)(+YV)	×	**82.8**	80.2	**85.2**
MPM(ours)	×	**78.0**	**75.1**	**80.9**
MPM(ours)(+YV)	×	**82.8**	**80.3**	**85.2**
Test-dev Set				
OnAVOS(Voigtlaender and Leibe, 2017)	✓	56.5	53.4	59.6
PReMVOS(Luiten et al, 2019)	✓	71.6	67.5	75.7
OSMN(Yang et al, 2018)	×	41.3	37.7	44.9
RGMP(Oh et al, 2018)	×	52.9	51.3	54.4
RANet(Wang et al, 2019)	×	55.3	53.4	57.2
FEELVOS(Voigtlaender et al, 2019)(+YV)	×	57.8	55.2	60.5
STM(Oh et al, 2019)(+YV)	×	72.2	69.3	75.2
MPM(ours)(+YV)	×	**75.2**	**71.7**	**78.7**

[a] OL denotes online learning method.
[b] (+YV) indicates training with both DAVIS 2017 and Youtube-VOS datasets.

method has achieved competitive results on DAVIS 2017.

DAVIS 2016. As shown in Table 4, since DAVIS 2016 is a single object dataset, the evaluation on this dataset is not affected by the interaction of multiple objects, so its performance highly depends on the accuracy of segmentation details. On this dataset, our method also achieves the optimal results. We also report the frame rate per second (FPS) on this dataset. For fair comparison with other methods, the FPS result is runned on 1 Tesla P100 GPU. Although our FPS index is not as good as RANet(Wang et al, 2019), GC, A-GAME(Johnander et al, 2019), our $\mathcal{J}\&\mathcal{F}$ result (no training on YouTube-VOS) exceeds RANet by 2%. It exceeds GC by 0.9% and A-GAME by 5.4%, and the $\mathcal{J}\&\mathcal{F}$ index exceeds STM by 1% when there is no YouTube-VOS to train, our MPM achieves the same $\mathcal{J}\&\mathcal{F}$ with STM when YouTube-VOS is added to train, but MPM get the faster segmentation speed, which proves that the method in this paper achieves competitive results both in accuracy and speed.

Table 4 The quantitative evaluation on DAVIS 2016 dataset.

Method	OL[a]	$\mathcal{J}\&\mathcal{F}\uparrow(\%)$	$\mathcal{J}\uparrow(\%)$	$\mathcal{F}\uparrow(\%)$	FPS↑[b]
OSVOS(Caelles et al, 2017)	✓	80.2	79.8	80.6	0.11
OnAVOS(Voigtlaender and Leibe, 2017)	✓	85.5	86.1	84.9	0.08
LSE(Ci et al, 2018)	✓	81.6	82.9	80.3	-
PReMVOS(Luiten et al, 2019)	✓	86.8	84.9	88.6	0.03
MaskRNN(Hu et al, 2018a)	×	80.8	80.7	80.9	-
FAVOS(Cheng et al, 2018)	×	81.0	82.4	79.5	0.56
RGMP(Oh et al, 2018)	×	81.8	81.5	82.0	7.69
CINN(Bao et al, 2018)	×	84.2	83.4	85.0	0.03
FEELVOS(Voigtlaender et al, 2019)(+YV)[c]	×	81.7	81.1	82.2	2.22
A-GAME(Johnander et al, 2019)	×	82.1	82.0	82.2	14.29
RANet(Wang et al, 2019)	×	85.5	85.5	85.4	**33.33**
STM(Oh et al, 2019)	×	86.5	84.8	88.1	8.93
STM(Oh et al, 2019)(+YV)	×	**89.3**	88.7	**89.9**	8.93
GC(Li et al, 2020)	×	86.6	**87.6**	85.7	25
MPM(ours)	×	**87.5**	**86.3**	**88.6**	**9.78**
MPM(ours)(+YV)	×	**89.3**	**88.9**	**89.7**	**9.78**

[a] OL denotes online learning method.
[b] The FPS data in this table is from the paper Hu et al (2021).
[c] (+YV) indicates training with both DAVIS 2017 and Youtube-VOS datasets.

Table 5 The quantitative evaluation on YouTube-VOS dataset.

			Seen		Unseen	
Method	OL[a]	$\mathcal{J}\&\mathcal{F}\uparrow(\%)$	$\mathcal{J}\uparrow(\%)$	$\mathcal{F}\uparrow(\%)$	$\mathcal{J}\uparrow(\%)$	$\mathcal{F}\uparrow(\%)$
MSK(Perazzi et al, 2017)	✓	53.1	59.9	59.5	45.0	47.9
OnAVOS(Voigtlaender and Leibe, 2017)	✓	55.2	60.1	62.7	46.6	51.4
OSVOS(Caelles et al, 2017)	✓	58.8	59.8	60.5	54.2	60.7
PReMVOS(Luiten et al, 2019)	✓	66.9	71.4	75.9	56.5	63.7
OSMN(Yang et al, 2018)	×	51.2	60.0	60.1	40.6	44.0
RGMP(Oh et al, 2018)	×	53.8	59.5	-	45.2	-
S2S(Xu et al, 2018)	×	64.4	71.0	70.0	55.5	61.2
RVOS(Ventura et al, 2019)	×	56.8	63.6	67.2	45.5	51.0
DMM(Zeng et al, 2019)	×	58.0	60.3	63.5	50.6	57.4
CapsuleVOS(Duarte et al, 2019)	×	62.3	67.3	53.7	68.1	59.9
A-GAME(Johnander et al, 2019)	×	66.1	67.8	-	60.8	-
STM(Oh et al, 2019)	×	79.4	79.7	84.2	72.8	80.9
AFB URR(Liang et al, 2020)	×	79.6	78.8	83.1	74.1	82.6
GraphMem(Lu et al, 2020)	×	**80.2**	**80.7**	**85.1**	74.0	80.9
MPM(ours)	×	**80.1**	**80.3**	**84.2**	**74.4**	**81.6**

[a] OL denotes online learning method.

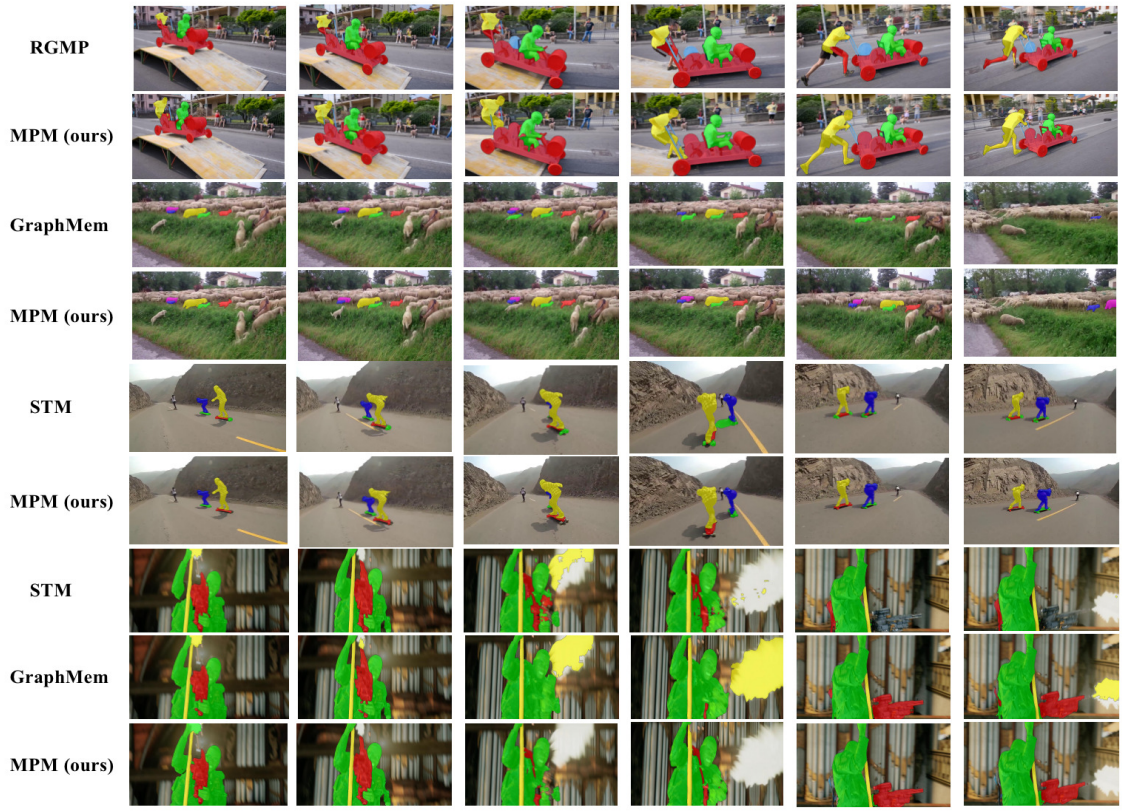


Fig. 7 Visualization of qualitative results compared with state-of-the-art methods. RGMP is a propagation-based method, STM and GraphMem are matching-based methods, and all video sequences are from DAVIS 2017 Validation and YouTube-VOS datasets.

YouTube-VOS. As shown in Table 5, our MPM model also achieves quite good results on YouTube-VOS. The total $\mathcal{J}\&\mathcal{F}$ index reached 80.1%. Compared with GraphMem, our $\mathcal{J}\&\mathcal{F}$ index is 0.1% lower. By comparing the indexes on seen objects and unseen objects with GraphMem, we find that although our MPM method has lower $\mathcal{J}$ index and $\mathcal{F}$ index on seen objects than GraphMem, our method has 0.4% higher $\mathcal{J}$ index and 0.7% higher $\mathcal{F}$ index on unseen objects than GraphMem. This is enough to prove that our method has good generalization performance. In addition, the $\mathcal{J}\&\mathcal{F}$ index of our method exceeds AFB URR by 0.5% and STM by 0.7%. YouTube-VOS is a large-scale video dataset with many kinds of videos. The results obtained by MPM are enough to prove the superiority of this method.

Qualitative results. Figure 7 shows some qualitative results on DAVIS 2017 and YouTube-VOS. The first two lines in the figure are the comparison between MPM and RGMP(Oh et al, 2018). RGMP is a propagation-based method, when object moves rapidly, it is easy to segment incompletely. Our method gets better segmentation results in this case. STM and GraphMem are matching-based methods, compared with STM and GraphMem, the result of STM and GraphMem produce false matching of objects in background and confusion among segmentation results of multiple objects, while our method can reduce some false matching relatively.

5 Conclusion

In this paper, the task of semi-supervised video object segmentation is studied, and a network based on memory propagation and matching (MPM) is proposed,which combines the propagation-based method and matching-based method. The Memory Propagation (MP) module is proposed to propagate time continuity information, and Multi-object

Matching (MOM) module is proposed to solve the problems of occlusion and object disappearance and reproduction. In addition, this paper also proposes a High Frequency Refine (HFR) module to refine the edge, so as to make the segmentation results better. In short, this method achieves competitive performance in the VOS benchmark.

6 Acknowledgments

This work is supported by R&D Program of Beijing Municipal Education Commission(KJZD20191000402) and National Nature Science Foundation of China(51827813, 61472029).

References

Ballas N, Yao L, Pal C, et al (2015) Delving deeper into convolutional networks for learning video representations. arXiv preprint arXiv:151106432

Bao L, Wu B, Liu W (2018) Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5977–5986, https://doi.org/10.1109/CVPR.2018.00626

Caelles S, Maninis KK, Pont-Tuset J, et al (2017) One-shot video object segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 5320–5329, https://doi.org/10.1109/CVPR.2017.565

Chen LC, Papandreou G, Kokkinos I, et al (2018a) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Transactions on Pattern Analysis and Machine Intelligence 40(4):834–848. https://doi.org/10.1109/TPAMI.2017.2699184

Chen Y, Pont-Tuset J, Montes A, et al (2018b) Blazingly fast video object segmentation with pixel-wise metric learning. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1189–1198, https://doi.org/10.1109/CVPR.2018.00130

Cheng J, Tsai YH, Hung WC, et al (2018) Fast and accurate online video object segmentation via tracking parts. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 7415–7424, https://doi.org/10.1109/CVPR.2018.00774

Ci H, Wang C, Wang Y (2018) Video object segmentation by learning location-sensitive embeddings. In: Ferrari V, Hebert M, Sminchisescu C, et al (eds) Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 524–539

Duarte K, Rawat Y, Shah M (2019) Capsulevos: Semi-supervised video object segmentation using capsule routing. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp 8479–8488, https://doi.org/10.1109/ICCV.2019.00857

He K, Zhang X, Ren S, et al (2016) Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 770–778, https://doi.org/10.1109/CVPR.2016.90

He K, Gkioxari G, Dollár P, et al (2020) Mask r-cnn. IEEE Transactions on Pattern Analysis and Machine Intelligence 42(2):386–397. https://doi.org/10.1109/TPAMI.2018.2844175

Hu L, Zhang P, Zhang B, et al (2021) Learning position and target consistency for memory-based video object segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 4144–4154

Hu P, Liu J, Wang G, et al (2020) Dipnet: Dynamic identity propagation network for video object segmentation. In: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), pp 1893–1902, https://doi.org/10.1109/WACV45572.2020.9093333

Hu YT, Huang JB, Schwing AG (2018a) Maskrnn: Instance level video object segmentation. arXiv preprint arXiv:180311187

Hu YT, Huang JB, Schwing AG (2018b) Videomatch: Matching based video object segmentation. In: Ferrari V, Hebert M, Sminchisescu C, et al (eds) Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 56–73

Huang X, Xu J, Tai YW, et al (2020) Fast video object segmentation with temporal aggregation network and dynamic template matching. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 8876–8886, https://doi.org/10.1109/CVPR42600.2020.00890

Ilg E, Mayer N, Saikia T, et al (2017) Flownet 2.0: Evolution of optical flow estimation with deep networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1647–1655, https://doi.org/10.1109/CVPR.2017.179

Johnander J, Danelljan M, Brissman E, et al (2019) A generative appearance model for end-to-end video object segmentation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 8945–8954, https://doi.org/10.1109/CVPR.2019.00916

Li X, Loy CC (2018) Video object segmentation with joint re-identification and attention-aware mask propagation. In: Proceedings of the European conference on computer vision (ECCV), pp 90–105

Li Y, Shen Z, Shan Y (2020) Fast video object segmentation using the global context module. In: Vedaldi A, Bischof H, Brox T, et al (eds) Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp 735–750

Liang Y, Li X, Jafari N, et al (2020) Video object segmentation with adaptive feature bank and uncertain-region refinement. arXiv preprint arXiv:201007958

Lin H, Qi X, Jia J (2019) Agss-vos: Attention guided single-shot video object segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)

Lin TY, Maire M, Belongie S, et al (2014) Microsoft coco: Common objects in context. In: Fleet D, Pajdla T, Schiele B, et al (eds) Computer Vision – ECCV 2014. Springer International Publishing, Cham, pp 740–755

Lu X, Wang W, Danelljan M, et al (2020) Video object segmentation with episodic graph memory networks. In: Vedaldi A, Bischof H, Brox T, et al (eds) Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp 661–679

Luiten J, Voigtlaender P, Leibe B (2019) Premvos: Proposal-generation, refinement and merging for video object segmentation. In: Jawahar C, Li H, Mori G, et al (eds) Computer Vision – ACCV 2018. Springer International Publishing, Cham, pp 565–580

Maninis KK, Caelles S, Chen Y, et al (2019) Video object segmentation without temporal information. IEEE Transactions on Pattern Analysis and Machine Intelligence 41(6):1515–1530. https://doi.org/10.1109/TPAMI.2018.2838670

Oh SW, Lee JY, Sunkavalli K, et al (2018) Fast video object segmentation by reference-guided mask propagation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 7376–7385, https://doi.org/10.1109/CVPR.2018.00770

Oh SW, Lee JY, Xu N, et al (2019) Video object segmentation using space-time memory networks. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp 9225–9234, https://doi.org/10.1109/ICCV.2019.00932

Paul M, Mayer C, Gool LV, et al (2020) Efficient video semantic segmentation with labels propagation and refinement. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 2873–2882

Perazzi F, Pont-Tuset J, McWilliams B, et al (2016) A benchmark dataset and evaluation methodology for video object segmentation. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 724–732, https://doi.org/10.1109/CVPR.2016.85

Perazzi F, Khoreva A, Benenson R, et al (2017) Learning video object segmentation from static images. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 3491–3500, https://doi.org/10.1109/CVPR.2017.372

Pont-Tuset J, Perazzi F, Caelles S, et al (2017) The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:170400675

Ren S, He K, Girshick R, et al (2017) Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 39(6):1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

Seong H, Hyun J, Kim E (2020) Kernelized memory network for video object segmentation. In: Vedaldi A, Bischof H, Brox T, et al (eds) Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp 629–645

Son J, Jung I, Park K, et al (2015) Tracking-by-segmentation with online gradient boosting decision tree. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp 3056–3064, https://doi.org/10.1109/ICCV.2015.350

Song H, Wang W, Zhao S, et al (2018) Pyramid dilated deeper convlstm for video salient object detection. In: Proceedings of the European Conference on Computer Vision (ECCV)

Tokmakov P, Alahari K, Schmid C (2017) Learning video object segmentation with visual memory. In: 2017 IEEE International Conference on Computer Vision (ICCV). IEEE Computer Society, Los Alamitos, CA, USA, pp 4491–4500, https://doi.org/10.1109/ICCV.2017.480, URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2017.480

Ventura C, Bellver M, Girbau A, et al (2019) Rvos: End-to-end recurrent network for video object segmentation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 5272–5281, https://doi.org/10.1109/CVPR.2019.00542

Voigtlaender P, Leibe B (2017) Online adaptation of convolutional neural networks for video object segmentation. arXiv preprint arXiv:170609364

Voigtlaender P, Chai Y, Schroff F, et al (2019) Feelvos: Fast end-to-end embedding learning for video object segmentation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 9473–9482, https://doi.org/10.1109/CVPR.2019.00971

Wang H, Jiang X, Ren H, et al (2021) Swiftnet: Real-time video object segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1296–1305

Wang Z, Xu J, Liu L, et al (2019) Ranet: Ranking attention network for fast video object segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 3978–3987

Wu J, Wen Z, Zhao S, et al (2020) Video semantic segmentation via feature propagation with holistic attention. Pattern Recognition 104:107,268. https://doi.org/https://doi.org/10.1016/j.patcog.2020.107268, URL https://www.sciencedirect.com/science/article/pii/S003132032030073X

Xie H, Yao H, Zhou S, et al (2021) Efficient regional memory network for video object segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1286–1295

Xu K, Yang X, Yin B, et al (2020) Learning to restore low-light images via decomposition-and-enhancement. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 2278–2287, https://doi.org/10.1109/CVPR42600.2020.00235

Xu N, Yang L, Fan Y, et al (2018) Youtube-vos: Sequence-to-sequence video object segmentation. In: Ferrari V, Hebert M, Sminchisescu C, et al (eds) Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 603–619

Yang L, Wang Y, Xiong X, et al (2018) Efficient video object segmentation via network modulation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6499–6507, https://doi.org/10.1109/CVPR.2018.00680

Yang T, Chan AB (2017) Recurrent filter learning for visual tracking. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), pp 2010–2019, https://doi.org/10.1109/ICCVW.2017.235

Yang T, Chan AB (2018) Learning dynamic memory networks for object tracking. In: Ferrari V, Hebert M, Sminchisescu C, et al (eds) Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 153–169

Yang Z, Wei Y, Yang Y (2020) Collaborative video object segmentation by foreground-background integration. In: Vedaldi A, Bischof H, Brox T, et al (eds) Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp 332–348

Zeng X, Liao R, Gu L, et al (2019) Dmm-net: Differentiable mask-matching network for video object segmentation. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp 3928–3937, https://doi.org/10.1109/ICCV.2019.00403

Zhang K, Wang L, Liu D, et al (2020) Dual temporal memory network for efficient video object segmentation. In: Proceedings of the 28th ACM International Conference on Multimedia. Association for Computing Machinery, New York, NY, USA, MM '20, p 1515–1523, https://doi.org/10.1145/3394171.3413942, URL https://doi.org/10.1145/3394171.3413942

Zhu W, Li J, Lu J, et al (2021) Separable structure modeling for semi-supervised video object segmentation. IEEE Transactions on Circuits and Systems for Video Technology pp 1–1. https://doi.org/10.1109/TCSVT.2021.3060015

Declarations

- **Funding** This work is supported by R&D Program of Beijing Municipal Education Commission(KJZD20191000402) and National Nature Science Foundation of China(51827813, 61472029).
- **Competing Interests** The authors have no relevant financial or non-financial interests to disclose.
- **Author Contributions** All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Jiale Wang and Hongli Xu. The algorithm design is mainly completed by Jiale Wang and Hongli Xu. The algorithm tuning was completed by Kexuan Fan, Hui Yin and Longfei Xia. All the authors have carried out experiments, analysis and discussion on this study.The first draft of the manuscript was written by Jiale Wang and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.
- **Data Availability** Public DAVIS benchmark address: https://davischallenge.org/. Public YouTube-VOS benchmark address: https://competitions.codalab.org/competitions/19544#participate. Public MS-COCO dataset address: https://cocodataset.org/#download. The code and results generated during the current study are available in the MPM repository, https://github.com/16301076/MPM.